Basic CAR Model

Alvin Sheng

6/30/2021

```
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 1.3.0 --

## v tibble  3.0.5     v dplyr   1.0.3
## v tidyr   1.1.2     v stringr 1.4.0
## v readr   1.4.0     v forcats 0.5.0
## 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_NC_census_tract.Rmd")
```

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

CAR model results

Inference is based on 3 markov chains, each of which has been run for 100000 samples, the first 10000 of which has been removed for burn-in. The remaining 90000 samples are thinned by 5, resulting in 18000 * 3 = 54000 samples for inference across the 3 Markov chains.

```
load(here("modeling_files/model_3chains_sw_states_var_exclude.RData"))
```

Output for the first chain is shown below.

```
chain1
```

```
##
## #################
## #### Model fitted
## #################
## Likelihood model - Gaussian (identity link function)
## Random effects model - Leroux CAR
## Regression equation - Y ~ X
## <environment: 0x7f81adb99100>
## Number of missing observations - 0
##
## ############
## #### Results
## ############
## Posterior quantities and DIC
##
##                          Median    2.5%   97.5% n.effective Geweke.diag
## (Intercept)              7.5819  7.5701  7.5940     18853.3         1.3
## Xavg_risk_fsf_2020_100  -0.0090 -0.0595  0.0413      5563.6        -0.8
## Xavg_risk_score_sfha     0.0253 -0.0097  0.0585      8236.2         0.3
## Xavg_risk_score_no_sfha -0.0544 -0.1142  0.0036      5990.4        -0.8
## Xpct_floodfactor2        0.0300  0.0024  0.0578      6785.7         1.1
## Xpct_floodfactor3        0.0225 -0.0115  0.0567      7246.4        -0.6
## Xpct_floodfactor4       -0.0108 -0.0470  0.0258      8102.3        -0.4
## Xpct_floodfactor5       -0.0052 -0.0401  0.0290      7752.4         1.3
## Xpct_floodfactor6        0.0599  0.0187  0.1024      6649.9         1.2
## Xpct_floodfactor7       -0.0150 -0.0395  0.0096      8072.7        -1.6
## Xpct_floodfactor8        0.0340  0.0068  0.0611     10011.0         0.5
## Xpct_floodfactor9       -0.0060 -0.0488  0.0353      6397.0         0.9
## Xpct_floodfactor10       0.0857  0.0445  0.1272      6062.7         1.3
## XEP_POV                  0.4088  0.3774  0.4408     12456.6         0.0
## XEP_UNEMP                0.0330  0.0131  0.0528     15957.9        -0.9
## XEP_PCI                 -0.1648 -0.1938 -0.1357     10924.4        -1.5
## XEP_NOHSDP               0.1713  0.1338  0.2085     11952.3        -0.8
## XEP_AGE65                1.8478  1.8155  1.8793      8820.7         0.4
## XEP_AGE17                0.3188  0.2896  0.3482     15361.8         2.3
## XEP_DISABL               0.2621  0.2352  0.2890     13328.2         0.5
## XEP_SNGPNT              -0.0931 -0.1180 -0.0676     16422.5        -3.3
## XEP_MINRTY              -0.0494 -0.0851 -0.0136      7336.2        -0.7
## XEP_LIMENG               0.0130 -0.0208  0.0467      7619.8        -1.6
## XEP_MUNIT               -0.0854 -0.1085 -0.0622     12898.6         1.7
## XEP_MOBILE               0.1526  0.1279  0.1773      9942.2         1.1
## XEP_CROWD               -0.0373 -0.0581 -0.0164     16223.2        -0.1
## XEP_NOVEH                0.1254  0.1001  0.1510     13799.2         0.4
## XEP_GROUPQ              -0.1500 -0.1689 -0.1310     16513.4         1.0
## XEP_UNINSUR              0.0902  0.0637  0.1159     12584.1         1.7
## Xco                      0.0597  0.0239  0.0956      5715.9         2.0
## Xno2                    -0.0780 -0.1423 -0.0105      1720.3        -2.6
## Xo3                     -0.0642 -0.3046  0.1607        50.6         2.2
## Xpm10                   -0.1814 -0.2269 -0.1340       451.6        -3.0
## Xpm25                    0.4767  0.3969  0.5554      1310.1         1.1
## Xso2                     0.0474  0.0144  0.0812      3328.9         0.7
## XData_Value_CSMOKING     0.5910  0.5373  0.6438      6729.2         1.0
## nu2                      0.4806  0.4550  0.5058      2487.4        -0.7
## tau2                     0.9630  0.8607  1.0771      1812.6         0.5
```

```
## rho                        0.9981  0.9917  0.9997       152.6         3.7
##
## DIC =  30523.76       p.d =  3813.566       LMPL =  -15643.28
```

The smallest effective sample size is 935.8, for ozone (o3).

```
chain1$accept
```

```
##      beta       phi       nu2      tau2       rho
## 100.00000 100.00000 100.00000 100.00000  49.97669
```

It appears that beta, phi, nu2, and tau2 probably have Gibbs steps, whereas rho has a Metropolis-Hastings step. In any case, the acceptance probabilities are acceptable.

Model Diagnostics

Beta samples

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

```
saveRDS(beta_samples, file = here("modeling_files/model_3chains_sw_states_var_exclude_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,]       1.00       1.00
## [25,]       1.00       1.00
```

```
## [26,]       1.00       1.00
## [27,]       1.00       1.00
## [28,]       1.00       1.00
## [29,]       1.00       1.00
## [30,]       1.00       1.00
## [31,]       1.00       1.01
## [32,]       1.16       1.47
## [33,]       1.03       1.10
## [34,]       1.00       1.02
## [35,]       1.00       1.00
## [36,]       1.00       1.00
##
## Multivariate psrf
##
## 1.11
```

Examining tau2, nu2, rho

```
tau2_samples <- mcmc.list(chain1$samples$tau2, chain2$samples$tau2,
                          chain3$samples$tau2)

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

rho_samples <- mcmc.list(chain1$samples$rho, chain2$samples$rho,
                          chain3$samples$rho)
```

```
plot(tau2_samples)

plot(nu2_samples)

plot(rho_samples)
```

```
gelman.diag(tau2_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
```

```
gelman.diag(rho_samples)
```

```
## Potential scale reduction factors:
##
##      Point est. Upper C.I.
## [1,]       1.07       1.22
```

