# GAM air pollution

Afiqah

2023-04-07

## Work flow

### 1. Preparation

#### 1.1. Load package

#### 1.2. Load data

### 2. Data exploration

#### 2.1. Data wrangling

##### 2.1.1. Read and describe data

##### 2.1.2. Imputation of missing data

##### 2.1.3. Change variable format

##### 2.1.4. Visualize air pollutant data

##### 2.1.5. Visualize weather data

##### 2.1.6. Total respiratory diagnosis by city

##### 2.1.7. Ranking of respiratory diseases by period

##### 2.1.8. Air pollution by city, by period

##### 2.1.9. Respiratory diseases by city, by period

#### 2.2. Variable selection

##### 2.2.1. Correlation

#### 2.2.2. Comparison of air pollution and respiratory diseases between periods

#### 2.2.3. Univariate analysis

## 3. Association analysis (Generalized additive model)

### 3.1. Model selection and checking for assumptions

### 3.2. Kota Bharu analysis

### 3.3. Johor Bahru analysis

# Analysis

## 1. Preparation

### 1.1. Load package

```
library(mice)
```

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

## The following object is masked from 'package:stats':
##
##     filter

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

```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(ggplot2)
library(ggpubr)
library(lubridate)
```

```
##
## Attaching package: 'lubridate'

## The following objects are masked from 'package:base':
##
##     date, intersect, setdiff, union
```

```
library(gtsummary)
library(corrplot)
```

```
## corrplot 0.92 loaded
```

```
library(mgcv)
```

```
## Loading required package: nlme

##
## Attaching package: 'nlme'

## The following object is masked from 'package:dplyr':
##
##     collapse

## This is mgcv 1.8-41. For overview type 'help("mgcv-package")'.
```

```
library(ISLR)
```

```
## Warning: package 'ISLR' was built under R version 4.2.3
```

```
library(voxel)
```

```
## Warning: package 'voxel' was built under R version 4.2.3

## Loading required package: lmerTest

## Warning: package 'lmerTest' was built under R version 4.2.3

## Loading required package: lme4

## Loading required package: Matrix

##
## Attaching package: 'lme4'

## The following object is masked from 'package:nlme':
##
##     lmList
```

```
##
## Attaching package: 'lmerTest'

## The following object is masked from 'package:lme4':
##
##     lmer

## The following object is masked from 'package:stats':
##
##     step
```

```
library(gridExtra)
```

```
##
## Attaching package: 'gridExtra'

## The following object is masked from 'package:dplyr':
##
##     combine
```

```
library(mvmeta)
```

```
## Warning: package 'mvmeta' was built under R version 4.2.3

## This is mvmeta 1.0.3. For an overview type: help('mvmeta-package').
```

```
library(splines)
library(plot3D)
```

```
## Warning: package 'plot3D' was built under R version 4.2.3
```

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.2 --

## v tibble  3.1.7     v purrr   0.3.4
## v tidyr   1.2.0     v stringr 1.4.0
## v readr   2.1.2     v forcats 0.5.1
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x lubridate::as.difftime() masks base::as.difftime()
## x nlme::collapse()         masks dplyr::collapse()
## x gridExtra::combine()     masks dplyr::combine()
## x lubridate::date()        masks base::date()
## x tidyr::expand()          masks Matrix::expand()
## x dplyr::filter()          masks mice::filter(), stats::filter()
## x lubridate::intersect()   masks base::intersect()
## x dplyr::lag()             masks stats::lag()
## x tidyr::pack()            masks Matrix::pack()
## x lubridate::setdiff()     masks base::setdiff()
## x lubridate::union()       masks base::union()
## x tidyr::unpack()          masks Matrix::unpack()
```

```r
library(rstatix)
```

```
##
## Attaching package: 'rstatix'
##
## The following object is masked from 'package:stats':
##
##     filter
```

```r
library(foreign)
library(nnet)
```

```
##
## Attaching package: 'nnet'
##
## The following object is masked from 'package:mgcv':
##
##     multinom
```

```r
library(AER)
```

```
## Loading required package: car
## Loading required package: carData
##
## Attaching package: 'car'
##
## The following object is masked from 'package:purrr':
##
##     some
##
## The following object is masked from 'package:dplyr':
##
##     recode
##
## Loading required package: lmtest
## Loading required package: zoo
##
## Attaching package: 'zoo'
##
## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
##
## Loading required package: sandwich
## Loading required package: survival
```

```r
library(ggbump)
library(moments)
library(MASS)
```

```
##
```

```
## Attaching package: 'MASS'
##
## The following object is masked from 'package:rstatix':
##
##     select
##
## The following object is masked from 'package:gtsummary':
##
##     select
##
## The following object is masked from 'package:dplyr':
##
##     select
```

### 1.2. Load data

```r
data <- read.csv("citydata.csv")
```

## 2. Data exploration

## 2. Data exploration

### 2.1. Data wrangling

```r
glimpse(data)
```

#### 2.1.1. Read and describe data

```
## Rows: 3,654
## Columns: 24
## $ City                                                                  <chr> ~
## $ Date                                                                  <chr> ~
## $ DOW                                                                   <int> ~
## $ Period                                                                <int> ~
## $ ED                                                                    <int> ~
## $ A00.B99.Certain.infectious.and.parasitic.diseases                     <int> ~
## $ J00.J06_Acute.upper.respiratory.infections                            <int> ~
## $ J09.J18_Influenza.and.pneumonia                                       <int> ~
## $ J20.J22_Other.acute.lower.respiratory.infections                      <int> ~
## $ J40.J47_Chronic.lower.respiratory.diseases                            <int> ~
## $ R00.R09.Symptoms.and.signs.involving.the.circulatory.and.respiratory.systems <int> ~
## $ U07.1_COVID19                                                         <int> ~
## $ API                                                                   <dbl> ~
## $ PM10                                                                  <dbl> ~
## $ PM2.5                                                                 <dbl> ~
## $ SO2                                                                   <dbl> ~
## $ NO2                                                                   <dbl> ~
## $ CO3                                                                   <dbl> ~
```

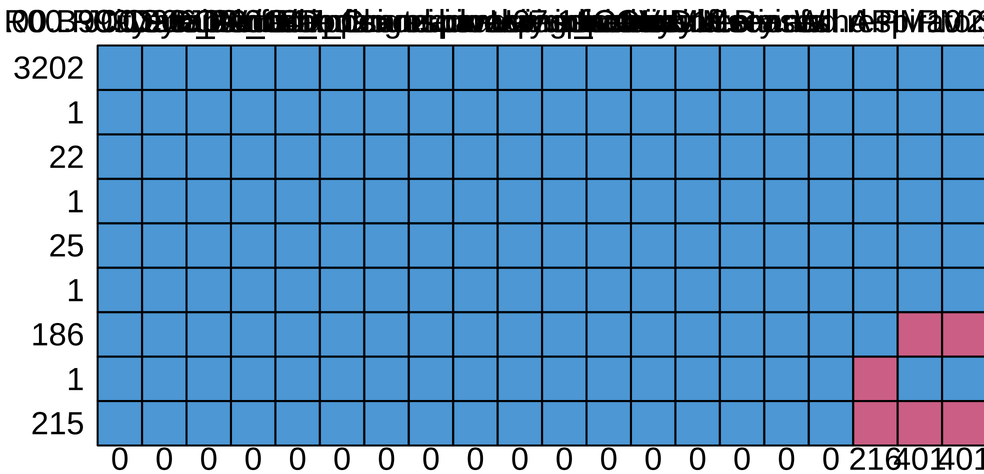
```
## $ O3                                                                                                <dbl> ~
## $ Max                                                                                               <dbl> ~
## $ Min                                                                                               <dbl> ~
## $ Mean                                                                                              <dbl> ~
## $ Rainfall                                                                                          <dbl> ~
## $ Wind                                                                                              <dbl> ~
```

```
summary(data)
```

```
##      City               Date                DOW        Period
##  Length:3654        Length:3654        Min.   :1   Min.   :1.00
##  Class :character   Class :character   1st Qu.:2   1st Qu.:1.00
##  Mode  :character   Mode  :character   Median :4   Median :1.00
##                                        Mean   :4   Mean   :1.44
##                                        3rd Qu.:6   3rd Qu.:2.00
##                                        Max.   :7   Max.   :3.00
##
##        ED        A00.B99.Certain.infectious.and.parasitic.diseases
##  Min.   : 0.00   Min.   :0.000000
##  1st Qu.: 6.00   1st Qu.:0.000000
##  Median :12.00   Median :0.000000
##  Mean   :14.42   Mean   :0.009031
##  3rd Qu.:20.00   3rd Qu.:0.000000
##  Max.   :86.00   Max.   :2.000000
##
##  J00.J06_Acute.upper.respiratory.infections J09.J18_Influenza.and.pneumonia
##  Min.   : 0.000                             Min.   : 0.000
##  1st Qu.: 1.000                             1st Qu.: 0.000
##  Median : 4.000                             Median : 1.000
##  Mean   : 8.194                             Mean   : 1.929
##  3rd Qu.:12.000                             3rd Qu.: 3.000
##  Max.   :75.000                             Max.   :16.000
##
##  J20.J22_Other.acute.lower.respiratory.infections
##  Min.   : 0.00
##  1st Qu.: 0.00
##  Median : 1.00
##  Mean   : 1.69
##  3rd Qu.: 3.00
##  Max.   :15.00
##
##  J40.J47_Chronic.lower.respiratory.diseases
##  Min.   : 0.000
##  1st Qu.: 0.000
##  Median : 1.000
##  Mean   : 1.519
##  3rd Qu.: 2.000
##  Max.   :13.000
##
##  R00.R09.Symptoms.and.signs.involving.the.circulatory.and.respiratory.systems
##  Min.   : 0.0000
##  1st Qu.: 0.0000
##  Median : 0.0000
##  Mean   : 0.9354
```

```
##  3rd Qu.: 1.0000
##  Max.   :12.0000
##
##  U07.1_COVID19          API              PM10             PM2.5
##  Min.   : 0.0000   Min.   : 13.75   Min.   :  5.684   Min.   :  2.504
##  1st Qu.: 0.0000   1st Qu.: 43.97   1st Qu.: 18.163   1st Qu.: 10.624
##  Median : 0.0000   Median : 53.33   Median : 24.144   Median : 15.103
##  Mean   : 0.1475   Mean   : 51.09   Mean   : 26.104   Mean   : 16.710
##  3rd Qu.: 0.0000   3rd Qu.: 58.25   3rd Qu.: 31.298   3rd Qu.: 20.598
##  Max.   :12.0000   Max.   :114.46   Max.   :126.936   Max.   :101.211
##                    NA's   :216      NA's   :401       NA's   :401
##       SO2              NO2              CO3               O3
##  Min.   :0.0001   Min.   :0.0004   Min.   :0.0012   Min.   :0.0028
##  1st Qu.:0.0006   1st Qu.:0.0033   1st Qu.:0.0136   1st Qu.:0.0131
##  Median :0.0010   Median :0.0049   Median :0.1450   Median :0.0410
##  Mean   :0.0011   Mean   :0.0075   Mean   :0.2464   Mean   :0.3354
##  3rd Qu.:0.0013   3rd Qu.:0.0113   3rd Qu.:0.4600   3rd Qu.:0.6282
##  Max.   :0.0107   Max.   :0.0341   Max.   :1.2333   Max.   :1.4083
##  NA's   :402      NA's   :448      NA's   :427      NA's   :449
##       Max             Min              Mean           Rainfall
##  Min.   :23.10   Min.   :19.50   Min.   :22.20   Min.   :  0.000
##  1st Qu.:30.80   1st Qu.:23.60   1st Qu.:26.60   1st Qu.:  0.000
##  Median :32.10   Median :24.10   Median :27.30   Median :  0.100
##  Mean   :31.85   Mean   :24.17   Mean   :27.26   Mean   :  7.223
##  3rd Qu.:33.10   3rd Qu.:24.70   3rd Qu.:28.00   3rd Qu.:  6.200
##  Max.   :36.10   Max.   :27.50   Max.   :30.80   Max.   :231.400
##
##       Wind
##  Min.   :0.50
##  1st Qu.:1.30
##  Median :1.80
##  Mean   :1.92
##  3rd Qu.:2.30
##  Max.   :8.50
##
```

```r
# Missing data pattern
md.pattern(data)
```




#### 2.1.2. Imputation of missing data

```
##      City Date DOW Period ED A00.B99.Certain.infectious.and.parasitic.diseases
## 3202    1    1   1      1  1                                                1
## 1       1    1   1      1  1                                                1
## 22      1    1   1      1  1                                                1
## 1       1    1   1      1  1                                                1
## 25      1    1   1      1  1                                                1
## 1       1    1   1      1  1                                                1
## 186     1    1   1      1  1                                                1
## 1       1    1   1      1  1                                                1
## 215     1    1   1      1  1                                                1
##         0    0   0      0  0                                                0
##      J00.J06_Acute.upper.respiratory.infections J09.J18_Influenza.and.pneumonia
## 3202                                          1                               1
## 1                                             1                               1
## 22                                            1                               1
## 1                                             1                               1
## 25                                            1                               1
## 1                                             1                               1
## 186                                           1                               1
## 1                                             1                               1
## 215                                           1                               1
##                                               0                               0
##      J20.J22_Other.acute.lower.respiratory.infections
## 3202                                                1
## 1                                                   1
## 22                                                  1
```

```
## 1                                                      1
## 25                                                     1
## 1                                                      1
## 186                                                    1
## 1                                                      1
## 215                                                    1
##                                                        0
##      J40.J47_Chronic.lower.respiratory.diseases
## 3202                                             1
## 1                                                1
## 22                                               1
## 1                                                1
## 25                                               1
## 1                                                1
## 186                                              1
## 1                                                1
## 215                                              1
##                                                  0
##      R00.R09.Symptoms.and.signs.involving.the.circulatory.and.respiratory.systems
## 3202                                                                            1
## 1                                                                               1
## 22                                                                              1
## 1                                                                               1
## 25                                                                              1
## 1                                                                               1
## 186                                                                             1
## 1                                                                               1
## 215                                                                             1
##                                                                                 0
##      U07.1_COVID19 Max Min Mean Rainfall Wind API PM10 PM2.5 SO2 CO3 NO2  O3
## 3202             1   1   1    1        1    1   1    1     1   1   1   1   1
## 1                1   1   1    1        1    1   1    1     1   1   1   1   0
## 22               1   1   1    1        1    1   1    1     1   1   1   0   0
## 1                1   1   1    1        1    1   1    1     1   1   0   1   1
## 25               1   1   1    1        1    1   1    1     1   1   0   0   0
## 1                1   1   1    1        1    1   1    1     1   0   1   1   1
## 186              1   1   1    1        1    1   1    0     0   0   0   0   0
## 1                1   1   1    1        1    1   0    1     1   1   1   1   1
## 215              1   1   1    1        1    1   0    0     0   0   0   0   0
##                  0   0   0    0        0    0 216  401   401 402 427 448 449
##
## 3202    0
## 1       1
## 22      2
## 1       1
## 25      3
## 1       1
## 186     6
## 1       1
## 215     7
##      2744
```

```
# Imputation of missing data
tempData <- mice(data,m=5,maxit=50,meth='pmm',seed=500)
```

```
##
##  iter imp variable
##   1   1  API  PM10  PM2.5  SO2  NO2  CO3  O3
##   1   2  API  PM10  PM2.5  SO2  NO2  CO3  O3
##   1   3  API  PM10  PM2.5  SO2  NO2  CO3  O3
##   1   4  API  PM10  PM2.5  SO2  NO2  CO3  O3
##   1   5  API  PM10  PM2.5  SO2  NO2  CO3  O3
##   2   1  API  PM10  PM2.5  SO2  NO2  CO3  O3
##   2   2  API  PM10  PM2.5  SO2  NO2  CO3  O3
##   2   3  API  PM10  PM2.5  SO2  NO2  CO3  O3
##   2   4  API  PM10  PM2.5  SO2  NO2  CO3  O3
##   2   5  API  PM10  PM2.5  SO2  NO2  CO3  O3
##   3   1  API  PM10  PM2.5  SO2  NO2  CO3  O3
##   3   2  API  PM10  PM2.5  SO2  NO2  CO3  O3
##   3   3  API  PM10  PM2.5  SO2  NO2  CO3  O3
##   3   4  API  PM10  PM2.5  SO2  NO2  CO3  O3
##   3   5  API  PM10  PM2.5  SO2  NO2  CO3  O3
##   4   1  API  PM10  PM2.5  SO2  NO2  CO3  O3
##   4   2  API  PM10  PM2.5  SO2  NO2  CO3  O3
##   4   3  API  PM10  PM2.5  SO2  NO2  CO3  O3
##   4   4  API  PM10  PM2.5  SO2  NO2  CO3  O3
##   4   5  API  PM10  PM2.5  SO2  NO2  CO3  O3
##   5   1  API  PM10  PM2.5  SO2  NO2  CO3  O3
##   5   2  API  PM10  PM2.5  SO2  NO2  CO3  O3
##   5   3  API  PM10  PM2.5  SO2  NO2  CO3  O3
##   5   4  API  PM10  PM2.5  SO2  NO2  CO3  O3
##   5   5  API  PM10  PM2.5  SO2  NO2  CO3  O3
##   6   1  API  PM10  PM2.5  SO2  NO2  CO3  O3
##   6   2  API  PM10  PM2.5  SO2  NO2  CO3  O3
##   6   3  API  PM10  PM2.5  SO2  NO2  CO3  O3
##   6   4  API  PM10  PM2.5  SO2  NO2  CO3  O3
##   6   5  API  PM10  PM2.5  SO2  NO2  CO3  O3
##   7   1  API  PM10  PM2.5  SO2  NO2  CO3  O3
##   7   2  API  PM10  PM2.5  SO2  NO2  CO3  O3
##   7   3  API  PM10  PM2.5  SO2  NO2  CO3  O3
##   7   4  API  PM10  PM2.5  SO2  NO2  CO3  O3
##   7   5  API  PM10  PM2.5  SO2  NO2  CO3  O3
##   8   1  API  PM10  PM2.5  SO2  NO2  CO3  O3
##   8   2  API  PM10  PM2.5  SO2  NO2  CO3  O3
##   8   3  API  PM10  PM2.5  SO2  NO2  CO3  O3
##   8   4  API  PM10  PM2.5  SO2  NO2  CO3  O3
##   8   5  API  PM10  PM2.5  SO2  NO2  CO3  O3
##   9   1  API  PM10  PM2.5  SO2  NO2  CO3  O3
##   9   2  API  PM10  PM2.5  SO2  NO2  CO3  O3
##   9   3  API  PM10  PM2.5  SO2  NO2  CO3  O3
##   9   4  API  PM10  PM2.5  SO2  NO2  CO3  O3
##   9   5  API  PM10  PM2.5  SO2  NO2  CO3  O3
##   10   1  API  PM10  PM2.5  SO2  NO2  CO3  O3
##   10   2  API  PM10  PM2.5  SO2  NO2  CO3  O3
##   10   3  API  PM10  PM2.5  SO2  NO2  CO3  O3
##   10   4  API  PM10  PM2.5  SO2  NO2  CO3  O3
##   10   5  API  PM10  PM2.5  SO2  NO2  CO3  O3
##   11   1  API  PM10  PM2.5  SO2  NO2  CO3  O3
##   11   2  API  PM10  PM2.5  SO2  NO2  CO3  O3
```

```
## 11 3 API PM10 PM2.5 SO2 NO2 CO3 O3
## 11 4 API PM10 PM2.5 SO2 NO2 CO3 O3
## 11 5 API PM10 PM2.5 SO2 NO2 CO3 O3
## 12 1 API PM10 PM2.5 SO2 NO2 CO3 O3
## 12 2 API PM10 PM2.5 SO2 NO2 CO3 O3
## 12 3 API PM10 PM2.5 SO2 NO2 CO3 O3
## 12 4 API PM10 PM2.5 SO2 NO2 CO3 O3
## 12 5 API PM10 PM2.5 SO2 NO2 CO3 O3
## 13 1 API PM10 PM2.5 SO2 NO2 CO3 O3
## 13 2 API PM10 PM2.5 SO2 NO2 CO3 O3
## 13 3 API PM10 PM2.5 SO2 NO2 CO3 O3
## 13 4 API PM10 PM2.5 SO2 NO2 CO3 O3
## 13 5 API PM10 PM2.5 SO2 NO2 CO3 O3
## 14 1 API PM10 PM2.5 SO2 NO2 CO3 O3
## 14 2 API PM10 PM2.5 SO2 NO2 CO3 O3
## 14 3 API PM10 PM2.5 SO2 NO2 CO3 O3
## 14 4 API PM10 PM2.5 SO2 NO2 CO3 O3
## 14 5 API PM10 PM2.5 SO2 NO2 CO3 O3
## 15 1 API PM10 PM2.5 SO2 NO2 CO3 O3
## 15 2 API PM10 PM2.5 SO2 NO2 CO3 O3
## 15 3 API PM10 PM2.5 SO2 NO2 CO3 O3
## 15 4 API PM10 PM2.5 SO2 NO2 CO3 O3
## 15 5 API PM10 PM2.5 SO2 NO2 CO3 O3
## 16 1 API PM10 PM2.5 SO2 NO2 CO3 O3
## 16 2 API PM10 PM2.5 SO2 NO2 CO3 O3
## 16 3 API PM10 PM2.5 SO2 NO2 CO3 O3
## 16 4 API PM10 PM2.5 SO2 NO2 CO3 O3
## 16 5 API PM10 PM2.5 SO2 NO2 CO3 O3
## 17 1 API PM10 PM2.5 SO2 NO2 CO3 O3
## 17 2 API PM10 PM2.5 SO2 NO2 CO3 O3
## 17 3 API PM10 PM2.5 SO2 NO2 CO3 O3
## 17 4 API PM10 PM2.5 SO2 NO2 CO3 O3
## 17 5 API PM10 PM2.5 SO2 NO2 CO3 O3
## 18 1 API PM10 PM2.5 SO2 NO2 CO3 O3
## 18 2 API PM10 PM2.5 SO2 NO2 CO3 O3
## 18 3 API PM10 PM2.5 SO2 NO2 CO3 O3
## 18 4 API PM10 PM2.5 SO2 NO2 CO3 O3
## 18 5 API PM10 PM2.5 SO2 NO2 CO3 O3
## 19 1 API PM10 PM2.5 SO2 NO2 CO3 O3
## 19 2 API PM10 PM2.5 SO2 NO2 CO3 O3
## 19 3 API PM10 PM2.5 SO2 NO2 CO3 O3
## 19 4 API PM10 PM2.5 SO2 NO2 CO3 O3
## 19 5 API PM10 PM2.5 SO2 NO2 CO3 O3
## 20 1 API PM10 PM2.5 SO2 NO2 CO3 O3
## 20 2 API PM10 PM2.5 SO2 NO2 CO3 O3
## 20 3 API PM10 PM2.5 SO2 NO2 CO3 O3
## 20 4 API PM10 PM2.5 SO2 NO2 CO3 O3
## 20 5 API PM10 PM2.5 SO2 NO2 CO3 O3
## 21 1 API PM10 PM2.5 SO2 NO2 CO3 O3
## 21 2 API PM10 PM2.5 SO2 NO2 CO3 O3
## 21 3 API PM10 PM2.5 SO2 NO2 CO3 O3
## 21 4 API PM10 PM2.5 SO2 NO2 CO3 O3
## 21 5 API PM10 PM2.5 SO2 NO2 CO3 O3
## 22 1 API PM10 PM2.5 SO2 NO2 CO3 O3
```

```
## 22 2 API PM10 PM2.5 SO2 NO2 CO3 O3
## 22 3 API PM10 PM2.5 SO2 NO2 CO3 O3
## 22 4 API PM10 PM2.5 SO2 NO2 CO3 O3
## 22 5 API PM10 PM2.5 SO2 NO2 CO3 O3
## 23 1 API PM10 PM2.5 SO2 NO2 CO3 O3
## 23 2 API PM10 PM2.5 SO2 NO2 CO3 O3
## 23 3 API PM10 PM2.5 SO2 NO2 CO3 O3
## 23 4 API PM10 PM2.5 SO2 NO2 CO3 O3
## 23 5 API PM10 PM2.5 SO2 NO2 CO3 O3
## 24 1 API PM10 PM2.5 SO2 NO2 CO3 O3
## 24 2 API PM10 PM2.5 SO2 NO2 CO3 O3
## 24 3 API PM10 PM2.5 SO2 NO2 CO3 O3
## 24 4 API PM10 PM2.5 SO2 NO2 CO3 O3
## 24 5 API PM10 PM2.5 SO2 NO2 CO3 O3
## 25 1 API PM10 PM2.5 SO2 NO2 CO3 O3
## 25 2 API PM10 PM2.5 SO2 NO2 CO3 O3
## 25 3 API PM10 PM2.5 SO2 NO2 CO3 O3
## 25 4 API PM10 PM2.5 SO2 NO2 CO3 O3
## 25 5 API PM10 PM2.5 SO2 NO2 CO3 O3
## 26 1 API PM10 PM2.5 SO2 NO2 CO3 O3
## 26 2 API PM10 PM2.5 SO2 NO2 CO3 O3
## 26 3 API PM10 PM2.5 SO2 NO2 CO3 O3
## 26 4 API PM10 PM2.5 SO2 NO2 CO3 O3
## 26 5 API PM10 PM2.5 SO2 NO2 CO3 O3
## 27 1 API PM10 PM2.5 SO2 NO2 CO3 O3
## 27 2 API PM10 PM2.5 SO2 NO2 CO3 O3
## 27 3 API PM10 PM2.5 SO2 NO2 CO3 O3
## 27 4 API PM10 PM2.5 SO2 NO2 CO3 O3
## 27 5 API PM10 PM2.5 SO2 NO2 CO3 O3
## 28 1 API PM10 PM2.5 SO2 NO2 CO3 O3
## 28 2 API PM10 PM2.5 SO2 NO2 CO3 O3
## 28 3 API PM10 PM2.5 SO2 NO2 CO3 O3
## 28 4 API PM10 PM2.5 SO2 NO2 CO3 O3
## 28 5 API PM10 PM2.5 SO2 NO2 CO3 O3
## 29 1 API PM10 PM2.5 SO2 NO2 CO3 O3
## 29 2 API PM10 PM2.5 SO2 NO2 CO3 O3
## 29 3 API PM10 PM2.5 SO2 NO2 CO3 O3
## 29 4 API PM10 PM2.5 SO2 NO2 CO3 O3
## 29 5 API PM10 PM2.5 SO2 NO2 CO3 O3
## 30 1 API PM10 PM2.5 SO2 NO2 CO3 O3
## 30 2 API PM10 PM2.5 SO2 NO2 CO3 O3
## 30 3 API PM10 PM2.5 SO2 NO2 CO3 O3
## 30 4 API PM10 PM2.5 SO2 NO2 CO3 O3
## 30 5 API PM10 PM2.5 SO2 NO2 CO3 O3
## 31 1 API PM10 PM2.5 SO2 NO2 CO3 O3
## 31 2 API PM10 PM2.5 SO2 NO2 CO3 O3
## 31 3 API PM10 PM2.5 SO2 NO2 CO3 O3
## 31 4 API PM10 PM2.5 SO2 NO2 CO3 O3
## 31 5 API PM10 PM2.5 SO2 NO2 CO3 O3
## 32 1 API PM10 PM2.5 SO2 NO2 CO3 O3
## 32 2 API PM10 PM2.5 SO2 NO2 CO3 O3
## 32 3 API PM10 PM2.5 SO2 NO2 CO3 O3
## 32 4 API PM10 PM2.5 SO2 NO2 CO3 O3
## 32 5 API PM10 PM2.5 SO2 NO2 CO3 O3
```

```
## 33 1 API PM10 PM2.5 SO2 NO2 CO3 O3
## 33 2 API PM10 PM2.5 SO2 NO2 CO3 O3
## 33 3 API PM10 PM2.5 SO2 NO2 CO3 O3
## 33 4 API PM10 PM2.5 SO2 NO2 CO3 O3
## 33 5 API PM10 PM2.5 SO2 NO2 CO3 O3
## 34 1 API PM10 PM2.5 SO2 NO2 CO3 O3
## 34 2 API PM10 PM2.5 SO2 NO2 CO3 O3
## 34 3 API PM10 PM2.5 SO2 NO2 CO3 O3
## 34 4 API PM10 PM2.5 SO2 NO2 CO3 O3
## 34 5 API PM10 PM2.5 SO2 NO2 CO3 O3
## 35 1 API PM10 PM2.5 SO2 NO2 CO3 O3
## 35 2 API PM10 PM2.5 SO2 NO2 CO3 O3
## 35 3 API PM10 PM2.5 SO2 NO2 CO3 O3
## 35 4 API PM10 PM2.5 SO2 NO2 CO3 O3
## 35 5 API PM10 PM2.5 SO2 NO2 CO3 O3
## 36 1 API PM10 PM2.5 SO2 NO2 CO3 O3
## 36 2 API PM10 PM2.5 SO2 NO2 CO3 O3
## 36 3 API PM10 PM2.5 SO2 NO2 CO3 O3
## 36 4 API PM10 PM2.5 SO2 NO2 CO3 O3
## 36 5 API PM10 PM2.5 SO2 NO2 CO3 O3
## 37 1 API PM10 PM2.5 SO2 NO2 CO3 O3
## 37 2 API PM10 PM2.5 SO2 NO2 CO3 O3
## 37 3 API PM10 PM2.5 SO2 NO2 CO3 O3
## 37 4 API PM10 PM2.5 SO2 NO2 CO3 O3
## 37 5 API PM10 PM2.5 SO2 NO2 CO3 O3
## 38 1 API PM10 PM2.5 SO2 NO2 CO3 O3
## 38 2 API PM10 PM2.5 SO2 NO2 CO3 O3
## 38 3 API PM10 PM2.5 SO2 NO2 CO3 O3
## 38 4 API PM10 PM2.5 SO2 NO2 CO3 O3
## 38 5 API PM10 PM2.5 SO2 NO2 CO3 O3
## 39 1 API PM10 PM2.5 SO2 NO2 CO3 O3
## 39 2 API PM10 PM2.5 SO2 NO2 CO3 O3
## 39 3 API PM10 PM2.5 SO2 NO2 CO3 O3
## 39 4 API PM10 PM2.5 SO2 NO2 CO3 O3
## 39 5 API PM10 PM2.5 SO2 NO2 CO3 O3
## 40 1 API PM10 PM2.5 SO2 NO2 CO3 O3
## 40 2 API PM10 PM2.5 SO2 NO2 CO3 O3
## 40 3 API PM10 PM2.5 SO2 NO2 CO3 O3
## 40 4 API PM10 PM2.5 SO2 NO2 CO3 O3
## 40 5 API PM10 PM2.5 SO2 NO2 CO3 O3
## 41 1 API PM10 PM2.5 SO2 NO2 CO3 O3
## 41 2 API PM10 PM2.5 SO2 NO2 CO3 O3
## 41 3 API PM10 PM2.5 SO2 NO2 CO3 O3
## 41 4 API PM10 PM2.5 SO2 NO2 CO3 O3
## 41 5 API PM10 PM2.5 SO2 NO2 CO3 O3
## 42 1 API PM10 PM2.5 SO2 NO2 CO3 O3
## 42 2 API PM10 PM2.5 SO2 NO2 CO3 O3
## 42 3 API PM10 PM2.5 SO2 NO2 CO3 O3
## 42 4 API PM10 PM2.5 SO2 NO2 CO3 O3
## 42 5 API PM10 PM2.5 SO2 NO2 CO3 O3
## 43 1 API PM10 PM2.5 SO2 NO2 CO3 O3
## 43 2 API PM10 PM2.5 SO2 NO2 CO3 O3
## 43 3 API PM10 PM2.5 SO2 NO2 CO3 O3
## 43 4 API PM10 PM2.5 SO2 NO2 CO3 O3
```

```
##   43   5  API  PM10  PM2.5  SO2  NO2  CO3  O3
##   44   1  API  PM10  PM2.5  SO2  NO2  CO3  O3
##   44   2  API  PM10  PM2.5  SO2  NO2  CO3  O3
##   44   3  API  PM10  PM2.5  SO2  NO2  CO3  O3
##   44   4  API  PM10  PM2.5  SO2  NO2  CO3  O3
##   44   5  API  PM10  PM2.5  SO2  NO2  CO3  O3
##   45   1  API  PM10  PM2.5  SO2  NO2  CO3  O3
##   45   2  API  PM10  PM2.5  SO2  NO2  CO3  O3
##   45   3  API  PM10  PM2.5  SO2  NO2  CO3  O3
##   45   4  API  PM10  PM2.5  SO2  NO2  CO3  O3
##   45   5  API  PM10  PM2.5  SO2  NO2  CO3  O3
##   46   1  API  PM10  PM2.5  SO2  NO2  CO3  O3
##   46   2  API  PM10  PM2.5  SO2  NO2  CO3  O3
##   46   3  API  PM10  PM2.5  SO2  NO2  CO3  O3
##   46   4  API  PM10  PM2.5  SO2  NO2  CO3  O3
##   46   5  API  PM10  PM2.5  SO2  NO2  CO3  O3
##   47   1  API  PM10  PM2.5  SO2  NO2  CO3  O3
##   47   2  API  PM10  PM2.5  SO2  NO2  CO3  O3
##   47   3  API  PM10  PM2.5  SO2  NO2  CO3  O3
##   47   4  API  PM10  PM2.5  SO2  NO2  CO3  O3
##   47   5  API  PM10  PM2.5  SO2  NO2  CO3  O3
##   48   1  API  PM10  PM2.5  SO2  NO2  CO3  O3
##   48   2  API  PM10  PM2.5  SO2  NO2  CO3  O3
##   48   3  API  PM10  PM2.5  SO2  NO2  CO3  O3
##   48   4  API  PM10  PM2.5  SO2  NO2  CO3  O3
##   48   5  API  PM10  PM2.5  SO2  NO2  CO3  O3
##   49   1  API  PM10  PM2.5  SO2  NO2  CO3  O3
##   49   2  API  PM10  PM2.5  SO2  NO2  CO3  O3
##   49   3  API  PM10  PM2.5  SO2  NO2  CO3  O3
##   49   4  API  PM10  PM2.5  SO2  NO2  CO3  O3
##   49   5  API  PM10  PM2.5  SO2  NO2  CO3  O3
##   50   1  API  PM10  PM2.5  SO2  NO2  CO3  O3
##   50   2  API  PM10  PM2.5  SO2  NO2  CO3  O3
##   50   3  API  PM10  PM2.5  SO2  NO2  CO3  O3
##   50   4  API  PM10  PM2.5  SO2  NO2  CO3  O3
##   50   5  API  PM10  PM2.5  SO2  NO2  CO3  O3
```

```
## Warning: Number of logged events: 1252
```

```
summary(tempData)
```

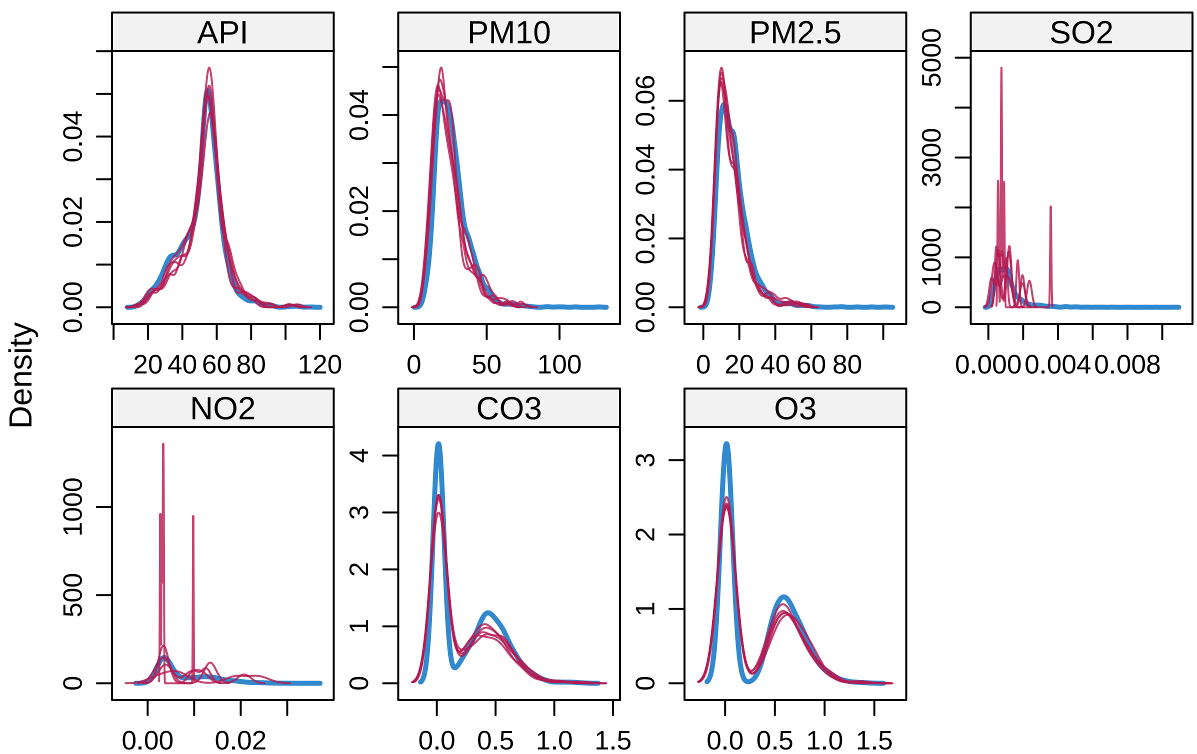
```
## Class: mids
## Number of multiple imputations:  5
## Imputation methods:
##                                                                      City
##                                                                        ""
##                                                                      Date
##                                                                        ""
##                                                                       DOW
##                                                                        ""
##                                                                    Period
##                                                                        ""
##                                                                        ED
```

```
##                                                                                ""
##                           A00.B99.Certain.infectious.and.parasitic.diseases
##                                                                                ""
##                                  J00.J06_Acute.upper.respiratory.infections
##                                                                                ""
##                                             J09.J18_Influenza.and.pneumonia
##                                                                                ""
##                            J20.J22_Other.acute.lower.respiratory.infections
##                                                                                ""
##                                  J40.J47_Chronic.lower.respiratory.diseases
##                                                                                ""
## R00.R09.Symptoms.and.signs.involving.the.circulatory.and.respiratory.systems
##                                                                                ""
##                                                                 U07.1_COVID19
##                                                                                ""
##                                                                           API
##                                                                         "pmm"
##                                                                          PM10
##                                                                         "pmm"
##                                                                         PM2.5
##                                                                         "pmm"
##                                                                           SO2
##                                                                         "pmm"
##                                                                           NO2
##                                                                         "pmm"
##                                                                           CO3
##                                                                         "pmm"
##                                                                            O3
##                                                                         "pmm"
##                                                                           Max
##                                                                            ""
##                                                                           Min
##                                                                            ""
##                                                                          Mean
##                                                                            ""
##                                                                      Rainfall
##                                                                            ""
##                                                                          Wind
##                                                                            ""
## PredictorMatrix:
##                                                   City Date DOW Period ED
## City                                                 0    0   1      1  1
## Date                                                 0    0   1      1  1
## DOW                                                  0    0   0      1  1
## Period                                               0    0   1      0  1
## ED                                                   0    0   1      1  0
## A00.B99.Certain.infectious.and.parasitic.diseases    0    0   1      1  1
##                                                   A00.B99.Certain.infectious.and.parasitic.diseases
## City                                                                                              1
## Date                                                                                              1
## DOW                                                                                               1
## Period                                                                                            1
## ED                                                                                                1
## A00.B99.Certain.infectious.and.parasitic.diseases                                                 0
```

```
##                                                   J00.J06_Acute.upper.respiratory.infections
## City                                                                                       1
## Date                                                                                       1
## DOW                                                                                        1
## Period                                                                                     1
## ED                                                                                         1
## A00.B99.Certain.infectious.and.parasitic.diseases                                          1
##                                                   J09.J18_Influenza.and.pneumonia
## City                                                                            1
## Date                                                                            1
## DOW                                                                             1
## Period                                                                          1
## ED                                                                              1
## A00.B99.Certain.infectious.and.parasitic.diseases                               1
##                                                   J20.J22_Other.acute.lower.respiratory.infections
## City                                                                                             1
## Date                                                                                             1
## DOW                                                                                              1
## Period                                                                                           1
## ED                                                                                               1
## A00.B99.Certain.infectious.and.parasitic.diseases                                                1
##                                                   J40.J47_Chronic.lower.respiratory.diseases
## City                                                                                       1
## Date                                                                                       1
## DOW                                                                                        1
## Period                                                                                     1
## ED                                                                                         1
## A00.B99.Certain.infectious.and.parasitic.diseases                                          1
##                                                   R00.R09.Symptoms.and.signs.involving.the.circulator
## City
## Date
## DOW
## Period
## ED
## A00.B99.Certain.infectious.and.parasitic.diseases
##                                                   U07.1_COVID19 API PM10 PM2.5
## City                                                          1   1    1     1
## Date                                                          1   1    1     1
## DOW                                                           1   1    1     1
## Period                                                        1   1    1     1
## ED                                                            1   1    1     1
## A00.B99.Certain.infectious.and.parasitic.diseases             1   1    1     1
##                                                   SO2 NO2 CO3 O3 Max Min Mean
## City                                                1   1   1  1   1   1    1
## Date                                                1   1   1  1   1   1    1
## DOW                                                 1   1   1  1   1   1    1
## Period                                              1   1   1  1   1   1    1
## ED                                                  1   1   1  1   1   1    1
## A00.B99.Certain.infectious.and.parasitic.diseases   1   1   1  1   1   1    1
##                                                   Rainfall Wind
## City                                                     1    1
## Date                                                     1    1
## DOW                                                      1    1
## Period                                                   1    1
```

```
## ED                                                          1    1
## A00.B99.Certain.infectious.and.parasitic.diseases           1    1
## Number of logged events:  1252
##   it im   dep      meth          out
## 1  0  0        constant         City
## 2  0  0        constant         Date
## 3  1  1   API       pmm ED, SO2, NO2
## 4  1  1  PM10       pmm ED, SO2, NO2
## 5  1  1 PM2.5       pmm ED, SO2, NO2
## 6  1  1   CO3       pmm ED, SO2, NO2
```

```
# Explore imputated data
densityplot(tempData)
```




```
#New dataframe
data2 <- complete(tempData,1)

summary(data2)
```

```
##      City               Date                DOW         Period
##  Length:3654        Length:3654        Min.   :1   Min.   :1.00
##  Class :character   Class :character   1st Qu.:2   1st Qu.:1.00
##  Mode  :character   Mode  :character   Median :4   Median :1.00
##                                        Mean   :4   Mean   :1.44
##                                        3rd Qu.:6   3rd Qu.:2.00
```

```
##                                         Max.   :7   Max.   :3.00
##        ED        A00.B99.Certain.infectious.and.parasitic.diseases
##  Min.   : 0.00   Min.   :0.000000
##  1st Qu.: 6.00   1st Qu.:0.000000
##  Median :12.00   Median :0.000000
##  Mean   :14.42   Mean   :0.009031
##  3rd Qu.:20.00   3rd Qu.:0.000000
##  Max.   :86.00   Max.   :2.000000
##  J00.J06_Acute.upper.respiratory.infections J09.J18_Influenza.and.pneumonia
##  Min.   : 0.000                             Min.   : 0.000
##  1st Qu.: 1.000                             1st Qu.: 0.000
##  Median : 4.000                             Median : 1.000
##  Mean   : 8.194                             Mean   : 1.929
##  3rd Qu.:12.000                             3rd Qu.: 3.000
##  Max.   :75.000                             Max.   :16.000
##  J20.J22_Other.acute.lower.respiratory.infections
##  Min.   : 0.00
##  1st Qu.: 0.00
##  Median : 1.00
##  Mean   : 1.69
##  3rd Qu.: 3.00
##  Max.   :15.00
##  J40.J47_Chronic.lower.respiratory.diseases
##  Min.   : 0.000
##  1st Qu.: 0.000
##  Median : 1.000
##  Mean   : 1.519
##  3rd Qu.: 2.000
##  Max.   :13.000
##  R00.R09.Symptoms.and.signs.involving.the.circulatory.and.respiratory.systems
##  Min.   : 0.0000
##  1st Qu.: 0.0000
##  Median : 0.0000
##  Mean   : 0.9354
##  3rd Qu.: 1.0000
##  Max.   :12.0000
##  U07.1_COVID19          API              PM10             PM2.5
##  Min.   : 0.0000   Min.   : 13.75   Min.   :  5.684   Min.   :  2.504
##  1st Qu.: 0.0000   1st Qu.: 44.25   1st Qu.: 17.952   1st Qu.: 10.505
##  Median : 0.0000   Median : 53.38   Median : 23.843   Median : 14.814
##  Mean   : 0.1475   Mean   : 51.20   Mean   : 25.875   Mean   : 16.556
##  3rd Qu.: 0.0000   3rd Qu.: 58.38   3rd Qu.: 31.094   3rd Qu.: 20.324
##  Max.   :12.0000   Max.   :114.46   Max.   :126.936   Max.   :101.211
##       SO2                  NO2                 CO3                 O3
##  Min.   :0.0000900   Min.   :0.0004333   Min.   :0.001213   Min.   :0.002782
##  1st Qu.:0.0006174   1st Qu.:0.0033660   1st Qu.:0.013501   1st Qu.:0.013127
##  Median :0.0009957   Median :0.0052478   Median :0.039322   Median :0.029000
##  Mean   :0.0011208   Mean   :0.0080263   Mean   :0.244358   Mean   :0.330319
##  3rd Qu.:0.0013000   3rd Qu.:0.0120989   3rd Qu.:0.458033   3rd Qu.:0.625544
##  Max.   :0.0106826   Max.   :0.0340727   Max.   :1.233261   Max.   :1.408348
##       Max             Min             Mean          Rainfall
##  Min.   :23.10   Min.   :19.50   Min.   :22.20   Min.   :  0.000
##  1st Qu.:30.80   1st Qu.:23.60   1st Qu.:26.60   1st Qu.:  0.000
##  Median :32.10   Median :24.10   Median :27.30   Median :  0.100
```

```
## Mean   :31.85   Mean   :24.17   Mean   :27.26   Mean   :  7.223
## 3rd Qu.:33.10   3rd Qu.:24.70   3rd Qu.:28.00   3rd Qu.:  6.200
## Max.   :36.10   Max.   :27.50   Max.   :30.80   Max.   :231.400
##      Wind
## Min.   :0.50
## 1st Qu.:1.30
## Median :1.80
## Mean   :1.92
## 3rd Qu.:2.30
## Max.   :8.50
```

```
data2$Date <- dmy(data2$Date)

data2$Period <- as.factor(data2$Period)
data2$City <- as.factor(data2$City)

data2$Infection <- data2$A00.B99.Certain.infectious.and.parasitic.diseases
data2$URTI <- data2$J00.J06_Acute.upper.respiratory.infections
data2$Pneumonia <- data2$J09.J18_Influenza.and.pneumonia
data2$LRTI <- data2$J20.J22_Other.acute.lower.respiratory.infections
data2$CLD <- data2$J40.J47_Chronic.lower.respiratory.diseases
data2$Pleura <- data2$J90.J94.Other.diseases.of.the.pleura
data2$Symptoms_Signs <- data2$R00.R09.Symptoms.and.signs.involving.the.circulatory.and.respiratory.syst
data2$COVID <- data2$U07.1_COVID19

kbdata <- data2[data2$City=='KB',]
jbdata <- data2[data2$City=='JB',]
```

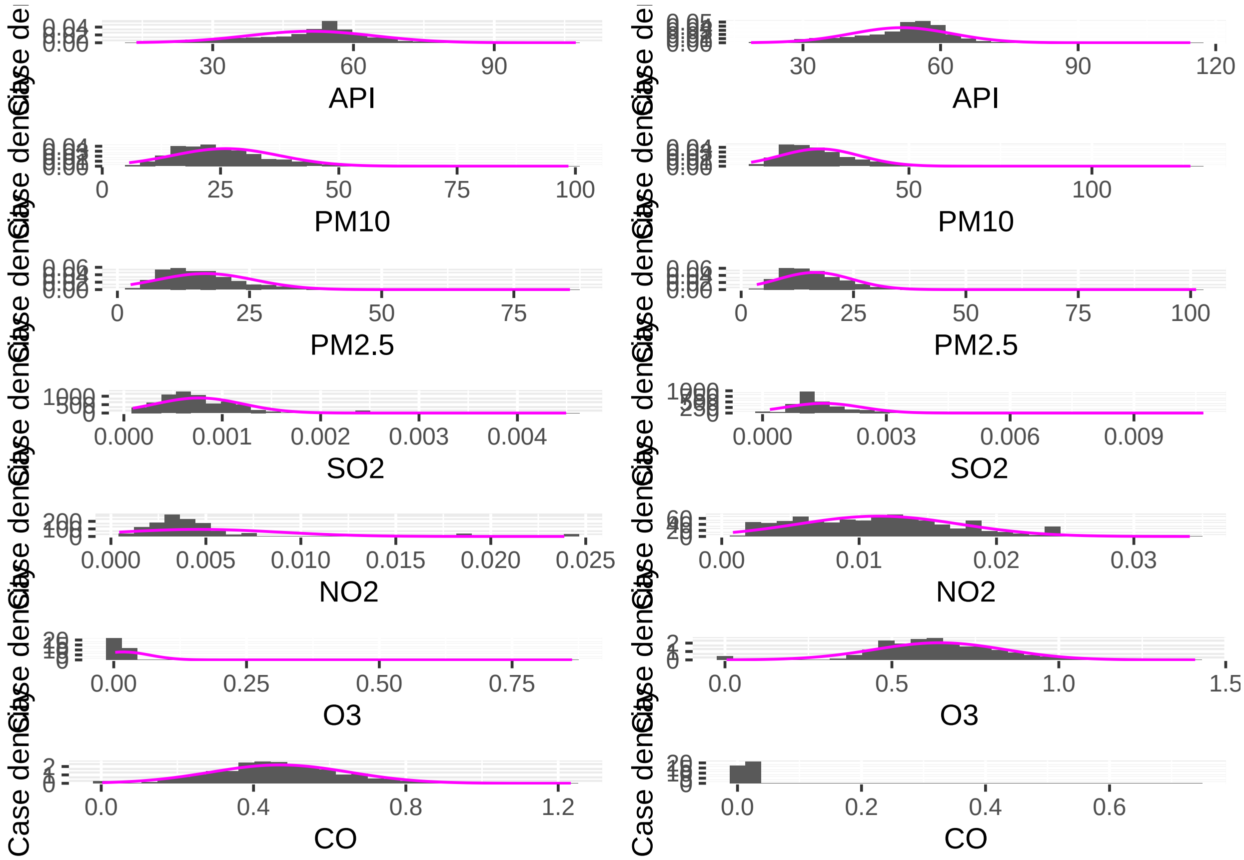
### 2.1.3 Change variable format

```
# Individual plots for air pollutants based on locations

APIKB <- ggplot(kbdata, aes(API))+ geom_histogram(aes(y=..density..)) +
  stat_function(fun=dnorm, colour="magenta", args=list(mean=mean(kbdata$API), sd=sd(kbdata$API))) + labs

APIJB <- ggplot(jbdata, aes(API))+ geom_histogram(aes(y=..density..)) +
  stat_function(fun=dnorm, colour="magenta", args=list(mean=mean(jbdata$API), sd=sd(jbdata$API))) + labs

PM10KB <- ggplot(kbdata, aes(PM10))+ geom_histogram(aes(y=..density..)) +
  stat_function(fun=dnorm, colour="magenta", args=list(mean=mean(kbdata$PM10), sd=sd(kbdata$PM10))) + la

PM10JB <- ggplot(data2[data2$City=='JB',], aes(PM10))+ geom_histogram(aes(y=..density..)) +
  stat_function(fun=dnorm, colour="magenta", args=list(mean=mean(data2[data2$City=='JB',]$PM10), sd=sd(

PM2.5KB <- ggplot(data2[data2$City=='KB',], aes(PM2.5))+ geom_histogram(aes(y=..density..)) +
  stat_function(fun=dnorm, colour="magenta", args=list(mean=mean(data2[data2$City=='KB',]$PM2.5), sd=sd

PM2.5JB <- ggplot(data2[data2$City=='JB',], aes(PM2.5))+ geom_histogram(aes(y=..density..)) +
```

```r
  stat_function(fun=dnorm, colour="magenta", args=list(mean=mean(data2[data2$City=='JB',]$PM2.5), sd=sd

SO2KB <- ggplot(data2[data2$City=='KB',], aes(SO2))+ geom_histogram(aes(y=..density..)) +
  stat_function(fun=dnorm, colour="magenta", args=list(mean=mean(data2[data2$City=='KB',]$SO2), sd=sd(da

SO2JB <- ggplot(data2[data2$City=='JB',], aes(SO2))+ geom_histogram(aes(y=..density..)) +
  stat_function(fun=dnorm, colour="magenta", args=list(mean=mean(data2[data2$City=='JB',]$SO2), sd=sd(da

NO2KB <- ggplot(data2[data2$City=='KB',], aes(NO2))+ geom_histogram(aes(y=..density..)) +
  stat_function(fun=dnorm, colour="magenta", args=list(mean=mean(data2[data2$City=='KB',]$NO2), sd=sd(da

NO2JB <- ggplot(data2[data2$City=='JB',], aes(NO2))+ geom_histogram(aes(y=..density..)) +
  stat_function(fun=dnorm, colour="magenta", args=list(mean=mean(data2[data2$City=='JB',]$NO2), sd=sd(da

O3KB <- ggplot(data2[data2$City=='KB',], aes(O3))+ geom_histogram(aes(y=..density..)) +
  stat_function(fun=dnorm, colour="magenta", args=list(mean=mean(data2[data2$City=='KB',]$O3), sd=sd(dat

O3JB <- ggplot(data2[data2$City=='JB',], aes(O3))+ geom_histogram(aes(y=..density..)) +
  stat_function(fun=dnorm, colour="magenta", args=list(mean=mean(data2[data2$City=='JB',]$O3), sd=sd(dat

COKB <- ggplot(data2[data2$City=='KB',], aes(CO3))+ geom_histogram(aes(y=..density..)) +
  stat_function(fun=dnorm, colour="magenta", args=list(mean=mean(data2[data2$City=='KB',]$CO3), sd=sd(da

COJB <- ggplot(data2[data2$City=='JB',], aes(CO3))+ geom_histogram(aes(y=..density..)) +
  stat_function(fun=dnorm, colour="magenta", args=list(mean=mean(data2[data2$City=='JB',]$CO3), sd=sd(da

# Combine plots into one

ggarrange(APIKB, APIJB, PM10KB, PM10JB, PM2.5KB, PM2.5JB, SO2KB, SO2JB, NO2KB, NO2JB, O3KB, O3JB, COKB,
          heights = 100,
          ncol = 2, nrow = 7)
```

#### 2.1.4 Visualize air pollutant data

```
## Warning: The dot-dot notation (`..density..`) was deprecated in ggplot2 3.4.0.
## i Please use `after_stat(density)` instead.

## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.

## Warning: Removed 101 rows containing missing values (`geom_function()`).
```




Test for equal variance

```
leveneTest(API ~ Period, data = kbdata)
```

```
## Levene's Test for Homogeneity of Variance (center = median)
##         Df F value Pr(>F)
## group    2  1.6019 0.2018
##       1824
```

Checking for skewness

```
skewness(data2[data2$City=='JB',]$CO3, na.rm = TRUE)
```

```
## [1] 10.23371
```

```
skewness(data2[data2$City=='KB',]$O3, na.rm = TRUE)
```

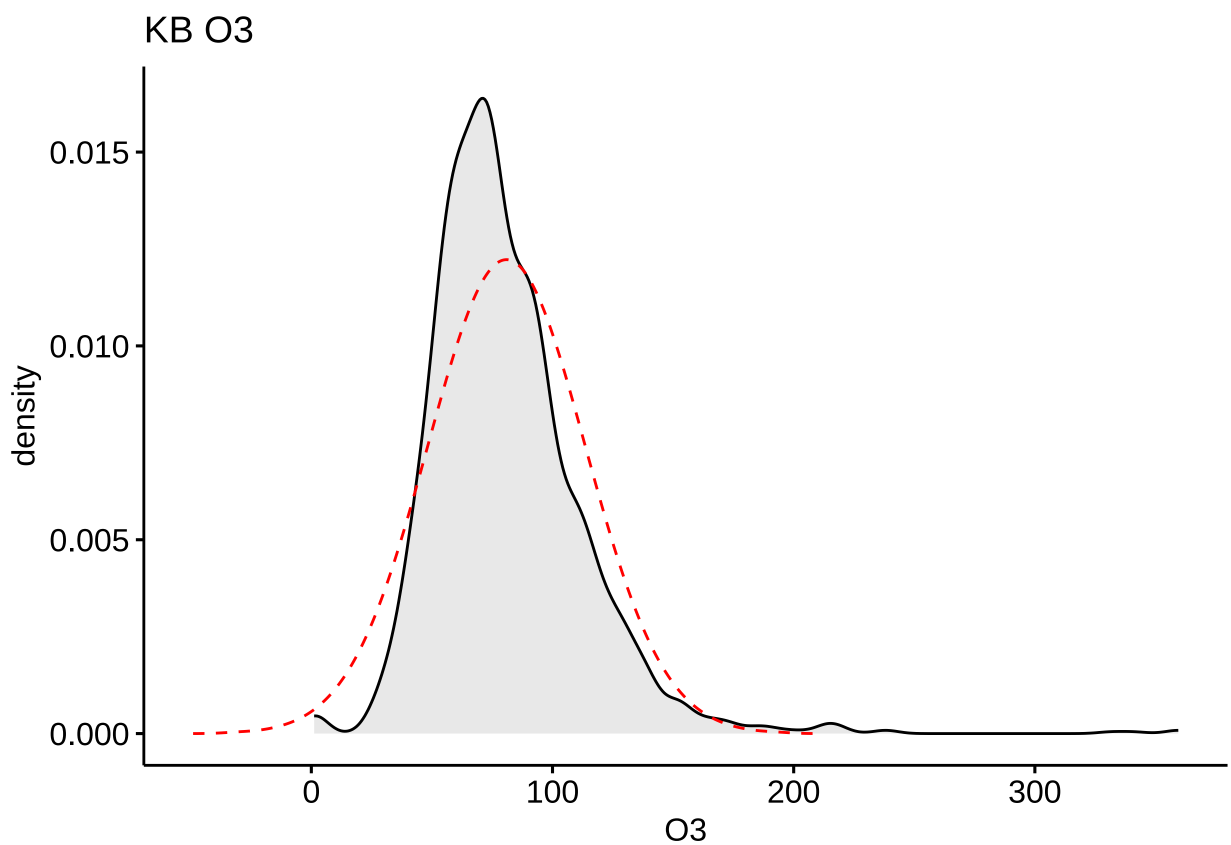
```
## [1] 13.5013
```

Transform skewed data

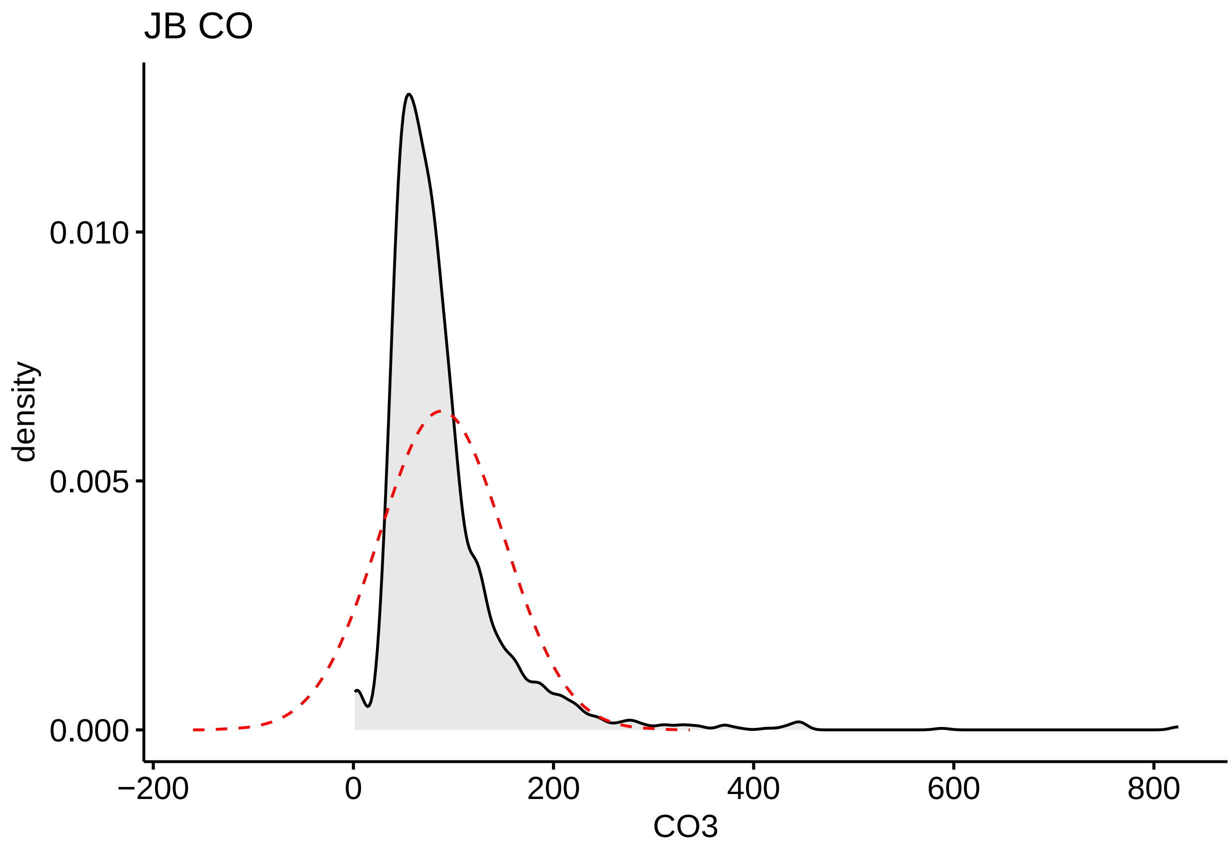
```
kbdata$O3 <- 1/(kbdata$O3)

jbdata$CO3 <- 1/(jbdata$CO3)
```

Rechecking the data

```r
# Distribution of KB O3 variable
ggdensity(kbdata, x = "O3", fill = "lightgray", title = "KB O3") +
  stat_overlay_normal_density(color = "red", linetype = "dashed")
```




```r
# Distribution of JB CO variable
ggdensity(jbdata, x = "CO3", fill = "lightgray", title = "JB CO") +
  stat_overlay_normal_density(color = "red", linetype = "dashed")
```




```
skewness(kbdata$O3, na.rm = TRUE)
```

```
## [1] 2.070165
```

```
skewness(jbdata$CO3, na.rm = TRUE)
```

```
## [1] 4.036607
```

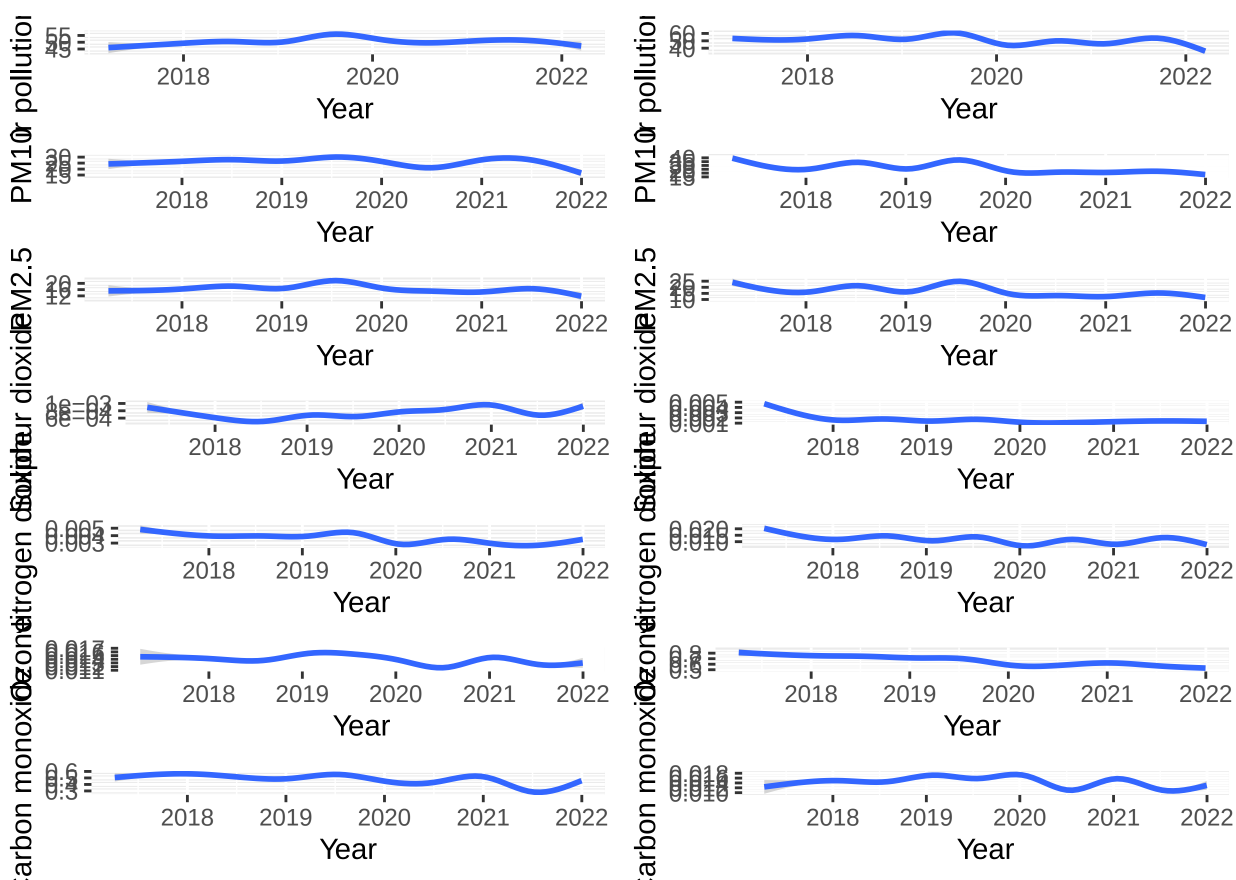
Time series of air pollutant

```
data$Date <- dmy(data$Date)

APIKB2 <- ggplot(data[data$City=='KB',], aes(x=Date, y=API)) + geom_smooth() + labs(x = "Year") + ylab(

APIJB2 <- ggplot(data[data$City=='JB',], aes(x=Date, y=API)) + geom_smooth() + labs(x = "Year") + ylab(

PM10KB2 <- ggplot(data[data$City=='KB',], aes(x=Date, y=PM10)) + geom_smooth()+ labs(x = "Year") + ylab

PM10JB2 <- ggplot(data[data$City=='JB',], aes(x=Date, y=PM10)) + geom_smooth() + labs(x = "Year") + ylal

PM2.5KB2 <- ggplot(data[data$City=='KB',], aes(x=Date, y=PM2.5)) + geom_smooth() + labs(x = "Year") + yl

PM2.5JB2 <- ggplot(data[data$City=='JB',], aes(x=Date, y=PM2.5)) + geom_smooth() + labs(x = "Year") + yl
```

```r
SO2KB2 <- ggplot(data[data$City=='KB',], aes(x=Date, y=SO2)) + geom_smooth() + labs(x = "Year") + ylab(

SO2JB2 <- ggplot(data[data$City=='JB',], aes(x=Date, y=SO2)) + geom_smooth() + labs(x = "Year") + ylab(

NO2KB2 <- ggplot(data[data$City=='KB',], aes(x=Date, y=NO2)) + geom_smooth() + labs(x = "Year") + ylab(

NO2JB2 <- ggplot(data[data$City=='JB',], aes(x=Date, y=NO2)) + geom_smooth() + labs(x = "Year") + ylab(

O3KB2 <- ggplot(data[data$City=='KB',], aes(x=Date, y=O3)) + geom_smooth() + labs(x = "Year") + ylab("O:

O3JB2 <- ggplot(data[data$City=='JB',], aes(x=Date, y=O3)) + geom_smooth() + labs(x = "Year") + ylab("O:

COKB2 <- ggplot(data[data$City=='KB',], aes(x=Date, y=CO3)) + geom_smooth() + labs(x = "Year") + ylab("(

COJB2 <- ggplot(data[data$City=='JB',], aes(x=Date, y=CO3)) + geom_smooth() + labs(x = "Year") + ylab("(

# Combine plots into one

ggarrange(APIKB2, APIJB2, PM10KB2, PM10JB2, PM2.5KB2, PM2.5JB2, SO2KB2, SO2JB2, NO2KB2, NO2JB2, O3KB2, (
          heights = 100,
          ncol = 2, nrow = 7)
```

```
## `geom_smooth()` using method = 'gam' and formula = 'y ~ s(x, bs = "cs")'

## Warning: Removed 109 rows containing non-finite values (`stat_smooth()`).

## `geom_smooth()` using method = 'gam' and formula = 'y ~ s(x, bs = "cs")'

## Warning: Removed 107 rows containing non-finite values (`stat_smooth()`).

## `geom_smooth()` using method = 'gam' and formula = 'y ~ s(x, bs = "cs")'

## Warning: Removed 201 rows containing non-finite values (`stat_smooth()`).

## `geom_smooth()` using method = 'gam' and formula = 'y ~ s(x, bs = "cs")'

## Warning: Removed 200 rows containing non-finite values (`stat_smooth()`).

## `geom_smooth()` using method = 'gam' and formula = 'y ~ s(x, bs = "cs")'

## Warning: Removed 201 rows containing non-finite values (`stat_smooth()`).

## `geom_smooth()` using method = 'gam' and formula = 'y ~ s(x, bs = "cs")'

## Warning: Removed 200 rows containing non-finite values (`stat_smooth()`).

## `geom_smooth()` using method = 'gam' and formula = 'y ~ s(x, bs = "cs")'

## Warning: Removed 202 rows containing non-finite values (`stat_smooth()`).
```

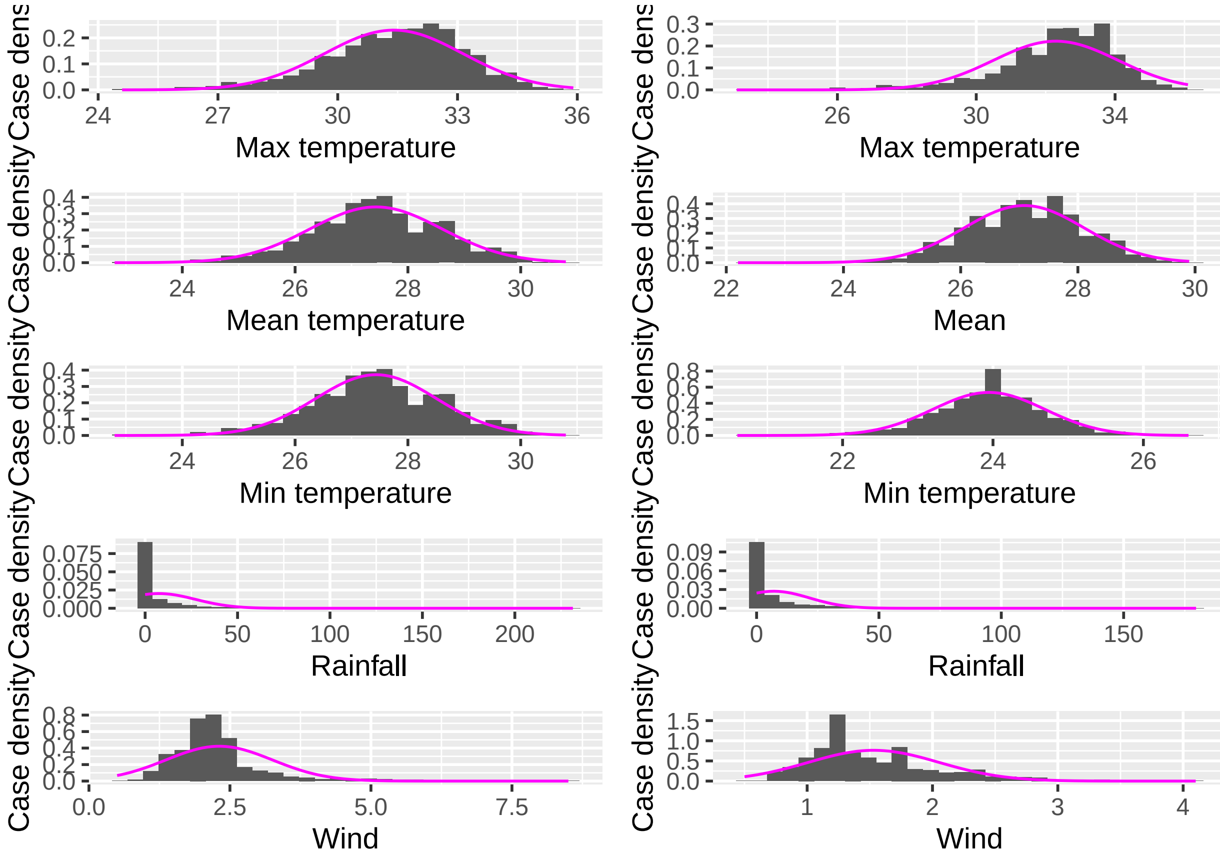
```
## `geom_smooth()` using method = 'gam' and formula = 'y ~ s(x, bs = "cs")'

## Warning: Removed 200 rows containing non-finite values (`stat_smooth()`).

## `geom_smooth()` using method = 'gam' and formula = 'y ~ s(x, bs = "cs")'

## Warning: Removed 223 rows containing non-finite values (`stat_smooth()`).

## `geom_smooth()` using method = 'gam' and formula = 'y ~ s(x, bs = "cs")'

## Warning: Removed 225 rows containing non-finite values (`stat_smooth()`).

## `geom_smooth()` using method = 'gam' and formula = 'y ~ s(x, bs = "cs")'

## Warning: Removed 224 rows containing non-finite values (`stat_smooth()`).

## `geom_smooth()` using method = 'gam' and formula = 'y ~ s(x, bs = "cs")'

## Warning: Removed 225 rows containing non-finite values (`stat_smooth()`).

## `geom_smooth()` using method = 'gam' and formula = 'y ~ s(x, bs = "cs")'

## Warning: Removed 202 rows containing non-finite values (`stat_smooth()`).

## `geom_smooth()` using method = 'gam' and formula = 'y ~ s(x, bs = "cs")'

## Warning: Removed 225 rows containing non-finite values (`stat_smooth()`).
```



```r
# Individual plots for weather variables based on locations

maxtempkb <- ggplot(data2[data2$City=='KB',], aes(Max))+ geom_histogram(aes(y=..density..)) +
  stat_function(fun=dnorm, colour="magenta", args=list(mean=mean(data2[data2$City=='KB',]$Max), sd=sd(da

maxtempjb <- ggplot(data2[data2$City=='JB',], aes(Max))+ geom_histogram(aes(y=..density..)) +
  stat_function(fun=dnorm, colour="magenta", args=list(mean=mean(data2[data2$City=='JB',]$Max), sd=sd(da

meantempkb <- ggplot(data2[data2$City=='KB',], aes(Mean))+ geom_histogram(aes(y=..density..)) +
  stat_function(fun=dnorm, colour="magenta", args=list(mean=mean(data2[data2$City=='KB',]$Mean), sd=sd(

meantempjb <- ggplot(data2[data2$City=='JB',], aes(Mean))+ geom_histogram(aes(y=..density..)) +
  stat_function(fun=dnorm, colour="magenta", args=list(mean=mean(data2[data2$City=='JB',]$Mean), sd=sd(

mintempkb <- ggplot(data2[data2$City=='KB',], aes(Mean))+ geom_histogram(aes(y=..density..)) +
  stat_function(fun=dnorm, colour="magenta", args=list(mean=mean(data2[data2$City=='KB',]$Mean), sd=sd(

mintempjb <- ggplot(data2[data2$City=='JB',], aes(Min))+ geom_histogram(aes(y=..density..)) +
  stat_function(fun=dnorm, colour="magenta", args=list(mean=mean(data2[data2$City=='JB',]$Min), sd=sd(da

rainkb <- ggplot(data2[data2$City=='KB',], aes(Rainfall))+ geom_histogram(aes(y=..density..)) +
  stat_function(fun=dnorm, colour="magenta", args=list(mean=mean(data2[data2$City=='KB',]$Rainfall), sd=

rainjb <- ggplot(data2[data2$City=='JB',], aes(Rainfall))+ geom_histogram(aes(y=..density..)) +
  stat_function(fun=dnorm, colour="magenta", args=list(mean=mean(data2[data2$City=='JB',]$Rainfall), sd=

windkb <- ggplot(data2[data2$City=='KB',], aes(Wind))+ geom_histogram(aes(y=..density..)) +
  stat_function(fun=dnorm, colour="magenta", args=list(mean=mean(data2[data2$City=='KB',]$Wind), sd=sd(

windjb <- ggplot(data2[data2$City=='JB',], aes(Wind))+ geom_histogram(aes(y=..density..)) +
  stat_function(fun=dnorm, colour="magenta", args=list(mean=mean(data2[data2$City=='JB',]$Wind), sd=sd(

# Combine plots into one

ggarrange(maxtempkb, maxtempjb, meantempkb, meantempjb, mintempkb, mintempjb, rainkb, rainjb, windkb, wi
          heights = 100,
          ncol = 2, nrow = 5)
```

### 2.1.5 Visualize weather data

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```




Checking for skewness of rainfall data

```
skewness(kbdata$Rainfall)
```

```
## [1] 5.359294
```

```
skewness(jbdata$Rainfall)
```

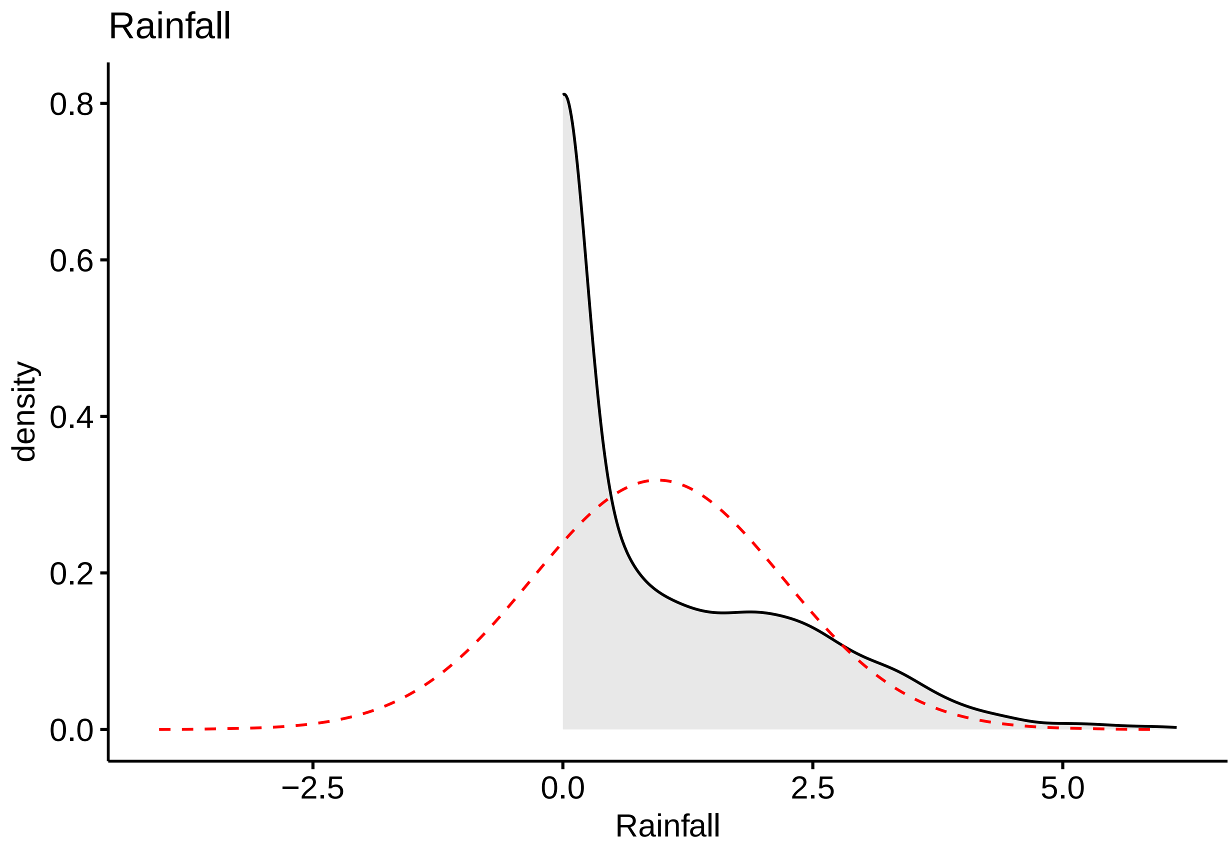
```
## [1] 3.801234
```

Cube root transformation of the skewed data

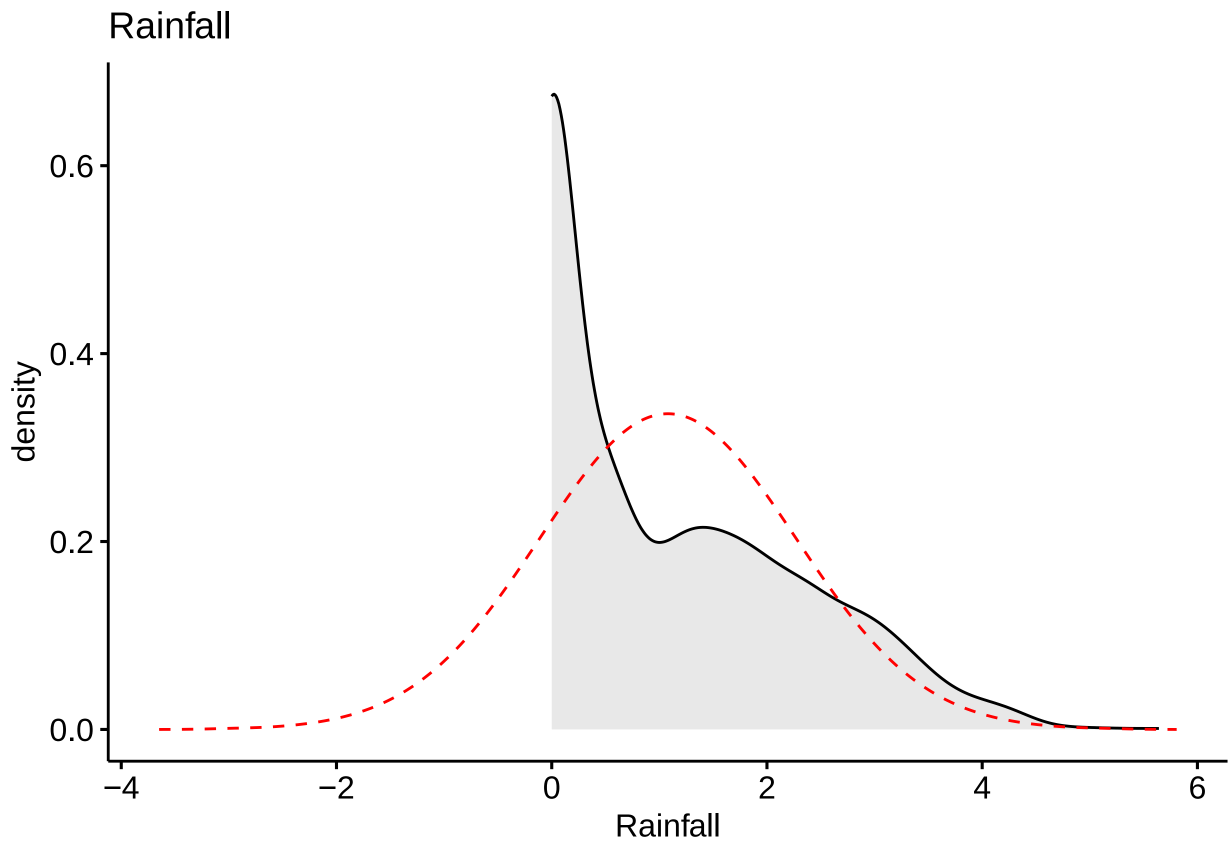
```
kbdata$Rainfall <- kbdata$Rainfall^(1/3)
jbdata$Rainfall <- jbdata$Rainfall^(1/3)
```

Rechecking for rainfall data

