

Sensitivity Analysis: Omitting Subsets of Variables

Alvin Sheng

Contents

CAR model results, Coronary Heart Disease, Omitting SVI	3
Beta samples	3
Examining sigma2, nu2, rho	3
Examining a sample of the 3108 phi parameters	3
Inference	4
Credible Interval plots for the coefficients, in ggplot	4
CAR model results, High Blood Pressure, Omitting SVI	6
Inference	7
CAR model results, Asthma, Omitting SVI	9
Inference	9
CAR model results, Poor Mental Health, Omitting SVI	12
Inference	12

```

library(here)

## Warning in readLines(f, n): line 1 appears to contain an embedded nul
## Warning in readLines(f, n): incomplete final line found on '/Volumes/
## ALVINDRIVE2/flood-risk-health-effects/._flood-risk-health-effects.Rproj'
## here() starts at /Volumes/ALVINDRIVE2/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  3.1.8      v dplyr   1.0.10
## v tidyr   1.2.1      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()

fhs_model_df <- readRDS("intermediary_data/fhs_model_df_fr_and_pollute_pc.rds")

var_names <- c("Intercept", names(fhs_model_df[, 19:(ncol(fhs_model_df) - 4)]))

names_omit_svi <- var_names[!(var_names %in% c("EP_POV", "EP_UNEMP", "EP_PCI", "EP_NOHSDP",
"EP_AGE65", "EP_AGE17", "EP_DISABL", "EP_SNGPNT",
"EP_MINRTY", "EP_LIMENG", "EP_MUNIT", "EP_MOBILE",
"EP_CROWD", "EP_NOVEH", "EP_GROUPQ"))]

```

Function for post-processing the inference

```

pc_idx <- 2:6
pc2flip <- c(-1, 1, -1, -1, -1)

post_flip <- function(beta_inf_subset, pc2flip) {
  names_temp <- colnames(beta_inf_subset)
  beta_inf_subset[pc2flip == -1, ] <- beta_inf_subset[pc2flip == -1, c(1, 3, 2)]
  colnames(beta_inf_subset) <- names_temp
  return(sweep(beta_inf_subset, 1, pc2flip, FUN = "*"))
}

```

CAR model results, Coronary Heart Disease, Omitting SVI

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/sensitivity_analysis/omit_subsets_of_variables/CHD_omit_SVI.RData"))
```

Beta samples

```
beta_samples <- mcmc.list(chain_list[[1]]$samples$beta, chain_list[[2]]$samples$beta,
                          chain_list[[3]]$samples$beta)
```

```
effectiveSize(beta_samples)
```

```
##          var1          var2          var3          var4          var5          var6
## 118293.2136 19294.8213 17638.7457 26481.3651 33488.2709 36803.3486
##          var7          var8          var9          var10          var11          var12
## 25122.6708 1077.2927 629.9440 818.3618 310.6115 604.5310
##          var13
## 12494.0905
```

Examining sigma2, nu2, rho

```
sigma2_samples <- mcmc.list(chain_list[[1]]$samples$sigma2, chain_list[[2]]$samples$sigma2,
                           chain_list[[3]]$samples$sigma2)
```

```
nu2_samples <- mcmc.list(chain_list[[1]]$samples$nu2, chain_list[[2]]$samples$nu2,
                        chain_list[[3]]$samples$nu2)
```

```
effectiveSize(sigma2_samples)
```

```
##          var1
## 8506.588
```

```
effectiveSize(nu2_samples)
```

```
##          var1
## 7001.059
```

Examining a sample of the 3108 phi parameters

```
phi_samples <- mcmc.list(chain_list[[1]]$samples$phi, chain_list[[2]]$samples$phi,
                        chain_list[[3]]$samples$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]
```

```
effectiveSize(phi_samples_subset)
```

```
##      var1      var2      var3      var4      var5      var6      var7      var8
## 55330.46 28272.73 12741.53 83225.89 136580.70 90191.29 60974.97 90332.31
##      var9      var10
## 111332.19 57230.14
```

Inference

```
beta_samples_matrix <- rbind(chain_list[[1]]$samples$beta, chain_list[[2]]$samples$beta,
                             chain_list[[3]]$samples$beta)
```

```
colnames(beta_samples_matrix) <- names_omit_svi
```

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

```
# flipping the inference results according to the flipped PCs
```

```
beta_inference[pc_idx, ] <- post_flip(beta_inference[pc_idx, ], pc2flip)
```

```
beta_inference
```

```
##              50%      2.5%      97.5%
## Intercept      6.66851  6.66278  6.67424
## flood_risk_pc1  0.06123  0.04511  0.07725
## flood_risk_pc2  0.00238 -0.01557  0.02047
## flood_risk_pc3  0.02183  0.00824  0.03552
## flood_risk_pc4 -0.00109 -0.01345  0.01117
## flood_risk_pc5  0.02138  0.00951  0.03318
## EP_UNINSUR     -0.06841 -0.08513 -0.05166
## pollute_conc_pc1 -0.24288 -0.30096 -0.18375
## pollute_conc_pc2 -0.51017 -0.59414 -0.42923
## pollute_conc_pc3 -0.36204 -0.43589 -0.28392
## tmmx           0.02246 -0.10094  0.14309
## rmax           0.11529  0.02433  0.20249
## Data_Value_CSMOKING 0.87054  0.84991  0.89070
```

```
saveRDS(beta_inference, file = here("modeling_files/sensitivity_analysis/omit_subsets_of_variables/beta_inference.rds"))
```

List of significant beta coefficients:

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

```
## [1] "Intercept"      "flood_risk_pc1"  "flood_risk_pc3"
## [4] "flood_risk_pc5"  "EP_UNINSUR"      "pollute_conc_pc1"
## [7] "pollute_conc_pc2" "pollute_conc_pc3" "rmax"
## [10] "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))