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(1:3108, 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,]       1.07       1.21
##  [3,]       1.05       1.17
##  [4,]       1.05       1.18
##  [5,]       1.00       1.01
##  [6,]       1.01       1.04
##  [7,]       1.01       1.02
##  [8,]       1.00       1.01
##  [9,]       1.01       1.03
## [10,]       1.00       1.00
##
## Multivariate psrf
##
## 1.08
```

Inference

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

colnames(beta_samples_matrix) <- colnames(chain1$X)
```

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

```
##                            50%     2.5%    97.5%
## (Intercept)            7.58199  7.56991  7.59408
## Xavg_risk_fsf_2020_100  -0.00913 -0.05964  0.04166
## Xavg_risk_score_sfha     0.02507 -0.00943  0.05925
## Xavg_risk_score_no_sfha -0.05504 -0.11481  0.00429
## Xpct_floodfactor2        0.03005  0.00237  0.05767
## Xpct_floodfactor3        0.02276 -0.01145  0.05686
## Xpct_floodfactor4       -0.01150 -0.04765  0.02481
## Xpct_floodfactor5       -0.00474 -0.03948  0.02942
## Xpct_floodfactor6        0.06034  0.01834  0.10235
## Xpct_floodfactor7       -0.01505 -0.03936  0.00929
## Xpct_floodfactor8        0.03410  0.00709  0.06119
## Xpct_floodfactor9       -0.00601 -0.04809  0.03587
```

```
## Xpct_floodfactor10       0.08602  0.04485  0.12748
## XEP_POV                  0.40899  0.37734  0.44090
## XEP_UNEMP                0.03302  0.01280  0.05298
## XEP_PCI                 -0.16491 -0.19401 -0.13586
## XEP_NOHSDP               0.17124  0.13429  0.20838
## XEP_AGE65                1.84760  1.81544  1.87931
## XEP_AGE17                0.31887  0.28975  0.34806
## XEP_DISABL               0.26224  0.23559  0.28885
## XEP_SNGPNT              -0.09324 -0.11864 -0.06805
## XEP_MINRTY              -0.04942 -0.08509 -0.01314
## XEP_LIMENG               0.01289 -0.02086  0.04646
## XEP_MUNIT               -0.08538 -0.10870 -0.06214
## XEP_MOBILE               0.15290  0.12806  0.17726
## XEP_CROWD               -0.03733 -0.05803 -0.01640
## XEP_NOVEH                0.12526  0.09984  0.15076
## XEP_GROUPQ              -0.14998 -0.16877 -0.13099
## XEP_UNINSUR              0.09022  0.06409  0.11619
## Xco                      0.06005  0.02419  0.09584
## Xno2                    -0.07817 -0.14319 -0.01202
## Xo3                     -0.05466 -0.30651  0.21225
## Xpm10                   -0.18242 -0.22929 -0.13521
## Xpm25                    0.47814  0.39824  0.55656
## Xso2                     0.04735  0.01422  0.08041
## XData_Value_CSMOKING     0.59079  0.53713  0.64387
```

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?

```
row.names(beta_inference)
```

```
##  [1] "(Intercept)"              "Xavg_risk_fsf_2020_100"
##  [3] "Xavg_risk_score_sfha"     "Xavg_risk_score_no_sfha"
##  [5] "Xpct_floodfactor2"        "Xpct_floodfactor3"
##  [7] "Xpct_floodfactor4"        "Xpct_floodfactor5"
##  [9] "Xpct_floodfactor6"        "Xpct_floodfactor7"
## [11] "Xpct_floodfactor8"        "Xpct_floodfactor9"
## [13] "Xpct_floodfactor10"       "XEP_POV"
## [15] "XEP_UNEMP"                "XEP_PCI"
## [17] "XEP_NOHSDP"               "XEP_AGE65"
## [19] "XEP_AGE17"                "XEP_DISABL"
## [21] "XEP_SNGPNT"               "XEP_MINRTY"
## [23] "XEP_LIMENG"               "XEP_MUNIT"
## [25] "XEP_MOBILE"               "XEP_CROWD"
## [27] "XEP_NOVEH"                "XEP_GROUPQ"
## [29] "XEP_UNINSUR"              "Xco"
## [31] "Xno2"                     "Xo3"
## [33] "Xpm10"                    "Xpm25"
## [35] "Xso2"                     "XData_Value_CSMOKING"
```

List of significant beta coefficients:

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

```
##  [1] "(Intercept)"         "Xpct_floodfactor2"    "Xpct_floodfactor6"
##  [4] "Xpct_floodfactor8"   "Xpct_floodfactor10"   "XEP_POV"
##  [7] "XEP_UNEMP"           "XEP_PCI"              "XEP_NOHSDP"
```

```
## [10] "XEP_AGE65"             "XEP_AGE17"             "XEP_DISABL"
## [13] "XEP_SNGPNT"            "XEP_MINRTY"            "XEP_MUNIT"
## [16] "XEP_MOBILE"            "XEP_CROWD"             "XEP_NOVEH"
## [19] "XEP_GROUPQ"            "XEP_UNINSUR"           "Xco"
## [22] "Xno2"                  "Xpm10"                 "Xpm25"
## [25] "Xso2"                  "XData_Value_CSMOKING"
```

Presenting the Results

Retrieving the standard deviation used to scale the covariates

```r
fhs_model_df <- readRDS(here("intermediary_data/fhs_model_df_sw_states_census_tract.rds"))
```

```r
X <- fhs_model_df[, 14:(ncol(fhs_model_df) - 1)]

X <- X[, names(X) != "pct_floodfactor1"]

X <- as.matrix(X)




X           <- scale(X) # Scale covariates
# X[is.na(X)] <- 0        # Fill in missing values with the mean
```

```r
(covariate_sds <- attr(X, "scaled:scale"))
```

```
##     pct_fs_risk_2020_5     pct_fs_risk_2050_5   pct_fs_risk_2020_100
##           9.007959e+00           1.265018e+01           1.832829e+01
##   pct_fs_risk_2050_100   pct_fs_risk_2020_500   pct_fs_risk_2050_500
##           2.032002e+01           2.189616e+01           2.475486e+01
##     avg_risk_score_all    avg_risk_score_2_10   avg_risk_fsf_2020_100
##           1.411278e+00           1.150522e+00           8.710923e-01
##  avg_risk_fsf_2020_500    avg_risk_score_sfha avg_risk_score_no_sfha
##           1.036067e+00           1.679216e+00           1.078335e+00
##       pct_floodfactor2       pct_floodfactor3       pct_floodfactor4
##           4.767530e+00           4.066078e+00           3.640777e+00
##       pct_floodfactor5       pct_floodfactor6       pct_floodfactor7
##           2.912124e+00           9.782551e+00           2.021735e+00
##       pct_floodfactor8       pct_floodfactor9      pct_floodfactor10
##           6.387504e-01           5.704575e+00           7.812772e+00
##                 EP_POV               EP_UNEMP                 EP_PCI
##           1.175289e+01           4.654859e+00           1.500430e+04
##              EP_NOHSDP               EP_AGE65               EP_AGE17
##           8.789647e+00           1.019053e+01           6.606354e+00
##              EP_DISABL              EP_SNGPNT              EP_MINRTY
##           5.980646e+00           6.191460e+00           2.804633e+01
##              EP_LIMENG               EP_MUNIT              EP_MOBILE
##           6.129216e+00           1.778989e+01           1.488064e+01
##               EP_CROWD               EP_NOVEH              EP_GROUPQ
##           2.863150e+00           7.268094e+00           9.488092e+00
##             EP_UNINSUR                     co                    no2
##           6.794178e+00           3.182961e-02           1.936575e+00
##                     o3                   pm10                   pm25
##           4.771631e+00           3.510770e+00           1.117311e+00
```

```
##                  so2    Data_Value_CSMOKING
##         2.944090e-01           5.536335e+00
```