```
# Distribution of KB rainfall
ggdensity(kbdata, x = "Rainfall", fill = "lightgray", title = "Rainfall") +
  stat_overlay_normal_density(color = "red", linetype = "dashed")
```




```
# Distribution of JB rainfall
ggdensity(jbdata, x = "Rainfall", fill = "lightgray", title = "Rainfall") +
  stat_overlay_normal_density(color = "red", linetype = "dashed")
```




```
skewness(kbdata$Rainfall)
```

```
## [1] 1.237896
```

```
skewness(jbdata$Rainfall)
```

```
## [1] 0.8538948
```

Summary of meteorological variables

```
data2[data2$City=='KB',] %>%
    get_summary_stats(c("Rainfall", "Min", "Max", "Mean", "Rainfall", "Wind"), type = "mean_sd")
```

```
## # A tibble: 5 x 4
##   variable     n  mean     sd
##   <fct>    <dbl> <dbl>  <dbl>
## 1 Rainfall  1827  7.44 19.7
## 2 Min       1827 24.4   1.07
## 3 Max       1827 31.4   1.74
## 4 Mean      1827 27.4   1.17
## 5 Wind      1827  2.31  0.945
```

```r
data2[data2$City=='JB',] %>%
    get_summary_stats(c("Rainfall", "Min", "Max", "Mean", "Rainfall", "Wind"), type = "mean_sd")
```

```
## # A tibble: 5 x 4
##   variable     n  mean     sd
##   <fct>    <dbl> <dbl>  <dbl>
## 1 Rainfall  1827  7.00 14.6
## 2 Min       1827 23.9   0.746
## 3 Max       1827 32.3   1.80
## 4 Mean      1827 27.1   1.03
## 5 Wind      1827  1.53  0.522
```

Comparison of meteorological variables in JB and KB

```r
rainkb <- rnorm(1827, mean = 7.444, sd = 19.74)
rainjb <- rnorm(1827, mean = 7.00, sd = 14.64)
t.test(rainkb, rainjb, var.equal = FALSE)
```

```
##
##  Welch Two Sample t-test
##
## data:  rainkb and rainjb
## t = 0.87631, df = 3342.2, p-value = 0.3809
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.643923  1.684674
## sample estimates:
## mean of x mean of y
##  7.564674  7.044299
```

```r
windkb <- rnorm(1827, mean = 2.31, sd = 0.95)
windjb <- rnorm(1827, mean = 1.53, sd = 0.52)
t.test(windkb, windjb, var.equal = FALSE)
```

```
##
##  Welch Two Sample t-test
##
## data:  windkb and windjb
## t = 31.162, df = 2829.6, p-value < 2.2e-16
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  0.7421266 0.8417920
## sample estimates:
## mean of x mean of y
##  2.343794  1.551835
```

```r
minkb <- rnorm(1827, mean = 24.4, sd = 1.07)
minjb <- rnorm(1827, mean = 23.95, sd = 0.75)
t.test(minkb, minjb, var.equal = FALSE)
```

```
##
##  Welch Two Sample t-test
##
## data:  minkb and minjb
## t = 14.333, df = 3354.3, p-value < 2.2e-16
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  0.3728299 0.4909961
## sample estimates:
## mean of x mean of y
##  24.37435  23.94244
```

```
meankb <- rnorm(1827, mean = 27.44, sd = 1.17)
meanjb <- rnorm(1827, mean = 27.08, sd = 1.03)
t.test(meankb, meanjb, var.equal = FALSE)
```

```
##
##  Welch Two Sample t-test
##
## data:  meankb and meanjb
## t = 11.218, df = 3600, p-value < 2.2e-16
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  0.3310590 0.4712968
## sample estimates:
## mean of x mean of y
##  27.47732  27.07614
```

```
maxkb <- rnorm(1827, mean = 31.4, sd = 1.74)
maxjb <- rnorm(1827, mean = 32.29, sd = 2.8)
t.test(maxkb, maxjb, var.equal = FALSE)
```

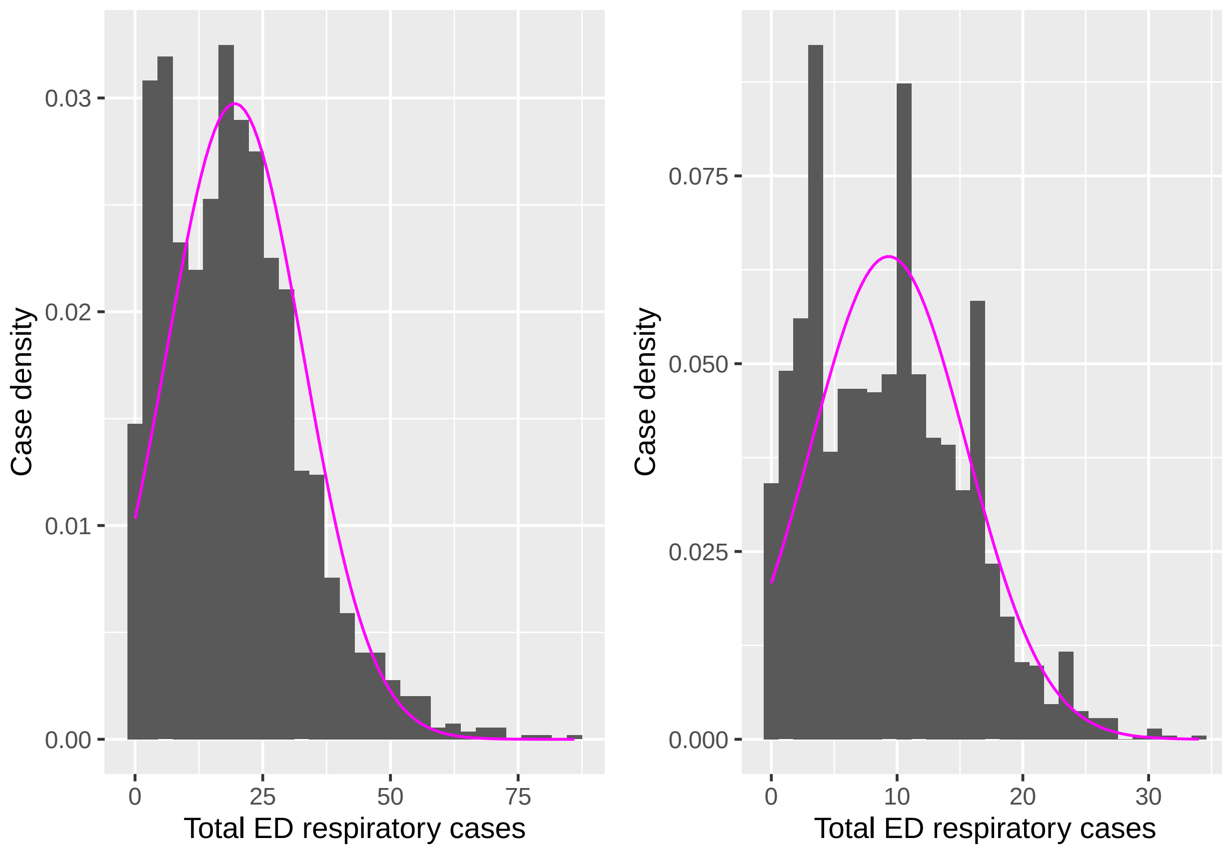
```
##
##  Welch Two Sample t-test
##
## data:  maxkb and maxjb
## t = -12.563, df = 3005.7, p-value < 2.2e-16
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -1.1158301 -0.8145458
## sample estimates:
## mean of x mean of y
##  31.39703  32.36222
```

```
RespiKB <- ggplot(data2[data2$City=='KB',], aes(ED))+ geom_histogram(aes(y=..density..)) +
  stat_function(fun=dnorm, colour="magenta", args=list(mean=mean(data2[data2$City=='KB',]$ED), sd=sd(da

RespiJB <- ggplot(data2[data2$City=='JB',], aes(ED))+ geom_histogram(aes(y=..density..)) +
  stat_function(fun=dnorm, colour="magenta", args=list(mean=mean(data2[data2$City=='JB',]$ED), sd=sd(da
```

```
ggarrange(RespiKB, RespiJB,
          heights = 100,
          ncol = 2, nrow = 1)
```

#### 2.1.6. Total respiratory diagnosis by city

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```




```
my_theme <- function() {

  # Colors
  color.background = "white"
  color.text = "#22211d"

  # Begin construction of chart
  theme_bw(base_size=15) +

    # Format background colors
    theme(panel.background = element_rect(fill=color.background, color=color.background)) +
```

```r
    theme(plot.background  = element_rect(fill=color.background, color=color.background)) +
    theme(panel.border     = element_rect(color=color.background)) +
    theme(strip.background = element_rect(fill=color.background, color=color.background)) +

    # Format the grid
    theme(panel.grid.major.y = element_blank()) +
    theme(panel.grid.minor.y = element_blank()) +
    theme(axis.ticks       = element_blank()) +

    # Format the legend
    theme(legend.position = "none") +

    # Format title and axis labels
    theme(plot.title       = element_text(color=color.text, size=20, face = "bold")) +
    theme(axis.title.x     = element_text(size=14, color="black", face = "bold")) +
    theme(axis.title.y     = element_text(size=14, color="black", face = "bold", vjust=1.25)) +
    theme(axis.text.x      = element_text(size=10, vjust=0.5, hjust=0.5, color = color.text)) +
    theme(axis.text.y      = element_text(size=10, color = color.text)) +
    theme(strip.text       = element_text(face = "bold")) +

    # Plot margins
    theme(plot.margin = unit(c(0.35, 0.2, 0.3, 0.35), "cm"))
}
```

#### 2.1.7. Ranking of respiratory diseases by period

```r
rankkb <- read.csv('respirankkb.csv')

rankkb$Period2 <- as.factor(rankkb$Period2)

kb.rank <- rankkb %>%
  group_by(Period) %>%
  arrange(Period2, Rank, Diagnosis) %>%
  mutate(ranking = row_number()) %>%
  as.data.frame()

head(kb.rank)
```

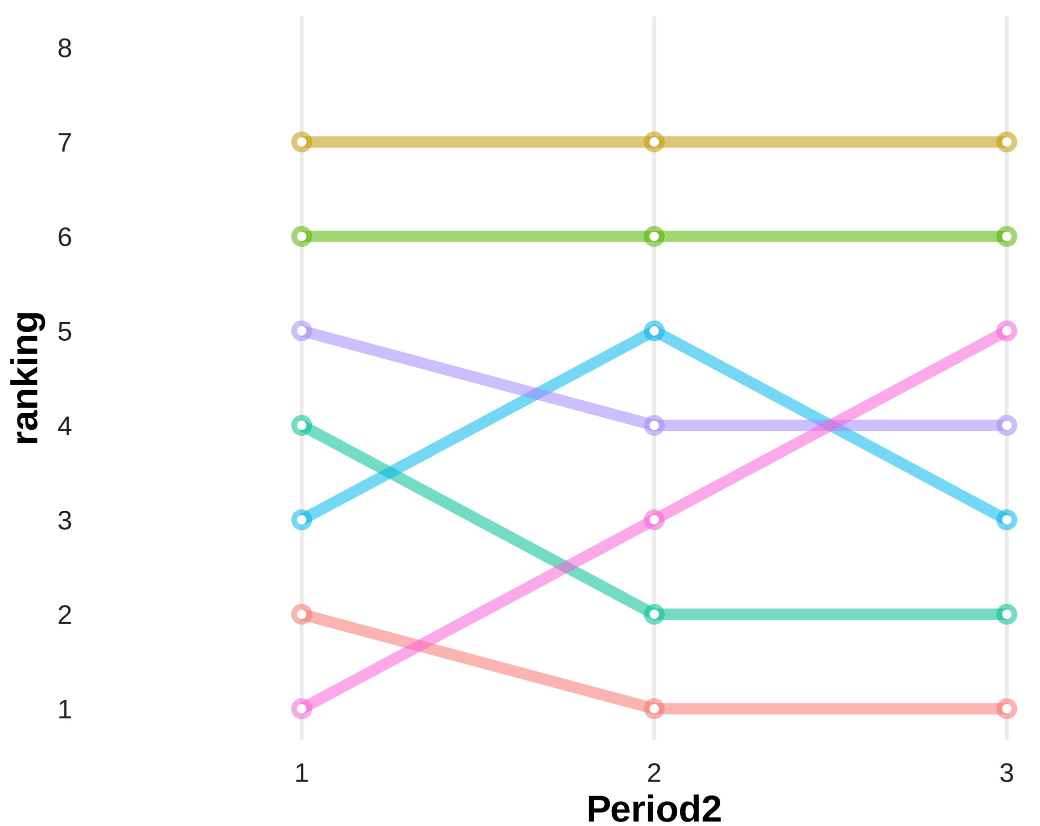
**KB rank**

```
##   Period Period2
## 1 PreMCO       1
## 2 PreMCO       1
## 3 PreMCO       1
## 4 PreMCO       1
## 5 PreMCO       1
## 6 PreMCO       1
##                                                       Diagnosis
## 1                                                   U07.1_COVID19
## 2                              A00-B99 Certain infectious and parasitic diseases
```

```
## 3                                    J40-J47_Chronic lower respiratory diseases
## 4                              J20-J22_Other acute lower respiratory infections
## 5 R00-R09 Symptoms and signs involving the circulatory and respiratory systems
## 6                                                J09-J18_Influenza and pneumonia
##   Rank ranking
## 1    1       1
## 2    2       2
## 3    3       3
## 4    4       4
## 5    5       5
## 6    6       6
```

```r
ggplot(data = kb.rank, aes(x = Period2, y = ranking, group = Diagnosis)) +
  geom_line(aes(color = Diagnosis, alpha = 1), size = 2) +
  geom_point(aes(color = Diagnosis, alpha = 1), size = 3) +
  geom_point(color = "#FFFFFF", size = 1) +
  scale_y_reverse(breaks = 1:8) +
  coord_cartesian(ylim = c(1,8)) +
  theme(legend.position = "none") +
  my_theme()
```

```
## Warning: Using ‘size‘ aesthetic for lines was deprecated in ggplot2 3.4.0.
## i Please use ‘linewidth‘ instead.
```

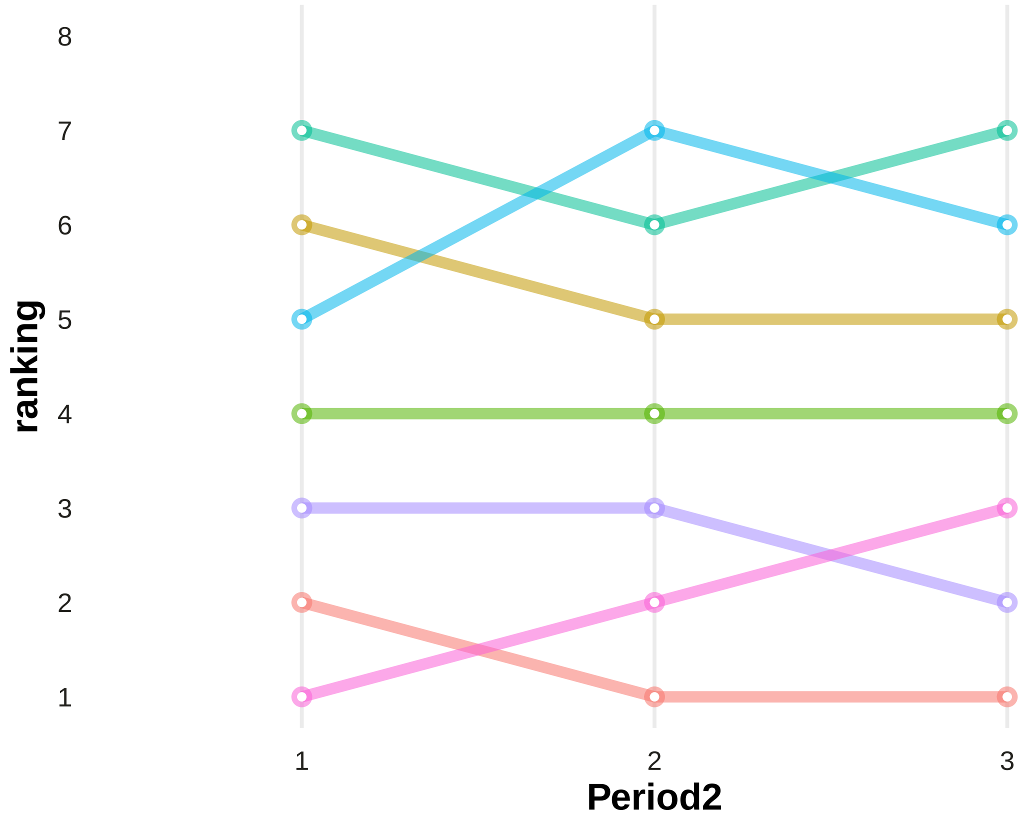


```r
rankjb <- read.csv('respirankjb.csv')

rankjb$Period2 <- as.factor(rankjb$Period2)

jb.rank <- rankjb %>%
  group_by(Period) %>%
  arrange(Period2, Rank, Diagnosis) %>%
  mutate(ranking = row_number()) %>%
  as.data.frame()

head(jb.rank)
```

**JB rank**

```
##   Period Period2
## 1 PreMCO       1
## 2 PreMCO       1
## 3 PreMCO       1
## 4 PreMCO       1
## 5 PreMCO       1
## 6 PreMCO       1
##                                                                      Diagnosis
## 1                                                                U07.1_COVID19
## 2                                   A00-B99 Certain infectious and parasitic diseases
## 3 R00-R09 Symptoms and signs involving the circulatory and respiratory systems
## 4                                                 J09-J18_Influenza and pneumonia
## 5                                          J40-J47_Chronic lower respiratory diseases
## 6                                          J00-J06_Acute upper respiratory infections
##   Rank ranking
## 1    1       1
## 2    2       2
## 3    3       3
## 4    4       4
## 5    5       5
## 6    6       6
```

```r
ggplot(data = jb.rank, aes(x = Period2, y = ranking, group = Diagnosis)) +
  geom_line(aes(color = Diagnosis, alpha = 1), size = 2) +
  geom_point(aes(color = Diagnosis, alpha = 1), size = 3) +
  geom_point(color = "#FFFFFF", size = 1) +
  scale_y_reverse(breaks = 1:8) +
  coord_cartesian(ylim = c(1,8)) +
  theme(legend.position = "none") +
  my_theme()
```




```
data2[data2$City=='KB',] %>%
  group_by(Period) %>%
  get_summary_stats(c("API", "PM10", "PM2.5", "SO2", "NO2", "CO3", "O3"), type = "mean_sd")
```

### 2.1.8. Air pollution by city, by period

```
## # A tibble: 21 x 5
##    Period variable     n   mean     sd
##    <fct>  <fct>    <dbl>  <dbl>  <dbl>
##  1 1      API       1097 51.2   13.5
##  2 1      PM10      1097 27.3   11.9
##  3 1      PM2.5     1097 17.8   10.2
##  4 1      SO2       1097  0.001  0
##  5 1      NO2       1097  0.005  0.004
##  6 1      CO3       1097  0.517  0.168
##  7 1      O3        1097  0.02   0.062
##  8 2      API        656 50.4   13.9
##  9 2      PM10       656 24.6   10.2
## 10 2      PM2.5      656 15.1    7.33
## # ... with 11 more rows
```

```r
data2[data2$City=='JB',] %>%
  group_by(Period) %>%
  get_summary_stats(c("API", "PM10", "PM2.5", "SO2", "NO2", "CO3", "O3"), type = "mean_sd")
```

```
## # A tibble: 21 x 5
##    Period variable     n   mean     sd
##    <fct>  <fct>    <dbl>  <dbl>  <dbl>
##  1 1      API       1097 53.7   10.8
##  2 1      PM10      1097 29.1   12.0
##  3 1      PM2.5     1097 18.7    9.28
##  4 1      SO2       1097  0.002  0.001
##  5 1      NO2       1097  0.012  0.006
##  6 1      CO3       1097  0.017  0.024
##  7 1      O3        1097  0.704  0.184
##  8 2      API        656 49.0   10.3
##  9 2      PM10       656 21.1    6.13
## 10 2      PM2.5      656 13.5    4.70
## # ... with 11 more rows
```

```r
data2[data2$City=='KB',] %>%
  group_by(Period) %>%
  get_summary_stats(c("Infection", "URTI", "Pneumonia", "LRTI", "CLD", "Symptoms_Signs", "COVID"))
```

### 2.1.9. Respiratory diseases by city, by period

```
## # A tibble: 21 x 14
##    Period variable     n   min   max median    q1    q3   iqr   mad   mean    sd
##    <fct>  <fct>    <dbl> <dbl> <dbl>  <dbl> <dbl> <dbl> <dbl> <dbl>  <dbl> <dbl>
##  1 1      Infecti~  1097     0     2      0     0     0     0  0     0.018 0.141
##  2 1      URTI      1097     0    75     18    13    25    12  8.90 20.0   9.90
##  3 1      Pneumon~  1097     0    16      3     2     4     2  1.48  3.08  2.07
##  4 1      LRTI      1097     0     7      1     0     1     1  1.48  0.864 0.978
##  5 1      CLD       1097     0     8      0     0     1     1  0     0.809 1.10
##  6 1      Symptom~  1097     0     9      2     1     3     2  1.48  1.91  1.66
##  7 1      COVID     1097     0     2      0     0     0     0  0     0.003 0.067
##  8 2      Infecti~   656     0     0      0     0     0     0  0     0     0
##  9 2      URTI       656     0    30      4     2     6     4  2.96  4.31  3.73
## 10 2      Pneumon~   656     0    10      1     0     2     2  1.48  1.55  1.92
## # ... with 11 more rows, and 2 more variables: se <dbl>, ci <dbl>
```

```r
data2[data2$City=='JB',] %>%
  group_by(Period) %>%
  get_summary_stats(c("Infection", "URTI", "Pneumonia", "LRTI", "CLD", "Symptoms_Signs", "COVID"))
```

```
## # A tibble: 21 x 14
##    Period variable      n   min   max median    q1    q3   iqr   mad  mean    sd
##    <fct>  <fct>     <dbl> <dbl> <dbl>  <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
##  1 1      Infection  1097     0     1      0     0     0     0  0    0.008 0.09
##  2 1      URTI       1097     0    13      3     1     4     3  2.96 3.06  2.28
```

```
## 3 1 Pneumonia 1097 0 9 1 0 2 2 1.48 1.48 1.38
## 4 1 LRTI 1097 0 15 3 2 5 3 2.96 3.68 2.43
## 5 1 CLD 1097 0 13 3 1 4 3 2.96 3.03 2.28
## 6 1 Symptoms~ 1097 0 12 0 0 1 1 0 0.828 1.15
## 7 1 COVID 1097 0 2 0 0 0 0 0 0.004 0.085
## 8 2 Infection 656 0 1 0 0 0 0 0 0.006 0.078
## 9 2 URTI 656 0 8 1 0 2 2 1.48 1.08 1.35
## 10 2 Pneumonia 656 0 8 0 0 1 1 0 0.86 1.18
## # ... with 11 more rows, and 2 more variables: se <dbl>, ci <dbl>
```

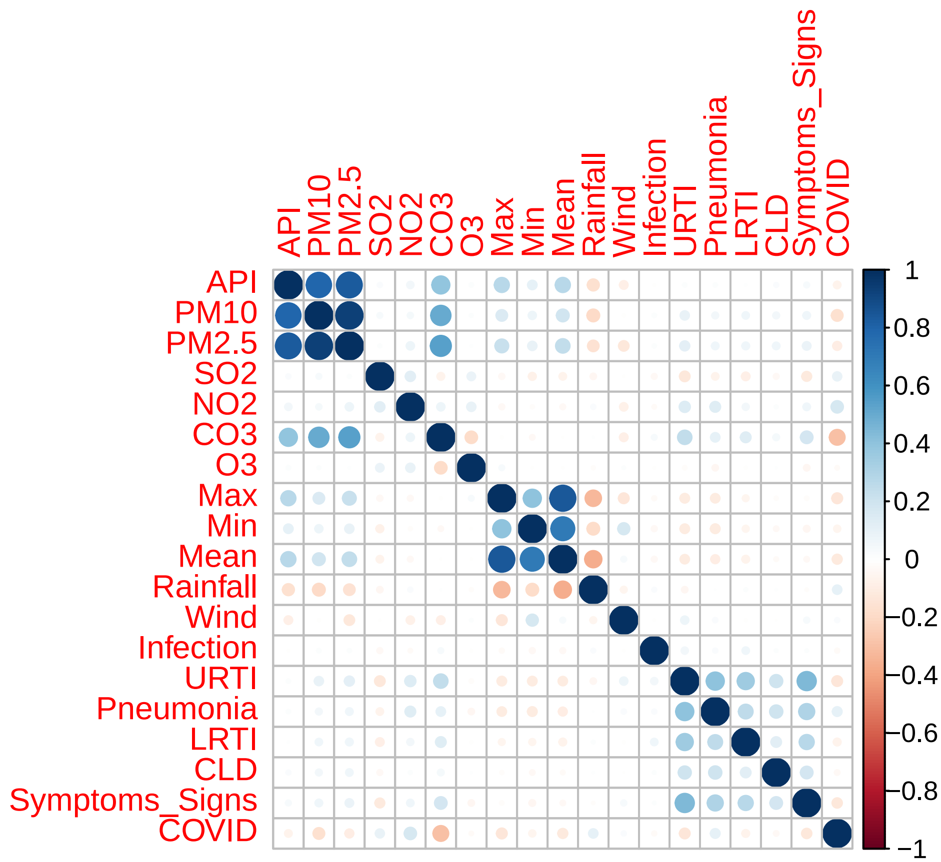
### 2.2. Variable selection

```r
# Kota Bharu
##Select numerical variables

datakb <- data2[data2$City=='KB', -1:-12]

##Correlation matrix
cor.datakb <- cor(datakb, use="complete.obs", method="pearson")
head(round(cor.datakb,2))
```

#### 2.2.1. Correlation

```
##         API PM10 PM2.5   SO2  NO2   CO3    O3   Max   Min  Mean Rainfall  Wind
## API    1.00 0.79  0.83  0.03 0.06  0.40  0.02  0.27  0.10  0.28    -0.17 -0.08
## PM10   0.79 1.00  0.93  0.04 0.04  0.50  0.01  0.16  0.07  0.19    -0.19 -0.01
## PM2.5  0.83 0.93  1.00  0.01 0.08  0.54  0.01  0.23  0.10  0.25    -0.15 -0.13
## SO2    0.03 0.04  0.01  1.00 0.12 -0.07  0.08 -0.04 -0.07 -0.06    -0.04  0.00
## NO2    0.06 0.04  0.08  0.12 1.00  0.08  0.10 -0.04 -0.02 -0.03     0.03 -0.08
## CO3    0.40 0.50  0.54 -0.07 0.08  1.00 -0.18 -0.01 -0.03  0.00     0.00 -0.08
##       Infection  URTI Pneumonia  LRTI   CLD Symptoms_Signs COVID
## API        0.00  0.02      0.02  0.00  0.02           0.04 -0.06
## PM10       0.02  0.10      0.05  0.06  0.05           0.07 -0.16
## PM2.5      0.02  0.12      0.06  0.07  0.06           0.08 -0.09
## SO2       -0.04 -0.13     -0.06 -0.08 -0.04          -0.11  0.10
## NO2       -0.02  0.14      0.13  0.05  0.01           0.06  0.18
## CO3        0.03  0.25      0.10  0.13  0.05           0.19 -0.30
```

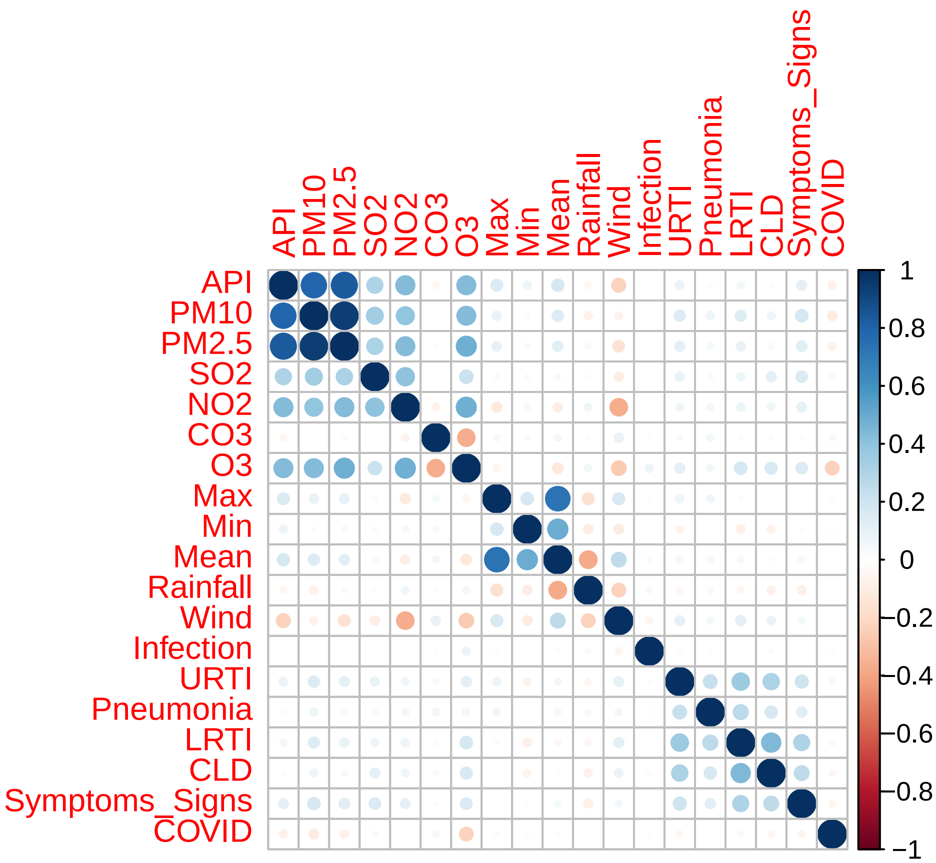
```r
##Plot correlogram
corrplot(cor.datakb, method="circle")
```




```r
# Johor Bahru
##Select numerical variables

datajb <- data2[data2$City=='JB', -1:-12]

##Correlation matrix
cor.datajb <- cor(datajb, use="complete.obs", method="pearson")
head(round(cor.datajb,2))
```

```
##          API  PM10 PM2.5  SO2   NO2   CO3    O3   Max   Min  Mean Rainfall  Wind
## API     1.00  0.79  0.84 0.31  0.44 -0.04  0.43  0.16  0.08  0.17    -0.05 -0.23
## PM10    0.79  1.00  0.95 0.34  0.39 -0.01  0.43  0.09  0.02  0.15    -0.08 -0.07
## PM2.5   0.84  0.95  1.00 0.32  0.44 -0.03  0.49  0.10  0.03  0.12    -0.03 -0.16
## SO2     0.31  0.34  0.32 1.00  0.40  0.01  0.21 -0.03  0.03  0.04    -0.02 -0.10
## NO2     0.44  0.39  0.44 0.40  1.00 -0.06  0.49 -0.12  0.05 -0.09     0.06 -0.37
## CO3    -0.04 -0.01 -0.03 0.01 -0.06  1.00 -0.37  0.04 -0.03  0.05    -0.02  0.09
##       Infection URTI Pneumonia  LRTI   CLD Symptoms_Signs COVID
## API        0.01 0.08      0.03  0.05  0.02           0.11 -0.07
## PM10       0.00 0.15      0.08  0.14  0.07           0.18 -0.10
## PM2.5      0.00 0.11      0.05  0.10  0.04           0.13 -0.08
## SO2       -0.01 0.09      0.03  0.08  0.11           0.15 -0.03
## NO2        0.00 0.06      0.05  0.08  0.06           0.11  0.00
## CO3       -0.01 0.04      0.06 -0.03 -0.03          -0.02  0.04
```

```r
##Plot correlogram
corrplot(cor.datajb, method="circle")
```




Since the air pollution index and particulate matters are highly correlated, two separate analysis with the first containing API only and the other for the separate pollutants will be conducted. These models will then be compared to determine robustness of the models.

Considering that the maximum and minimum temperature are highly correlated with mean temperature, only the mean temperature will be included as covariates in the models.

### 2.2.2. Comparison of air pollution and respiratory diseases between periods

```r
statJBAPI <- data2[data2$City=='JB',] %>%
  t_test(API ~ Period) %>%
  add_significance()
statJBAPI
```

**JBKB API t-test**

```
## # A tibble: 3 x 10
##   .y.   group1 group2    n1    n2 statistic     df         p     p.adj
##   <chr> <chr>  <chr>  <int> <int>     <dbl>  <dbl>     <dbl>     <dbl>
```

```
## 1 API   1      2       1097   656      9.16 1434.  1.79e-19 5.37e-19
## 2 API   1      3       1097    74     10.2    85.5 2.04e-16 4.08e-16
## 3 API   2      3        656    74      6.06   92.2 2.94e- 8 2.94e- 8
## # ... with 1 more variable: p.adj.signif <chr>
```

```
statKBAPI <- data2[data2$City=='KB',] %>%
  t_test(API ~ Period) %>%
  add_significance()
statKBAPI
```

```
## # A tibble: 3 x 10
##   .y.   group1 group2    n1    n2 statistic     df     p p.adj p.adj.signif
##   <chr> <chr>  <chr>  <int> <int>     <dbl>  <dbl> <dbl> <dbl> <chr>
## 1 API   1      2       1097   656     1.20   1349.  0.232 0.696 ns
## 2 API   1      3       1097    74     0.401   84.3 0.689 1     ns
## 3 API   2      3        656    74    -0.120   93.3 0.905 1     ns
```

```
statJBPM10 <- data2[data2$City=='JB',] %>%
  t_test(PM10 ~ Period) %>%
  add_significance()
statJBPM10
```

**JBKB PM10 t-test**

```
## # A tibble: 3 x 10
##   .y.   group1 group2    n1    n2 statistic    df        p    p.adj p.adj.signif
##   <chr> <chr>  <chr>  <int> <int>     <dbl> <dbl>    <dbl>    <dbl> <chr>
## 1 PM10  1      2       1097   656     18.4  1714. 5.65e-69 1.7 e-68 ****
## 2 PM10  1      3       1097    74     19.4   151. 8.45e-43 1.69e-42 ****
## 3 PM10  2      3        656    74      7.84  104. 4.21e-12 4.21e-12 ****
```

```
statKBPM10 <- data2[data2$City=='KB',] %>%
  t_test(PM10 ~ Period) %>%
  add_significance()
statKBPM10
```

```
## # A tibble: 3 x 10
##   .y.   group1 group2    n1    n2 statistic     df        p    p.adj p.adj.signif
##   <chr> <chr>  <chr>  <int> <int>     <dbl>  <dbl>    <dbl>    <dbl> <chr>
## 1 PM10  1      2       1097   656      4.91 1541.   1.02e-6 2.04e-6 ****
## 2 PM10  1      3       1097    74      5.41   91.1  5.04e-7 1.51e-6 ****
## 3 PM10  2      3        656    74      2.99   95.6  4    e-3 4    e-3 **
```

```
statJBPM2.5 <- data2[data2$City=='JB',] %>%
  t_test(PM2.5 ~ Period) %>%
  add_significance()
statJBPM2.5
```

**JBKB PM2.5 t-test**

```
## # A tibble: 3 x 10
##   .y.   group1 group2    n1    n2 statistic    df        p    p.adj p.adj.signif
##   <chr> <chr>  <chr>  <int> <int>     <dbl> <dbl>    <dbl>    <dbl> <chr>
## 1 PM2.5 1      2       1097   656      15.4 1711. 4.45e-50 1.34e-49 ****
## 2 PM2.5 1      3       1097    74      19.9  172. 1.54e-46 3.08e-46 ****
## 3 PM2.5 2      3        656    74      10.0  111. 3.22e-17 3.22e-17 ****
```

```
statKBPM2.5 <- data2[data2$City=='KB',] %>%
  t_test(PM2.5 ~ Period) %>%
  add_significance()
statKBPM2.5
```

```
## # A tibble: 3 x 10
##   .y.   group1 group2    n1    n2 statistic    df        p    p.adj p.adj.signif
##   <chr> <chr>  <chr>  <int> <int>     <dbl> <dbl>    <dbl>    <dbl> <chr>
## 1 PM2.5 1      2       1097   656      6.33 1695. 3.06e-10 9.18e-10 ****
## 2 PM2.5 1      3       1097    74      5.55  107. 2.07e- 7 4.14e- 7 ****
## 3 PM2.5 2      3        656    74      1.96  102. 5.3 e- 2 5.3 e- 2 ns
```

```


statJBSO2 <- data2[data2$City=='JB',] %>%
  t_test(SO2 ~ Period) %>%
  add_significance()
statJBSO2
```

**JBKB SO2 t-test**

```
## # A tibble: 3 x 10
##   .y.   group1 group2    n1    n2 statistic     df        p    p.adj
##   <chr> <chr>  <chr>  <int> <int>     <dbl>  <dbl>    <dbl>    <dbl>
## 1 SO2   1      2       1097   656      9.77 1354.  8.12e-22 2.44e-21
## 2 SO2   1      3       1097    74      5.25   96.7 9.2 e- 7 1.84e- 6
## 3 SO2   2      3        656    74      1.57   75.8 1.2 e- 1 1.2 e- 1
## # ... with 1 more variable: p.adj.signif <chr>
```

```
statKBSO2 <- data2[data2$City=='KB',] %>%
  t_test(SO2 ~ Period) %>%
  add_significance()
statKBSO2
```

```
## # A tibble: 3 x 10
##   .y.   group1 group2    n1    n2 statistic     df        p    p.adj
##   <chr> <chr>  <chr>  <int> <int>     <dbl>  <dbl>    <dbl>    <dbl>
## 1 SO2   1      2       1097   656     -6.78 1626.  1.68e-11 5.04e-11
## 2 SO2   1      3       1097    74     -5.05   76.6 2.95e- 6 5.9 e- 6
## 3 SO2   2      3        656    74     -3.55   76.8 6.68e- 4 6.68e- 4
## # ... with 1 more variable: p.adj.signif <chr>
```

```r
statJBNO2 <- data2[data2$City=='JB',] %>%
  t_test(NO2 ~ Period) %>%
  add_significance()
statJBNO2
```

**JBKB NO2 t-test**

```
## # A tibble: 3 x 10
##   .y.   group1 group2    n1    n2 statistic     df        p   p.adj p.adj.signif
##   <chr> <chr>  <chr>  <int> <int>     <dbl>  <dbl>    <dbl>   <dbl> <chr>
## 1 NO2   1      2       1097   656     6.12  1635.   1.14e-9 3.42e-9 ****
## 2 NO2   1      3       1097    74     1.31    78.6  1.95e-1 3.9 e-1 ns
## 3 NO2   2      3        656    74    -0.355   78.7  7.23e-1 7.23e-1 ns
```

```r
statKBNO2 <- data2[data2$City=='KB',] %>%
  t_test(NO2 ~ Period) %>%
  add_significance()
statKBNO2
```

```
## # A tibble: 3 x 10
##   .y.   group1 group2    n1    n2 statistic     df        p    p.adj
##   <chr> <chr>  <chr>  <int> <int>     <dbl>  <dbl>    <dbl>    <dbl>
## 1 NO2   1      2       1097   656     13.0  1489.  1.04e-36 3.12e-36
## 2 NO2   1      3       1097    74     -8.22   75.8 4.16e-12 4.16e-12
## 3 NO2   2      3        656    74    -10.2    73.6 8.18e-16 1.64e-15
## # ... with 1 more variable: p.adj.signif <chr>
```

```r
statJBCO <- data2[data2$City=='JB',] %>%
  t_test(CO3 ~ Period) %>%
  add_significance()
statJBCO
```

**JBKB CO t-test**

```
## # A tibble: 3 x 10
##   .y.   group1 group2    n1    n2 statistic     df        p   p.adj p.adj.signif
##   <chr> <chr>  <chr>  <int> <int>     <dbl>  <dbl>    <dbl>   <dbl> <chr>
## 1 CO3   1      2       1097   656      5.08 1324.   4.24e-7 1.27e-6 ****
## 2 CO3   1      3       1097    74     -4.84   73.2  7.12e-6 7.12e-6 ****
## 3 CO3   2      3        656    74     -5.03   73.0  3.41e-6 6.82e-6 ****
```

```r
statKBCO <- data2[data2$City=='KB',] %>%
  t_test(CO3 ~ Period) %>%
  add_significance()
statKBCO
```

```
## # A tibble: 3 x 10
##   .y.   group1 group2    n1    n2 statistic     df        p    p.adj
```

```
##   <chr> <chr>  <chr>  <int> <int>     <dbl>  <dbl>    <dbl>    <dbl>
## 1 CO3   1      2       1097   656      12.6  1348.  1.11e-34 3.33e-34
## 2 CO3   1      3       1097    74       8.58   79.0 6.57e-13 1.31e-12
## 3 CO3   2      3        656    74       4.38   83.7 3.42e- 5 3.42e- 5
## # ... with 1 more variable: p.adj.signif <chr>
```

```r
statJBO3 <- data2[data2$City=='JB',] %>%
  t_test(O3 ~ Period) %>%
  add_significance()
statJBO3
```

**JBKB 03 t-test**

```
## # A tibble: 3 x 10
##   .y.   group1 group2    n1    n2 statistic     df        p    p.adj
##   <chr> <chr>  <chr>  <int> <int>     <dbl>  <dbl>    <dbl>    <dbl>
## 1 O3    1      2       1097   656      16.4  1716.  4.98e-56 1.49e-55
## 2 O3    1      3       1097    74      13.5    77.6 4.17e-22 8.34e-22
## 3 O3    2      3        656    74       9.75   76.6 4.46e-15 4.46e-15
## # ... with 1 more variable: p.adj.signif <chr>
```

```r
statKBO3 <- data2[data2$City=='KB',] %>%
  t_test(O3 ~ Period) %>%
  add_significance()
statKBO3
```

```
## # A tibble: 3 x 10
##   .y.   group1 group2    n1    n2 statistic    df     p p.adj p.adj.signif
##   <chr> <chr>  <chr>  <int> <int>     <dbl> <dbl> <dbl> <dbl> <chr>
## 1 O3    1      2       1097   656      3.01 1560. 0.003 0.008 **
## 2 O3    1      3       1097    74      2.35 1128. 0.019 0.038 *
## 3 O3    2      3        656    74     -1.45  472. 0.147 0.147 ns
```

```r
statJBparasitic <- data2[data2$City=='JB',] %>%
  t_test(Infection ~ Period) %>%
  add_significance()
statJBparasitic
```

**JBKB infectious or other parasitic t test**

```
## # A tibble: 3 x 10
##   .y.       group1 group2    n1    n2 statistic    df     p p.adj p.adj.signif
##   <chr>     <chr>  <chr>  <int> <int>     <dbl> <dbl> <dbl> <dbl> <chr>
## 1 Infection 1      2       1097   656     0.516 1537. 0.606 0.606 ns
## 2 Infection 1      3       1097    74     3.01  1096  0.003 0.008 **
## 3 Infection 2      3        656    74     2.00   655  0.045 0.091 ns
```

```r
statKBparasitic <- data2[data2$City=='KB',] %>%
  t_test(Infection ~ Period) %>%
  add_significance()
statKBparasitic
```

```
## # A tibble: 3 x 10
##   .y.       group1 group2    n1    n2 statistic    df           p       p.adj
##   <chr>     <chr>  <chr>  <int> <int>     <dbl> <dbl>       <dbl>       <dbl>
## 1 Infection 1      2       1097   656      4.30  1096   0.0000188   0.0000376
## 2 Infection 1      3       1097    74      4.30  1096   0.0000188   0.0000376
## 3 Infection 2      3        656    74    NaN      NaN NaN         NaN
## # ... with 1 more variable: p.adj.signif <chr>
```

```r
statJBurti <- data2[data2$City=='JB',] %>%
  t_test(URTI ~ Period) %>%
  add_significance()
statJBurti
```

**JBKB URTI t test**

```
## # A tibble: 3 x 10
##   .y.   group1 group2    n1    n2 statistic     df         p     p.adj
##   <chr> <chr>  <chr>  <int> <int>     <dbl>  <dbl>     <dbl>     <dbl>
## 1 URTI  1      2       1097   656     22.9  1751.  1.62e-101 4.86e-101
## 2 URTI  1      3       1097    74      2.14   84.6 3.5 e-  2 3.5 e-  2
## 3 URTI  2      3        656    74     -5.64   79.7 2.47e-  7 4.94e-  7
## # ... with 1 more variable: p.adj.signif <chr>
```

```r
statKBurti <- data2[data2$City=='KB',] %>%
  t_test(URTI ~ Period) %>%
  add_significance()
statKBurti
```

```
## # A tibble: 3 x 10
##   .y.   group1 group2    n1    n2 statistic     df         p     p.adj
##   <chr> <chr>  <chr>  <int> <int>     <dbl>  <dbl>     <dbl>     <dbl>
## 1 URTI  1      2       1097   656     47.3  1533.  3.01e-302 9.03e-302
## 2 URTI  1      3       1097    74     12.3   113.  1.16e- 22 2.32e- 22
## 3 URTI  2      3        656    74    -12.1    81.8 7.39e- 20 7.39e- 20
## # ... with 1 more variable: p.adj.signif <chr>
```

```r
statJBinfluenza <- data2[data2$City=='JB',] %>%
  t_test(Pneumonia ~ Period) %>%
  add_significance()
statJBinfluenza
```

**JBKB influenza or pneumonia t test**

```
## # A tibble: 3 x 10
##   .y.       group1 group2    n1    n2 statistic     df        p    p.adj
##   <chr>     <chr>  <chr>  <int> <int>     <dbl>  <dbl>    <dbl>    <dbl>
## 1 Pneumonia 1      2       1097   656      9.91 1548.  1.76e-22 5.28e-22
## 2 Pneumonia 1      3       1097    74     -2.55   80.6 1.3 e- 2 1.3 e- 2
## 3 Pneumonia 2      3        656    74     -5.77   82.3 1.36e- 7 2.72e- 7
## # ... with 1 more variable: p.adj.signif <chr>
```

```
statKBinfluenza <- data2[data2$City=='KB',] %>%
  t_test(Pneumonia ~ Period) %>%
  add_significance()
statKBinfluenza
```

```
## # A tibble: 3 x 10
##   .y.       group1 group2    n1    n2 statistic     df        p    p.adj
##   <chr>     <chr>  <chr>  <int> <int>     <dbl>  <dbl>    <dbl>    <dbl>
## 1 Pneumonia 1      2       1097   656     15.7  1458.  1.44e-51 4.32e-51
## 2 Pneumonia 1      3       1097    74     -4.11   80.2 9.55e- 5 9.55e- 5
## 3 Pneumonia 2      3        656    74     -9.27   83.4 1.78e-14 3.56e-14
## # ... with 1 more variable: p.adj.signif <chr>
```

```
statJBlrti <- data2[data2$City=='JB',] %>%
  t_test(LRTI ~ Period) %>%
  add_significance()
statJBlrti
```

**JBKB LRTI t test**

```
## # A tibble: 3 x 10
##   .y.   group1 group2    n1    n2 statistic     df         p     p.adj
##   <chr> <chr>  <chr>  <int> <int>     <dbl>  <dbl>     <dbl>     <dbl>
## 1 LRTI  1      2       1097   656    25.8   1724.  3.76e-124 1.13e-123
## 2 LRTI  1      3       1097    74     0.895   83.9 3.74e-  1 3.74e-  1
## 3 LRTI  2      3        656    74    -8.08    81.3 5.05e- 12 1.01e- 11
## # ... with 1 more variable: p.adj.signif <chr>
```

```
statKBlrti <- data2[data2$City=='KB',] %>%
  t_test(LRTI ~ Period) %>%
  add_significance()
statKBlrti
```

```
## # A tibble: 3 x 10
##   .y.   group1 group2    n1    n2 statistic     df        p    p.adj
##   <chr> <chr>  <chr>  <int> <int>     <dbl>  <dbl>    <dbl>    <dbl>
## 1 LRTI  1      2       1097   656     20.2  1673.  1.09e-81 3.27e-81
## 2 LRTI  1      3       1097    74      1.06   83.7 2.94e- 1 2.94e- 1
## 3 LRTI  2      3        656    74     -5.16   76.9 1.86e- 6 3.72e- 6
## # ... with 1 more variable: p.adj.signif <chr>
```

```r
statJBcld <- data2[data2$City=='JB',] %>%
  t_test(CLD ~ Period) %>%
  add_significance()
statJBcld
```

**JBKB CLD t test**

```
## # A tibble: 3 x 10
##   .y.   group1 group2    n1    n2 statistic     df        p    p.adj
##   <chr> <chr>  <chr>  <int> <int>     <dbl>  <dbl>    <dbl>    <dbl>
## 1 CLD   1      2       1097   656    20.5   1740.  3.79e-84 1.14e-83
## 2 CLD   1      3       1097    74     0.876   85.6 3.84e- 1 3.84e- 1
## 3 CLD   2      3        656    74    -6.62    81.7 3.48e- 9 6.96e- 9
## # ... with 1 more variable: p.adj.signif <chr>
```

```r
statKBcld <- data2[data2$City=='KB',] %>%
  t_test(CLD ~ Period) %>%
  add_significance()
statKBcld
```

```
## # A tibble: 3 x 10
##   .y.   group1 group2    n1    n2 statistic     df        p    p.adj
##   <chr> <chr>  <chr>  <int> <int>     <dbl>  <dbl>    <dbl>    <dbl>
## 1 CLD   1      2       1097   656     7.64  1652.  3.56e-14 1.07e-13
## 2 CLD   1      3       1097    74     0.230   89.2 8.18e- 1 8.18e- 1
## 3 CLD   2      3        656    74    -3.10    88.7 3    e- 3 5    e- 3
## # ... with 1 more variable: p.adj.signif <chr>
```

```r
statJBsx <- data2[data2$City=='JB',] %>%
  t_test(Symptoms_Signs ~ Period) %>%
  add_significance()
statJBsx
```

**JBKB Symptoms_Signs t test**

```
## # A tibble: 3 x 10
##   .y.            group1 group2    n1    n2 statistic     df        p    p.adj
##   <chr>          <chr>  <chr>  <int> <int>     <dbl>  <dbl>    <dbl>    <dbl>
## 1 Symptoms_Signs 1      2       1097   656     20.4   1310.  1.37e-80 4.11e-80
## 2 Symptoms_Signs 1      3       1097    74      5.80   103.  7.44e- 8 1.49e- 7
## 3 Symptoms_Signs 2      3        656    74     -3.01   75.9 4    e- 3 4    e- 3
## # ... with 1 more variable: p.adj.signif <chr>
```

```r
statKBsx <- data2[data2$City=='KB',] %>%
  t_test(Symptoms_Signs ~ Period) %>%
  add_significance()
statKBsx
```

```
## # A tibble: 3 x 10
##   .y.            group1 group2    n1    n2 statistic     df         p     p.adj
##   <chr>          <chr>  <chr>  <int> <int>     <dbl>  <dbl>     <dbl>     <dbl>
## 1 Symptoms_Signs 1      2       1097   656     27.1   1615.  1.81e-133 5.43e-133
## 2 Symptoms_Signs 1      3       1097    74      2.93   84.8 4   e-  3 4   e-  3
## 3 Symptoms_Signs 2      3        656    74     -5.55   76.5 3.97e-  7 7.94e-  7
## # ... with 1 more variable: p.adj.signif <chr>
```

```
statJBcovid <- data2[data2$City=='JB',] %>%
  t_test(COVID ~ Period) %>%
  add_significance()
statJBcovid
```

**JBKB covid t test**

```
## # A tibble: 3 x 10
##   .y.   group1 group2    n1    n2 statistic    df         p   p.adj p.adj.signif
##   <chr> <chr>  <chr>  <int> <int>     <dbl> <dbl>     <dbl>   <dbl> <chr>
## 1 COVID 1      2       1097   656     -4.94 716.    9.69e-7 2.91e-6 ****
## 2 COVID 1      3       1097    74     -3.39  73.0   1   e-3 2   e-3 **
## 3 COVID 2      3        656    74     -2.99  73.9   4   e-3 4   e-3 **
```

```
statKBcovid <- data2[data2$City=='KB',] %>%
  t_test(COVID ~ Period) %>%
  add_significance()
statKBcovid
```

```
## # A tibble: 3 x 10
##   .y.   group1 group2    n1    n2 statistic    df        p    p.adj p.adj.signif
##   <chr> <chr>  <chr>  <int> <int>     <dbl> <dbl>    <dbl>    <dbl> <chr>
## 1 COVID 1      2       1097   656    -10.4  660.  1.53e-23 4.59e-23 ****
## 2 COVID 1      3       1097    74     -8.36  73.0 2.97e-12 5.94e-12 ****
## 3 COVID 2      3        656    74     -7.34  74.3 2.19e-10 2.19e-10 ****
```

```
library(VGAM)
```

**Multinomial logistic regression for comparing air pollution and respiratory diseases between periods**

```
## Warning: package 'VGAM' was built under R version 4.2.3

## Loading required package: stats4

##
## Attaching package: 'VGAM'
```

```
## The following object is masked from 'package:AER':
##
##     tobit

## The following object is masked from 'package:lmtest':
##
##     lrtest

## The following object is masked from 'package:car':
##
##     logit

## The following object is masked from 'package:mgcv':
##
##     s
```

```r
respidata <- read.csv("CleanRespi.csv")

respidata$Period <- as.factor(respidata$Period)
respidata$Respiratory.ICD.10 <- as.factor(respidata$Respiratory.ICD.10)
respidata$Hospital <- as.factor(respidata$Hospital)

respidata$Period <- relevel(respidata$Period, ref = "1")
respidata$Respiratory.ICD.10 <- relevel(respidata$Respiratory.ICD.10, ref = "U07.1_COVID19")

kbrespi <- respidata[respidata$Hospital=="HRPZ",]
jbrespi <- respidata[respidata$Hospital=="HSI",]

# Multi-nomial model

modelkb <- vglm(Period ~ RespiDx , multinomial(refLevel = 1), data = kbrespi)
summary(modelkb)
```

```
##
## Call:
## vglm(formula = Period ~ RespiDx, family = multinomial(refLevel = 1),
##     data = kbrespi)
##
## Coefficients:
##                 Estimate Std. Error z value Pr(>|z|)
## (Intercept):1   -1.09749    0.06713 -16.350  < 2e-16 ***
## (Intercept):2   -2.72740    0.13553 -20.124  < 2e-16 ***
## RespiDxcovid:1   5.44129    0.58465   9.307  < 2e-16 ***
## RespiDxcovid:2   7.02242    0.59657  11.771  < 2e-16 ***
## RespiDxflu:1    -0.10472    0.07606  -1.377 0.168568
## RespiDxflu:2     0.35949    0.14771   2.434 0.014944 *
## RespiDxlrti:1   -1.07474    0.12174  -8.828  < 2e-16 ***
## RespiDxlrti:2   -0.11962    0.19392  -0.617 0.537337
## RespiDxsx:1     -1.09549    0.09600 -11.411  < 2e-16 ***
## RespiDxsx:2     -0.31429    0.16954  -1.854 0.063767 .
## RespiDxurti:1   -0.95375    0.07004 -13.618  < 2e-16 ***
## RespiDxurti:2   -0.50116    0.13987  -3.583 0.000339 ***
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Names of linear predictors: log(mu[,2]/mu[,1]), log(mu[,3]/mu[,1])
##
## Residual deviance: 38260.11 on 71310 degrees of freedom
##
## Log-likelihood: -19130.06 on 71310 degrees of freedom
##
## Number of Fisher scoring iterations: 7
##
## Warning: Hauck-Donner effect detected in the following estimate(s):
## '(Intercept):2', 'RespiDxcovid:1', 'RespiDxcovid:2'
##
##
## Reference group is level  1  of the response
```

```r
modeljb <- vglm(Period ~ RespiDx , multinomial(refLevel = 1), data = jbrespi)
summary(modeljb)
```

```
##
## Call:
## vglm(formula = Period ~ RespiDx, family = multinomial(refLevel = 1),
##     data = jbrespi)
##
## Coefficients:
##                 Estimate Std. Error z value Pr(>|z|)
## (Intercept):1  -1.448426   0.039790 -36.402  < 2e-16 ***
## (Intercept):2  -2.770182   0.071477 -38.757  < 2e-16 ***
## RespiDxcovid:1  3.799801   0.524779   7.241 4.46e-13 ***
## RespiDxcovid:2  5.047449   0.529858   9.526  < 2e-16 ***
## RespiDxflu:1    0.395688   0.063015   6.279 3.40e-10 ***
## RespiDxflu:2    0.357352   0.112353   3.181 0.001470 **
## RespiDxlrti:1  -0.206342   0.055912  -3.690 0.000224 ***
## RespiDxlrti:2   0.003764   0.096403   0.039 0.968856
## RespiDxsx:1    -1.312238   0.141121  -9.299  < 2e-16 ***
## RespiDxsx:2    -0.872872   0.218783  -3.990 6.62e-05 ***
## RespiDxurti:1  -0.099494   0.057286  -1.737 0.082425 .
## RespiDxurti:2  -0.122874   0.103842  -1.183 0.236697
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Names of linear predictors: log(mu[,2]/mu[,1]), log(mu[,3]/mu[,1])
##
## Residual deviance: 21612.42 on 34074 degrees of freedom
##
## Log-likelihood: -10806.21 on 34074 degrees of freedom
##
## Number of Fisher scoring iterations: 6
##
## Warning: Hauck-Donner effect detected in the following estimate(s):
## '(Intercept):2'
##
##
## Reference group is level  1  of the response
```

Getting the odds ratio for the models

```
exp(confintvglm(modelkb))
```

```
##                        2.5 %       97.5 %
## (Intercept):1    0.29257090 3.806318e-01
## (Intercept):2    0.05013486 8.528427e-02
## RespiDxcovid:1  73.36153170 7.257322e+02
## RespiDxcovid:2 348.33487573 3.610744e+03
## RespiDxflu:1     0.77585977 1.045350e+00
## RespiDxflu:2     1.07248638 1.913627e+00
## RespiDxlrti:1    0.26892095 4.333811e-01
## RespiDxlrti:2    0.60671387 1.297526e+00
## RespiDxsx:1      0.27702486 4.035974e-01
## RespiDxsx:2      0.52383802 1.018160e+00
## RespiDxurti:1    0.33587472 4.419836e-01
## RespiDxurti:2    0.46056832 7.968958e-01
```

```
exp(coef(modelkb))
```

```
##  (Intercept):1  (Intercept):2 RespiDxcovid:1 RespiDxcovid:2   RespiDxflu:1
##   3.337091e-01   6.538895e-02   2.307397e+02   1.121494e+03   9.005804e-01
##   RespiDxflu:2  RespiDxlrti:1  RespiDxlrti:2    RespiDxsx:1    RespiDxsx:2
##   1.432599e+00   3.413873e-01   8.872581e-01   3.343748e-01   7.303090e-01
##  RespiDxurti:1  RespiDxurti:2
##   3.852936e-01   6.058259e-01
```

```
exp(confintvglm(modeljb))
```

```
##                      2.5 %       97.5 %
## (Intercept):1   0.21731369   0.25399546
## (Intercept):2   0.05446084   0.07207194
## RespiDxcovid:1 15.97865891 125.00438100
## RespiDxcovid:2 55.08877820 439.63836952
## RespiDxflu:1    1.31282550   1.68067275
## RespiDxflu:2    1.14699406   1.78168382
## RespiDxlrti:1   0.72911216   0.90777771
## RespiDxlrti:2   0.83095296   1.21253082
## RespiDxsx:1     0.20416425   0.35499692
## RespiDxsx:2     0.27207467   0.64142400
## RespiDxurti:1   0.80914863   1.01286695
## RespiDxurti:2   0.72151610   1.08399350
```

```
exp(coef(modeljb))
```

```
##  (Intercept):1  (Intercept):2 RespiDxcovid:1 RespiDxcovid:2   RespiDxflu:1
##      0.2349398      0.0626506     44.6923077    155.6250000      1.4854057
##   RespiDxflu:2  RespiDxlrti:1  RespiDxlrti:2    RespiDxsx:1    RespiDxsx:2
##      1.4295387      0.8135550      1.0037709      0.2692168      0.4177502
##  RespiDxurti:1  RespiDxurti:2
##      0.9052955      0.8843748
```

```r
UVKB <- tbl_uvregression(
 data2[data2$City=='KB', c("ED", "API", "PM10", "PM2.5", "SO2", "CO3", "O3", "NO2")],
  method = glm,
  y = ED) %>%
  bold_p()

UVJB <- tbl_uvregression(
  data2[data2$City=='JB', c("ED", "API", "PM10", "PM2.5", "SO2", "CO3", "O3", "NO2")],
  method = glm,
  y = ED) %>%
  bold_p()

tbl_merge(tbls = list(UVKB, UVJB),
          tab_spanner = c("**Kota Bharu**", "**Johor Bahru**")
          )
```

### 2.2.3. Univariate analysis

```
## Table printed with `knitr::kable()`, not {gt}. Learn why at
## https://www.danieldsjoberg.com/gtsummary/articles/rmarkdown.html
## To suppress this message, include `message = FALSE` in code chunk header.
```

| Characteristic | N | Beta | 95% CI | p-value | N | Beta | 95% CI | p-value |
|---|---|---|---|---|---|---|---|---|
| API | 1,827 | 0.02 | -0.03, 0.06 | 0.4 | 1,827 | 0.04 | 0.02, 0.07 | **0.001** |
| PM10 | 1,827 | 0.11 | 0.06, 0.16 | **<0.001** | 1,827 | 0.10 | 0.07, 0.12 | **<0.001** |
| PM2.5 | 1,827 | 0.17 | 0.11, 0.24 | **<0.001** | 1,827 | 0.09 | 0.05, 0.12 | **<0.001** |
| SO2 | 1,827 | -3,939 | -5,310, -2,568 | **<0.001** | 1,827 | 881 | 573, 1,189 | **<0.001** |
| CO3 | 1,827 | 17 | 14, 20 | **<0.001** | 1,827 | 0.60 | -5.9, 7.1 | 0.9 |
| O3 | 1,827 | -8.1 | -20, 4.2 | 0.2 | 1,827 | 6.3 | 4.8, 7.7 | **<0.001** |
| NO2 | 1,827 | 519 | 377, 662 | **<0.001** | 1,827 | 110 | 62, 158 | **<0.001** |

## 3. Association analysis

Change of variable format

```r
kbdata$time <- as.numeric(kbdata$Date)/100
kbdata$doy <- as.numeric(format(kbdata$Date, format = "%j"))
kbdata$year <- format(kbdata$Date, format = "%Y")


jbdata$time <- as.numeric(jbdata$Date)/100
jbdata$doy <- as.numeric(format(jbdata$Date, format = "%j"))
jbdata$year <- format(jbdata$Date, format = "%Y")

kbdata$Rainfall2 <- kbdata$Rainfall^(1/3)
kbdata$O32 <- 1/kbdata$O3
```

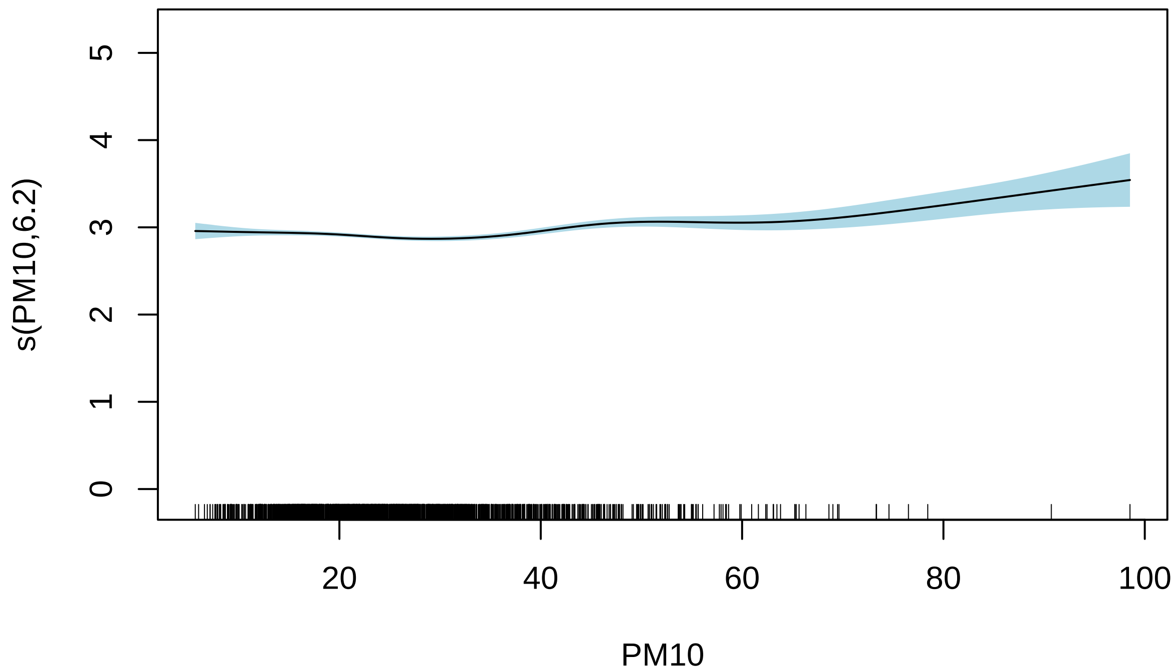
### 3.1. KB ED

```r
kb_total <- gam(ED ~ s(PM10) + s(NO2) + s(CO3) + s(SO2) + s(O3) + s(Mean) + s(Rainfall) + s(Wind) + s(do
summary(kb_total)
```

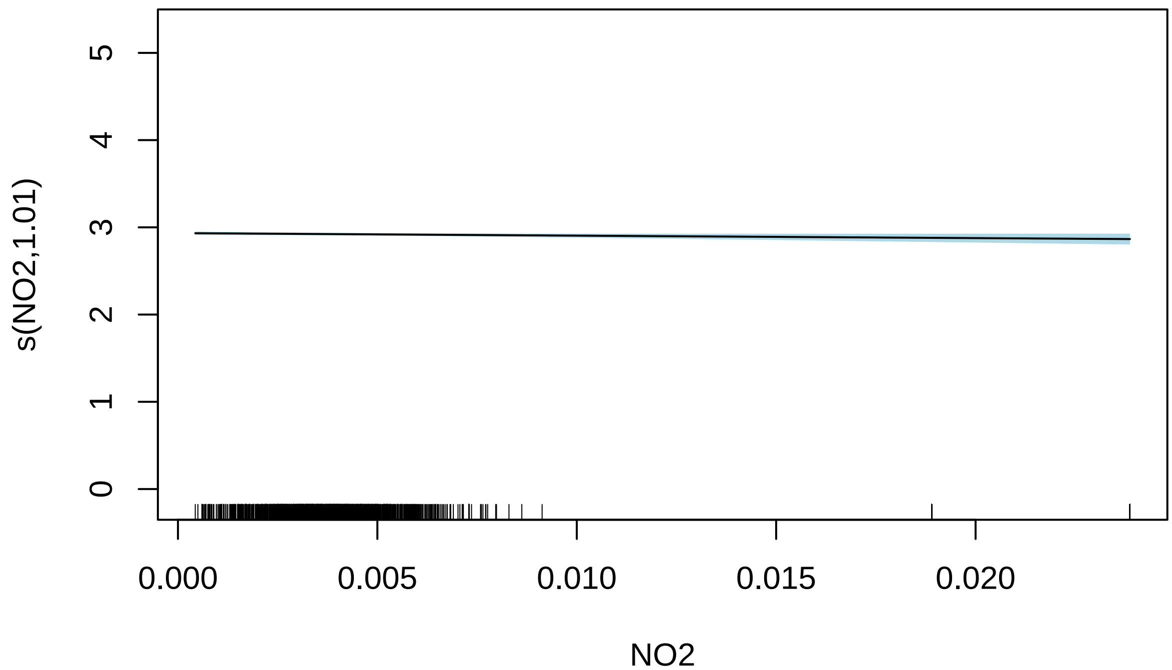
#### 3.1.1. Model selection and checking

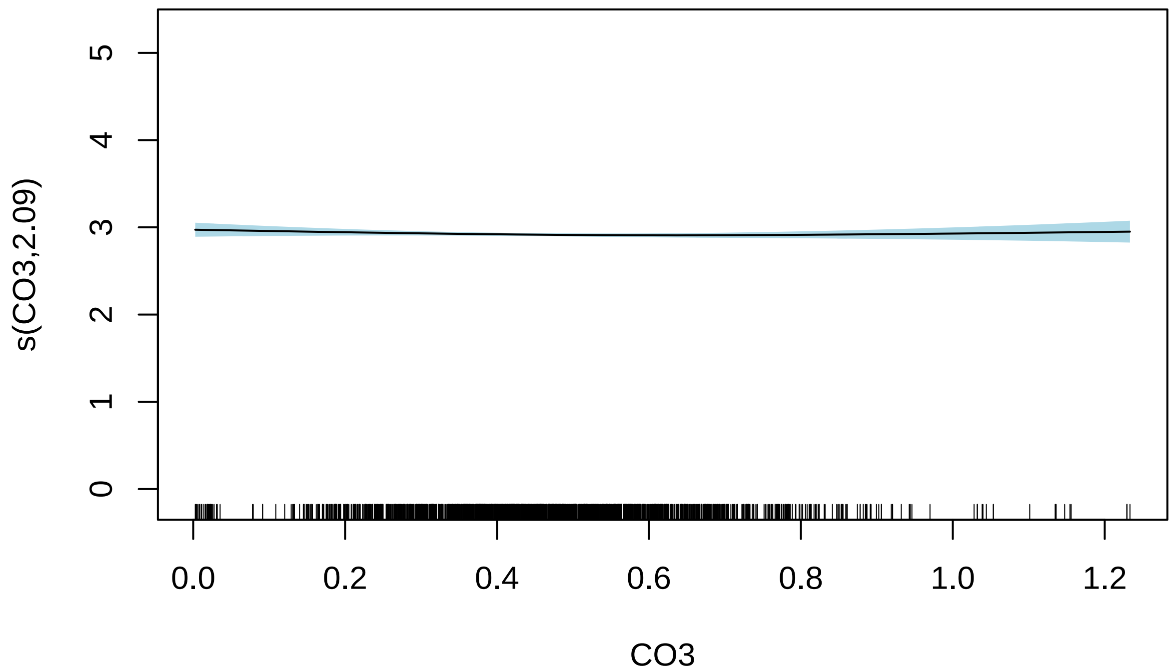
```
##
## Family: poisson
## Link function: log
##
## Formula:
## ED ~ s(PM10) + s(NO2) + s(CO3) + s(SO2) + s(O3) + s(Mean) + s(Rainfall) +
##     s(Wind) + s(doy, bs = "cc", k = 365) + s(time, k = 80) +
##     Period
##
## Parametric coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  2.92034    0.07273  40.152  < 2e-16 ***
## Period2     -0.45337    0.18034  -2.514 0.011937 *
## Period3     -0.85047    0.24651  -3.450 0.000561 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Approximate significance of smooth terms:
##                 edf  Ref.df   Chi.sq p-value
## s(PM10)       6.204   7.351   70.375 < 2e-16 ***
## s(NO2)        1.013   1.025    3.267 0.07250 .
## s(CO3)        2.086   2.707    2.676 0.36508
## s(SO2)        4.770   5.753   26.127 0.00016 ***
## s(O3)         8.278   8.833  107.328 < 2e-16 ***
## s(Mean)       5.725   6.822   17.035 0.01416 *
## s(Rainfall)   1.855   2.317    9.969 0.01080 *
## s(Wind)       3.890   4.810   41.850 < 2e-16 ***
## s(doy)      128.505 363.000  536.164 < 2e-16 ***
## s(time)      57.828  62.868 1887.077 < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-sq.(adj) =  0.679   Deviance explained = 76.2%
## -REML = 6741.2  Scale est. = 1         n = 1827
```

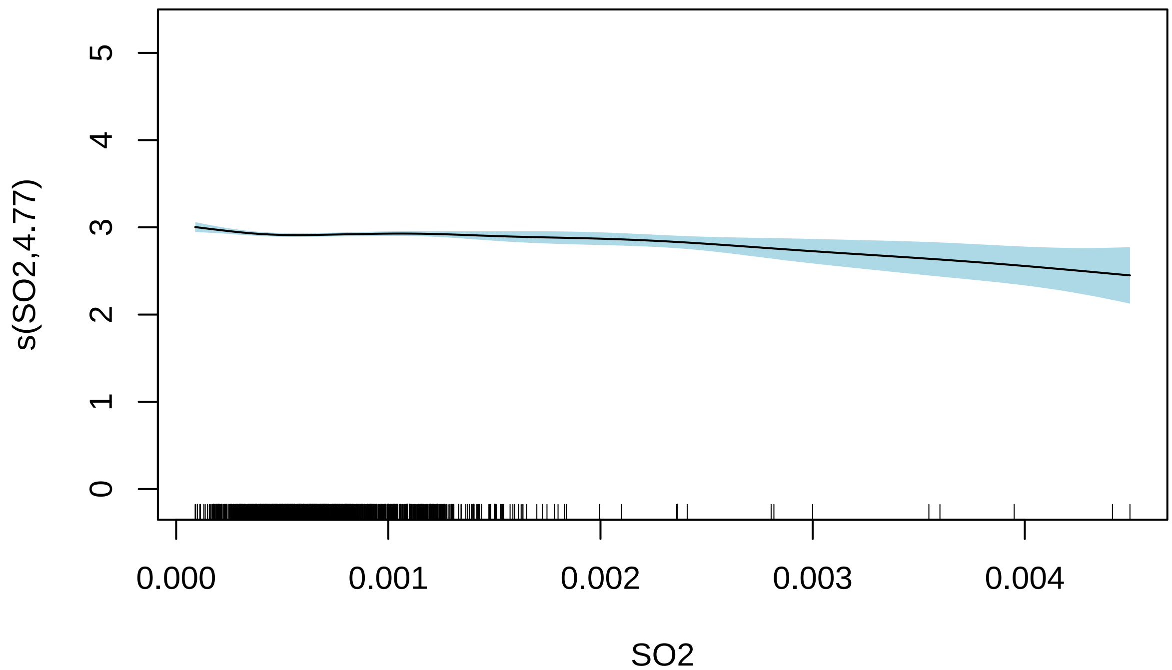
```r
plot.gam(kb_total, shade = TRUE, shade.col = "lightblue", shift = coef(kb_total)[1],  seWithMean = TRUE
```



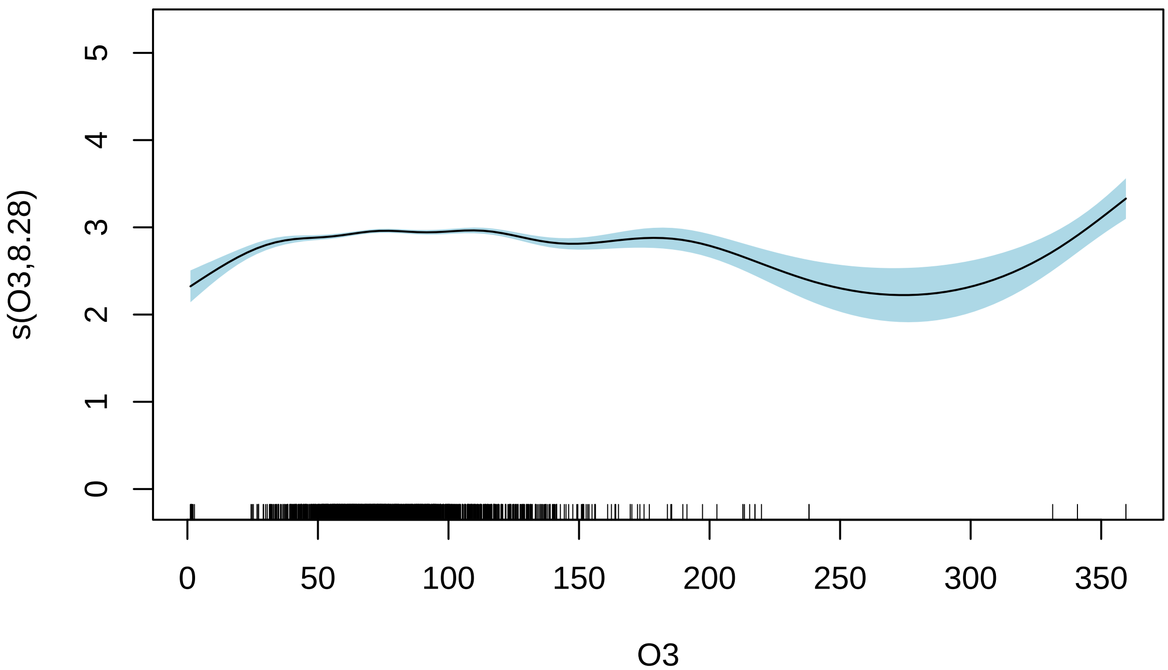
s(PM10,6.2)
0
1
2
3
4
5
20
40
60
80
100
PM10


s(NO2,1.01)
0
1
2
3
4
5
0.000
0.005
0.010
0.015
0.020
NO2

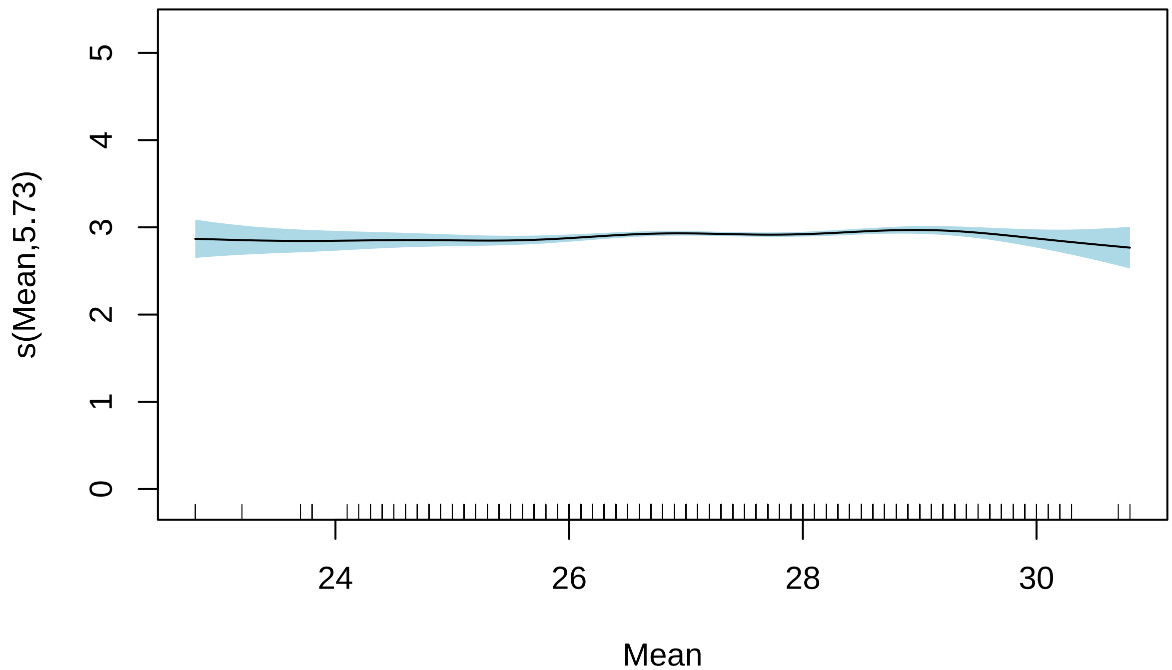

s(CO3,2.09)
0
1
2
3
4
5
0.0
0.2
0.4
0.6
0.8
1.0
1.2
CO3



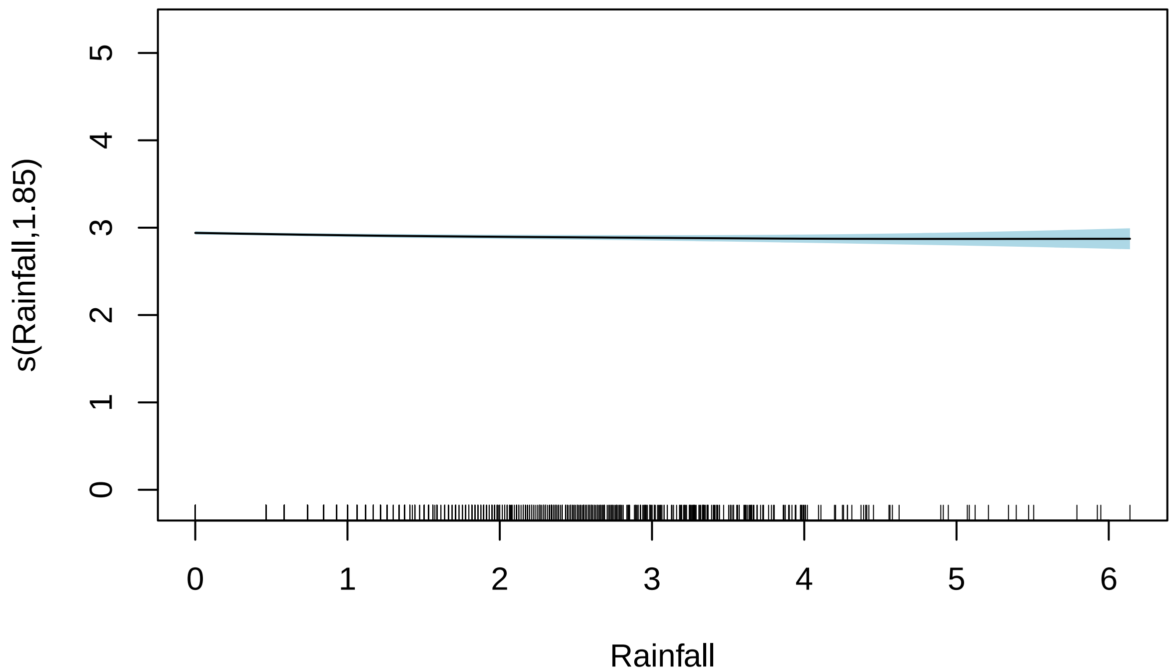
s(SO2,4.77)
0
1
2
3
4
5
0.000
0.001
0.002
0.003
0.004
SO2



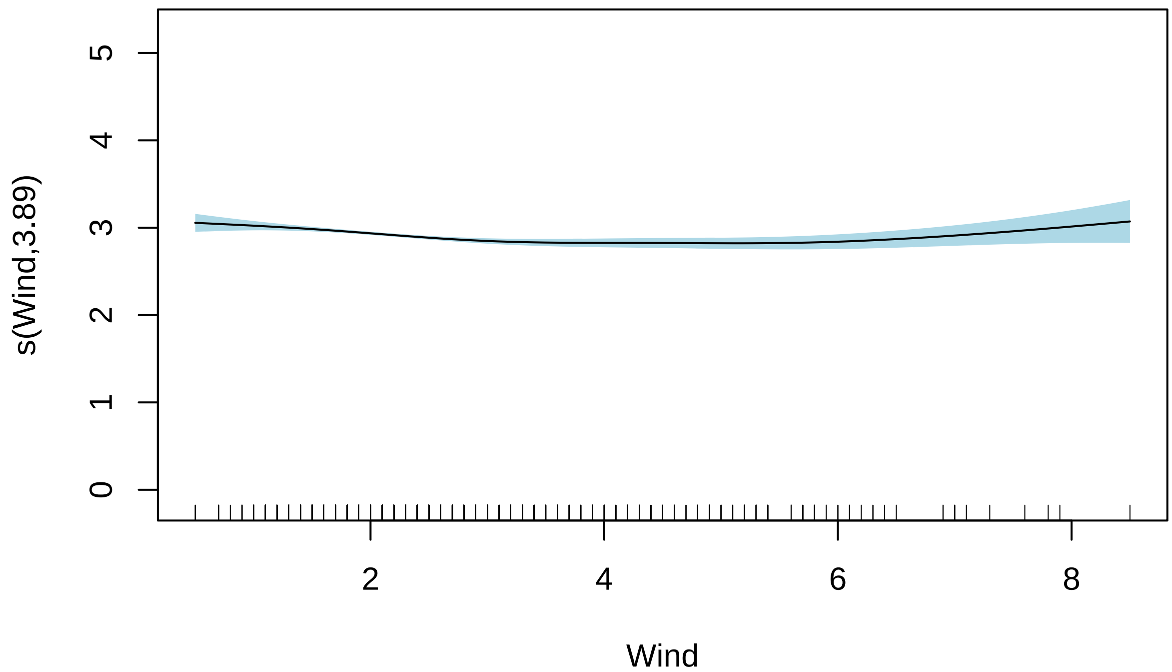
s(O3,8.28)
0
1
2
3
4
5
0
50
100
150
200
250
300
350
O3



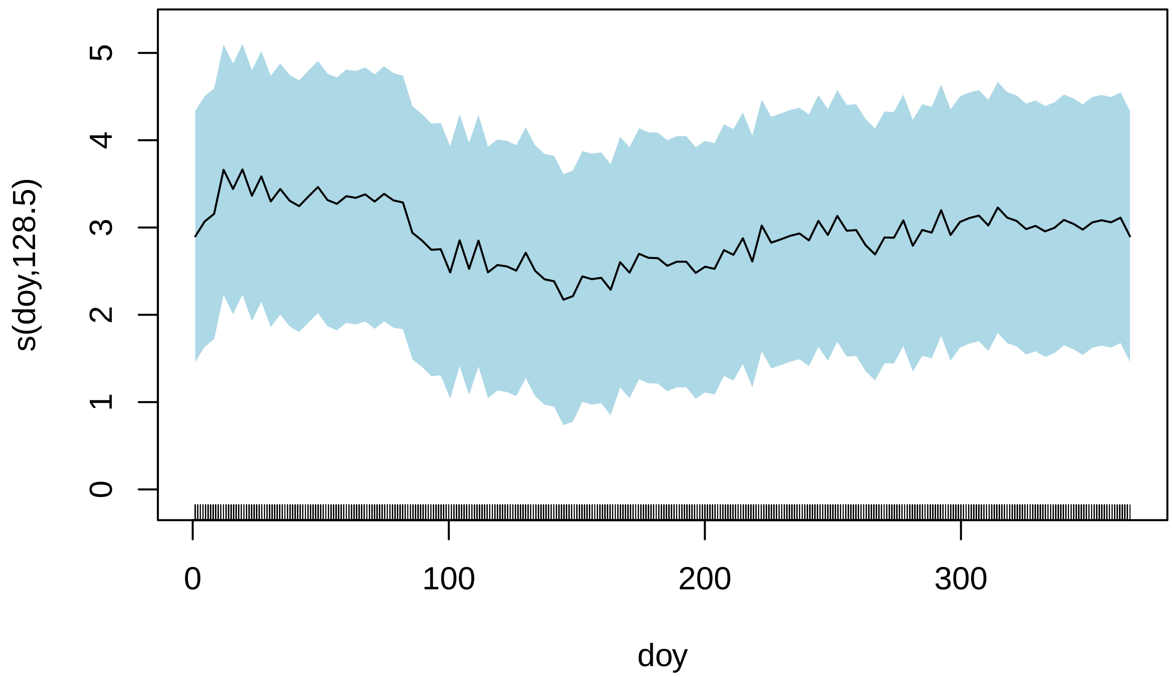
s(Mean,5.73)
0
1
2
3
4
5
24
26
28
30
Mean



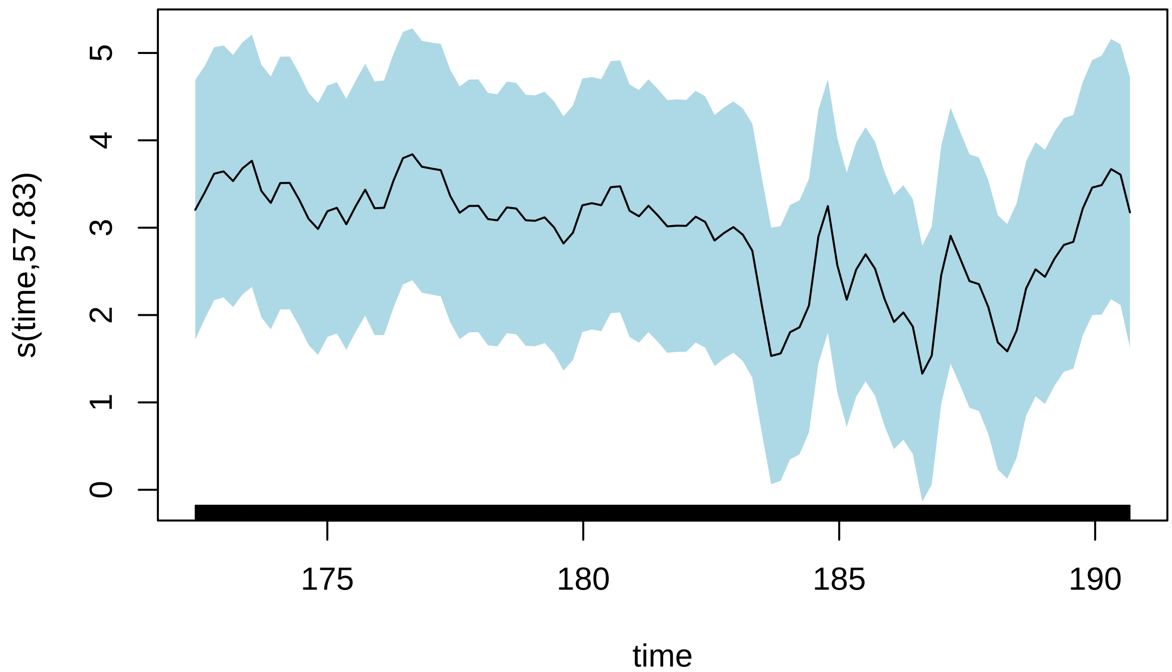
s(Rainfall,1.85)
0
1
2
3
4
5
0
1
2
3
4
5
6
Rainfall



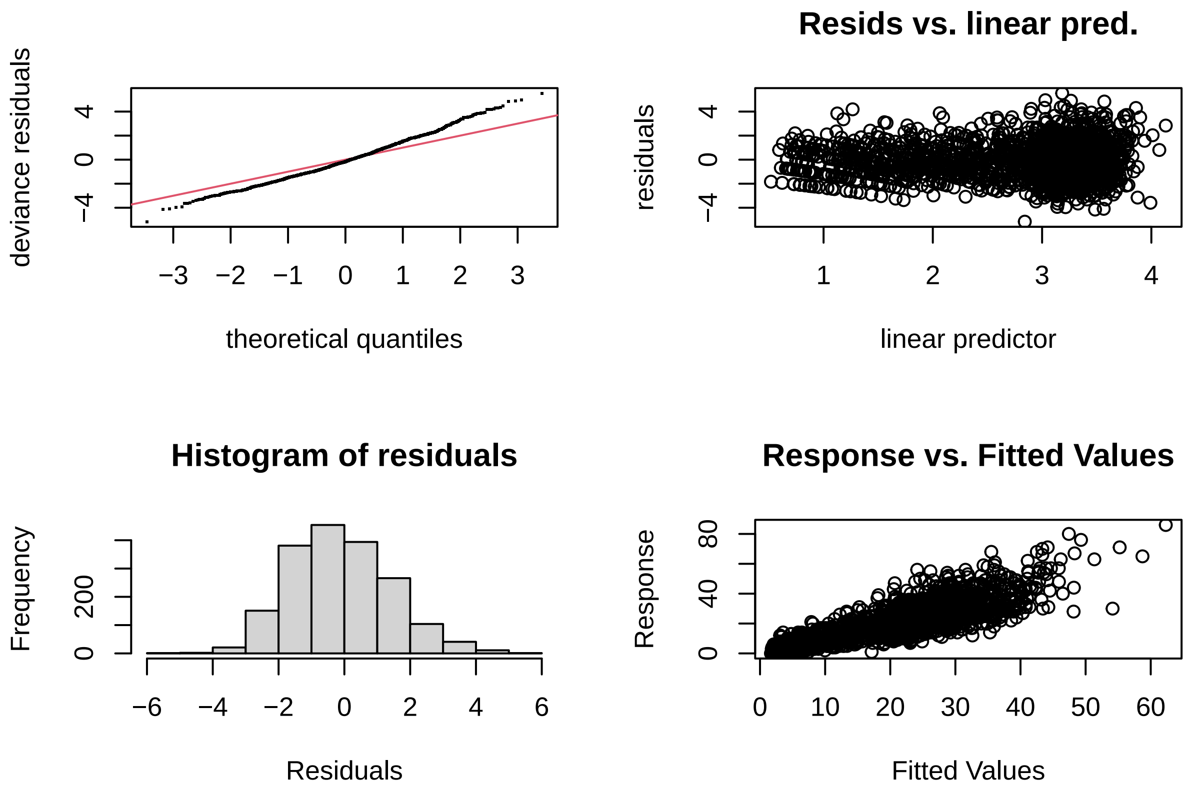
s(Wind,3.89)
0
1
2
3
4
5
2
4
6
8
Wind



s(doy,128.5)
0
1
2
3
4
5
0
100
200
300
doy




