STA723 Case Study - Group 1

Melody Jiang, Irene Ji, Keru Wu

1/20/2020

Abstract

This report evaluates association between exposures Dichlorodiphenyldichloroethylene (DDE) & Polychlorinated Biphenyls (PCBs) and the risk of preterm birth, using a subset of data from the National Collaborative Perinatal Project(CPP). We adopted the Generalized Additive Model (GAM) as well as its Bayesian counterpart for analysis. Our approach successfully captured the nonlinearity between interested variables and risk of premature delivery, and it also aligns with known conclusions in epidemiology. Our results demonstrate that exposure to higher concentrations of DDE and PCBs are associated higher risk of preterm birth.

1. Introduction

The dataset taken from CPP was studied by Longnecker et al., including 2380 samples of women and their children which (2001). It is of interest to study how chemical exposures relate to preganancy since abnormalities in pregnancy such as preterm birth might cause unfavorable developmental outcomes in children. Preterm birth is deliveries occuring earlier than 37 weeks of gestational age. DDE and PCBs are of primary interest among all exposures, both of which are breakdown products in the body of chemicals. These chemicals build up fatty deposits and exist universally in our body, potentially impacting our health. The dataset also contains other possible confounding covariates such as cholesterol, triglyceridesage, maternal age, race, smoking status and testing center. The goal of our analysis is to assess how DDE and PCBs relate the risk of preterm birth, controlling for counfounding variables.

Our findings demonstrate that there is positive association between two interested exposures and the risk of preterm birth. Higher concentrations of DDE and PCBs are related to higher risk of premature delivery. Although various PCBs show diffrent significance in the relationship, generally they all contribute to higher risk. Some noticable PCBs are pcb_074, pcb_105, pcb_118, pcb_153 and pcb_170. Another finding of our analysis is that when concentration of these exposures is below a threshold, both exposures have little effect, while they have a constant effect when they are above a higher threshold.

2. Materials & Methods

We grouped Gestational Age into binary response variable (Preterm Delivery or Non-preterm Delivery). On handeling missing data, we removed albumin from our dataset because around 93% of this variable is missing. There is only one observation missing DDE and PCBs, and we removed this observation. We imputed missing data in the covariates using the *MICE* R package. We applied Principle Component Analysis (PCA) to PCBs and obtained 5 Principle Components (PCs), which can be seen as aggregation of PCBs.

After data pre-processing, in order to allow for non-linear dose-response relationship, we applied logistic Generalized Additive Model (GAM) to fit the data.The model smoothes numeric covariates, including DDE, Pricipal Components (PCs) 1 to 4 of PCBs, Maternal Age, triglycerides level and cholesterol level to allow for non-linear association. Furthermore, we adjusted for categorical confounding variables by including them in the model.

We conducted model checking on the fitted GAM model and examined the effect plots of DDE and PCs. In order to quantify the association for DDE and PCBs (via PCs), we controled for all other covariates and computed the change in probability of Preterm Delivery at different levels of DDE and PCs.

However, frequentist approach may overestimate uncertainty and produce a non-significant p-value. In addition, residual plot of previous GAM model indicates that assumptions of frequentist GAM could be unwarranted. To improve performance of our model, we instead use a Bayesian Generalized Additive Model. We add priors on the common regression coefficients and priors on the standard deviations of the smooth terms. We use default settings in $stan_gamm4$ function in R package $rstanarm$, which adopts a weak informative normal prior for all common regression coefficients and standard deviations of the smooth terms.

3. Results

3.1 Exploratary Data Analysis and Preprocessing

We examined the correlation among explanatory variables and found high correlation between PCBs, as shown in Figure 1A. Such high correlation might distort modeling result, so we made PCBs into Principal Components (PCs), as shown in Figure 1B.

After data pre-processing, we examined the distribution of DDE and first principle component of PCB, and found that for both DDE and PCB, the concentration of chemical is generally higher in observations that has preterm delivery, as shown in Figure 1C and Figure 1D. Among the possible covariates, most notably, there is a nonlinear relationship between age of mother and proportion of early delivery, as shown in Figure 1E.

3.2 Main Results

Using frequentist GAM, we found that there are slightly positive association between chemical exposures and preterm delivery. Among all covariates, DDE, PC1, As shown in Figure 2 in the appendix, as DDE level increases, the log-odds of preterm delivery increases. Increasing PC1 is also associated with increasing log-odds of preterm delivery. The change in probability of preterm delivery at different levels of DDE and PC1 are summarized in Tables 1 & 2 in the Appendix.

As shown in the tables, higher DDE level is associated with higher probability of preterm delivery. But the rate of increment decreases as DDE level increases, which aligns with our presumption of dose-response effect (the effect may be less significant as chemical level increases). As for PC1, there is also positive association between PC1 and the probability of preterm delivery. As PC1 has positive relationship with PCBs, the PCBs also have positive association with preterm delivery. Hence, we conclude that the chemicals (DDE and PCBs) have positive association with preterm delivery.

Results from our Bayesian Generalized Additive Model align with previous results in frequentist approach. In Figure 3, higher concentration of DDE and PCBs is related to higher risks for pregnant women. In addition, both DDE and the first principle component have significant p-values, indicating the importance of these exposures in the relationship with the response. If we further consier use estimates of Bayesian GAM to check residual assumptions of GAM, Figure 4 verifies the correctness of Bayesian GAM, which outperforms that of frequentist approach. Estimated effects of DDE and PCBs also have narrower credible intervals compared to frequentist ones. Our model capture some flat regions when concentration is relatively low and high.