Rescaling beta_inference (except the intercept)

```
beta_inference_scaled <- beta_inference

for (i in 2:nrow(beta_inference)) {

  beta_inference_scaled[i, ] <- beta_inference_scaled[i, ] / covariate_sds[i - 1]

}
```

```
beta_inference_scaled
```

```
##                                 50%          2.5%         97.5%
## (Intercept)            7.581990e+00  7.569910e+00  7.594080e+00
## Xavg_risk_fsf_2020_100 -1.013548e-03 -6.620812e-03  4.624799e-03
## Xavg_risk_score_sfha    1.981790e-03 -7.454440e-04  4.683728e-03
## Xavg_risk_score_no_sfha -3.003008e-03 -6.264087e-03  2.340644e-04
## Xpct_floodfactor2       1.478837e-03  1.166337e-04  2.838087e-03
## Xpct_floodfactor3       1.039452e-03 -5.229227e-04  2.596802e-03
## Xpct_floodfactor4      -4.645553e-04 -1.924875e-03  1.002228e-03
## Xpct_floodfactor5      -3.358659e-03 -2.797465e-02  2.084636e-02
## Xpct_floodfactor6       5.244577e-02  1.594059e-02  8.895963e-02
## Xpct_floodfactor7      -1.727716e-02 -4.518465e-02  1.066477e-02
## Xpct_floodfactor8       3.291294e-02  6.843190e-03  5.905991e-02
## Xpct_floodfactor9      -3.579051e-03 -2.863836e-02  2.136116e-02
## Xpct_floodfactor10      7.977116e-02  4.159192e-02  1.182193e-01
## XEP_POV                 8.578656e-02  7.914791e-02  9.247976e-02
## XEP_UNEMP               8.120847e-03  3.147997e-03  1.302975e-02
## XEP_PCI                -4.529528e-02 -5.328808e-02 -3.731621e-02
## XEP_NOHSDP              5.880244e-02  4.611411e-02  7.155602e-02
## XEP_AGE65               1.888669e-01  1.855794e-01  1.921084e-01
## XEP_AGE17               1.577210e-01  1.433175e-01  1.721591e-01
## XEP_DISABL              4.105516e-01  3.688295e-01  4.522111e-01
## XEP_SNGPNT             -1.634478e-02 -2.079734e-02 -1.192902e-02
## XEP_MINRTY             -6.325540e-03 -1.089114e-02 -1.681862e-03
## XEP_LIMENG              1.096752e-03 -1.774883e-03  3.953071e-03
## XEP_MUNIT              -1.834212e-02 -2.335194e-02 -1.334949e-02
## XEP_MOBILE              1.019041e-05  8.534884e-06  1.181394e-05
## XEP_CROWD              -4.247042e-03 -6.602085e-03 -1.865831e-03
## XEP_NOVEH               1.229181e-02  9.797335e-03  1.479413e-02
## XEP_GROUPQ             -2.270239e-02 -2.554662e-02 -1.982788e-02
## XEP_UNINSUR             1.508533e-02  1.071623e-02  1.942767e-02
## Xco                     9.698843e-03  3.906994e-03  1.547939e-02
## Xno2                   -2.787174e-03 -5.105480e-03 -4.285765e-04
## Xo3                    -8.917943e-03 -5.000803e-02  3.462923e-02
## Xpm10                  -1.025414e-02 -1.288878e-02 -7.600384e-03
## Xpm25                   3.213168e-02  2.676229e-02  3.740161e-02
## Xso2                    1.653773e-02  4.966558e-03  2.808445e-02
## XData_Value_CSMOKING    8.128540e-02  7.390245e-02  8.858856e-02
```