```
gam.check(kb_total)
```




```
##
## Method: REML   Optimizer: outer newton
## full convergence after 9 iterations.
## Gradient range [-0.003754816,0.0001854597]
## (score 6741.206 & scale 1).
## Hessian positive definite, eigenvalue range [0.003737993,20.73303].
## Model rank =  517 / 517
##
## Basis dimension (k) checking results. Low p-value (k-index<1) may
## indicate that k is too low, especially if edf is close to k'.
##
##                 k'    edf k-index p-value
## s(PM10)       9.00   6.20    1.03    0.93
## s(NO2)        9.00   1.01    1.04    0.94
## s(CO3)        9.00   2.09    0.98    0.24
## s(SO2)        9.00   4.77    1.00    0.59
## s(O3)         9.00   8.28    1.03    0.92
## s(Mean)       9.00   5.73    1.01    0.76
## s(Rainfall)   9.00   1.85    1.00    0.56
## s(Wind)       9.00   3.89    0.98    0.24
## s(doy)      363.00 128.50    1.09    1.00
## s(time)      79.00  57.83    0.91  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Checking for concurvity

```r
concurvity(kb_total, full = TRUE)
```

```
##                para   s(PM10)    s(NO2)    s(CO3)    s(SO2)     s(O3)   s(Mean)
## worst    0.9946778 0.7219770 0.6503853 0.7819525 0.5785787 0.6636402 0.8313246
## observed 0.9946778 0.5078476 0.5803016 0.7244342 0.3764171 0.4998676 0.6210195
## estimate 0.9946778 0.6424822 0.5493960 0.6753138 0.4301517 0.3958626 0.7416858
##          s(Rainfall)   s(Wind)     s(doy)   s(time)
## worst      0.6028750 0.7307935 1.00000000 1.0000000
## observed   0.5536688 0.6778514 0.90634334 0.7285082
## estimate   0.5269794 0.6476035 0.08835029 0.8289316
```

```r
concurvity(kb_total, full = FALSE)
```

```
## $worst
##                     para      s(PM10)       s(NO2)       s(CO3)       s(SO2)
## para        1.000000e+00 9.199415e-25 1.633921e-23 7.654002e-26 3.224342e-24
## s(PM10)     9.298561e-25 1.000000e+00 8.186587e-02 4.775189e-01 3.110101e-02
## s(NO2)      1.630980e-23 8.186587e-02 1.000000e+00 1.874808e-01 1.443973e-01
## s(CO3)      7.562525e-26 4.775189e-01 1.874808e-01 1.000000e+00 8.515451e-02
## s(SO2)      3.228128e-24 3.110101e-02 1.443973e-01 8.515451e-02 1.000000e+00
## s(O3)       2.925355e-24 9.104588e-02 1.088520e-01 3.435849e-01 2.874823e-02
## s(Mean)     1.087111e-26 1.594769e-01 5.332458e-02 1.372516e-02 3.347065e-02
## s(Rainfall) 1.959930e-23 1.229726e-01 2.305877e-02 1.601719e-02 1.546627e-02
## s(Wind)     1.746125e-17 4.621260e-02 2.649943e-01 2.614712e-02 1.685886e-02
## s(doy)      6.472162e-28 2.698112e-01 2.677204e-01 2.438119e-01 2.661177e-01
## s(time)     1.652710e-17 2.771264e-01 4.797937e-01 4.898406e-01 3.775484e-01
##                    s(O3)      s(Mean)  s(Rainfall)      s(Wind)       s(doy)
## para        2.930819e-24 1.060998e-26 1.960396e-23 1.746122e-17 6.466954e-28
## s(PM10)     9.104588e-02 1.594769e-01 1.229726e-01 4.621260e-02 2.698112e-01
## s(NO2)      1.088520e-01 5.332458e-02 2.305877e-02 2.649943e-01 2.677204e-01
## s(CO3)      3.435849e-01 1.372516e-02 1.601719e-02 2.614712e-02 2.438119e-01
## s(SO2)      2.874823e-02 3.347065e-02 1.546627e-02 1.685886e-02 2.661177e-01
## s(O3)       1.000000e+00 6.747102e-02 4.338534e-02 1.425481e-01 2.708820e-01
## s(Mean)     6.747102e-02 1.000000e+00 2.795349e-01 1.217727e-01 5.937521e-01
## s(Rainfall) 4.338534e-02 2.795349e-01 1.000000e+00 9.405340e-02 3.842862e-01
## s(Wind)     1.425481e-01 1.217727e-01 9.405340e-02 1.000000e+00 4.578728e-01
## s(doy)      2.708820e-01 5.937521e-01 3.842862e-01 4.578728e-01 1.000000e+00
## s(time)     2.379103e-01 6.269204e-01 2.555699e-01 4.271586e-01 1.000000e+00
##                  s(time)
## para        1.652579e-17
## s(PM10)     2.771264e-01
## s(NO2)      4.797937e-01
## s(CO3)      4.898406e-01
## s(SO2)      3.775484e-01
## s(O3)       2.379103e-01
## s(Mean)     6.269204e-01
## s(Rainfall) 2.555699e-01
## s(Wind)     4.271586e-01
## s(doy)      1.000000e+00
## s(time)     1.000000e+00
##
## $observed
```

```
##                     para      s(PM10)       s(NO2)       s(CO3)       s(SO2)
## para        1.000000e+00 2.433743e-27 4.768859e-32 4.093268e-30 1.575121e-27
## s(PM10)     9.298561e-25 1.000000e+00 8.412873e-03 1.060618e-01 1.307777e-02
## s(NO2)      1.630980e-23 4.696417e-02 1.000000e+00 1.057569e-01 7.618278e-03
## s(CO3)      7.562525e-26 2.025675e-01 6.283002e-02 1.000000e+00 9.156956e-03
## s(SO2)      3.228128e-24 1.575986e-02 1.056569e-01 4.259889e-02 1.000000e+00
## s(O3)       2.925355e-24 2.079489e-02 1.392841e-02 9.360214e-02 3.714093e-03
## s(Mean)     1.087111e-26 1.921880e-02 4.596360e-03 7.643955e-03 4.806152e-03
## s(Rainfall) 1.959930e-23 6.276566e-03 3.613742e-03 8.750530e-03 3.399691e-03
## s(Wind)     1.746125e-17 1.732127e-02 1.124098e-02 1.507289e-02 4.912794e-03
## s(doy)      6.472162e-28 1.946044e-01 1.367227e-01 1.692346e-01 2.340966e-01
## s(time)     1.652710e-17 1.059776e-01 4.174717e-01 4.345771e-01 1.259826e-01
##                    s(O3)      s(Mean)  s(Rainfall)      s(Wind)       s(doy)
## para        3.420185e-25 5.220295e-28 8.527501e-30 3.177377e-23 3.805813e-30
## s(PM10)     3.326389e-03 6.679494e-02 8.392019e-02 2.291416e-02 1.198139e-02
## s(NO2)      2.795978e-02 1.147849e-02 6.210794e-03 8.891094e-02 2.147193e-02
## s(CO3)      1.700903e-01 4.921620e-03 5.048579e-03 1.805869e-02 2.062347e-02
## s(SO2)      9.670138e-03 2.211366e-02 7.161868e-03 8.701526e-03 2.031934e-02
## s(O3)       1.000000e+00 3.804980e-02 3.799170e-02 1.134142e-01 3.851405e-02
## s(Mean)     1.554923e-02 1.000000e+00 1.907975e-01 6.027116e-02 3.300388e-01
## s(Rainfall) 1.486849e-02 1.408820e-01 1.000000e+00 2.750321e-02 5.306261e-03
## s(Wind)     2.274615e-02 6.491626e-02 4.024417e-02 1.000000e+00 1.297542e-01
## s(doy)      1.796519e-01 3.853637e-01 2.963837e-01 3.912000e-01 1.000000e+00
## s(time)     9.985385e-02 3.502504e-01 2.350360e-01 3.699679e-01 8.955866e-01
##                  s(time)
## para        1.017775e-20
## s(PM10)     5.203233e-03
## s(NO2)      1.850217e-01
## s(CO3)      1.081699e-01
## s(SO2)      1.121596e-01
## s(O3)       1.320602e-02
## s(Mean)     1.492646e-02
## s(Rainfall) 2.683220e-03
## s(Wind)     9.678485e-03
## s(doy)      2.187963e-03
## s(time)     1.000000e+00
##
## $estimate
##                     para      s(PM10)       s(NO2)       s(CO3)       s(SO2)
## para        1.000000e+00 8.580581e-27 1.026711e-26 5.335137e-28 1.179882e-26
## s(PM10)     9.298561e-25 1.000000e+00 6.306198e-03 2.613814e-01 1.387541e-02
## s(NO2)      1.630980e-23 4.660729e-02 1.000000e+00 9.353095e-02 2.418839e-02
## s(CO3)      7.562525e-26 2.997216e-01 4.800781e-02 1.000000e+00 2.341136e-02
## s(SO2)      3.228128e-24 9.082469e-03 1.087308e-01 2.278135e-02 1.000000e+00
## s(O3)       2.925355e-24 5.537000e-02 1.516773e-02 4.656674e-02 1.086470e-02
## s(Mean)     1.087111e-26 3.378748e-02 4.398202e-03 4.160526e-03 8.621464e-03
## s(Rainfall) 1.959930e-23 4.448022e-02 3.241504e-03 6.770717e-03 4.485910e-03
## s(Wind)     1.746125e-17 8.898718e-03 7.114278e-03 1.206632e-02 4.558058e-03
## s(doy)      6.472162e-28 2.083627e-01 1.464349e-01 1.788483e-01 2.128614e-01
## s(time)     1.652710e-17 1.746976e-01 3.770889e-01 2.833233e-01 1.942725e-01
##                    s(O3)      s(Mean)  s(Rainfall)      s(Wind)       s(doy)
## para        1.959644e-26 6.983978e-29 2.554342e-26 7.935544e-20 1.992387e-30
## s(PM10)     3.124229e-02 6.755369e-02 8.658179e-02 1.731109e-02 5.114938e-03
## s(NO2)      3.824129e-02 1.421656e-02 8.550262e-03 1.690829e-01 4.955584e-03
```

```
## s(CO3)      1.788986e-02 5.864852e-03 4.426618e-03 1.895237e-02 4.775029e-03
## s(SO2)      8.983447e-03 1.361214e-02 5.495924e-03 6.052490e-03 5.185592e-03
## s(O3)       1.000000e+00 4.035224e-02 3.052591e-02 9.332373e-02 5.258749e-03
## s(Mean)     1.656702e-02 1.000000e+00 1.928749e-01 3.989011e-02 7.060233e-03
## s(Rainfall) 1.420353e-02 1.555186e-01 1.000000e+00 1.724694e-02 5.722659e-03
## s(Wind)     6.321685e-02 6.956467e-02 3.644168e-02 1.000000e+00 6.875511e-03
## s(doy)      2.125147e-01 4.924280e-01 2.894482e-01 3.719426e-01 1.000000e+00
## s(time)     1.109351e-01 4.975934e-01 2.063716e-01 3.115058e-01 4.533858e-02
##                  s(time)
## para        9.929493e-23
## s(PM10)     1.961127e-02
## s(NO2)      7.427583e-02
## s(CO3)      1.867846e-01
## s(SO2)      4.254509e-02
## s(O3)       2.269668e-02
## s(Mean)     2.022616e-02
## s(Rainfall) 7.396160e-03
## s(Wind)     4.229221e-02
## s(doy)      4.970174e-02
## s(time)     1.000000e+00
```

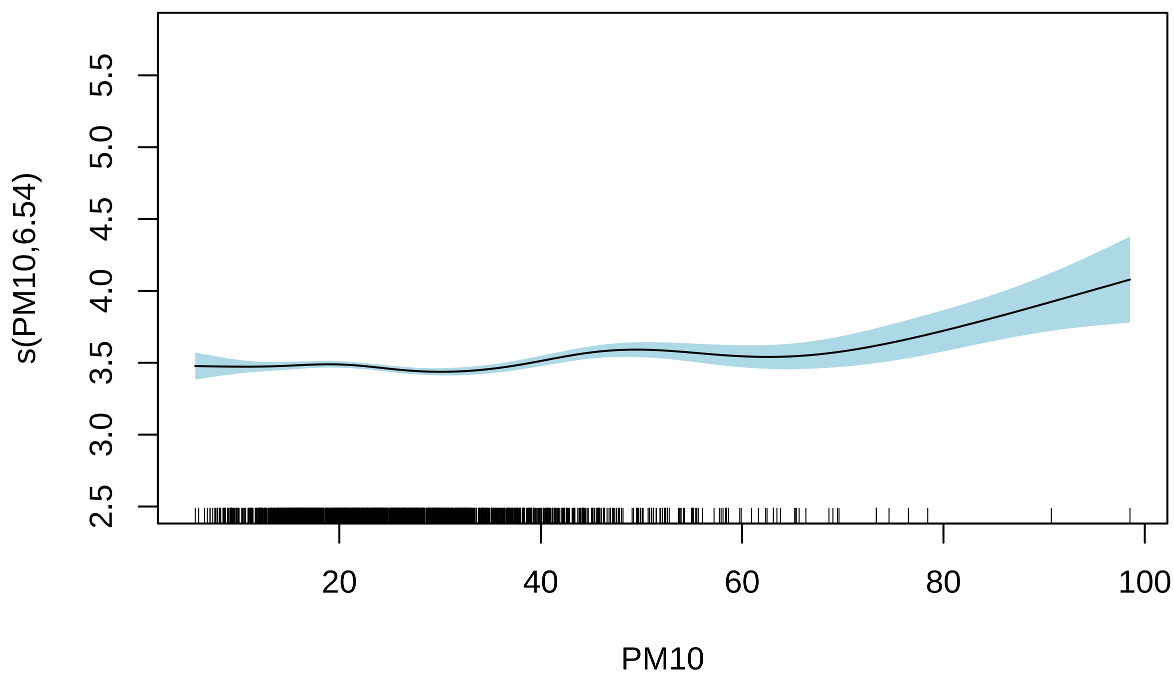
A value of more than 0.8 is considered to have high concurvity between 2 variables. 2 pairings are noted to have high concurvity, PM10-PM2.5, and CO-O3. PM2.5 is excluded from the final model. However, Co and O3 is maintained until after comparison with JB data

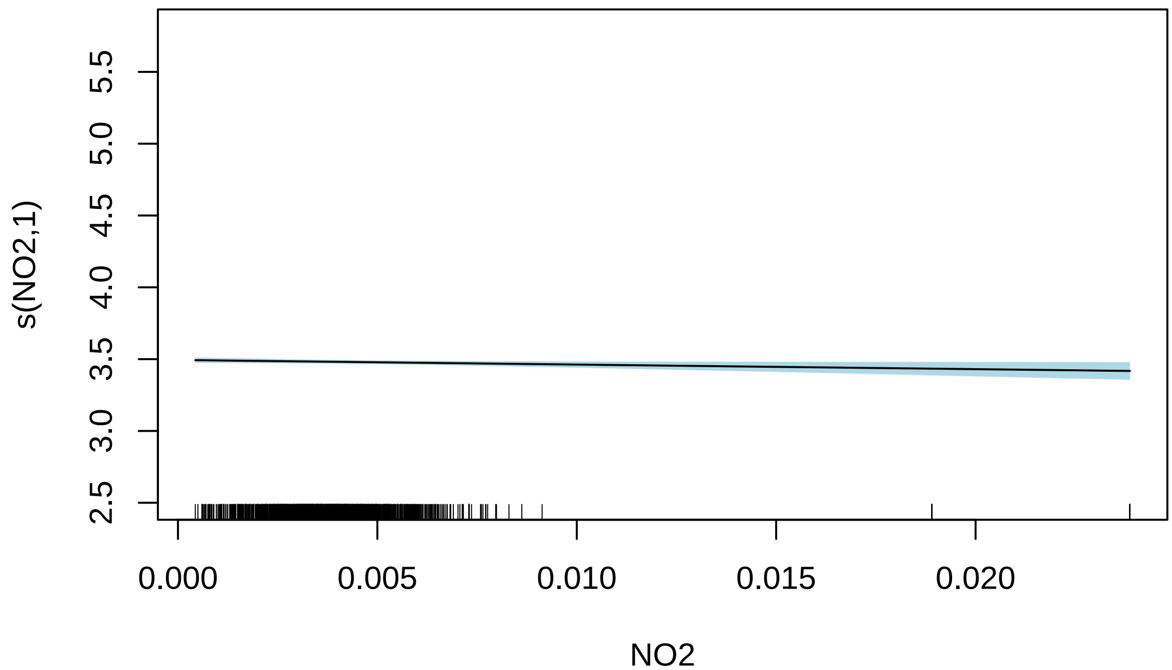
```
kb_nopm2.5 <- gam(ED ~ s(PM10) + s(NO2) + s(CO3) + s(SO2) + s(O3) + s(Mean) + s(Rainfall) + s(Wind) + s
summary(kb_nopm2.5)
```

```
##
## Family: poisson
## Link function: log
##
## Formula:
## ED ~ s(PM10) + s(NO2) + s(CO3) + s(SO2) + s(O3) + s(Mean) + s(Rainfall) +
##     s(Wind) + s(doy, bs = "cc", k = 365) + s(time) + Period
##
## Parametric coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  3.47931    0.03239  107.40   <2e-16 ***
## Period2     -1.74673    0.08320  -21.00   <2e-16 ***
## Period3     -2.26039    0.10989  -20.57   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Approximate significance of smooth terms:
##                 edf  Ref.df   Chi.sq  p-value
## s(PM10)       6.541   7.666   52.109  < 2e-16 ***
## s(NO2)        1.000   1.001    4.208 0.040292 *
## s(CO3)        1.002   1.004    2.861 0.091002 .
## s(SO2)        8.200   8.785   81.354  < 2e-16 ***
## s(O3)         8.250   8.828  123.021  < 2e-16 ***
## s(Mean)       6.920   7.904   64.285  < 2e-16 ***
## s(Rainfall)   1.887   2.364   16.110 0.000644 ***
```

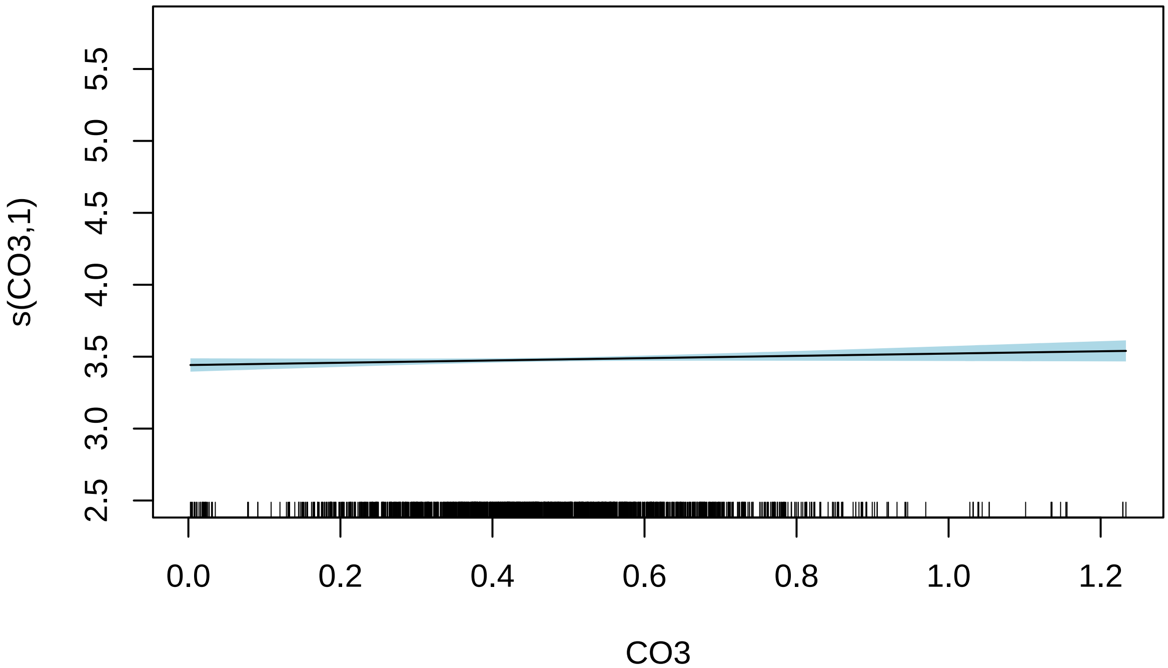
```
## s(Wind)       5.712   6.793   51.015  < 2e-16 ***
## s(doy)      131.300 363.000 1045.696  < 2e-16 ***
## s(time)       8.901   8.996 1018.449  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-sq.(adj) =  0.635   Deviance explained = 70.2%
## -REML = 7155.4  Scale est. = 1         n = 1827
```

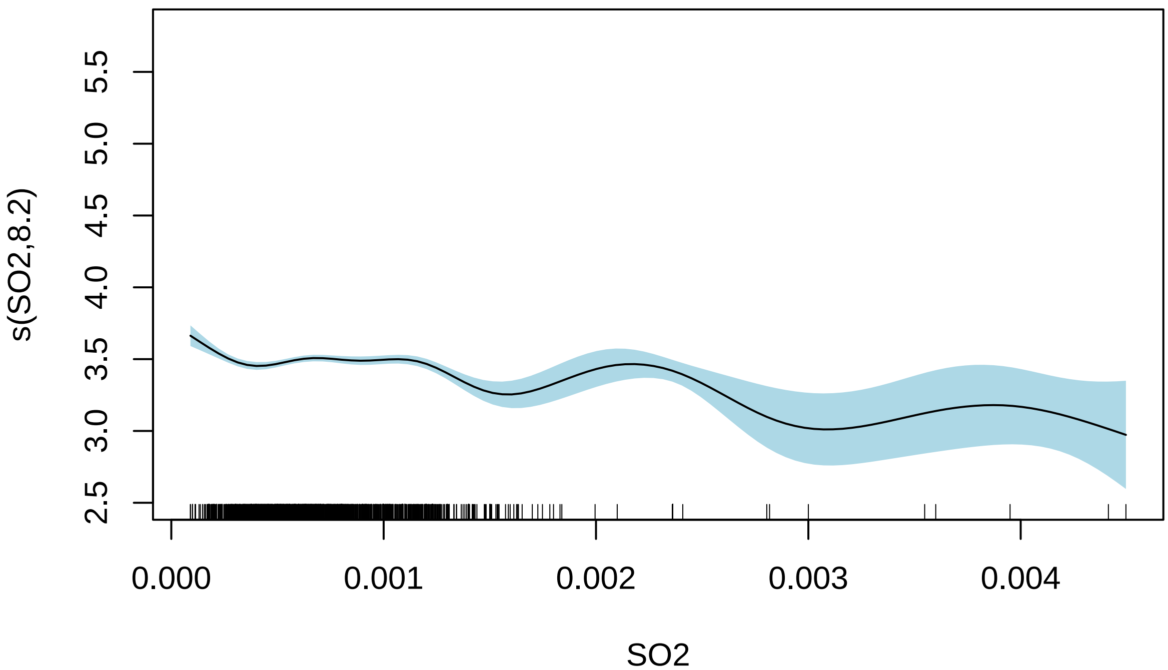
```
plot.gam(kb_nopm2.5, shade = TRUE, shade.col = "lightblue", shift = coef(kb_nopm2.5)[1],  seWithMean = '
```



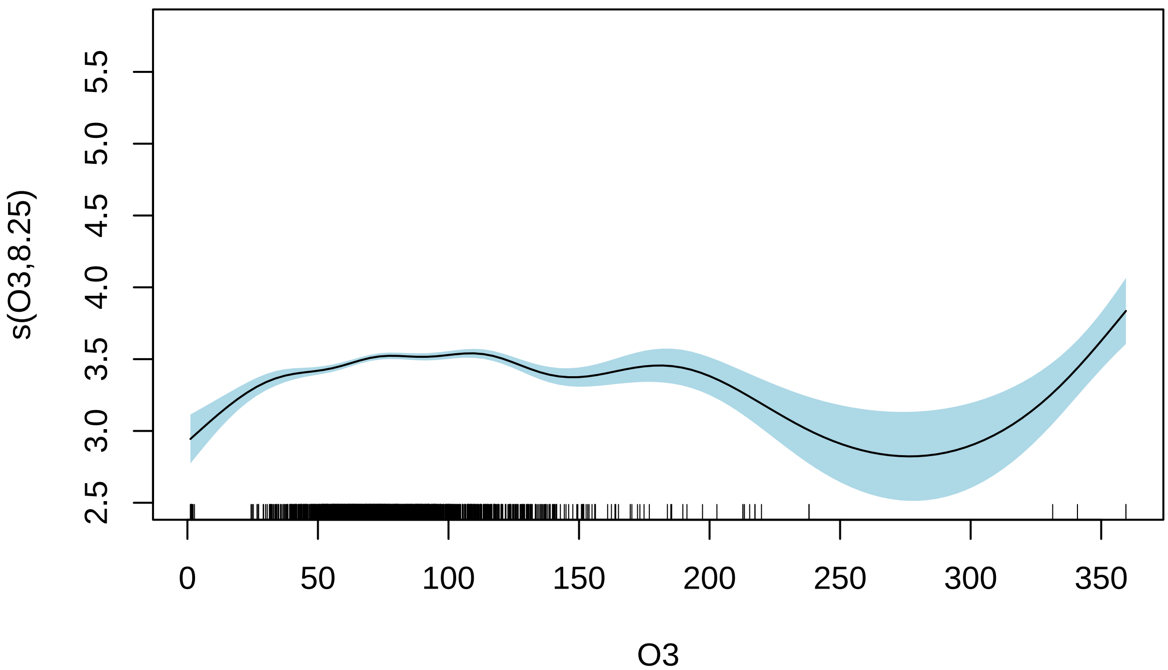


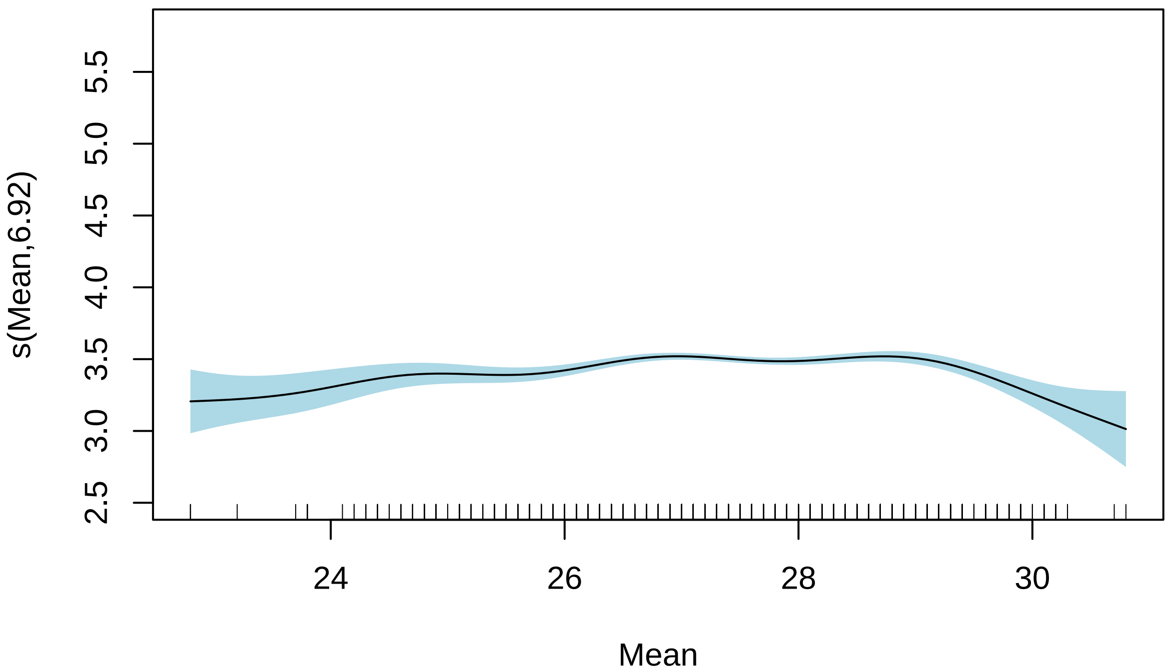
s(NO2,1)
2.5
3.0
3.5
4.0
4.5
5.0
5.5
0.000
0.005
0.010
0.015
0.020
NO2


s(CO3,1)
2.5
3.0
3.5
4.0
4.5
5.0
5.5
0.0
0.2
0.4
0.6
0.8
1.0
1.2
CO3

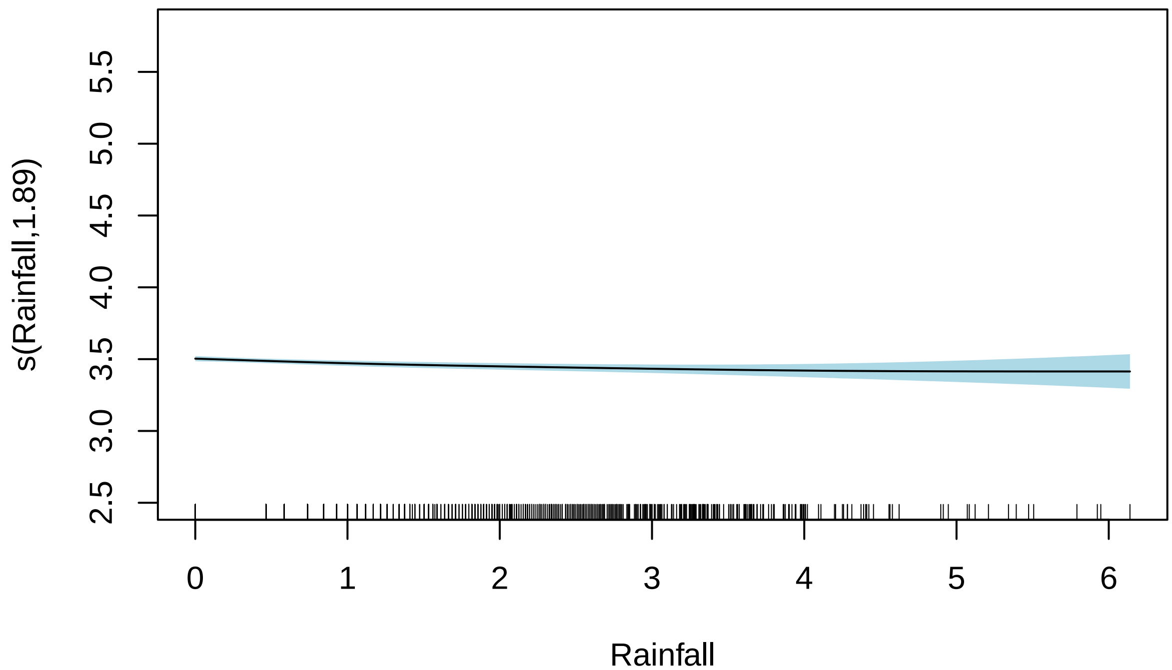


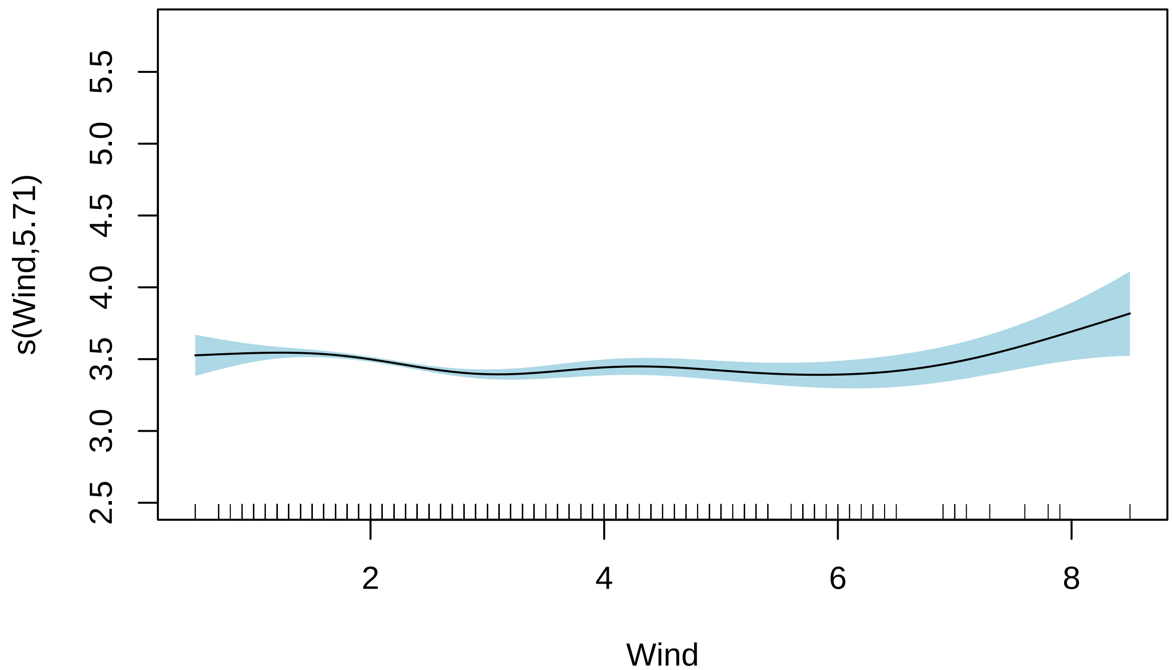
s(SO2,8.2)
2.5
3.0
3.5
4.0
4.5
5.0
5.5
0.000
0.001
0.002
0.003
0.004
SO2


s(O3,8.25)
2.5
3.0
3.5
4.0
4.5
5.0
5.5
0
50
100
150
200
250
300
350
O3

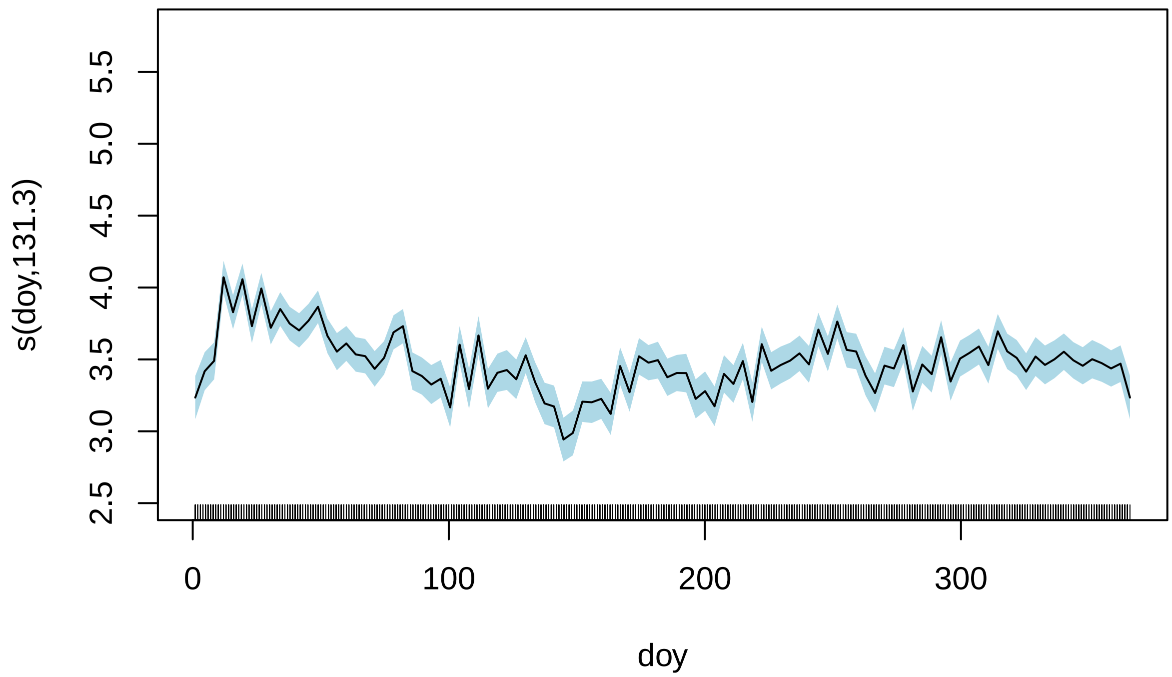


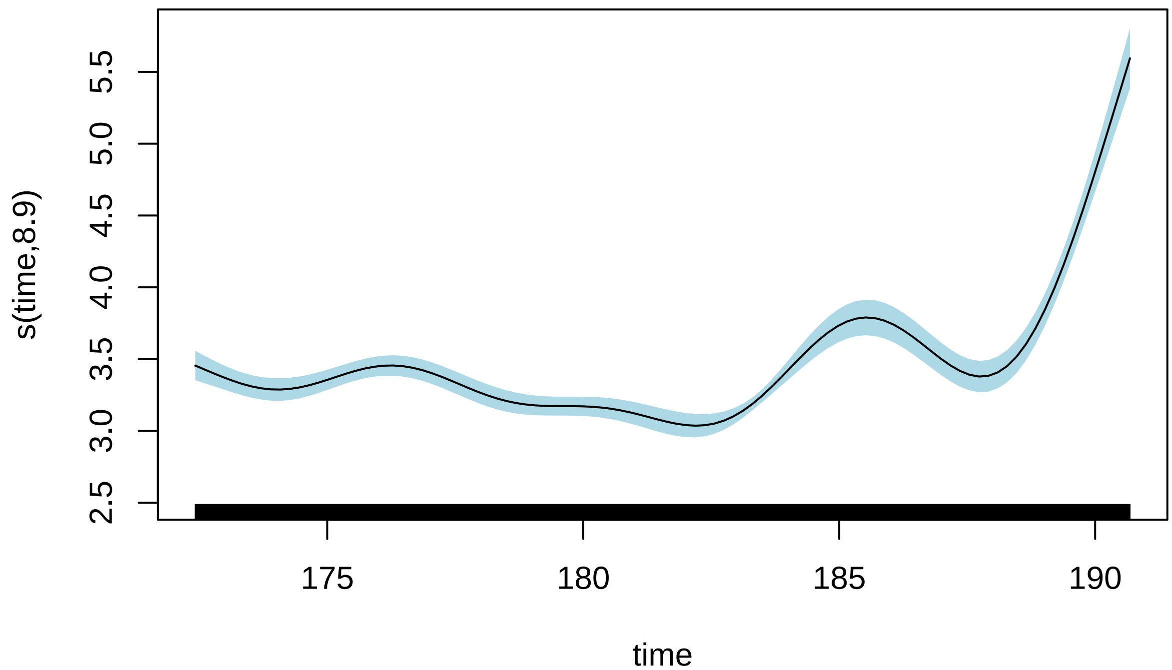
s(Mean,6.92)
2.5
3.0
3.5
4.0
4.5
5.0
5.5
24
26
28
30
Mean


s(Rainfall,1.89)
2.5
3.0
3.5
4.0
4.5
5.0
5.5
0
1
2
3
4
5
6
Rainfall

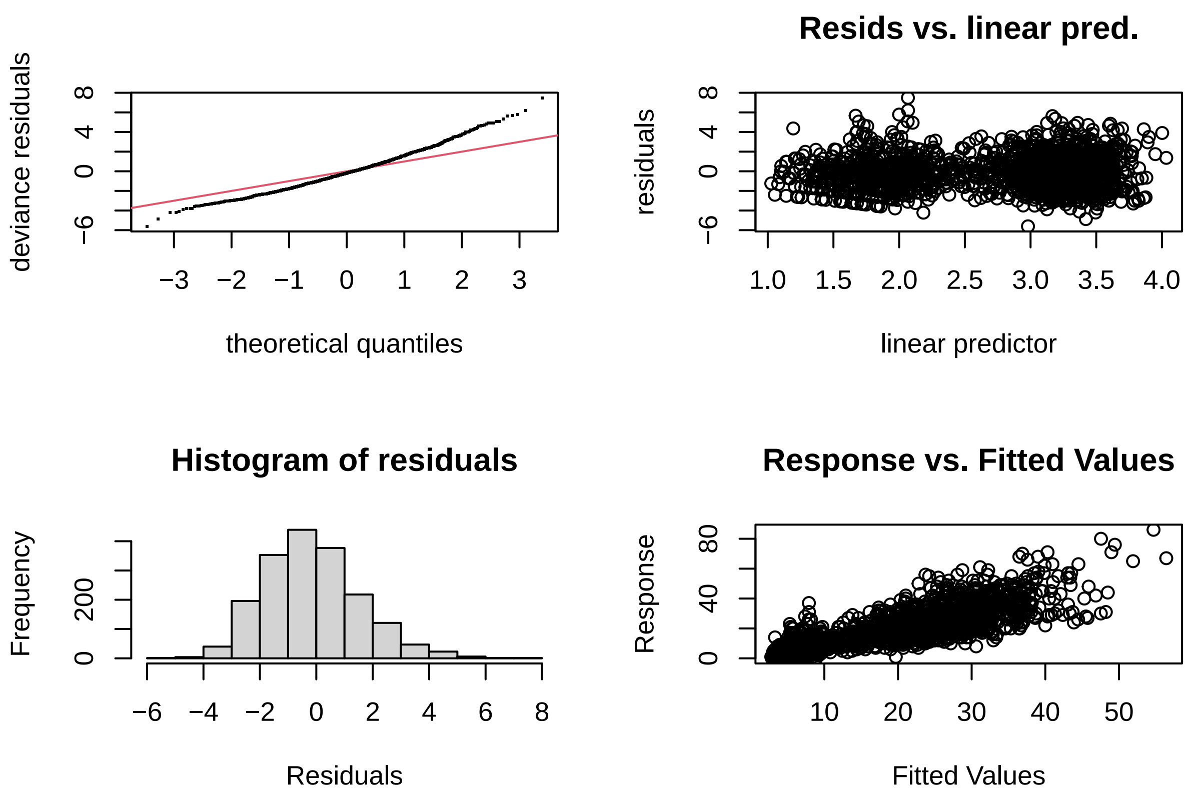


s(Wind,5.71)
2.5
3.0
3.5
4.0
4.5
5.0
5.5
2
4
6
8
Wind


s(doy,131.3)
2.5
3.0
3.5
4.0
4.5
5.0
5.5
0
100
200
300
doy




```
gam.check(kb_nopm2.5)
```




```
##
## Method: REML   Optimizer: outer newton
## full convergence after 9 iterations.
## Gradient range [-0.0002176046,4.652811e-05]
## (score 7155.441 & scale 1).
## Hessian positive definite, eigenvalue range [1.897562e-05,18.66696].
## Model rank =  447 / 447
##
## Basis dimension (k) checking results. Low p-value (k-index<1) may
## indicate that k is too low, especially if edf is close to k'.
##
##                  k'    edf k-index p-value
## s(PM10)        9.00   6.54    1.02   0.850
## s(NO2)         9.00   1.00    0.98   0.310
## s(CO3)         9.00   1.00    0.97   0.150
## s(SO2)         9.00   8.20    0.93   0.015 *
## s(O3)          9.00   8.25    1.04   0.965
## s(Mean)        9.00   6.92    0.93  <2e-16 ***
## s(Rainfall)    9.00   1.89    0.86  <2e-16 ***
## s(Wind)        9.00   5.71    0.89  <2e-16 ***
## s(doy)       363.00 131.30    1.13   1.000
## s(time)        9.00   8.90    0.73  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
concurvity(kb_nopm2.5, full = FALSE)
```

```
## $worst
##                     para      s(PM10)       s(NO2)       s(CO3)       s(SO2)
## para        1.000000e+00 9.199415e-25 1.633921e-23 7.654002e-26 3.224342e-24
## s(PM10)     9.298561e-25 1.000000e+00 8.186587e-02 4.775189e-01 3.110101e-02
## s(NO2)      1.630980e-23 8.186587e-02 1.000000e+00 1.874808e-01 1.443973e-01
## s(CO3)      7.562525e-26 4.775189e-01 1.874808e-01 1.000000e+00 8.515451e-02
## s(SO2)      3.228128e-24 3.110101e-02 1.443973e-01 8.515451e-02 1.000000e+00
## s(O3)       2.925355e-24 9.104588e-02 1.088520e-01 3.435849e-01 2.874823e-02
## s(Mean)     1.087111e-26 1.594769e-01 5.332458e-02 1.372516e-02 3.347065e-02
## s(Rainfall) 1.959930e-23 1.229726e-01 2.305877e-02 1.601719e-02 1.546627e-02
## s(Wind)     1.746125e-17 4.621260e-02 2.649943e-01 2.614712e-02 1.685886e-02
## s(doy)      6.472162e-28 2.698112e-01 2.677204e-01 2.438119e-01 2.661177e-01
## s(time)     2.182486e-26 8.167639e-02 4.084787e-01 2.934596e-01 2.233100e-01
##                    s(O3)      s(Mean)  s(Rainfall)      s(Wind)       s(doy)
## para        2.930819e-24 1.060998e-26 1.960396e-23 1.746122e-17 6.466954e-28
## s(PM10)     9.104588e-02 1.594769e-01 1.229726e-01 4.621260e-02 2.698112e-01
## s(NO2)      1.088520e-01 5.332458e-02 2.305877e-02 2.649943e-01 2.677204e-01
## s(CO3)      3.435849e-01 1.372516e-02 1.601719e-02 2.614712e-02 2.438119e-01
## s(SO2)      2.874823e-02 3.347065e-02 1.546627e-02 1.685886e-02 2.661177e-01
## s(O3)       1.000000e+00 6.747102e-02 4.338534e-02 1.425481e-01 2.708820e-01
## s(Mean)     6.747102e-02 1.000000e+00 2.795349e-01 1.217727e-01 5.937521e-01
## s(Rainfall) 4.338534e-02 2.795349e-01 1.000000e+00 9.405340e-02 3.842862e-01
## s(Wind)     1.425481e-01 1.217727e-01 9.405340e-02 1.000000e+00 4.578728e-01
## s(doy)      2.708820e-01 5.937521e-01 3.842862e-01 4.578728e-01 1.000000e+00
## s(time)     7.419843e-02 2.025860e-01 2.569032e-02 7.463623e-02 3.634025e-01
##                  s(time)
## para        1.894750e-26
## s(PM10)     8.167639e-02
## s(NO2)      4.084787e-01
## s(CO3)      2.934596e-01
## s(SO2)      2.233100e-01
## s(O3)       7.419843e-02
## s(Mean)     2.025860e-01
## s(Rainfall) 2.569032e-02
## s(Wind)     7.463623e-02
## s(doy)      3.634025e-01
## s(time)     1.000000e+00
##
## $observed
##                     para      s(PM10)       s(NO2)       s(CO3)       s(SO2)
## para        1.000000e+00 3.684174e-27 3.992936e-32 1.549747e-32 2.954217e-25
## s(PM10)     9.298561e-25 1.000000e+00 8.316722e-03 2.665467e-01 1.741572e-02
## s(NO2)      1.630980e-23 3.312977e-02 1.000000e+00 1.353366e-01 1.164453e-02
## s(CO3)      7.562525e-26 1.567665e-01 6.242034e-02 1.000000e+00 1.091167e-02
## s(SO2)      3.228128e-24 1.379664e-02 1.058633e-01 2.521392e-02 1.000000e+00
## s(O3)       2.925355e-24 2.486332e-02 1.390765e-02 4.924290e-02 8.513405e-03
## s(Mean)     1.087111e-26 1.489102e-02 4.547512e-03 5.707056e-03 1.688518e-02
## s(Rainfall) 1.959930e-23 5.275706e-03 3.599080e-03 9.520266e-03 3.053017e-03
## s(Wind)     1.746125e-17 1.240438e-02 1.102928e-02 1.488142e-02 9.933416e-03
## s(doy)      6.472162e-28 2.010842e-01 1.368581e-01 1.786314e-01 2.217657e-01
## s(time)     2.182486e-26 8.116375e-03 3.545164e-01 1.848354e-01 3.253604e-02
```

```
##                     s(O3)      s(Mean)  s(Rainfall)      s(Wind)       s(doy)
## para         3.115650e-25 3.313044e-28 1.413081e-29 5.017855e-21 9.282121e-31
## s(PM10)      1.104650e-02 3.857798e-02 8.580545e-02 1.823558e-02 7.288525e-03
## s(NO2)       4.129995e-02 1.532828e-02 6.404598e-03 5.819705e-02 1.591371e-02
## s(CO3)       1.361054e-01 4.594102e-03 4.816996e-03 1.401007e-02 1.047149e-02
## s(SO2)       1.101680e-02 1.662293e-02 6.980006e-03 9.325007e-03 3.153985e-02
## s(O3)        1.000000e+00 1.279844e-02 3.799992e-02 8.156888e-02 2.553019e-02
## s(Mean)      1.296990e-02 1.000000e+00 1.942358e-01 5.358952e-02 2.099928e-01
## s(Rainfall) 1.110788e-02 5.559940e-02 1.000000e+00 3.309505e-02 3.781140e-03
## s(Wind)      2.746686e-02 3.591401e-02 4.014398e-02 1.000000e+00 7.097780e-02
## s(doy)       1.800150e-01 2.691188e-01 2.979611e-01 3.714660e-01 1.000000e+00
## s(time)      3.012173e-02 2.530221e-02 1.940082e-02 4.909799e-02 1.076918e-01
##                   s(time)
## para         8.631283e-29
## s(PM10)      4.133207e-02
## s(NO2)       1.084880e-01
## s(CO3)       1.380341e-01
## s(SO2)       8.682545e-02
## s(O3)        1.331999e-02
## s(Mean)      6.278959e-02
## s(Rainfall) 1.755435e-02
## s(Wind)      4.015591e-02
## s(doy)       8.541329e-02
## s(time)      1.000000e+00
##
## $estimate
##                      para      s(PM10)       s(NO2)       s(CO3)       s(SO2)
## para         1.000000e+00 8.580581e-27 1.026711e-26 5.335137e-28 1.179882e-26
## s(PM10)      9.298561e-25 1.000000e+00 6.306198e-03 2.613814e-01 1.387541e-02
## s(NO2)       1.630980e-23 4.660729e-02 1.000000e+00 9.353095e-02 2.418839e-02
## s(CO3)       7.562525e-26 2.997216e-01 4.800781e-02 1.000000e+00 2.341136e-02
## s(SO2)       3.228128e-24 9.082469e-03 1.087308e-01 2.278135e-02 1.000000e+00
## s(O3)        2.925355e-24 5.537000e-02 1.516773e-02 4.656674e-02 1.086470e-02
## s(Mean)      1.087111e-26 3.378748e-02 4.398202e-03 4.160526e-03 8.621464e-03
## s(Rainfall) 1.959930e-23 4.448022e-02 3.241504e-03 6.770717e-03 4.485910e-03
## s(Wind)      1.746125e-17 8.898718e-03 7.114278e-03 1.206632e-02 4.558058e-03
## s(doy)       6.472162e-28 2.083627e-01 1.464349e-01 1.788483e-01 2.128614e-01
## s(time)      2.182486e-26 4.719470e-02 3.172082e-01 1.358470e-01 8.076702e-02
##                     s(O3)      s(Mean)  s(Rainfall)      s(Wind)       s(doy)
## para         1.959644e-26 6.983978e-29 2.554342e-26 7.935544e-20 1.992387e-30
## s(PM10)      3.124229e-02 6.755369e-02 8.658179e-02 1.731109e-02 5.114938e-03
## s(NO2)       3.824129e-02 1.421656e-02 8.550262e-03 1.690829e-01 4.955584e-03
## s(CO3)       1.788986e-02 5.864852e-03 4.426618e-03 1.895237e-02 4.775029e-03
## s(SO2)       8.983447e-03 1.361214e-02 5.495924e-03 6.052490e-03 5.185592e-03
## s(O3)        1.000000e+00 4.035224e-02 3.052591e-02 9.332373e-02 5.258749e-03
## s(Mean)      1.656702e-02 1.000000e+00 1.928749e-01 3.989011e-02 7.060233e-03
## s(Rainfall) 1.420353e-02 1.555186e-01 1.000000e+00 1.724694e-02 5.722659e-03
## s(Wind)      6.321685e-02 6.956467e-02 3.644168e-02 1.000000e+00 6.875511e-03
## s(doy)       2.125147e-01 4.924280e-01 2.894482e-01 3.719426e-01 1.000000e+00
## s(time)      2.101573e-02 1.378994e-01 1.678148e-02 5.951218e-02 1.400184e-03
##                   s(time)
## para         7.448868e-29
## s(PM10)      1.564021e-02
## s(NO2)       1.028051e-01
```

```
## s(CO3)      1.635950e-01
## s(SO2)      5.130988e-02
## s(O3)       2.328199e-02
## s(Mean)     1.391682e-02
## s(Rainfall) 5.195076e-03
## s(Wind)     3.406911e-02
## s(doy)      2.969835e-02
## s(time)     1.000000e+00
```

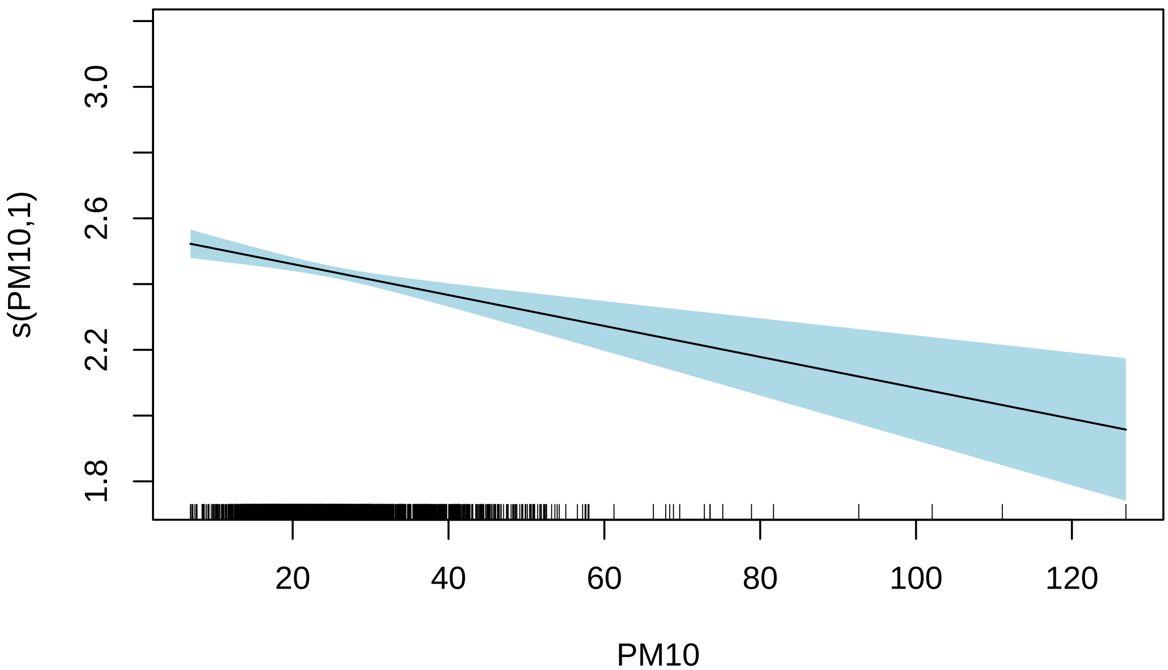
Much better. Proceed with comparison with JB data

JB model

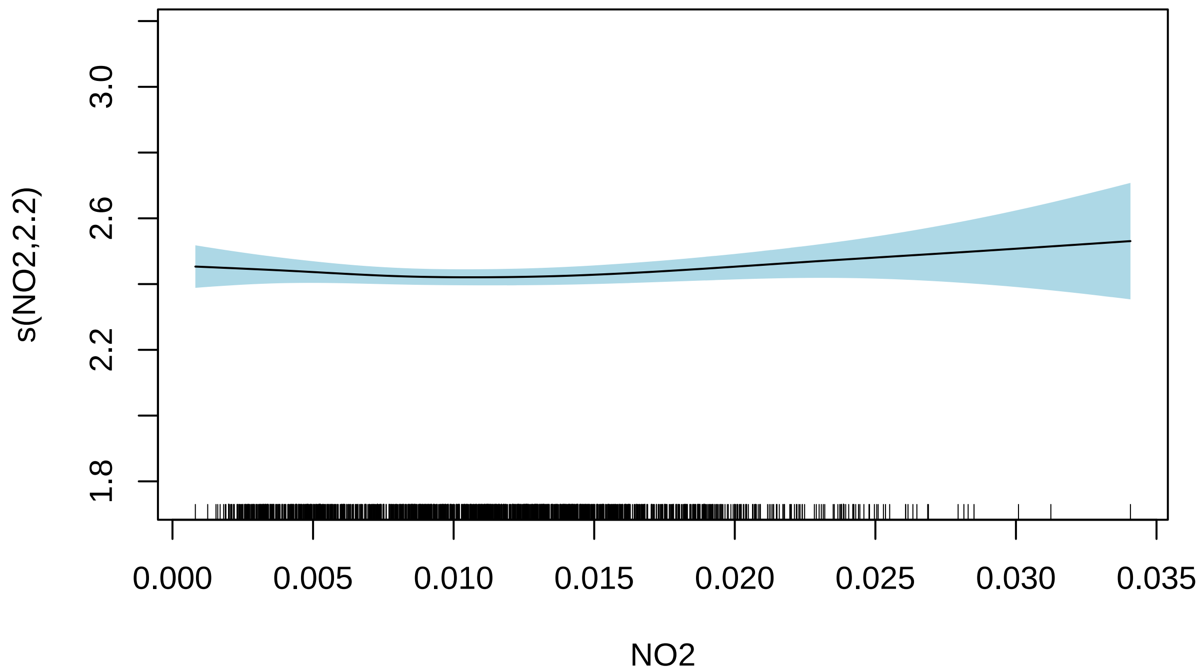
```
jb_mod <- gam(ED ~ s(PM10) + s(NO2) + s(CO3) + s(SO2) + s(O3) + s(Mean) + s(Rainfall) + s(Wind) + s(doy
summary(jb_mod)
```

```
##
## Family: poisson
## Link function: log
##
## Formula:
## ED ~ s(PM10) + s(NO2) + s(CO3) + s(SO2) + s(O3) + s(Mean) + s(Rainfall) +
##     s(Wind) + s(doy, bs = "cc", k = 365) + s(time, bs = "cc") +
##     Period
##
## Parametric coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  2.43415    0.03200  76.073   <2e-16 ***
## Period2     -0.90209    0.08481 -10.636   <2e-16 ***
## Period3     -0.10785    0.06790  -1.588    0.112
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Approximate significance of smooth terms:
##                edf  Ref.df  Chi.sq  p-value
## s(PM10)      1.001   1.001  19.602 1.02e-05 ***
## s(NO2)       2.197   2.810   3.776   0.2883
## s(CO3)       1.005   1.009   0.017   0.9140
## s(SO2)       4.334   5.363  12.040   0.0427 *
## s(O3)        7.599   8.474  57.845  < 2e-16 ***
## s(Mean)      6.142   7.166  18.022   0.0123 *
## s(Rainfall)  1.772   2.211   5.756   0.0731 .
## s(Wind)      4.425   5.409   9.449   0.1491
## s(doy)      93.993 363.000 470.456  < 2e-16 ***
## s(time)      7.687   8.000 236.271  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-sq.(adj) =   0.46   Deviance explained = 48.5%
## -REML =   5867  Scale est. = 1         n = 1827
```

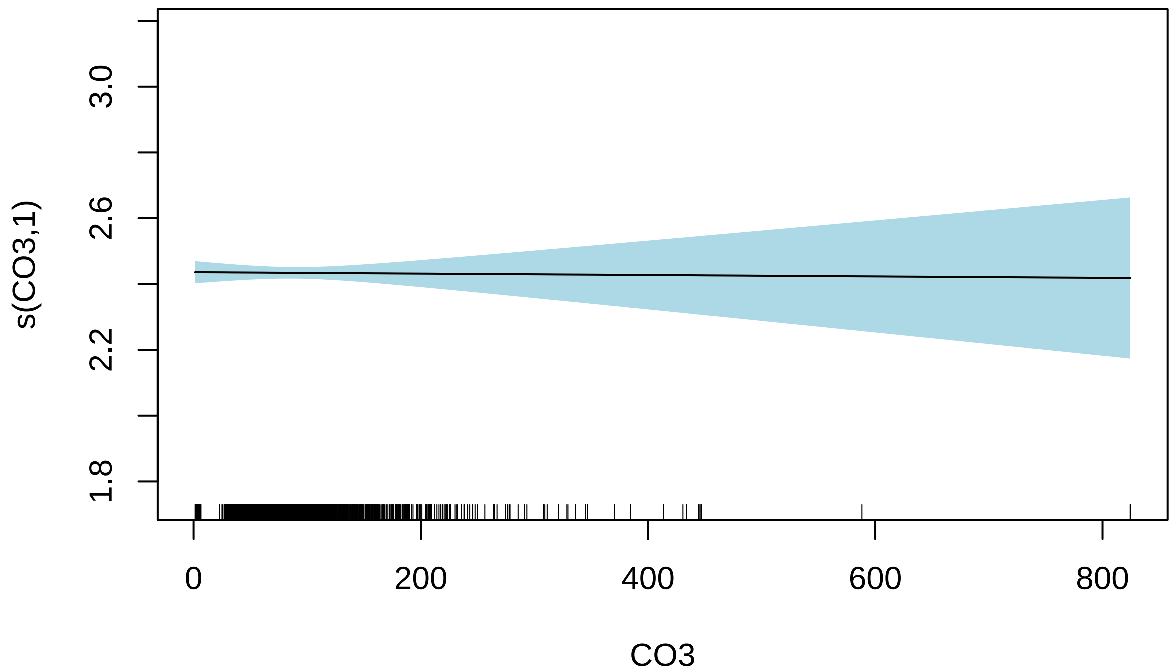
```
plot.gam(jb_mod, shade = TRUE, shade.col = "lightblue", shift = coef(jb_mod)[1],  seWithMean = TRUE)
```



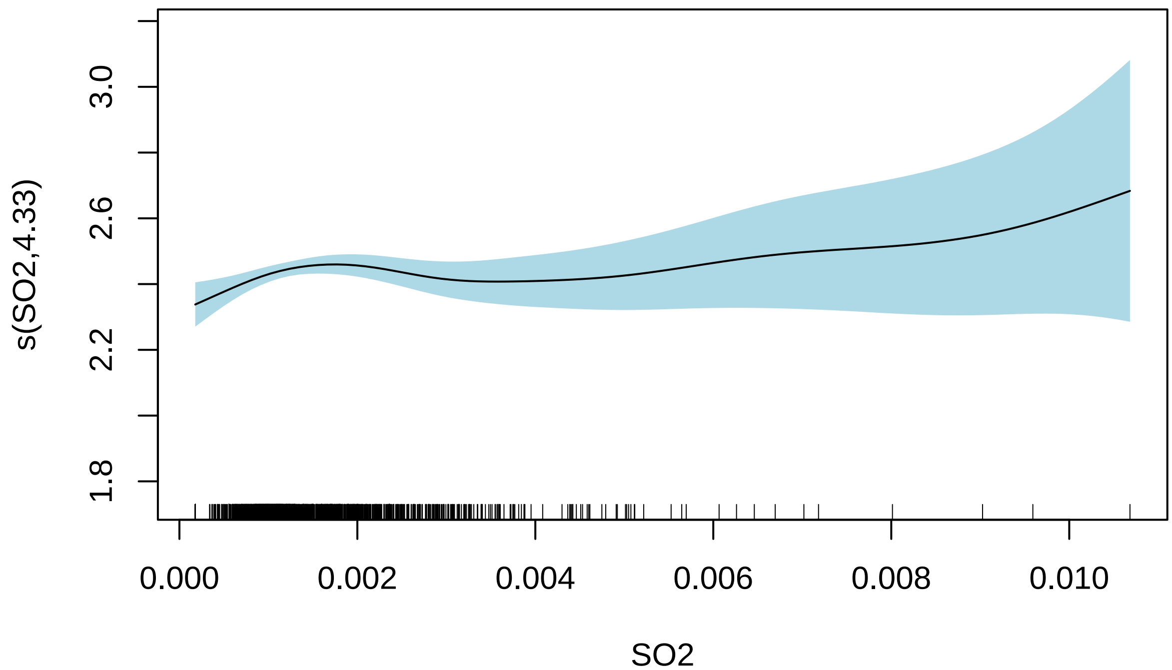
s(PM10,1)
3.0
2.6
2.2
1.8
20
40
60
80
100
120
PM10

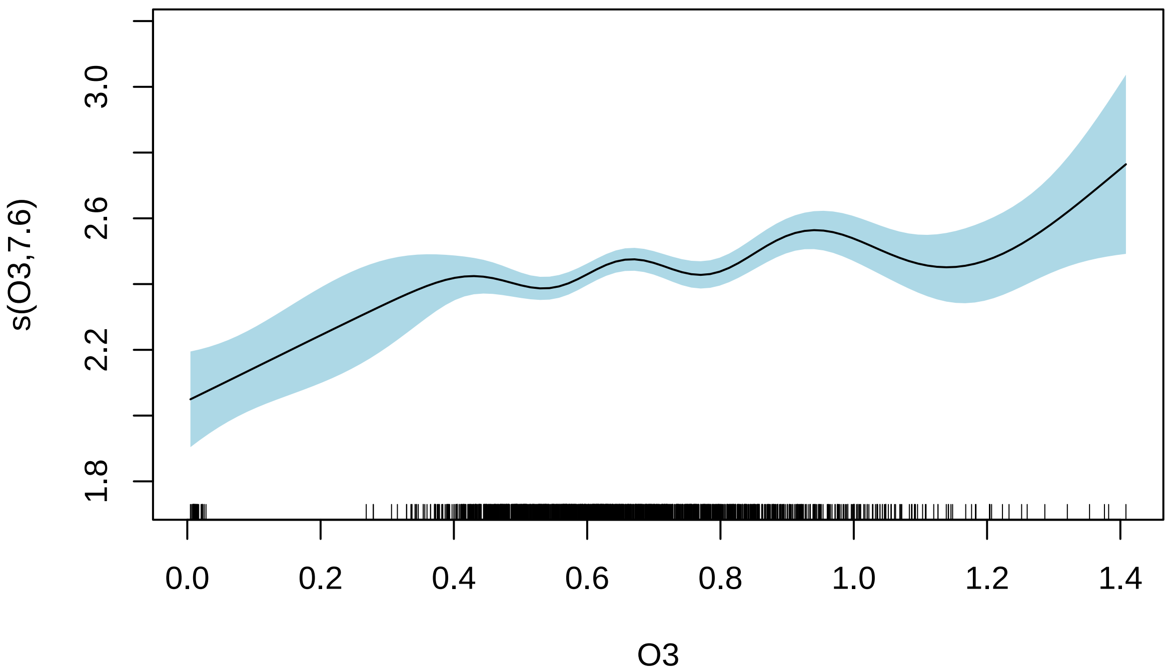


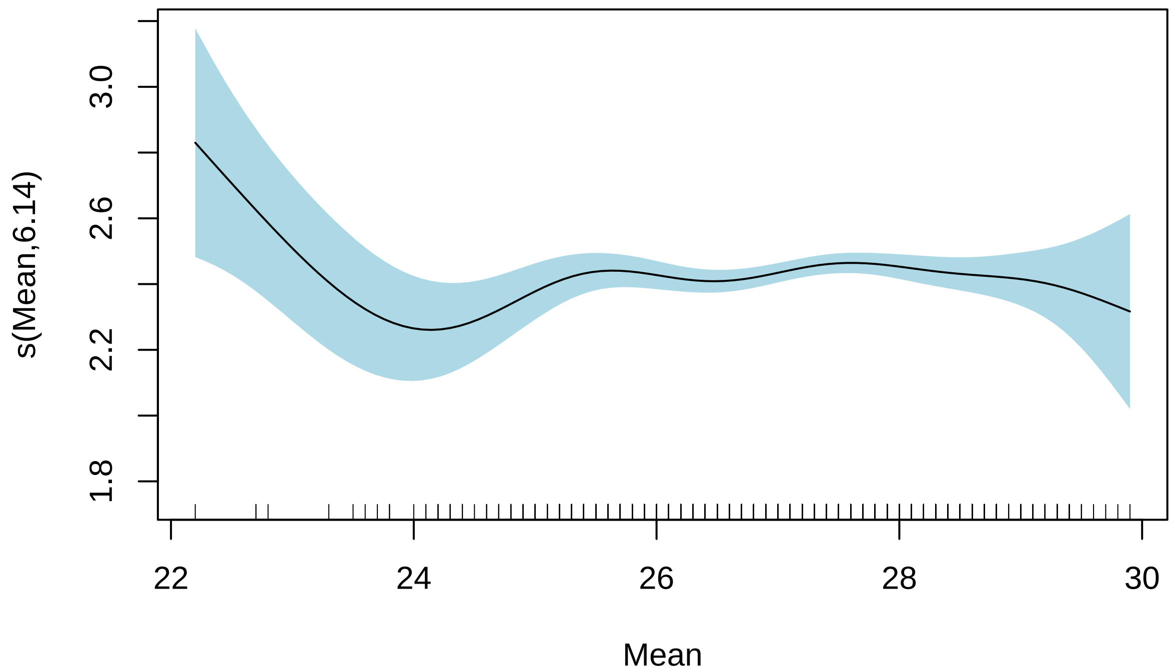
s(NO2,2.2)
1.8
2.2
2.6
3.0
0.000
0.005
0.010
0.015
0.020
0.025
0.030
0.035
NO2