Increase of 1 unit in DDE leads to approxiamately an increase of 0.017 in log odds when DDE has the lowest concentration 2.5 ug/dL. And it decrease to 0.015 when DDE reaches the average concentration 30 ug/dL. But when DDE reaches 120 ug/dL, its increase does not impact the log odds of risk (almost constant risk). This flat region result aligns with domain knowledge of epidemiology that chemical effects

become stable after reaching a upper bound. We conclude similar results when analyzing the first principle component of PCBs: when PC1 is lower than 0 or higher than 15, Bayesian GAM shows that change in PC1 has almost no effect. But when PC1 is around its mean value 7.5, one unit increase in PC1 results in an increase of 0.1 in log odds. Note that all loadings for the first PC are positive, where pcb_74, pcb_105, pcb_138, etc. have loadings over 0.3. Therefore after transforming back to original scale, flat region still exist for PCBs, and unit increase in one specific PCB can be attained. (e.g. one unit increase in pcb_138 leads to an increase of 0.66 in log odds when pcb_138 is at mean level 0.67 ng/dL)

3.3 Sensitivity Analysis

Frequentist GAM and Bayesian GAM give confidence intervals and credible intervals for measuring the effects respectively. Generally speaking, Bayesian GAM has narrower intervals and more significant p-values in anova tests. Both models show one common result that when concentrations of DDE & PCBs become higher, uncertainty increase greatly. This is partially due to the fact that we have limited data for higher level concentrations. Refer to attached figures for detailed CI intervals.

In addition, how to deal with the collinearity of PCBs and carry out dimensionality reduction has great influence in our approach. Our adopted PCA approach has the best performance compared to simple sum and factor analysis, which either have low sigfinicance for PCBs, or become unwarranted after model check.

4. Discussion

This report has analyzed how DDE and PCBs relate to the risk of premature delivery. After preprocessing data (e.g. impute missing data) and dimensionality reduction (PCA), we build up different models for the data and finally adopt the Generalized Additive Model (GAM) and its Bayesian version. Our approach has advantages that it captures the nonlinearity relationship between exposures and outcomes, and it also fits confounding variables. We conclude that higher exposure to DDE and certain PCBs may be associated with higher risk of premature delivery.

The first extension of our approach is to deal with different centers specially. Our model demonstrates that center 15 and center 37 may deviate from others, which genrally collect samples with higher risks. There are other ways which may perform better in dealing with centers. One can adopt a Bayesian hierarchical model which specifies different variances between centers. Another more direct extension of our GAM model is to include mixed effect. We may use Generalized Additive Mixed Model (GAMM) to consider random effect of centers and it can also be applied to other categorical variables like smoking status and race.

After we find out that DDE and PCBs are related to higher risk of preterm birth, we can examine the trend of exposures effects more accurately. Specifying a special prior (e.g. guarantee monotonicity) may benefit from narrower credible intervals compared to frequentist approach and naive bayes approach.

Furthermore, interaction between chemicals also impacts human health outcomes. Collinearity among PCBs indicates the need for a general dimension reduction method or a variable selection approach. Ferrari and Dunson (2019) build up a bayesian factor model designed for interactions. High correlation between exposure levels can be explained in this flexible dimension reduction approach. Another future research direction is to include penalty in GAM for variable selection.

References

Ferrari, Federico, and David B. Dunson. "Bayesian Factor Analysis for Inference on Interactions." arXiv preprint arXiv:1904.11603 (2019).

Gabry, Jonah, and Ben Goodrich. "rstanarm: Bayesian applied regression modeling via Stan." R package version 2.1 (2016).

Hastie, Trevor J. "Generalized additive models." Statistical models in S. Routledge, 249-307. (2017).