Showing just the results for the significant coefficients

```
beta_inference_scaled[sign(beta_inference_scaled[, 2]) == sign(beta_inference_scaled[, 3]), ]
```

```
##                             50%          2.5%         97.5%
## (Intercept)          7.581990e+00  7.569910e+00  7.594080e+00
## Xpct_floodfactor2     1.478837e-03  1.166337e-04  2.838087e-03
## Xpct_floodfactor6     5.244577e-02  1.594059e-02  8.895963e-02
## Xpct_floodfactor8     3.291294e-02  6.843190e-03  5.905991e-02
## Xpct_floodfactor10    7.977116e-02  4.159192e-02  1.182193e-01
## XEP_POV               8.578656e-02  7.914791e-02  9.247976e-02
## XEP_UNEMP             8.120847e-03  3.147997e-03  1.302975e-02
## XEP_PCI              -4.529528e-02 -5.328808e-02 -3.731621e-02
## XEP_NOHSDP            5.880244e-02  4.611411e-02  7.155602e-02
## XEP_AGE65             1.888669e-01  1.855794e-01  1.921084e-01
## XEP_AGE17             1.577210e-01  1.433175e-01  1.721591e-01
## XEP_DISABL            4.105516e-01  3.688295e-01  4.522111e-01
## XEP_SNGPNT           -1.634478e-02 -2.079734e-02 -1.192902e-02
## XEP_MINRTY           -6.325540e-03 -1.089114e-02 -1.681862e-03
## XEP_MUNIT            -1.834212e-02 -2.335194e-02 -1.334949e-02
## XEP_MOBILE            1.019041e-05  8.534884e-06  1.181394e-05
## XEP_CROWD            -4.247042e-03 -6.602085e-03 -1.865831e-03
## XEP_NOVEH             1.229181e-02  9.797335e-03  1.479413e-02
## XEP_GROUPQ           -2.270239e-02 -2.554662e-02 -1.982788e-02
## XEP_UNINSUR           1.508533e-02  1.071623e-02  1.942767e-02
## Xco                   9.698843e-03  3.906994e-03  1.547939e-02
## Xno2                 -2.787174e-03 -5.105480e-03 -4.285765e-04
## Xpm10                -1.025414e-02 -1.288878e-02 -7.600384e-03
## Xpm25                 3.213168e-02  2.676229e-02  3.740161e-02
## Xso2                  1.653773e-02  4.966558e-03  2.808445e-02
## XData_Value_CSMOKING  8.128540e-02  7.390245e-02  8.858856e-02
```

```r
round(beta_inference_scaled[sign(beta_inference_scaled[, 2]) == sign(beta_inference_scaled[, 3]), ], di
```

```
##                         50%   2.5%  97.5%
## (Intercept)           7.582  7.570  7.594
## Xpct_floodfactor2     0.001  0.000  0.003
## Xpct_floodfactor6     0.052  0.016  0.089
## Xpct_floodfactor8     0.033  0.007  0.059
## Xpct_floodfactor10    0.080  0.042  0.118
## XEP_POV               0.086  0.079  0.092
## XEP_UNEMP             0.008  0.003  0.013
## XEP_PCI              -0.045 -0.053 -0.037
## XEP_NOHSDP            0.059  0.046  0.072
## XEP_AGE65             0.189  0.186  0.192
## XEP_AGE17             0.158  0.143  0.172
## XEP_DISABL            0.411  0.369  0.452
## XEP_SNGPNT           -0.016 -0.021 -0.012
## XEP_MINRTY           -0.006 -0.011 -0.002
## XEP_MUNIT            -0.018 -0.023 -0.013
## XEP_MOBILE            0.000  0.000  0.000
## XEP_CROWD            -0.004 -0.007 -0.002
## XEP_NOVEH             0.012  0.010  0.015
## XEP_GROUPQ           -0.023 -0.026 -0.020
## XEP_UNINSUR           0.015  0.011  0.019
## Xco                   0.010  0.004  0.015
## Xno2                 -0.003 -0.005  0.000
## Xpm10                -0.010 -0.013 -0.008
## Xpm25                 0.032  0.027  0.037
```

```
## Xso2                    0.017  0.005  0.028
## XData_Value_CSMOKING   0.081  0.074  0.089
```

Multiplying by factor of 10, for more interpretability

```
round((beta_inference_scaled * 10)[sign(beta_inference_scaled[, 2]) == sign(beta_inference_scaled[, 3])
```

```
##                         50%  2.5% 97.5%
## (Intercept)           75.82 75.70 75.94
## Xpct_floodfactor2      0.01  0.00  0.03
## Xpct_floodfactor6      0.52  0.16  0.89
## Xpct_floodfactor8      0.33  0.07  0.59
## Xpct_floodfactor10     0.80  0.42  1.18
## XEP_POV                0.86  0.79  0.92
## XEP_UNEMP              0.08  0.03  0.13
## XEP_PCI               -0.45 -0.53 -0.37
## XEP_NOHSDP             0.59  0.46  0.72
## XEP_AGE65              1.89  1.86  1.92
## XEP_AGE17              1.58  1.43  1.72
## XEP_DISABL             4.11  3.69  4.52
## XEP_SNGPNT            -0.16 -0.21 -0.12
## XEP_MINRTY            -0.06 -0.11 -0.02
## XEP_MUNIT             -0.18 -0.23 -0.13
## XEP_MOBILE             0.00  0.00  0.00
## XEP_CROWD             -0.04 -0.07 -0.02
## XEP_NOVEH              0.12  0.10  0.15
## XEP_GROUPQ            -0.23 -0.26 -0.20
## XEP_UNINSUR            0.15  0.11  0.19
## Xco                    0.10  0.04  0.15
## Xno2                  -0.03 -0.05  0.00
## Xpm10                 -0.10 -0.13 -0.08
## Xpm25                  0.32  0.27  0.37
## Xso2                   0.17  0.05  0.28
## XData_Value_CSMOKING   0.81  0.74  0.89
```

Boxplots for the posterior distribution of coefficients

Scaling the posterior distributions of coefficients

```
beta_samples_scaled <- beta_samples_matrix

for (j in 2:ncol(beta_samples_matrix)) {

  beta_samples_scaled[, j] <- beta_samples_scaled[, j] / covariate_sds[j - 1]

}
```

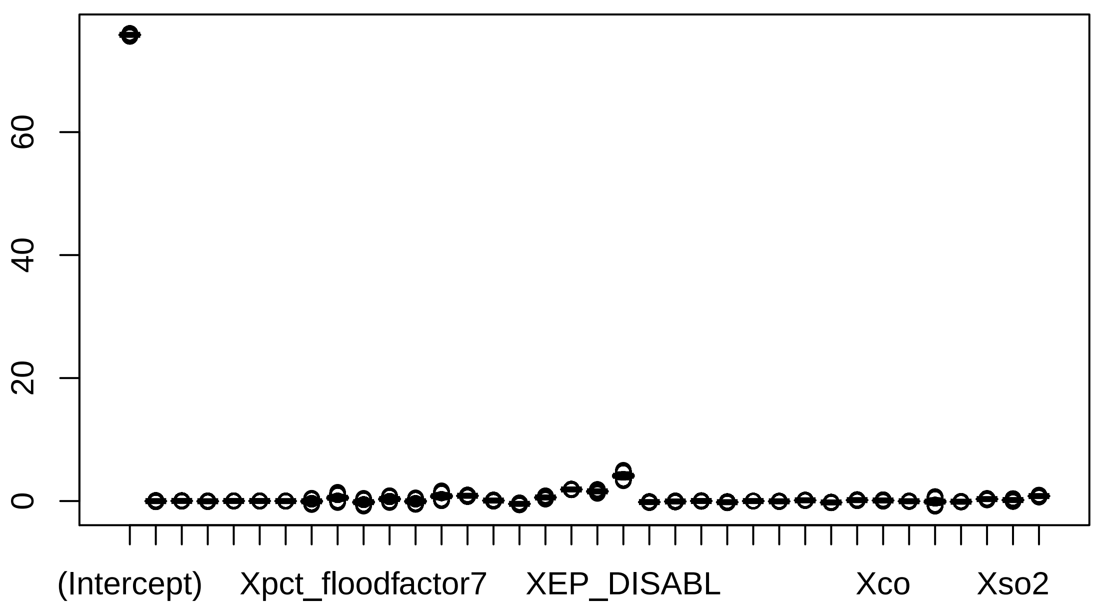
```
beta_samples_scaled_10 <- beta_samples_scaled * 10
```

```
round(t(apply(beta_samples_scaled_10, 2, quantile, c(0.5, 0.025, 0.975))),2)
```