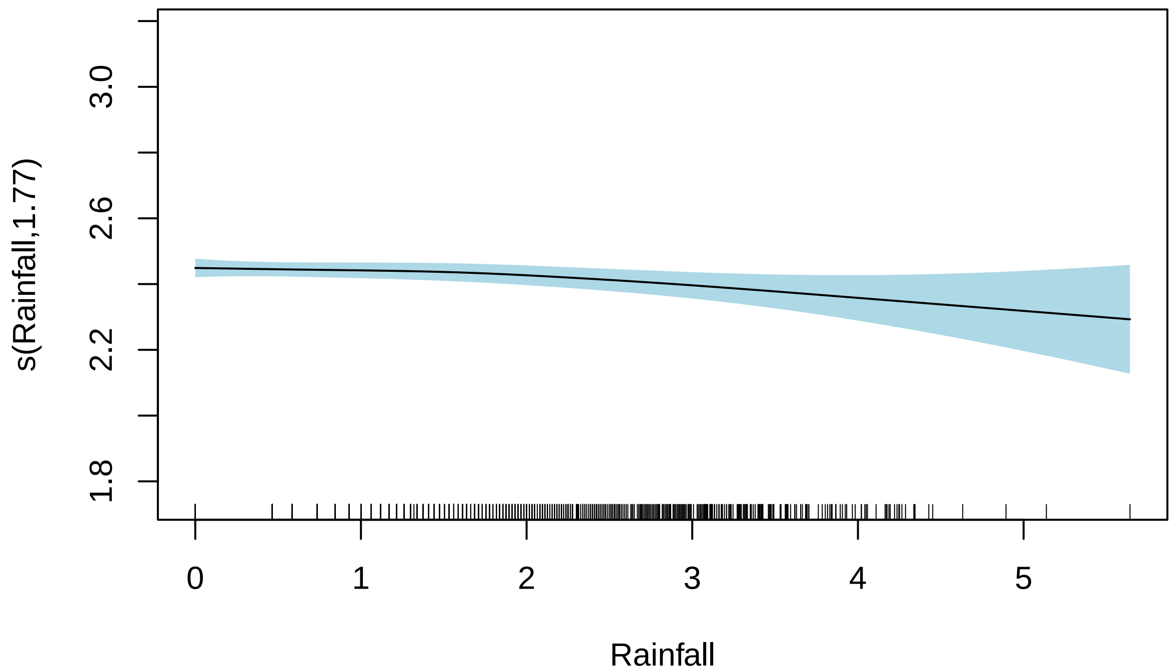


s(CO3,1)
3.0
2.6
2.2
1.8
0
200
400
600
800
CO3

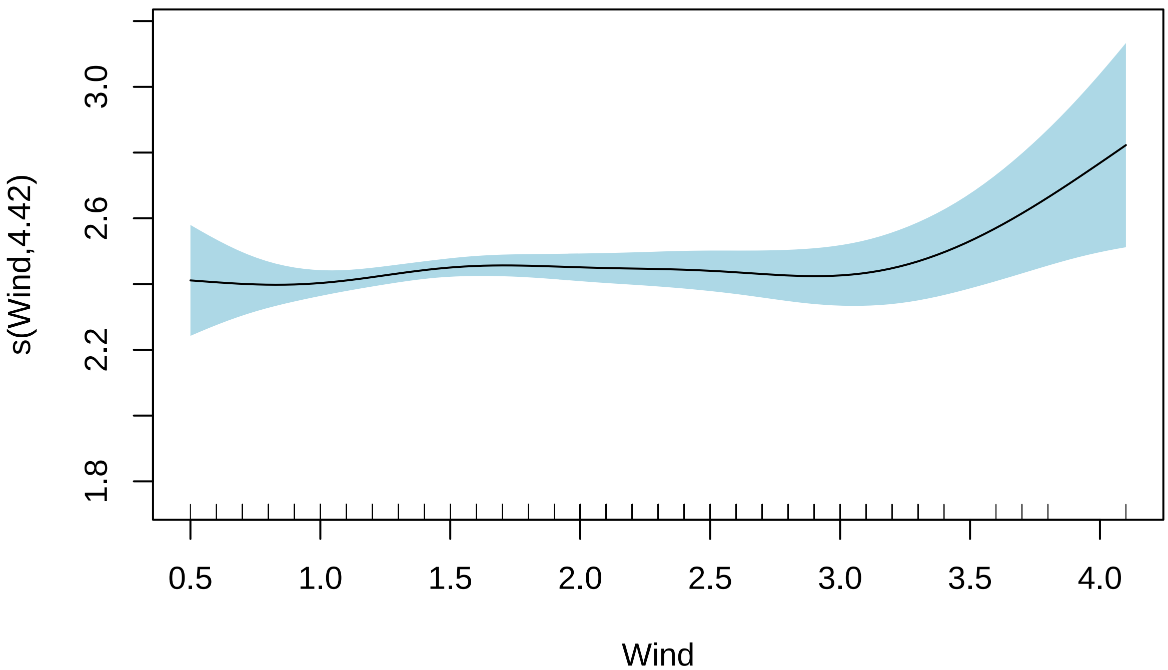

s(SO2,4.33)
1.8
2.2
2.6
3.0
0.000
0.002
0.004
0.006
0.008
0.010
SO2


s(O3,7.6)
1.8
2.2
2.6
3.0
0.0
0.2
0.4
0.6
0.8
1.0
1.2
1.4
O3

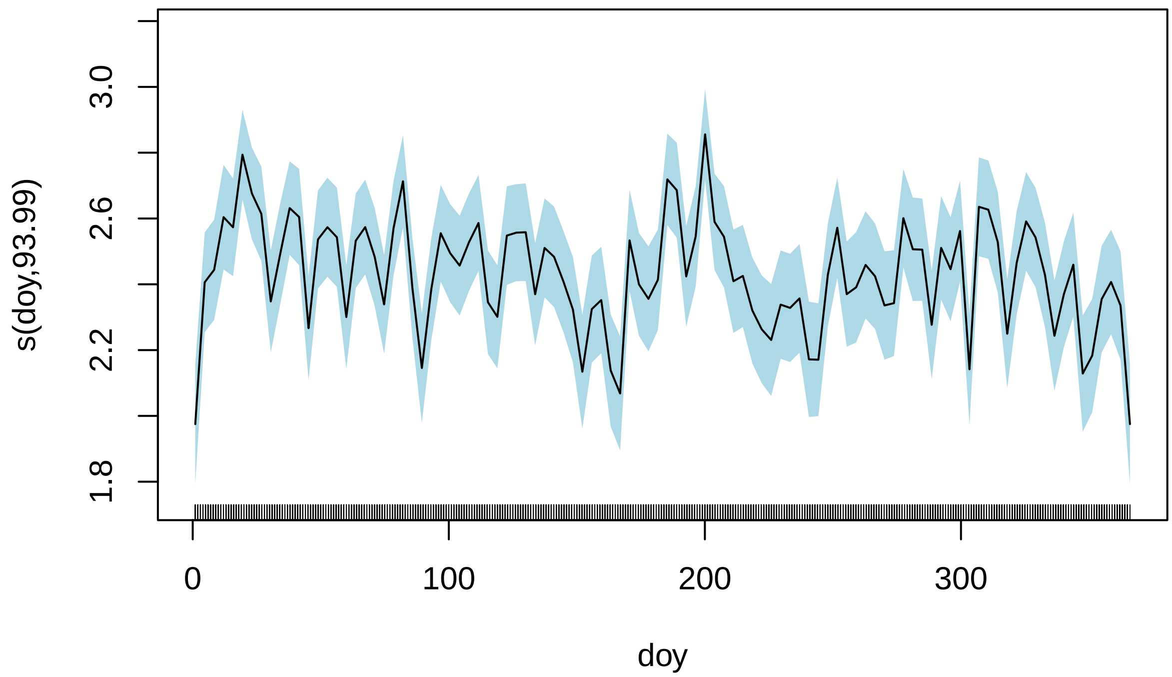

s(Mean,6.14)
1.8
2.2
2.6
3.0
22
24
26
28
30
Mean

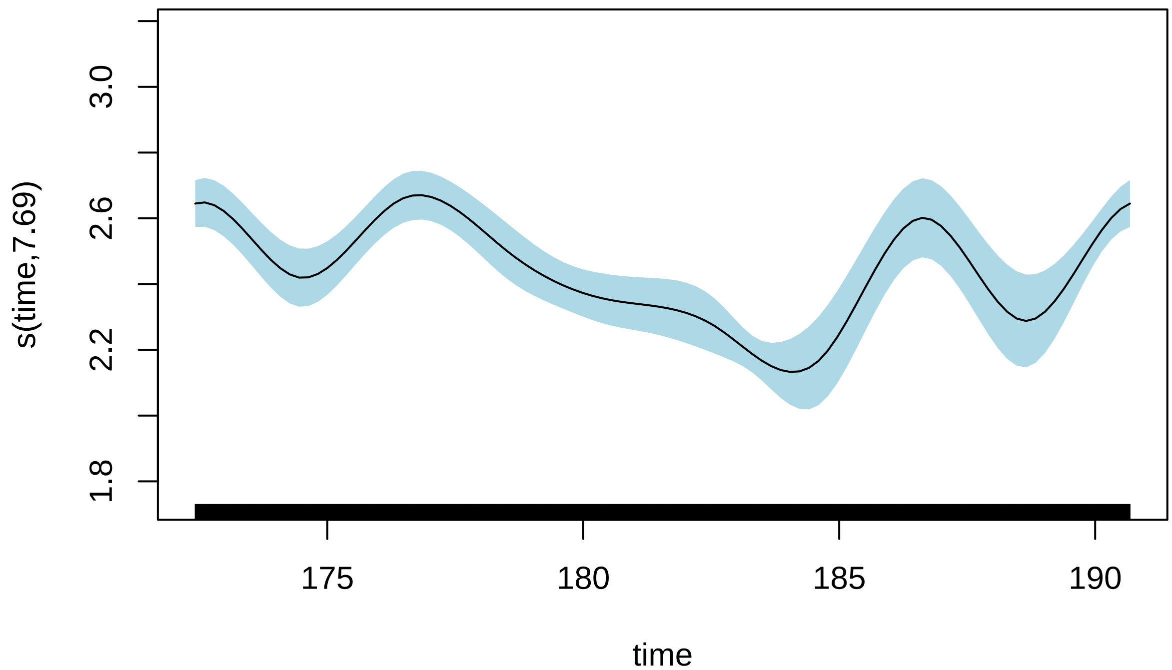


s(Rainfall,1.77)
1.8
2.2
2.6
3.0
0
1
2
3
4
5
Rainfall

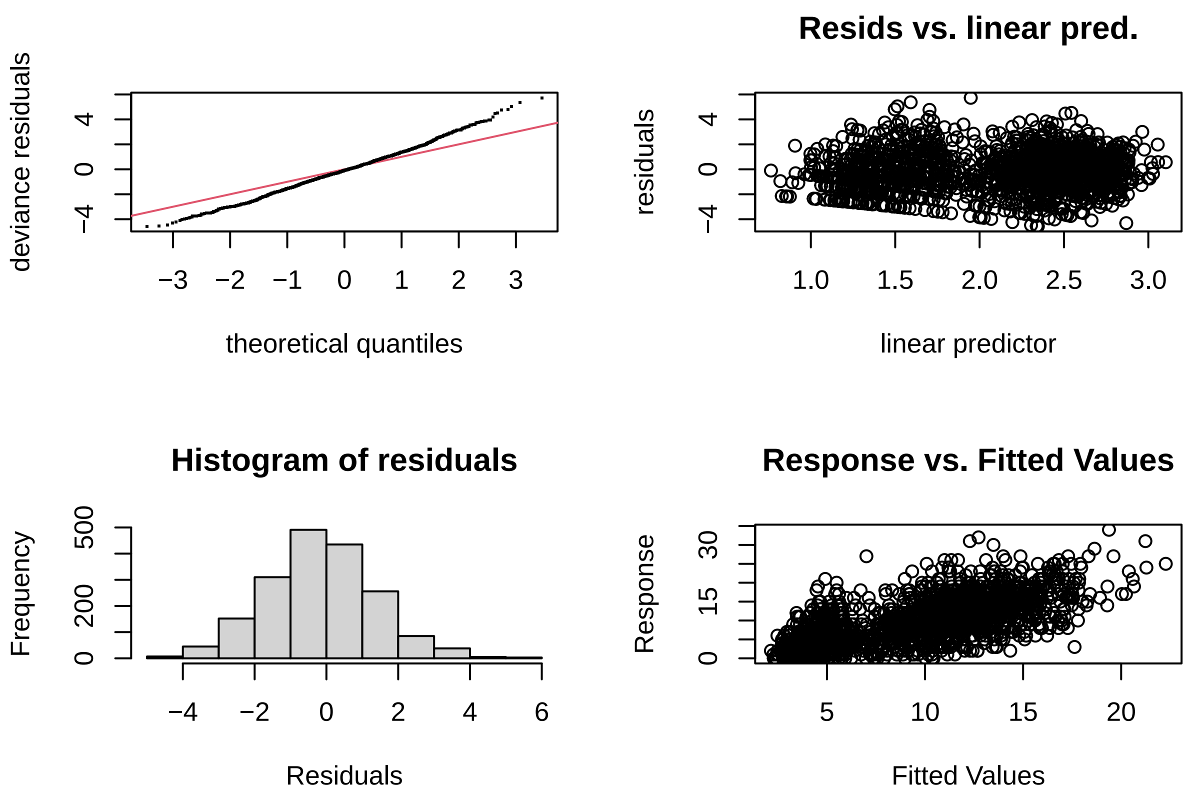


s(Wind,4.42)
1.8
2.2
2.6
3.0
0.5
1.0
1.5
2.0
2.5
3.0
3.5
4.0
Wind


s(doy,93.99)
1.8
2.2
2.6
3.0
0
100
200
300
doy




```
gam.check(jb_mod)
```




```
##
## Method: REML   Optimizer: outer newton
## full convergence after 11 iterations.
## Gradient range [-0.001485521,0.0001904878]
## (score 5866.989 & scale 1).
## Hessian positive definite, eigenvalue range [0.0001187714,3.139576].
## Model rank =  446 / 446
##
## Basis dimension (k) checking results. Low p-value (k-index<1) may
## indicate that k is too low, especially if edf is close to k'.
##
##                 k'    edf k-index p-value
## s(PM10)       9.00   1.00    0.98    0.46
## s(NO2)        9.00   2.20    0.96    0.09 .
## s(CO3)        9.00   1.00    0.98    0.35
## s(SO2)        9.00   4.33    0.87  <2e-16 ***
## s(O3)         9.00   7.60    0.98    0.38
## s(Mean)       9.00   6.14    0.88  <2e-16 ***
## s(Rainfall)   9.00   1.77    0.90  <2e-16 ***
## s(Wind)       9.00   4.42    0.81  <2e-16 ***
## s(doy)      363.00  93.99    1.04    0.98
## s(time)       8.00   7.69    0.77  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
concurvity(jb_mod, full = FALSE)
```

```
## $worst
##                     para      s(PM10)       s(NO2)       s(CO3)       s(SO2)
## para        1.000000e+00 3.450031e-24 2.060784e-20 2.826350e-25 6.462133e-24
## s(PM10)     3.442633e-24 1.000000e+00 2.323230e-01 7.572292e-02 3.090973e-01
## s(NO2)      2.060826e-20 2.323230e-01 1.000000e+00 1.436481e-01 2.453469e-01
## s(CO3)      2.821572e-25 7.572292e-02 1.436481e-01 1.000000e+00 1.997584e-02
## s(SO2)      6.467740e-24 3.090973e-01 2.453469e-01 1.997584e-02 1.000000e+00
## s(O3)       4.039737e-25 2.541089e-01 3.229570e-01 5.681647e-01 1.189206e-01
## s(Mean)     2.135953e-25 9.555324e-02 3.308287e-02 1.302181e-01 2.367738e-02
## s(Rainfall) 2.950959e-22 5.343340e-02 5.391818e-02 6.851237e-02 1.308210e-02
## s(Wind)     1.277608e-18 3.202624e-02 2.261062e-01 1.044072e-01 2.698660e-02
## s(doy)      6.493781e-28 3.233252e-01 4.503458e-01 2.933648e-01 3.759691e-01
## s(time)     1.259851e-30 2.003310e-01 1.733289e-01 1.017421e-01 3.746884e-01
##                    s(O3)      s(Mean)  s(Rainfall)      s(Wind)       s(doy)
## para        4.049660e-25 2.132835e-25 2.950919e-22 1.277616e-18 6.466954e-28
## s(PM10)     2.541089e-01 9.555324e-02 5.343340e-02 3.202624e-02 3.233252e-01
## s(NO2)      3.229570e-01 3.308287e-02 5.391818e-02 2.261062e-01 4.503458e-01
## s(CO3)      5.681647e-01 1.302181e-01 6.851237e-02 1.044072e-01 2.933648e-01
## s(SO2)      1.189206e-01 2.367738e-02 1.308210e-02 2.698660e-02 3.759691e-01
## s(O3)       1.000000e+00 3.760535e-02 3.375082e-02 9.369350e-02 2.720383e-01
## s(Mean)     3.760535e-02 1.000000e+00 3.073125e-01 1.558303e-01 2.989454e-01
## s(Rainfall) 3.375082e-02 3.073125e-01 1.000000e+00 2.272949e-01 2.719916e-01
## s(Wind)     9.369350e-02 1.558303e-01 2.272949e-01 1.000000e+00 4.894219e-01
## s(doy)      2.720383e-01 2.989454e-01 2.719916e-01 4.894219e-01 1.000000e+00
## s(time)     2.463409e-01 5.044443e-02 2.210754e-02 9.749994e-02 1.462838e-01
##                  s(time)
## para        6.391199e-31
## s(PM10)     2.003310e-01
## s(NO2)      1.733289e-01
## s(CO3)      1.017421e-01
## s(SO2)      3.746884e-01
## s(O3)       2.463409e-01
## s(Mean)     5.044443e-02
## s(Rainfall) 2.210754e-02
## s(Wind)     9.749994e-02
## s(doy)      1.462838e-01
## s(time)     1.000000e+00
##
## $observed
##                     para      s(PM10)       s(NO2)       s(CO3)       s(SO2)
## para        1.000000e+00 4.496525e-32 1.523345e-24 5.996381e-34 1.120803e-26
## s(PM10)     3.442633e-24 1.000000e+00 4.776485e-02 2.600963e-02 3.826646e-02
## s(NO2)      2.060826e-20 1.662660e-01 1.000000e+00 8.763652e-02 9.153970e-02
## s(CO3)      2.821572e-25 5.933614e-02 1.554627e-02 1.000000e+00 6.030322e-03
## s(SO2)      6.467740e-24 2.664218e-01 9.166038e-02 1.201779e-02 1.000000e+00
## s(O3)       4.039737e-25 2.077639e-01 1.686083e-01 4.623643e-02 1.073559e-02
## s(Mean)     2.135953e-25 3.030996e-02 1.021709e-02 1.255875e-01 1.638039e-02
## s(Rainfall) 2.950959e-22 1.007098e-02 8.497415e-03 4.552233e-02 3.209659e-03
## s(Wind)     1.277608e-18 1.157947e-02 2.573252e-02 6.306780e-02 1.084000e-02
## s(doy)      6.493781e-28 3.026655e-01 2.219902e-01 2.550651e-01 2.331197e-01
## s(time)     1.259851e-30 1.699133e-01 1.138008e-01 4.991878e-02 1.823568e-01
```

```
##                    s(O3)      s(Mean)  s(Rainfall)      s(Wind)       s(doy)
## para         9.927024e-28 1.019937e-27 1.753003e-27 2.743355e-22 3.700943e-32
## s(PM10)      1.255916e-01 2.248874e-02 1.683496e-02 1.095532e-02 5.044128e-03
## s(NO2)       9.041161e-02 4.867237e-03 2.015469e-02 7.171471e-02 8.720515e-03
## s(CO3)       3.441704e-01 5.511643e-02 2.833638e-02 4.923475e-02 8.568377e-03
## s(SO2)       3.822313e-02 3.122978e-03 3.005218e-03 8.085279e-03 6.293875e-03
## s(O3)        1.000000e+00 8.931334e-03 1.499442e-02 4.501829e-02 2.178657e-02
## s(Mean)      8.878770e-03 1.000000e+00 2.105386e-01 1.051139e-01 1.353931e-02
## s(Rainfall) 1.140070e-02 6.745198e-02 1.000000e+00 1.104111e-01 2.655718e-02
## s(Wind)      4.513979e-02 2.753456e-02 1.029650e-01 1.000000e+00 2.420349e-02
## s(doy)       2.262309e-01 2.009982e-01 2.361679e-01 2.967371e-01 1.000000e+00
## s(time)      1.741486e-01 1.206061e-02 6.769564e-03 3.775474e-02 3.254328e-03
##                   s(time)
## para         2.027103e-31
## s(PM10)      3.847965e-02
## s(NO2)       6.331317e-02
## s(CO3)       2.741304e-02
## s(SO2)       1.233717e-01
## s(O3)        1.149092e-01
## s(Mean)      2.763108e-02
## s(Rainfall) 5.955207e-03
## s(Wind)      3.272758e-02
## s(doy)       5.321300e-02
## s(time)      1.000000e+00
##
## $estimate
##                     para      s(PM10)       s(NO2)       s(CO3)       s(SO2)
## para         1.000000e+00 1.762228e-26 9.071461e-23 2.652595e-27 4.845931e-26
## s(PM10)      3.442633e-24 1.000000e+00 1.721951e-01 1.295836e-02 6.792475e-02
## s(NO2)       2.060826e-20 8.142494e-02 1.000000e+00 3.955205e-02 9.107830e-02
## s(CO3)       2.821572e-25 3.361443e-02 8.377059e-02 1.000000e+00 5.771193e-03
## s(SO2)       6.467740e-24 1.171041e-01 1.795731e-01 5.874695e-03 1.000000e+00
## s(O3)        4.039737e-25 1.016485e-01 2.580949e-01 3.183726e-02 2.940580e-02
## s(Mean)      2.135953e-25 1.817446e-02 1.294369e-02 9.250701e-02 5.470295e-03
## s(Rainfall) 2.950959e-22 6.169549e-03 2.650199e-02 2.012041e-02 3.754740e-03
## s(Wind)      1.277608e-18 1.136391e-02 1.169304e-01 2.670894e-02 8.660231e-03
## s(doy)       6.493781e-28 2.384690e-01 3.110395e-01 2.238959e-01 2.230615e-01
## s(time)      1.259851e-30 8.918370e-02 6.994732e-02 2.514871e-02 8.869453e-02
##                    s(O3)      s(Mean)  s(Rainfall)      s(Wind)       s(doy)
## para         1.945459e-27 1.299755e-27 2.309760e-25 7.535986e-21 1.992387e-30
## s(PM10)      1.449432e-01 4.565628e-02 1.104208e-02 1.946312e-02 5.371900e-03
## s(NO2)       1.852617e-01 1.861272e-02 3.343007e-02 1.604126e-01 5.897924e-03
## s(CO3)       1.670189e-01 6.468995e-02 4.424769e-02 7.596504e-02 4.924529e-03
## s(SO2)       5.775393e-02 8.436315e-03 2.386422e-03 1.638205e-02 4.878197e-03
## s(O3)        1.000000e+00 2.038454e-02 1.895486e-02 6.177008e-02 5.180828e-03
## s(Mean)      1.464796e-02 1.000000e+00 2.587716e-01 7.702800e-02 5.462395e-03
## s(Rainfall) 1.357997e-02 2.176717e-01 1.000000e+00 1.508776e-01 5.246720e-03
## s(Wind)      4.142580e-02 8.622256e-02 1.647288e-01 1.000000e+00 6.469611e-03
## s(doy)       2.185785e-01 2.454250e-01 2.506086e-01 4.031088e-01 1.000000e+00
## s(time)      1.483876e-01 3.500013e-02 1.059994e-02 7.077890e-02 9.144150e-04
##                   s(time)
## para         8.037831e-32
## s(PM10)      4.215765e-02
## s(NO2)       3.777826e-02
```

```
## s(CO3)      2.580696e-02
## s(SO2)      1.005769e-01
## s(O3)       6.318855e-02
## s(Mean)     1.233759e-02
## s(Rainfall) 8.746026e-03
## s(Wind)     1.971746e-02
## s(doy)      2.421660e-02
## s(time)     1.000000e+00
```

For JB data without PM 2.5, the concurvity analysis shows no pairings of variables with high concurvity, including ozone and carbon monoxide. Therefore, both these variables will be included in the final model for comparison between locations.

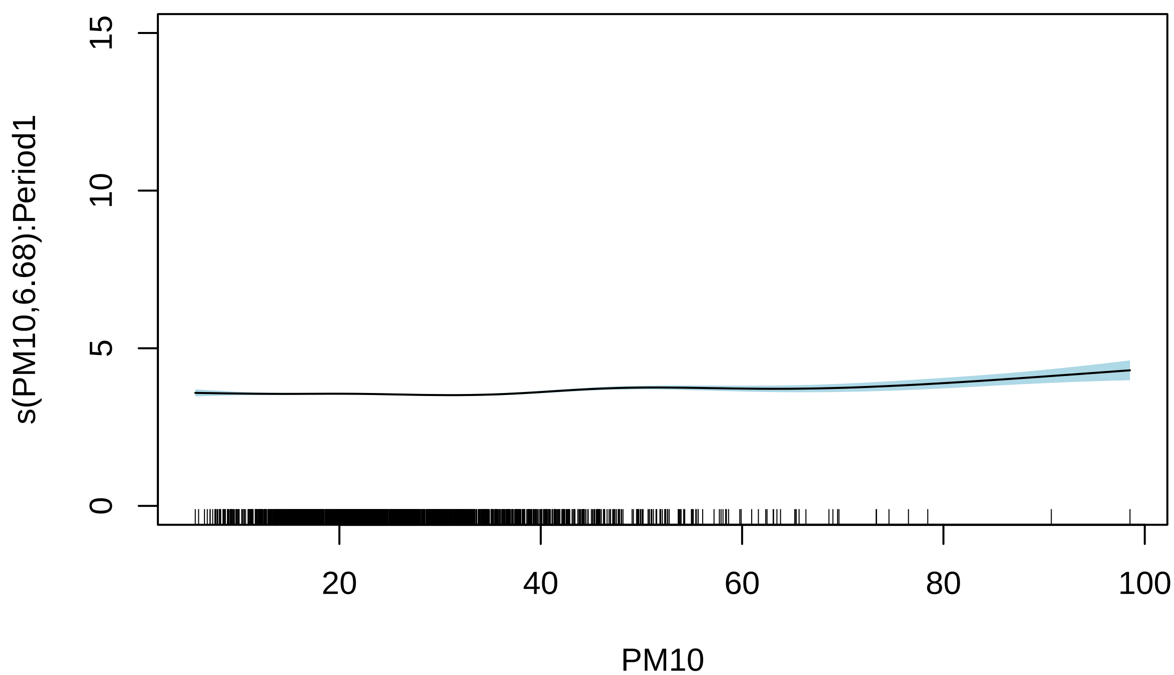
Checking for 'Period' as an interaction term

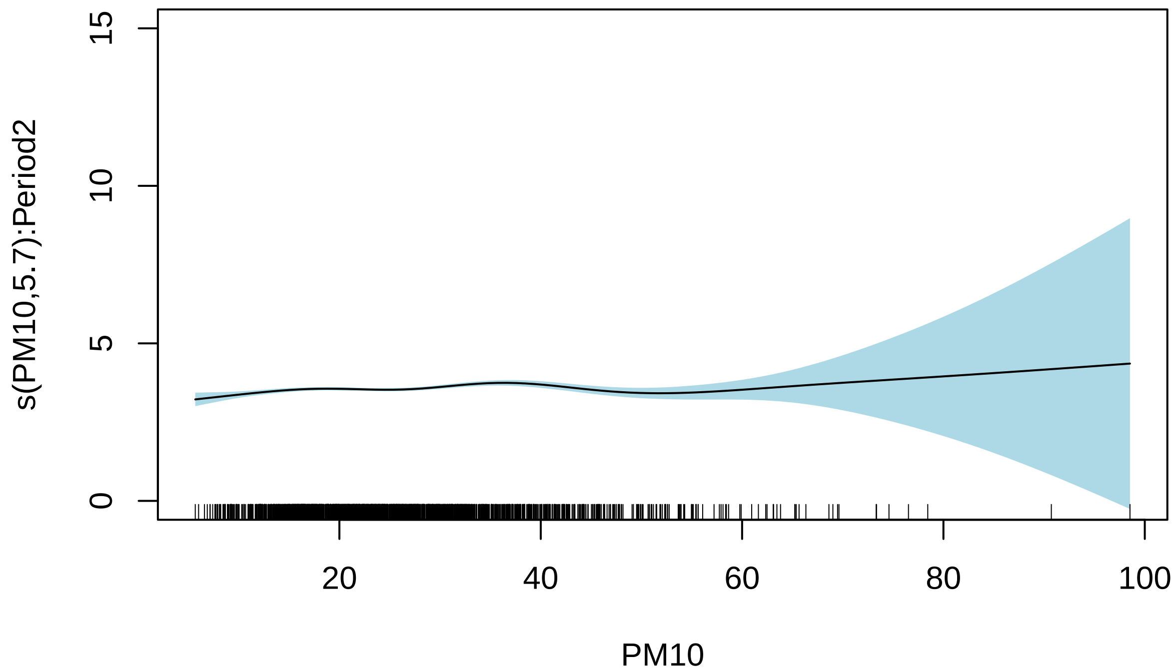
```
kb_mod_period <- gam(ED ~ s(PM10, by=Period) + s(NO2, by=Period) + s(CO3, by=Period) + s(SO2, by=Period

summary(kb_mod_period)
```

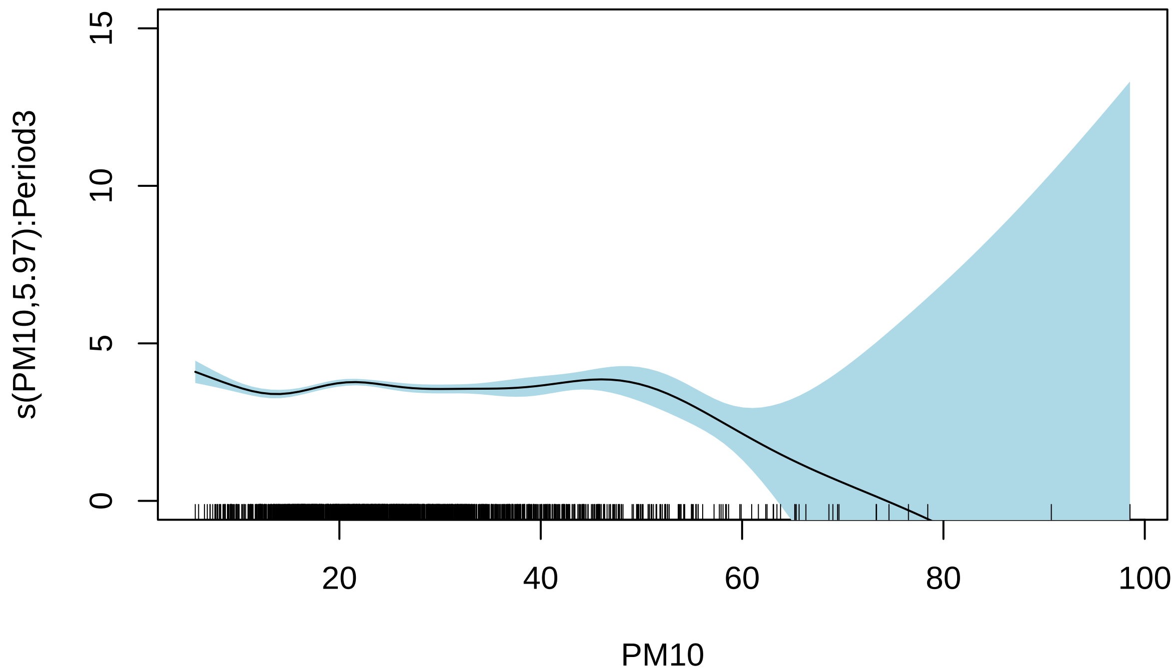
```
##
## Family: poisson
## Link function: log
##
## Formula:
## ED ~ s(PM10, by = Period) + s(NO2, by = Period) + s(CO3, by = Period) +
##     s(SO2, by = Period) + s(O3, by = Period) + s(Mean) + s(Rainfall) +
##     s(Wind) + s(doy, bs = "cc", k = 365) + s(time, bs = "bs") +
##     Period
##
## Parametric coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  3.56270    0.03174  112.26   <2e-16 ***
## Period2     -1.93964    0.08375  -23.16   <2e-16 ***
## Period3     -2.48879    0.13118  -18.97   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Approximate significance of smooth terms:
##                    edf  Ref.df  Chi.sq  p-value
## s(PM10):Period1   6.679   7.775  77.407  < 2e-16 ***
## s(PM10):Period2   5.701   6.652  31.611 3.08e-05 ***
## s(PM10):Period3   5.971   6.529  34.088 6.39e-06 ***
## s(NO2):Period1    1.001   1.002   0.899 0.343177
## s(NO2):Period2    1.001   1.002   0.022 0.884620
## s(NO2):Period3    3.125   3.451  22.831 8.49e-05 ***
## s(CO3):Period1    2.858   3.628   1.770 0.597975
## s(CO3):Period2    1.001   1.002   0.386 0.535038
## s(CO3):Period3    1.018   1.033   4.661 0.033891 *
## s(SO2):Period1    2.897   3.606  22.106 0.000163 ***
## s(SO2):Period2    7.139   7.811  51.477  < 2e-16 ***
## s(SO2):Period3    1.081   1.153   1.867 0.222994
## s(O3):Period1     8.261   8.814  80.432  < 2e-16 ***
## s(O3):Period2     4.080   5.052  45.330  < 2e-16 ***
## s(O3):Period3     2.684   3.296   8.265 0.050457 .
```

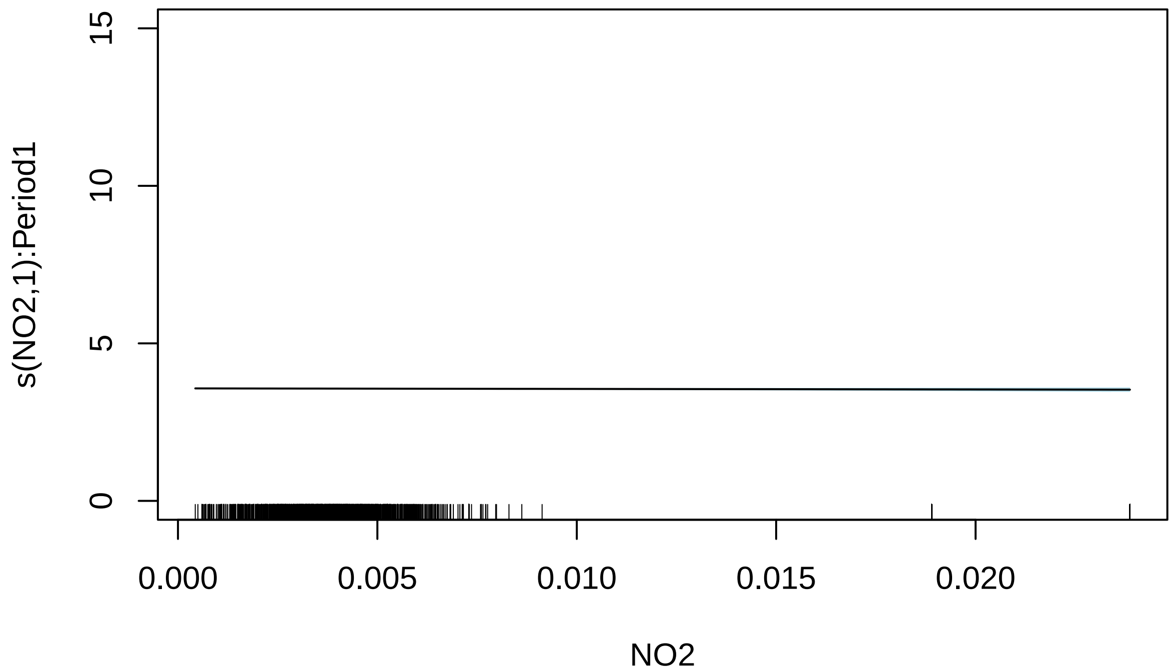
```
## s(Mean)           6.905   7.886  61.014  < 2e-16 ***
## s(Rainfall)       1.356   1.623  12.877 0.000789 ***
## s(Wind)           5.113   6.165  33.955 1.52e-06 ***
## s(doy)          133.400 363.000 985.921  < 2e-16 ***
## s(time)           8.725   8.954 933.653  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-sq.(adj) =  0.636   Deviance explained = 71.1%
## -REML = 7141.8  Scale est. = 1         n = 1827
```

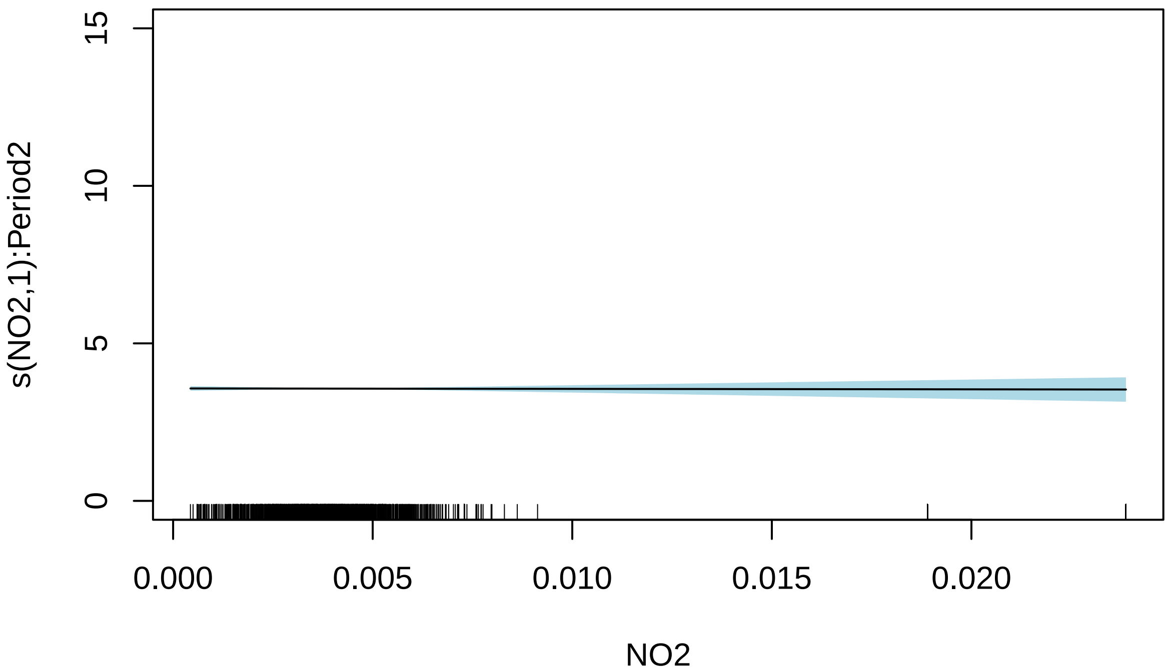
```
plot.gam(kb_mod_period, shade = TRUE, shade.col = "lightblue", shift = coef(kb_mod_period)[1],  seWithM
```



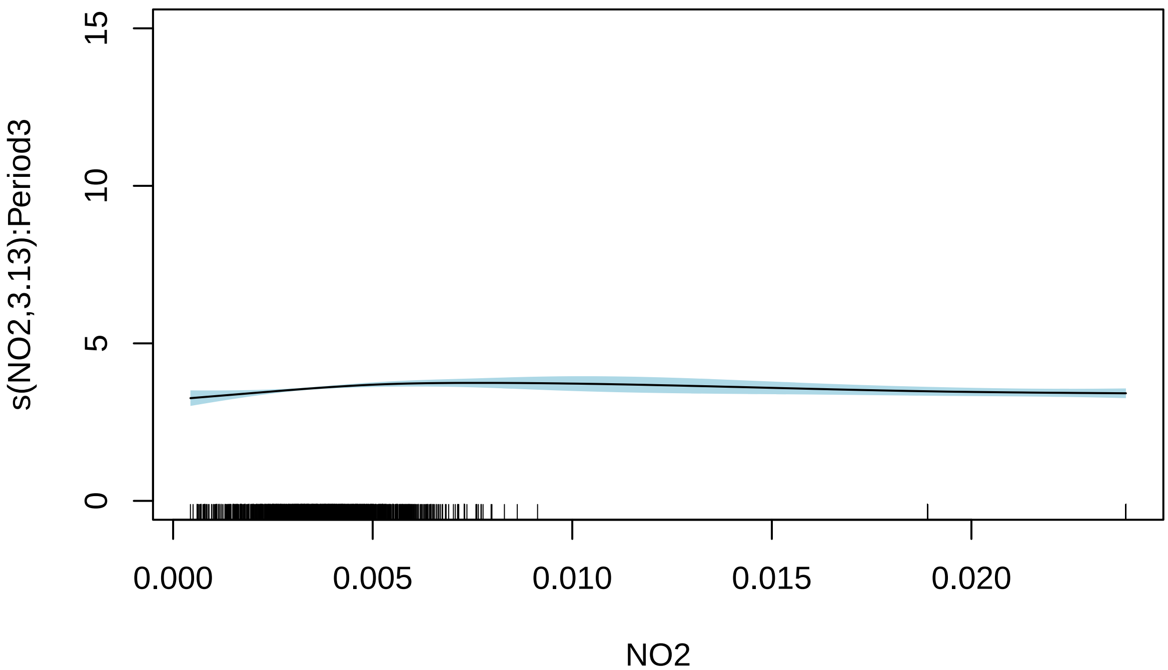

s(PM10,5.7):Period2
0
5
10
15
20
40
60
80
100
PM10

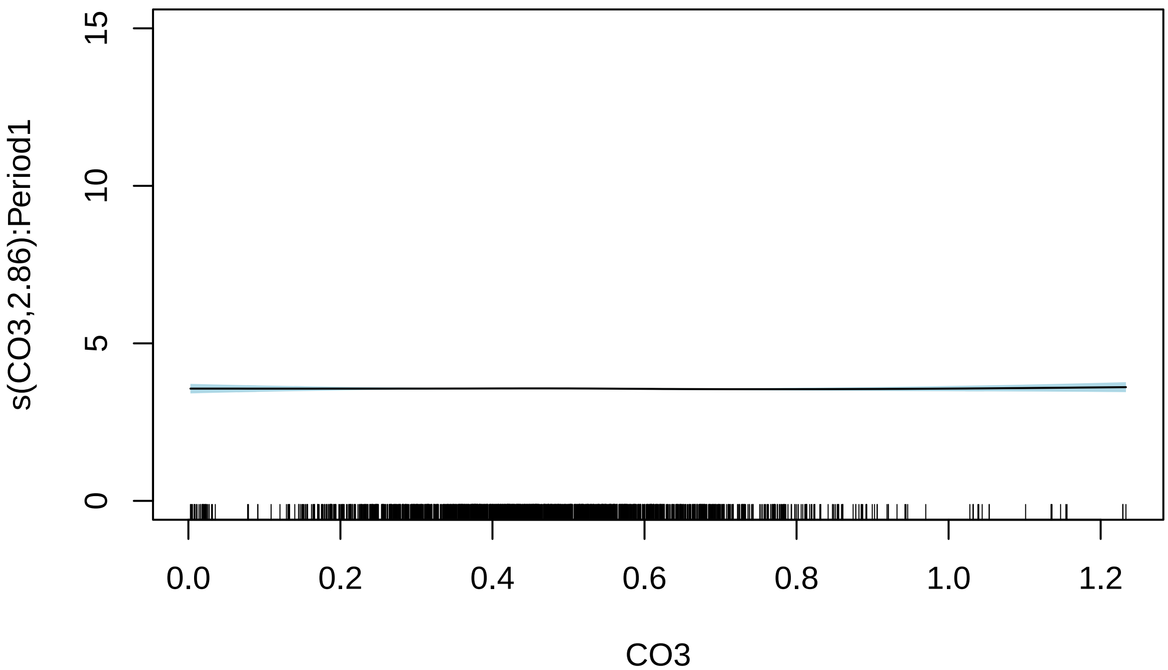

s(PM10,5.97):Period3
0
5
10
15
20
40
60
80
100
PM10

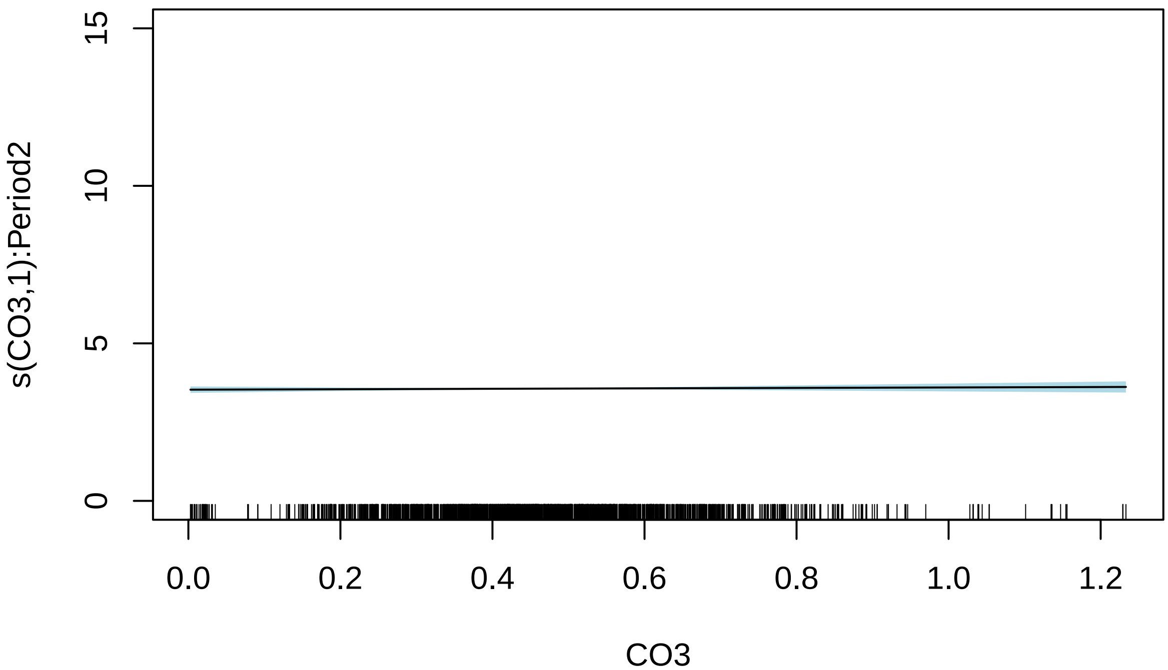

s(NO2,1):Period1
0
5
10
15
0.000
0.005
0.010
0.015
0.020
NO2

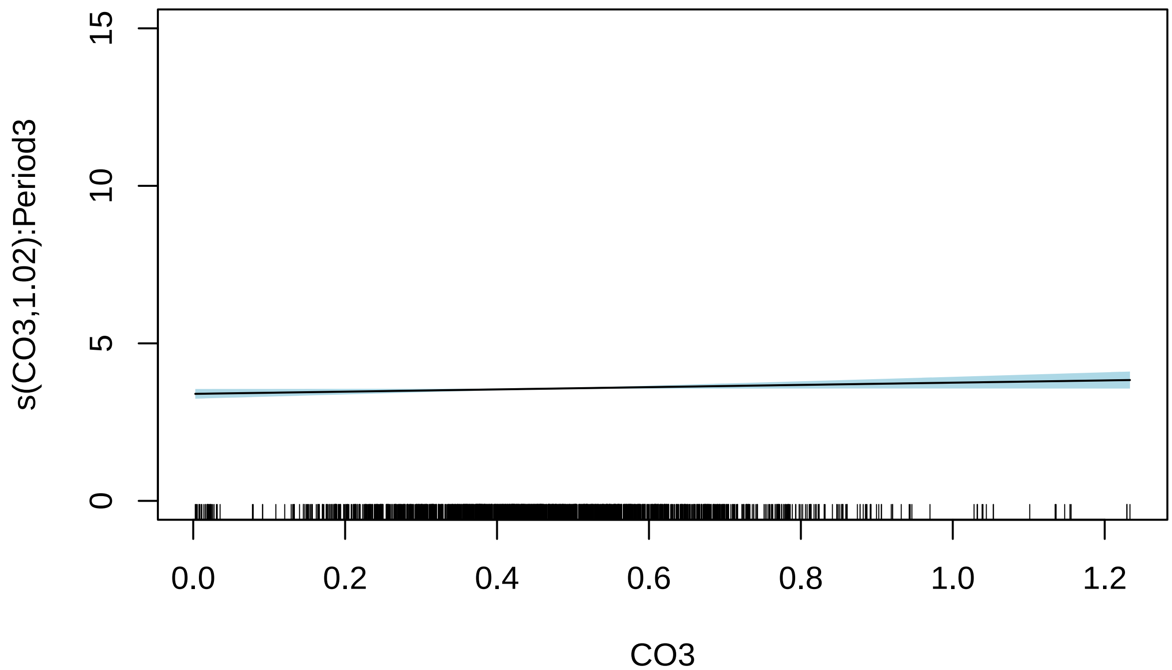


15
10
5
0
s(NO2,1):Period2
0.000
0.005
0.010
0.015
0.020
NO2

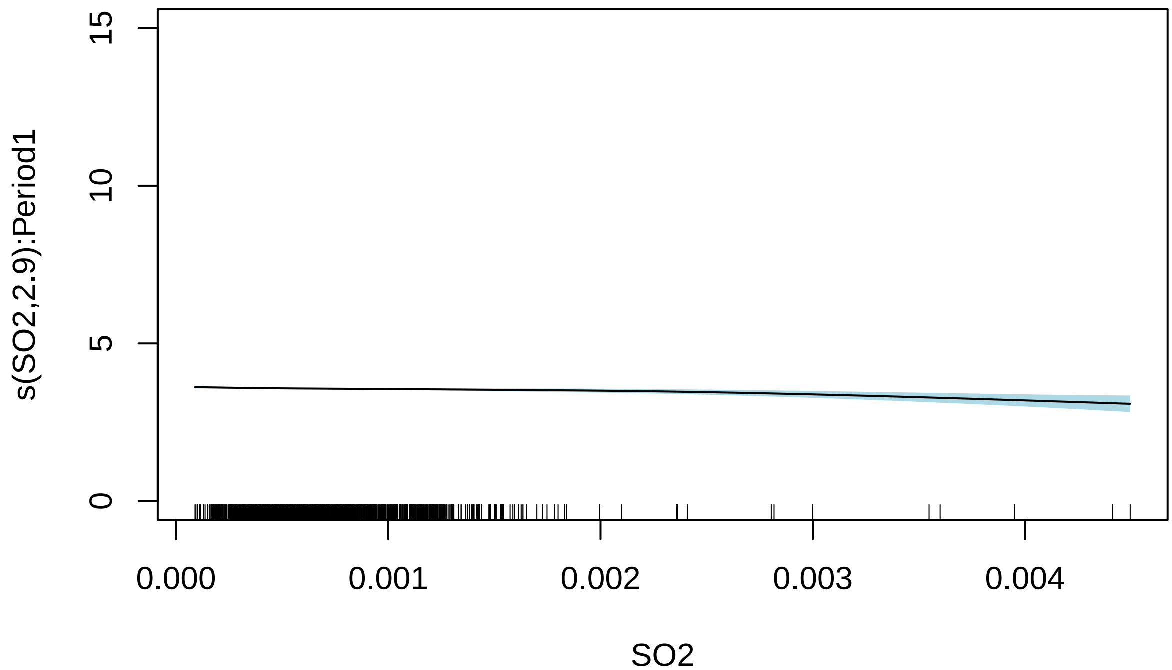

s(NO2,3.13):Period3
0
5
10
15
0.000
0.005
0.010
0.015
0.020
NO2


15
10
5
0
s(CO3,2.86):Period1
0.0
0.2
0.4
0.6
0.8
1.0
1.2
CO3

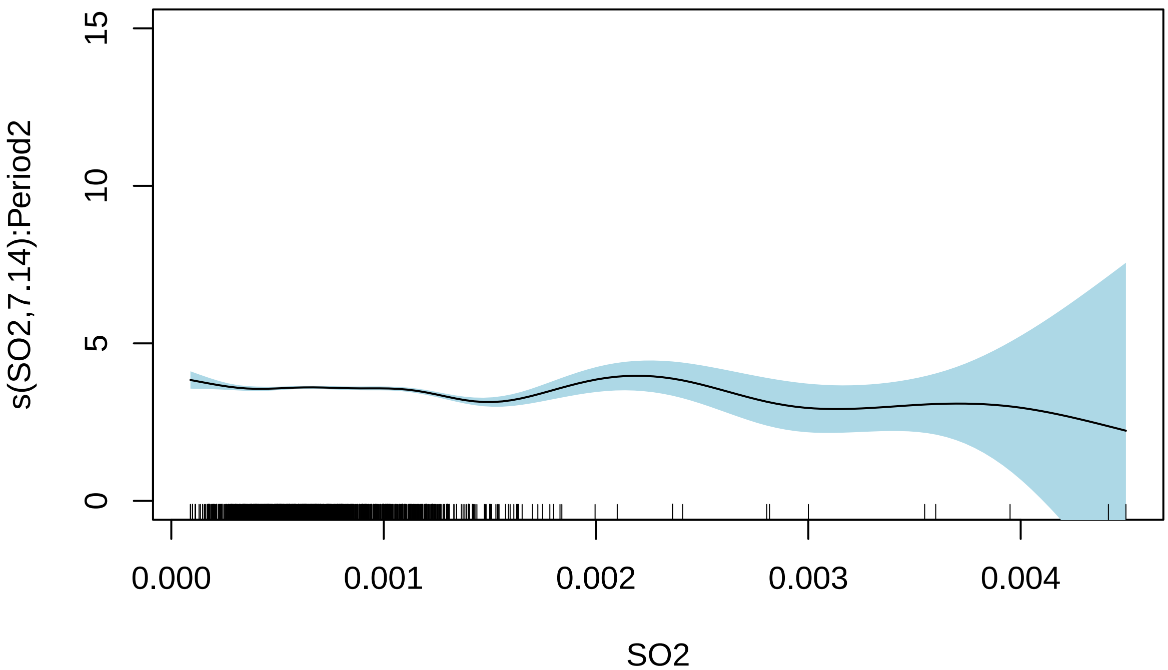

s(CO3,1):Period2
0
5
10
15
0.0
0.2
0.4
0.6
0.8
1.0
1.2
CO3



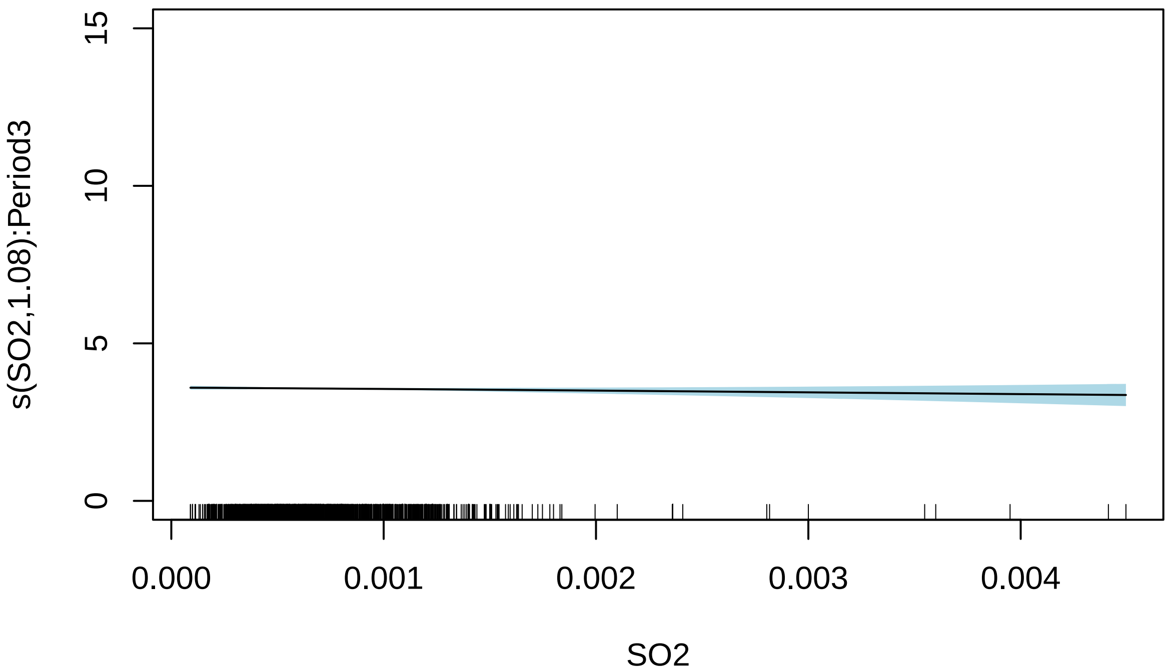
s(CO3,1.02):Period3
0
5
10
15
0.0
0.2
0.4
0.6
0.8
1.0
1.2
CO3

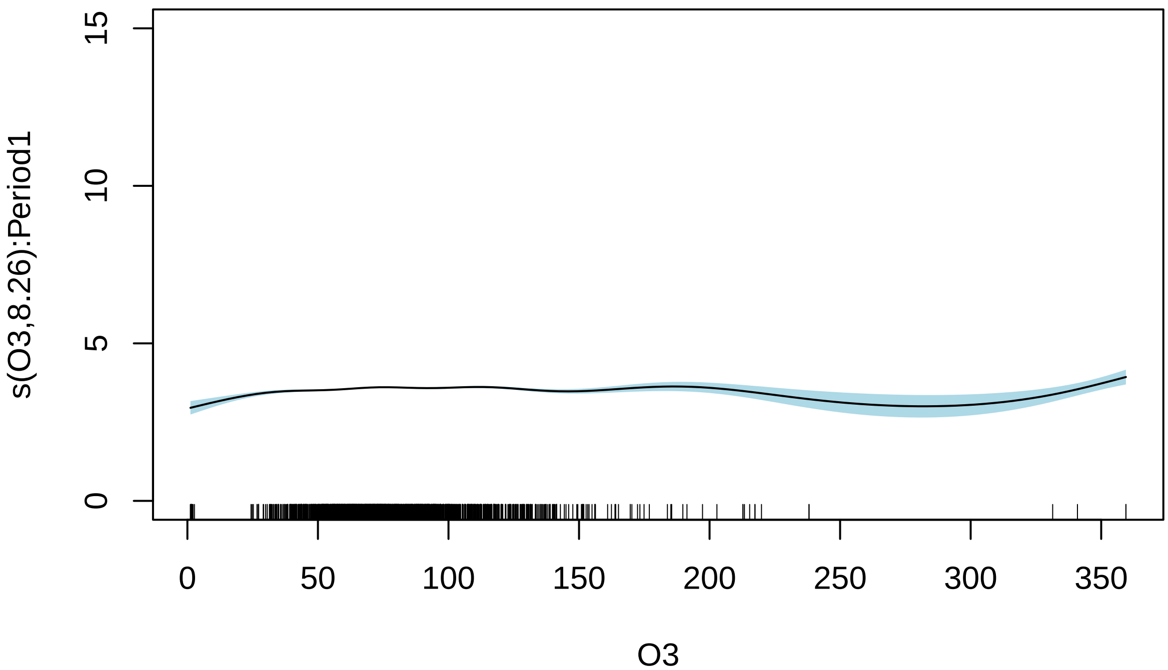


s(SO2,2.9):Period1
0
5
10
15
0.000
0.001
0.002
0.003
0.004
SO2

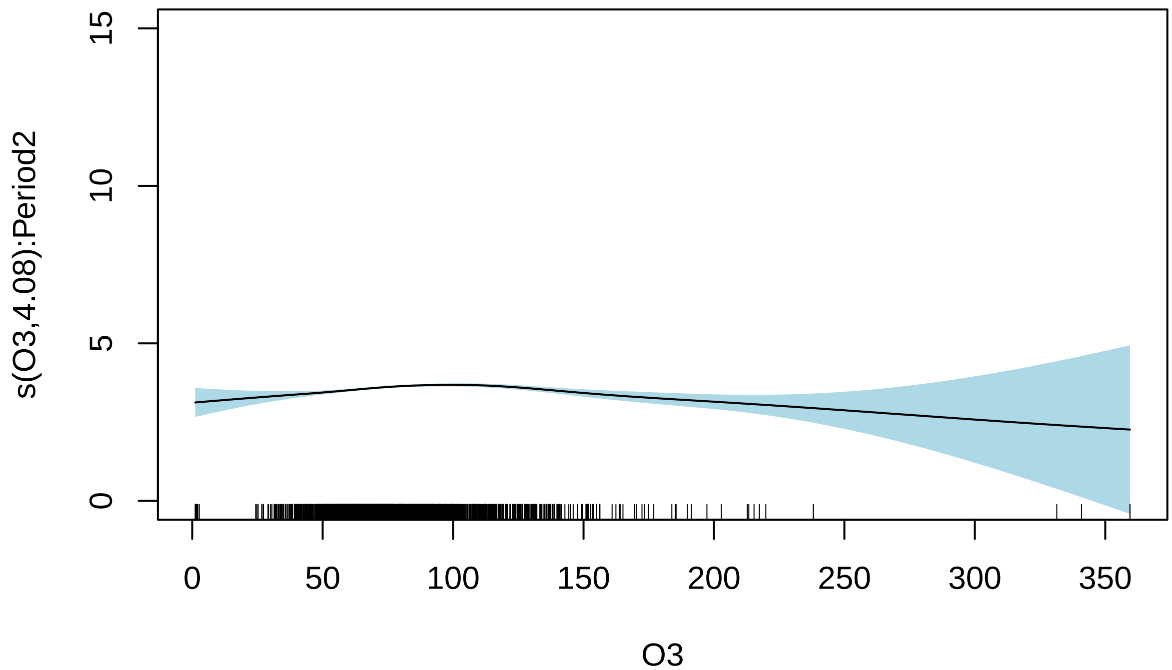


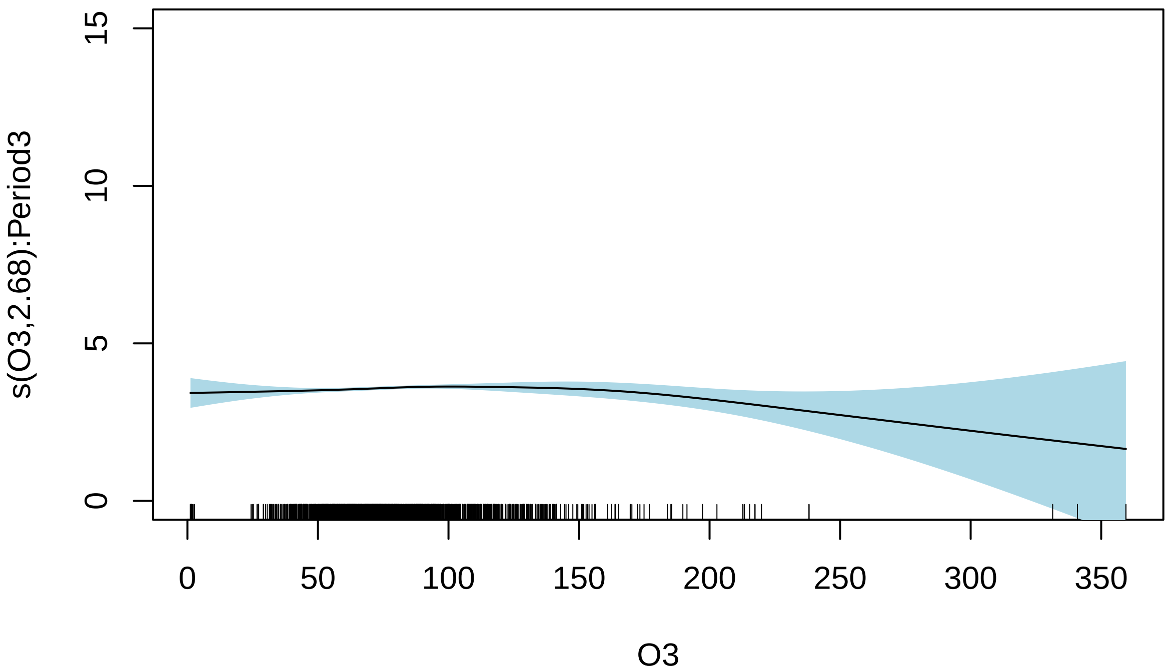
15
10
5
0
s(SO2,7.14):Period2
0.000
0.001
0.002
0.003
0.004
SO2

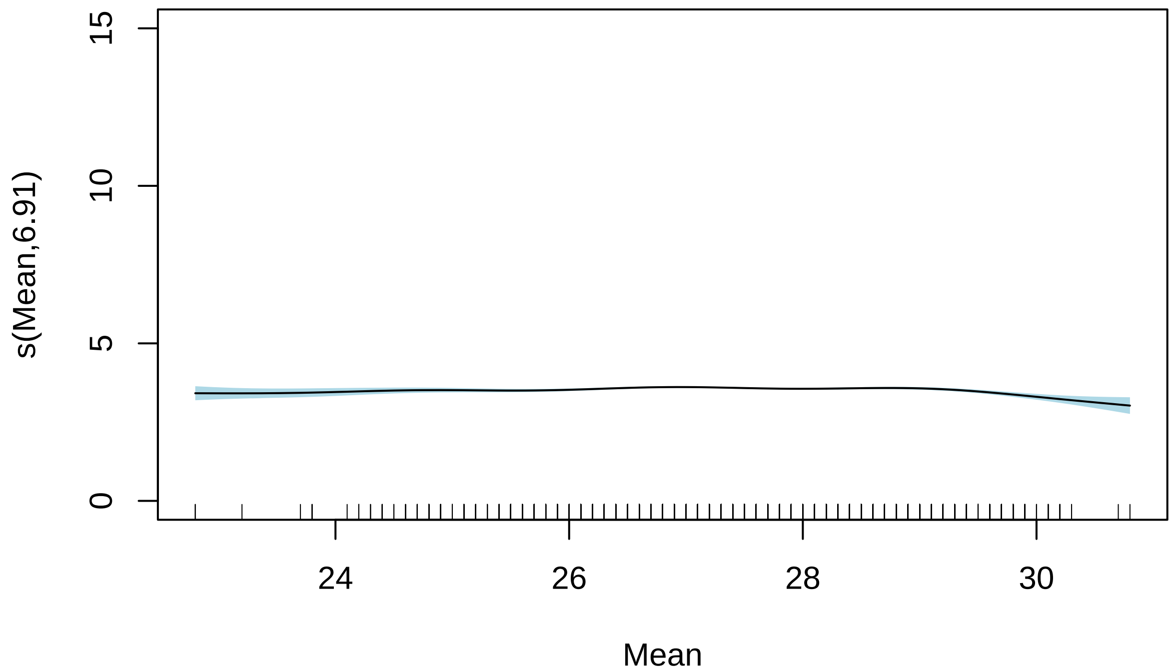

s(SO2,1.08):Period3
0
5
10
15
0.000
0.001
0.002
0.003
0.004
SO2

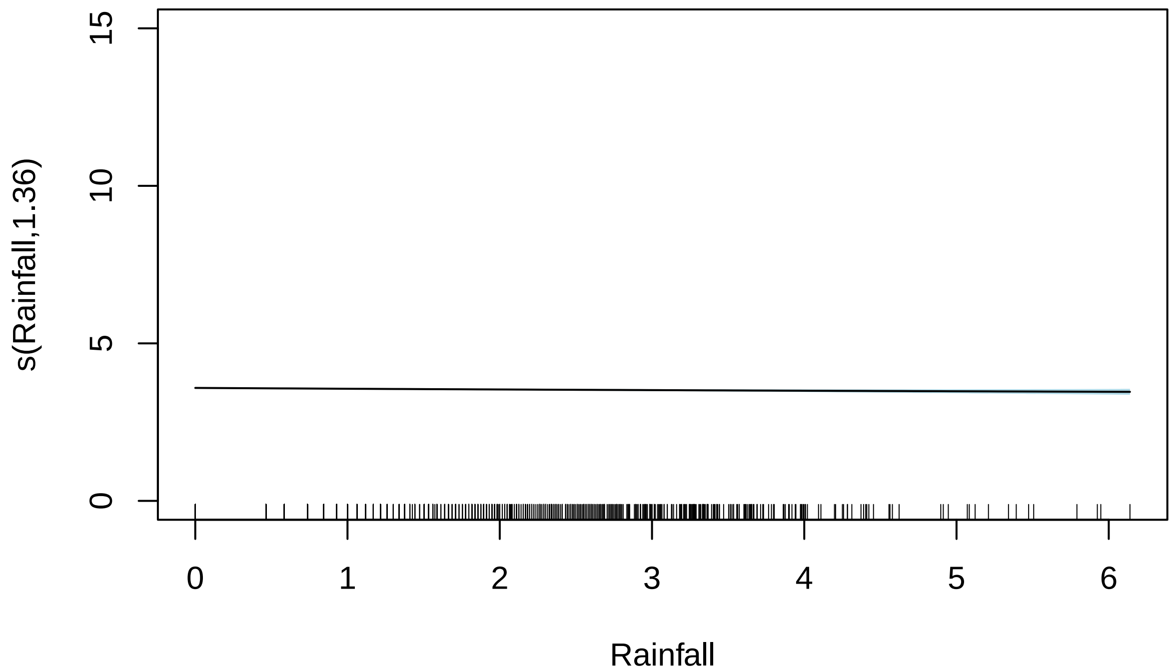


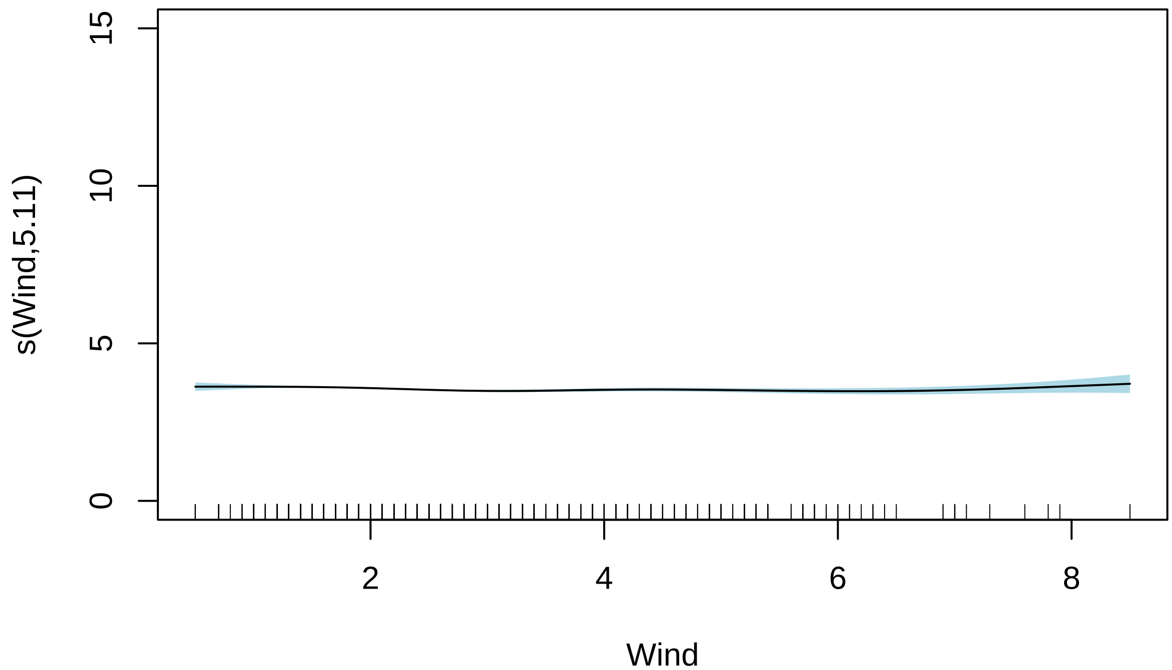
s(O3,8.26):Period1
0
5
10
15
0
50
100
150
200
250
300
350
O3


s(O3,4.08):Period2
0
5
10
15
0
50
100
150
200
250
300
350
O3

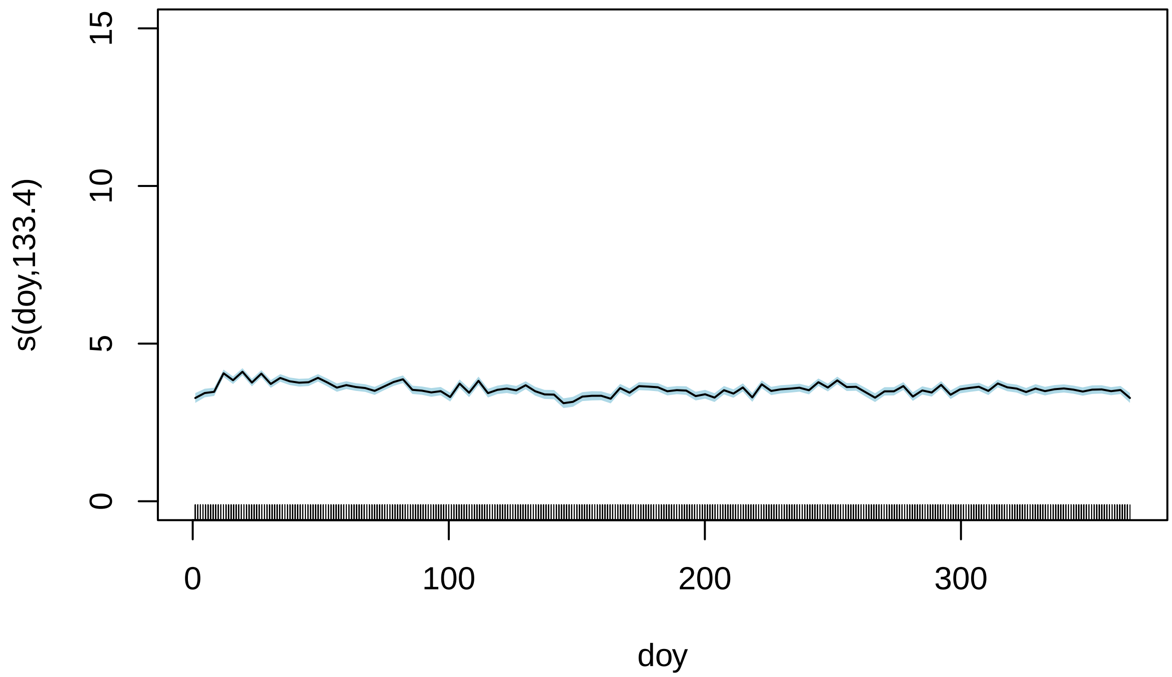

s(O3,2.68):Period3
0
5
10
15
0
50
100
150
200
250
300
350
O3

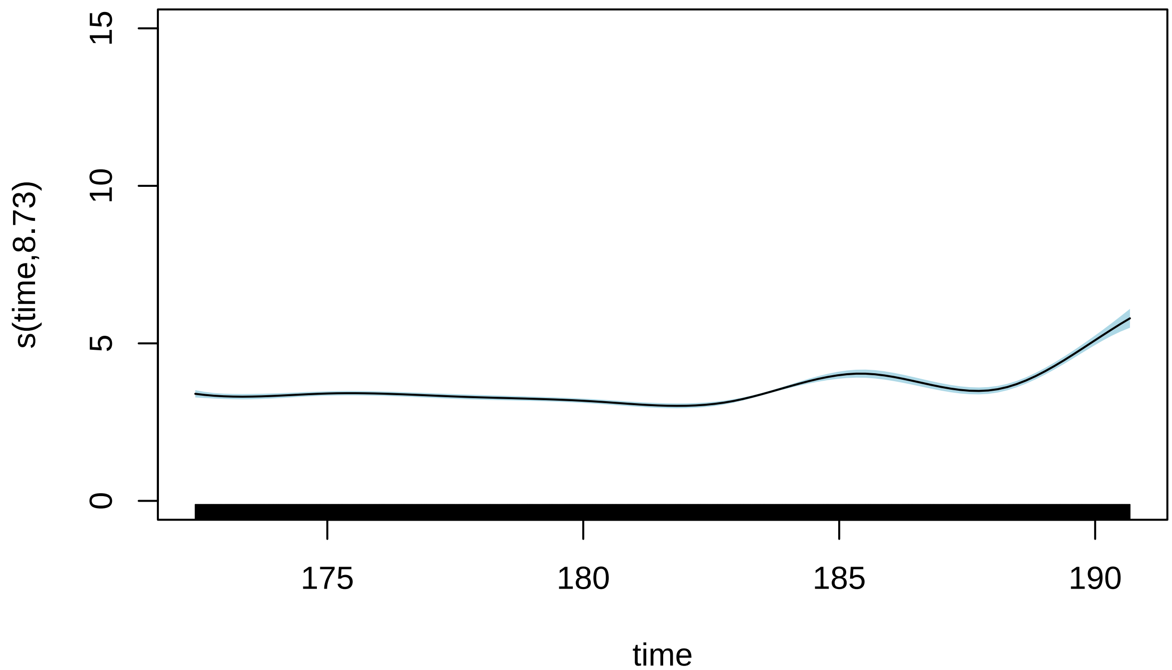

s(Mean,6.91)
0
5
10
15
24
26
28
30
Mean


s(Rainfall,1.36)
0
5
10
15
0
1
2
3
4
5
6
Rainfall

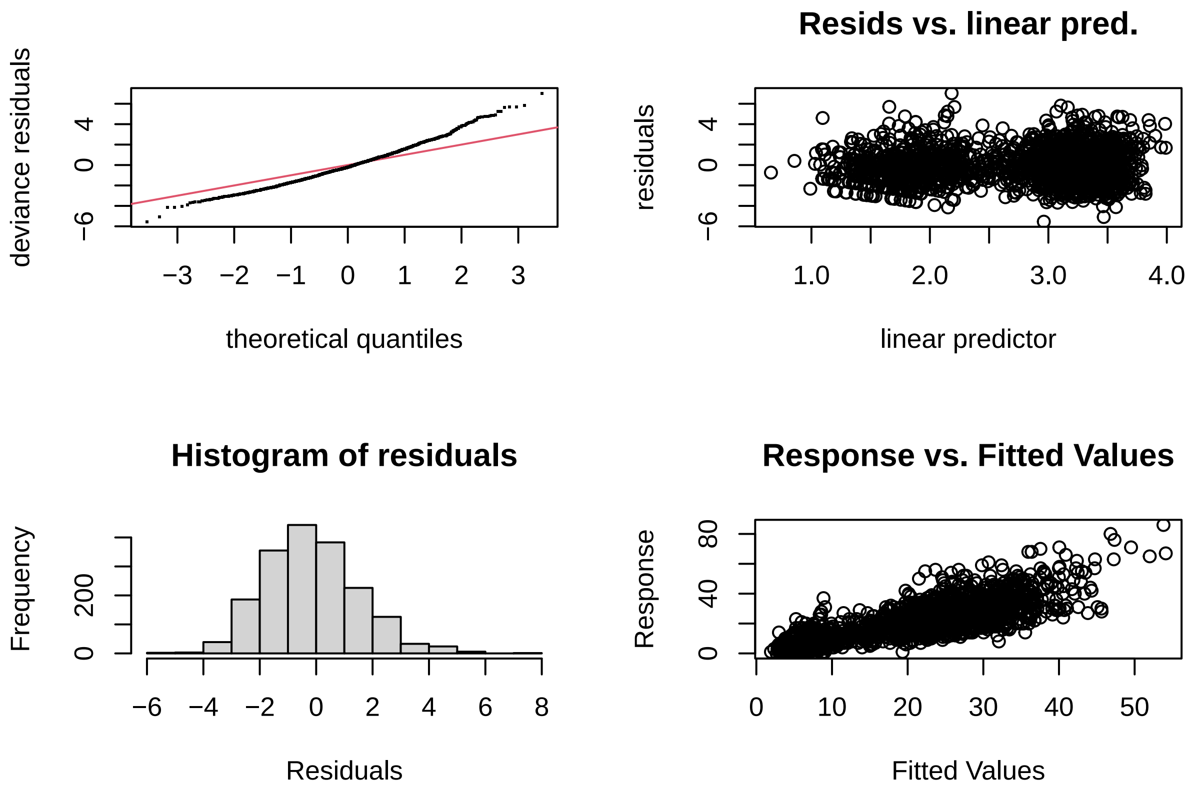


s(Wind,5.11)
0
5
10
15
2
4
6
8
Wind


s(doy,133.4)
15
10
5
0
0
100
200
300
doy




```
gam.check(kb_mod_period)
```

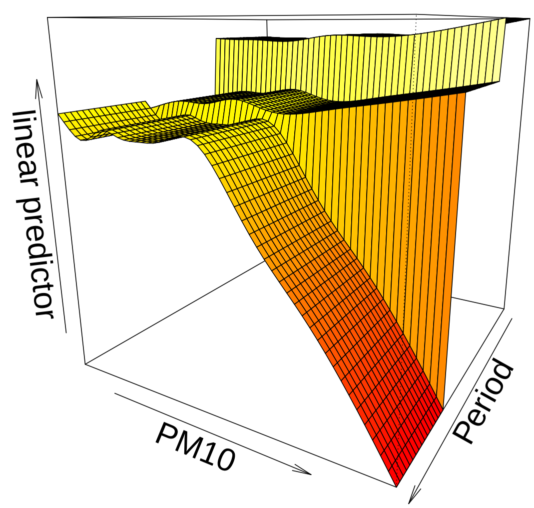



```
##
## Method: REML   Optimizer: outer newton
## full convergence after 16 iterations.
## Gradient range [-0.001241074,0.001889091]
## (score 7141.822 & scale 1).
## Hessian positive definite, eigenvalue range [9.756465e-05,20.00885].
## Model rank =  537 / 537
##
## Basis dimension (k) checking results. Low p-value (k-index<1) may
## indicate that k is too low, especially if edf is close to k'.
##
##                     k'  edf k-index p-value
## s(PM10):Period1   9.00 6.68    1.01   0.795
## s(PM10):Period2   9.00 5.70    1.01   0.800
## s(PM10):Period3   9.00 5.97    1.01   0.785
## s(NO2):Period1    9.00 1.00    0.99   0.470
## s(NO2):Period2    9.00 1.00    0.99   0.470
## s(NO2):Period3    9.00 3.13    0.99   0.510
## s(CO3):Period1    9.00 2.86    0.98   0.210
## s(CO3):Period2    9.00 1.00    0.98   0.230
## s(CO3):Period3    9.00 1.02    0.98   0.260
## s(SO2):Period1    9.00 2.90    0.94   0.005 **
## s(SO2):Period2    9.00 7.14    0.94   0.015 *
## s(SO2):Period3    9.00 1.08    0.94   0.020 *
## s(O3):Period1     9.00 8.26    1.04   0.960
## s(O3):Period2     9.00 4.08    1.04   0.970
```

```
## s(O3):Period3     9.00   2.68    1.04   0.960
## s(Mean)           9.00   6.91    0.94   0.015 *
## s(Rainfall)       9.00   1.36    0.88  <2e-16 ***
## s(Wind)           9.00   5.11    0.90  <2e-16 ***
## s(doy)          363.00 133.40    1.11   1.000
## s(time)           9.00   8.73    0.75  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Checking for the validity of including 'Period' as interaction within the model

```
vis.gam(kb_mod_period, theta = 120, n.grid = 51, lwd = 0.4)
```




Compare models with and without interaction

```
AIC(kb_nopm2.5, kb_mod_period)
```

```
##                     df      AIC
## kb_nopm2.5    189.9868 13758.48
## kb_mod_period 233.2632 13691.03
```

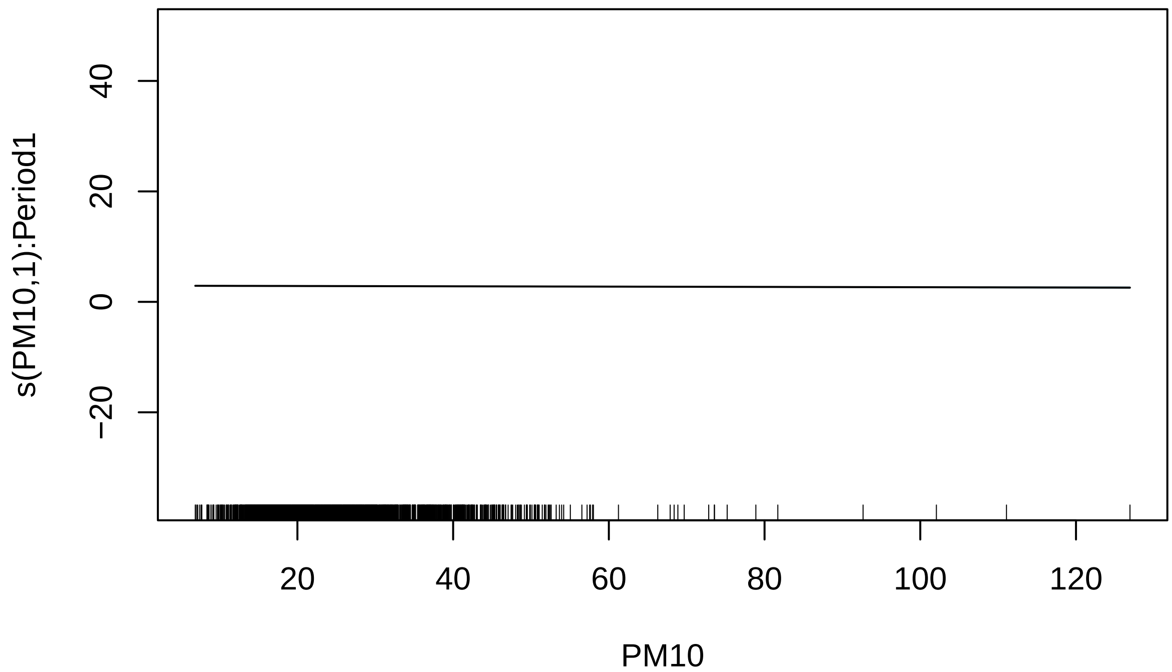
Since the AIC for the model with interaction is higher compared to without the interaction, the model with interaction will be included as the final model

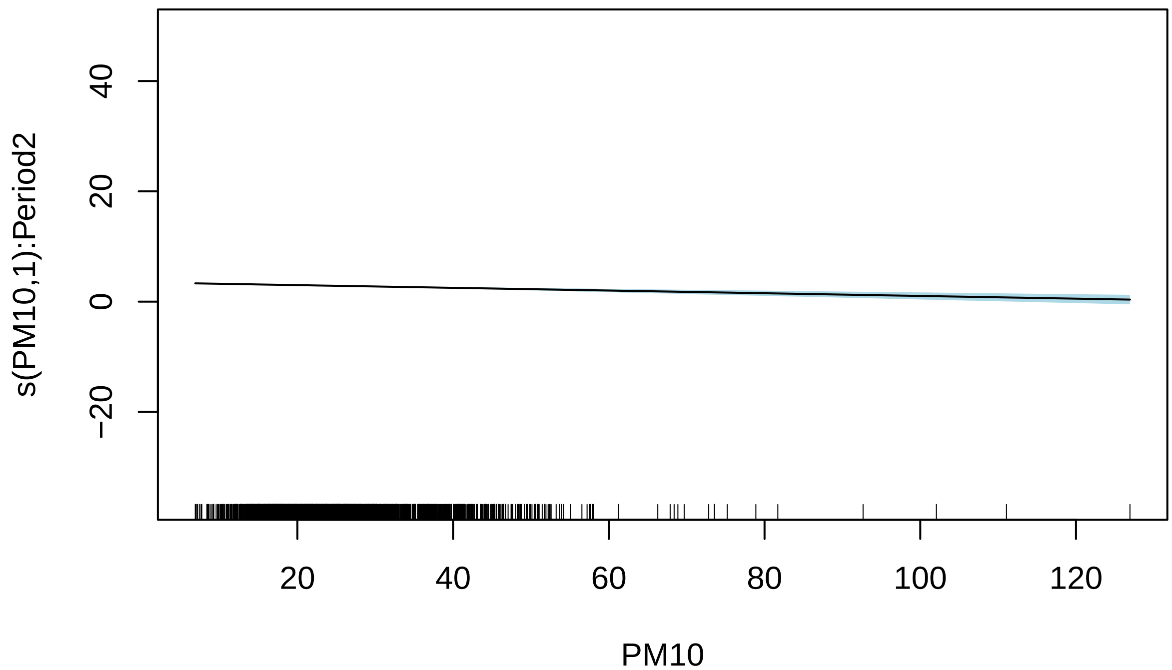
**JB by period**

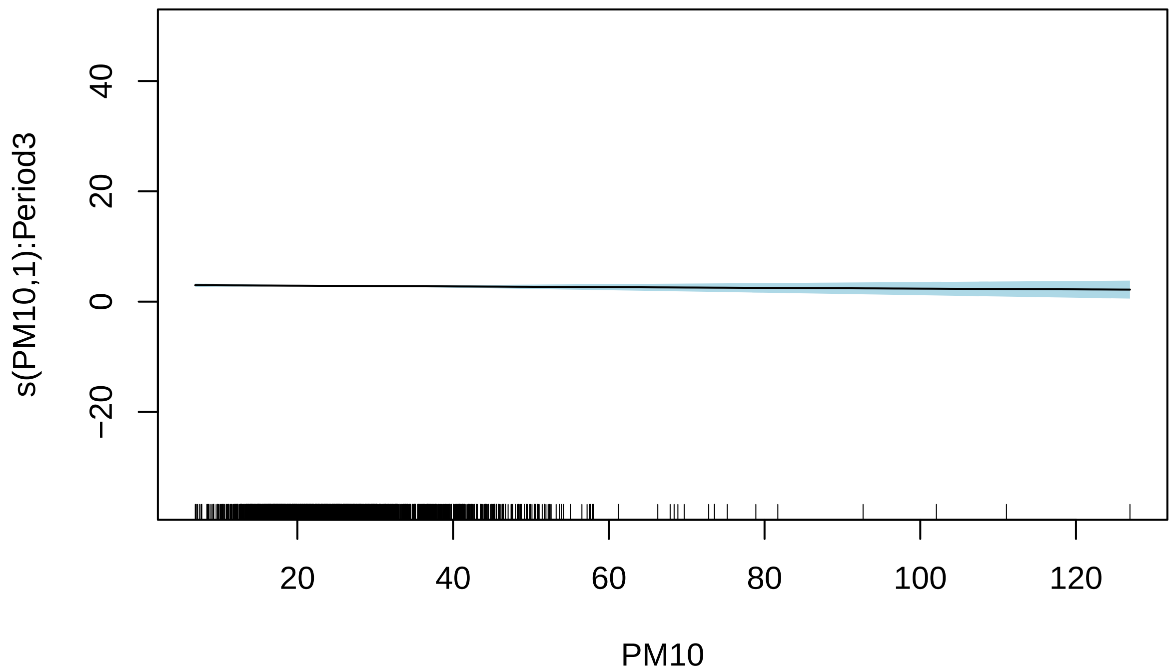
```
jb_mod_period <- gam(ED ~ s(PM10, by=Period) + s(NO2, by=Period) + s(CO3, by=Period) + s(SO2, by=Period)
summary(jb_mod_period)
```