Appendix A. Figures and Tables

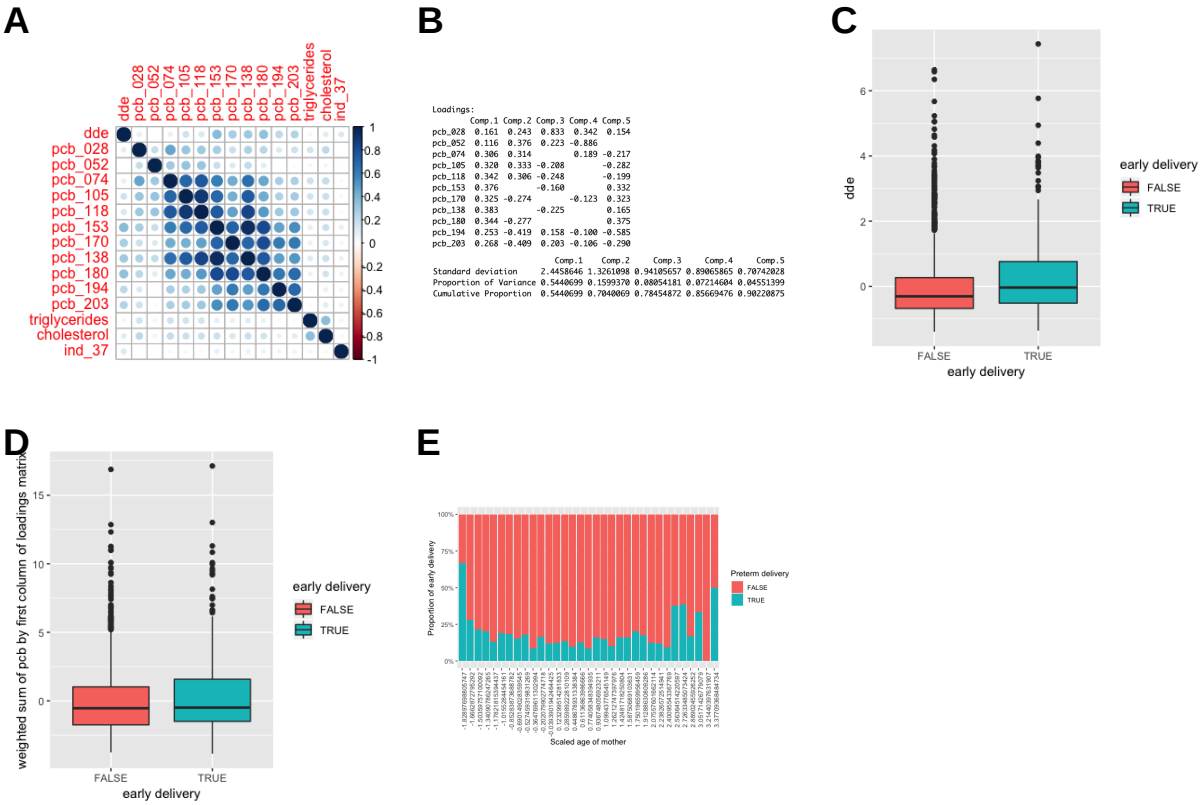


Figure 1. Plots for exploratory data analysis

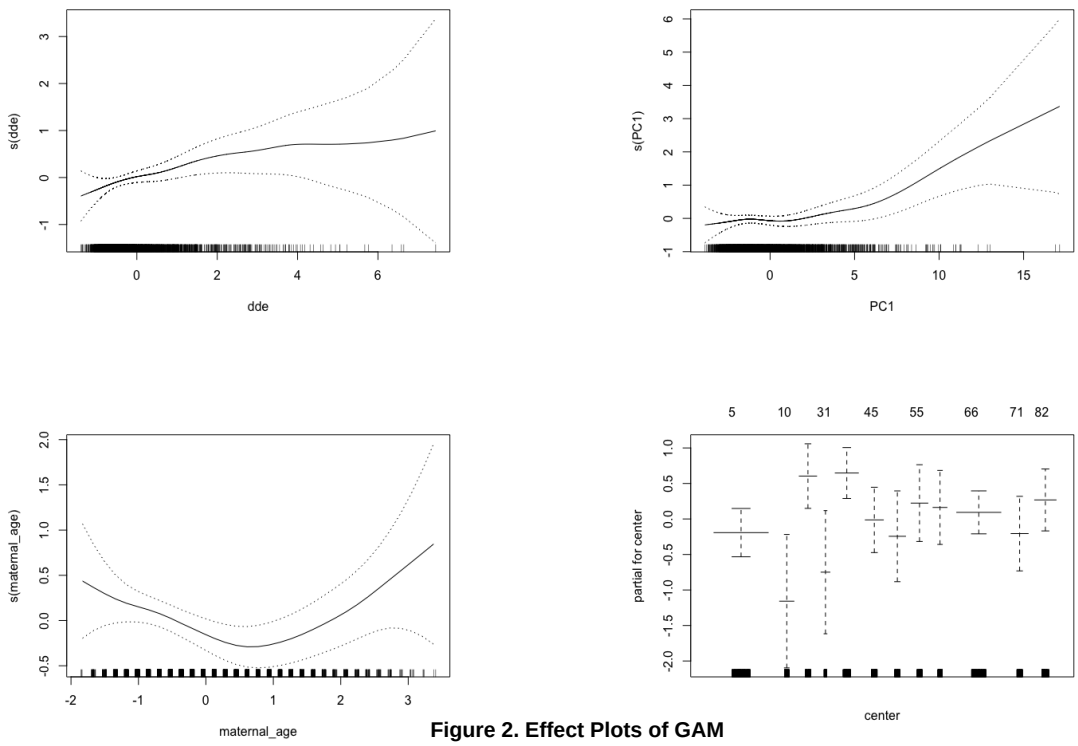


Figure 2. Effect Plots of GAM

	Scaled DDE	% Change in Probability
1	-1.00	1.43
2	2.00	1.23
3	4.00	0.32

Table 1: Change in Probability of Preterm Delivery at Different DDE Levels

	PC1	% Change in Probability
1	-3.00	0.35
2	3.00	0.66
3	7.00	2.11
4	10.00	4.88

Table 2: Change in Probability of Preterm Delivery at Different PC1 Levels

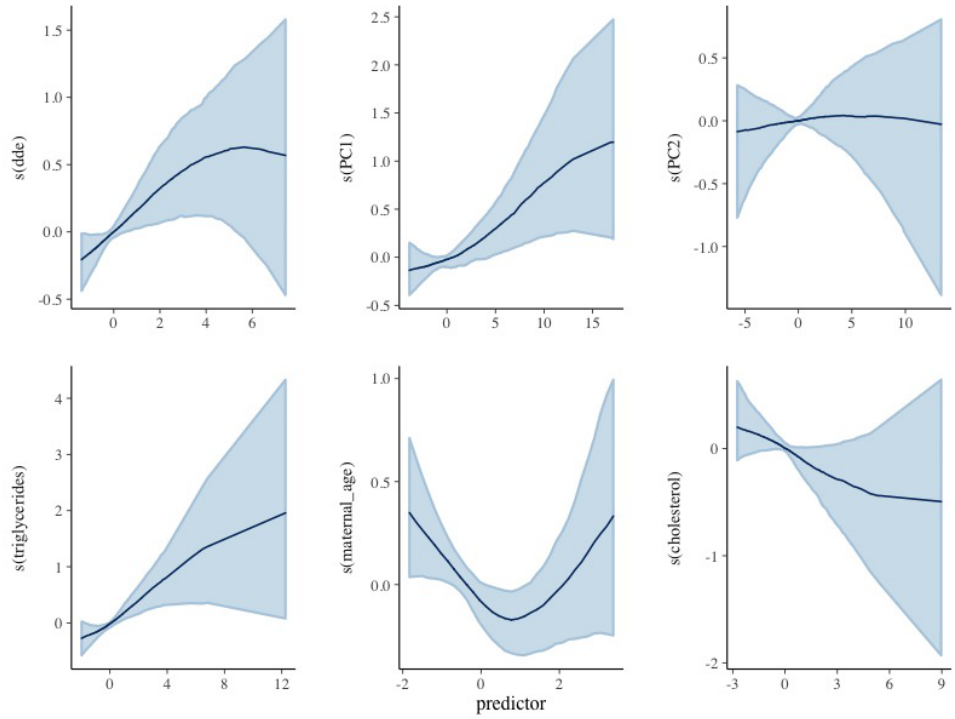


Figure 3. Nonlinear trend of several variables in Bayesian GAM

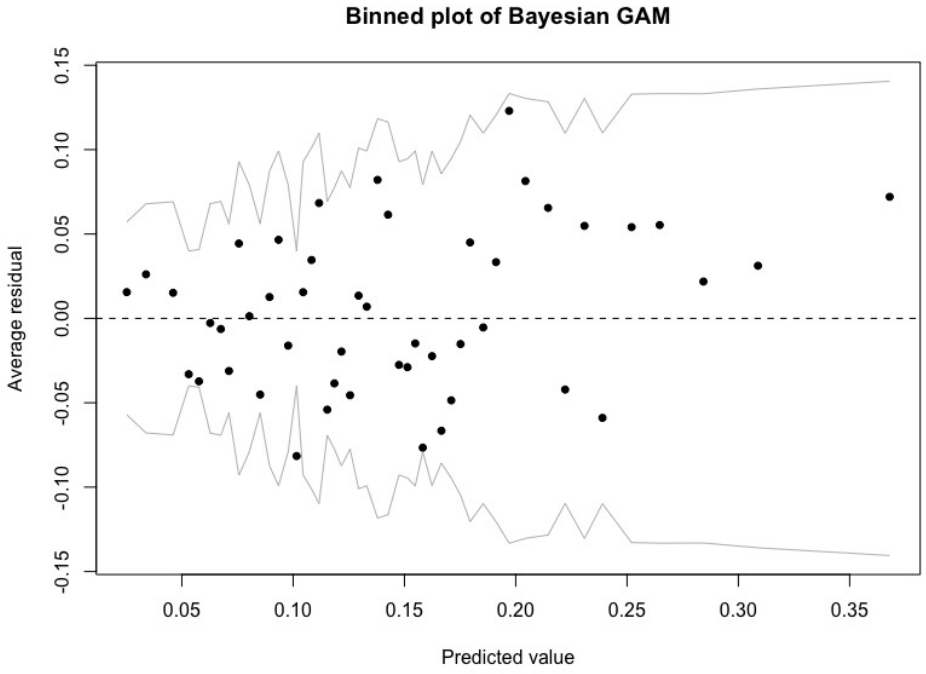


Figure 4. Bayesian GAM model check

Appendix B. Codes and Outputs

1 EDA

1.1

```
dat = readRDS("Longnecker.rds")
dat$center = factor(dat$center)
dat$smoking_status = factor(dat$smoking_status)

dat = dat[-1861,]
library(mice)
```

```
## Loading required package: lattice
```

```
##
## Attaching package: 'mice'
```

```
## The following objects are masked from 'package:base':
##
##     cbind, rbind
```

```
dat = dat[,!names(dat) %in% c('albumin')]
imp = mice(dat)
```

```
##
##  iter imp variable
##   1   1  score_education  score_income  score_occupation
##   1   2  score_education  score_income  score_occupation
##   1   3  score_education  score_income  score_occupation
##   1   4  score_education  score_income  score_occupation
##   1   5  score_education  score_income  score_occupation
##   2   1  score_education  score_income  score_occupation
##   2   2  score_education  score_income  score_occupation
##   2   3  score_education  score_income  score_occupation
##   2   4  score_education  score_income  score_occupation
##   2   5  score_education  score_income  score_occupation
##   3   1  score_education  score_income  score_occupation
##   3   2  score_education  score_income  score_occupation
##   3   3  score_education  score_income  score_occupation
##   3   4  score_education  score_income  score_occupation
##   3   5  score_education  score_income  score_occupation
##   4   1  score_education  score_income  score_occupation
##   4   2  score_education  score_income  score_occupation
##   4   3  score_education  score_income  score_occupation
##   4   4  score_education  score_income  score_occupation
##   4   5  score_education  score_income  score_occupation
##   5   1  score_education  score_income  score_occupation
##   5   2  score_education  score_income  score_occupation
##   5   3  score_education  score_income  score_occupation
##   5   4  score_education  score_income  score_occupation
##   5   5  score_education  score_income  score_occupation
```

```
dat = complete(imp)
```

```r
dat$ind_37 = dat$gestational_age < 37
dat[,c(1:12,13,15,16,17,18,20)] = scale(dat[,c(1:12,13,15,16,17,18,20)])
```