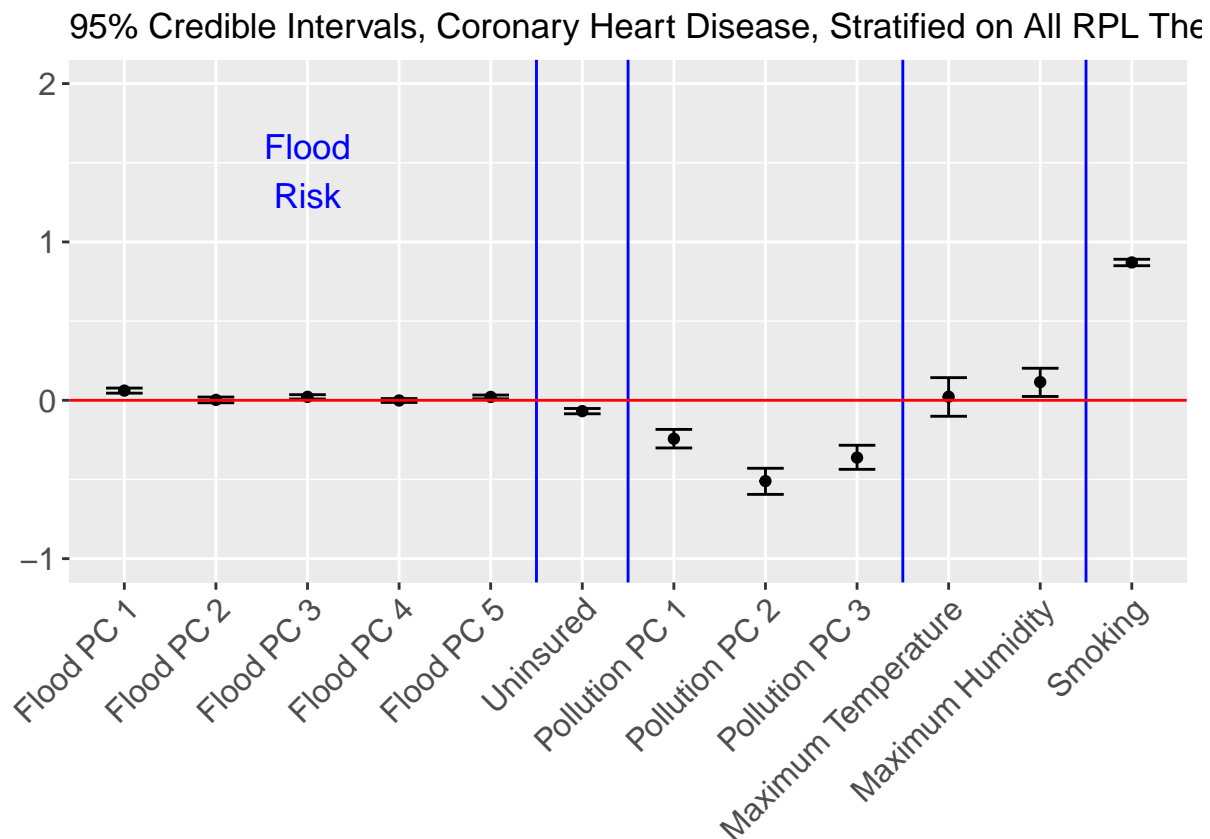
beta_inference_df <- rename(beta_inference_df,
                             post_median = `50%`,
                             post_2.5 = `2.5%`,
                             post_97.5 = `97.5%`)

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

p <- ggplot(beta_inference_df[-1, ], aes(x = var_name, y = post_median)) +
  geom_point() +
  ylim(c(-1, 2)) +
  theme(axis.text.x = element_text(angle = 45, vjust = 1, hjust=1), axis.title.x = element_blank(), axis.title.y = element_text(size=12),
        plot.margin = margin(5.5, 5.5, 5.5, 25)) +
  geom_errorbar(aes(ymin = post_2.5, ymax = post_97.5, width = 0.4)) +
  geom_vline(xintercept = c(5.5, 6.5, 9.5, 11.5), col = "blue") +
  geom_hline(yintercept = 0, col = "red") +
  annotate(geom = "text", x = 3, y = 1.45, label = "Flood\nRisk",
          col = "blue", size = 4.5) +
  scale_x_discrete(labels = c("Flood PC 1", "Flood PC 2", "Flood PC 3", "Flood PC 4", "Flood PC 5",
                              "Uninsured",
                              "Pollution PC 1", "Pollution PC 2", "Pollution PC 3",
                              "Maximum Temperature", "Maximum Humidity",
                              "Smoking")) + ggtitle("95% Credible Intervals, Coronary Heart Disease, Stratified on All RPL The

```

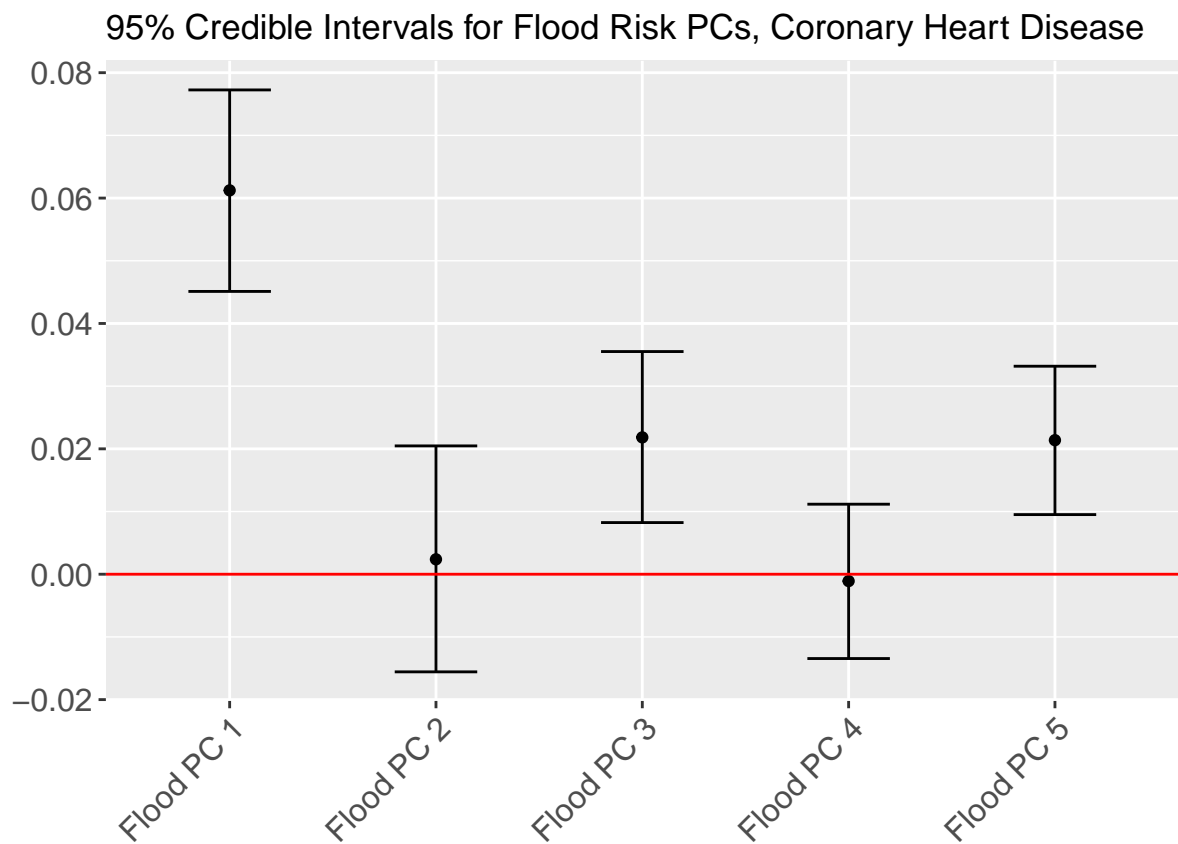
p