```
##
## Family: poisson
## Link function: log
##
## Formula:
## ED ~ s(PM10, by = Period) + s(NO2, by = Period) + s(CO3, by = Period) +
##     s(SO2, by = Period) + s(O3, by = Period) + s(Mean) + s(Rainfall) +
##     s(Wind) + s(doy, bs = "cc", k = 365) + s(time, bs = "bs") +
##     Period
##
## Parametric coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  2.85502    0.04384  65.127   <2e-16 ***
## Period2     -2.07956    0.15272 -13.617   <2e-16 ***
## Period3     -1.64576    0.17307  -9.509   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Approximate significance of smooth terms:
##                     edf  Ref.df  Chi.sq  p-value
## s(PM10):Period1   1.002   1.005   6.151 0.013384 *
## s(PM10):Period2   1.000   1.000  34.528  < 2e-16 ***
## s(PM10):Period3   1.000   1.001   0.699 0.403656
## s(NO2):Period1    1.919   2.440   1.850 0.606660
## s(NO2):Period2    7.361   7.909  43.811  < 2e-16 ***
## s(NO2):Period3    1.000   1.001   1.748 0.186268
## s(CO3):Period1    1.001   1.002   0.873 0.350790
## s(CO3):Period2    4.517   5.521  25.441 0.000195 ***
## s(CO3):Period3    1.001   1.003   1.094 0.295994
## s(SO2):Period1    3.235   4.061   4.610 0.329463
## s(SO2):Period2    4.022   4.529  36.140 3.59e-07 ***
## s(SO2):Period3    1.000   1.001   1.347 0.245812
## s(O3):Period1     6.520   7.657  30.888 0.000122 ***
## s(O3):Period2     5.053   5.824  68.363  < 2e-16 ***
## s(O3):Period3     1.743   2.089  11.514 0.003682 **
## s(Mean)           1.000   1.000   0.044 0.834348
## s(Rainfall)       1.831   2.283   9.730 0.012308 *
## s(Wind)           2.993   3.778   3.783 0.323538
## s(doy)          104.123 363.000 629.313  < 2e-16 ***
## s(time)           8.452   8.866 334.797  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-sq.(adj) =  0.492   Deviance explained =   53%
## -REML = 5747.7  Scale est. = 1         n = 1827
```

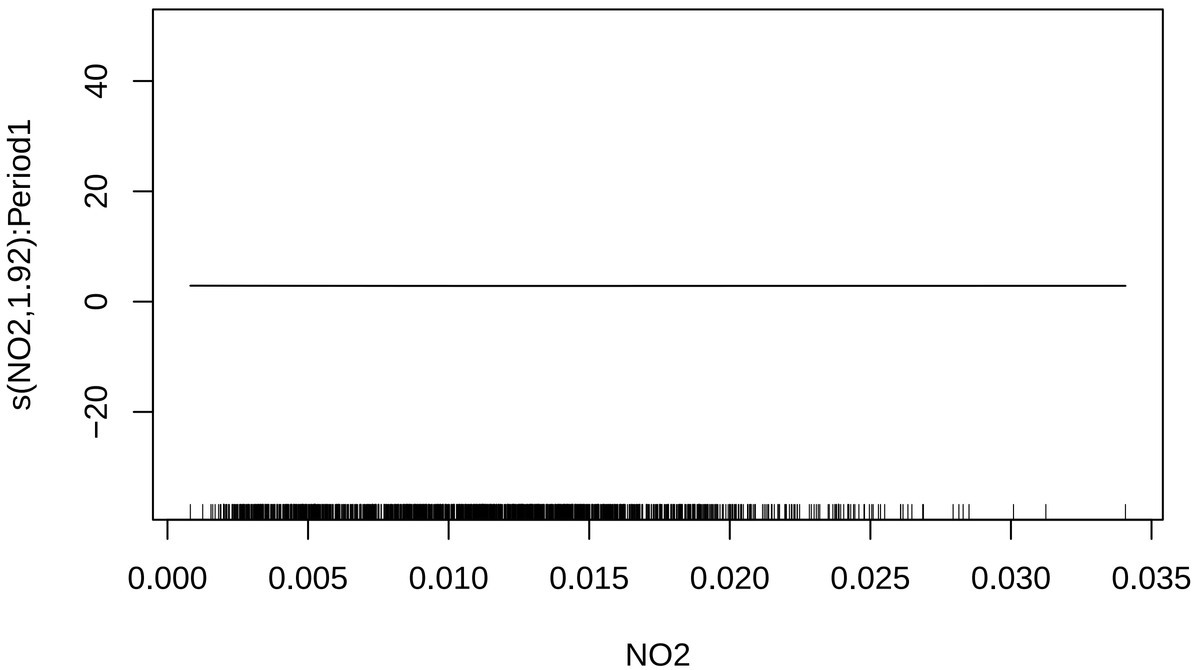
```
plot.gam(jb_mod_period, shade = TRUE, shade.col = "lightblue", shift = coef(jb_mod_period)[1],  seWithM
```

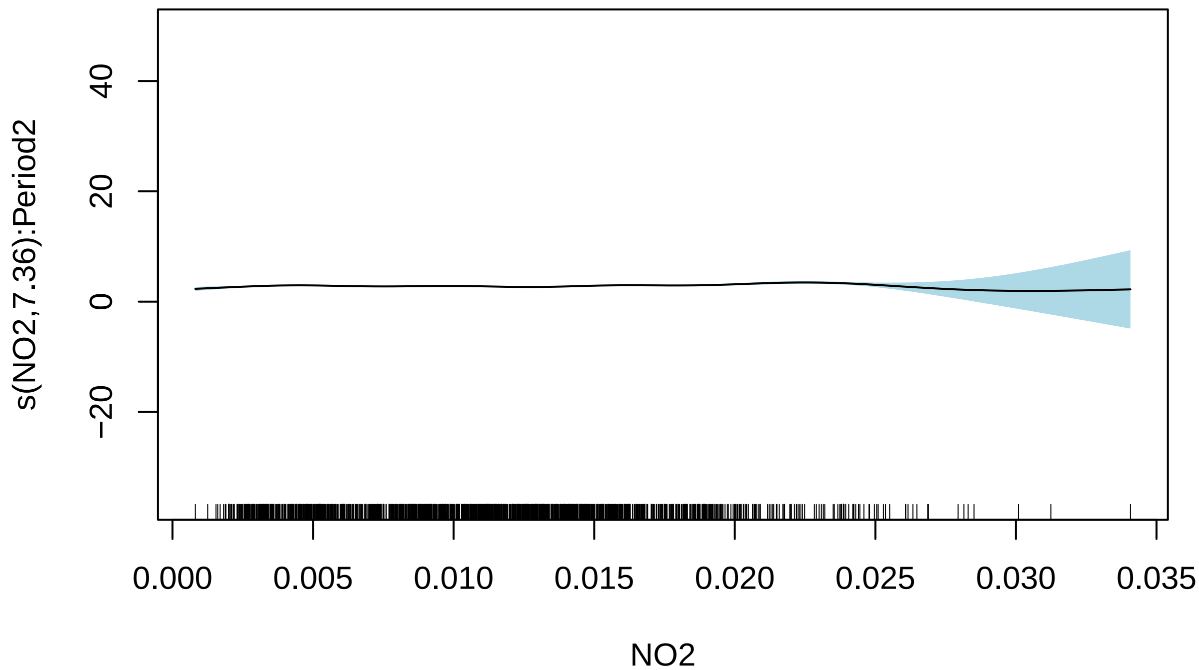


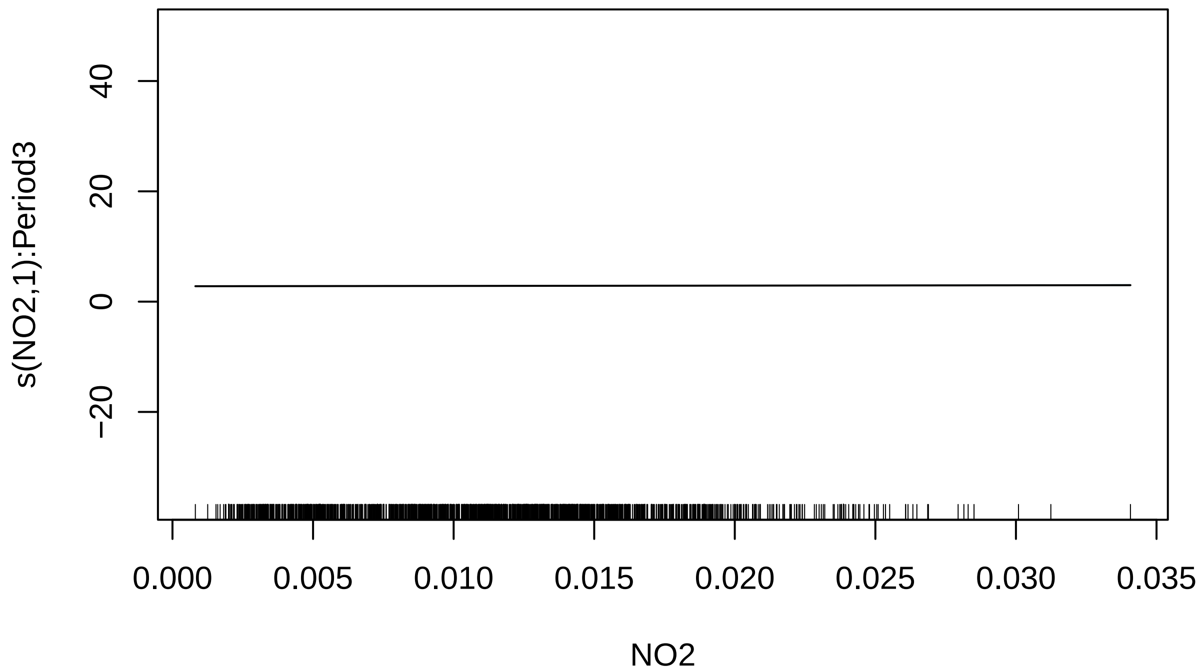

s(PM10,1):Period2
40
20
0
−20
20
40
60
80
100
120
PM10



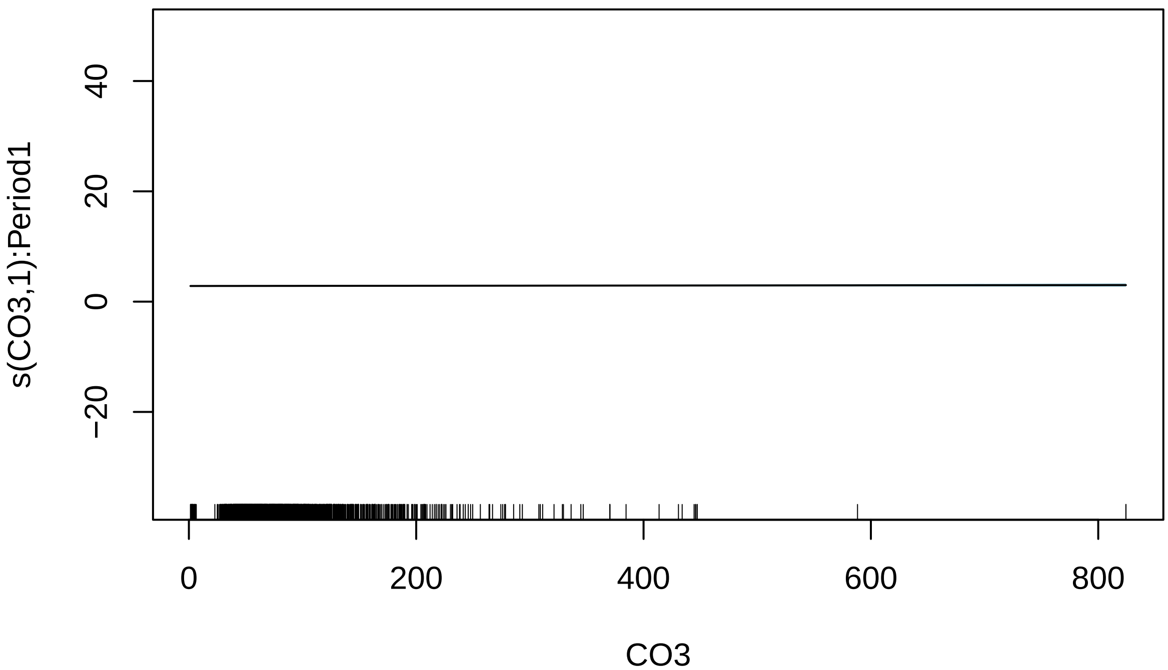
s(PM10,1):Period3
40
20
0
−20
20
40
60
80
100
120
PM10

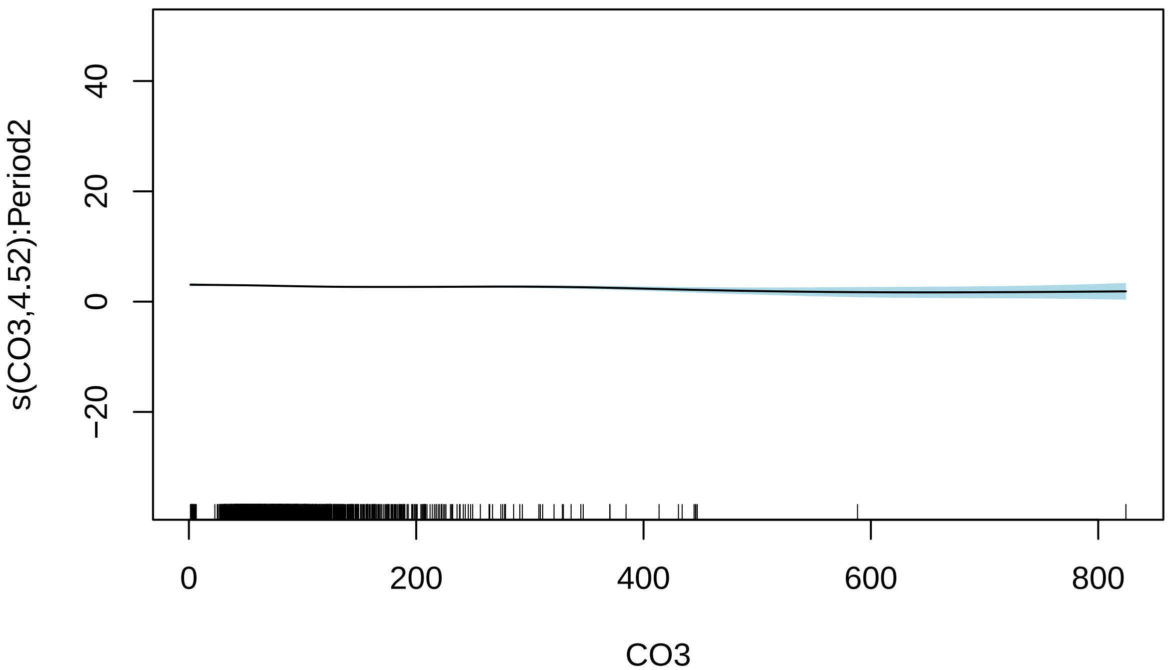

s(NO2,1.92):Period1
40
20
0
−20
0.000
0.005
0.010
0.015
0.020
0.025
0.030
0.035
NO2

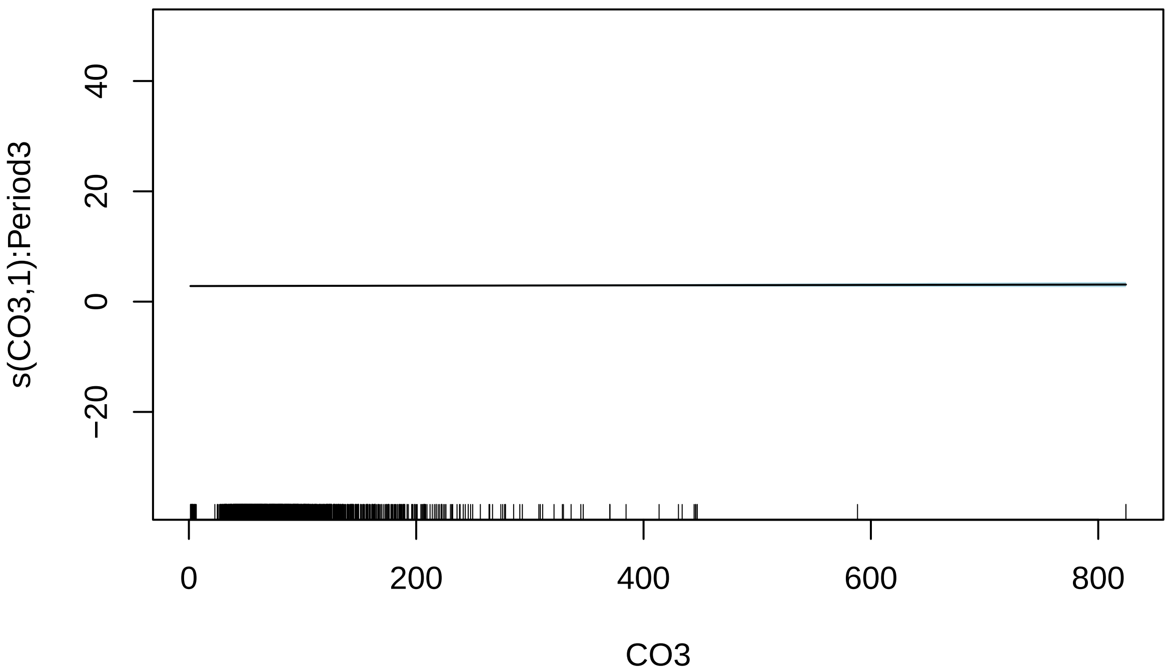

s(NO2,7.36):Period2
40
20
0
−20
0.000
0.005
0.010
0.015
0.020
0.025
0.030
0.035
NO2

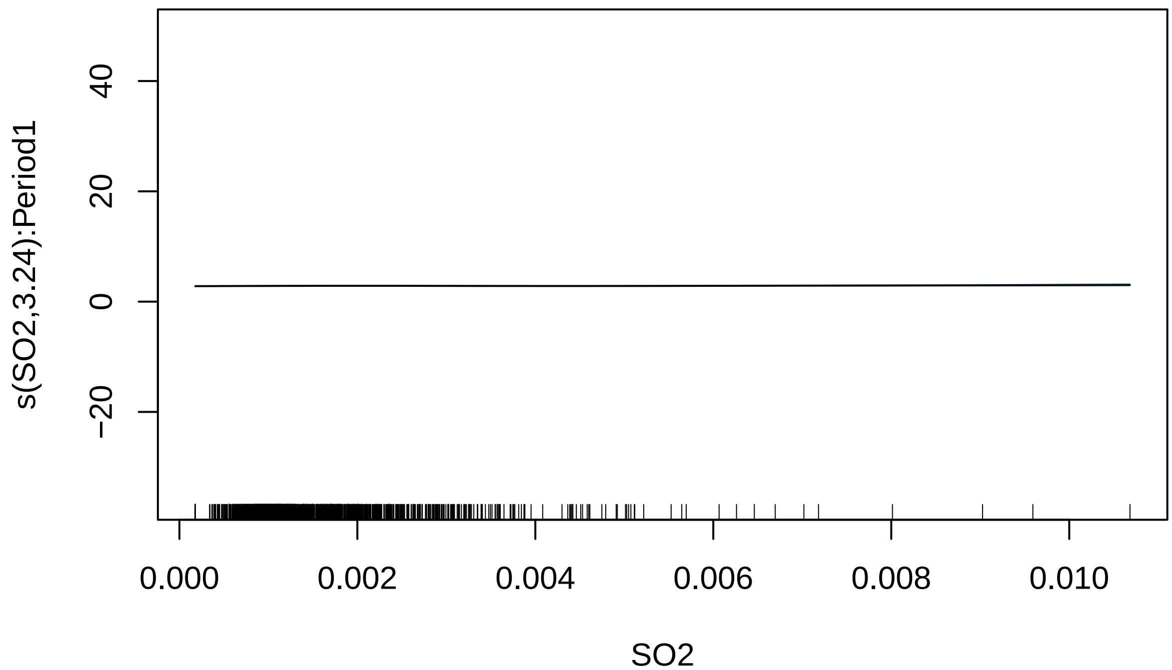


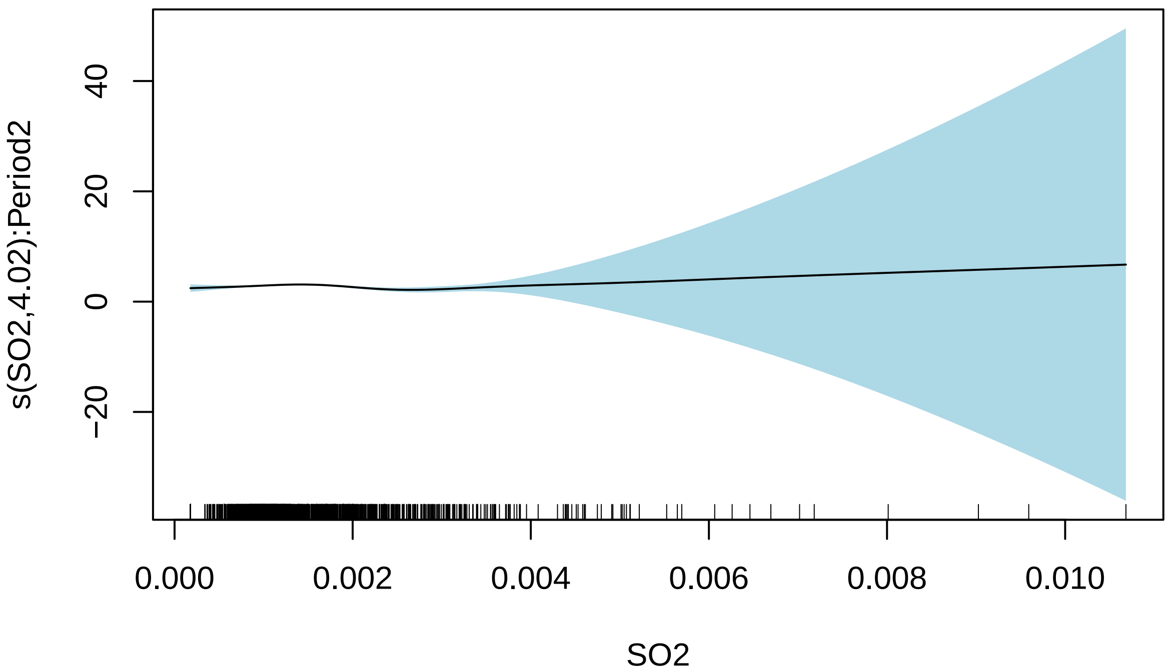
s(NO2,1):Period3
40
20
0
−20
0.000
0.005
0.010
0.015
0.020
0.025
0.030
0.035
NO2


s(CO3,1):Period1
40
20
0
−20
0
200
400
600
800
CO3

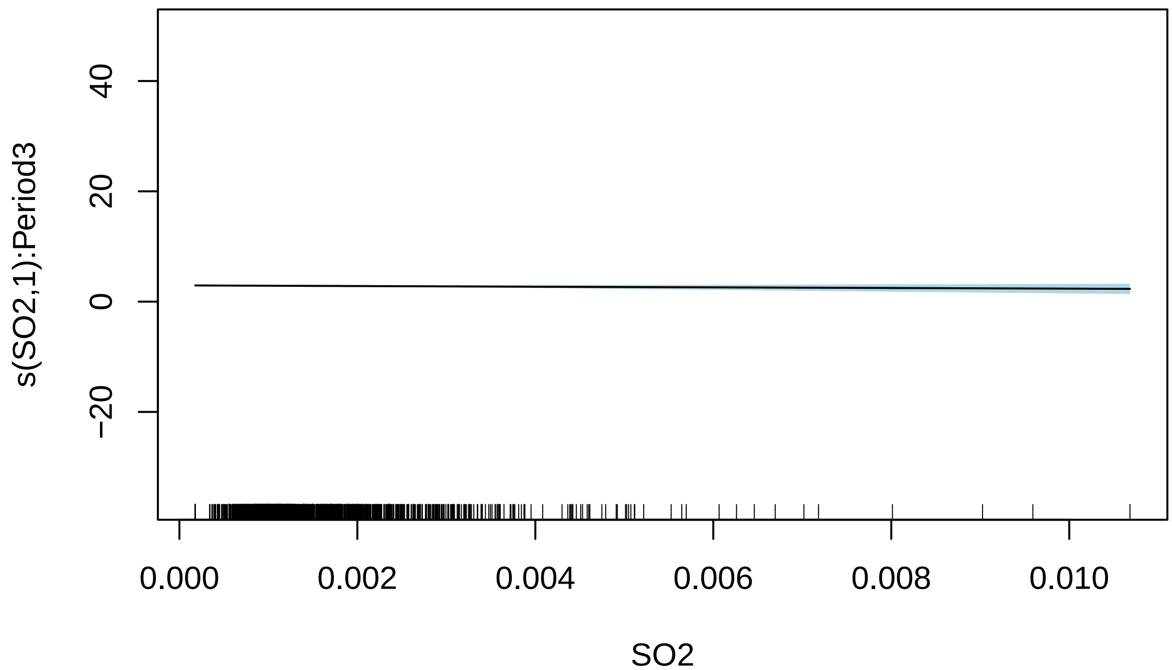

40
20
0
−20
s(CO3,4.52):Period2
0
200
400
600
800
CO3

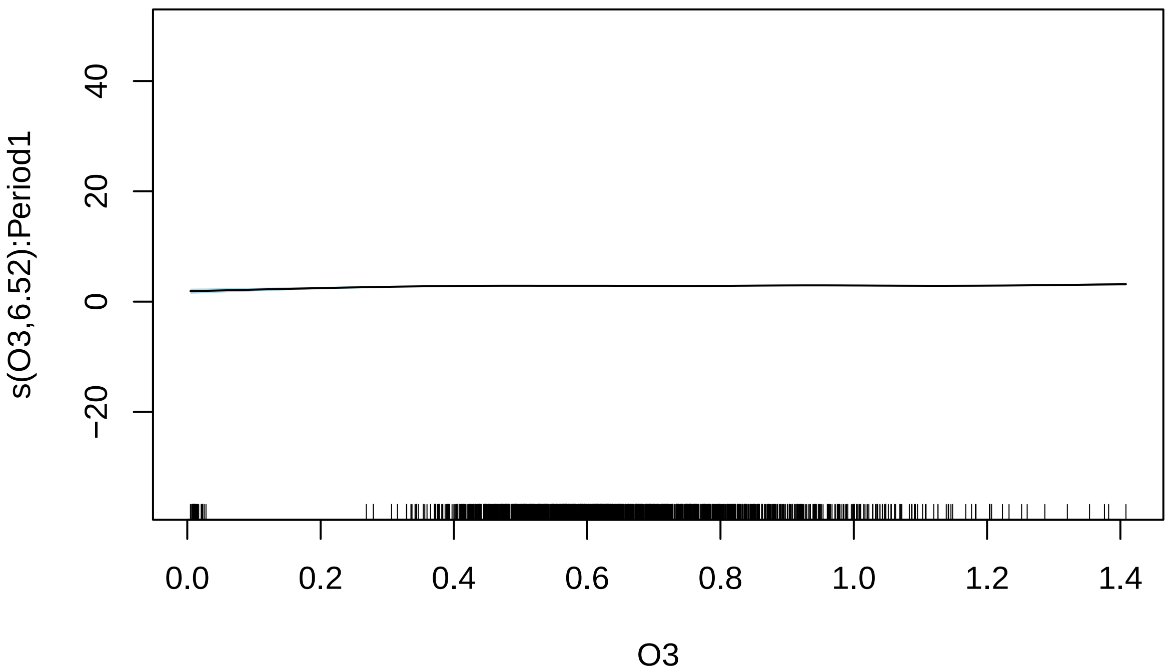

s(CO3,1):Period3
40
20
0
−20
0
200
400
600
800
CO3

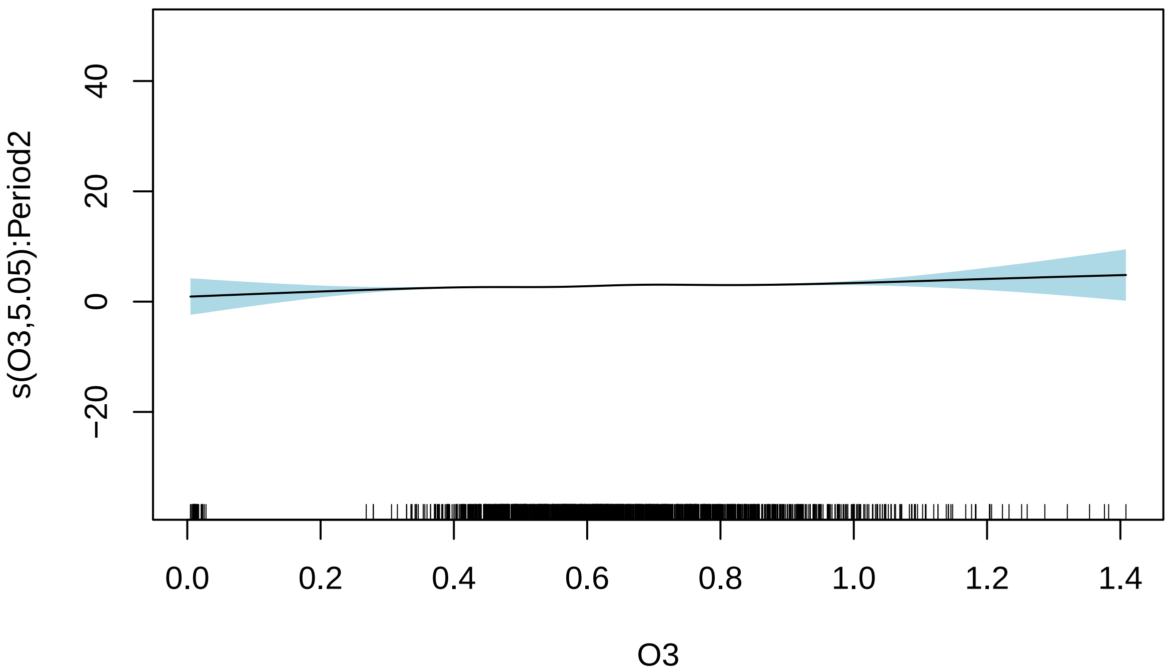

s(SO2,3.24):Period1
40
20
0
−20
0.000
0.002
0.004
0.006
0.008
0.010
SO2



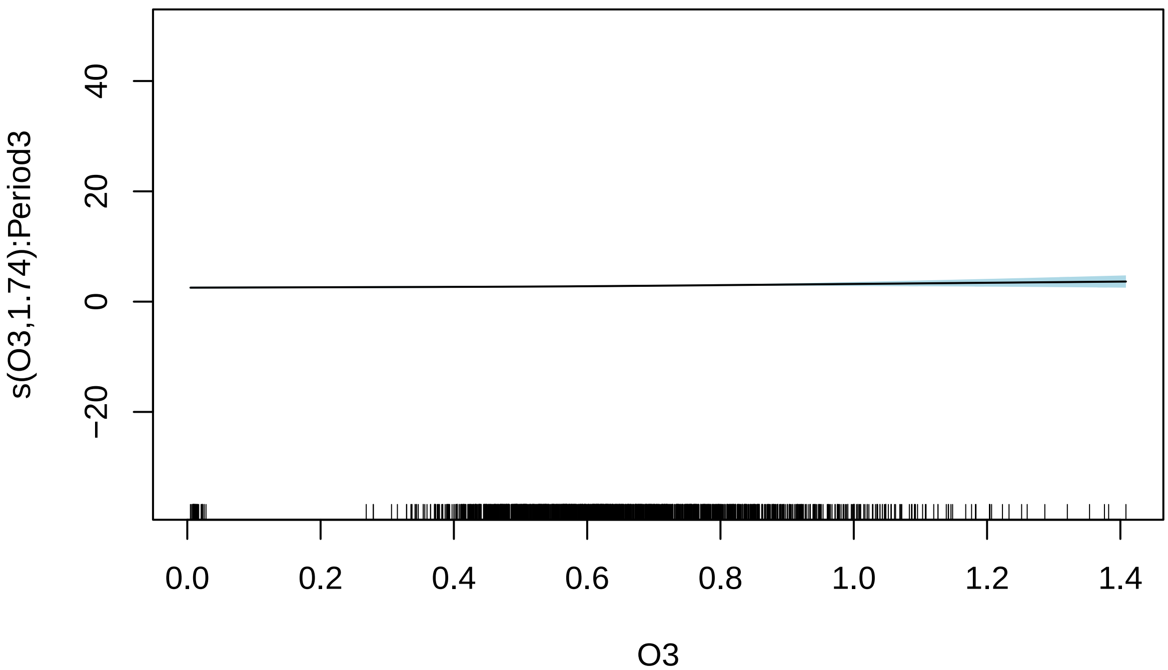
s(SO2,4.02):Period2
40
20
0
−20
0.000
0.002
0.004
0.006
0.008
0.010
SO2

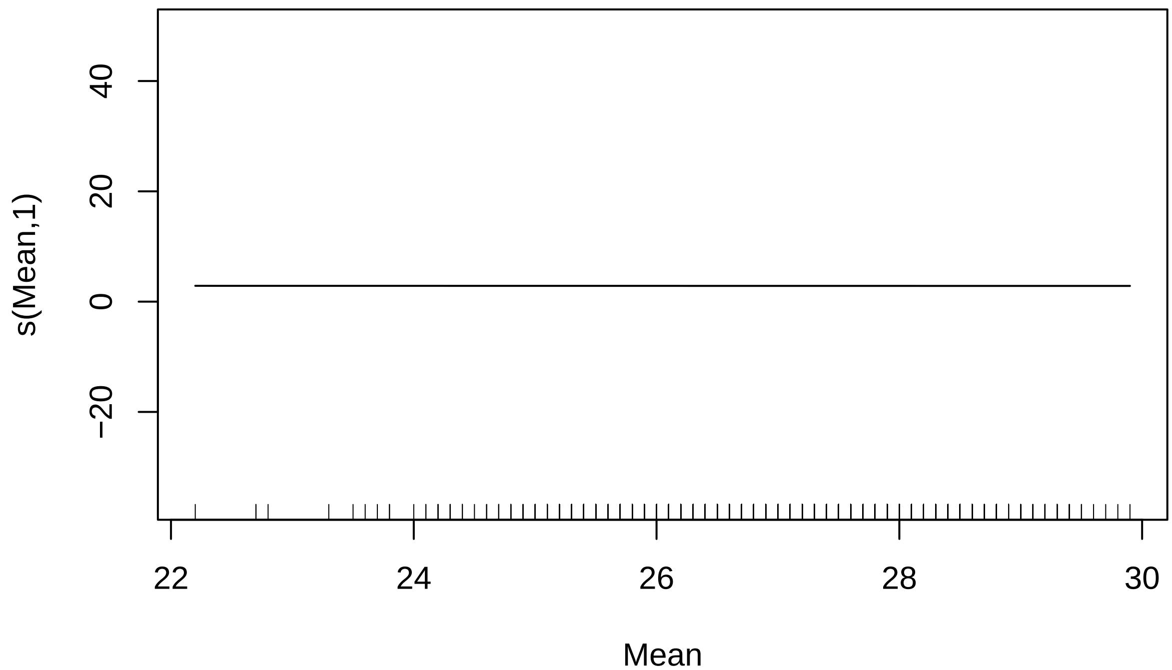

s(SO2,1):Period3
40
20
0
−20
0.000
0.002
0.004
0.006
0.008
0.010
SO2

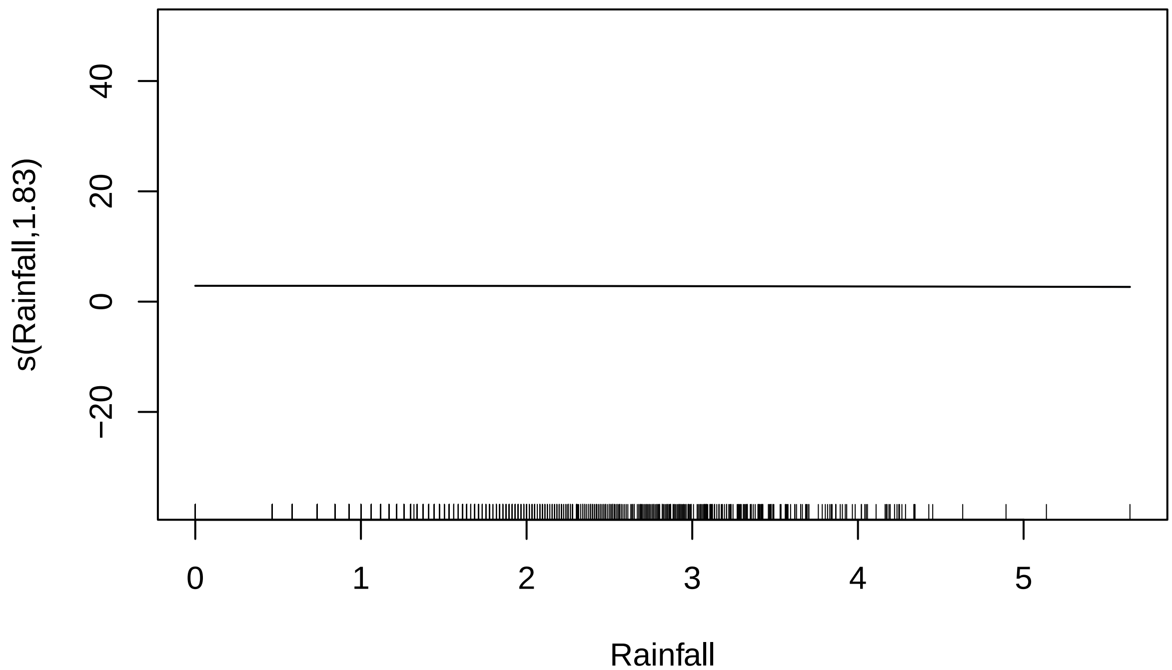

s(O3,6.52):Period1
40
20
0
−20
0.0
0.2
0.4
0.6
0.8
1.0
1.2
1.4
O3

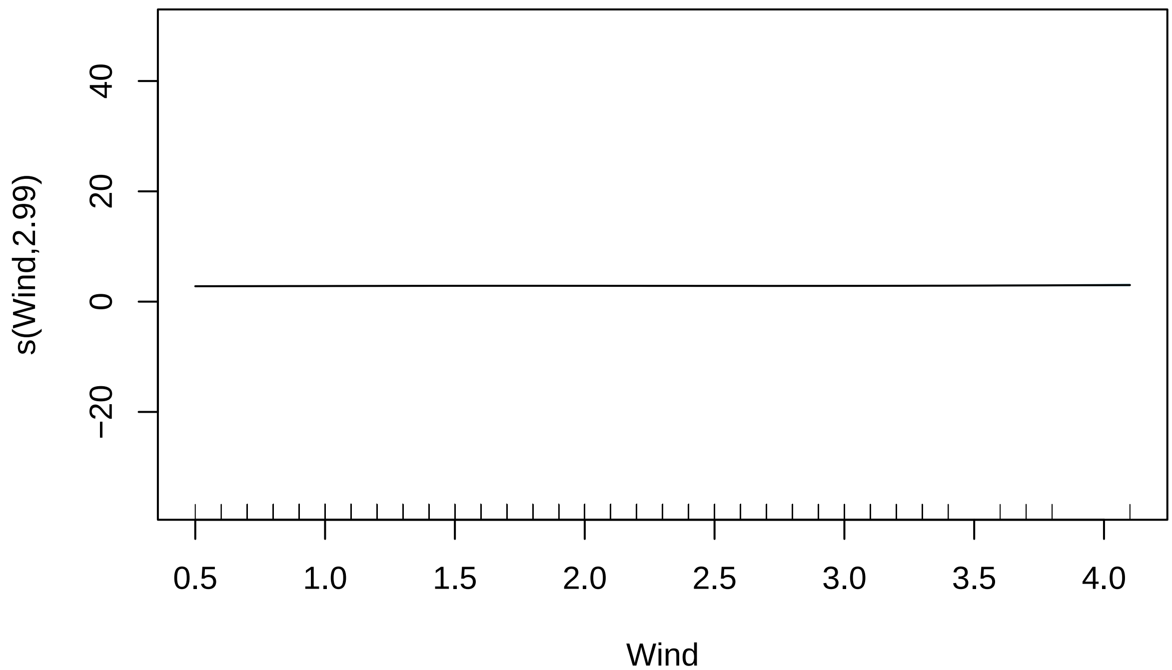


s(O3,5.05):Period2
40
20
0
−20
0.0
0.2
0.4
0.6
0.8
1.0
1.2
1.4
O3

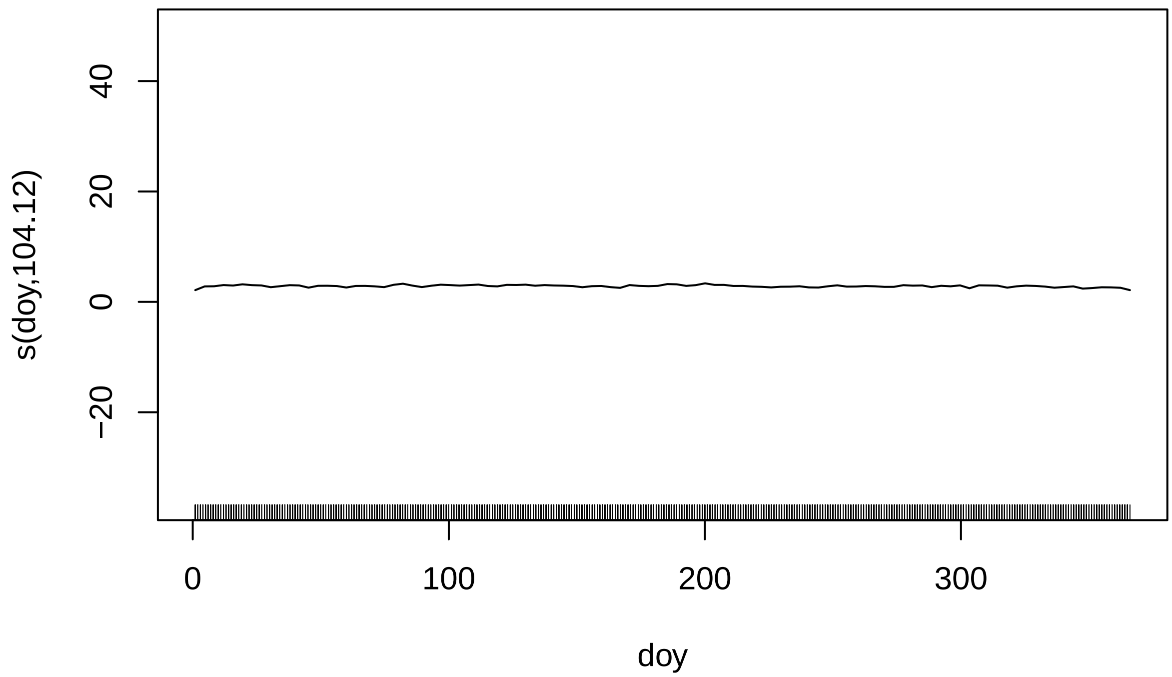

s(O3,1.74):Period3
40
20
0
−20
0.0
0.2
0.4
0.6
0.8
1.0
1.2
1.4
O3

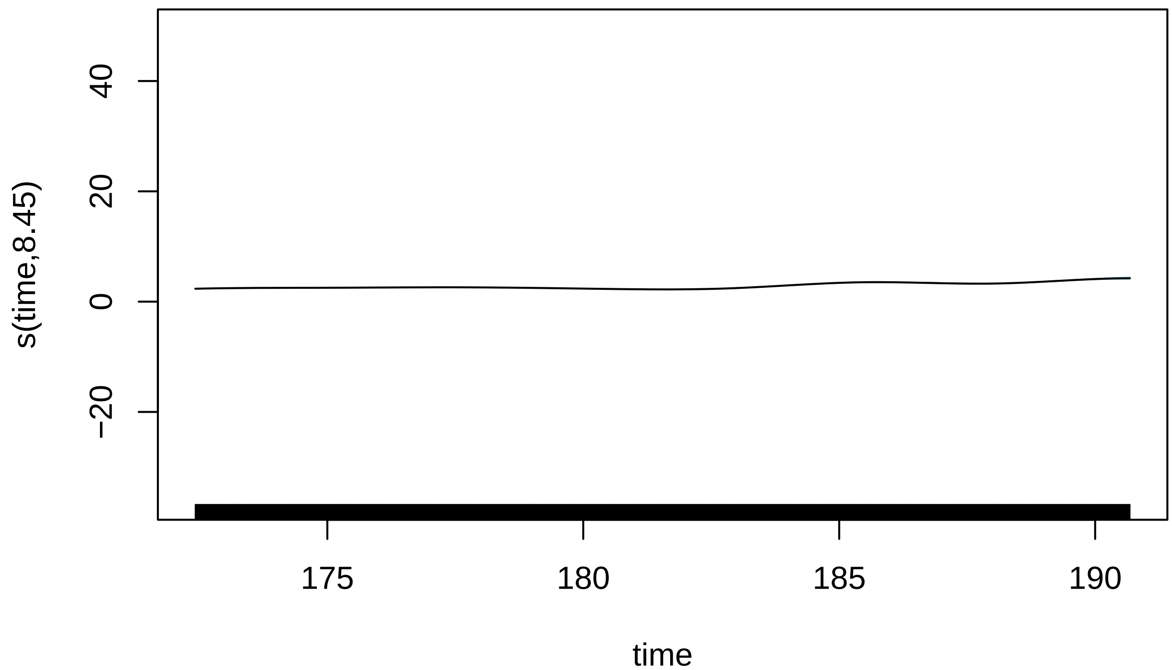

40
20
0
−20
s(Mean,1)
22
24
26
28
30
Mean


s(Rainfall,1.83)
40
20
0
−20
0
1
2
3
4
5
Rainfall

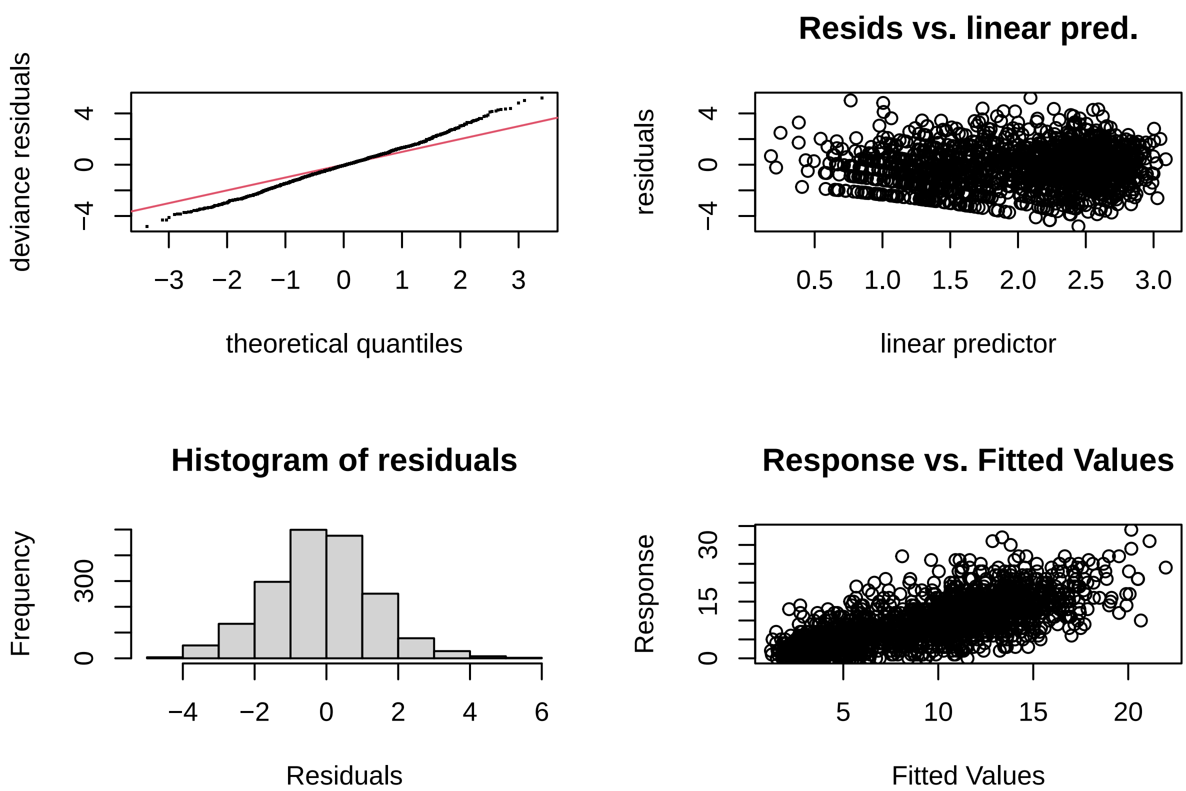


s(Wind,2.99)
40
20
0
−20
0.5
1.0
1.5
2.0
2.5
3.0
3.5
4.0
Wind


s(doy,104.12)
40
20
0
−20
0
100
200
300
doy




```
gam.check(jb_mod_period)
```




```
##
## Method: REML   Optimizer: outer newton
## full convergence after 13 iterations.
## Gradient range [-0.0004283131,0.0003544774]
## (score 5747.723 & scale 1).
## Hessian positive definite, eigenvalue range [3.78279e-05,5.918716].
## Model rank =  537 / 537
##
## Basis dimension (k) checking results. Low p-value (k-index<1) may
## indicate that k is too low, especially if edf is close to k'.
##
##                      k'   edf k-index p-value
## s(PM10):Period1    9.00  1.00    0.99   0.515
## s(PM10):Period2    9.00  1.00    0.99   0.565
## s(PM10):Period3    9.00  1.00    0.99   0.565
## s(NO2):Period1     9.00  1.92    0.98   0.370
## s(NO2):Period2     9.00  7.36    0.98   0.300
## s(NO2):Period3     9.00  1.00    0.98   0.380
## s(CO3):Period1     9.00  1.00    1.00   0.660
## s(CO3):Period2     9.00  4.52    1.00   0.680
## s(CO3):Period3     9.00  1.00    1.00   0.740
## s(SO2):Period1     9.00  3.24    0.92  <2e-16 ***
## s(SO2):Period2     9.00  4.02    0.92  <2e-16 ***
## s(SO2):Period3     9.00  1.00    0.92  <2e-16 ***
## s(O3):Period1      9.00  6.52    0.99   0.370
## s(O3):Period2      9.00  5.05    0.99   0.380
```

```
## s(O3):Period3     9.00   1.74    0.99   0.390
## s(Mean)           9.00   1.00    0.94   0.015 *
## s(Rainfall)       9.00   1.83    0.94   0.015 *
## s(Wind)           9.00   2.99    0.86  <2e-16 ***
## s(doy)          363.00 104.12    1.00   0.580
## s(time)           9.00   8.45    0.85  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Further analysis based on the different respiratory diagnoses

### 3.2 KB analysis

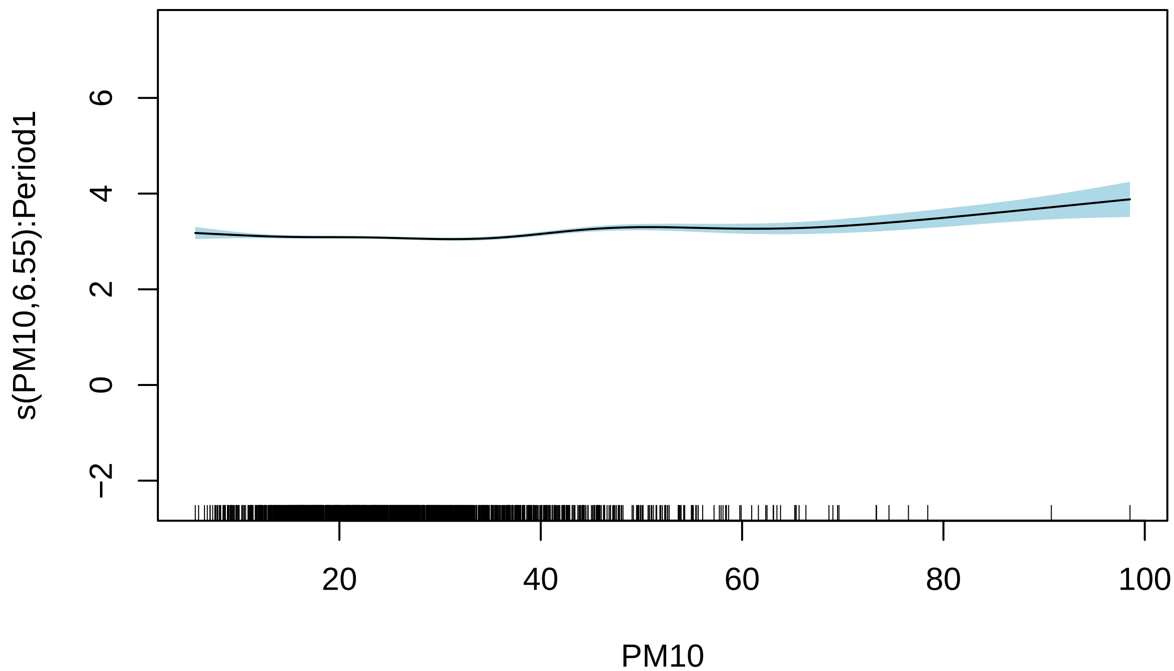
```
kb_URTI_period <- gam(URTI ~ s(PM10, by=Period) + s(NO2, by=Period) + s(CO3, by=Period) + s(SO2, by=Per
summary(kb_URTI_period)
```

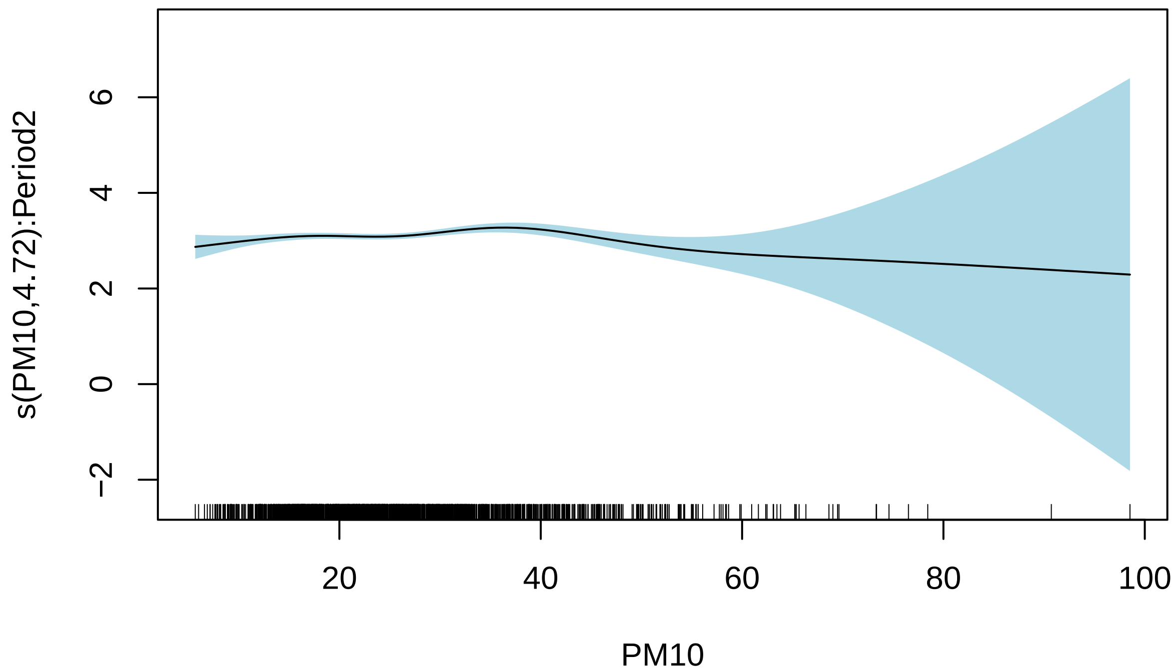
#### KB URTI by Period

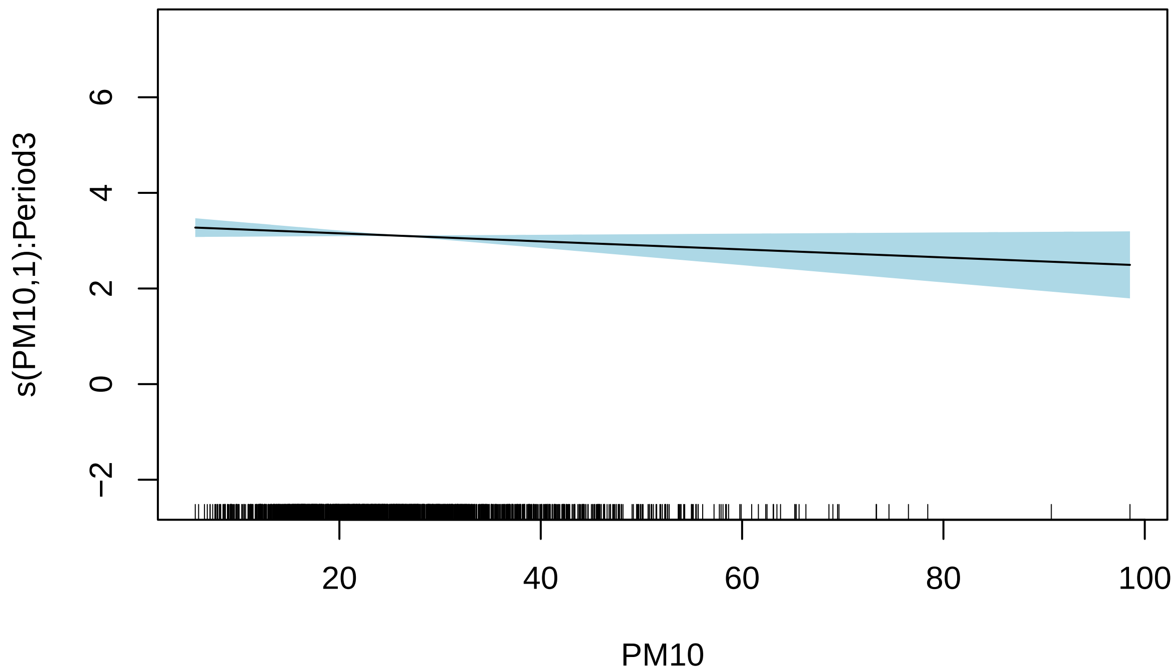
```
##
## Family: poisson
## Link function: log
##
## Formula:
## URTI ~ s(PM10, by = Period) + s(NO2, by = Period) + s(CO3, by = Period) +
##     s(SO2, by = Period) + s(O3, by = Period) + s(Mean) + s(Rainfall) +
##     s(Wind) + s(doy, bs = "cc", k = 365) + s(time, bs = "bs") +
##     Period
##
## Parametric coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  3.10271    0.03657   84.85   <2e-16 ***
## Period2     -1.75960    0.09647  -18.24   <2e-16 ***
## Period3     -1.70733    0.15221  -11.22   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Approximate significance of smooth terms:
##                     edf  Ref.df  Chi.sq  p-value
## s(PM10):Period1   6.548   7.658  69.238  < 2e-16 ***
## s(PM10):Period2   4.717   5.714  20.558  0.00324 **
## s(PM10):Period3   1.000   1.000   3.037  0.08141 .
## s(NO2):Period1    7.644   8.435  32.765 5.90e-05 ***
## s(NO2):Period2    1.001   1.001   0.054  0.81733
## s(NO2):Period3    1.000   1.000   5.930  0.01491 *
## s(CO3):Period1    1.936   2.482   1.536  0.48971
## s(CO3):Period2    1.009   1.017   0.011  0.94525
## s(CO3):Period3    1.000   1.000   7.259  0.00706 **
## s(SO2):Period1    1.442   1.761  15.900  0.00180 **
## s(SO2):Period2    6.570   7.459  35.085 2.37e-05 ***
## s(SO2):Period3    2.043   2.337   8.750  0.01934 *
## s(O3):Period1     7.984   8.691  57.007  < 2e-16 ***
```

```
## s(O3):Period2     4.113   5.095  27.818 4.46e-05 ***
## s(O3):Period3     2.581   3.171   9.059  0.03296 *
## s(Mean)           7.061   8.013  46.988  < 2e-16 ***
## s(Rainfall)       1.000   1.001  27.940  < 2e-16 ***
## s(Wind)           4.530   5.527  31.399 2.72e-05 ***
## s(doy)          126.211 363.000 880.534  < 2e-16 ***
## s(time)           8.601   8.914 491.612  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-sq.(adj) =  0.629   Deviance explained = 70.6%
## -REML = 6504.3  Scale est. = 1         n = 1827
```

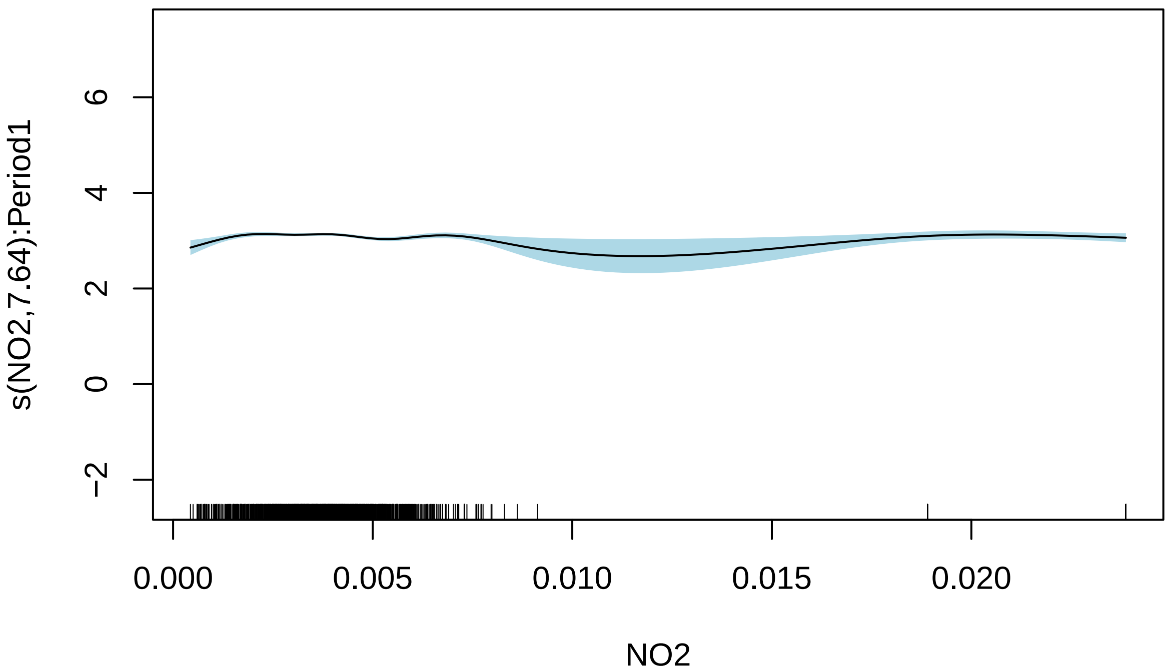
```r
plot.gam(kb_URTI_period, shade = TRUE, shade.col = "lightblue", shift = coef(kb_URTI_period)[1],  seWit
```

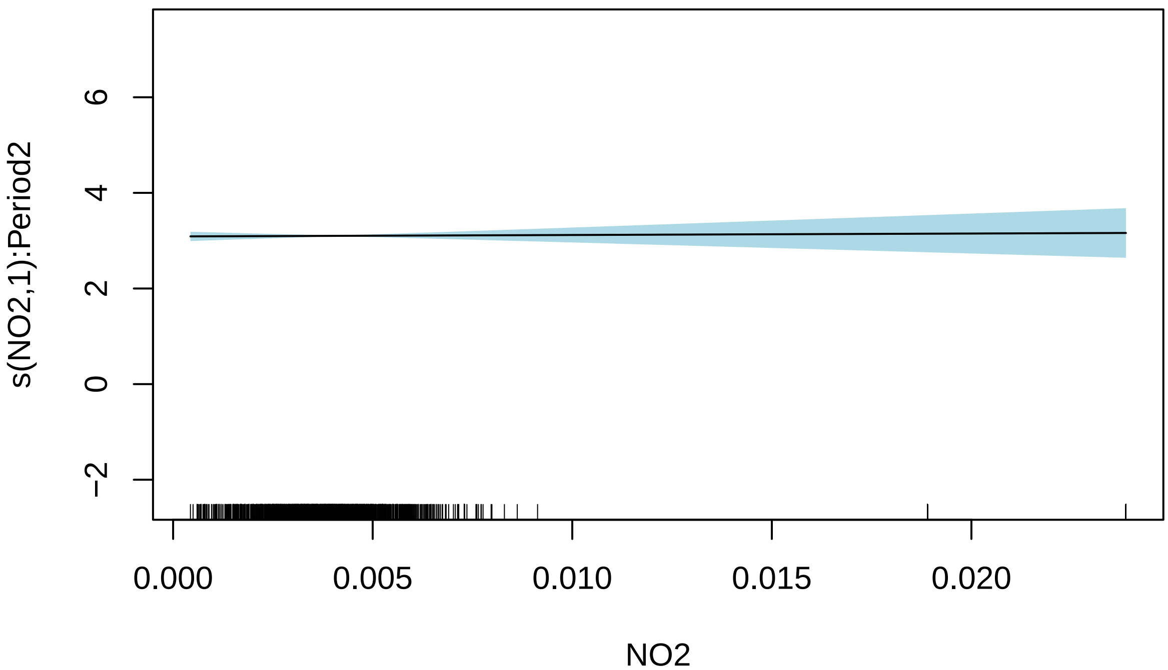



s(PM10,4.72):Period2
−2
0
2
4
6
20
40
60
80
100
PM10

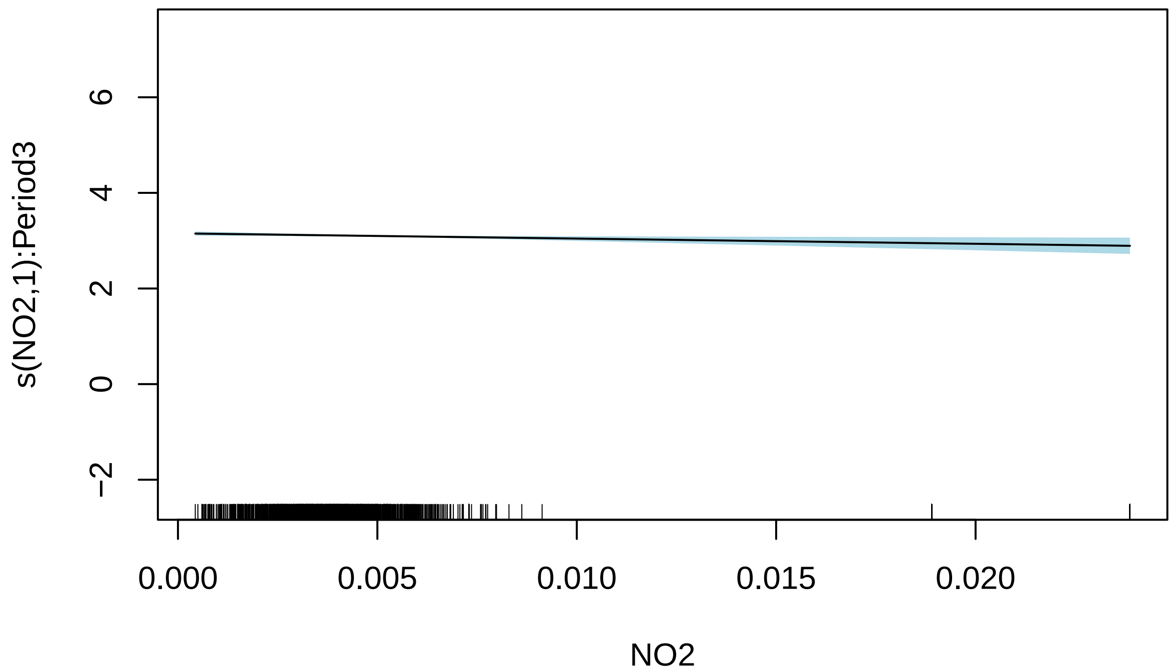


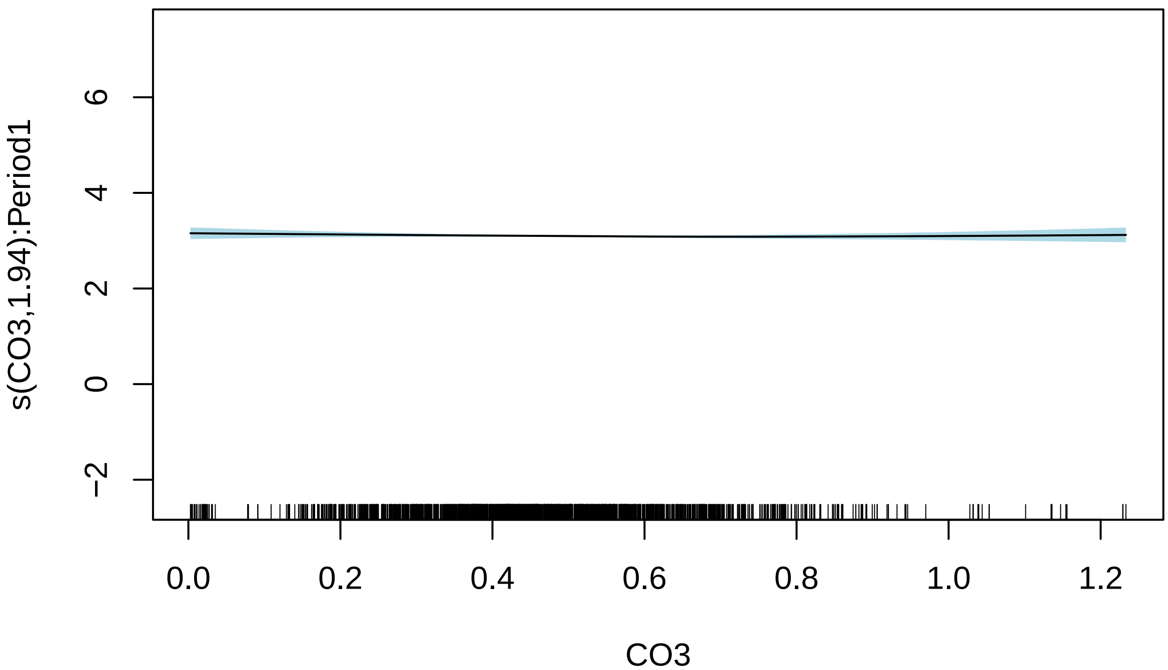
s(PM10,1):Period3
−2
0
2
4
6
20
40
60
80
100
PM10

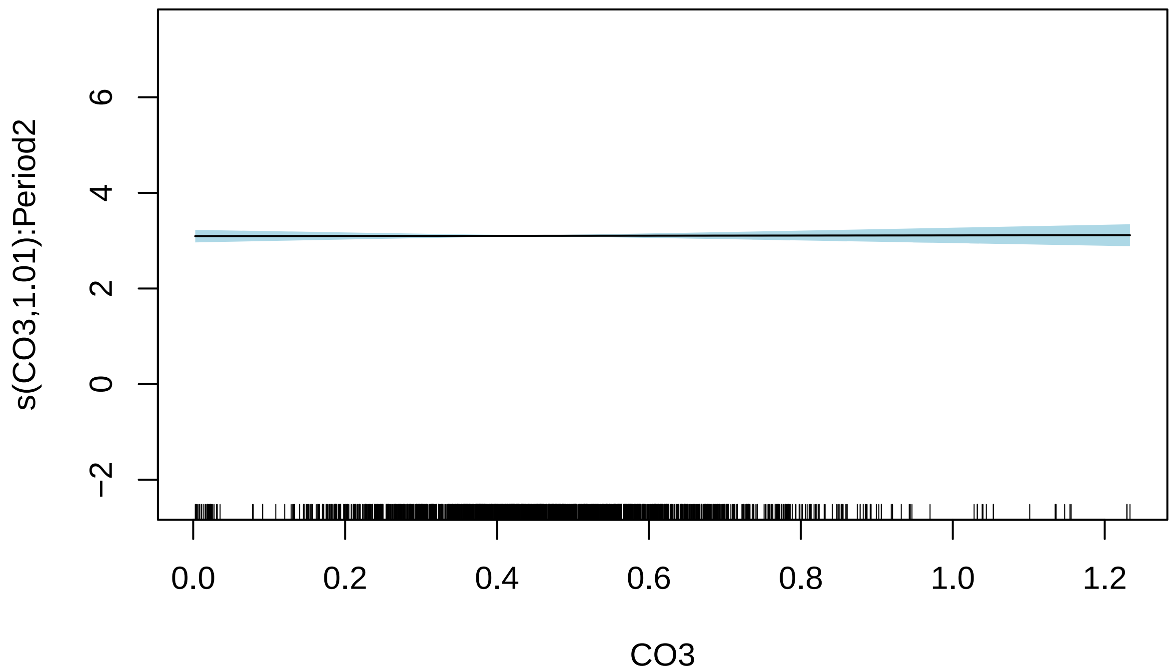

s(NO2,7.64):Period1
−2
0
2
4
6
0.000
0.005
0.010
0.015
0.020
NO2

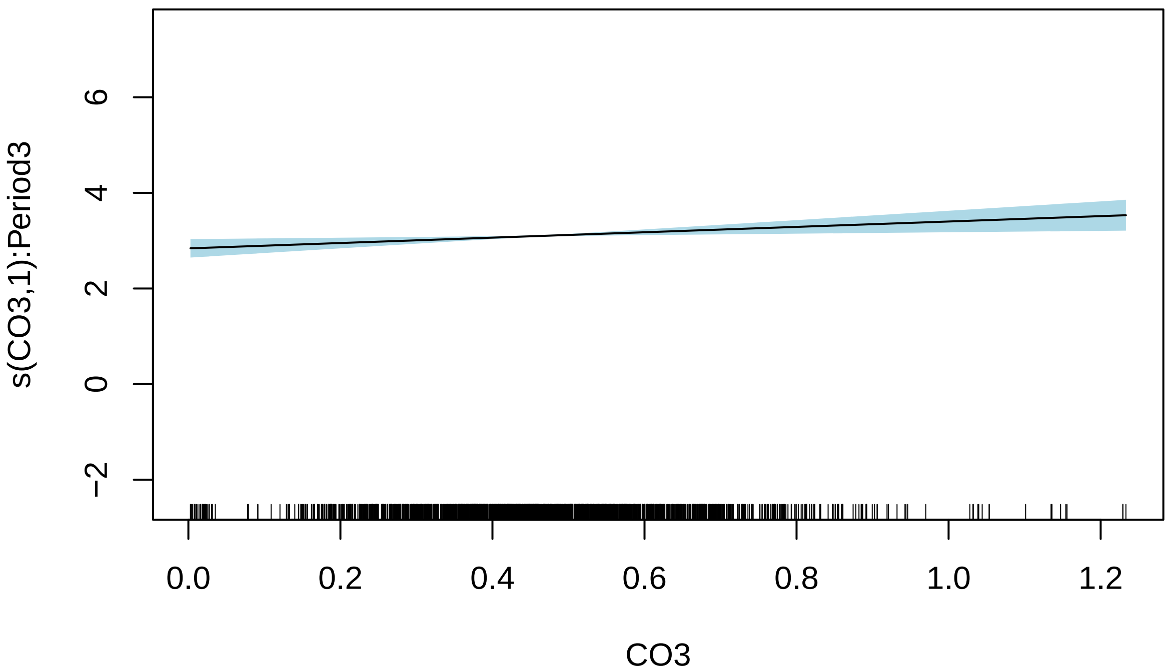


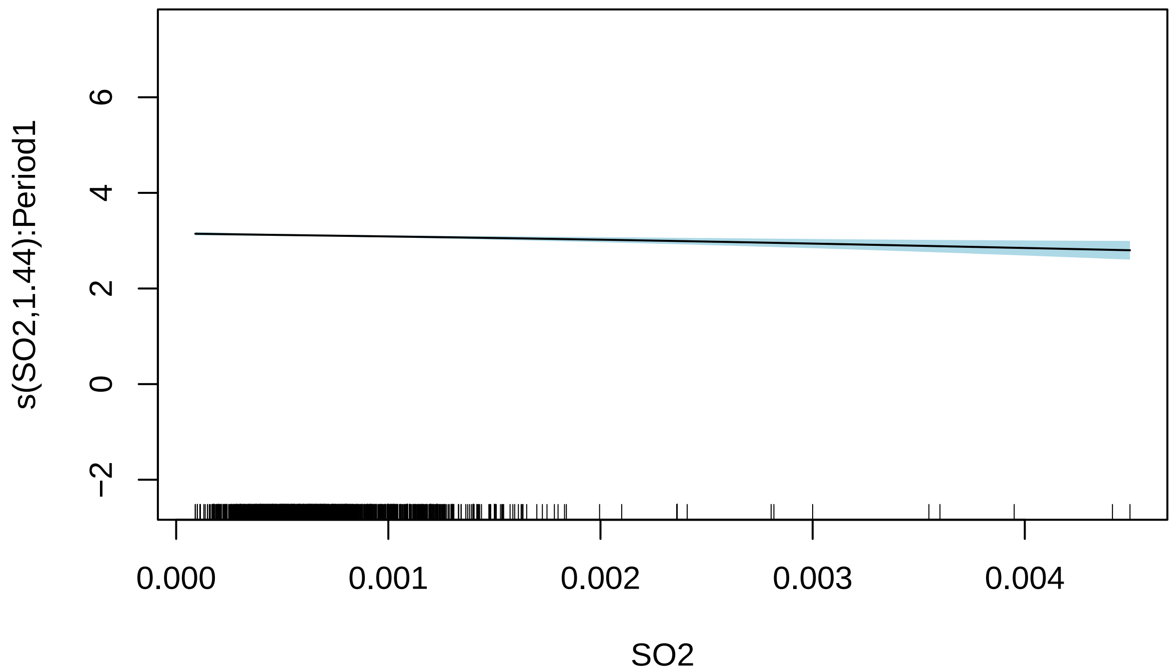
s(NO2,1):Period2
6
4
2
0
−2
0.000
0.005
0.010
0.015
0.020
NO2


s(NO2,1):Period3
6
4
2
0
−2
0.000
0.005
0.010
0.015
0.020
NO2

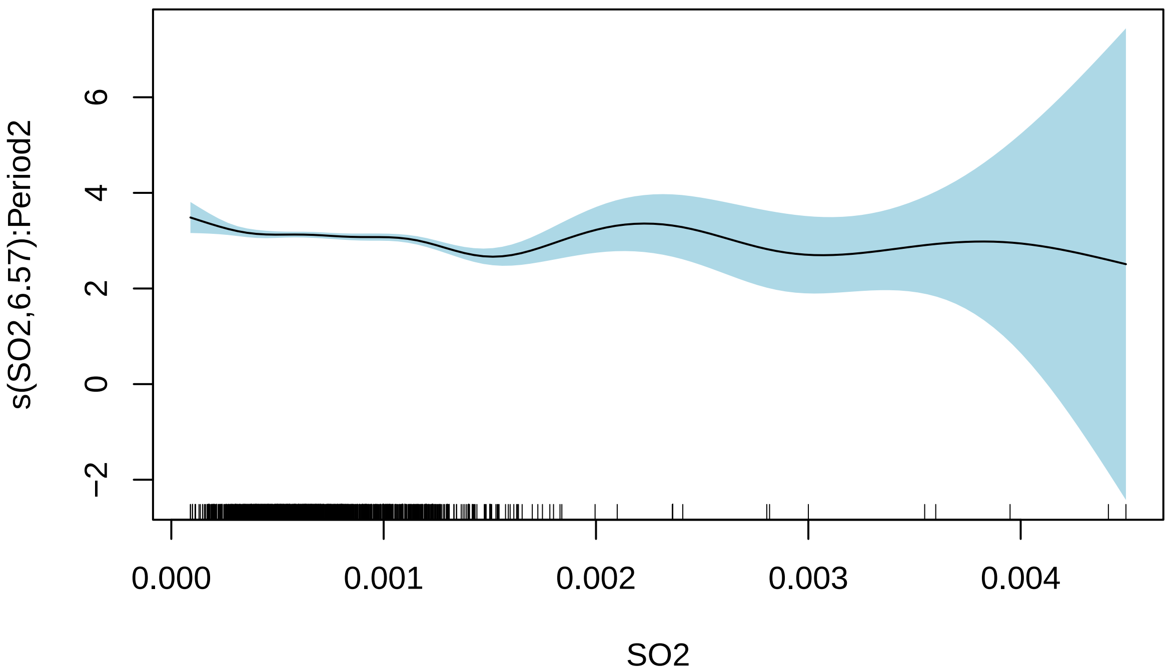

s(CO3,1.94):Period1
−2
0
2
4
6
0.0
0.2
0.4
0.6
0.8
1.0
1.2
CO3

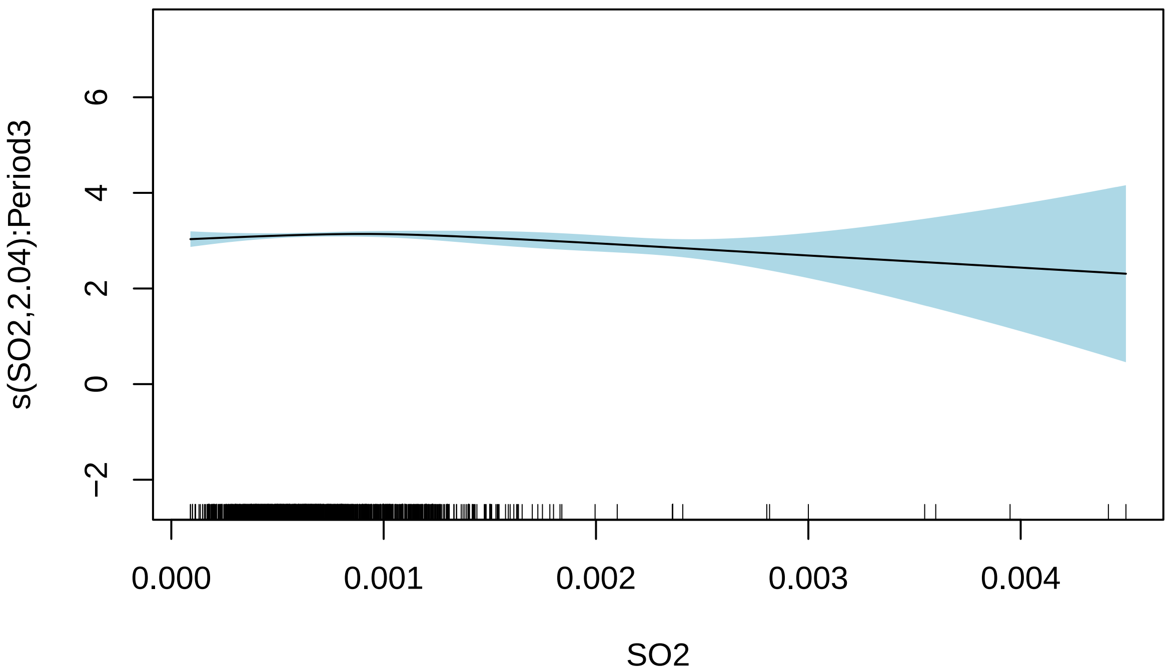

s(CO3,1.01):Period2
−2
0
2
4
6
0.0
0.2
0.4
0.6
0.8
1.0
1.2
CO3


s(CO3,1):Period3
−2
0
2
4
6
0.0
0.2
0.4
0.6
0.8
1.0
1.2
CO3

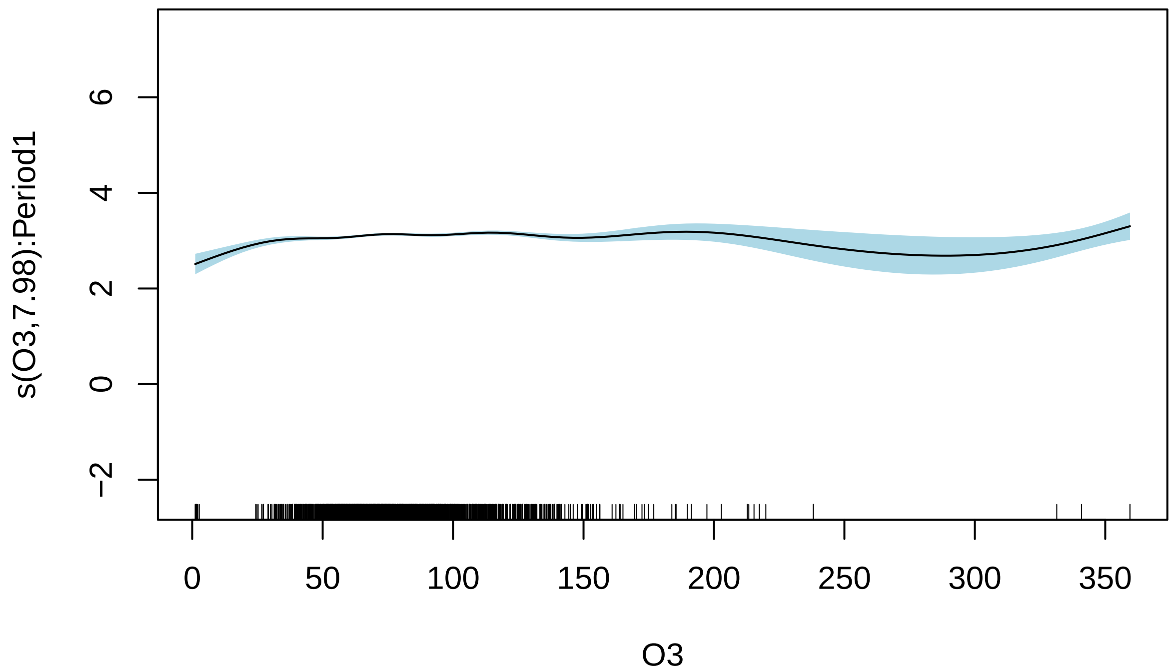


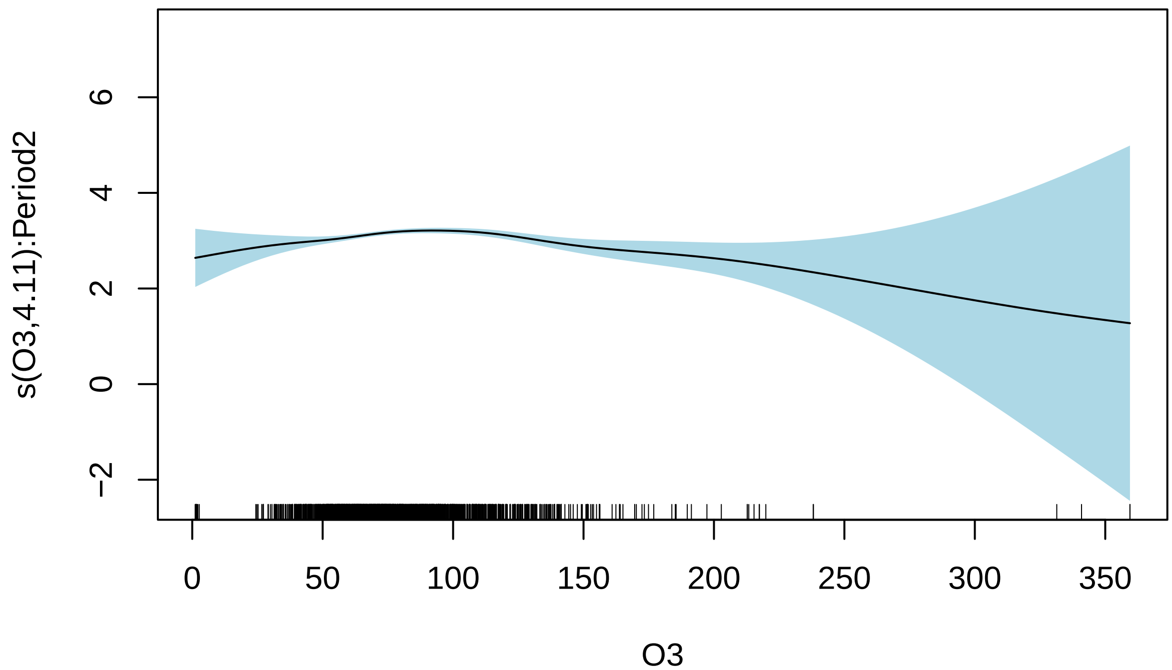
s(SO2,1.44):Period1
6
4
2
0
−2
0.000
0.001
0.002
0.003
0.004
SO2


s(SO2,6.57):Period2
−2
0
2
4
6
0.000
0.001
0.002
0.003
0.004
SO2

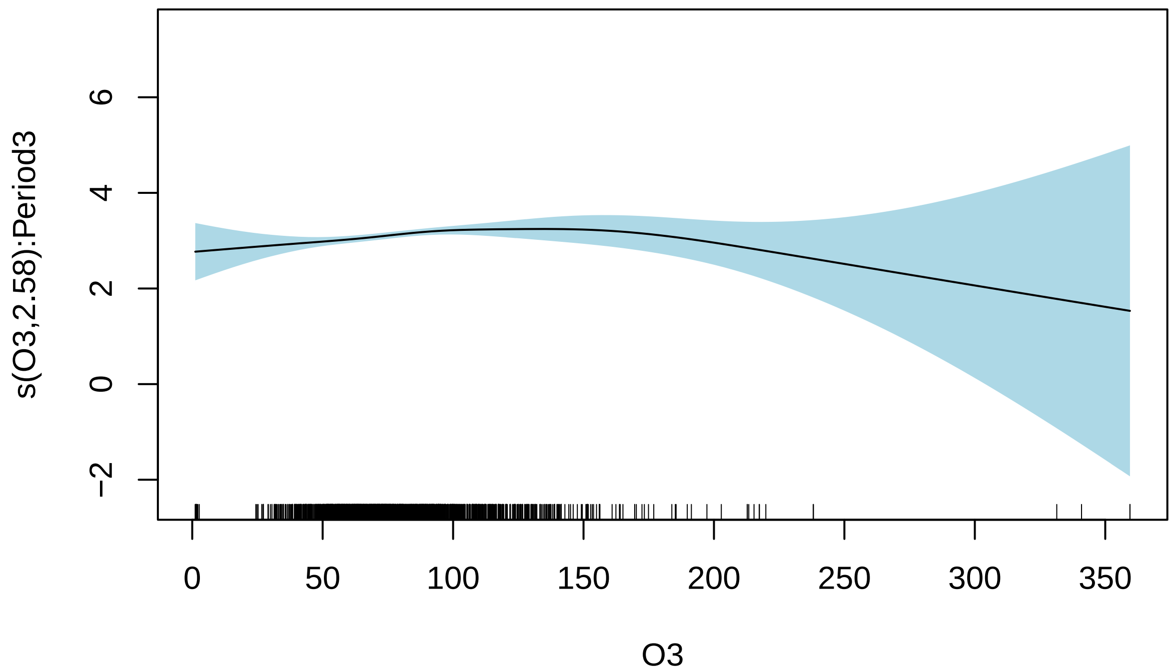


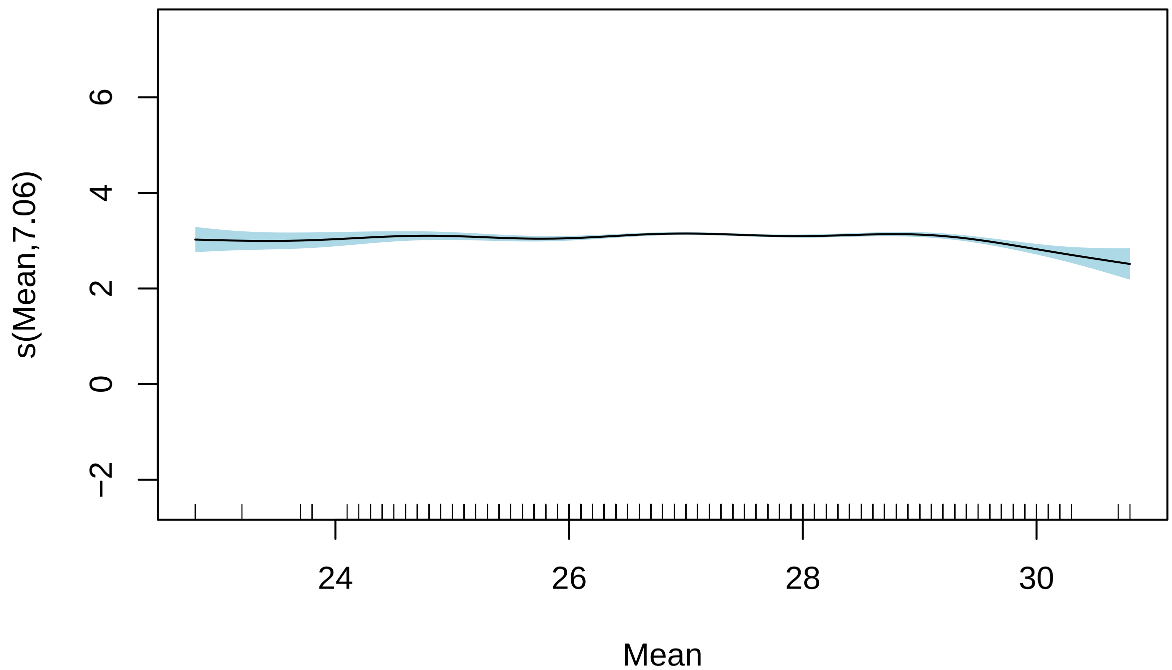
s(SO2,2.04):Period3
6
4
2
0
−2
0.000
0.001
0.002
0.003
0.004
SO2