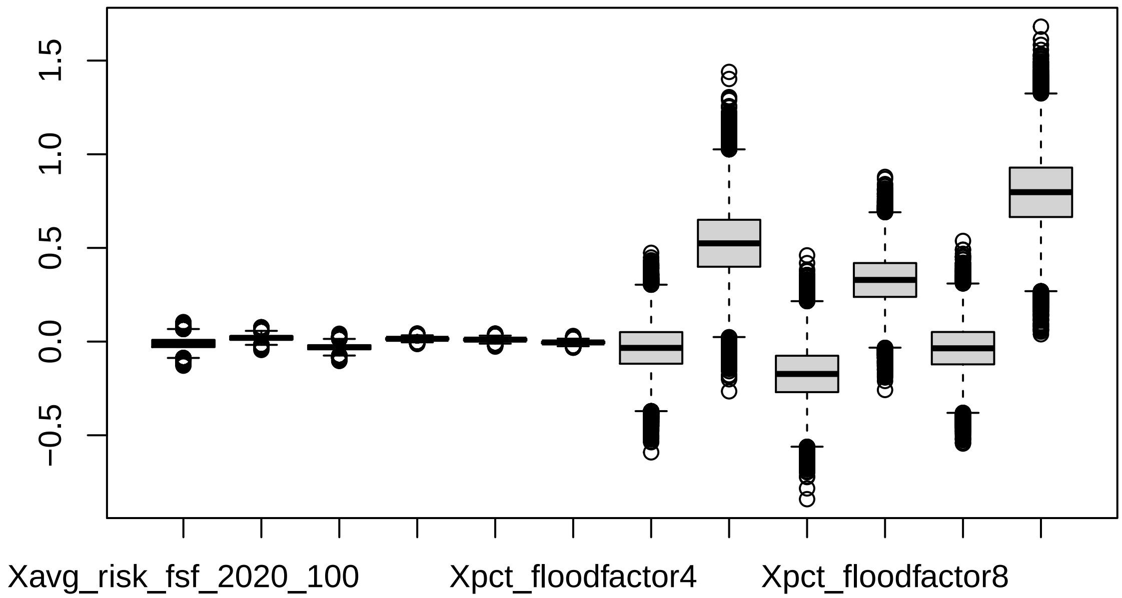
```
##                           50%  2.5% 97.5%
## (Intercept)             75.82 75.70 75.94
## Xavg_risk_fsf_2020_100  -0.01 -0.07  0.05
## Xavg_risk_score_sfha     0.02 -0.01  0.05
## Xavg_risk_score_no_sfha -0.03 -0.06  0.00
```

```
## Xpct_floodfactor2        0.01  0.00  0.03
## Xpct_floodfactor3        0.01 -0.01  0.03
## Xpct_floodfactor4        0.00 -0.02  0.01
## Xpct_floodfactor5       -0.03 -0.28  0.21
## Xpct_floodfactor6        0.52  0.16  0.89
## Xpct_floodfactor7       -0.17 -0.45  0.11
## Xpct_floodfactor8        0.33  0.07  0.59
## Xpct_floodfactor9       -0.04 -0.29  0.21
## Xpct_floodfactor10       0.80  0.42  1.18
## XEP_POV                  0.86  0.79  0.92
## XEP_UNEMP                0.08  0.03  0.13
## XEP_PCI                 -0.45 -0.53 -0.37
## XEP_NOHSDP               0.59  0.46  0.72
## XEP_AGE65                1.89  1.86  1.92
## XEP_AGE17                1.58  1.43  1.72
## XEP_DISABL               4.11  3.69  4.52
## XEP_SNGPNT              -0.16 -0.21 -0.12
## XEP_MINRTY              -0.06 -0.11 -0.02
## XEP_LIMENG               0.01 -0.02  0.04
## XEP_MUNIT               -0.18 -0.23 -0.13
## XEP_MOBILE               0.00  0.00  0.00
## XEP_CROWD               -0.04 -0.07 -0.02
## XEP_NOVEH                0.12  0.10  0.15
## XEP_GROUPQ              -0.23 -0.26 -0.20
## XEP_UNINSUR              0.15  0.11  0.19
## Xco                      0.10  0.04  0.15
## Xno2                    -0.03 -0.05  0.00
## Xo3                     -0.09 -0.50  0.35
## Xpm10                   -0.10 -0.13 -0.08
## Xpm25                    0.32  0.27  0.37
## Xso2                     0.17  0.05  0.28
## XData_Value_CSMOKING     0.81  0.74  0.89
```