```
ggsave(here("figures/final_figures/sensitivity_analysis/omit_SVI/CHD_omit_SVI.pdf"),
      plot = p, device = "pdf",
      width = 8, height = 6, units = "in")
```

```
p <- ggplot(beta_inference_df[2:6, ], 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.title.y = element_text(size=12),
        axis.text=element_text(size=12),
        plot.margin = margin(5.5, 5.5, 5.5, 25)) +
  geom_errorbar(aes(ymin = post_2.5, ymax = post_97.5, width = 0.4)) +
  geom_hline(yintercept = 0, col = "red") +
  scale_x_discrete(labels = c("Flood PC 1", "Flood PC 2", "Flood PC 3", "Flood PC 4", "Flood PC 5")) + g
```

p



```
ggsave(here("figures/final_figures/sensitivity_analysis/omit_SVI/CHD_cred_intervals_fr_only.pdf"),
      plot = p, device = "pdf",
      width = 8, height = 6, units = "in")
```

CAR model results, High Blood Pressure, Omitting SVI

```
load(here("modeling_files/sensitivity_analysis/omit_subsets_of_variables/BPHIGH_omit_SVI.RData"))
```

Inference

```
beta_samples_matrix <- rbind(chain_list[[1]]$samples$beta, chain_list[[2]]$samples$beta,
                             chain_list[[3]]$samples$beta)
```

```
colnames(beta_samples_matrix) <- names_omit_svi
```

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

```
# flipping the inference results according to the flipped PCs
```

```
beta_inference[pc_idx, ] <- post_flip(beta_inference[pc_idx, ], pc2flip)
```

```
beta_inference
```

##	50%	2.5%	97.5%
## Intercept	32.36136	32.34803	32.37465
## flood_risk_pc1	0.05729	0.00802	0.10586
## flood_risk_pc2	-0.04157	-0.09604	0.01290
## flood_risk_pc3	0.05554	0.01386	0.09699
## flood_risk_pc4	0.01926	-0.01802	0.05619
## flood_risk_pc5	0.07317	0.03716	0.10898
## EP_UNINSUR	-0.27171	-0.32183	-0.22105
## pollute_conc_pc1	-1.00816	-1.19056	-0.82774
## pollute_conc_pc2	-2.41100	-2.67566	-2.14600
## pollute_conc_pc3	-0.30876	-0.55987	-0.06857
## tmmx	0.01861	-0.41729	0.41541
## rmax	0.10795	-0.19944	0.39392
## Data_Value_CSMOKING	2.15956	2.09580	2.22193

```
saveRDS(beta_inference, file = here("modeling_files/sensitivity_analysis/omit_subsets_of_variables/beta_inference.rds"))
```

List of significant beta coefficients:

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

## [1] "Intercept"	"flood_risk_pc1"	"flood_risk_pc3"
## [4] "flood_risk_pc5"	"EP_UNINSUR"	"pollute_conc_pc1"
## [7] "pollute_conc_pc2"	"pollute_conc_pc3"	"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)
```