2 PCA

2.1 PCA results

```r
## PCA

pca = princomp(dat[,2:12])
pcb = as.matrix(dat[,2:12])
pcb_pc = pcb %*% pca$loadings
print(pca$loadings)
```

```
##
## Loadings:
##         Comp.1 Comp.2 Comp.3 Comp.4 Comp.5 Comp.6 Comp.7 Comp.8 Comp.9
## pcb_028  0.161  0.243  0.833  0.342  0.154  0.138  0.224
## pcb_052  0.116  0.376  0.223 -0.886
## pcb_074  0.306  0.314         0.189 -0.217 -0.547 -0.580 -0.233
## pcb_105  0.320  0.333 -0.208        -0.282  0.243  0.191  0.411 -0.497
## pcb_118  0.342  0.306 -0.248        -0.199                        0.202
## pcb_153  0.376        -0.160         0.332  0.188  0.106 -0.327  0.162
## pcb_170  0.325 -0.274        -0.123  0.323 -0.427         0.689  0.193
## pcb_138  0.383        -0.225         0.165  0.117  0.121 -0.213  0.414
## pcb_180  0.344 -0.277                0.375                -0.259 -0.676
## pcb_194  0.253 -0.419  0.158 -0.100 -0.585 -0.292  0.494 -0.220
## pcb_203  0.268 -0.409  0.203 -0.106 -0.290  0.546 -0.547  0.120  0.114
##         Comp.10 Comp.11
## pcb_028
## pcb_052
## pcb_074 -0.155
## pcb_105 -0.350   0.154
## pcb_118  0.686  -0.383
## pcb_153 -0.493  -0.544
## pcb_170
## pcb_138          0.723
## pcb_180  0.361
## pcb_194
## pcb_203
##
##                Comp.1 Comp.2 Comp.3 Comp.4 Comp.5 Comp.6 Comp.7 Comp.8
## SS loadings     1.000  1.000  1.000  1.000  1.000  1.000  1.000  1.000
## Proportion Var  0.091  0.091  0.091  0.091  0.091  0.091  0.091  0.091
## Cumulative Var  0.091  0.182  0.273  0.364  0.455  0.545  0.636  0.727
##                Comp.9 Comp.10 Comp.11
## SS loadings     1.000   1.000   1.000
## Proportion Var  0.091   0.091   0.091
## Cumulative Var  0.818   0.909   1.000
```

```r
summary(pca)
```

```
## Importance of components:
```

```
##                            Comp.1    Comp.2     Comp.3     Comp.4
## Standard deviation     2.4458646 1.3261098 0.94105657 0.89065865
## Proportion of Variance 0.5440699 0.1599370 0.08054181 0.07214604
## Cumulative Proportion  0.5440699 0.7040069 0.78454872 0.85669476
##                            Comp.5     Comp.6     Comp.7     Comp.8
## Standard deviation     0.70742028 0.58369249 0.52378706 0.46772252
## Proportion of Variance 0.04551399 0.03098547 0.02495166 0.01989603
## Cumulative Proportion  0.90220875 0.93319422 0.95814588 0.97804191
##                            Comp.9     Comp.10     Comp.11
## Standard deviation     0.35696738 0.284825079 0.181346021
## Proportion of Variance 0.01158903 0.007378131 0.002990928
## Cumulative Proportion  0.98963094 0.997009072 1.000000000
```

```
dat$PC1 = pcb_pc[,1]
dat$PC2 = pcb_pc[,2]
dat$PC3 = pcb_pc[,3]
dat$PC4 = pcb_pc[,4]
```

3 GAM model

3.1 Fit GAM model

```
library(gam)
```

```
## Loading required package: splines
```

```
## Loading required package: foreach
```