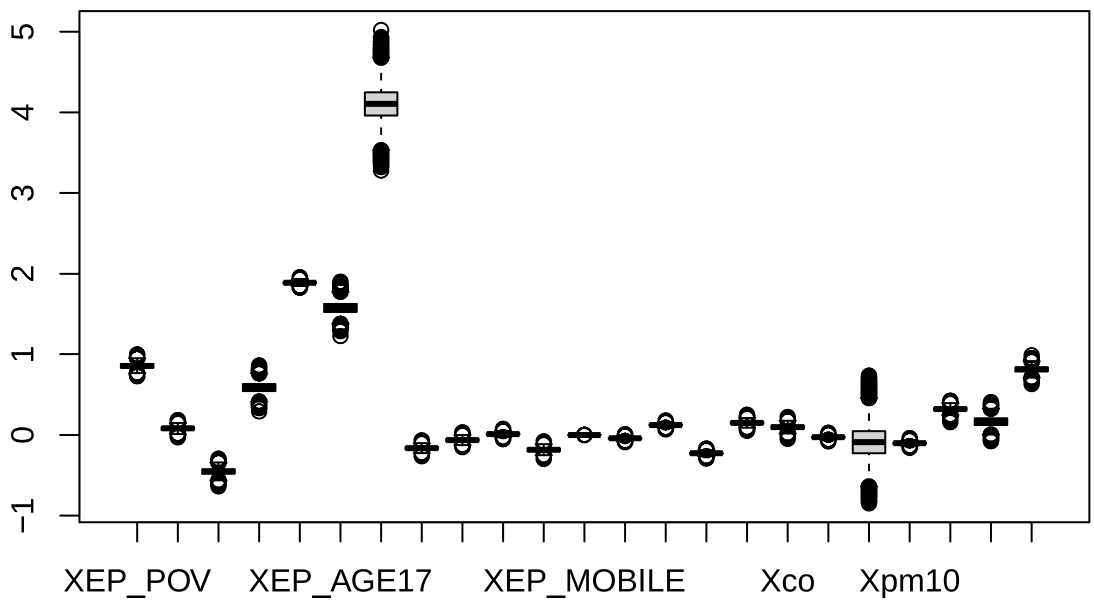
```
boxplot(beta_samples_scaled_10)
```



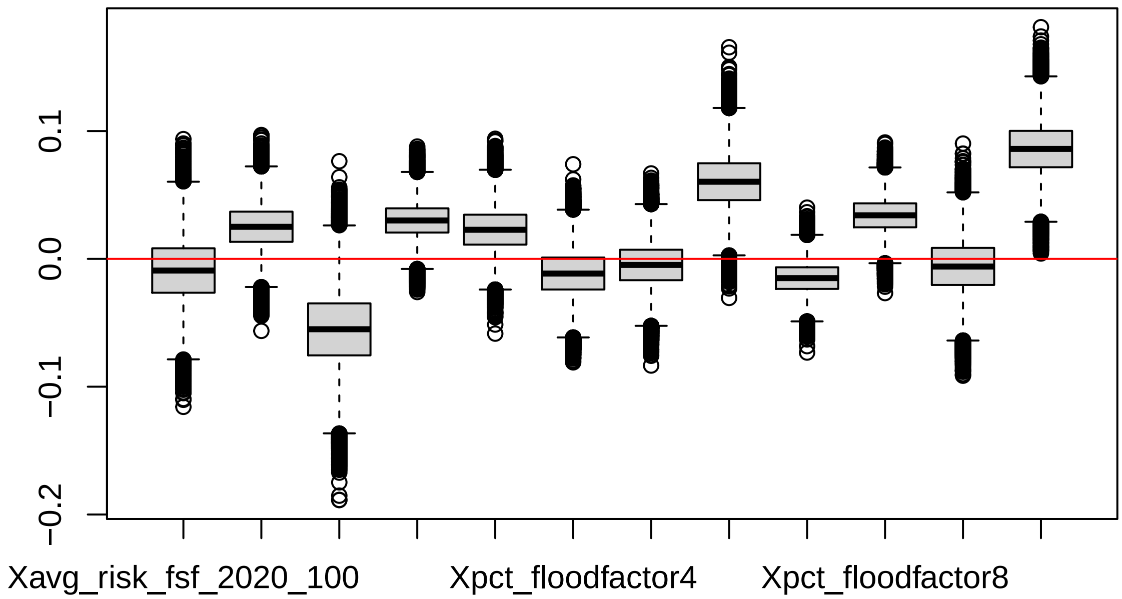
```r
boxplot(beta_samples_scaled_10[, 2:13])
```



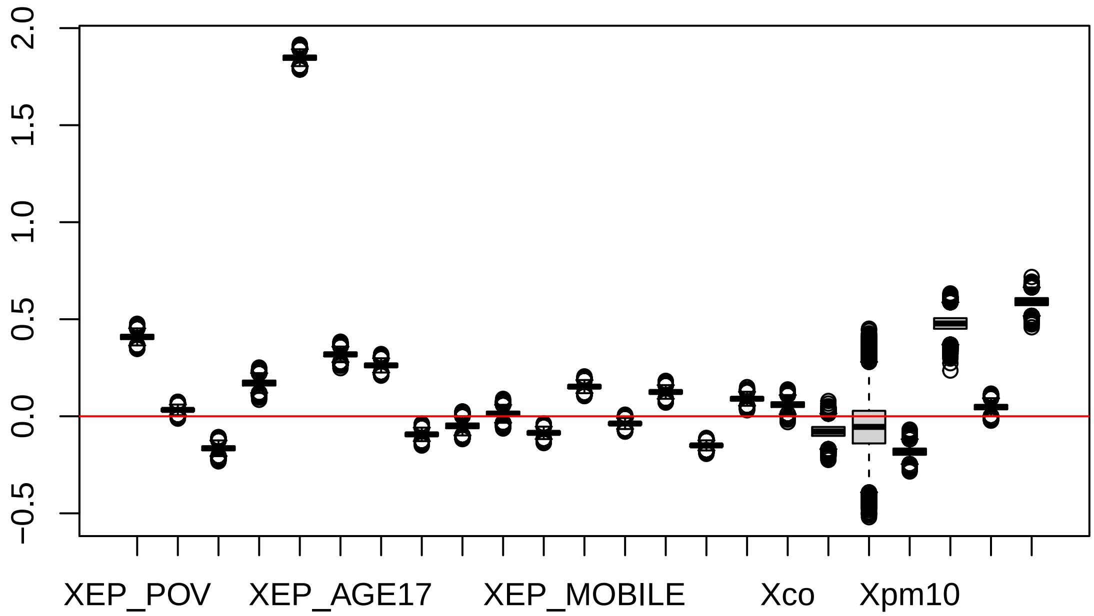

```r
boxplot(beta_samples_scaled_10[, 14:36])
```




```r
boxplot(beta_samples_matrix[, 2:13])
abline(h = 0, col = "red")
```