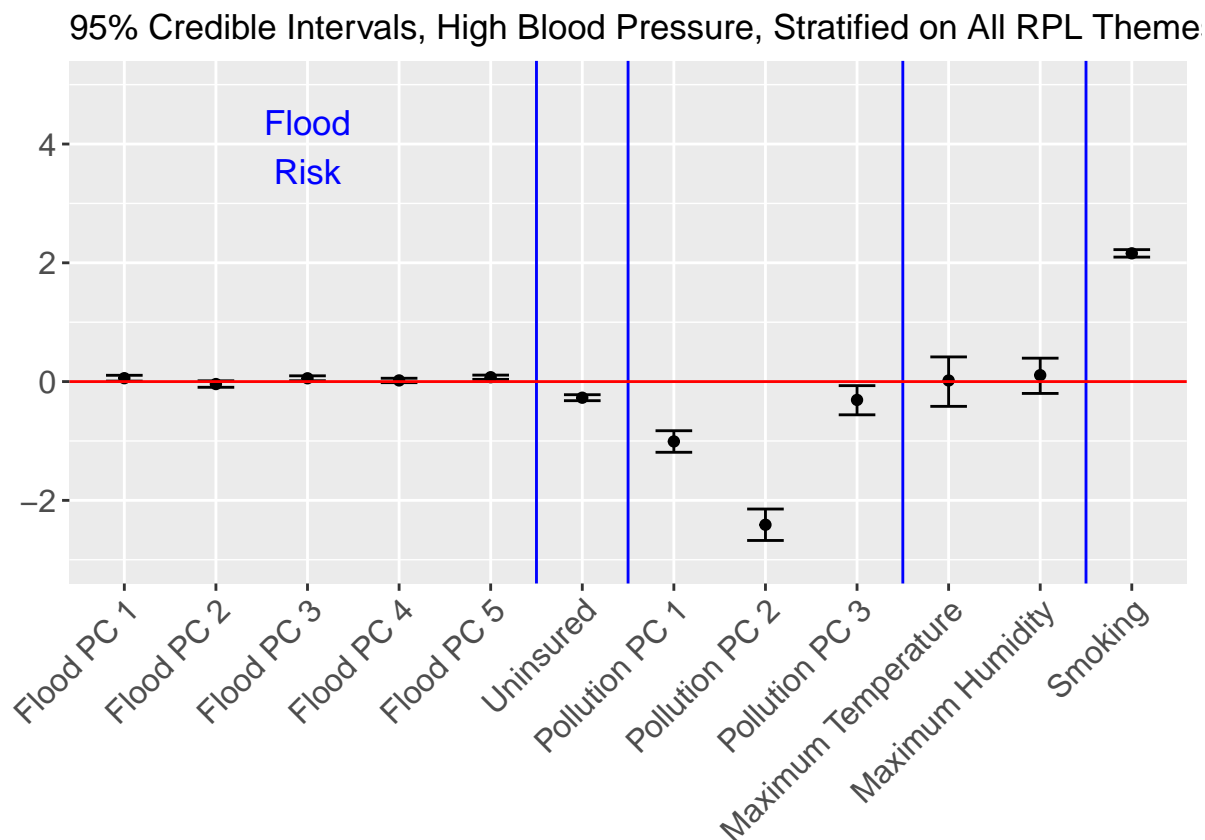
```
beta_inference_df <- mutate(beta_inference_df, var_name = row.names(beta_inference_df))
```

```
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)
```

```
p <- ggplot(beta_inference_df[-1, ], aes(x = var_name, y = post_median)) +
  geom_point() +
  ylim(c(-3, 5)) +
  theme(axis.text.x = element_text(angle = 45, vjust = 1, hjust=1), axis.title.x = element_blank(), axis.title.y = element_text(size=12),
        plot.margin = margin(5.5, 5.5, 5.5, 25)) +
  geom_errorbar(aes(ymin = post_2.5, ymax = post_97.5, width = 0.4)) +
  geom_vline(xintercept = c(5.5, 6.5, 9.5, 11.5), col = "blue") +
  geom_hline(yintercept = 0, col = "red") +
  annotate(geom = "text", x = 3, y = 3.95, label = "Flood\nRisk",
         col = "blue", size = 4.5) +
  scale_x_discrete(labels = c("Flood PC 1", "Flood PC 2", "Flood PC 3", "Flood PC 4", "Flood PC 5",
                             "Uninsured",
                             "Pollution PC 1", "Pollution PC 2", "Pollution PC 3",
                             "Maximum Temperature", "Maximum Humidity",
                             "Smoking")) + ggtitle("95% Credible Intervals, High Blood Pressure, Stratified on All RPL Theme")
```

p



```
ggsave(here("figures/final_figures/sensitivity_analysis/omit_SVI/BPHIGH_omit_SVI.pdf"),
       plot = p, device = "pdf",
       width = 8, height = 6, units = "in")
```

```
p <- ggplot(beta_inference_df[2:6, ], 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.title.y = element_text(size=12),
        plot.margin = margin(5.5, 5.5, 5.5, 25)) +
  geom_errorbar(aes(ymin = post_2.5, ymax = post_97.5, width = 0.4)) +
  geom_vline(xintercept = c(5.5, 6.5, 9.5, 11.5), col = "blue") +
  geom_hline(yintercept = 0, col = "red") +
  annotate(geom = "text", x = 3, y = 3.95, label = "Flood\nRisk",
         col = "blue", size = 4.5) +
  scale_x_discrete(labels = c("Flood PC 1", "Flood PC 2", "Flood PC 3", "Flood PC 4", "Flood PC 5",
                             "Uninsured",
                             "Pollution PC 1", "Pollution PC 2", "Pollution PC 3",
                             "Maximum Temperature", "Maximum Humidity",
                             "Smoking")) + ggtitle("95% Credible Intervals, High Blood Pressure, Stratified on All RPL Theme")
```