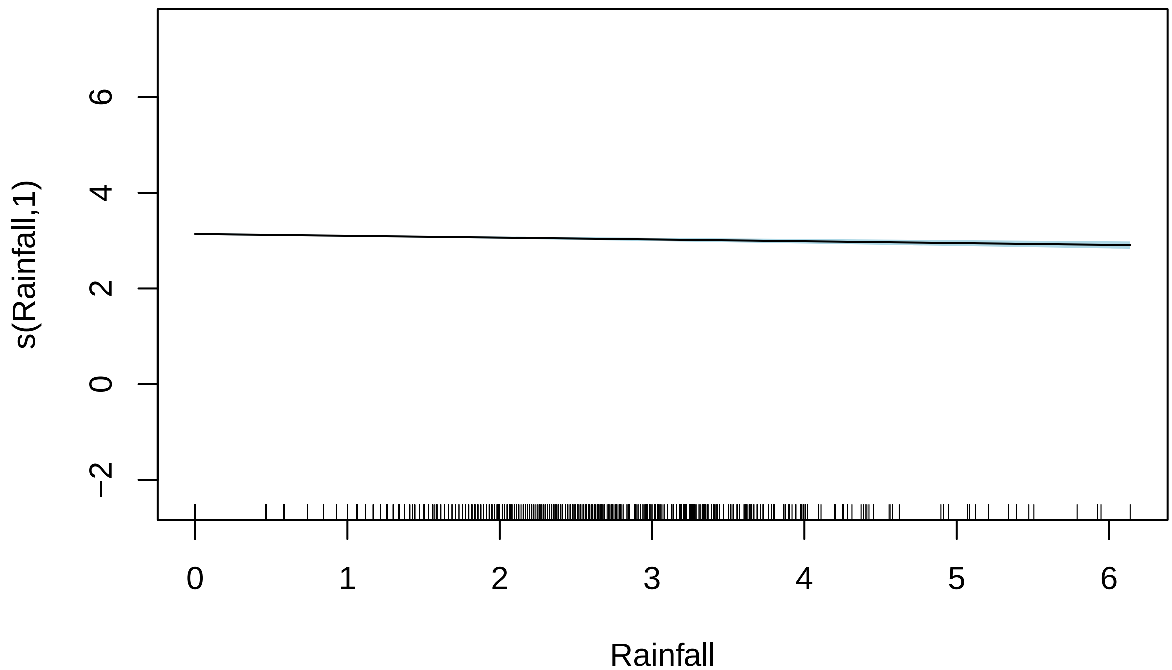

s(O3,7.98):Period1
−2
0
2
4
6
0
50
100
150
200
250
300
350
O3

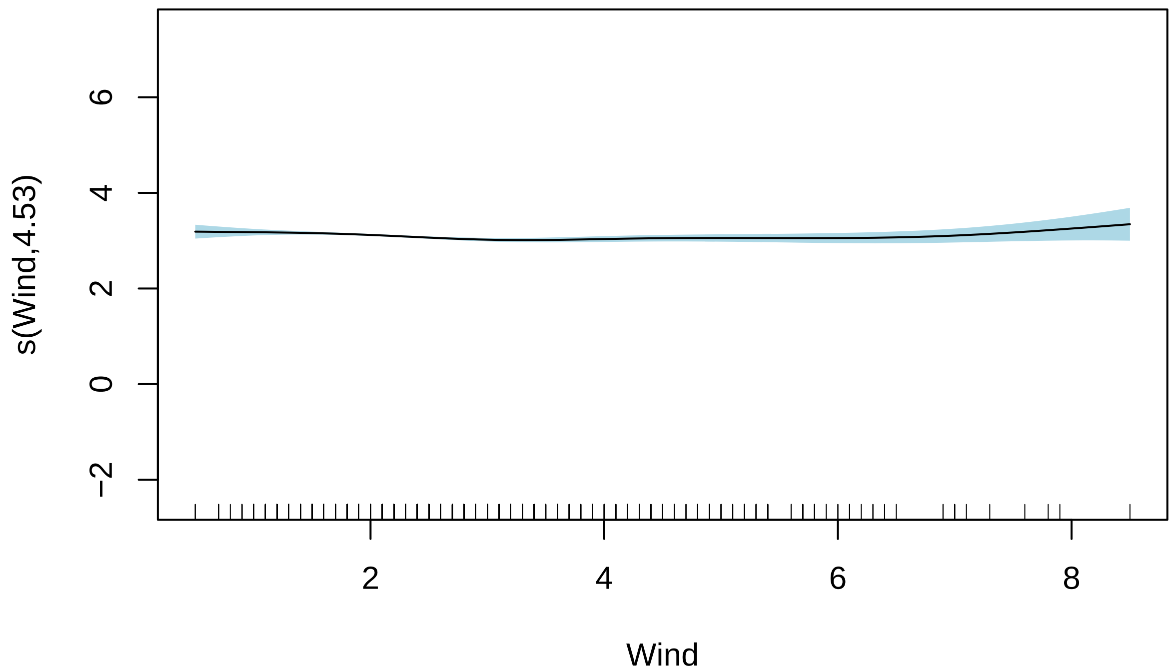


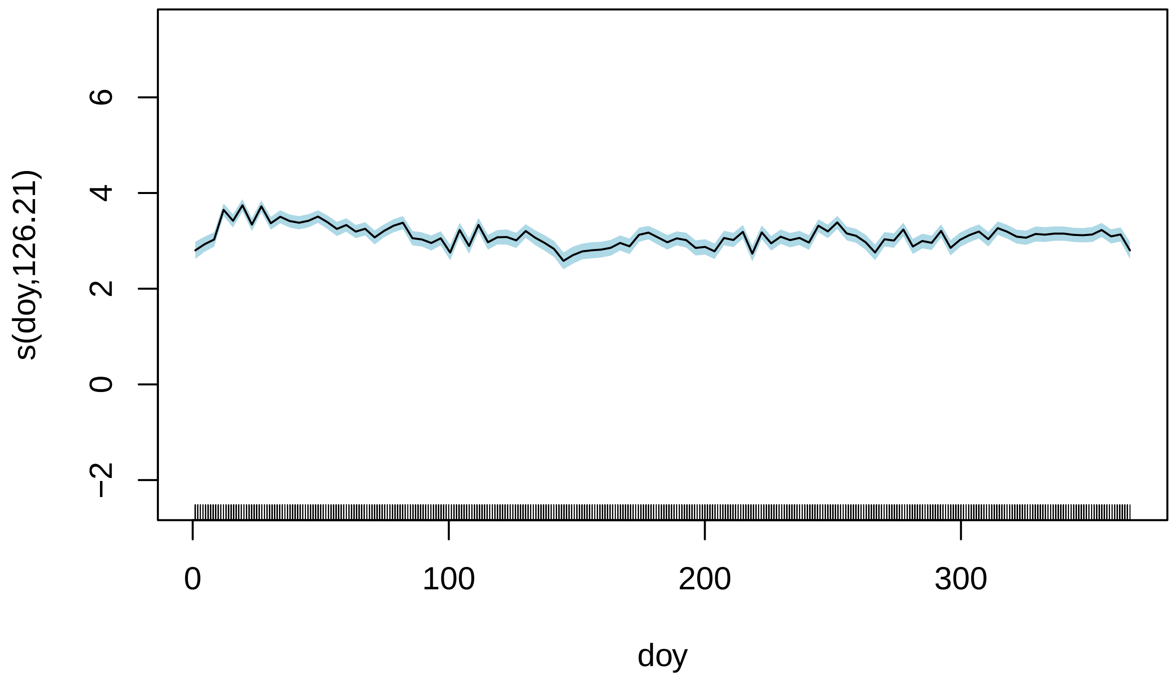
s(O3,4.11):Period2
−2
0
2
4
6
0
50
100
150
200
250
300
350
O3


s(O3,2.58):Period3
−2
0
2
4
6
0
50
100
150
200
250
300
350
O3

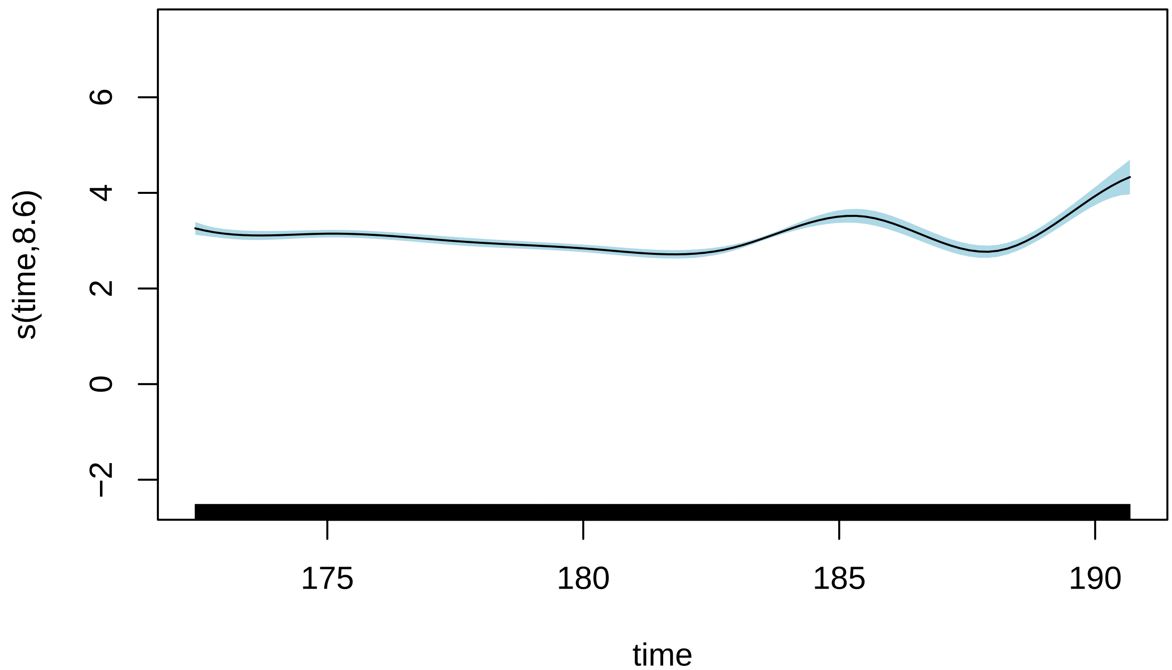

s(Mean,7.06)
6
4
2
0
−2
24
26
28
30
Mean


s(Rainfall,1)
6
4
2
0
−2
0
1
2
3
4
5
6
Rainfall

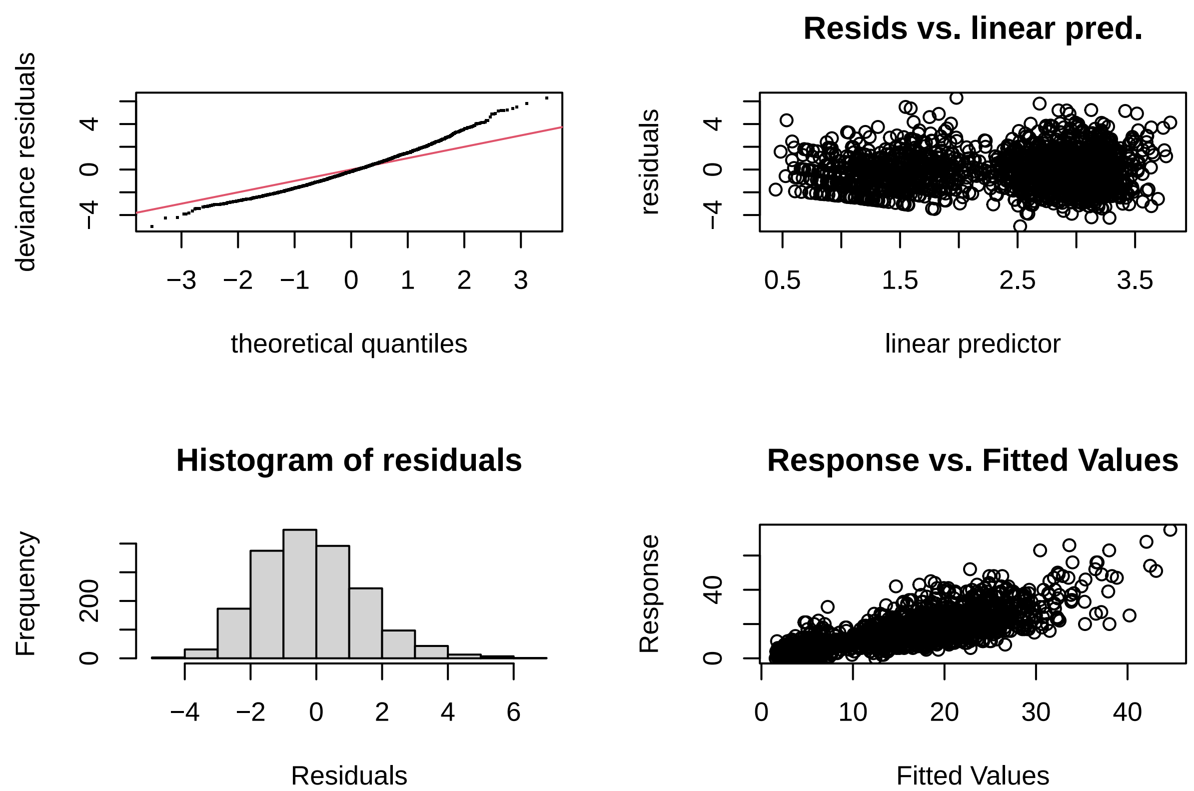

s(Wind,4.53)
−2
0
2
4
6
2
4
6
8
Wind



s(doy,126.21)
6
4
2
0
−2
0
100
200
300
doy




```
gam.check(kb_URTI_period)
```




```
##
## Method: REML   Optimizer: outer newton
## full convergence after 9 iterations.
## Gradient range [-0.001741251,0.0003148589]
## (score 6504.338 & scale 1).
## Hessian positive definite, eigenvalue range [3.671242e-05,15.96765].
## Model rank =  537 / 537
##
## Basis dimension (k) checking results. Low p-value (k-index<1) may
## indicate that k is too low, especially if edf is close to k'.
##
##                     k'   edf k-index p-value
## s(PM10):Period1   9.00  6.55    1.00   0.695
## s(PM10):Period2   9.00  4.72    1.00   0.690
## s(PM10):Period3   9.00  1.00    1.00   0.750
## s(NO2):Period1    9.00  7.64    1.00   0.690
## s(NO2):Period2    9.00  1.00    1.00   0.680
## s(NO2):Period3    9.00  1.00    1.00   0.745
## s(CO3):Period1    9.00  1.94    0.97   0.195
## s(CO3):Period2    9.00  1.01    0.97   0.205
## s(CO3):Period3    9.00  1.00    0.97   0.190
## s(SO2):Period1    9.00  1.44    0.95   0.055 .
## s(SO2):Period2    9.00  6.57    0.95   0.045 *
## s(SO2):Period3    9.00  2.04    0.95   0.035 *
## s(O3):Period1     9.00  7.98    1.04   0.985
## s(O3):Period2     9.00  4.11    1.04   0.975
```

```
## s(O3):Period3     9.00   2.58    1.04   0.990
## s(Mean)           9.00   7.06    0.98   0.275
## s(Rainfall)       9.00   1.00    0.91  <2e-16 ***
## s(Wind)           9.00   4.53    0.93  <2e-16 ***
## s(doy)          363.00 126.21    1.10   1.000
## s(time)           9.00   8.60    0.81  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

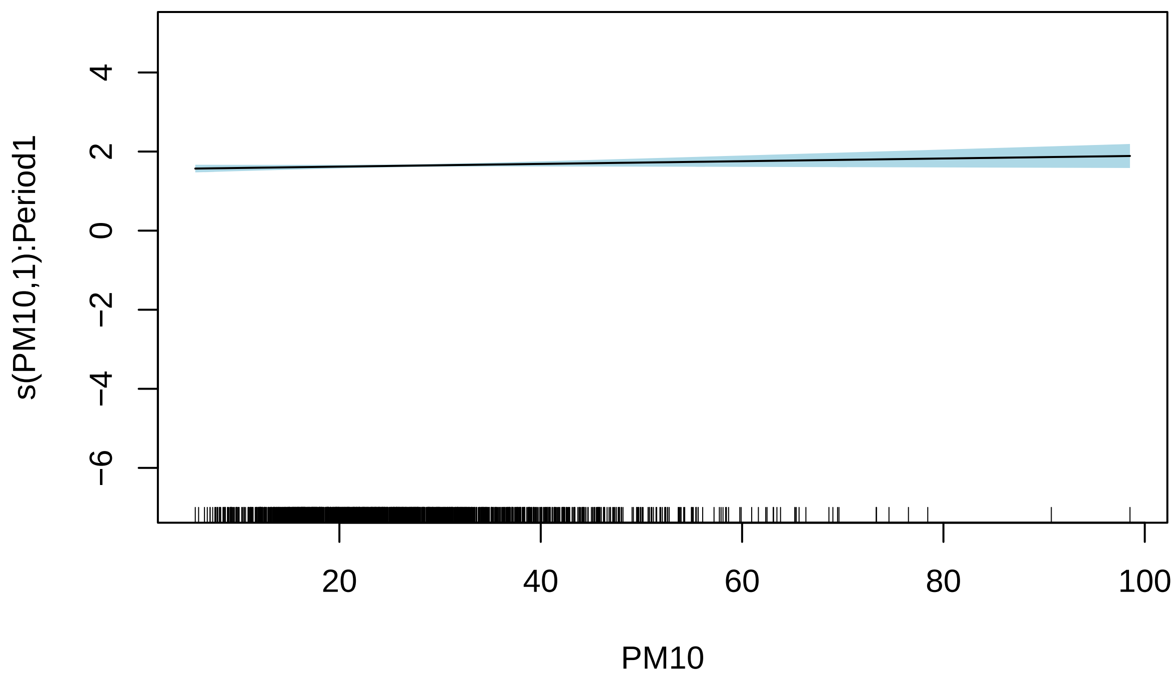
```r
kb_pneumo_period <- gam(Pneumonia ~ s(PM10, by=Period) + s(NO2, by=Period) + s(CO3, by=Period) + s(SO2,
summary(kb_pneumo_period)
```

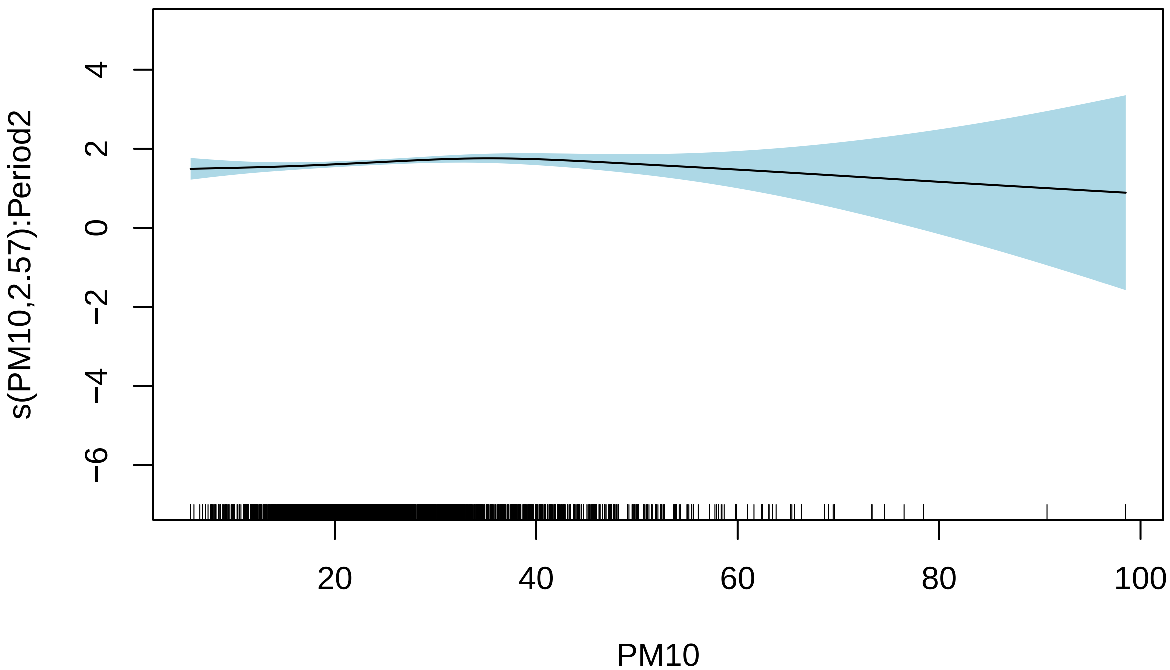
**KB influenza or pneumonia by period**

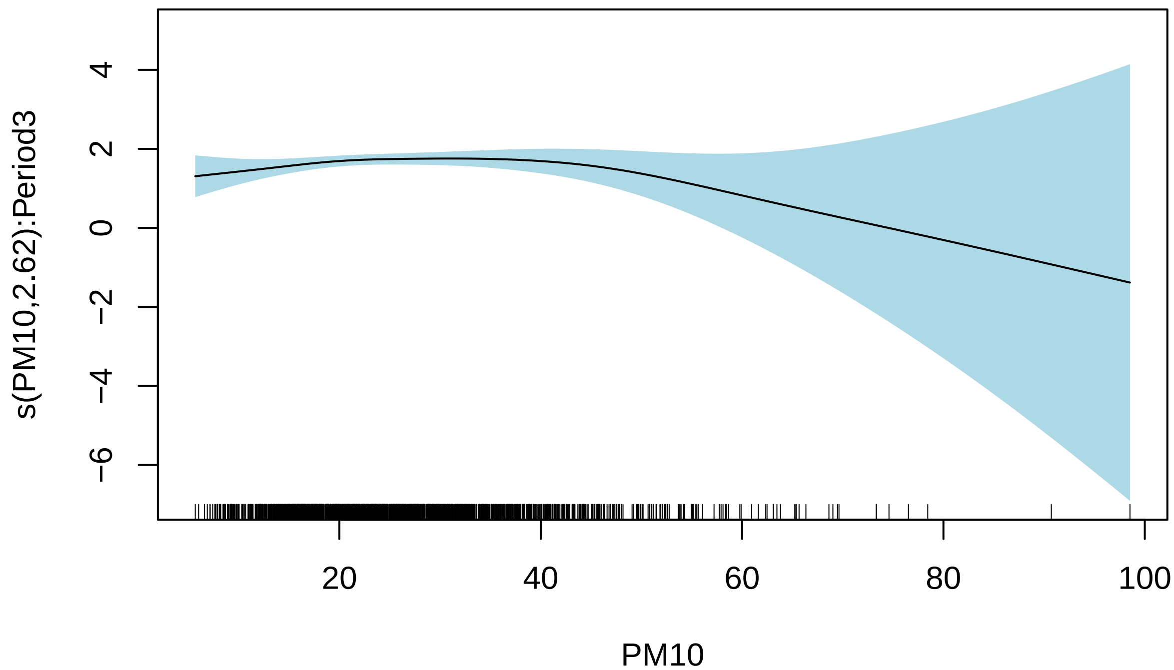
```
##
## Family: poisson
## Link function: log
##
## Formula:
## Pneumonia ~ s(PM10, by = Period) + s(NO2, by = Period) + s(CO3,
##     by = Period) + s(SO2, by = Period) + s(O3, by = Period) +
##     s(Mean) + s(Rainfall) + s(Wind) + s(doy, bs = "cc", k = 365) +
##     s(time, bs = "bs") + Period
##
## Parametric coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  1.63990    0.08095  20.258   <2e-16 ***
## Period2     -1.99913    0.21020  -9.511   <2e-16 ***
## Period3     -2.67951    0.27797  -9.640   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Approximate significance of smooth terms:
##                    edf  Ref.df  Chi.sq  p-value
## s(PM10):Period1  1.000   1.001   2.643   0.1041
## s(PM10):Period2  2.573   3.254   5.589   0.1407
## s(PM10):Period3  2.617   3.287   4.726   0.2276
## s(NO2):Period1   1.001   1.001   0.647   0.4215
## s(NO2):Period2   1.695   2.072   2.182   0.3560
## s(NO2):Period3   2.844   3.207   9.858   0.0227 *
## s(CO3):Period1   2.465   3.146   3.532   0.3141
## s(CO3):Period2   1.000   1.001   0.077   0.7825
## s(CO3):Period3   1.204   1.369   4.638   0.0423 *
## s(SO2):Period1   1.001   1.001   2.117   0.1460
## s(SO2):Period2   1.001   1.002   0.264   0.6080
## s(SO2):Period3   1.000   1.001   0.210   0.6473
## s(O3):Period1    4.719   5.793  33.960 6.38e-06 ***
## s(O3):Period2    2.036   2.588   1.502   0.4959
## s(O3):Period3    1.001   1.001   1.713   0.1909
## s(Mean)          3.106   3.957   9.026   0.0567 .
## s(Rainfall)      1.000   1.000   0.028   0.8675
```

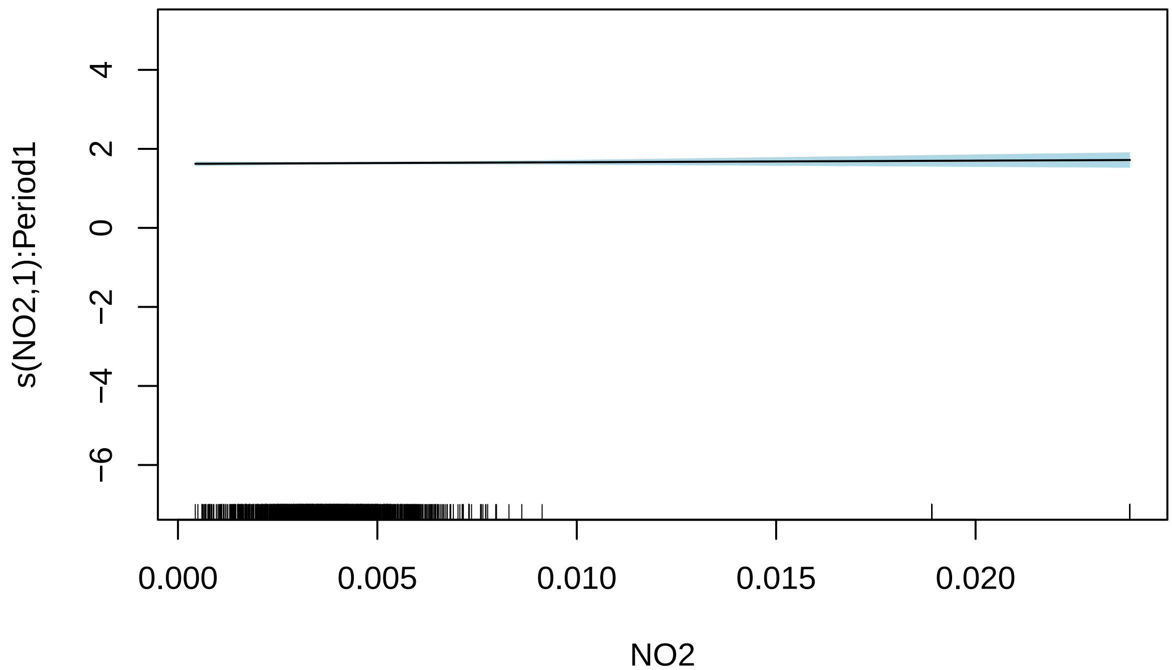
```
## s(Wind)          1.137   1.260   0.662   0.4484
## s(doy)          11.036 363.000  34.106 6.00e-05 ***
## s(time)          7.638   8.342 373.650  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-sq.(adj) =  0.268   Deviance explained = 29.8%
## -REML = 3540.9  Scale est. = 1         n = 1827
```

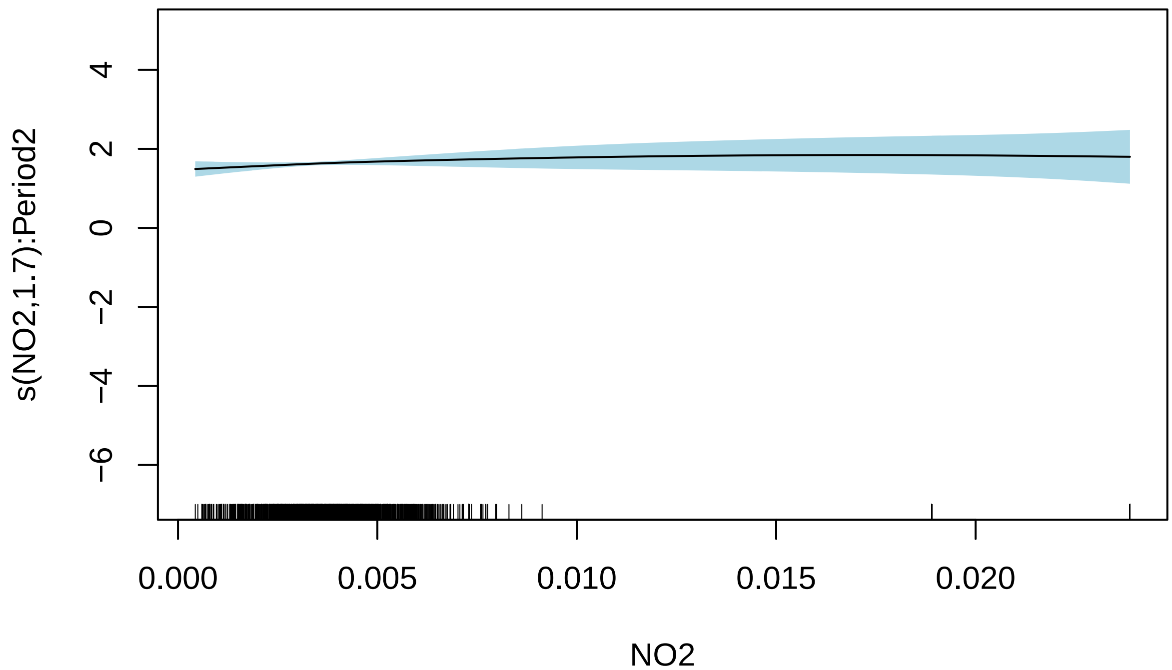
```
plot.gam(kb_pneumo_period, shade = TRUE, shade.col = "lightblue", shift = coef(kb_pneumo_period)[1],  s
```



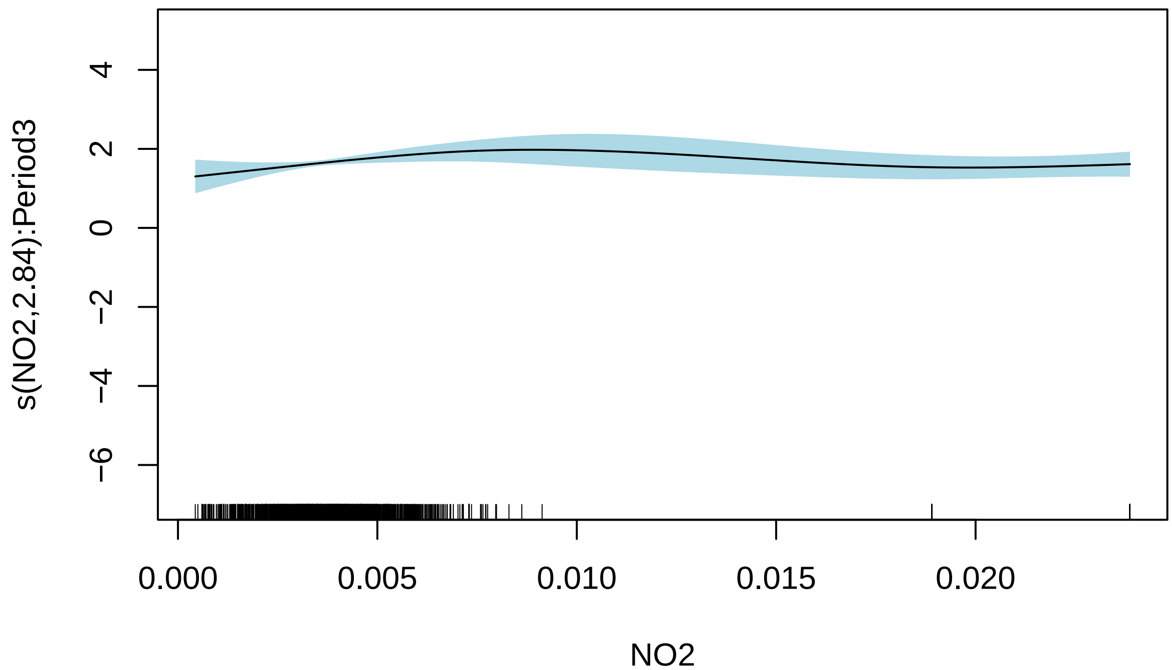

s(PM10,2.57):Period2
4
2
0
−2
−4
−6
20
40
60
80
100
PM10

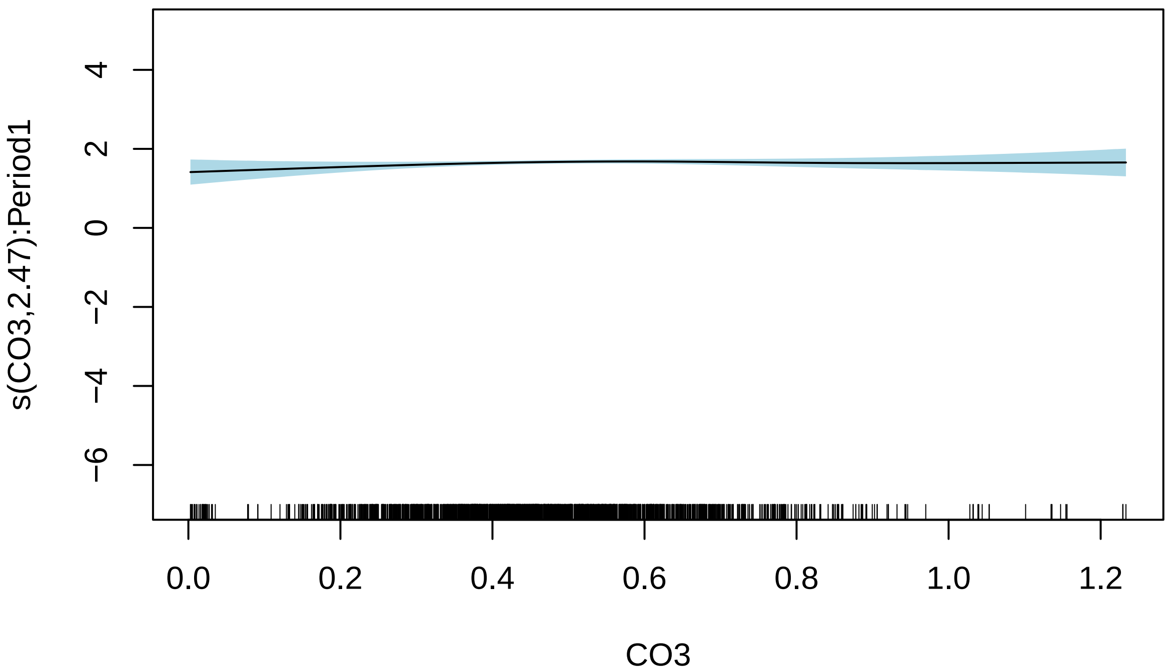

s(PM10,2.62):Period3
4
2
0
−2
−4
−6
20
40
60
80
100
PM10


s(NO2,1):Period1
4
2
0
−2
−4
−6
0.000
0.005
0.010
0.015
0.020
NO2

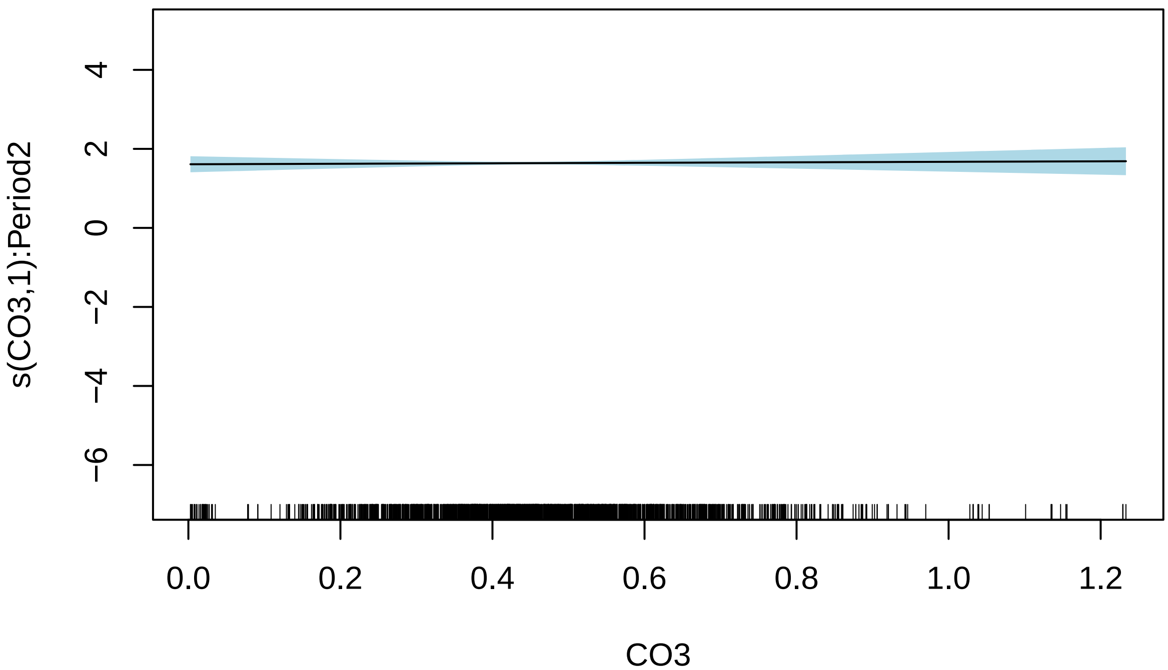


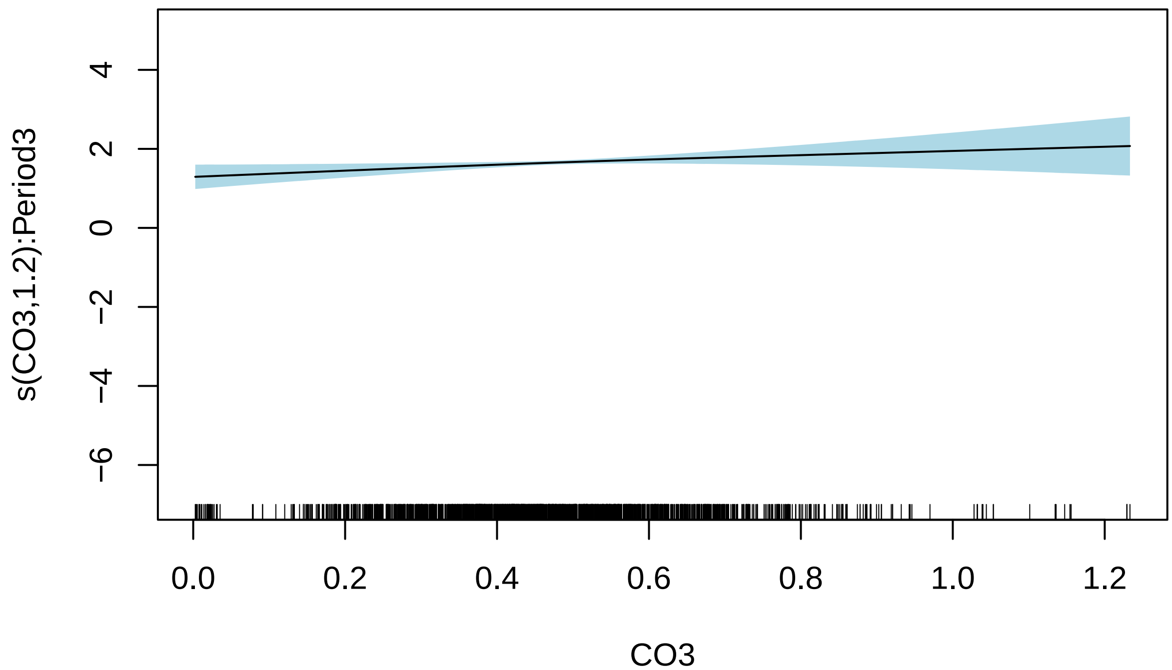
s(NO2,1.7):Period2
−6
−4
−2
0
2
4
0.000
0.005
0.010
0.015
0.020
NO2

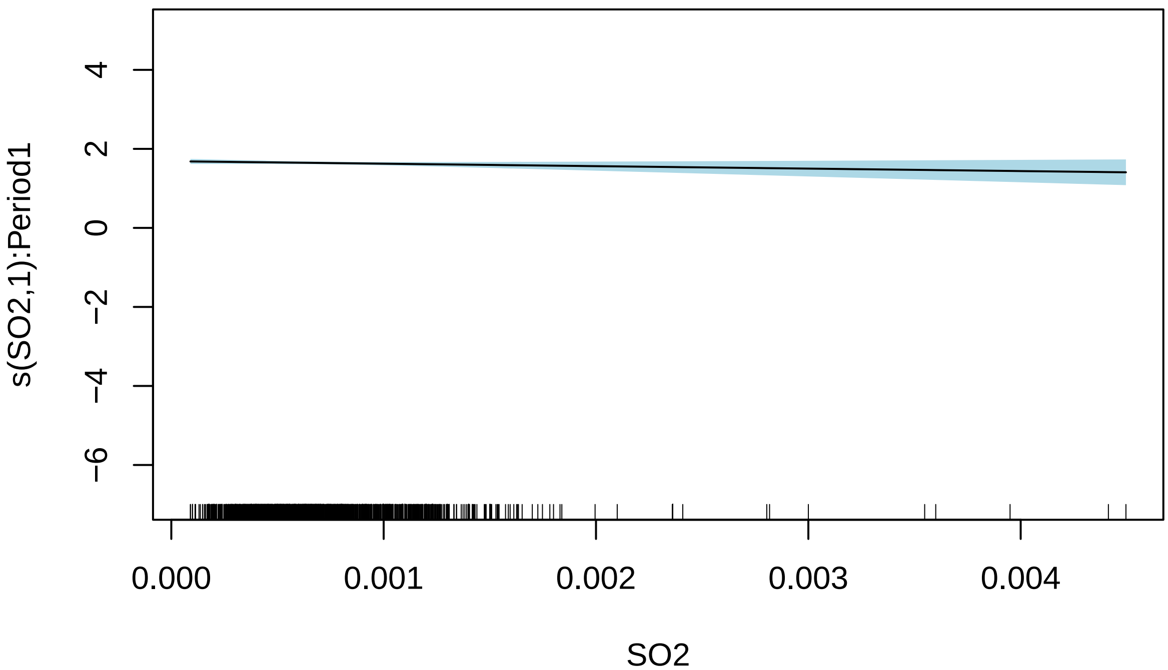

s(NO2,2.84):Period3
4
2
0
−2
−4
−6
0.000
0.005
0.010
0.015
0.020
NO2

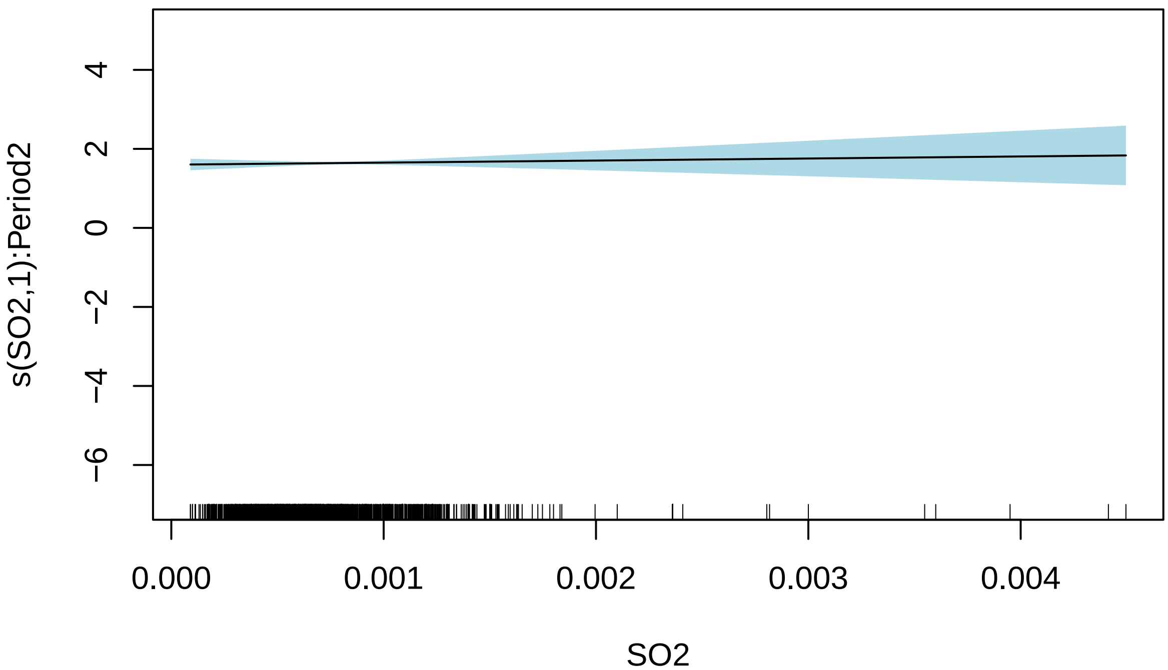


s(CO3,2.47):Period1
4
2
0
−2
−4
−6
0.0
0.2
0.4
0.6
0.8
1.0
1.2
CO3

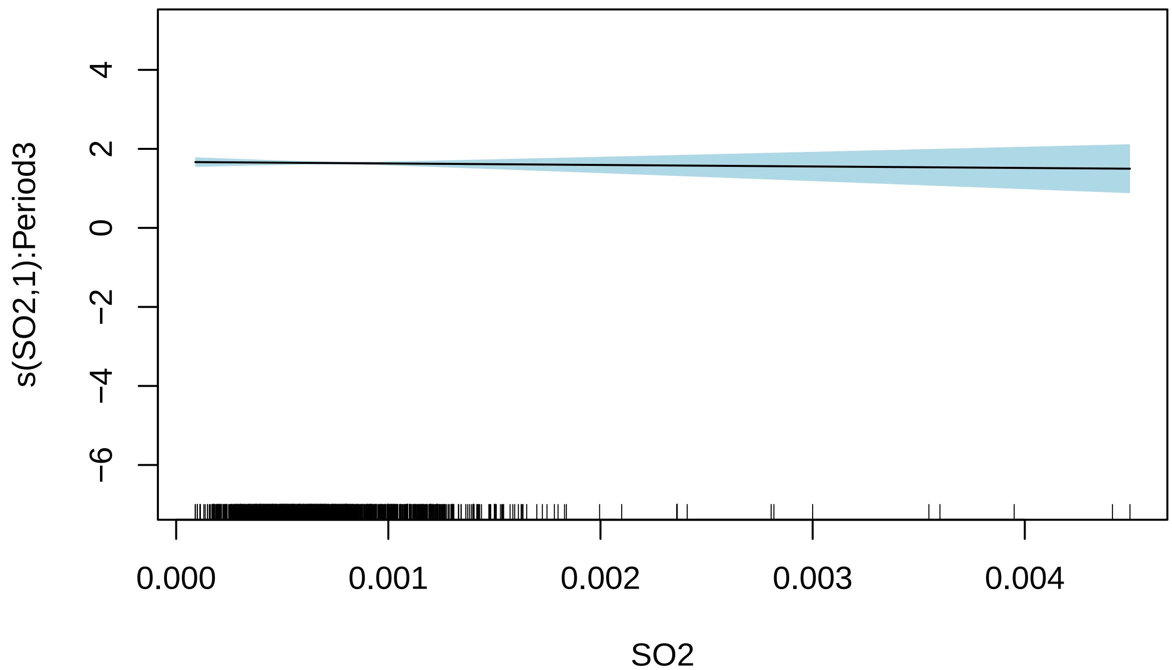

s(CO3,1):Period2
4
2
0
−2
−4
−6
0.0
0.2
0.4
0.6
0.8
1.0
1.2
CO3


s(CO3,1.2):Period3
4
2
0
−2
−4
−6
0.0
0.2
0.4
0.6
0.8
1.0
1.2
CO3

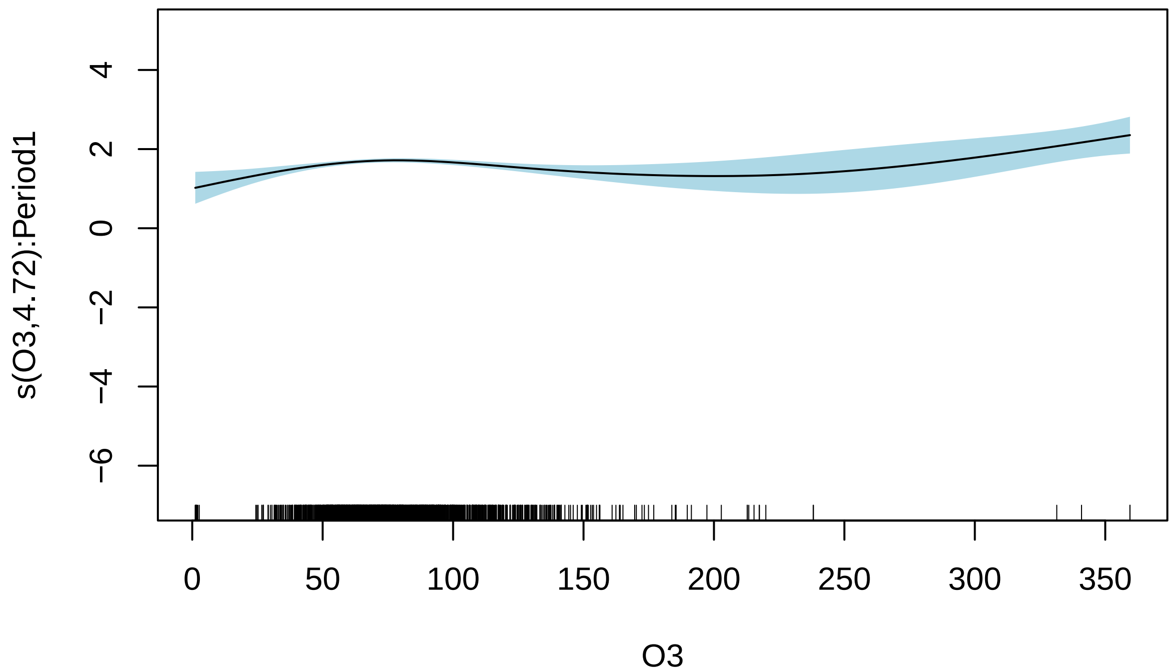

s(SO2,1):Period1
4
2
0
−2
−4
−6
0.000
0.001
0.002
0.003
0.004
SO2

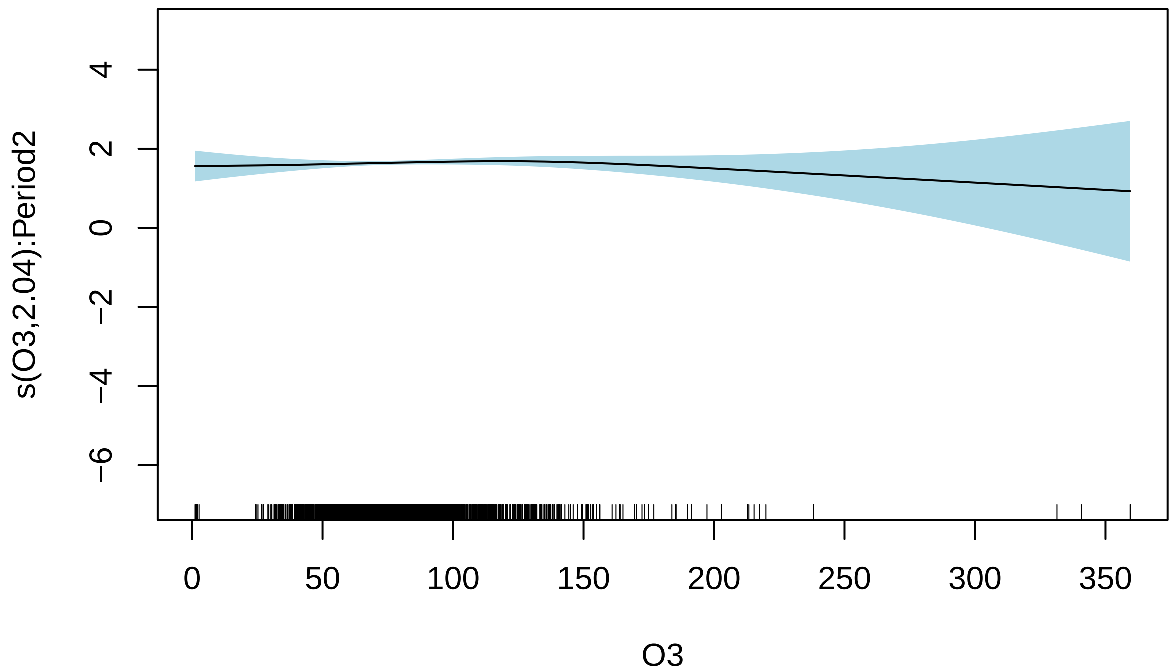


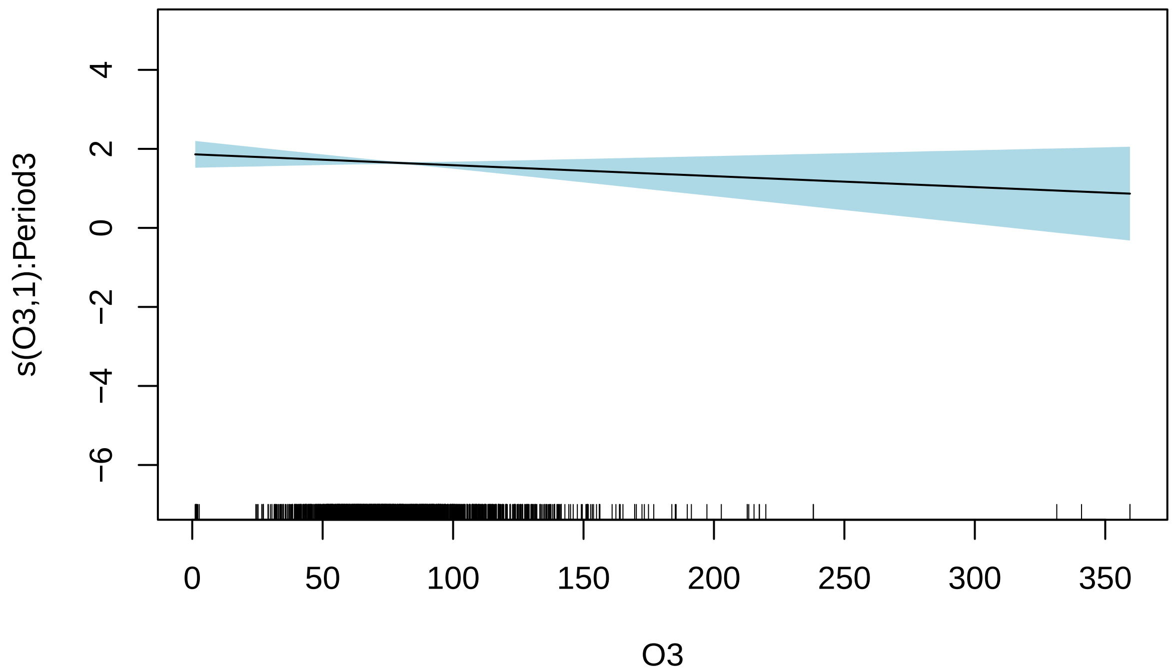
4
2
0
−2
−4
−6
s(SO2,1):Period2
0.000
0.001
0.002
0.003
0.004
SO2



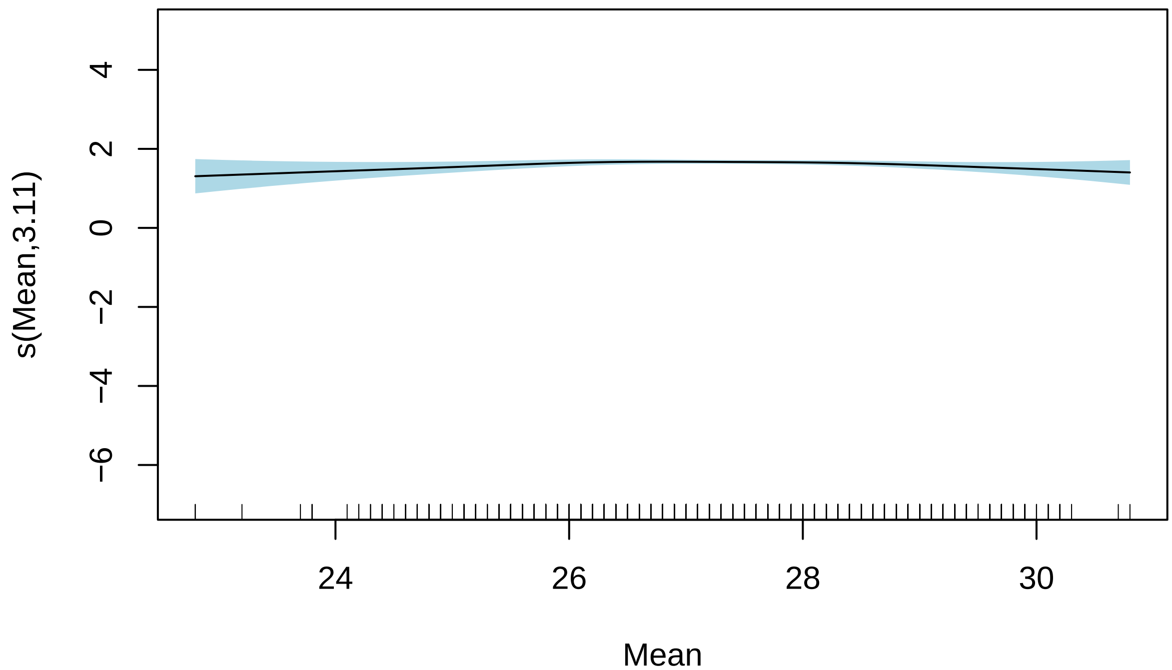
4
2
0
−2
−4
−6
s(SO2,1):Period3
0.000
0.001
0.002
0.003
0.004
SO2


s(O3,4.72):Period1
4
2
0
−2
−4
−6
0
50
100
150
200
250
300
350
O3

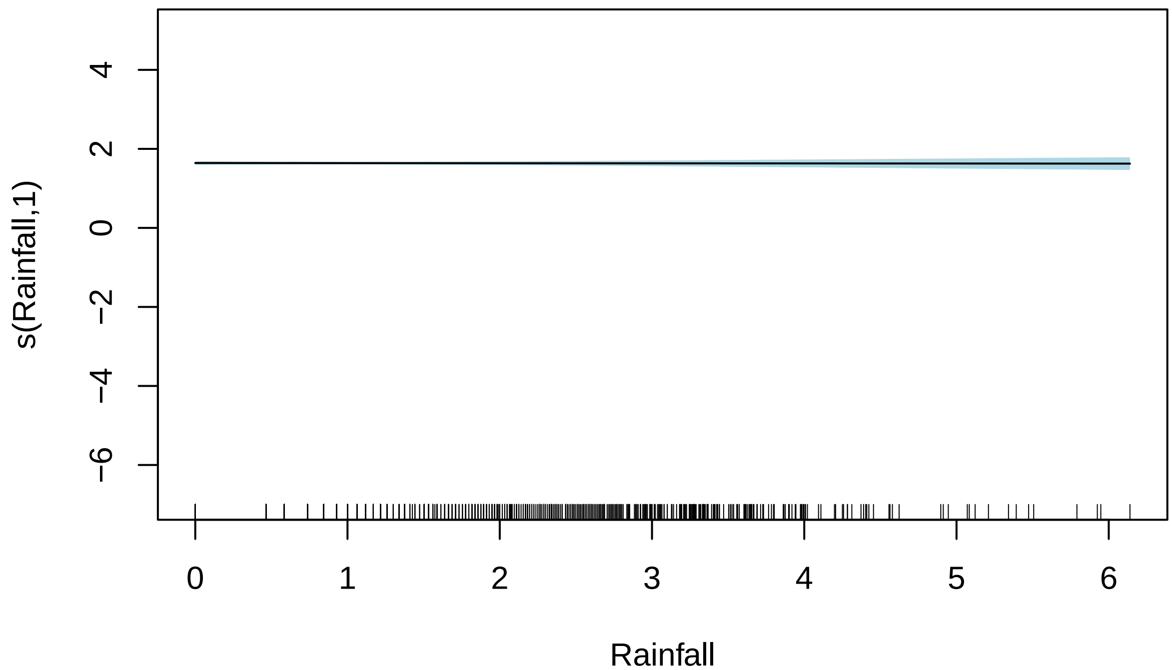

s(O3,2.04):Period2
4
2
0
−2
−4
−6
0
50
100
150
200
250
300
350
O3

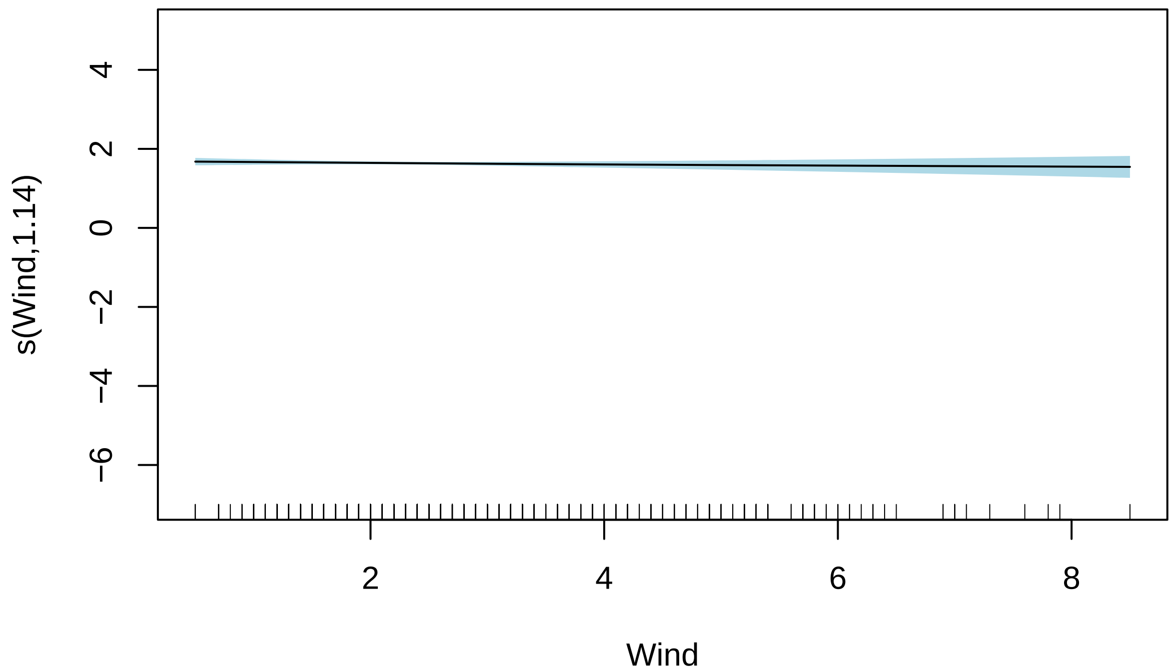


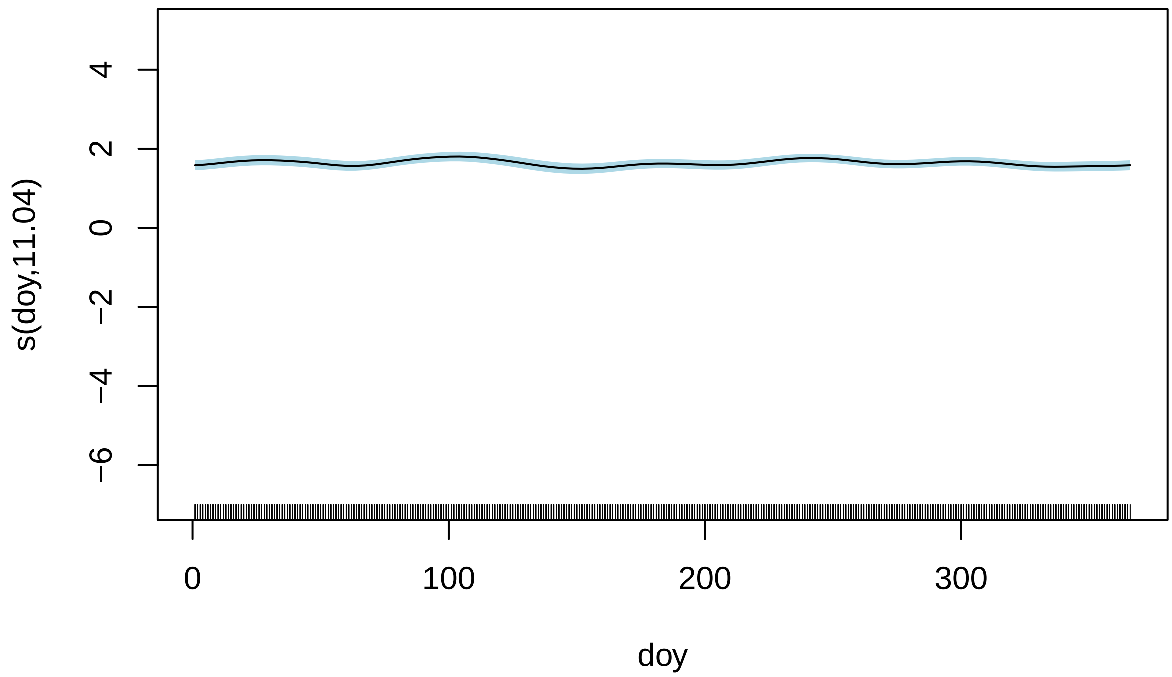
s(O3,1):Period3
4
2
0
−2
−4
−6
0
50
100
150
200
250
300
350
O3

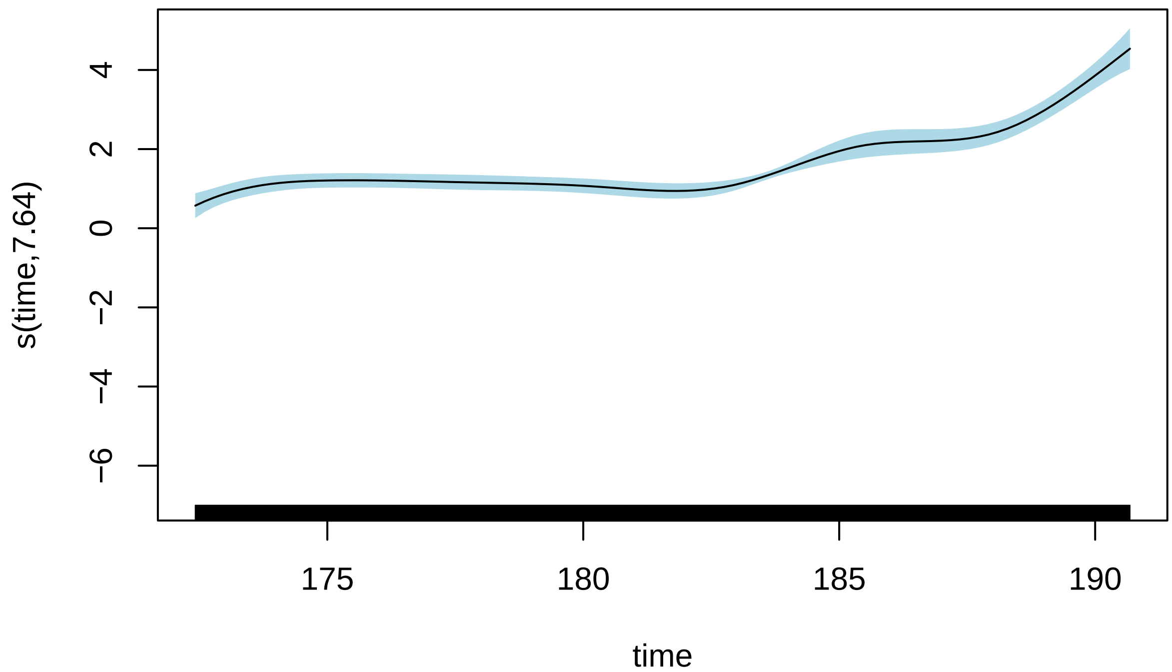


s(Mean,3.11)
4
2
0
−2
−4
−6
24
26
28
30
Mean

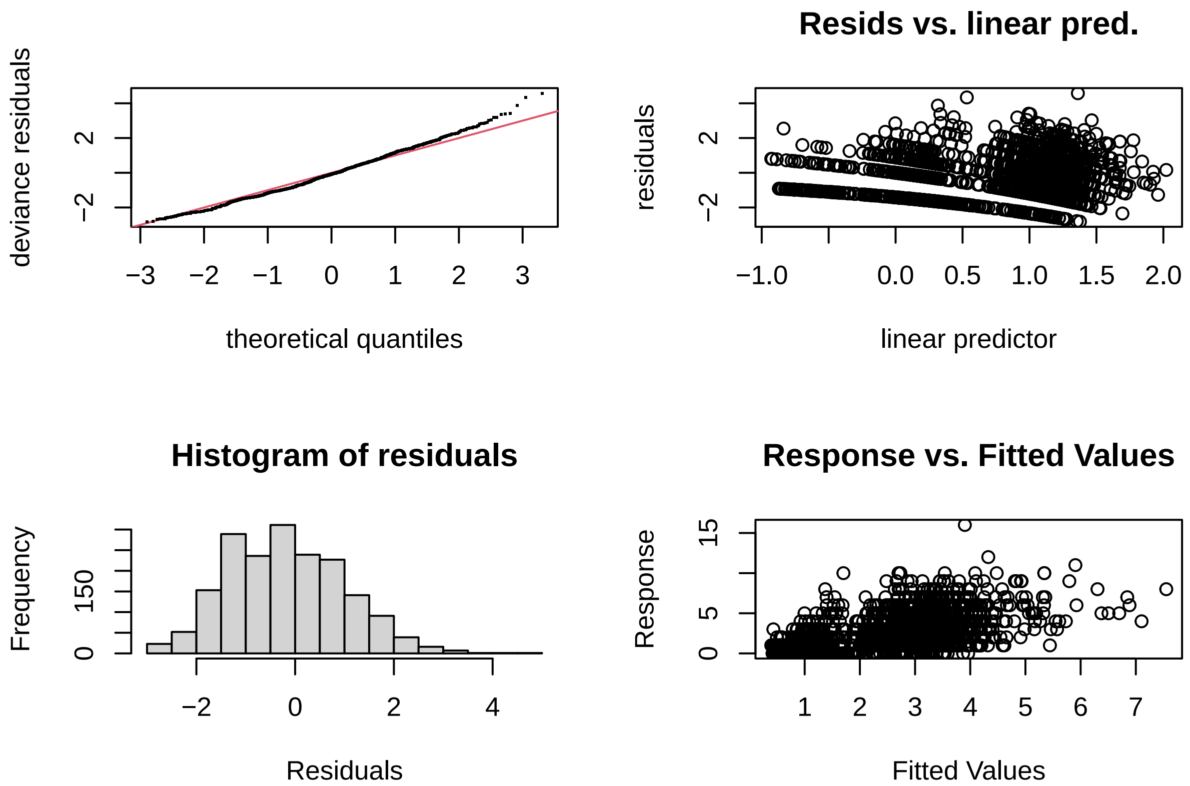

s(Rainfall,1)
4
2
0
−2
−4
−6
0
1
2
3
4
5
6
Rainfall


s(Wind,1.14)
4
2
0
−2
−4
−6
2
4
6
8
Wind


s(doy,11.04)
4
2
0
−2
−4
−6
0
100
200
300
doy




```
gam.check(kb_pneumo_period)
```




```
##
## Method: REML   Optimizer: outer newton
## full convergence after 13 iterations.
## Gradient range [-0.0003224411,5.442697e-05]
## (score 3540.93 & scale 1).
## Hessian positive definite, eigenvalue range [1.806098e-05,2.384548].
## Model rank =  537 / 537
##
## Basis dimension (k) checking results. Low p-value (k-index<1) may
## indicate that k is too low, especially if edf is close to k'.
##
##                      k'   edf k-index p-value
## s(PM10):Period1    9.00  1.00    0.99   0.645
## s(PM10):Period2    9.00  2.57    0.99   0.650
## s(PM10):Period3    9.00  2.62    0.99   0.710
## s(NO2):Period1     9.00  1.00    0.95   0.135
## s(NO2):Period2     9.00  1.70    0.95   0.150
## s(NO2):Period3     9.00  2.84    0.95   0.140
## s(CO3):Period1     9.00  2.47    0.94   0.070 .
## s(CO3):Period2     9.00  1.00    0.94   0.055 .
## s(CO3):Period3     9.00  1.20    0.94   0.035 *
## s(SO2):Period1     9.00  1.00    0.93   0.015 *
## s(SO2):Period2     9.00  1.00    0.93   0.025 *
## s(SO2):Period3     9.00  1.00    0.93   0.030 *
## s(O3):Period1      9.00  4.72    0.99   0.655
## s(O3):Period2      9.00  2.04    0.99   0.710
```

```
## s(O3):Period3     9.00   1.00    0.99   0.745
## s(Mean)           9.00   3.11    0.94   0.060 .
## s(Rainfall)       9.00   1.00    0.89  <2e-16 ***
## s(Wind)           9.00   1.14    0.94   0.035 *
## s(doy)          363.00  11.04    1.01   0.955
## s(time)           9.00   7.64    0.88  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

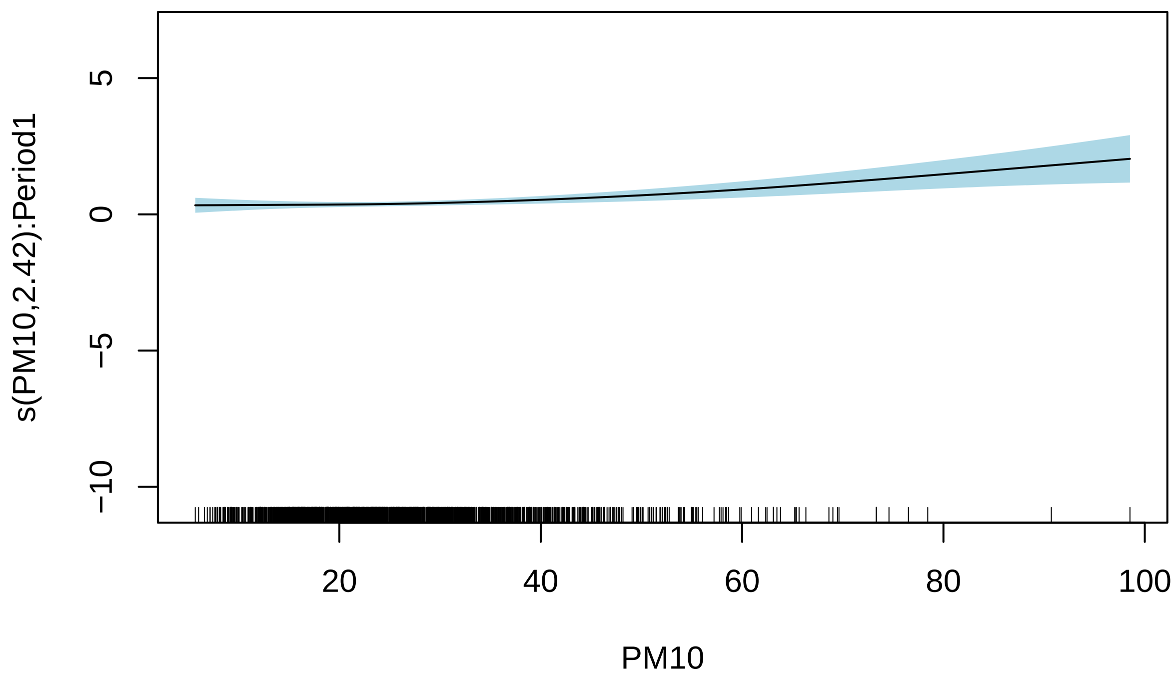
```
kb_CLD_period <- gam(CLD ~ s(PM10, by=Period) + s(NO2, by=Period) + s(CO3, by=Period) + s(SO2, by=Perio
summary(kb_CLD_period)
```