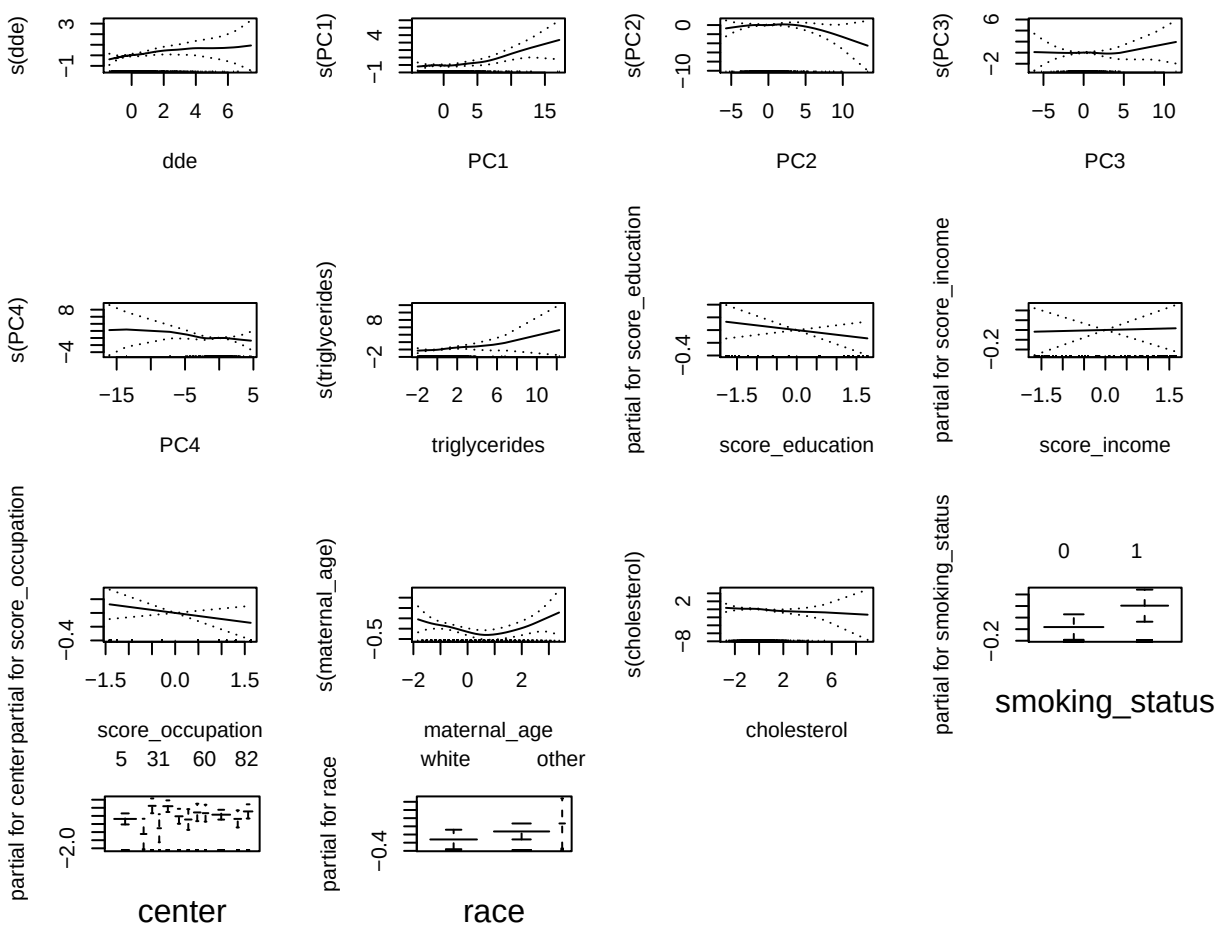
```
## Loaded gam 1.16.1
```

```
ga1 = gam(ind_37 ~ s(dde) + s(PC1) + s(PC2) + s(PC3) + s(PC4) +
            s(triglycerides) + score_education + score_income + score_occupation +
            s(maternal_age) + s(cholesterol) + smoking_status + center + race,
          family = binomial(link = 'logit'), data = dat)
summary(ga1)
```

```
##
## Call: gam(formula = ind_37 ~ s(dde) + s(PC1) + s(PC2) + s(PC3) + s(PC4) +
##     s(triglycerides) + score_education + score_income + score_occupation +
##     s(maternal_age) + s(cholesterol) + smoking_status + center +
##     race, family = binomial(link = "logit"), data = dat)
## Deviance Residuals:
##     Min      1Q  Median      3Q     Max
## -1.4927 -0.6149 -0.4745 -0.3430  2.7823
##
## (Dispersion Parameter for binomial family taken to be 1)
##
##     Null Deviance: 2025.59 on 2378 degrees of freedom
## Residual Deviance: 1862.921 on 2328.999 degrees of freedom
## AIC: 1962.923
##
## Number of Local Scoring Iterations: 8
##
## Anova for Parametric Effects
##                    Df  Sum Sq Mean Sq F value    Pr(>F)
```

```
## s(dde)              1   31.20 31.2046 30.7451 3.275e-08 ***
## s(PC1)              1    4.32  4.3213  4.2577 0.0391834 *
## s(PC2)              1    1.88  1.8821  1.8544 0.1734088
## s(PC3)              1    0.03  0.0315  0.0310 0.8601716
## s(PC4)              1    2.10  2.1044  2.0734 0.1500215
## s(triglycerides)    1    4.23  4.2262  4.1639 0.0414060 *
## score_education     1   11.65 11.6489 11.4773 0.0007162 ***
## score_income        1    2.54  2.5418  2.5043 0.1136692
## score_occupation    1    5.58  5.5828  5.5006 0.0190935 *
## s(maternal_age)     1    2.51  2.5096  2.4727 0.1159755
## s(cholesterol)      1    9.56  9.5613  9.4205 0.0021705 **
## smoking_status      1    0.74  0.7360  0.7252 0.3945438
## center             11   33.95  3.0864  3.0410 0.0004745 ***
## race                2    1.46  0.7291  0.7183 0.4876675
## Residuals        2329 2363.81  1.0149
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Anova for Nonparametric Effects
##                   Npar Df Npar Chisq  P(Chi)
## (Intercept)
## s(dde)                 3     1.8089 0.61302
## s(PC1)                 3     9.4233 0.02416 *
## s(PC2)                 3     6.9790 0.07257 .
## s(PC3)                 3     2.1757 0.53680
## s(PC4)                 3     5.2907 0.15171
## s(triglycerides)       3     2.7995 0.42362
## score_education
## score_income
## score_occupation
## s(maternal_age)        3    11.3180 0.01013 *
## s(cholesterol)         3     2.6233 0.45342
## smoking_status
## center
## race
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
par(mfrow = c(3,4))
plot(ga1, se = TRUE)
```