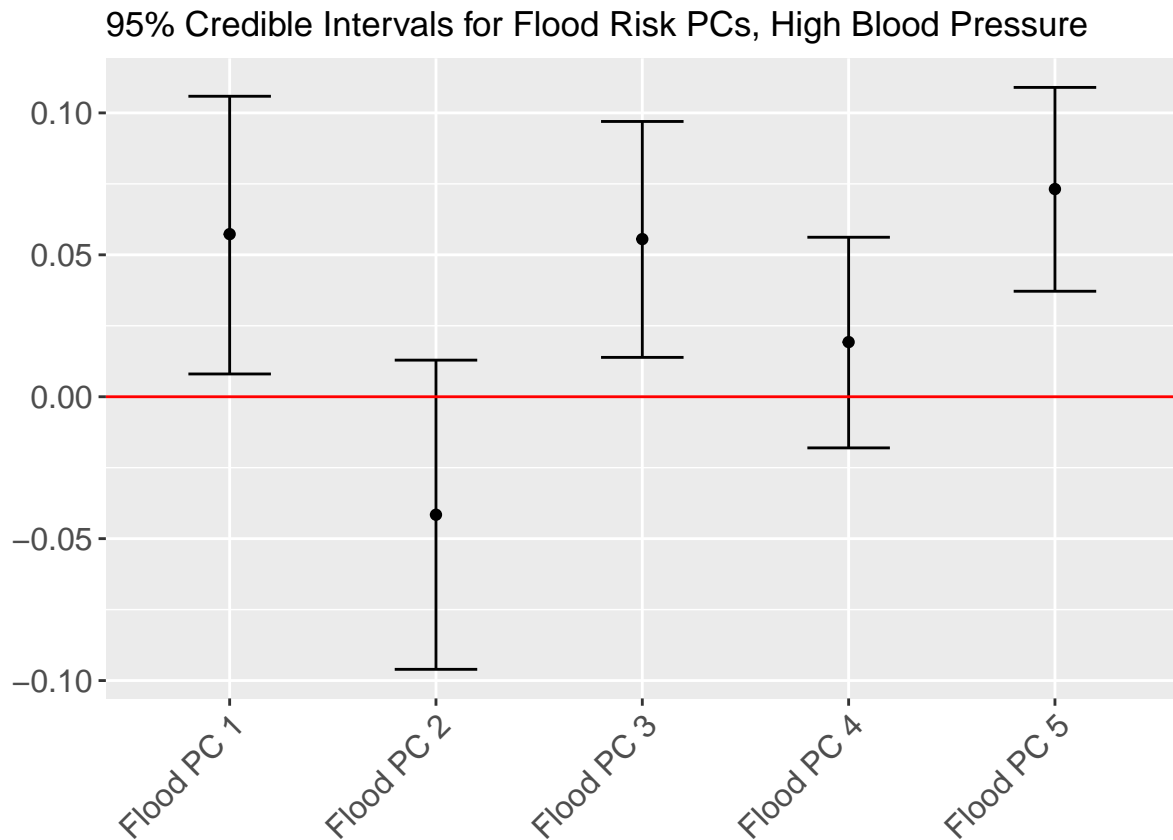


```

    plot.margin = margin(5.5, 5.5, 5.5, 25)) +
    geom_errorbar(aes(ymin = post_2.5, ymax = post_97.5, width = 0.4)) +
    geom_hline(yintercept = 0, col = "red") +
    scale_x_discrete(labels = c("Flood PC 1", "Flood PC 2", "Flood PC 3", "Flood PC 4", "Flood PC 5")) +

```

p



```

ggsave(here("figures/final_figures/sensitivity_analysis/omit_SVI/BPHIGH_cred_intervals_fr_only.pdf"),
    plot = p, device = "pdf",
    width = 8, height = 6, units = "in")

```

CAR model results, Asthma, Omitting SVI

```

load(here("modeling_files/sensitivity_analysis/omit_subsets_of_variables/CASTHMA_omit_SVI.RData"))

```

Inference

```

beta_samples_matrix <- rbind(chain_list[[1]]$samples$beta, chain_list[[2]]$samples$beta,
    chain_list[[3]]$samples$beta)

colnames(beta_samples_matrix) <- names_omit_svi

```

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

```
# flipping the inference results according to the flipped PCs
```

```
beta_inference[pc_idx, ] <- post_flip(beta_inference[pc_idx, ], pc2flip)
```

```
beta_inference
```

```
##              50%      2.5%      97.5%
## Intercept      9.89994  9.89856  9.90131
## flood_risk_pc1 -0.01258 -0.01838 -0.00675
## flood_risk_pc2 -0.01449 -0.02099 -0.00807
## flood_risk_pc3  0.00440 -0.00055  0.00931
## flood_risk_pc4  0.01729  0.01292  0.02169
## flood_risk_pc5  0.00326 -0.00096  0.00750
## EP_UNINSUR     -0.06175 -0.06763 -0.05571
## pollute_conc_pc1 0.16658  0.14446  0.18857
## pollute_conc_pc2 -0.18127 -0.21124 -0.15203
## pollute_conc_pc3 -0.12326 -0.15211 -0.09590
## tmmx           0.01235 -0.03431  0.06046
## rmax           -0.11750 -0.14901 -0.08471
## Data_Value_CSMOKING 1.30143  1.29417  1.30874
```

```
saveRDS(beta_inference, file = here("modeling_files/sensitivity_analysis/omit_subsets_of_variables/beta_inference.rds"))
```