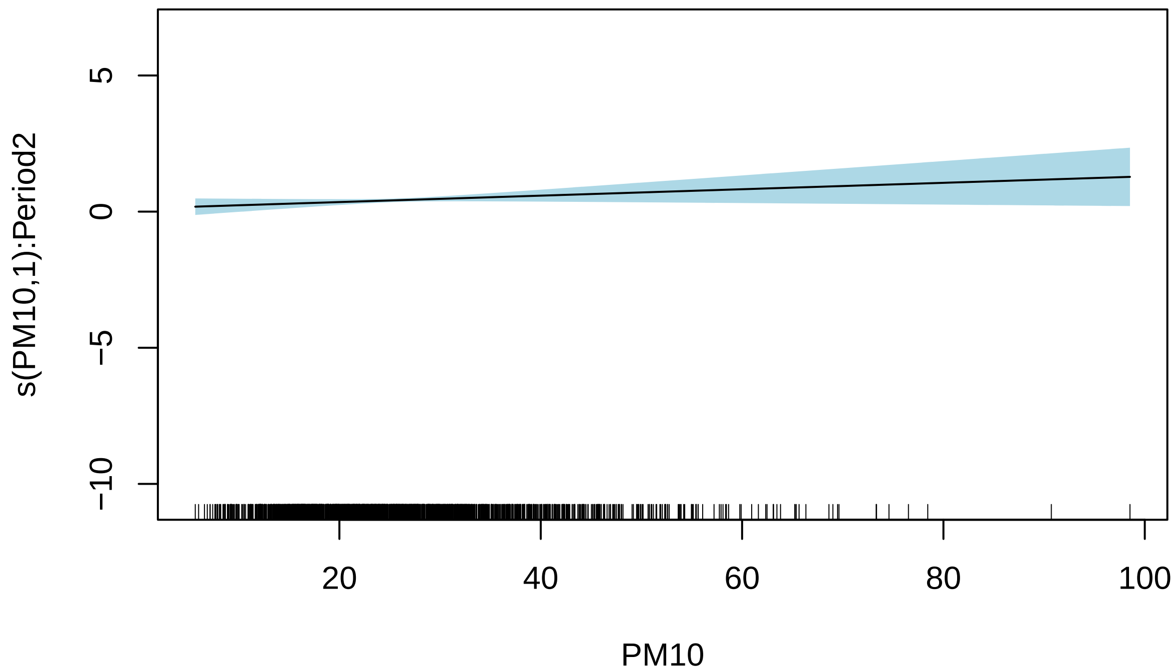
### KB CLD by period

```
##
## Family: poisson
## Link function: log
##
## Formula:
## CLD ~ s(PM10, by = Period) + s(NO2, by = Period) + s(CO3, by = Period) +
##     s(SO2, by = Period) + s(O3, by = Period) + s(Mean) + s(Rainfall) +
##     s(Wind) + s(doy, bs = "cc", k = 365) + s(time, bs = "bs") +
##     Period
##
## Parametric coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)   0.4225     0.1487   2.842  0.00448 **
## Period2      -2.2168     0.3881  -5.712 1.12e-08 ***
## Period3      -3.7331     0.5320  -7.017 2.27e-12 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Approximate significance of smooth terms:
##                    edf  Ref.df  Chi.sq  p-value
## s(PM10):Period1 2.415   3.060  16.705 0.000880 ***
## s(PM10):Period2 1.000   1.001   2.576 0.108569
## s(PM10):Period3 1.993   2.525   2.075 0.433007
## s(NO2):Period1  2.277   2.806   4.992 0.238494
## s(NO2):Period2  1.000   1.001   2.339 0.126219
## s(NO2):Period3  1.228   1.421   0.094 0.914714
## s(CO3):Period1  1.001   1.001   7.545 0.006039 **
## s(CO3):Period2  1.000   1.001   4.639 0.031334 *
## s(CO3):Period3  1.001   1.003   0.046 0.831501
## s(SO2):Period1  1.001   1.003   0.001 0.988469
## s(SO2):Period2  1.002   1.003   6.205 0.012767 *
## s(SO2):Period3  1.001   1.002   0.430 0.512248
## s(O3):Period1   2.037   2.628   1.286 0.607255
## s(O3):Period2   2.988   3.754  11.680 0.018134 *
## s(O3):Period3   1.799   2.191   2.948 0.243782
## s(Mean)         2.599   3.335   5.117 0.211109
## s(Rainfall)     1.000   1.001   1.023 0.311984
```

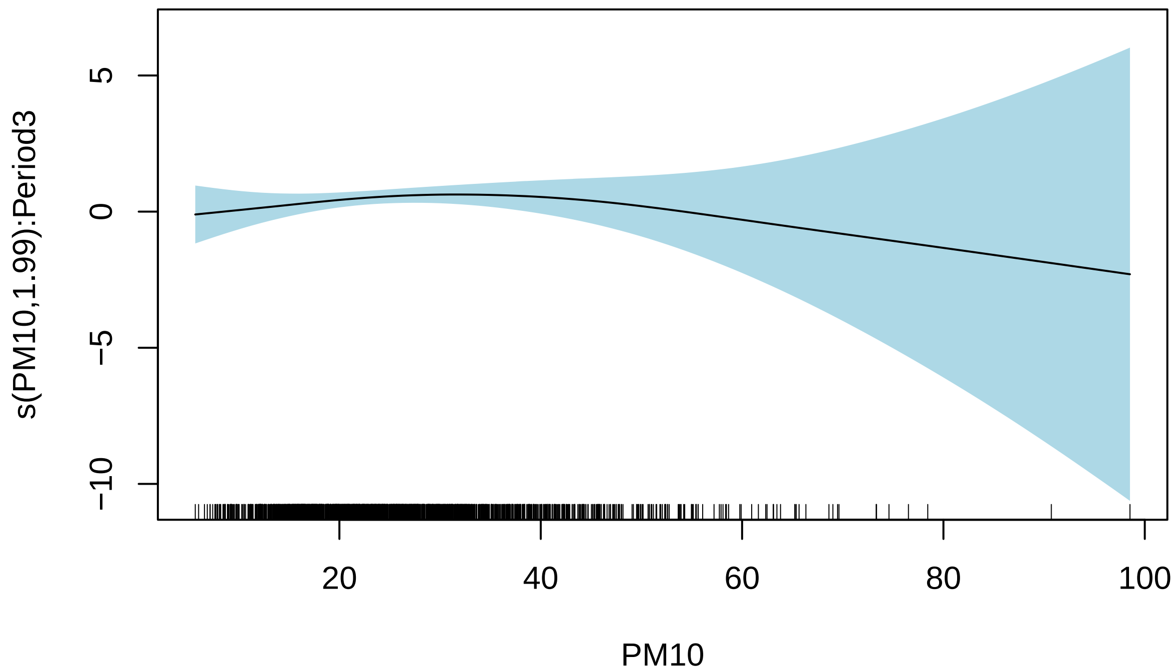
```
## s(Wind)          2.553   3.235   5.108 0.189361
## s(doy)           8.247 363.000  27.620 0.000127 ***
## s(time)          7.952   8.557 124.802  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-sq.(adj) =  0.109   Deviance explained = 14.2%
## -REML = 2054.6  Scale est. = 1         n = 1827
```

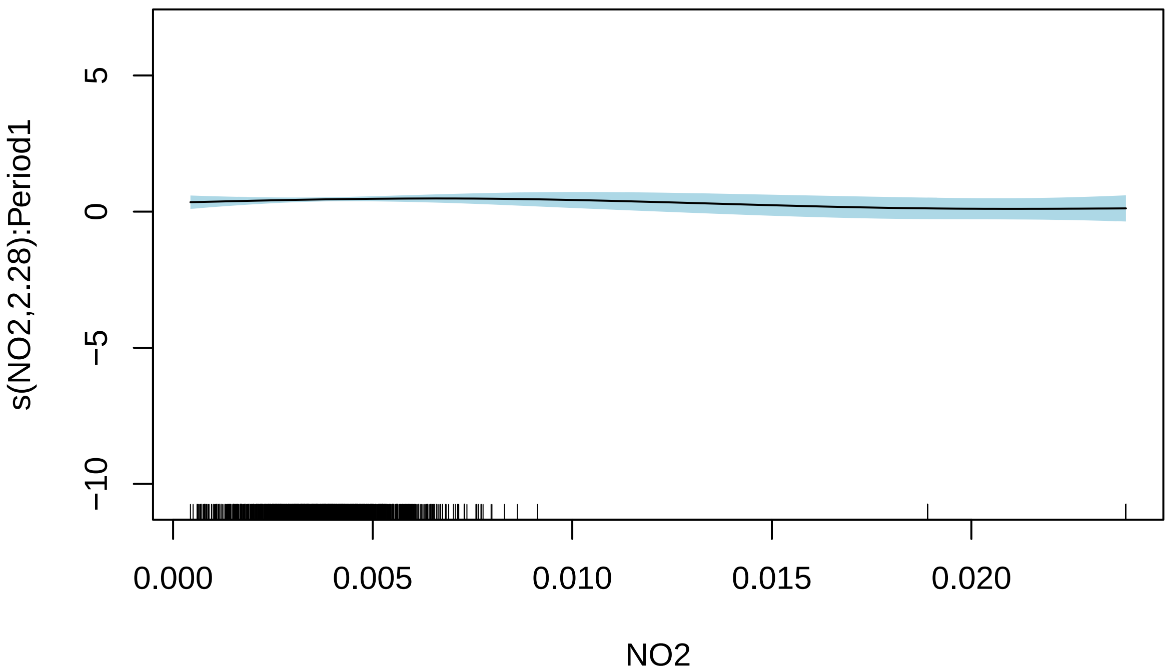
```
plot.gam(kb_CLD_period, shade = TRUE, shade.col = "lightblue", shift = coef(kb_CLD_period)[1],  seWithMe
```



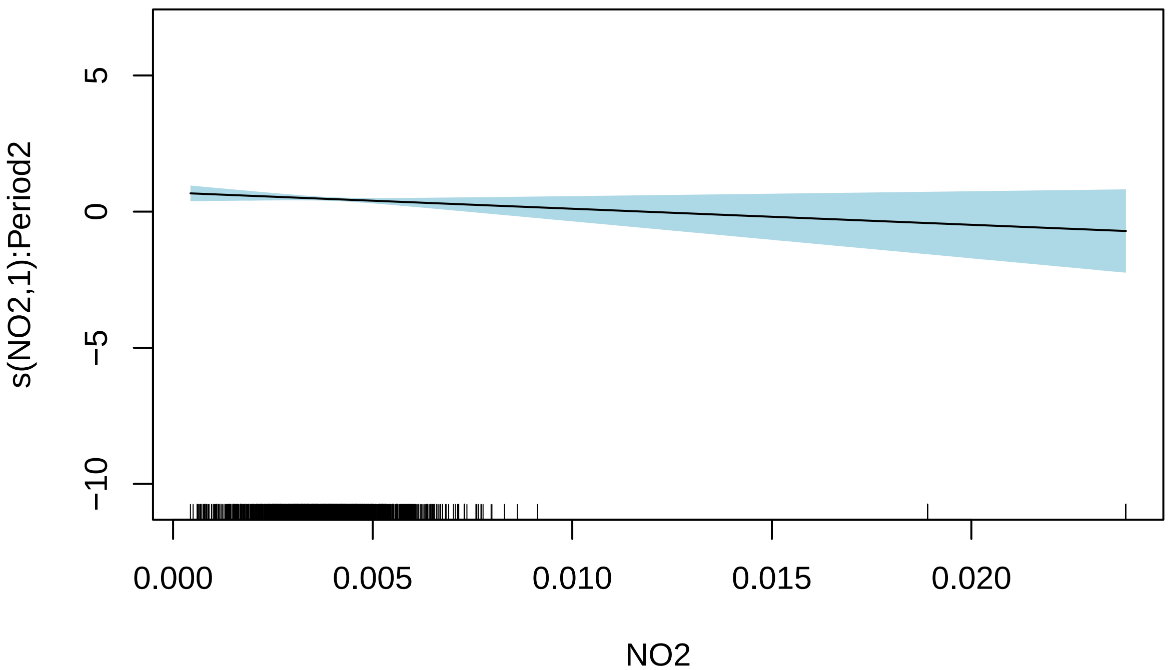


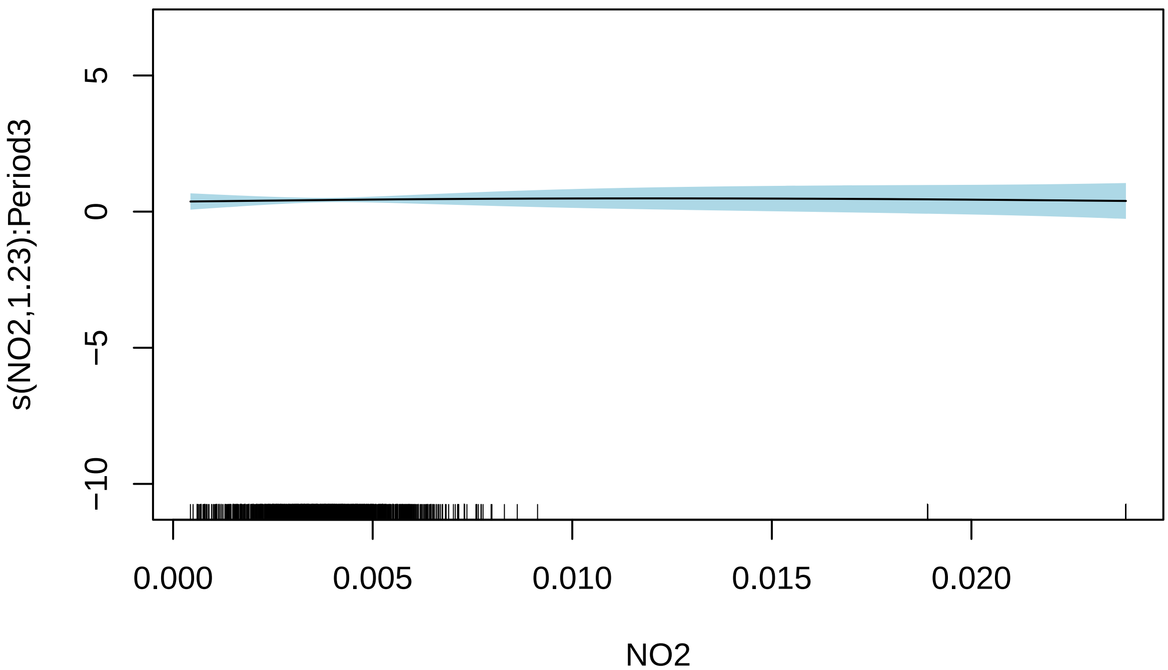
s(PM10,1):Period2
5
0
−5
−10
20
40
60
80
100
PM10


s(PM10,1.99):Period3
5
0
−5
−10
20
40
60
80
100
PM10

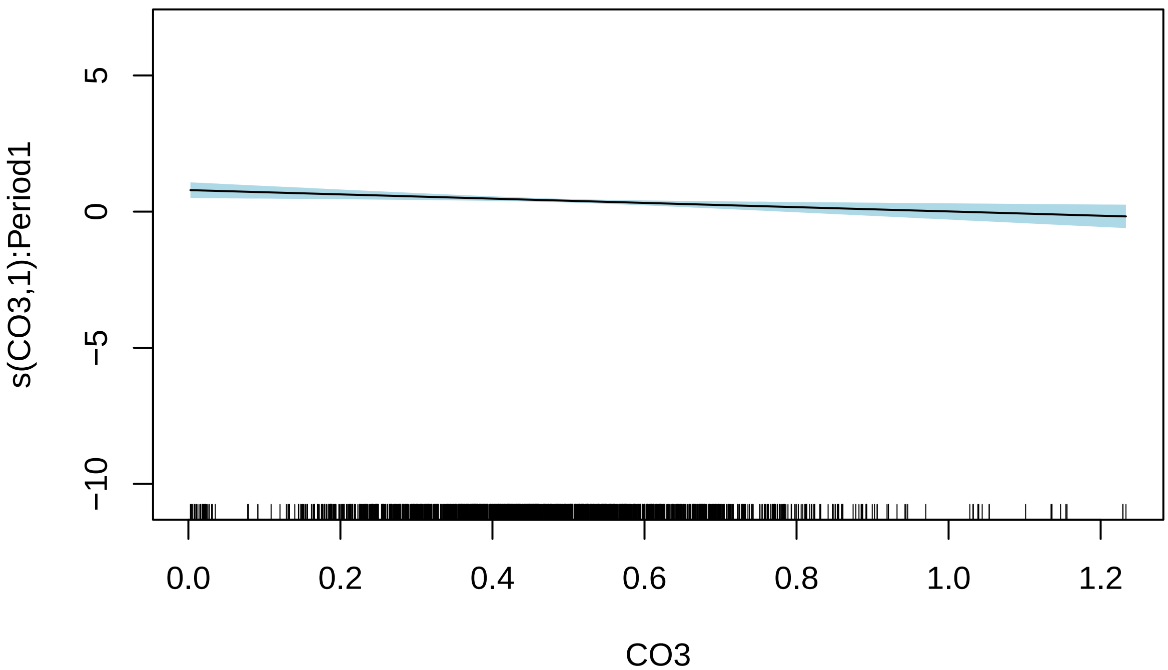


5
0
−5
−10
s(NO2,2.28):Period1
0.000
0.005
0.010
0.015
0.020
NO2

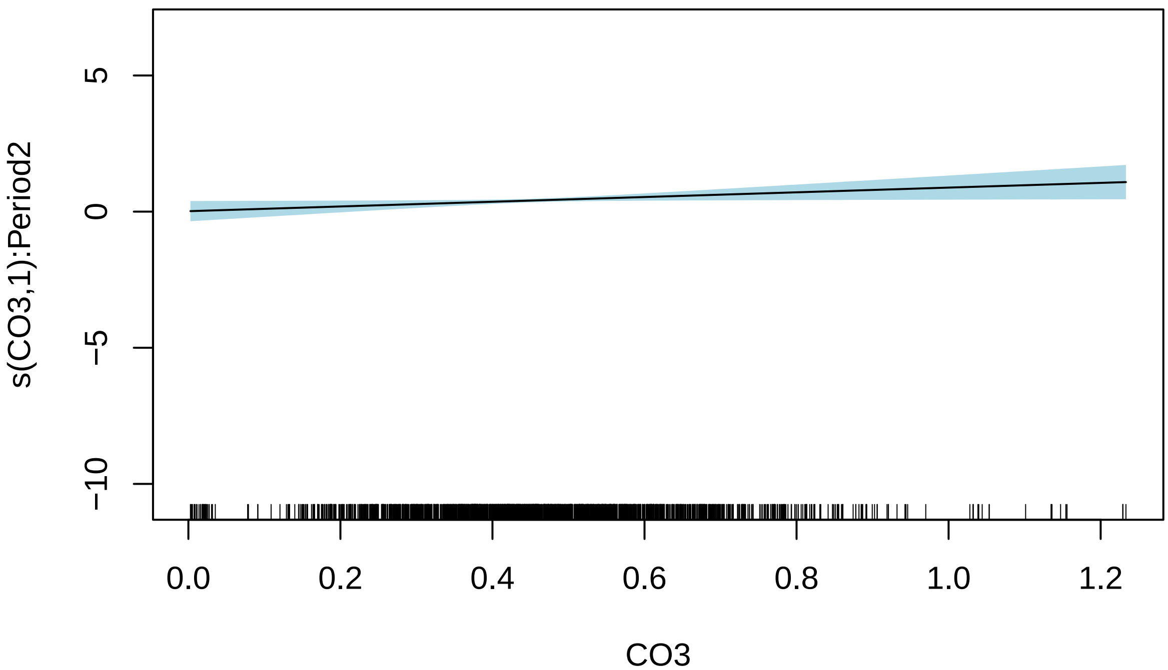

5
0
−5
−10
s(NO2,1):Period2
0.000
0.005
0.010
0.015
0.020
NO2



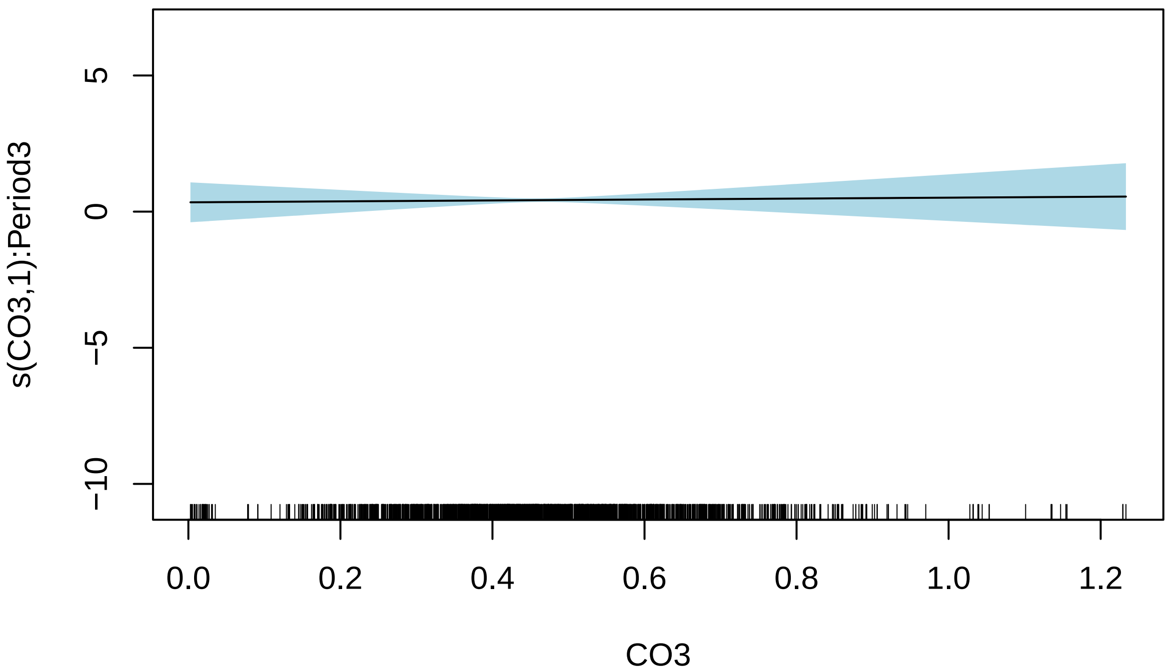
s(NO2,1.23):Period3
5
0
−5
−10
0.000
0.005
0.010
0.015
0.020
NO2



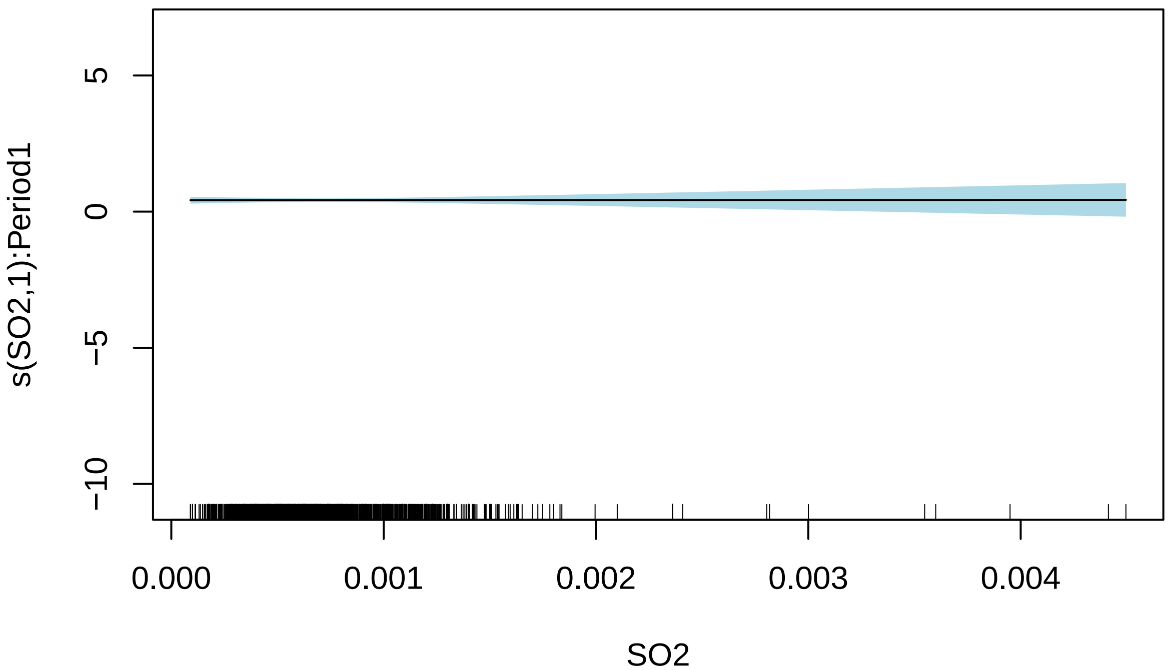
s(CO3,1):Period1
5
0
−5
−10
0.0
0.2
0.4
0.6
0.8
1.0
1.2
CO3

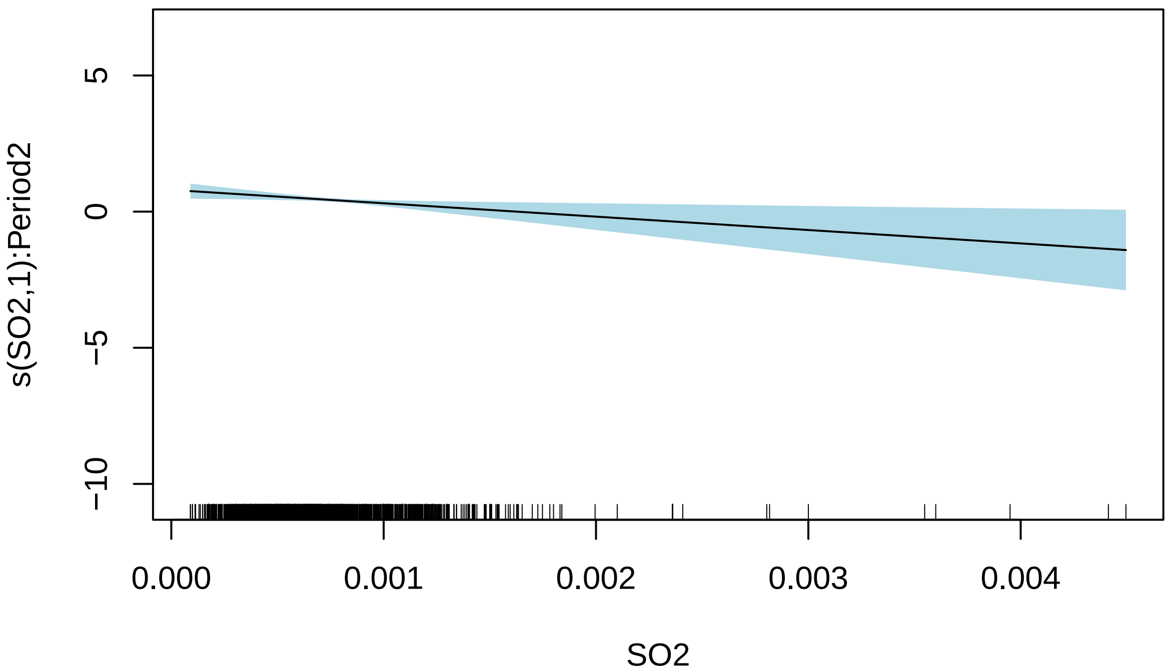


s(CO3,1):Period2
5
0
−5
−10
0.0
0.2
0.4
0.6
0.8
1.0
1.2
CO3

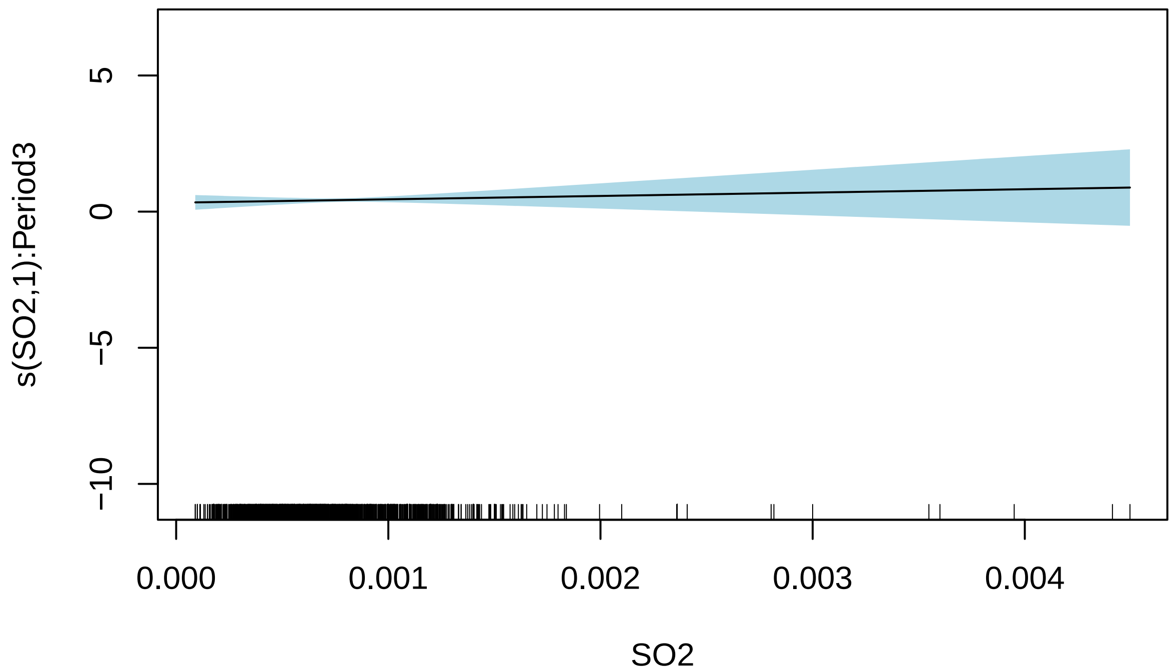


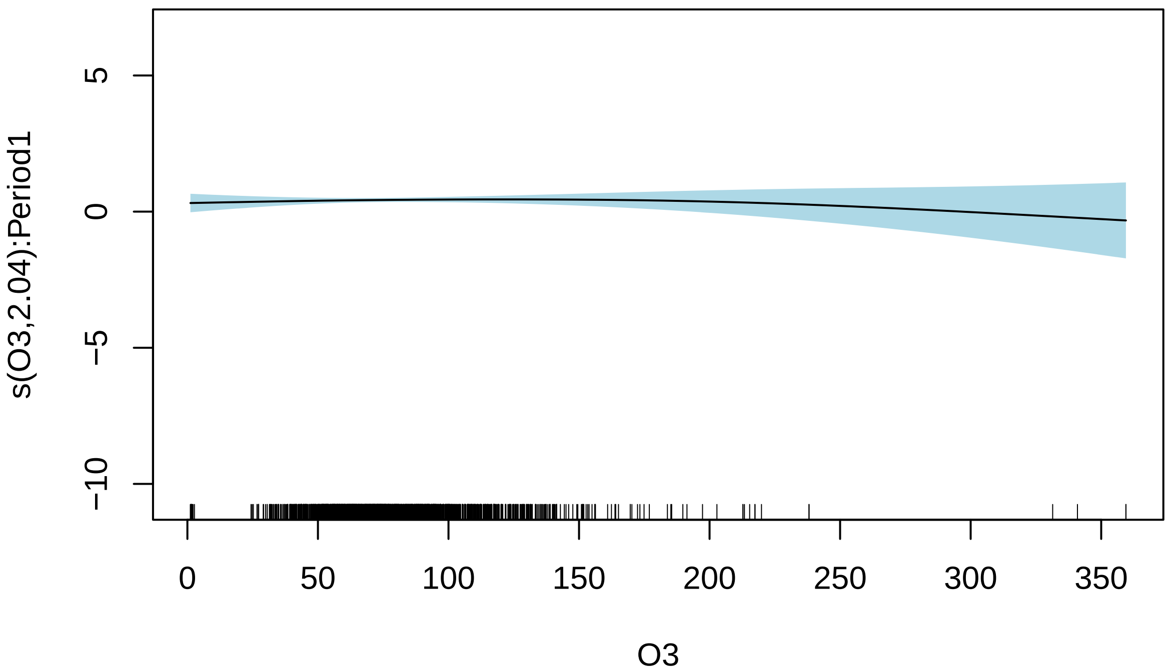
s(CO3,1):Period3
5
0
−5
−10
0.0
0.2
0.4
0.6
0.8
1.0
1.2
CO3

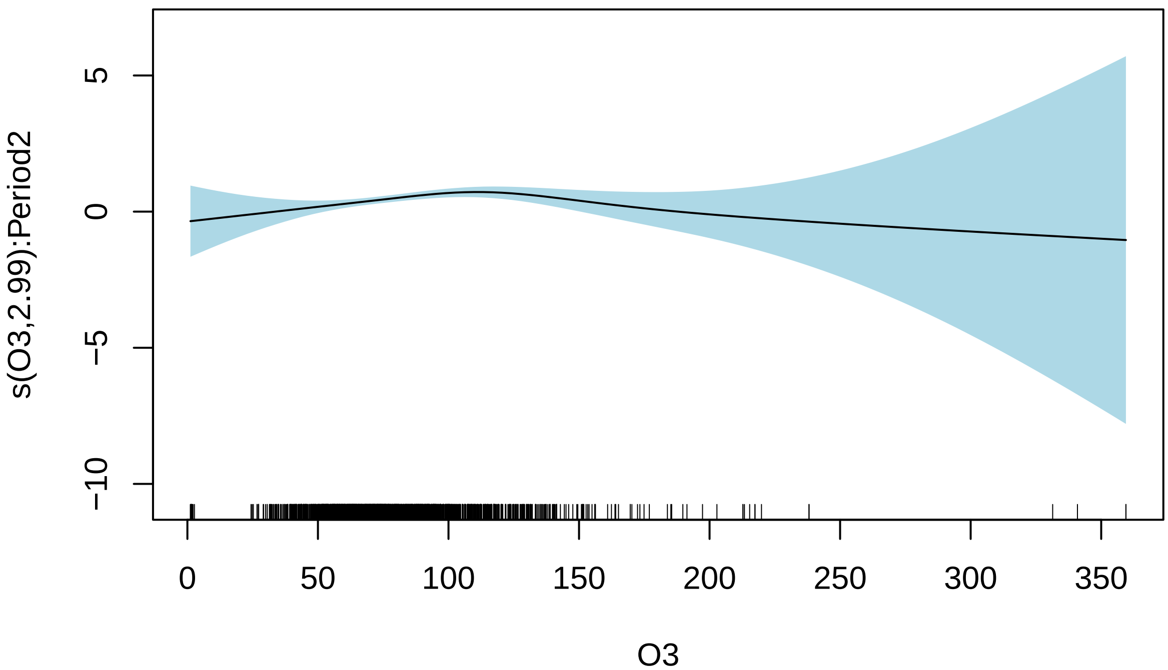

s(SO2,1):Period1
5
0
−5
−10
0.000
0.001
0.002
0.003
0.004
SO2



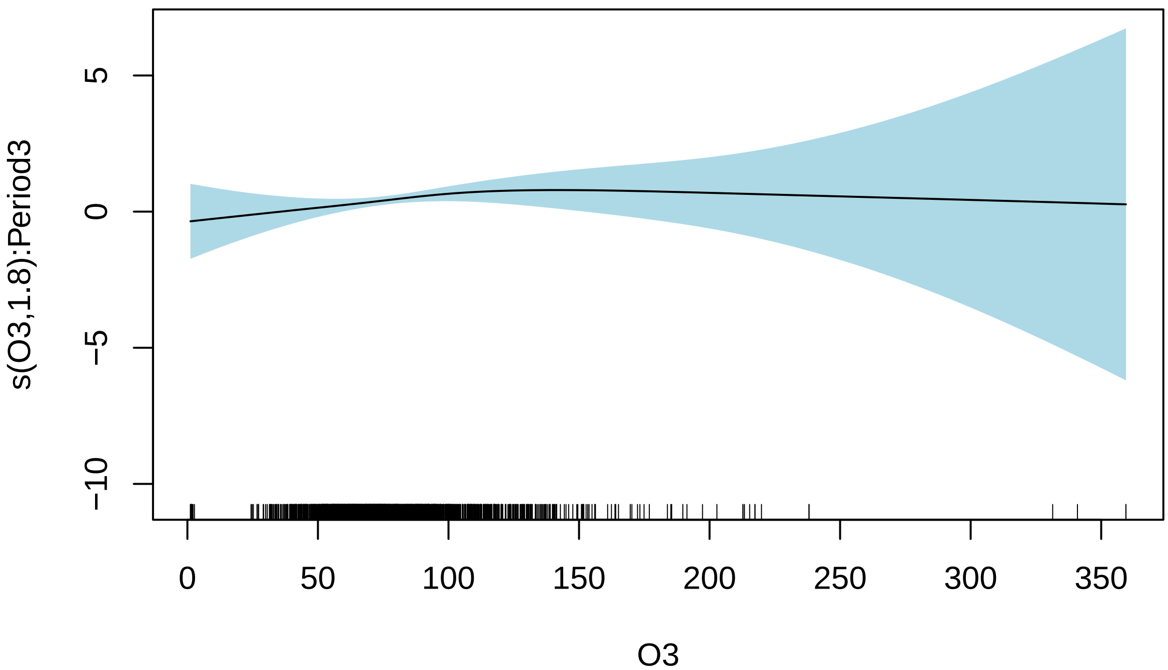
s(SO2,1):Period2
5
0
−5
−10
0.000
0.001
0.002
0.003
0.004
SO2

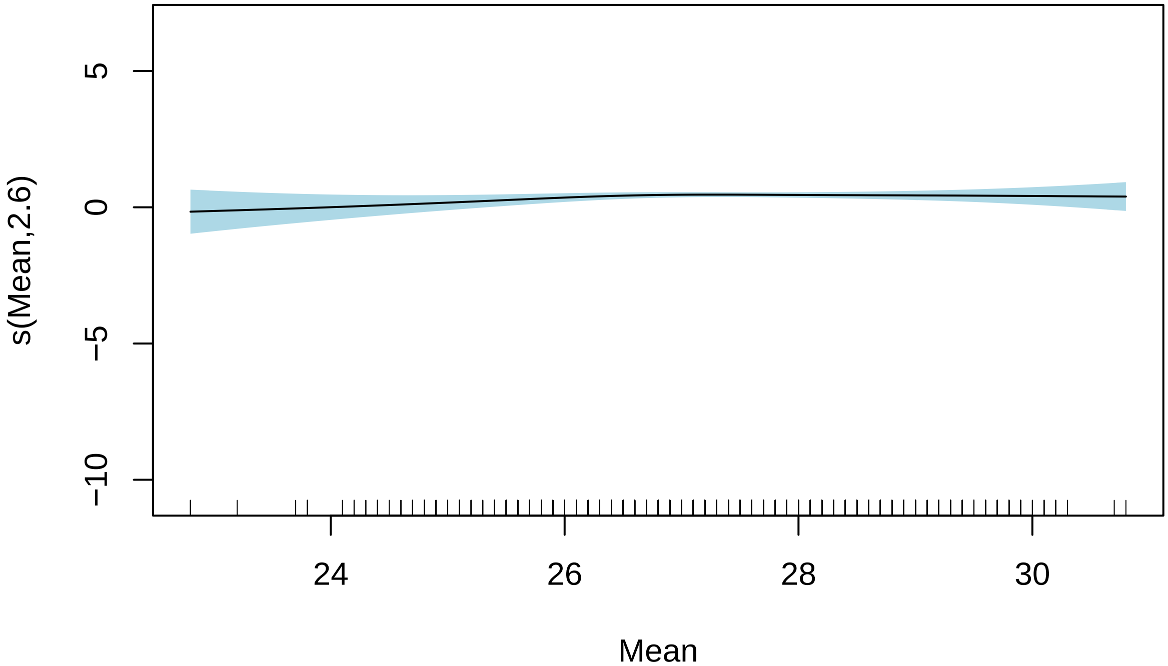

s(SO2,1):Period3
5
0
−5
−10
0.000
0.001
0.002
0.003
0.004
SO2


s(O3,2.04):Period1
5
0
−5
−10
0
50
100
150
200
250
300
350
O3

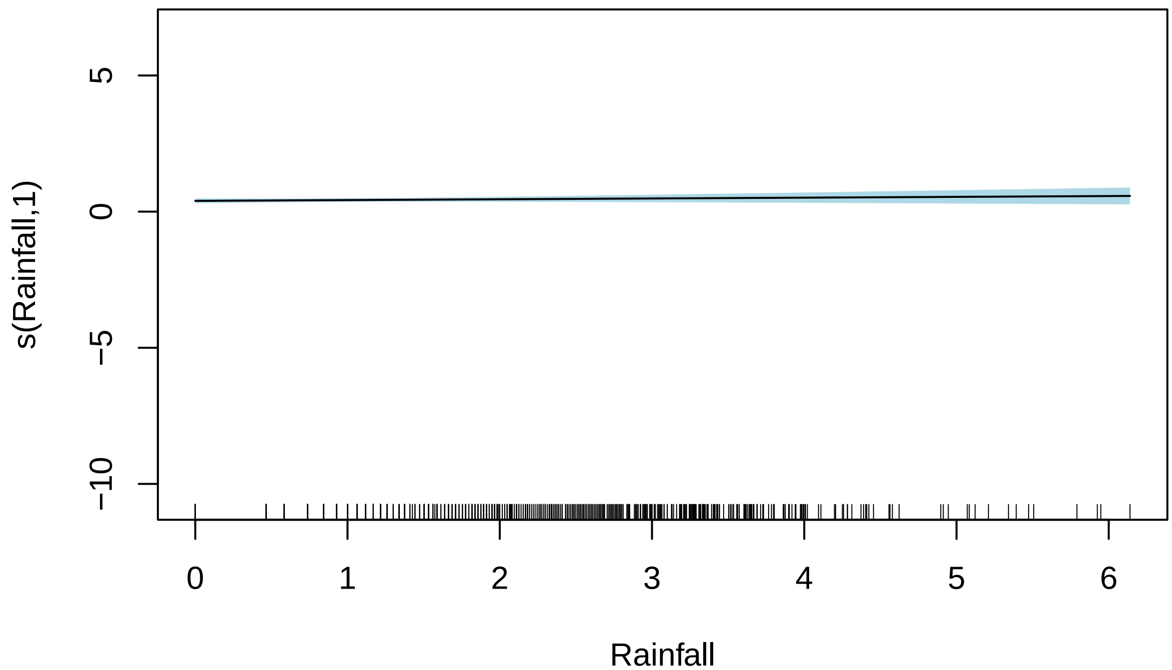


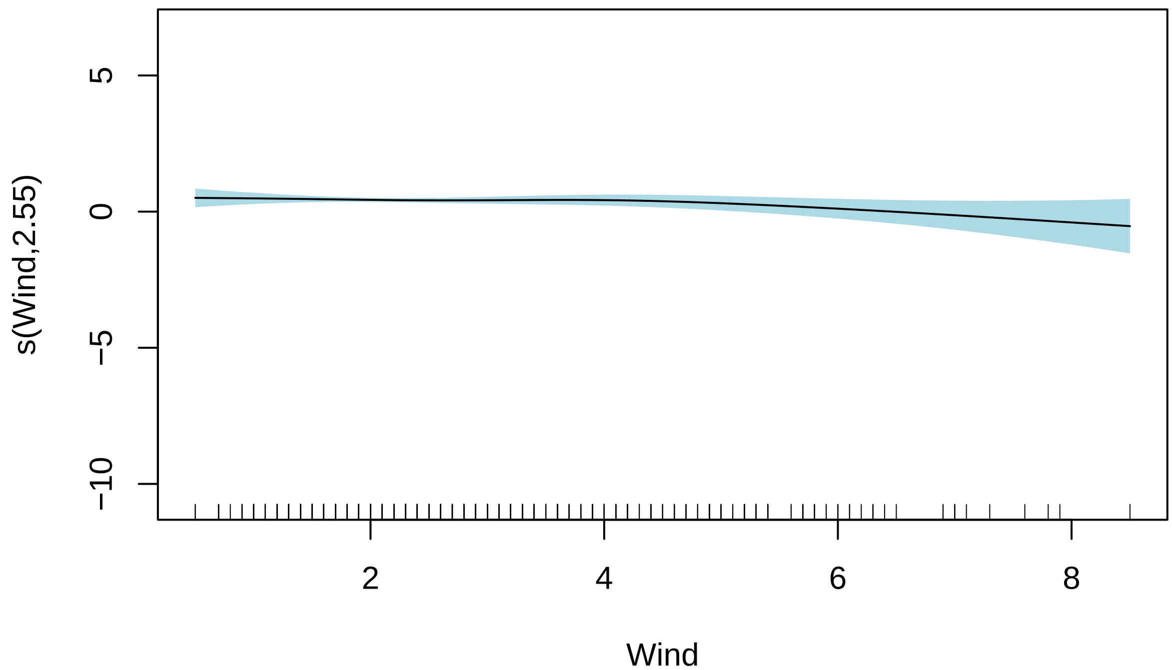
s(O3,2.99):Period2
5
0
−5
−10
0
50
100
150
200
250
300
350
O3

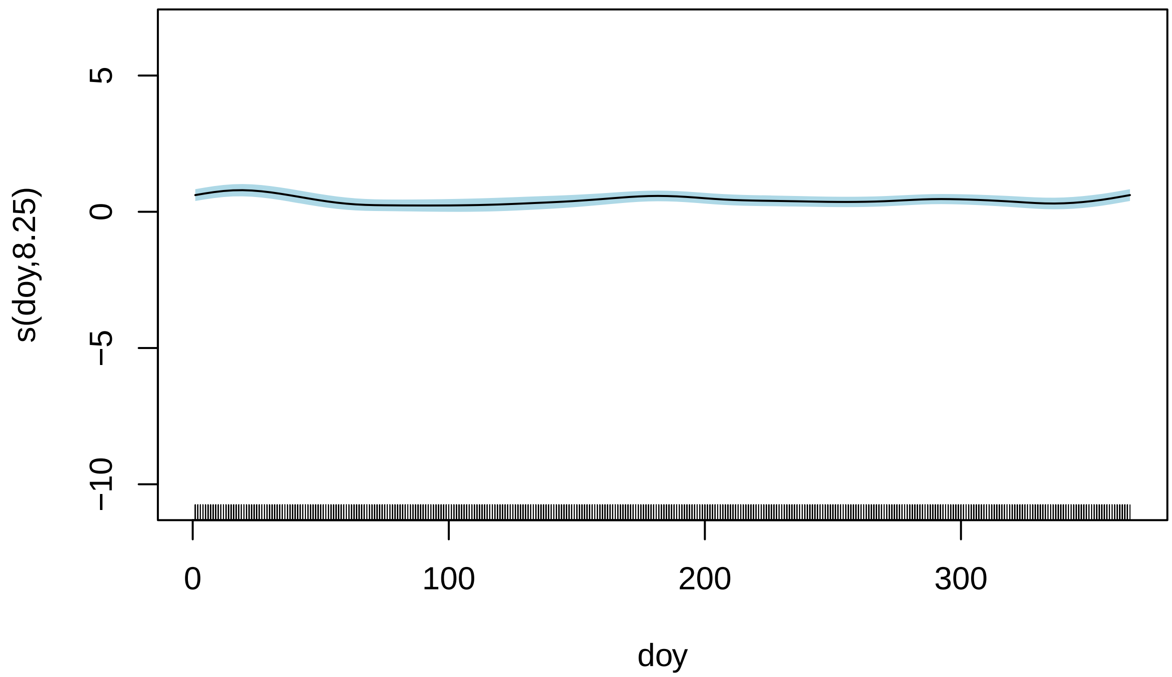

s(O3,1.8):Period3
5
0
−5
−10
0
50
100
150
200
250
300
350
O3



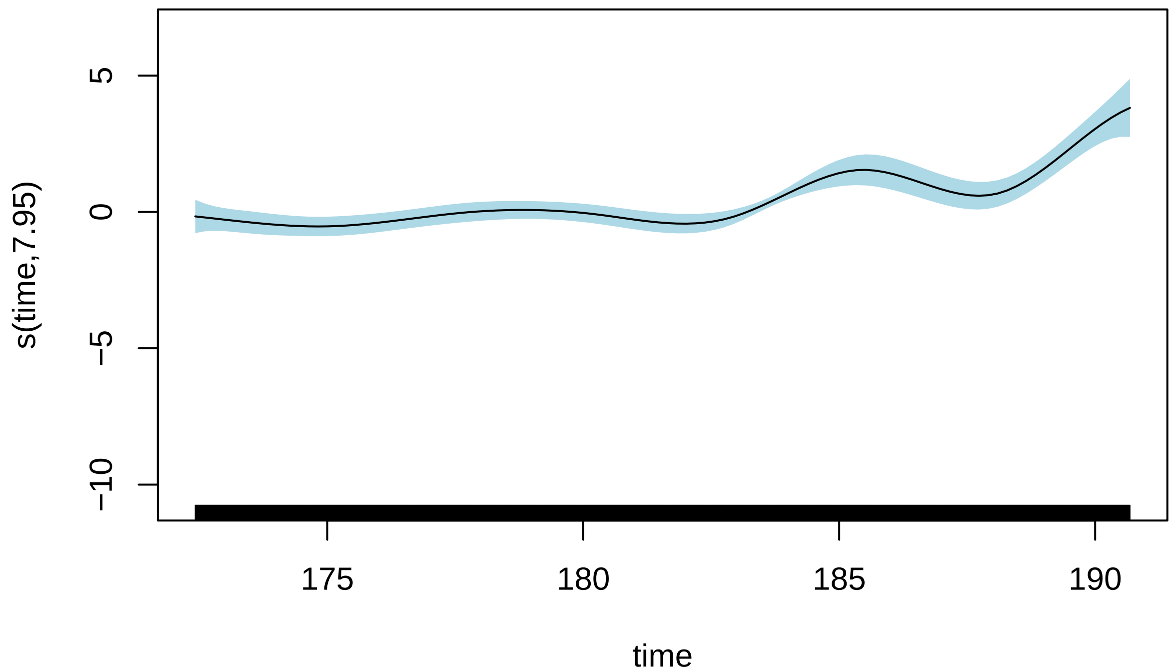
s(Mean,2.6)
5
0
−5
−10
24
26
28
30
Mean


5
0
−5
−10
s(Rainfall,1)
0
1
2
3
4
5
6
Rainfall

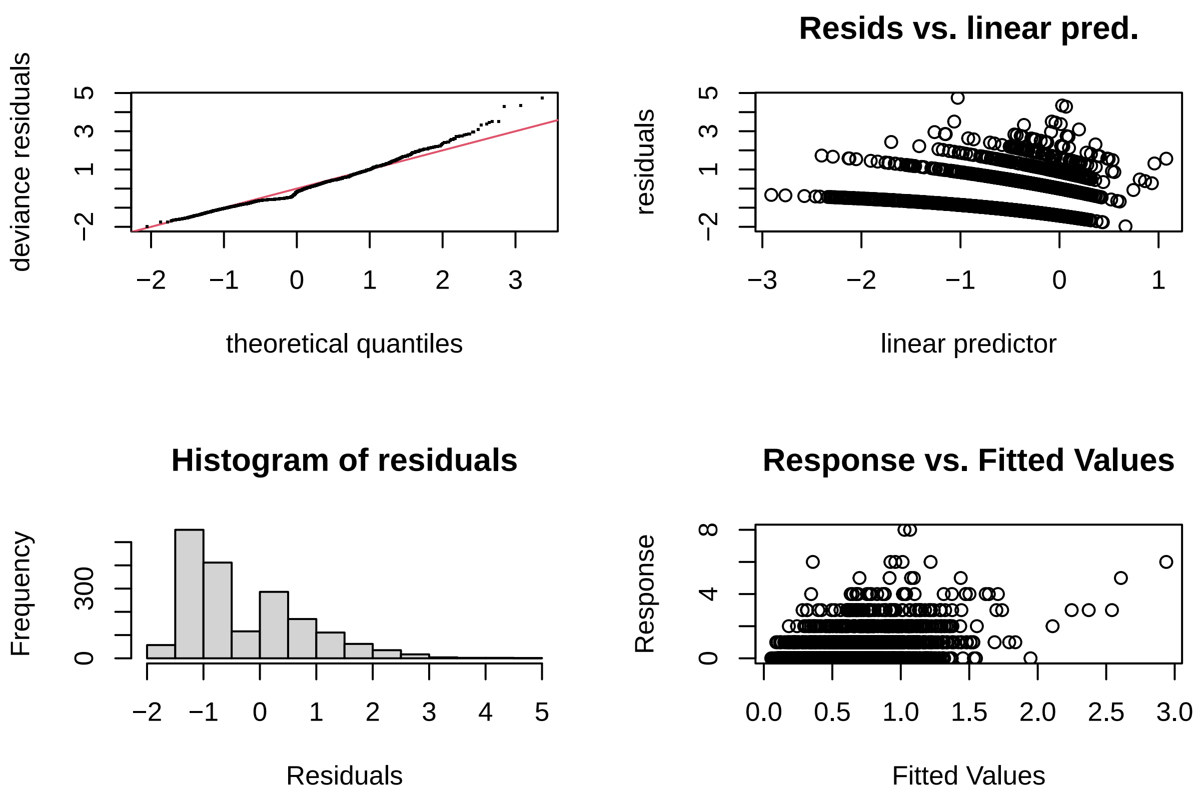

s(Wind,2.55)
5
0
−5
−10
2
4
6
8
Wind



5
0
−5
−10
s(doy,8.25)
0
100
200
300
doy




```
gam.check(kb_CLD_period)
```




```
##
## Method: REML   Optimizer: outer newton
## full convergence after 9 iterations.
## Gradient range [-0.0002255483,7.979545e-05]
## (score 2054.56 & scale 1).
## Hessian positive definite, eigenvalue range [5.074431e-05,2.397956].
## Model rank =  537 / 537
##
## Basis dimension (k) checking results. Low p-value (k-index<1) may
## indicate that k is too low, especially if edf is close to k'.
##
##                      k'   edf k-index p-value
## s(PM10):Period1    9.00  2.42    0.93    0.43
## s(PM10):Period2    9.00  1.00    0.93    0.49
## s(PM10):Period3    9.00  1.99    0.93    0.44
## s(NO2):Period1     9.00  2.28    0.96    0.90
## s(NO2):Period2     9.00  1.00    0.96    0.90
## s(NO2):Period3     9.00  1.23    0.96    0.90
## s(CO3):Period1     9.00  1.00    0.97    0.92
## s(CO3):Period2     9.00  1.00    0.97    0.95
## s(CO3):Period3     9.00  1.00    0.97    0.95
## s(SO2):Period1     9.00  1.00    0.95    0.72
## s(SO2):Period2     9.00  1.00    0.95    0.73
## s(SO2):Period3     9.00  1.00    0.95    0.68
## s(O3):Period1      9.00  2.04    0.94    0.66
## s(O3):Period2      9.00  2.99    0.94    0.62
```

```
## s(O3):Period3      9.00   1.80    0.94    0.61
## s(Mean)            9.00   2.60    0.92    0.22
## s(Rainfall)        9.00   1.00    0.93    0.46
## s(Wind)            9.00   2.55    0.96    0.82
## s(doy)           363.00   8.25    0.96    0.85
## s(time)            9.00   7.95    0.86  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

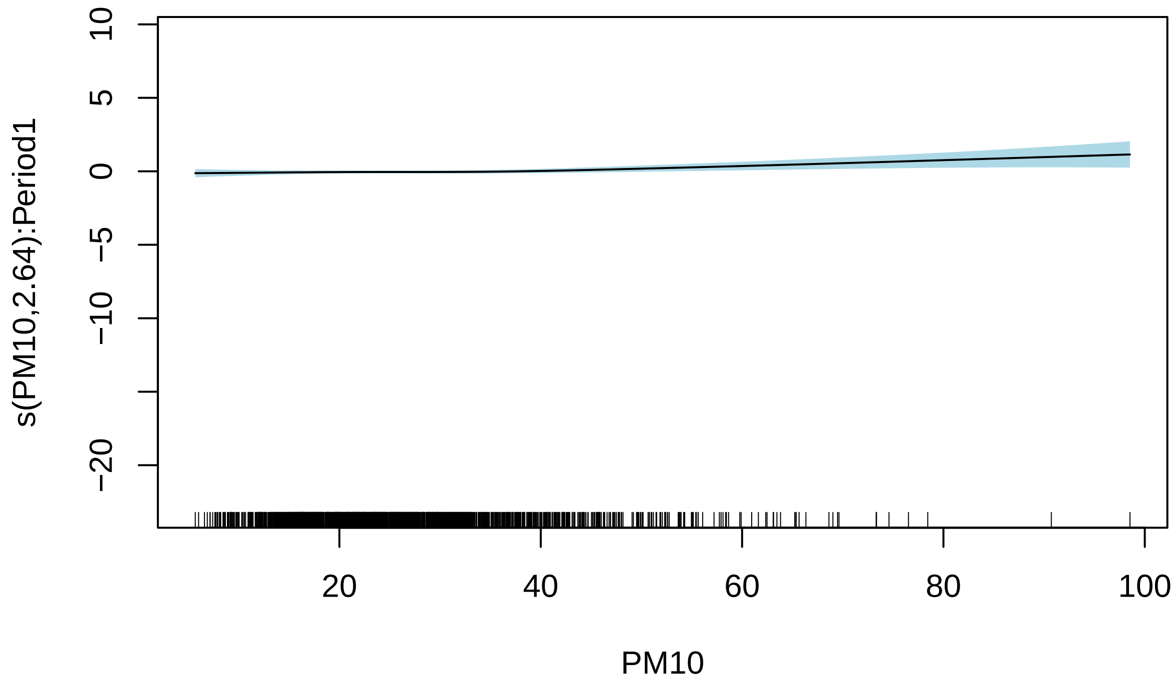
```
kb_LRTI_period <- gam(LRTI ~ s(PM10, by=Period) + s(NO2, by=Period) + s(CO3, by=Period) + s(SO2, by=Peri
summary(kb_LRTI_period)
```

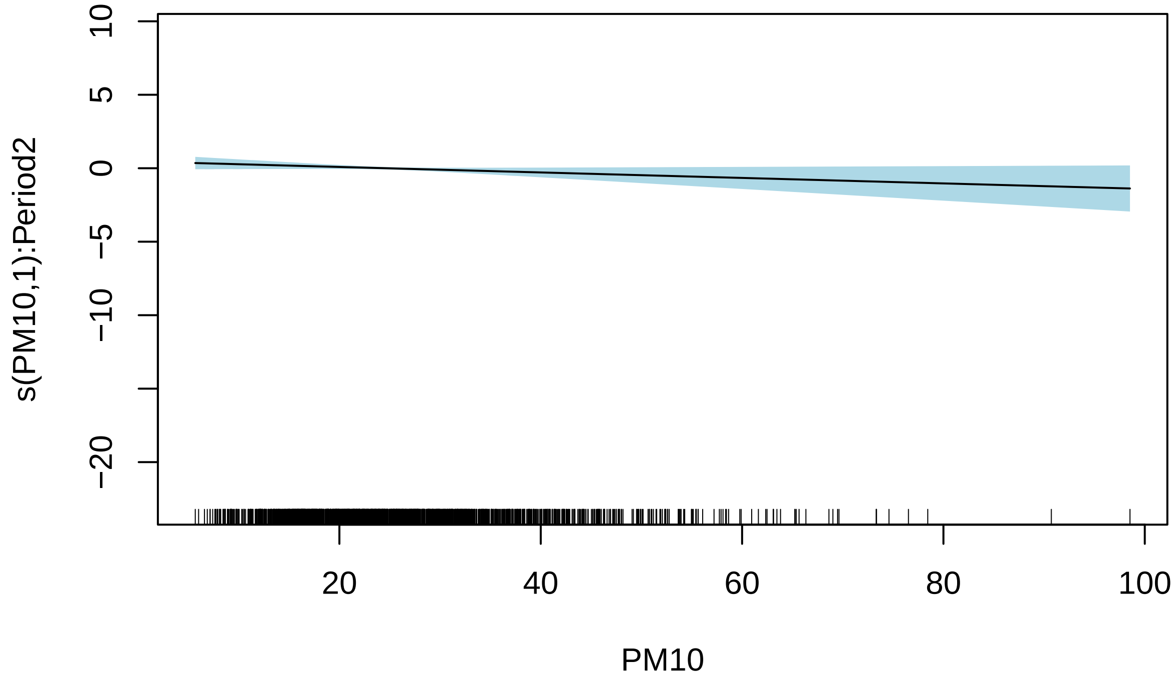
**KB LRTI by period**