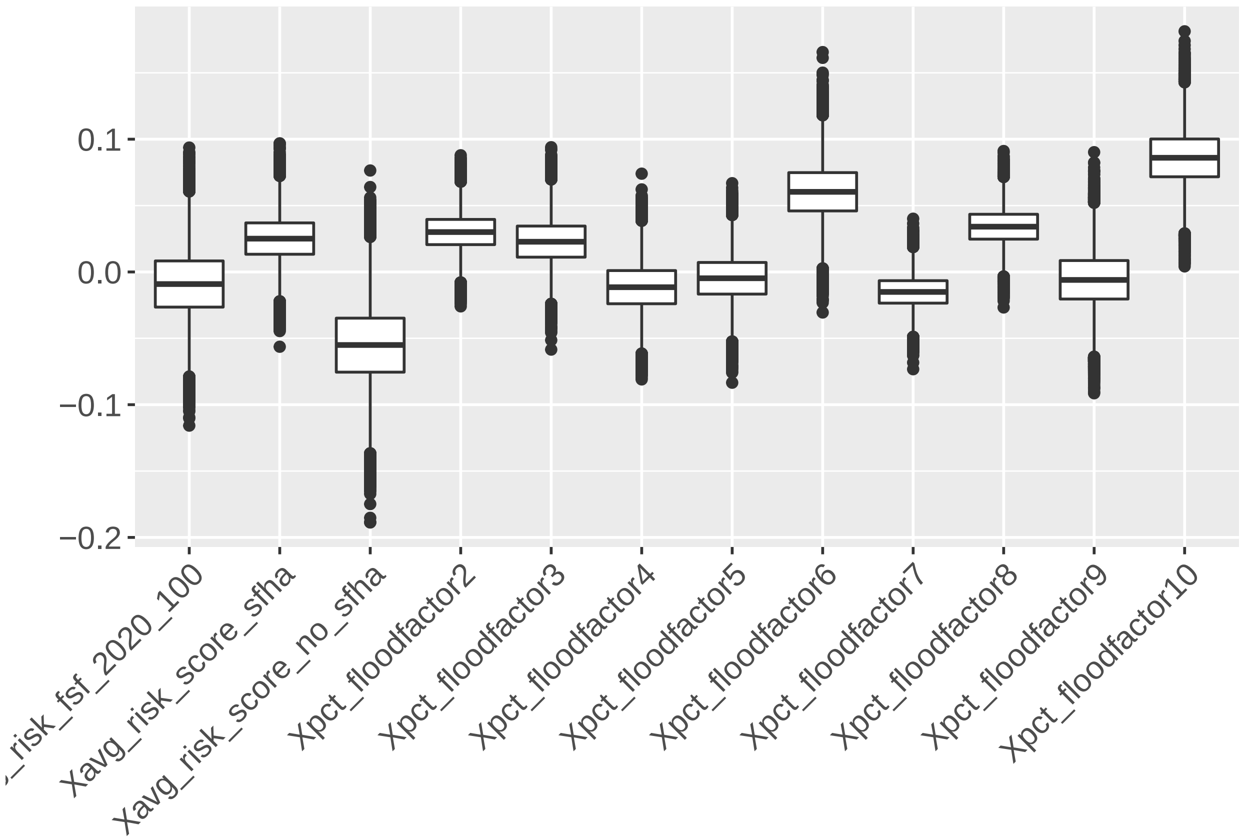

```r
boxplot(beta_samples_matrix[, 14:36])
abline(h = 0, col = "Red")
```



Boxplots for the posterior distribution of coefficients, in ggplot

```r
fl_coef_post <- stack(as.data.frame(beta_samples_matrix[, 2:13]))
```

```r
ggplot(fl_coef_post) +
  geom_boxplot(aes(x = ind, y = values)) +
  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, 20)) #+
```

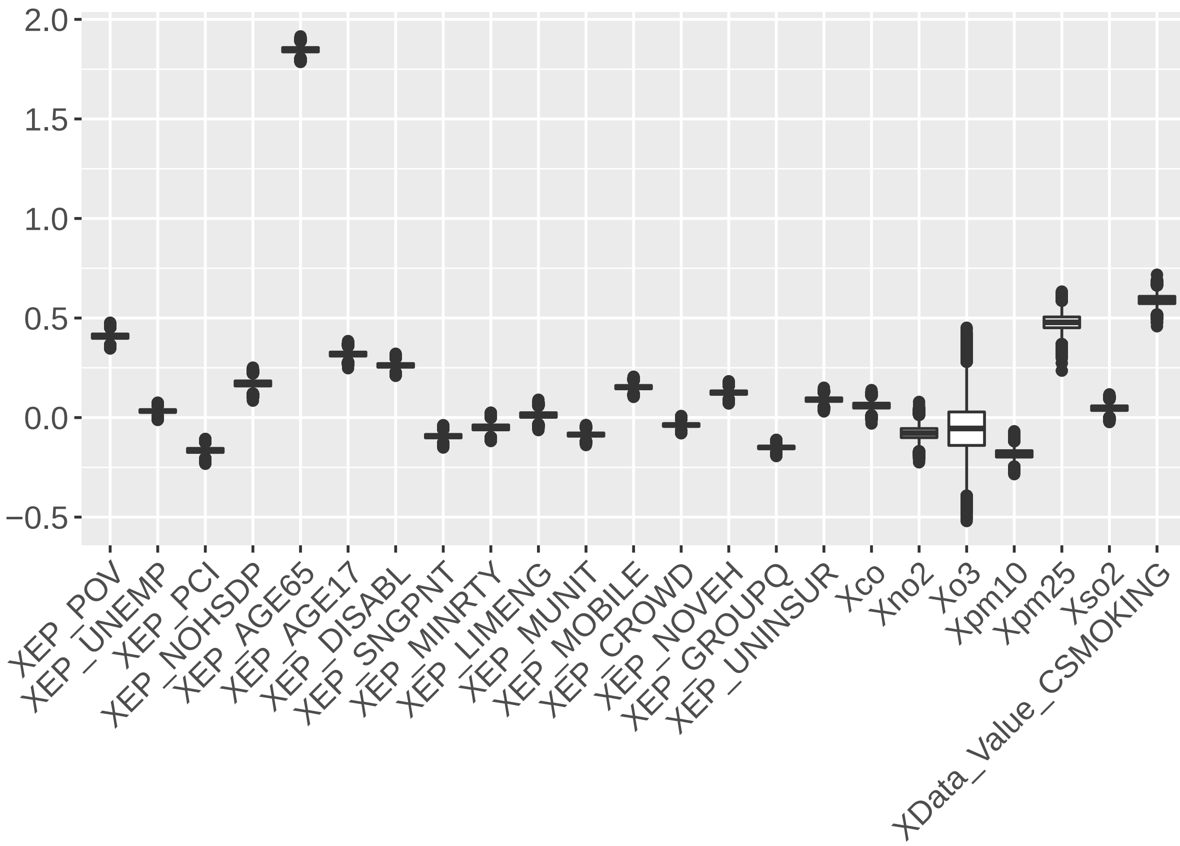



```
  # geom_vline(xintercept = c(6.5, 12.5), col = "blue") +
  # geom_hline(yintercept = 0, col = "red") +
  # annotate(geom = "text", x = 3.5, y = 20, label = "Percentage of Properties\nat Risk",
  #          col = "blue", size = 4.5) +
  # annotate(geom = "text", x = 9.5, y = 20, label = "Average Risk Score\nof Properties",
  #          col = "blue", size = 4.5) +
  # annotate(geom = "text", x = 17, y = 20, label = "Percent of Properties with Risk Score",
  #          col = "blue", size = 4.5) +
  # scale_x_discrete(labels = c("Certain, 2020", "Certain, 2050",
  #                             "Substantial, 2020", "Substantial, 2050",
  #                             "Any, 2020", "Any, 2050",
  #                             "All", "All except score 1",
  #                             "With Substantial Risk", "With Any Risk",
  #                             "In SFHA", "Not in SFHA",
  #                             "Score 2", "Score 3", "Score 4", "Score 5",
  #                             "Score 6", "Score 7", "Score 8", "Score 9",
  #                             "Score 10"))
```