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_UNINSUR"          "pollute_conc_pc1"
## [7] "pollute_conc_pc2"    "pollute_conc_pc3"    "rmax"
## [10] "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))
```

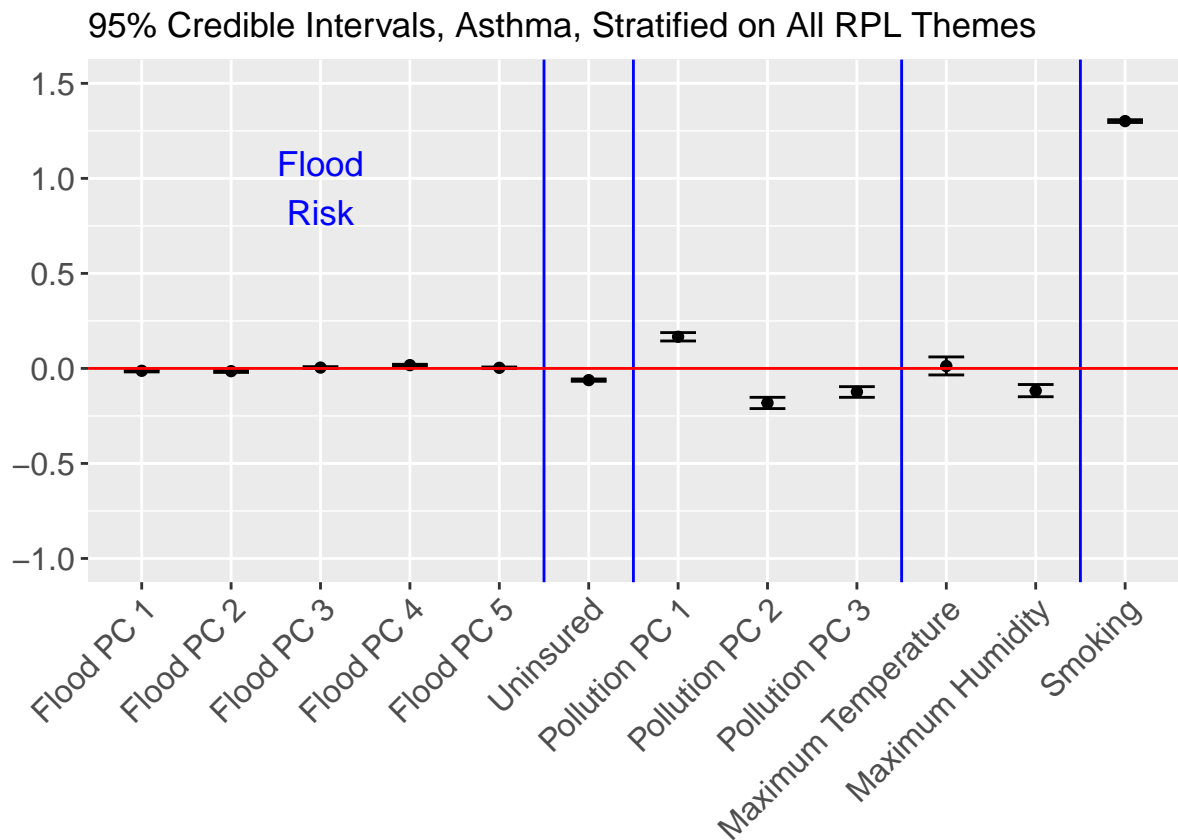
```
beta_inference_df <- rename(beta_inference_df,
                             post_median = `50%`,
                             post_2.5 = `2.5%`,
                             post_97.5 = `97.5%`)
```

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

```
p <- ggplot(beta_inference_df[-1, ], aes(x = var_name, y = post_median)) +
  geom_point() +
  ylim(c(-1, 1.5)) +
  theme(axis.text.x = element_text(angle = 45, vjust = 1, hjust=1), axis.title.x = element_blank(), axis.title.y = element_text(size=12),
        plot.margin = margin(5.5, 5.5, 5.5, 25)) +
```

```
geom_errorbar(aes(ymin = post_2.5, ymax = post_97.5, width = 0.4)) +
geom_vline(xintercept = c(5.5, 6.5, 9.5, 11.5), col = "blue") +
geom_hline(yintercept = 0, col = "red") +
annotate(geom = "text", x = 3, y = 0.95, label = "Flood\nRisk",
         col = "blue", size = 4.5) +
scale_x_discrete(labels = c("Flood PC 1", "Flood PC 2", "Flood PC 3", "Flood PC 4", "Flood PC 5",
                             "Uninsured",
                             "Pollution PC 1", "Pollution PC 2", "Pollution PC 3",
                             "Maximum Temperature", "Maximum Humidity",
                             "Smoking")) + ggtitle("95% Credible Intervals, Asthma, Stratified on All RPL Themes")
```

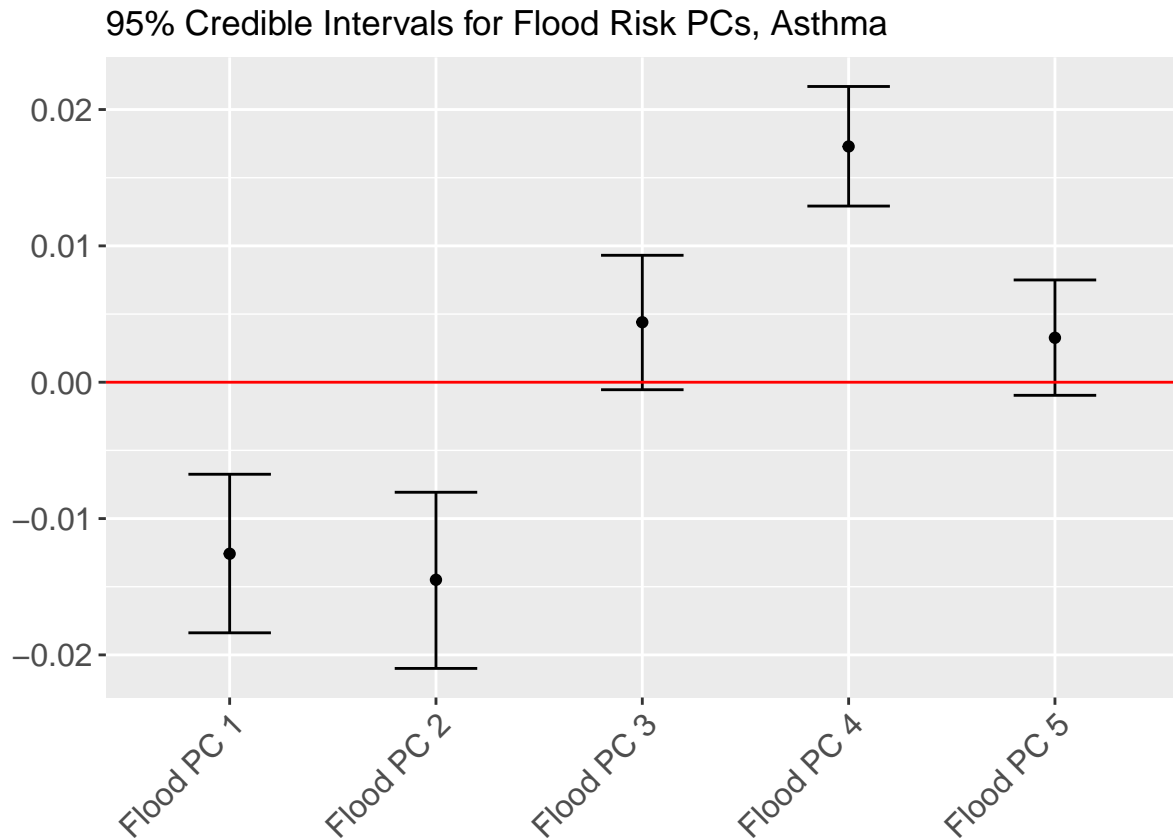
p