3.2 Effect of DDE and PC1

```
# Extract DDE effects
new_data <- new_data_add <- dat[1,]
dde_effect <- function(dde_test){
  new_data$dde <- dde_test
  pred_orig <- predict(ga1, new_data, type = "response")
  # response: prob, link: log odds
  new_data_add$dde <- dde_test + 0.001
  pred_add <- predict(ga1, new_data_add, type = "response")
  return ((pred_add-pred_orig) / 0.001)
}

summary(dat$dde)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## -1.3923 -0.6581 -0.2760  0.0000  0.3177  7.4362
```

```
Scale_DDE <- c(-1,2,4,6)
Percent_Prob_Change <- c(dde_effect(-1)*100,
                 dde_effect(2)*100,
                 dde_effect(4)*100,
                 dde_effect(6)*100)
df <- data.frame(Scale_DDE, Percent_Prob_Change)
```

```r
library(knitr)
kable(df)
```

Scale_DDE	Percent_Prob_Change
-1	1.4525613
2	1.2304967
4	0.3543148
6	0.7840376

```r
# Extract effects of PC1
new_data <- new_data_add <- dat[1,]
PC1_effect <- function(PC1_test){
  new_data$PC1 <- PC1_test
  pred_orig <- predict(ga1, new_data, type = "response")
  # response: prob, link: log odds
  new_data_add$PC1 <- PC1_test + 0.001
  pred_add <- predict(ga1, new_data_add, type = "response")
  return ((pred_add-pred_orig) / 0.001)
}

summary(dat$PC1)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## -3.8573 -1.7107 -0.5345  0.0000  1.0931 17.1345
```

```r
PC1_effect(-3)
```

```
##           1
## 0.003378139
```

```r
PC1_effect(3)
```

```
##           1
## 0.006787949
```

```r
PC1_effect(7)
```

```
##          1
## 0.02147304
```

```r
PC1_effect(10)
```

```
##          1
## 0.04950311
```

```r
PC1 <- c(-3,3,7,10)
Percent_Prob_Change <- c(PC1_effect(-3)*100,
                 PC1_effect(3)*100,
                 PC1_effect(7)*100,
                 PC1_effect(10)*100)
df <- data.frame(PC1, Percent_Prob_Change)

library(knitr)
kable(df)
```

PC1	Percent_Prob_Change
-3	0.3378139
3	0.6787949
7	2.1473038
10	4.9503108

4 Bayesian GAM

Bayesian Generalized Additive Model

$$g(Y_i) = \beta_0 + \sum_{j=1}^{m} f_j(x_{ij}) + \sum_{k=1}^{l} \beta_k z_{ik}$$

We add priors on the common regression coefficients, priors on the standard deviations of the smooth terms. The priors are set by default in *rstanarm* package, which is a weak informative normal prior.