Choropleth on the pct_floodfactor10

```
other_coef_post <- stack(as.data.frame(beta_samples_matrix[, 14:36]))
```

```
ggplot(other_coef_post) +
  geom_boxplot(aes(x = ind, y = values)) +
  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, 20)) #+
```




```
  # geom_vline(xintercept = c(16.5, 22.5), col = "blue") +
  # geom_hline(yintercept = 0, col = "red") +
  # annotate(geom = "text", x = 8.5, y = 1.5, label = "Social Vulnerability Index",
  #          col = "blue", size = 4.5) +
  # annotate(geom = "text", x = 19.5, y = 1.5, label = "Air Pollution",
  #          col = "blue", size = 4.5) +
  #   scale_x_discrete(labels = c("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",
  #                               "Smoking"))
```

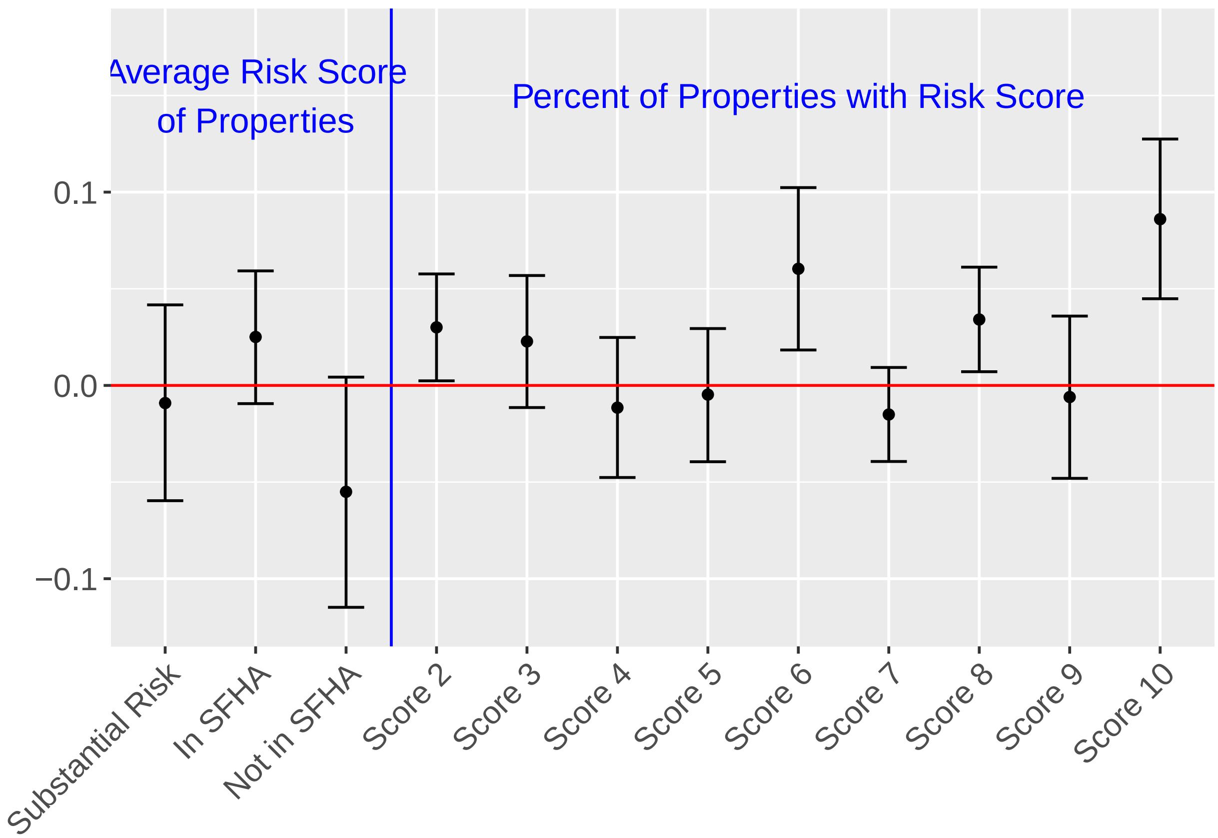
Credible Interval plots for the coefficients, in ggplot

```
# 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)
```

```
fl_subset <- beta_inference_df[2:13, ]
```

```
ggplot(fl_subset, 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(), a
       axis.text=element_text(size=12),
       plot.margin = margin(5.5, 5.5, 5.5, 20)) +
  geom_errorbar(aes(ymin = post_2.5, ymax = post_97.5, width = 0.4)) +
  geom_vline(xintercept = 3.5, col = "blue") +
  geom_hline(yintercept = 0, col = "red") +
  annotate(geom = "text", x = 2, y = 0.15, label = "Average Risk Score\nof Properties",
         col = "blue", size = 4.5) +
  annotate(geom = "text", x = 8, y = 0.15, label = "Percent of Properties with Risk Score",
         col = "blue", size = 4.5) +
  ylim(c(-0.12, 0.18)) +
  scale_x_discrete(labels = c("Substantial Risk",
                              "In SFHA", "Not in SFHA",
                              "Score 2", "Score 3", "Score 4", "Score 5",
                              "Score 6", "Score 7", "Score 8", "Score 9",
                              "Score 10"))
```




```
other_subset <- beta_inference_df[14:36, ]
```

```
ggplot(other_subset, 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(), ai
     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(16.5, 22.5), col = "blue") +
geom_hline(yintercept = 0, col = "red") +
annotate(geom = "text", x = 8.5, y = 1.5, label = "Social Vulnerability Index",
         col = "blue", size = 4.5) +
annotate(geom = "text", x = 19.5, y = 1.5, label = "Air Pollution",
         col = "blue", size = 4.5) +
  scale_x_discrete(labels = c("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",
                              "Smoking"))
```