```
ggsave(here("figures/final_figures/sensitivity_analysis/omit_SVI/CASTHMA_omit_SVI.pdf"),
       plot = p, device = "pdf",
       width = 8, height = 6, units = "in")
```

```
p <- ggplot(beta_inference_df[2:6, ], 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.title.y = element_text(size=12),
        plot.margin = margin(5.5, 5.5, 5.5, 25)) +
  geom_errorbar(aes(ymin = post_2.5, ymax = post_97.5, width = 0.4)) +
  geom_hline(yintercept = 0, col = "red") +
  scale_x_discrete(labels = c("Flood PC 1", "Flood PC 2", "Flood PC 3", "Flood PC 4", "Flood PC 5")) +
```

p



```
ggsave(here("figures/final_figures/sensitivity_analysis/omit_SVI/CASTHMA_cred_intervals_fr_only.pdf"),
       plot = p, device = "pdf",
       width = 8, height = 6, units = "in")
```

CAR model results, Poor Mental Health, Omitting SVI

```
load(here("modeling_files/sensitivity_analysis/omit_subsets_of_variables/MHLTH_omit_SVI.RData"))
```

Inference

```
beta_samples_matrix <- rbind(chain_list[[1]]$samples$beta, chain_list[[2]]$samples$beta,
                             chain_list[[3]]$samples$beta)

colnames(beta_samples_matrix) <- names_omit_svi

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

# flipping the inference results according to the flipped PCs
beta_inference[pc_idx, ] <- post_flip(beta_inference[pc_idx, ], pc2flip)

beta_inference
```

```
##              50%      2.5%      97.5%
## Intercept    14.26066 14.25834 14.26298
```

```
## flood_risk_pc1      -0.00082 -0.01187  0.01005
## flood_risk_pc2       0.00330 -0.00879  0.01539
## flood_risk_pc3      -0.01693 -0.02603 -0.00790
## flood_risk_pc4       0.00957  0.00141  0.01773
## flood_risk_pc5      -0.00766 -0.01558  0.00022
## EP_UNINSUR          0.06135  0.05031  0.07251
## pollute_conc_pc1     0.69398  0.65341  0.73471
## pollute_conc_pc2     0.40920  0.35179  0.46701
## pollute_conc_pc3    -0.52482 -0.57923 -0.47073
## tmmx                0.05782 -0.03127  0.14021
## rmax               -0.08589 -0.15206 -0.01127
## Data_Value_CSMOKING  3.34308  3.32955  3.35698
```

```
saveRDS(beta_inference, file = here("modeling_files/sensitivity_analysis/omit_subsets_of_variables/beta_inference.rds"))
```

List of significant beta coefficients:

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

```
## [1] "Intercept"      "flood_risk_pc3"    "flood_risk_pc4"
## [4] "EP_UNINSUR"     "pollute_conc_pc1"  "pollute_conc_pc2"
## [7] "pollute_conc_pc3" "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))
```

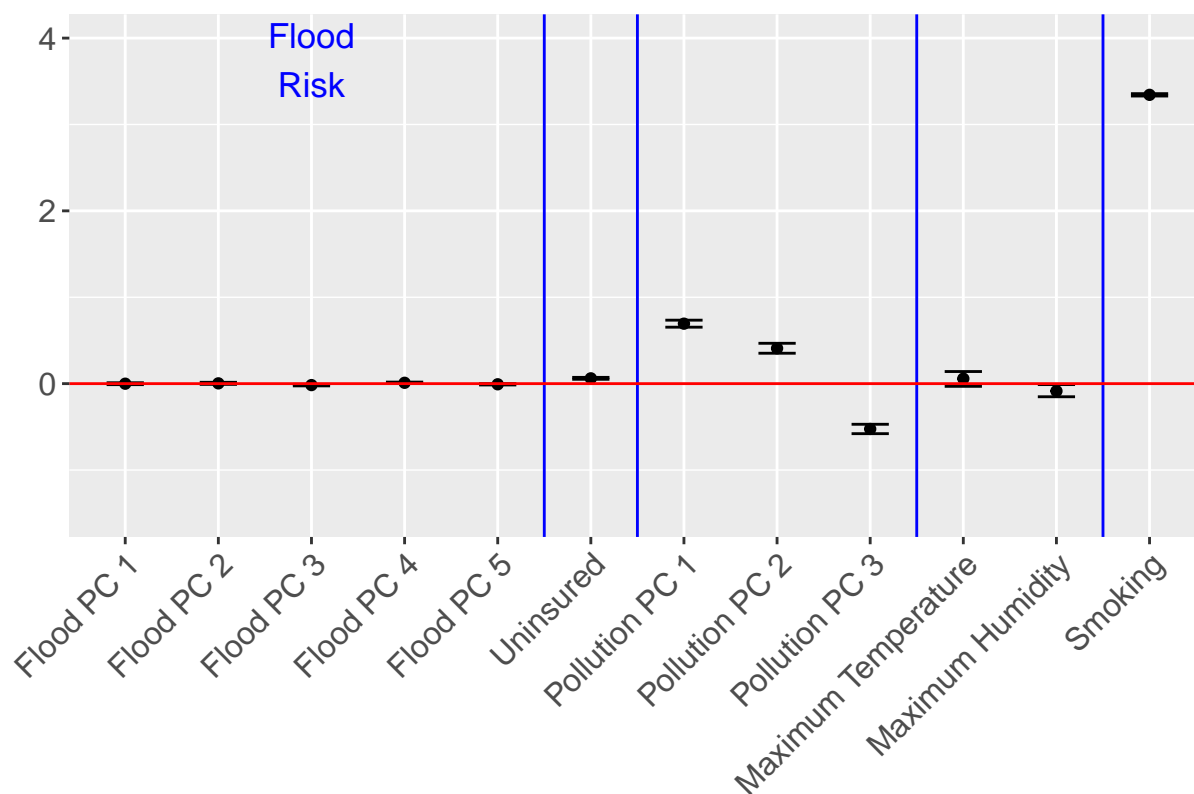
```
beta_inference_df <- rename(beta_inference_df,
                             post_median = `50%`,
                             post_2.5 = `2.5%`,
                             post_97.5 = `97.5%`)
```

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