4.1 Model results

```
library(rstanarm)
```

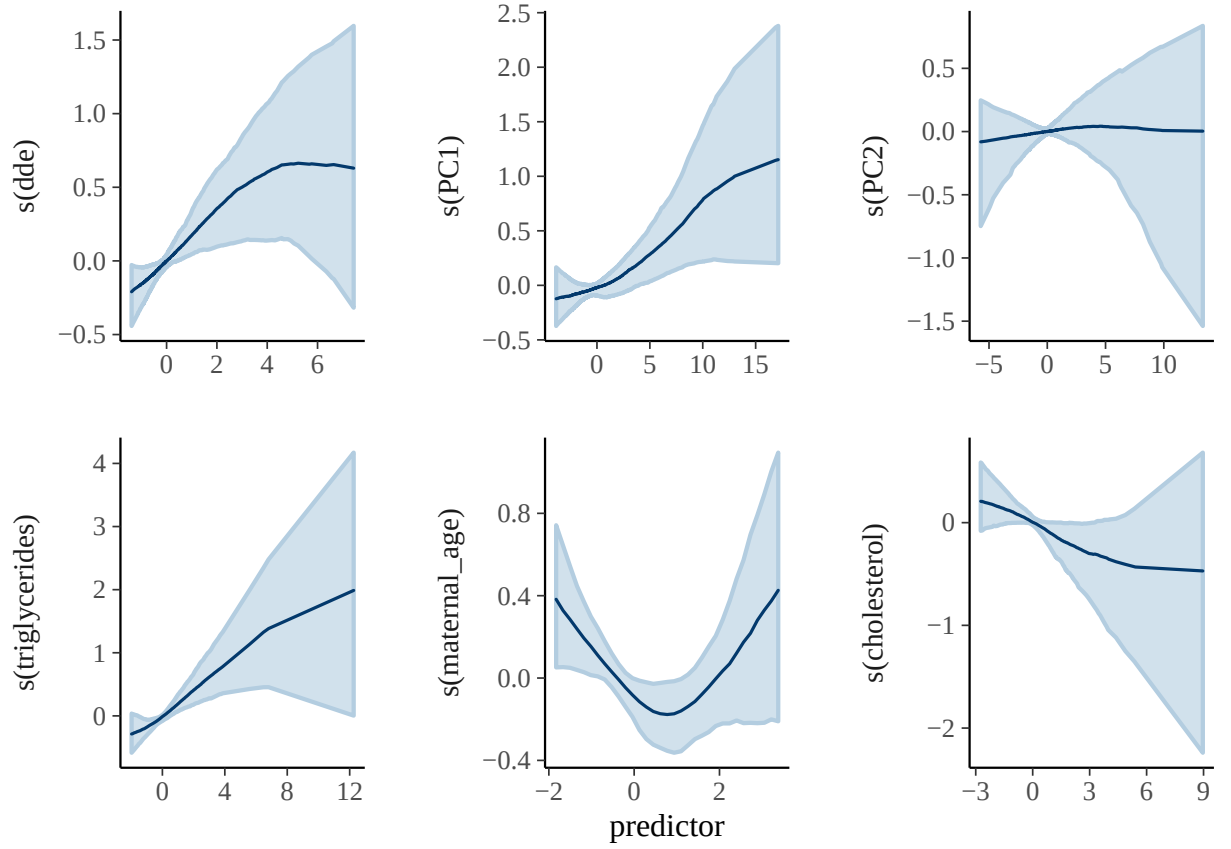
```
## Loading required package: Rcpp
## rstanarm (Version 2.19.2, packaged: 2019-10-01 20:20:33 UTC)
## - Do not expect the default priors to remain the same in future rstanarm versions.
## Thus, R scripts should specify priors explicitly, even if they are just the defaults.
## - For execution on a local, multicore CPU with excess RAM we recommend calling
## options(mc.cores = parallel::detectCores())
## - bayesplot theme set to bayesplot::theme_default()
##    * Does _not_ affect other ggplot2 plots
##    * See ?bayesplot_theme_set for details on theme setting
```

```
b_ga = stan_gamm4(ind_37 ~ s(dde) + s(PC1) + s(PC2)+
                    s(triglycerides) + race + score_education + score_income +
                    score_occupation + s(maternal_age) + smoking_status +
                    s(cholesterol) + center,
                  family = binomial(link = 'logit'), data = dat,
                  chain = 1, iter=1000)
```

```
##
## SAMPLING FOR MODEL 'bernoulli' NOW (CHAIN 1).
## Chain 1:
## Chain 1: Gradient evaluation took 0.001813 seconds
## Chain 1: 1000 transitions using 10 leapfrog steps per transition would take 18.13 seconds.
## Chain 1: Adjust your expectations accordingly!
## Chain 1:
## Chain 1:
## Chain 1: Iteration:   1 / 1000 [  0%]  (Warmup)
## Chain 1: Iteration: 100 / 1000 [ 10%]  (Warmup)
```

```
## Chain 1: Iteration: 200 / 1000 [ 20%]  (Warmup)
## Chain 1: Iteration: 300 / 1000 [ 30%]  (Warmup)
## Chain 1: Iteration: 400 / 1000 [ 40%]  (Warmup)
## Chain 1: Iteration: 500 / 1000 [ 50%]  (Warmup)
## Chain 1: Iteration: 501 / 1000 [ 50%]  (Sampling)
## Chain 1: Iteration: 600 / 1000 [ 60%]  (Sampling)
## Chain 1: Iteration: 700 / 1000 [ 70%]  (Sampling)
## Chain 1: Iteration: 800 / 1000 [ 80%]  (Sampling)
## Chain 1: Iteration: 900 / 1000 [ 90%]  (Sampling)
## Chain 1: Iteration: 1000 / 1000 [100%]  (Sampling)
## Chain 1:
## Chain 1:  Elapsed Time: 9.16323 seconds (Warm-up)
## Chain 1:                5.09929 seconds (Sampling)
## Chain 1:                14.2625 seconds (Total)
## Chain 1:
```

```
plot_nonlinear(b_ga)
```



4.2 Model check

```
library(arm)
```

```
## Loading required package: MASS
## Loading required package: Matrix
## Loading required package: lme4
```

```
##
## arm (Version 1.10-1, built: 2018-4-12)
## Working directory is /Users/yufeng/Developer/STA 723/case-study-1-team-1/report
```

```r
pred.probs_gam <-  predict(b_ga, dat, type = 'response') # Calculate predicted probabilities
resid_gam <- residuals(b_ga) # residuals

binnedplot(x = pred.probs_gam ,y = resid_gam, nclass=NULL,
           xlab="Predicted value", ylab="Average residual",
           main="Binned residual plot: residual vs estimated probabilities for gam",
           cex.pts=0.8, col.pts=1, col.int="gray")
```