```
p <- ggplot(beta_inference_df[-1, ], aes(x = var_name, y = post_median)) +
  geom_point() +
  ylim(c(-1.5, 4)) +
  theme(axis.text.x = element_text(angle = 45, vjust = 1, hjust=1), axis.title.x = element_blank(), axis.title.y = element_text(size=12),
        plot.margin = margin(5.5, 5.5, 5.5, 25)) +
  geom_errorbar(aes(ymin = post_2.5, ymax = post_97.5, width = 0.4)) +
  geom_vline(xintercept = c(5.5, 6.5, 9.5, 11.5), col = "blue") +
  geom_hline(yintercept = 0, col = "red") +
  annotate(geom = "text", x = 3, y = 3.75, label = "Flood\nRisk",
           col = "blue", size = 4.5) +
  scale_x_discrete(labels = c("Flood PC 1", "Flood PC 2", "Flood PC 3", "Flood PC 4", "Flood PC 5",
                              "Uninsured",
                              "Pollution PC 1", "Pollution PC 2", "Pollution PC 3",
                              "Maximum Temperature", "Maximum Humidity",
                              "Smoking")) + ggtitle("95% Credible Intervals, Poor Mental Health, Stratification")
```

p

95% Credible Intervals, Poor Mental Health, Stratified on All RPL Themes

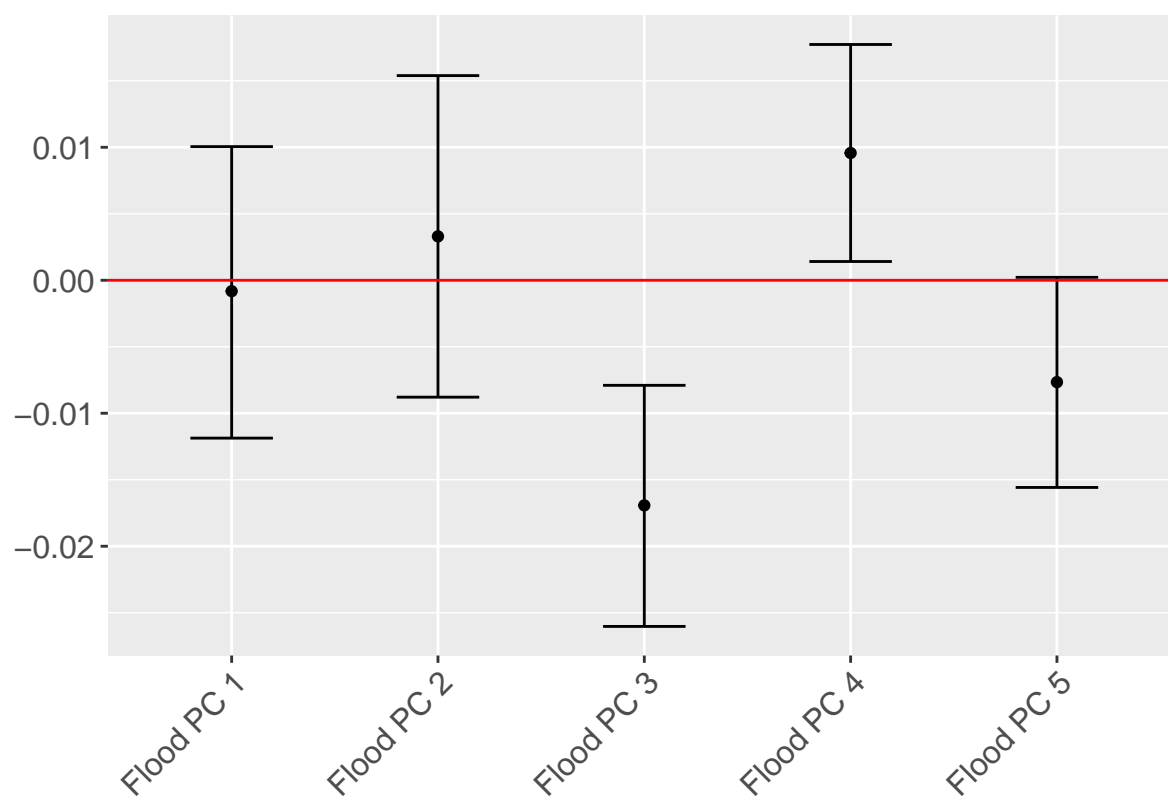


```
ggsave(here("figures/final_figures/sensitivity_analysis/omit_SVI/MHLTH_omit_SVI.pdf"),
  plot = p, device = "pdf",
  width = 8, height = 6, units = "in")
```

```
p <- ggplot(beta_inference_df[2:6, ], 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.title.y = element_text(size=12),
    plot.margin = margin(5.5, 5.5, 5.5, 25)) +
  geom_errorbar(aes(ymin = post_2.5, ymax = post_97.5, width = 0.4)) +
  geom_hline(yintercept = 0, col = "red") +
  scale_x_discrete(labels = c("Flood PC 1", "Flood PC 2", "Flood PC 3", "Flood PC 4", "Flood PC 5")) +
```

p

95% Credible Intervals for Flood Risk PCs, Poor Mental Health



```
ggsave(here("figures/final_figures/sensitivity_analysis/omit_SVI/MHLTH_cred_intervals_fr_only.pdf"),
       plot = p, device = "pdf",
       width = 8, height = 6, units = "in")
```