```
##
## Family: poisson
## Link function: log
##
## Formula:
## LRTI ~ s(PM10, by = Period) + s(NO2, by = Period) + s(CO3, by = Period) +
##     s(SO2, by = Period) + s(O3, by = Period) + s(Mean) + s(Rainfall) +
##     s(Wind) + s(doy, bs = "cc", k = 365) + s(time, bs = "bs") +
##     Period
##
## Parametric coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -0.02718    0.11908  -0.228   0.8194
## Period2     -1.88207    0.32070  -5.869 4.39e-09 ***
## Period3     -0.94974    0.45791  -2.074   0.0381 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Approximate significance of smooth terms:
##                   edf  Ref.df Chi.sq  p-value
## s(PM10):Period1 2.639   3.347 10.398   0.0225 *
## s(PM10):Period2 1.000   1.000  3.068   0.0799 .
## s(PM10):Period3 2.625   3.334  4.160   0.2811
## s(NO2):Period1  1.000   1.001  0.820   0.3653
## s(NO2):Period2  1.000   1.000  0.176   0.6748
## s(NO2):Period3  1.011   1.022  2.518   0.1183
## s(CO3):Period1  1.009   1.018  1.727   0.1941
## s(CO3):Period2  2.790   3.540 10.919   0.0214 *
## s(CO3):Period3  1.000   1.000  2.468   0.1163
## s(SO2):Period1  2.085   2.648  3.518   0.3601
## s(SO2):Period2  1.001   1.001  0.343   0.5585
## s(SO2):Period3  1.000   1.000  0.139   0.7096
## s(O3):Period1   1.325   1.580  1.163   0.3618
## s(O3):Period2   1.230   1.428  1.343   0.4347
## s(O3):Period3   2.124   2.627  1.956   0.3971
## s(Mean)         1.000   1.000  1.720   0.1898
## s(Rainfall)     1.000   1.000  0.002   0.9643
```

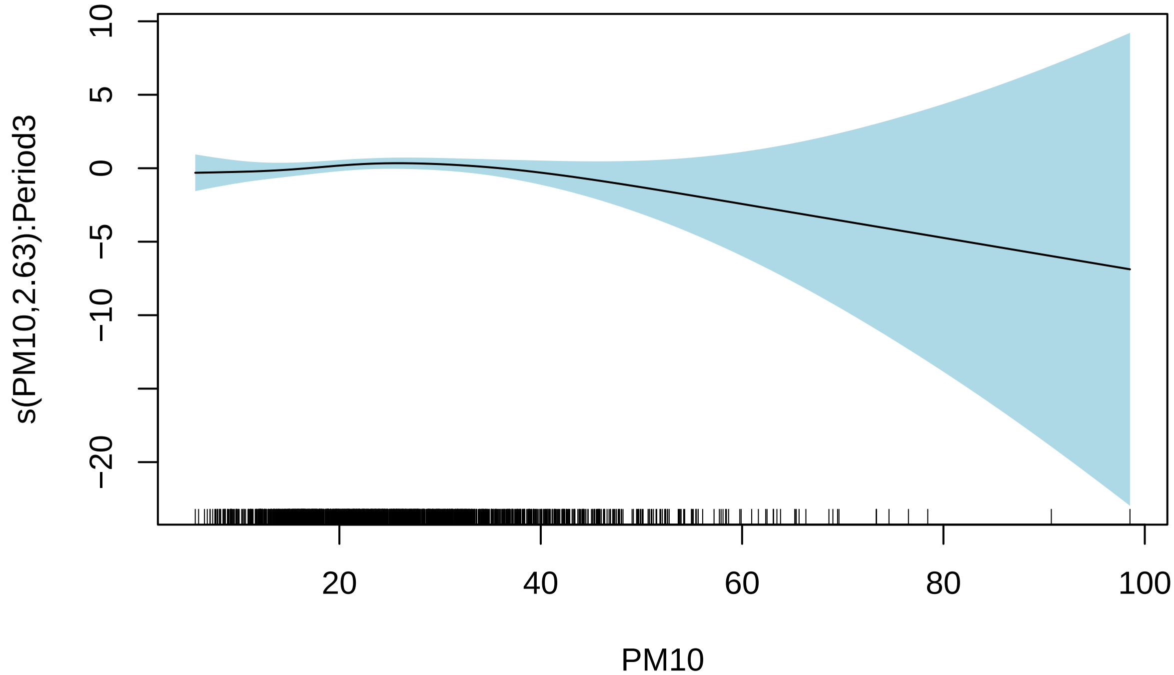
```
## s(Wind)         1.000   1.000  1.295   0.2553
## s(doy)          2.320 361.000  6.657   0.0242 *
## s(time)         4.505   5.357 34.430 3.18e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-sq.(adj) =  0.173   Deviance explained = 22.7%
## -REML = 1771.7  Scale est. = 1         n = 1827
```

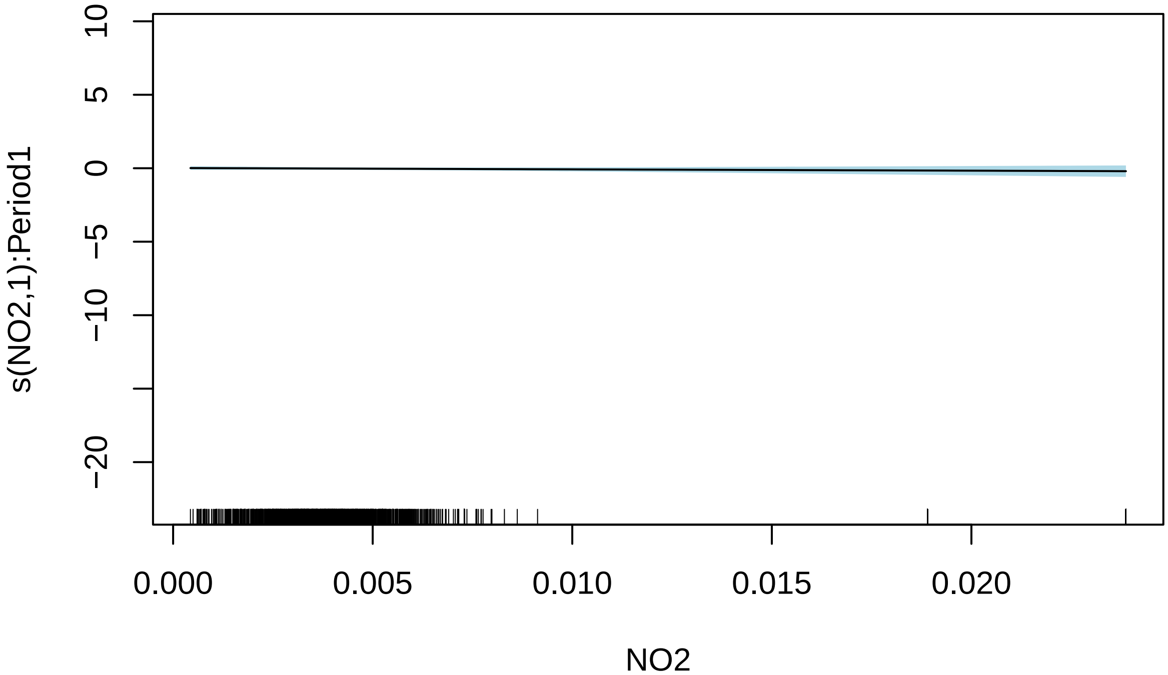
```
plot.gam(kb_LRTI_period, shade = TRUE, shade.col = "lightblue", shift = coef(kb_LRTI_period)[1],  seWit
```



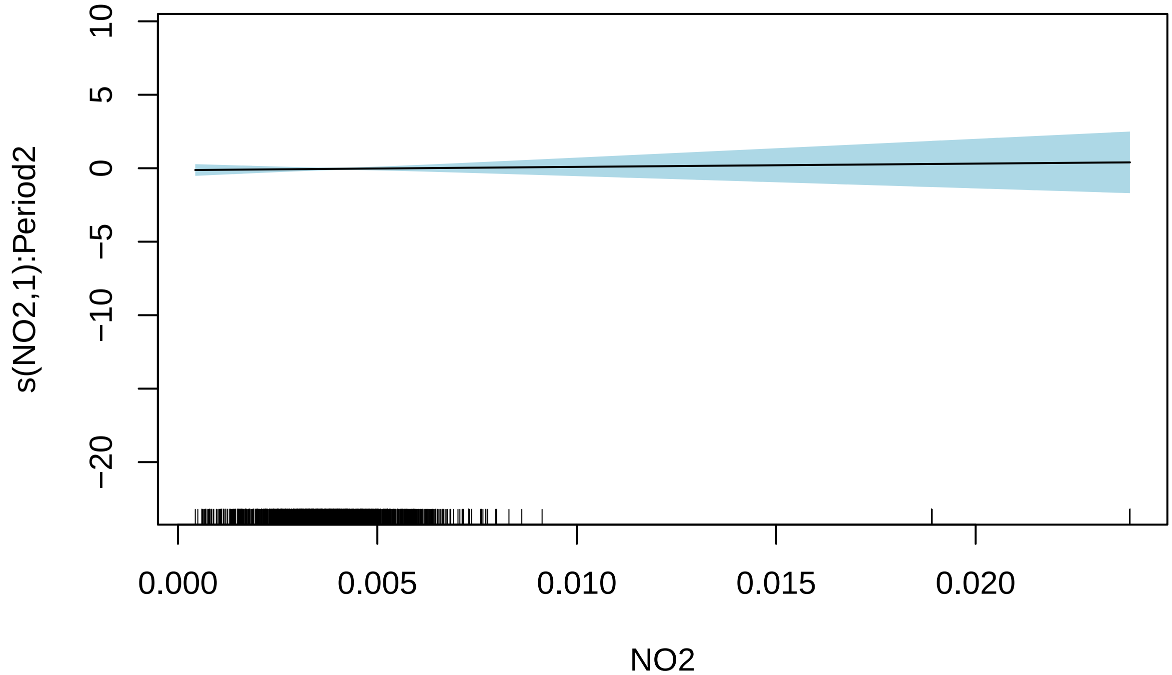


s(PM10,1):Period2
10
5
0
−5
−10
−20
20
40
60
80
100
PM10

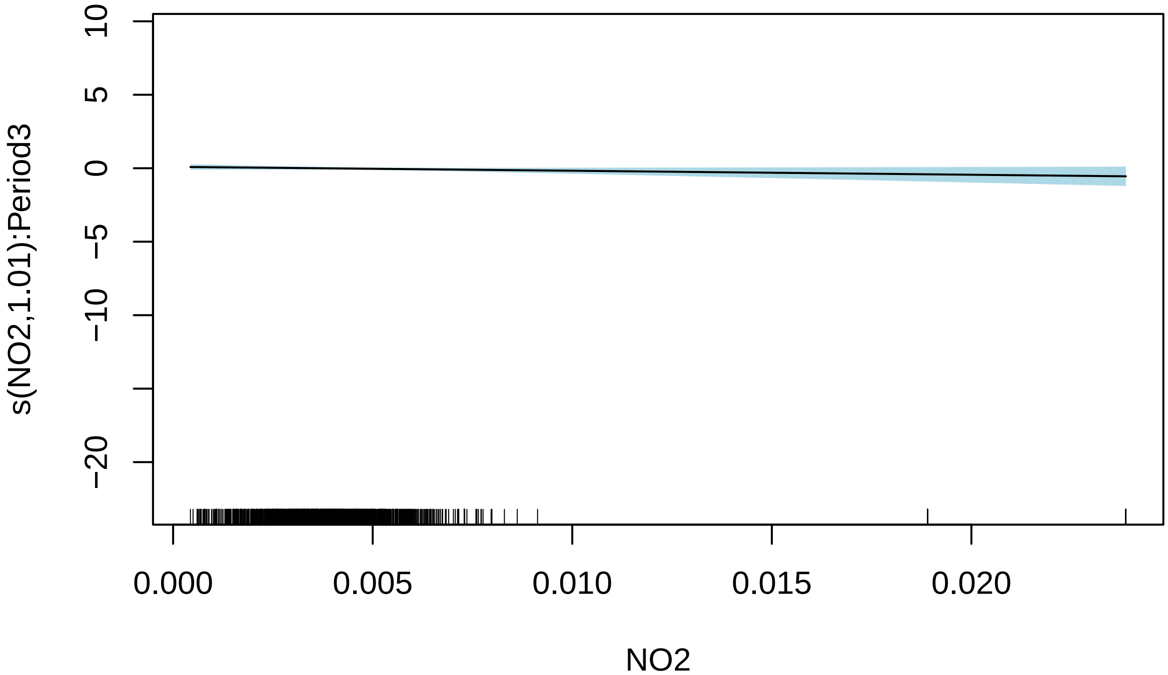

s(PM10,2.63):Period3
10
5
0
−5
−10
−20
20
40
60
80
100
PM10

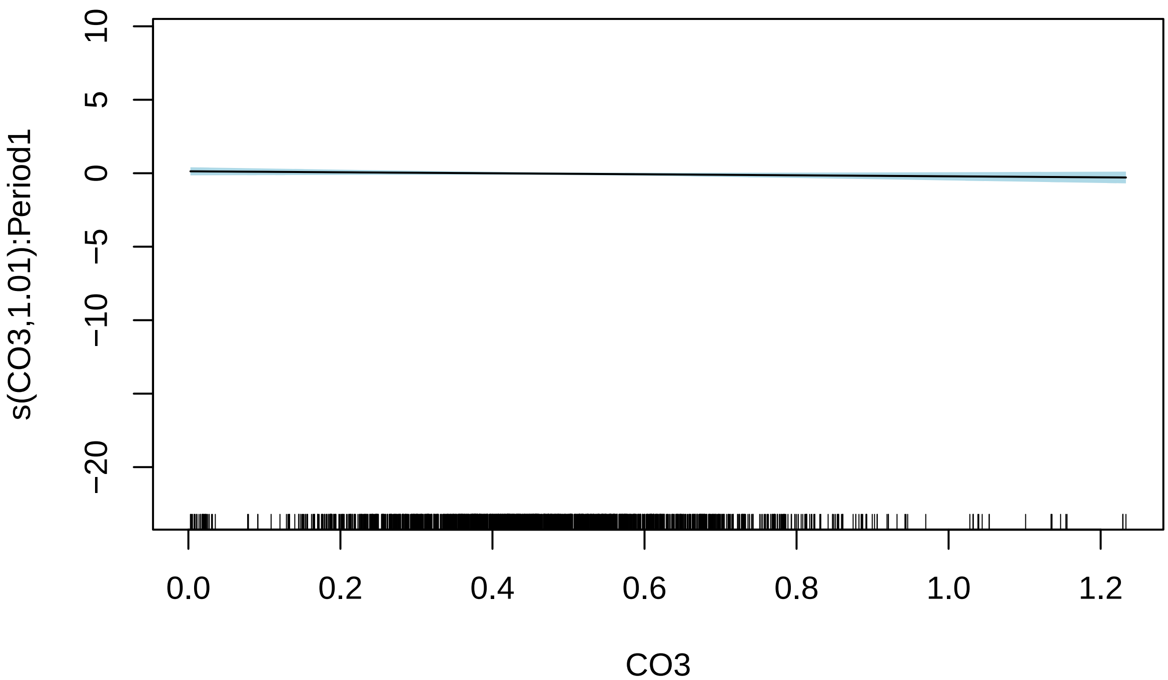


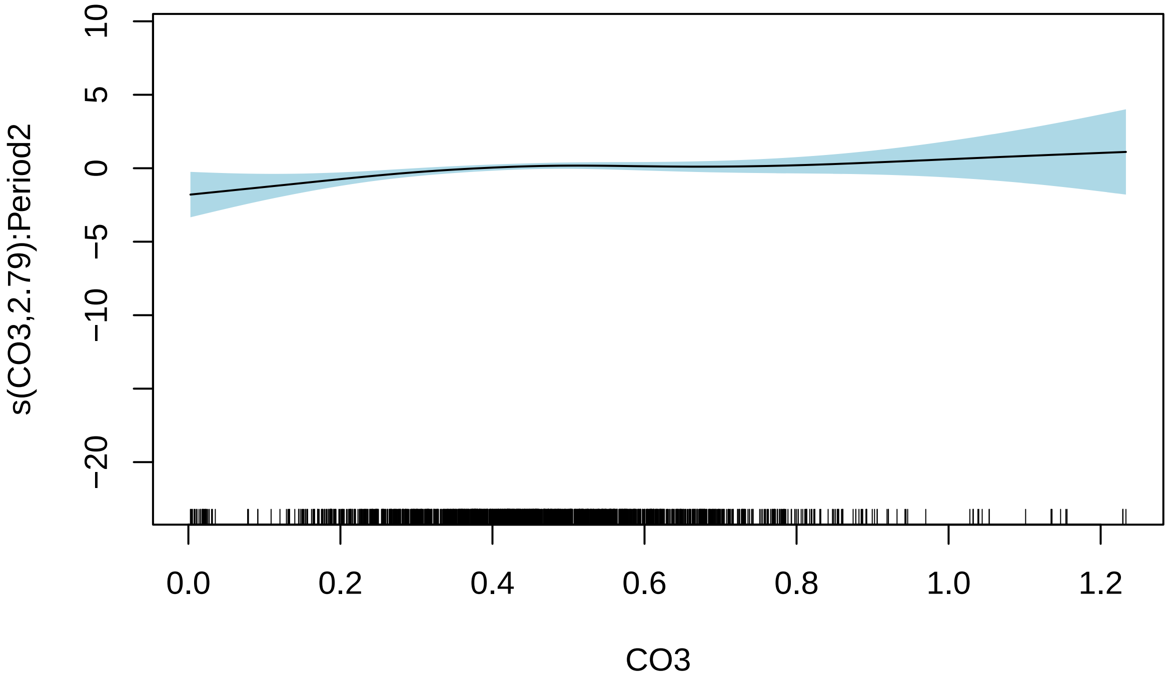
s(NO2,1):Period1
10
5
0
−5
−10
−20
0.000
0.005
0.010
0.015
0.020
NO2

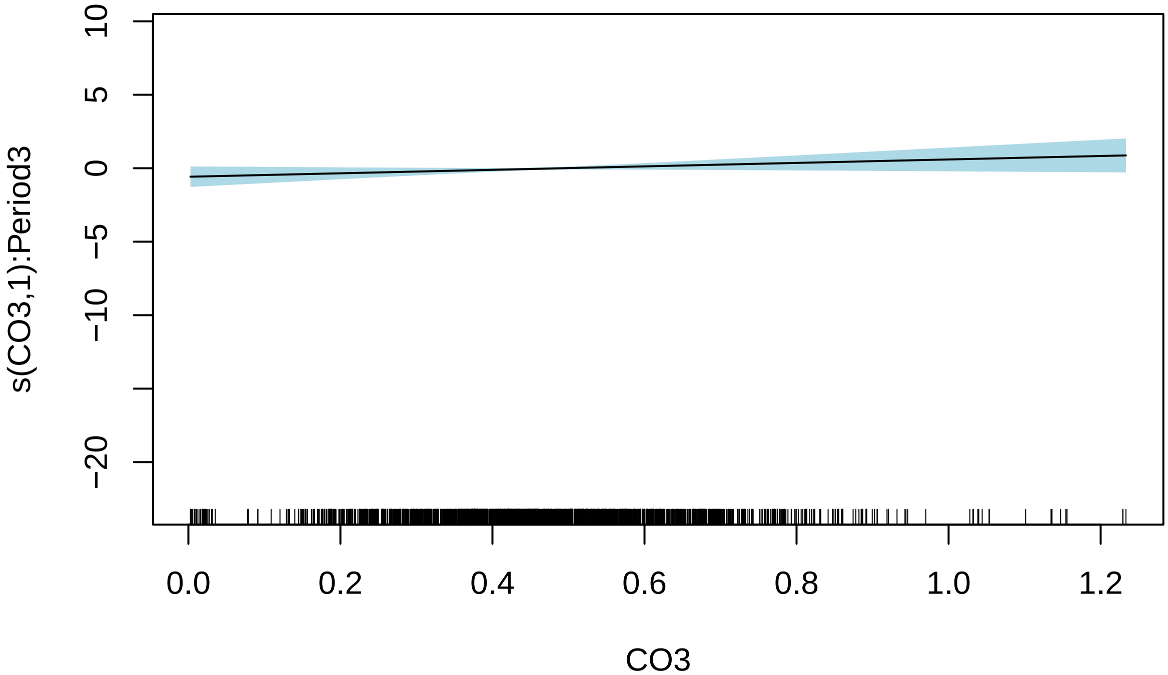


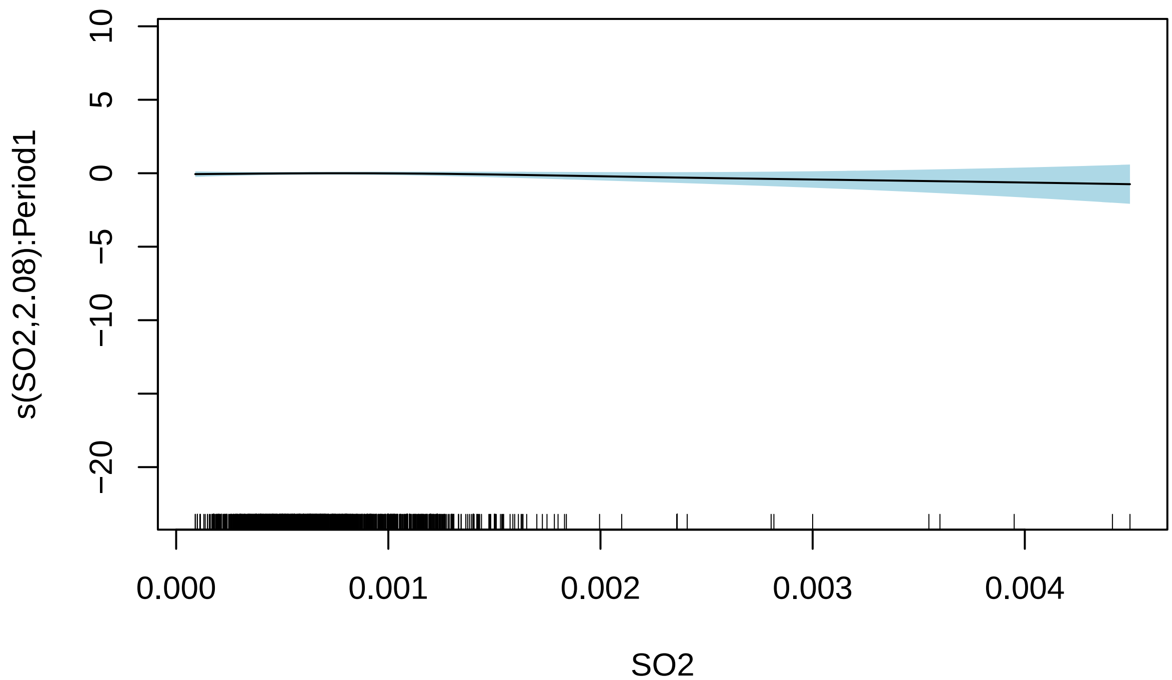
10
5
0
−5
−10
−20
s(NO2,1):Period2
0.000
0.005
0.010
0.015
0.020
NO2


s(NO2,1.01):Period3
10
5
0
−5
−10
−20
0.000
0.005
0.010
0.015
0.020
NO2

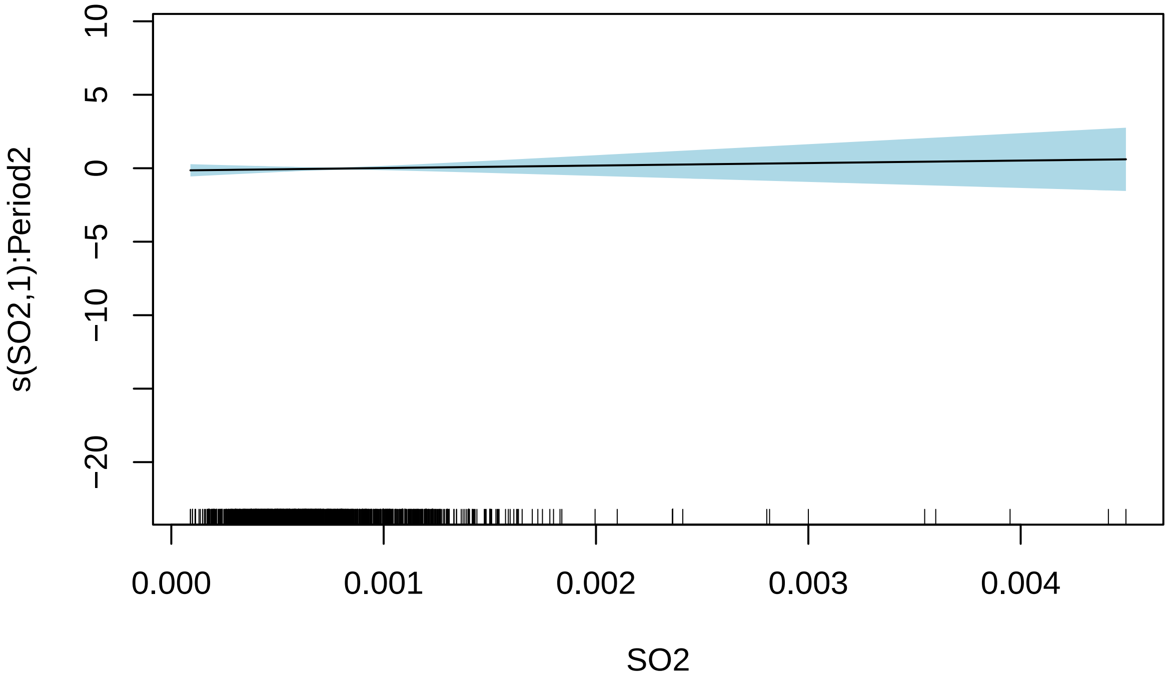

10
5
0
−5
−10
−20
s(CO3,1.01):Period1
0.0
0.2
0.4
0.6
0.8
1.0
1.2
CO3


s(CO3,2.79):Period2
10
5
0
−5
−10
−20
0.0
0.2
0.4
0.6
0.8
1.0
1.2
CO3

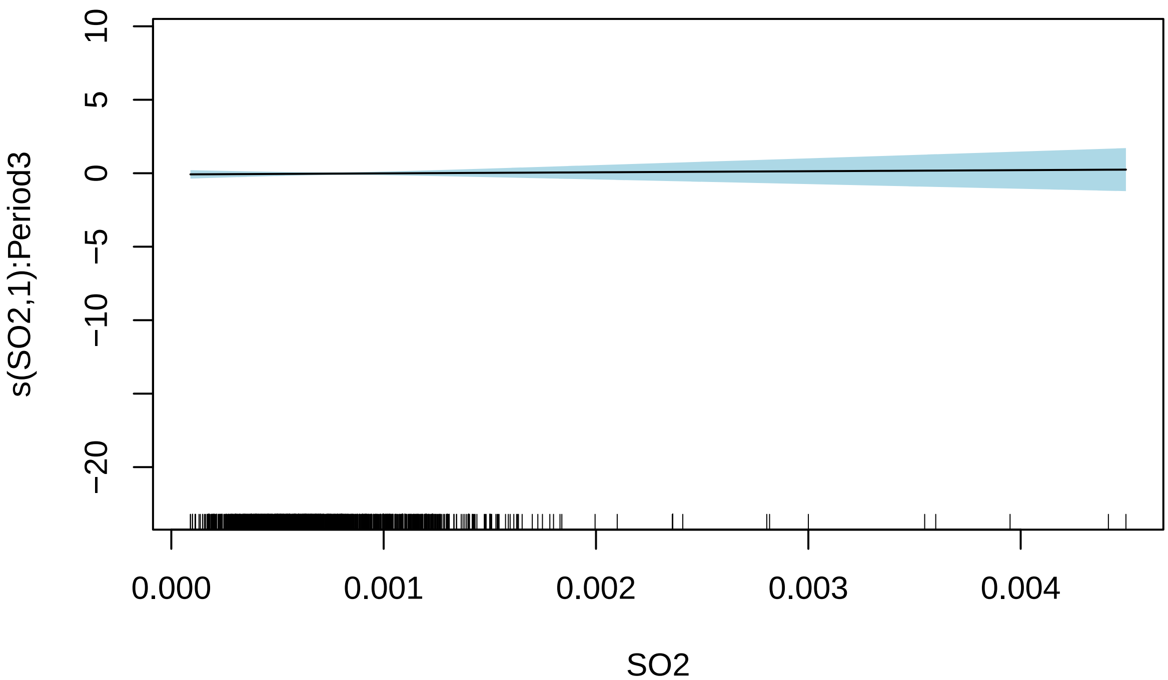

s(CO3,1):Period3
10
5
0
−5
−10
−20
0.0
0.2
0.4
0.6
0.8
1.0
1.2
CO3

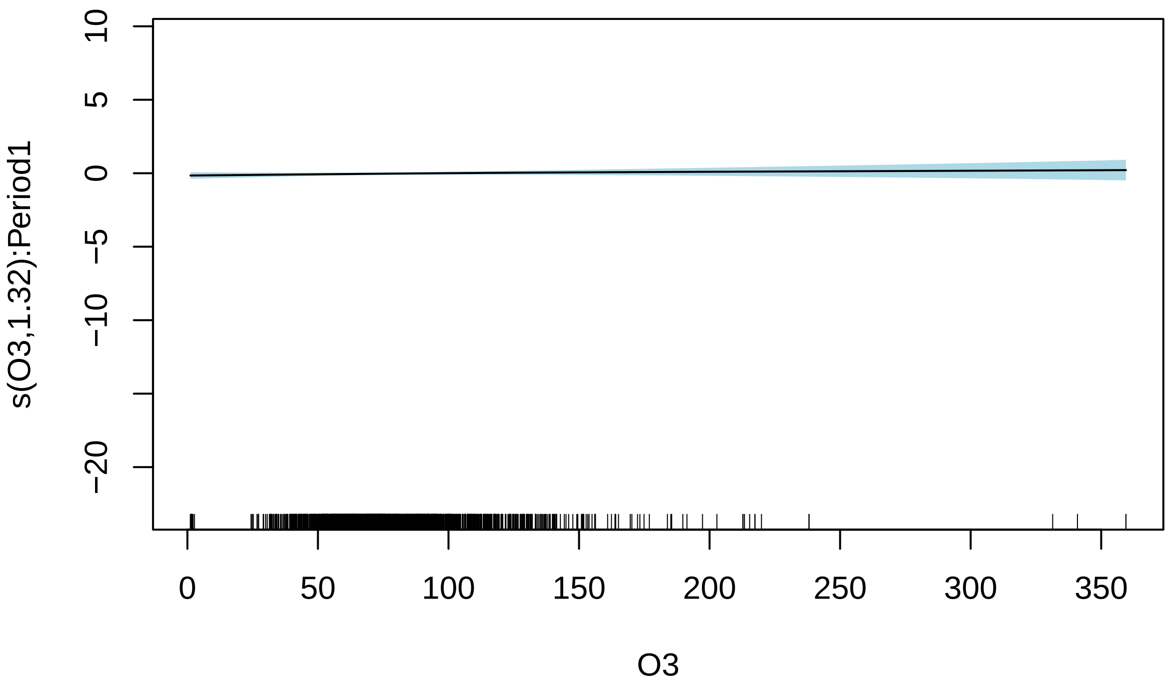


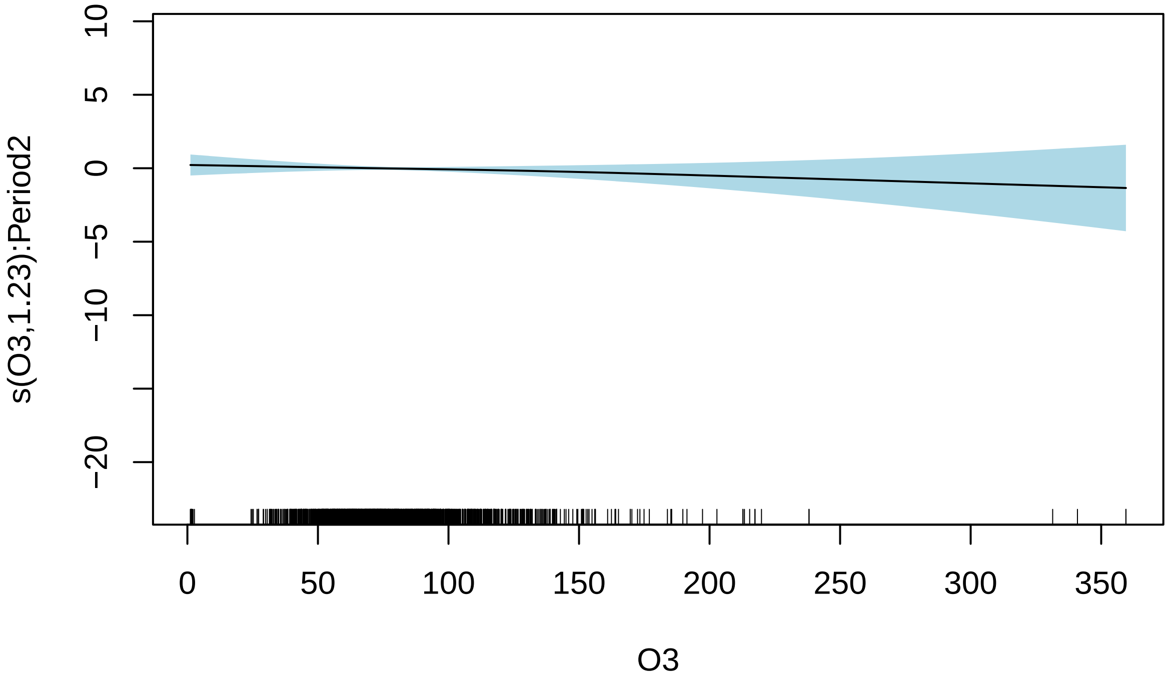
10
5
0
−5
−10
−20
s(SO2,2.08):Period1
0.000
0.001
0.002
0.003
0.004
SO2

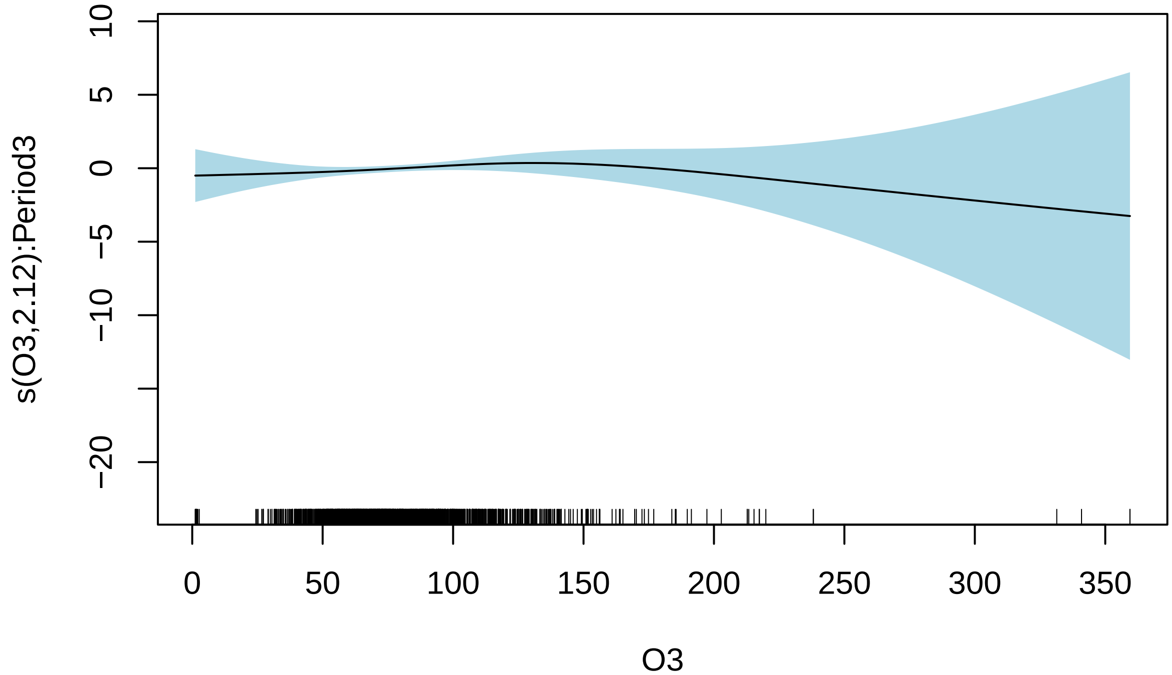


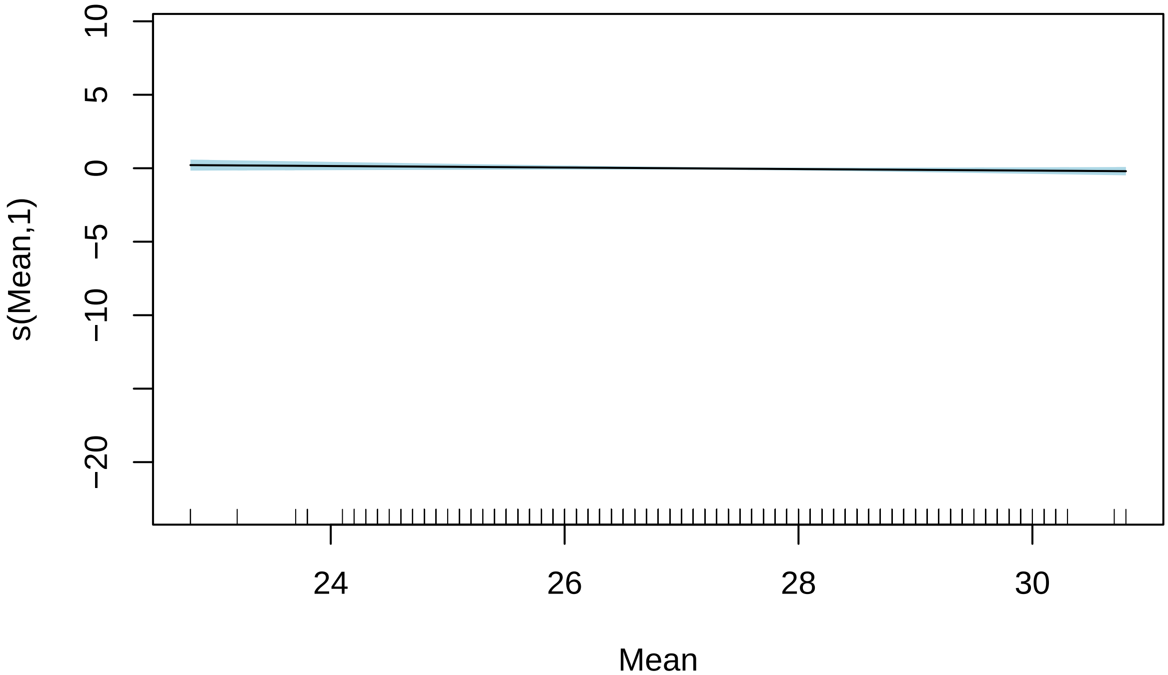
s(SO2,1):Period2
10
5
0
−5
−10
−20
0.000
0.001
0.002
0.003
0.004
SO2

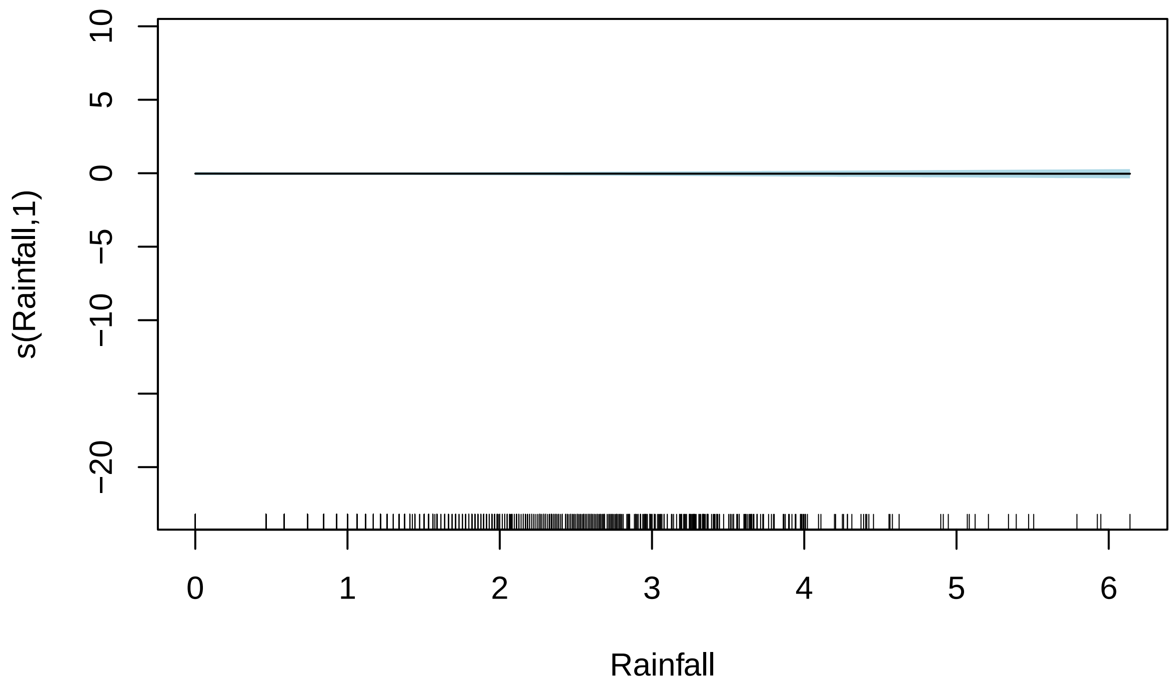

s(SO2,1):Period3
10
5
0
−5
−10
−20
0.000
0.001
0.002
0.003
0.004
SO2

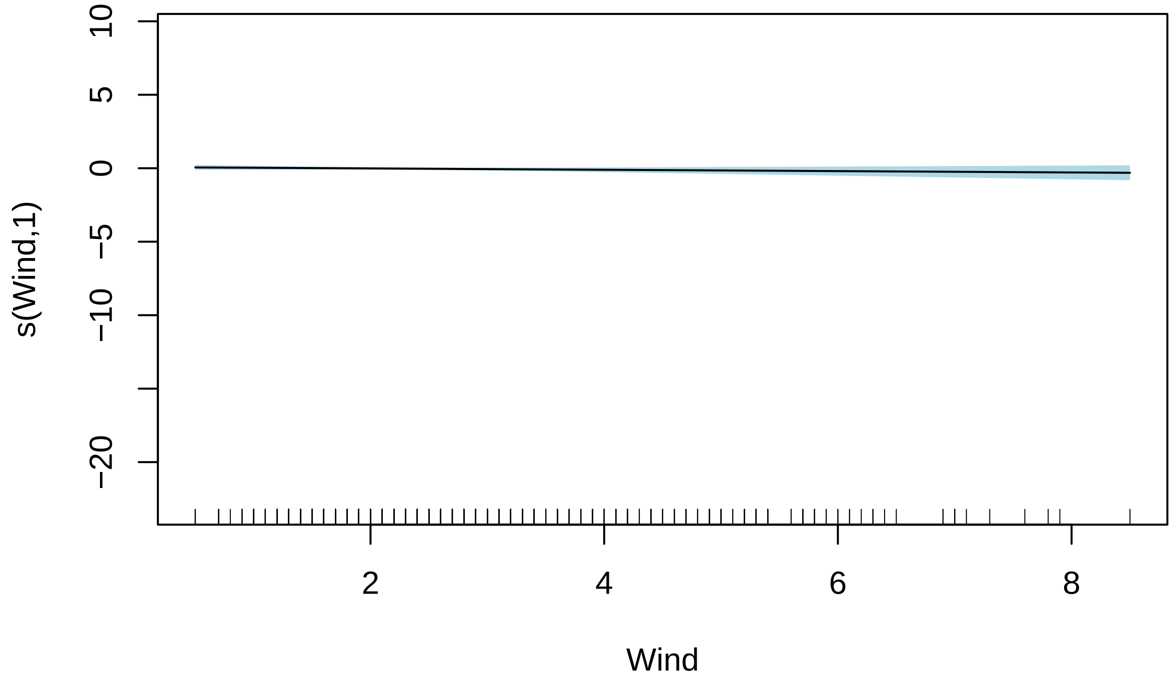

10
5
0
−5
−10
−20
s(O3,1.32):Period1
0
50
100
150
200
250
300
350
O3

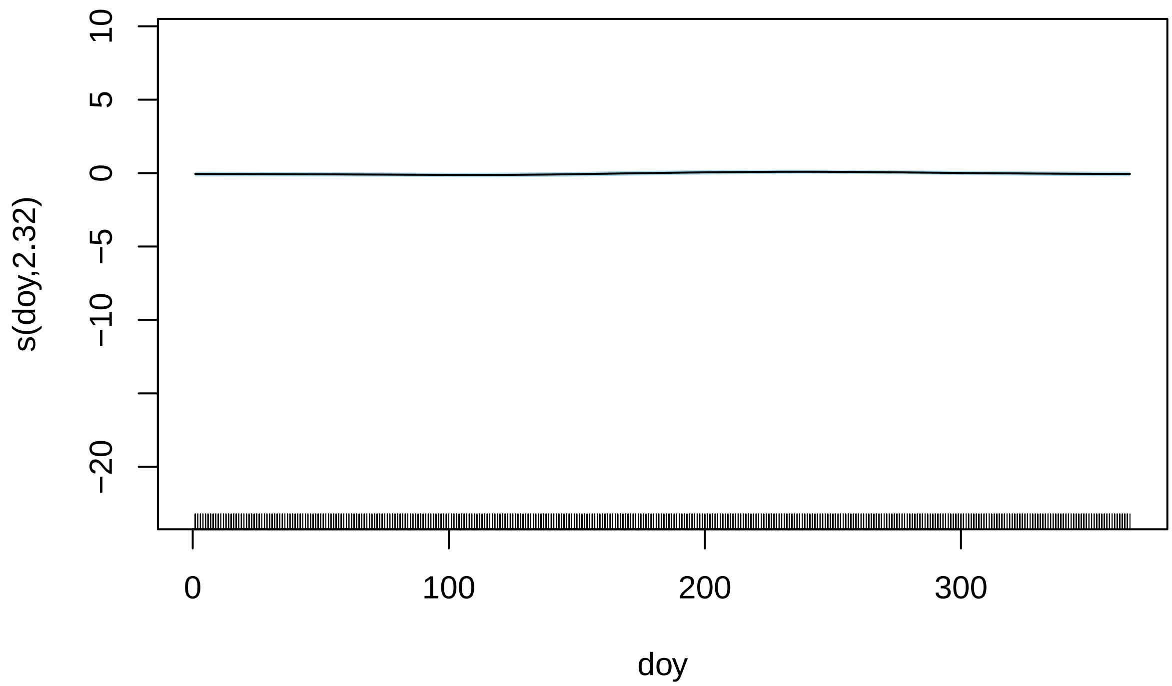

s(O3,1.23):Period2
10
5
0
−5
−10
−20
0
50
100
150
200
250
300
350
O3

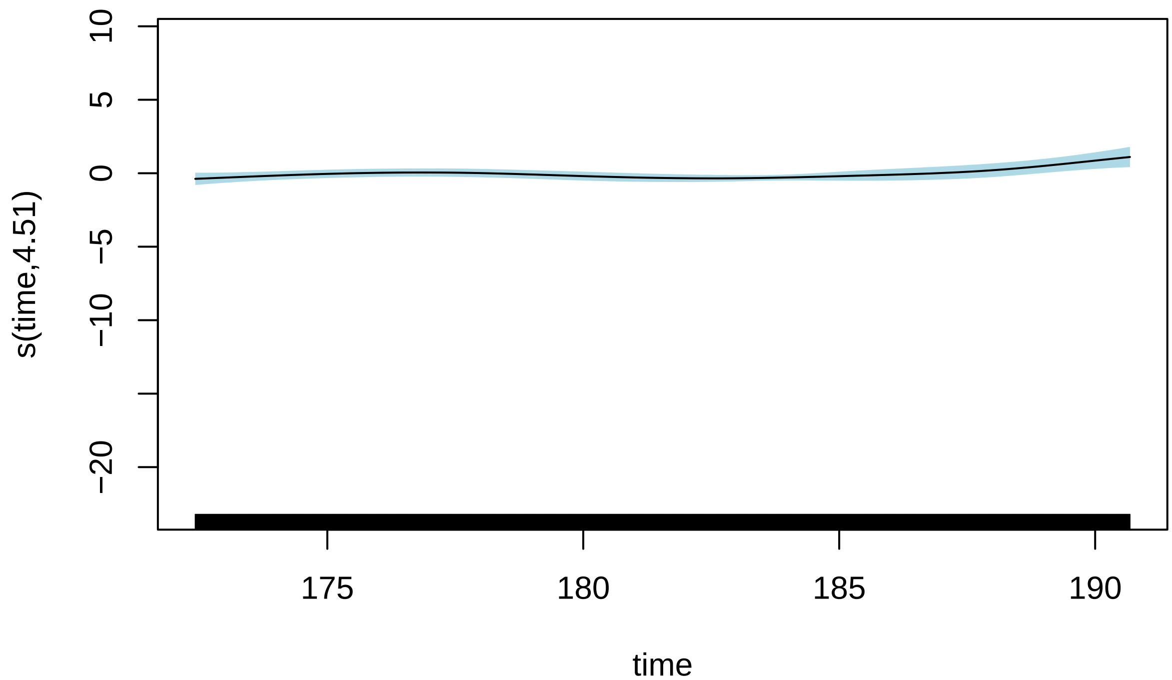

10
5
0
−5
−10
−20
s(O3,2.12):Period3
0
50
100
150
200
250
300
350
O3


s(Mean,1)
10
5
0
−5
−10
−20
24
26
28
30
Mean

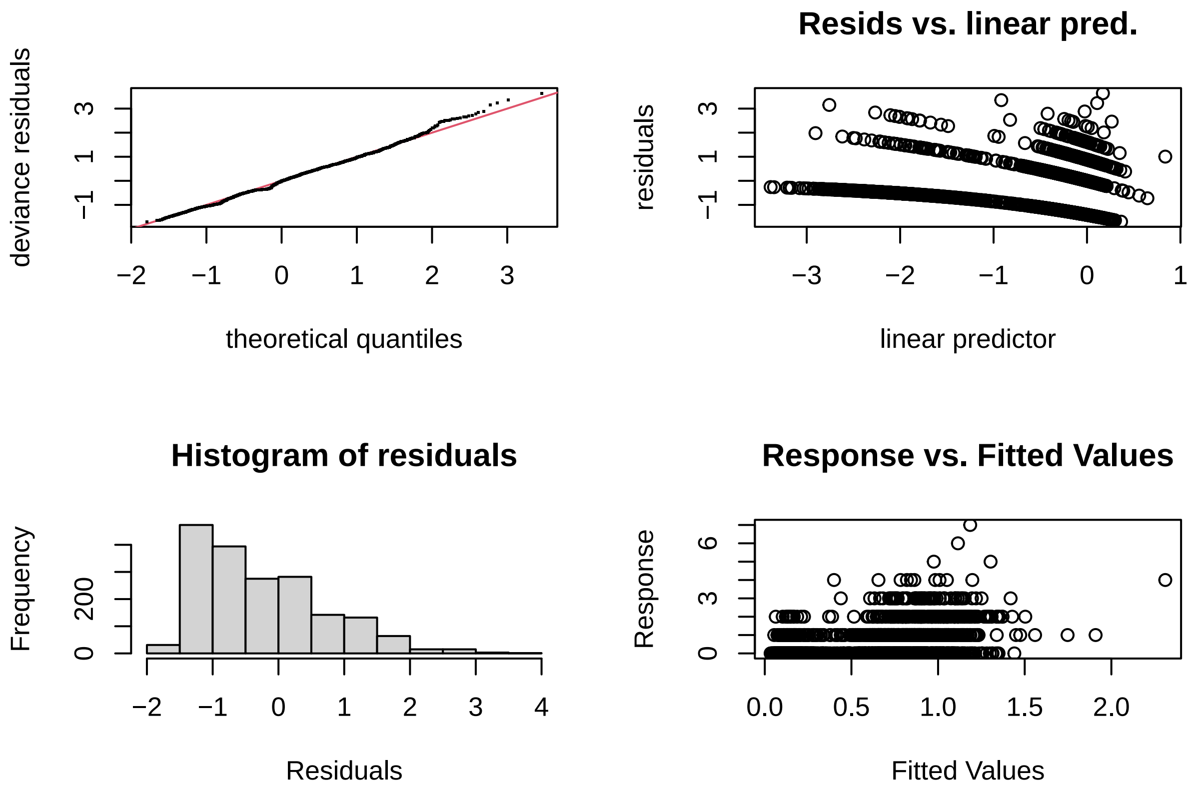

s(Rainfall,1)
10
5
0
−5
−10
−20
0
1
2
3
4
5
6
Rainfall


s(Wind,1)
10
5
0
−5
−10
−20
2
4
6
8
Wind


s(doy,2.32)
10
5
0
−5
−10
−20
0
100
200
300
doy




```
gam.check(kb_LRTI_period)
```




```
##
## Method: REML   Optimizer: outer newton
## full convergence after 10 iterations.
## Gradient range [-0.0003609757,0.0009960262]
## (score 1771.687 & scale 1).
## Hessian positive definite, eigenvalue range [1.7294e-05,0.8239119].
## Model rank =  537 / 537
##
## Basis dimension (k) checking results. Low p-value (k-index<1) may
## indicate that k is too low, especially if edf is close to k'.
##
##                      k'    edf k-index p-value
## s(PM10):Period1   9.00   2.64    0.95   0.655
## s(PM10):Period2   9.00   1.00    0.95   0.715
## s(PM10):Period3   9.00   2.63    0.95   0.660
## s(NO2):Period1    9.00   1.00    0.94   0.645
## s(NO2):Period2    9.00   1.00    0.94   0.590
## s(NO2):Period3    9.00   1.01    0.94   0.660
## s(CO3):Period1    9.00   1.01    0.90   0.045 *
## s(CO3):Period2    9.00   2.79    0.90   0.025 *
## s(CO3):Period3    9.00   1.00    0.90   0.025 *
## s(SO2):Period1    9.00   2.08    0.87   0.005 **
## s(SO2):Period2    9.00   1.00    0.87  <2e-16 ***
## s(SO2):Period3    9.00   1.00    0.87  <2e-16 ***
## s(O3):Period1     9.00   1.32    0.95   0.805
## s(O3):Period2     9.00   1.23    0.95   0.770
```

```
## s(O3):Period3     9.00   2.12    0.95   0.795
## s(Mean)           9.00   1.00    0.90   0.100 .
## s(Rainfall)       9.00   1.00    0.90   0.065 .
## s(Wind)           9.00   1.00    0.93   0.325
## s(doy)          363.00   2.32    0.93   0.380
## s(time)           9.00   4.51    0.88  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

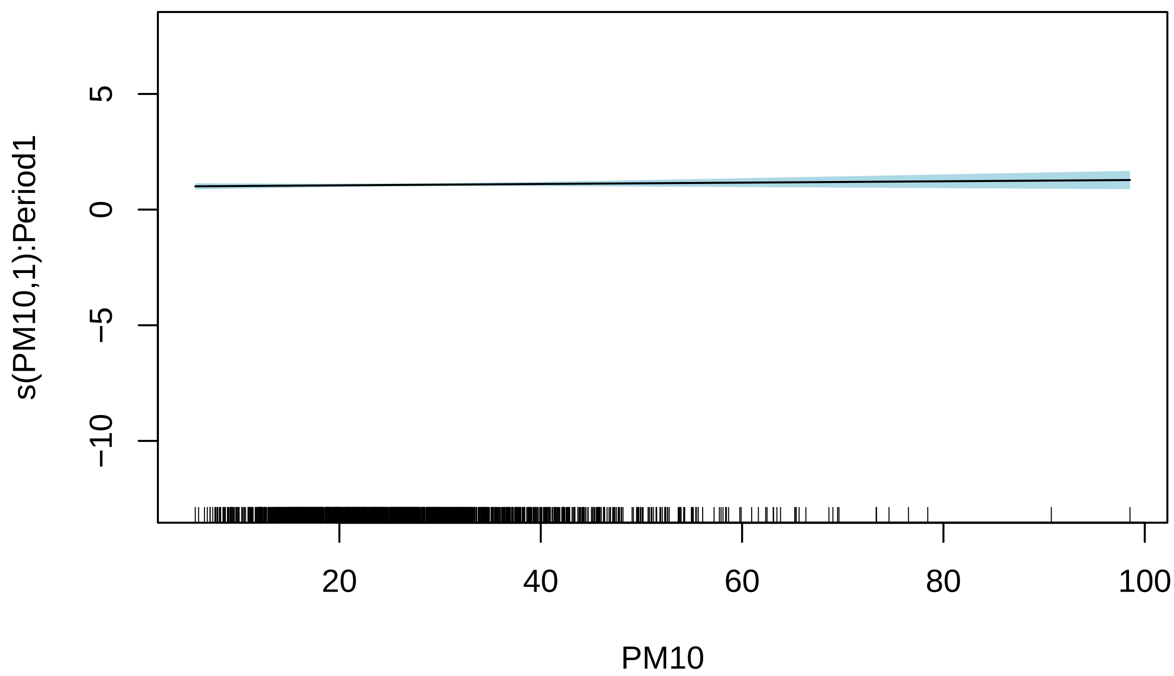
```
kb_sx_period <- gam(Symptoms_Signs ~ s(PM10, by=Period) + s(NO2, by=Period) + s(CO3, by=Period) + s(SO2
summary(kb_sx_period)
```

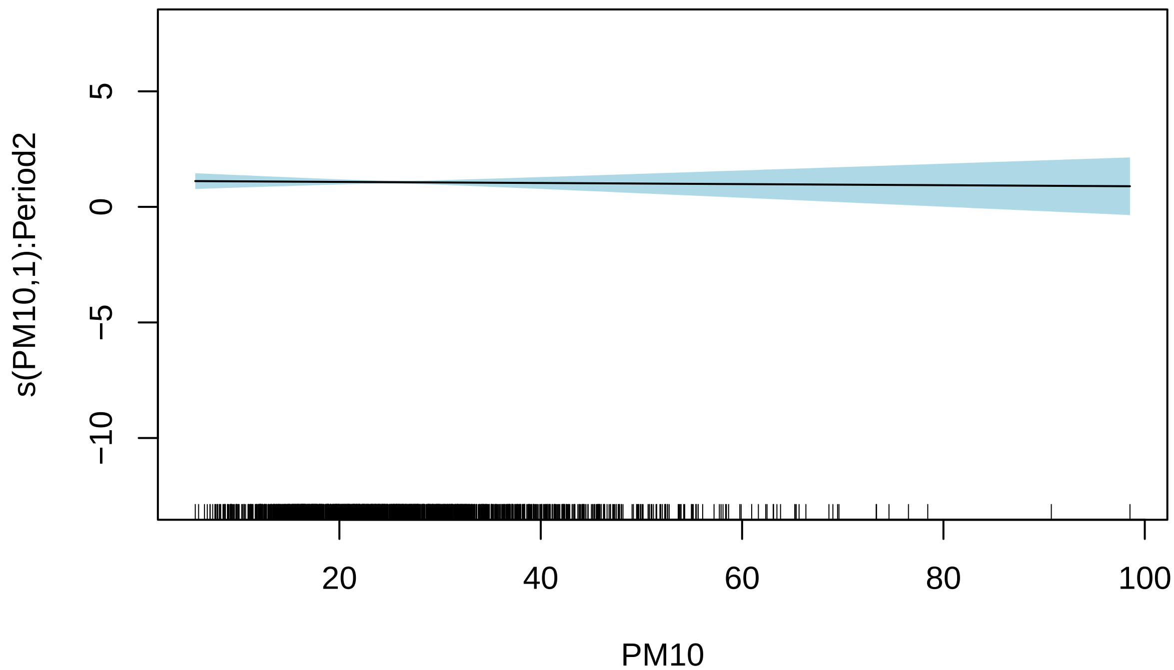
### KB SX by period

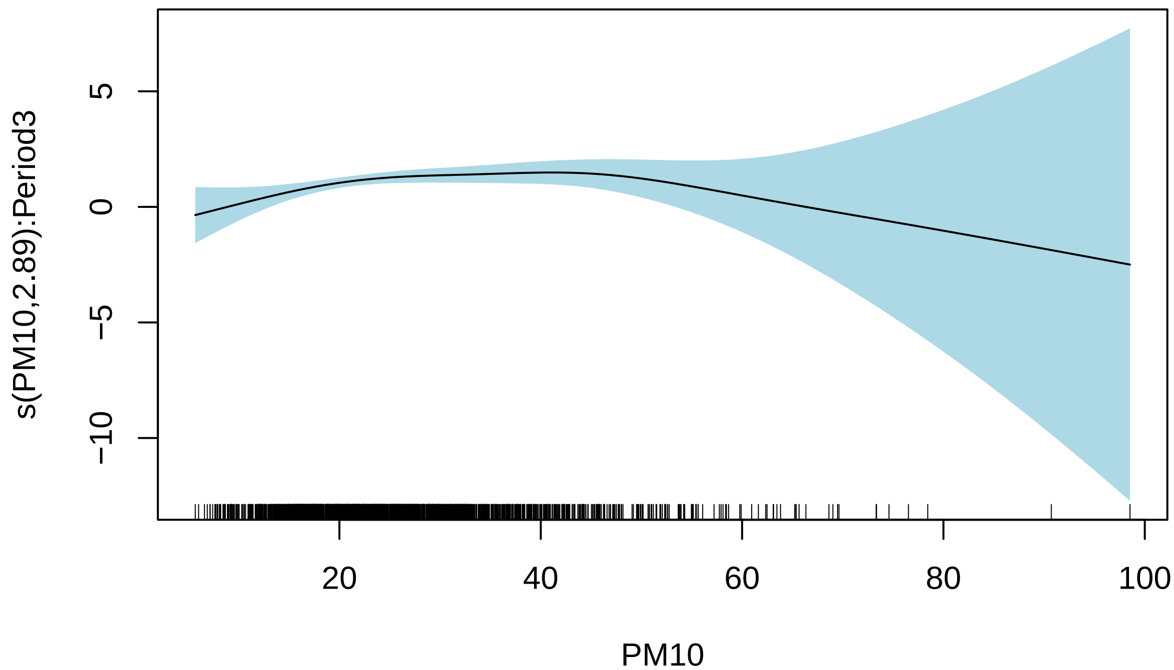
```
##
## Family: poisson
## Link function: log
##
## Formula:
## Symptoms_Signs ~ s(PM10, by = Period) + s(NO2, by = Period) +
##     s(CO3, by = Period) + s(SO2, by = Period) + s(O3, by = Period) +
##     s(Mean) + s(Rainfall) + s(Wind) + s(doy, bs = "cc", k = 365) +
##     s(time, bs = "bs") + Period
##
## Parametric coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)   1.0664     0.1319   8.087 6.13e-16 ***
## Period2      -2.7895     0.3482  -8.010 1.14e-15 ***
## Period3      -3.2148     0.4932  -6.519 7.09e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Approximate significance of smooth terms:
##                     edf  Ref.df  Chi.sq  p-value
## s(PM10):Period1  1.000   1.001   1.134 0.287096
## s(PM10):Period2  1.001   1.001   0.080 0.778554
## s(PM10):Period3  2.887   3.608   8.961 0.056127 .
## s(NO2):Period1   1.851   2.288   2.036 0.419836
## s(NO2):Period2   4.523   5.351   8.270 0.185326
## s(NO2):Period3   2.726   3.108   7.130 0.077017 .
## s(CO3):Period1   3.708   4.652   7.552 0.148672
## s(CO3):Period2   1.811   2.292   8.478 0.020666 *
## s(CO3):Period3   1.000   1.000  11.979 0.000539 ***
## s(SO2):Period1   1.000   1.001   1.029 0.310672
## s(SO2):Period2   1.659   2.128   3.446 0.196551
## s(SO2):Period3   1.768   2.052   5.872 0.050830 .
## s(O3):Period1    3.704   4.581   8.694 0.092305 .
## s(O3):Period2    1.002   1.004   0.134 0.716111
## s(O3):Period3    1.000   1.001   0.471 0.492706
## s(Mean)          2.324   2.995   5.935 0.117411
## s(Rainfall)      1.000   1.000   0.024 0.876996
```

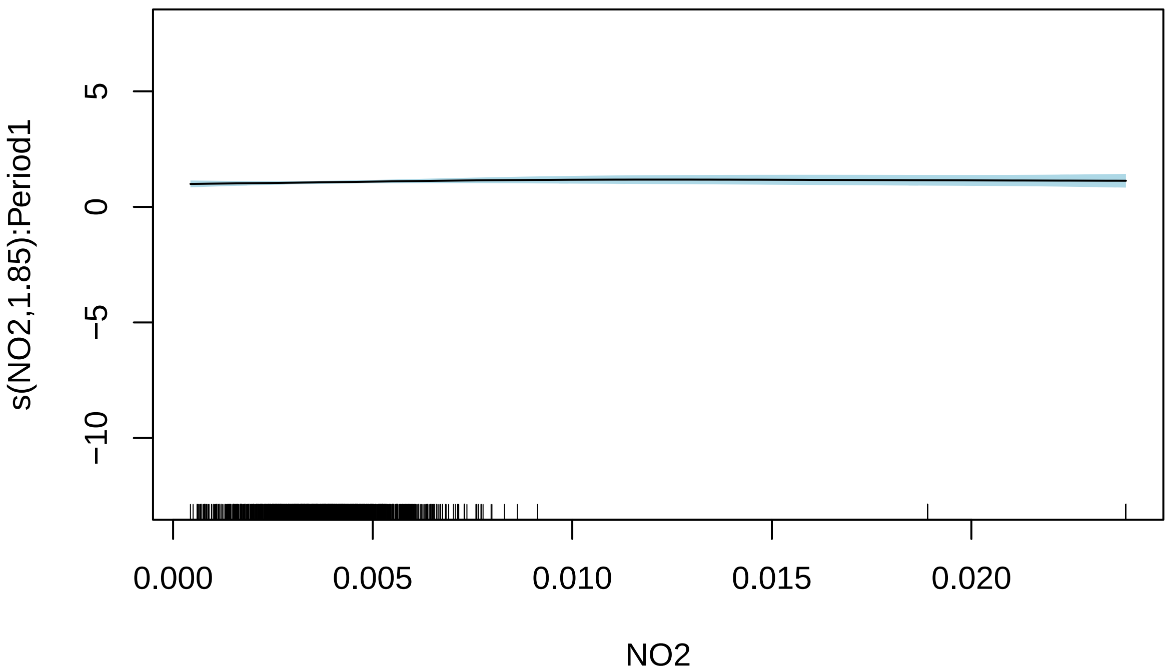
```
## s(Wind)           1.001   1.001   0.456 0.500217
## s(doy)           12.807 363.000  50.129 1.26e-07 ***
## s(time)           8.180   8.706 120.161  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-sq.(adj) =  0.304   Deviance explained = 37.3%
## -REML = 2581.2  Scale est. = 1         n = 1827
```

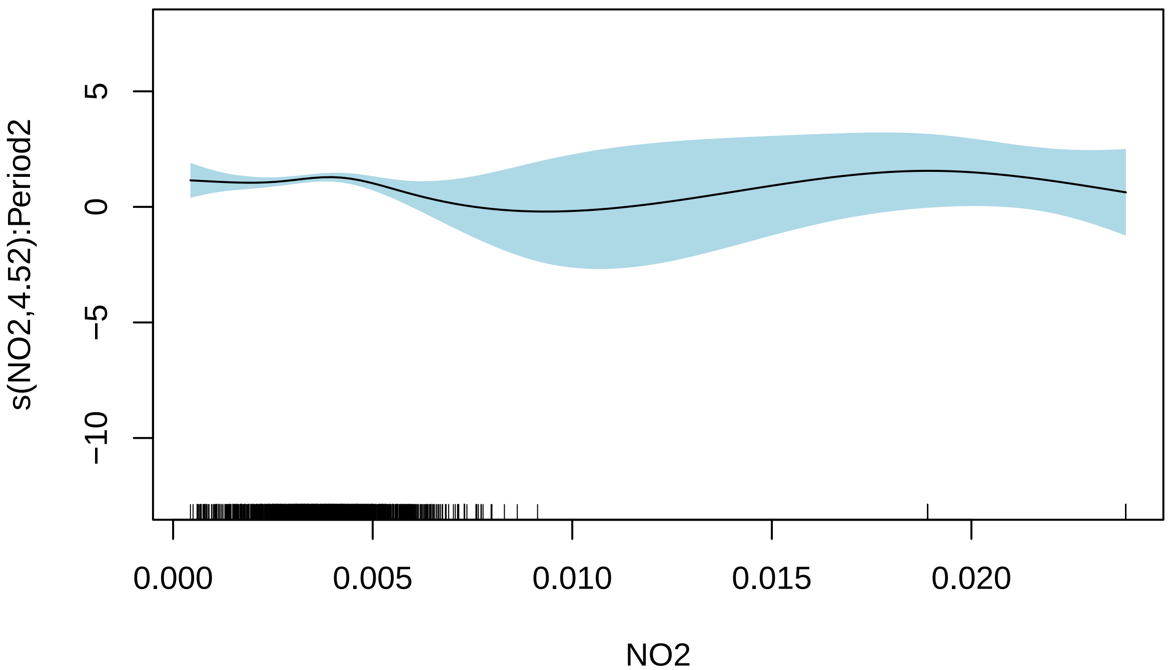
```
plot.gam(kb_sx_period, shade = TRUE, shade.col = "lightblue", shift = coef(kb_sx_period)[1],  seWithMean
```



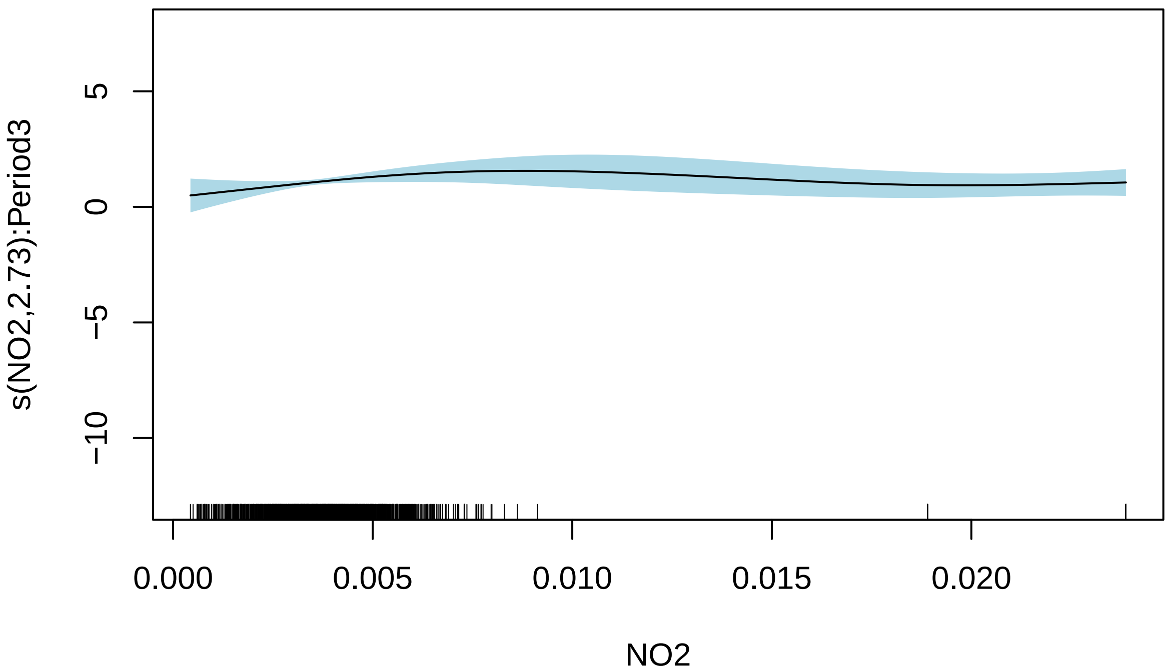

5
0
−5
−10
s(PM10,1):Period2
20
40
60
80
100
PM10

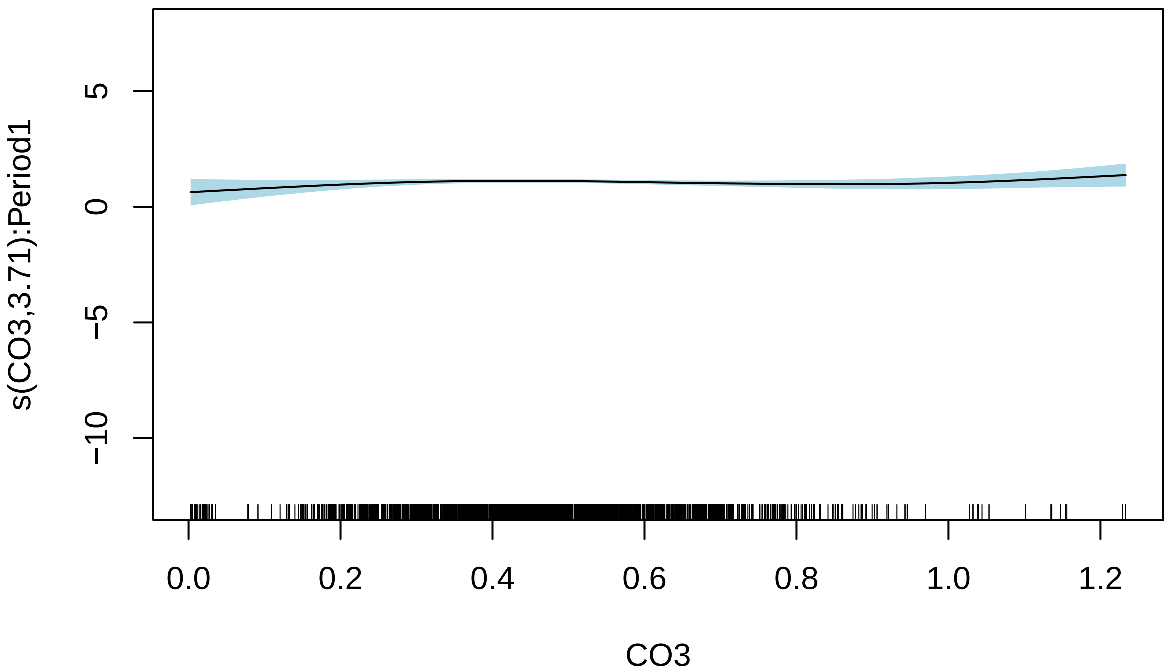

s(PM10,2.89):Period3
5
0
−5
−10
20
40
60
80
100
PM10

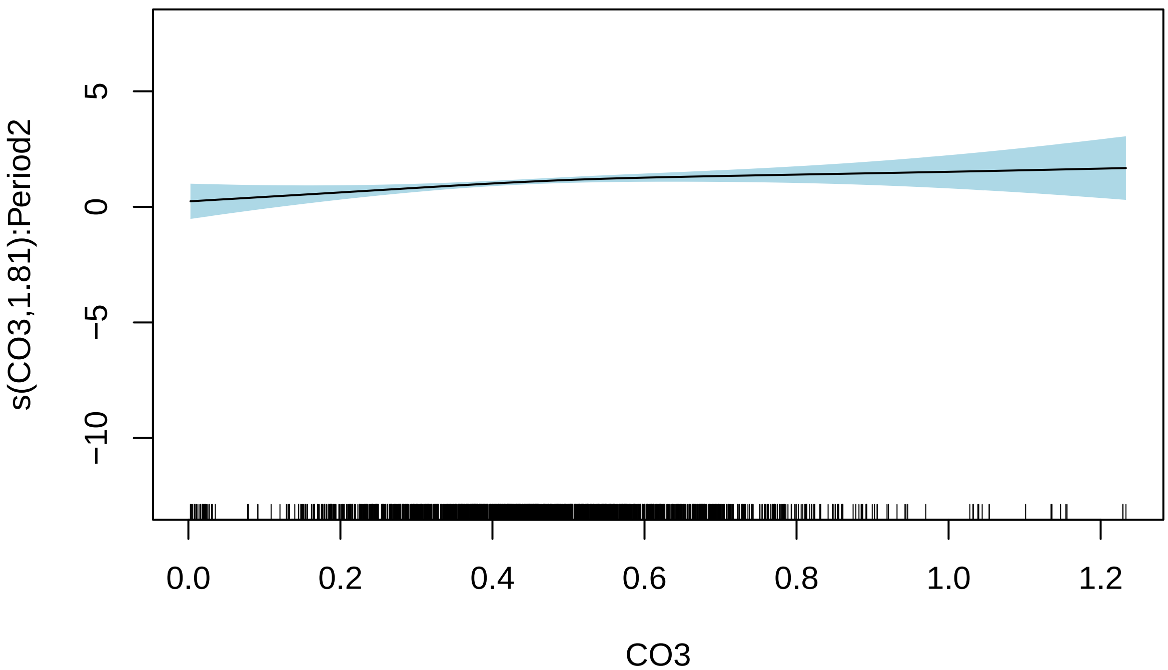

s(NO2,1.85):Period1
5
0
−5
−10
0.000
0.005
0.010
0.015
0.020
NO2

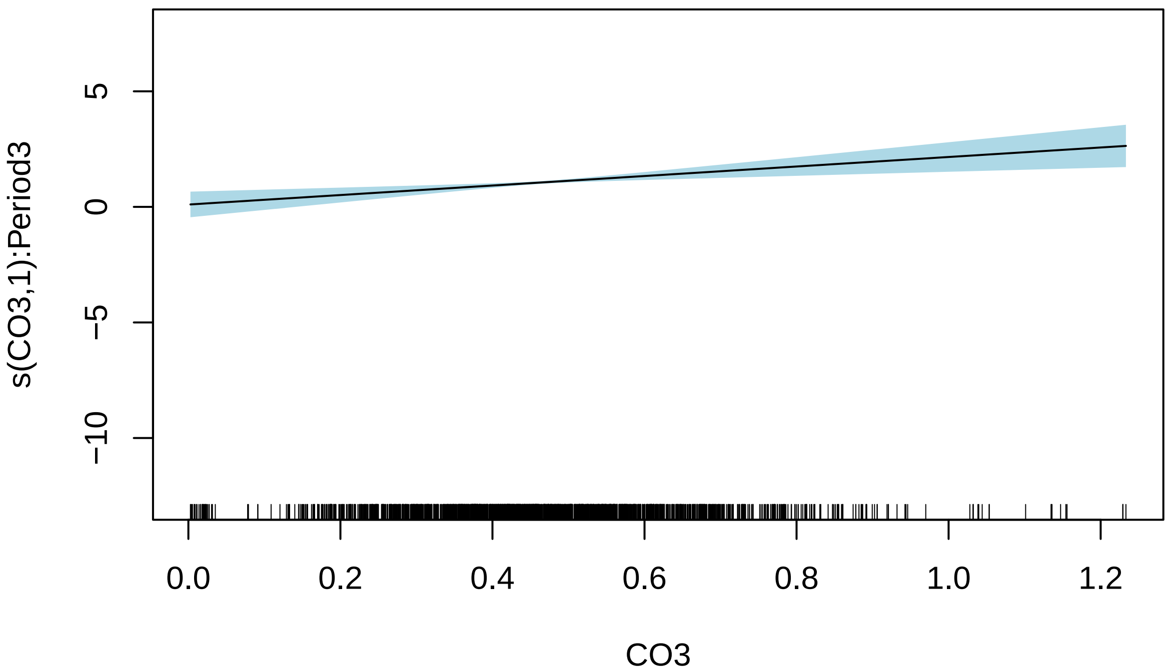


s(NO2,4.52):Period2
5
0
−5
−10
0.000
0.005
0.010
0.015
0.020
NO2

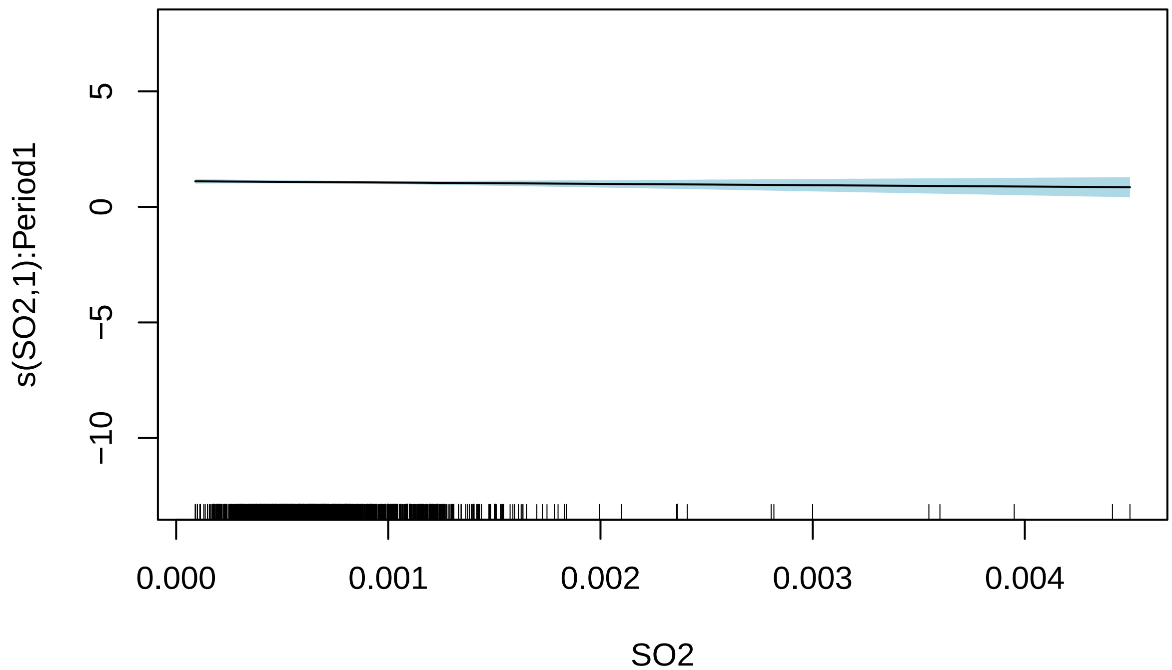

s(NO2,2.73):Period3
5
0
−5
−10
0.000
0.005
0.010
0.015
0.020
NO2

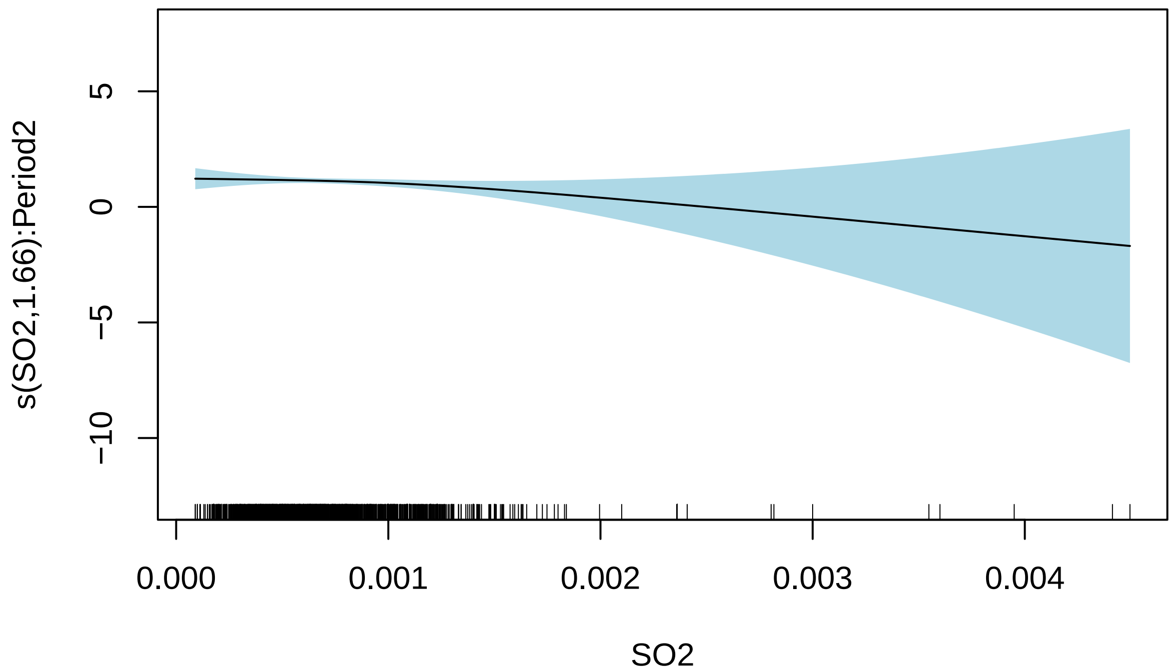

s(CO3,3.71):Period1
5
0
−5
−10
0.0
0.2
0.4
0.6
0.8
1.0
1.2
CO3


s(CO3,1.81):Period2
5
0
−5
−10
0.0
0.2
0.4
0.6
0.8
1.0
1.2
CO3

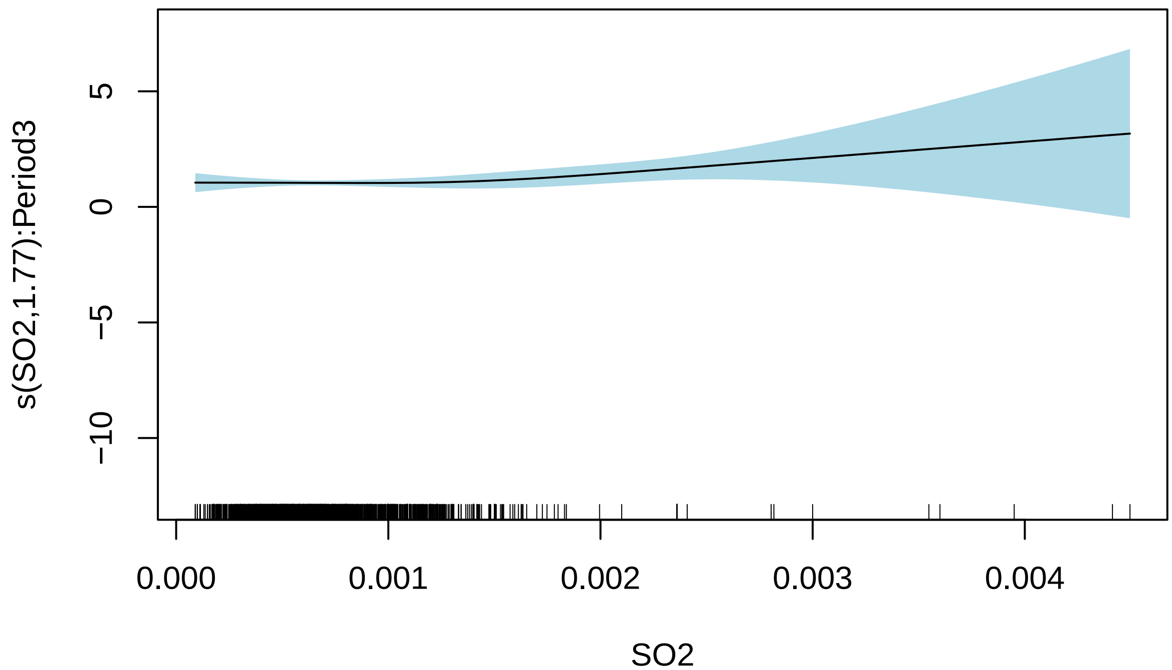


s(CO3,1):Period3
5
0
−5
−10
0.0
0.2
0.4
0.6
0.8
1.0
1.2
CO3

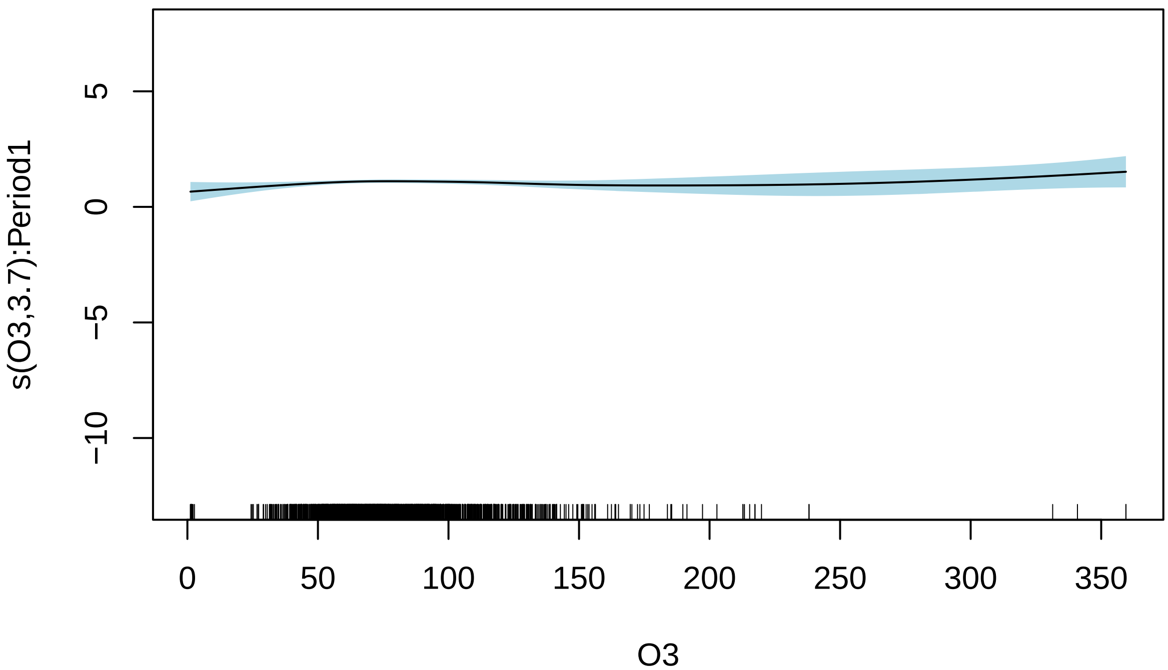

s(SO2,1):Period1
5
0
−5
−10
0.000
0.001
0.002
0.003
0.004
SO2

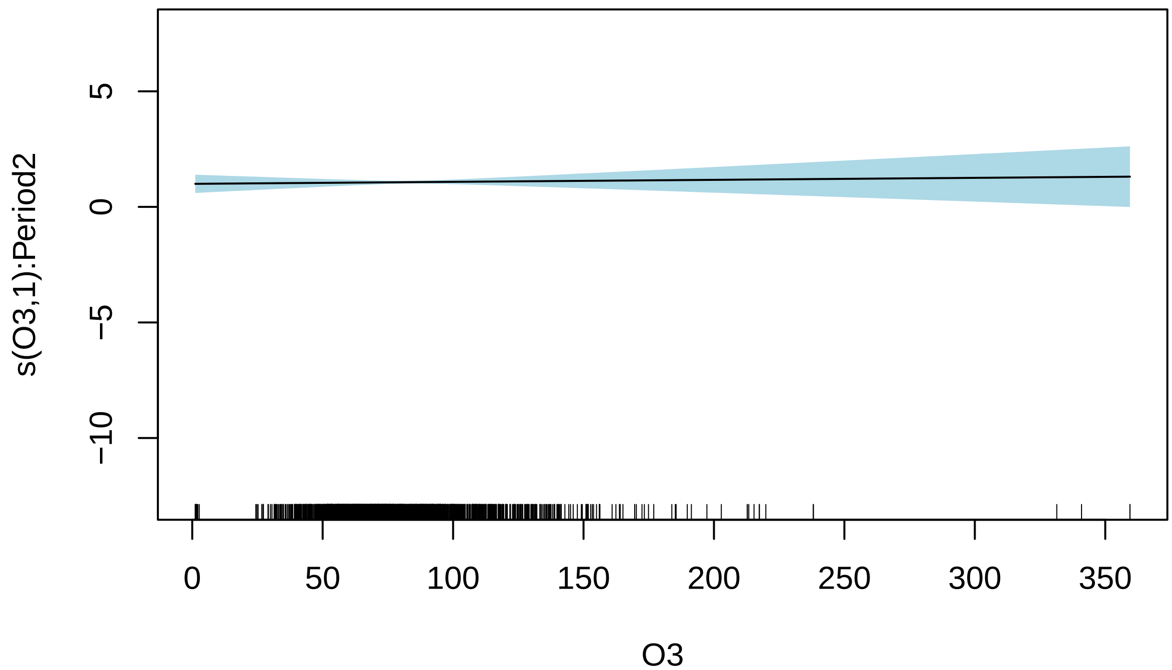


s(SO2,1.66):Period2
5
0
−5
−10
0.000
0.001
0.002
0.003
0.004
SO2

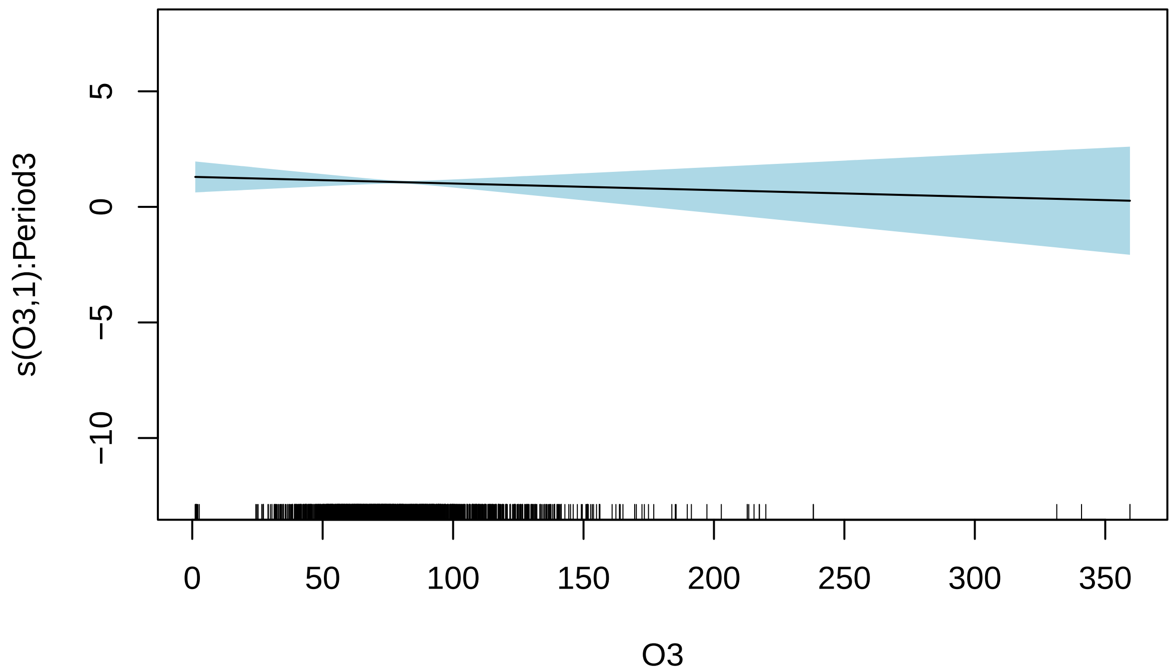


s(SO2,1.77):Period3
5
0
−5
−10
0.000
0.001
0.002
0.003
0.004
SO2

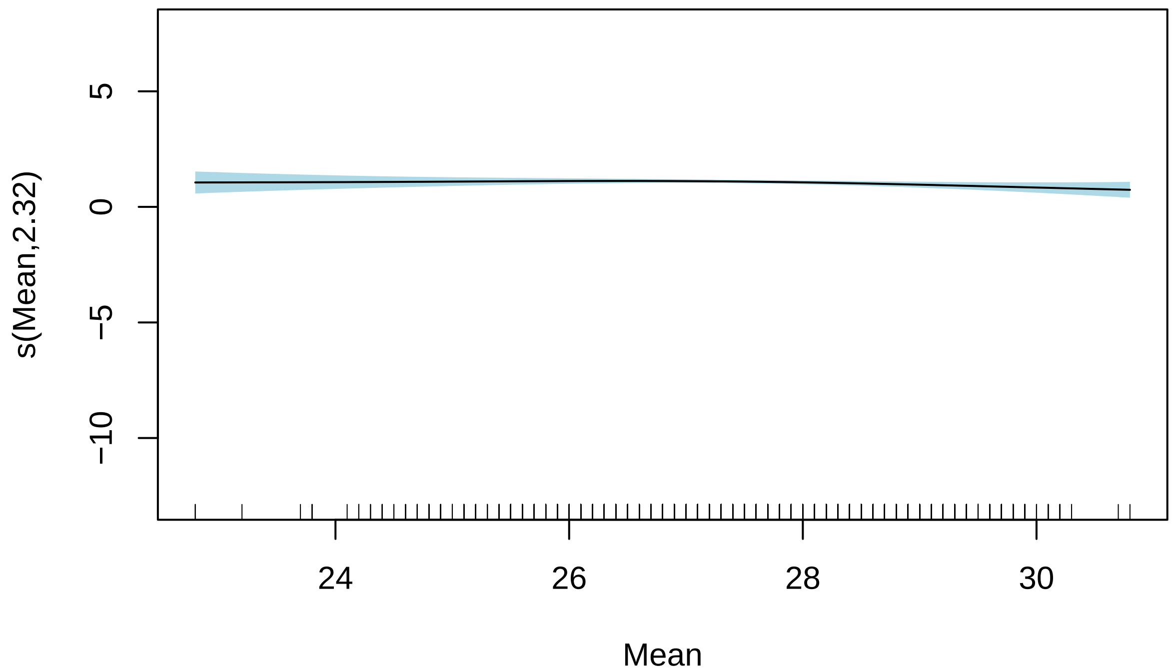

s(O3,3.7):Period1
5
0
−5
−10
0
50
100
150
200
250
300
350
O3

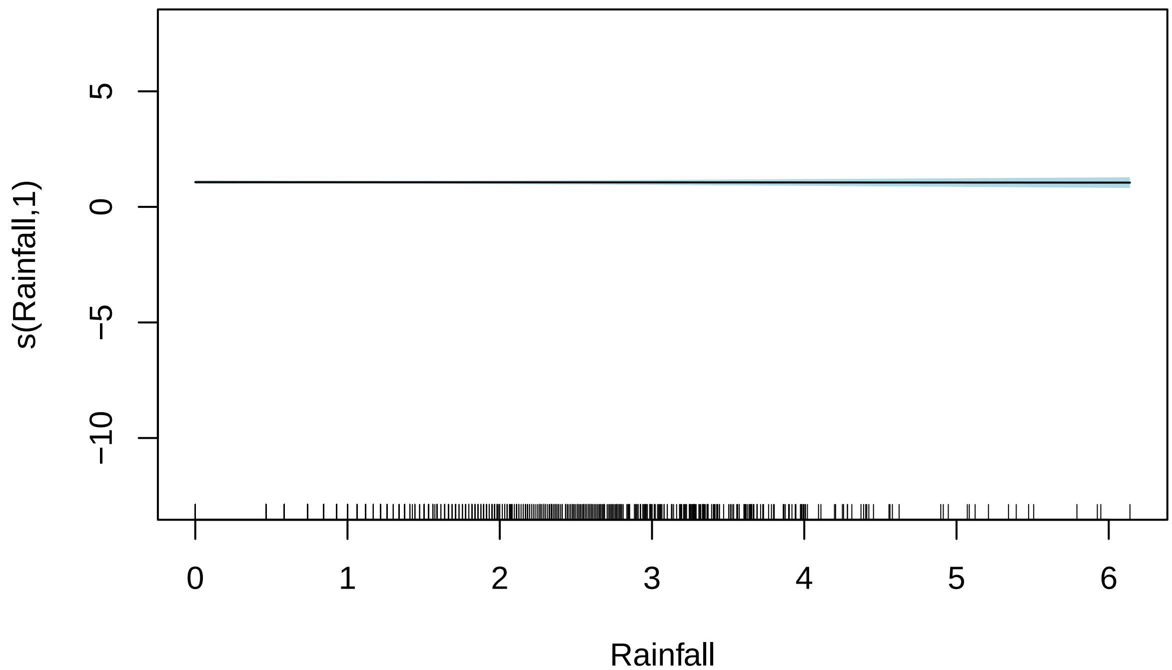


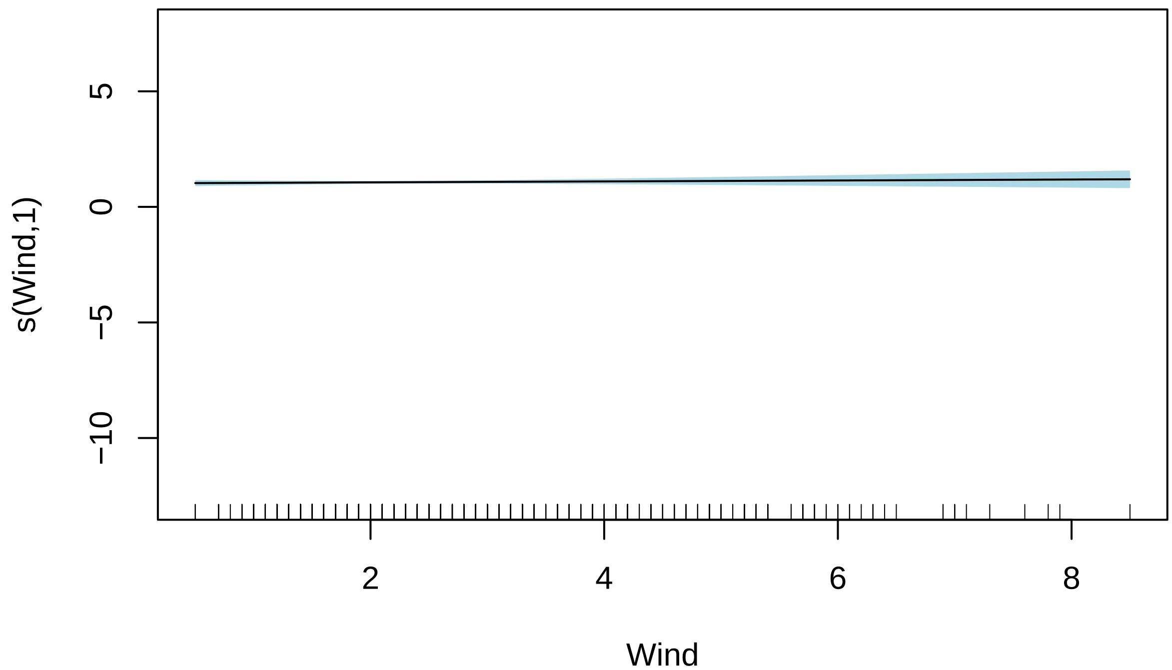
s(O3,1):Period2
5
0
−5
−10
0
50
100
150
200
250
300
350
O3

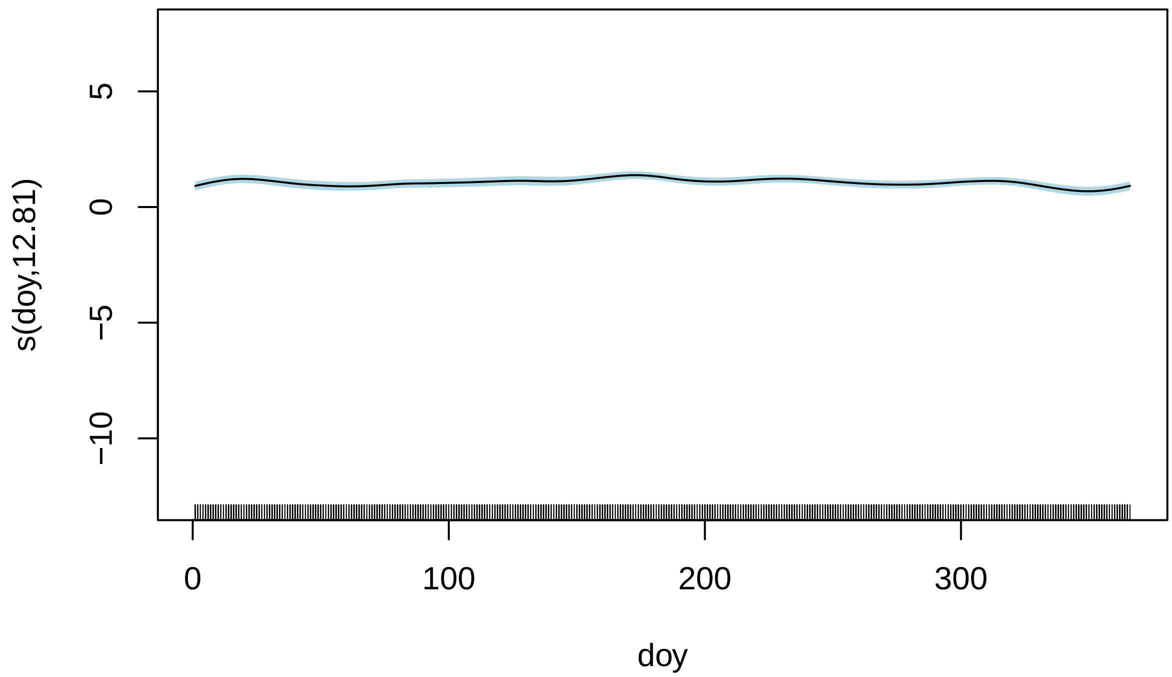


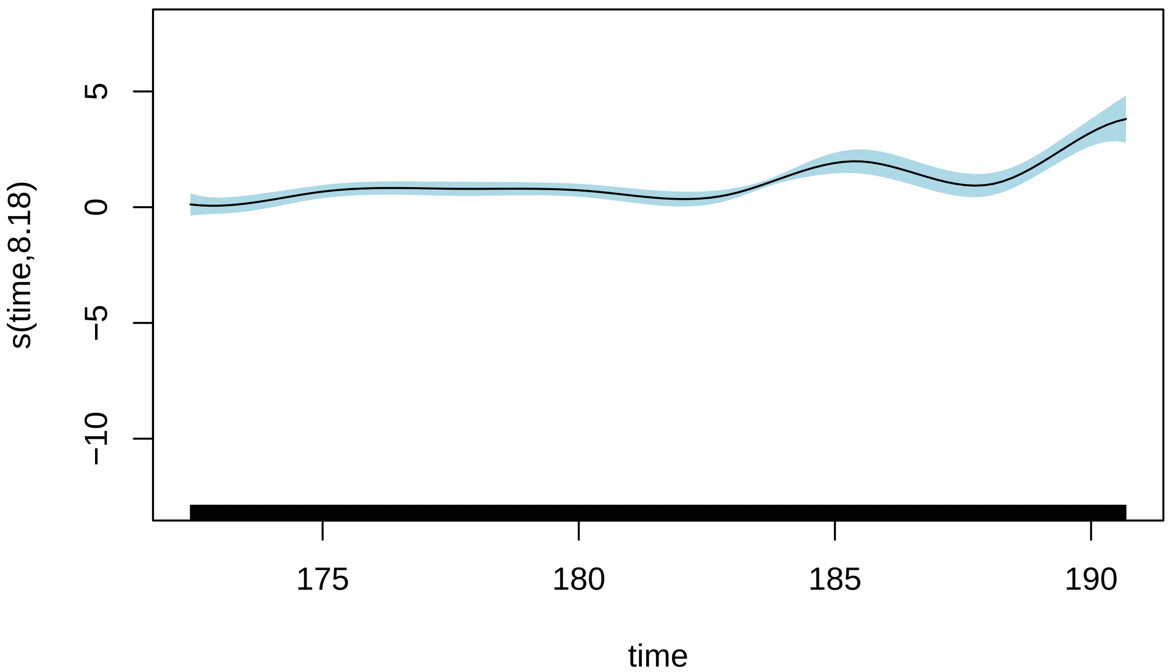
s(O3,1):Period3
5
0
−5
−10
0
50
100
150
200
250
300
350
O3


5
0
−5
−10
s(Mean,2.32)
24
26
28
30
Mean

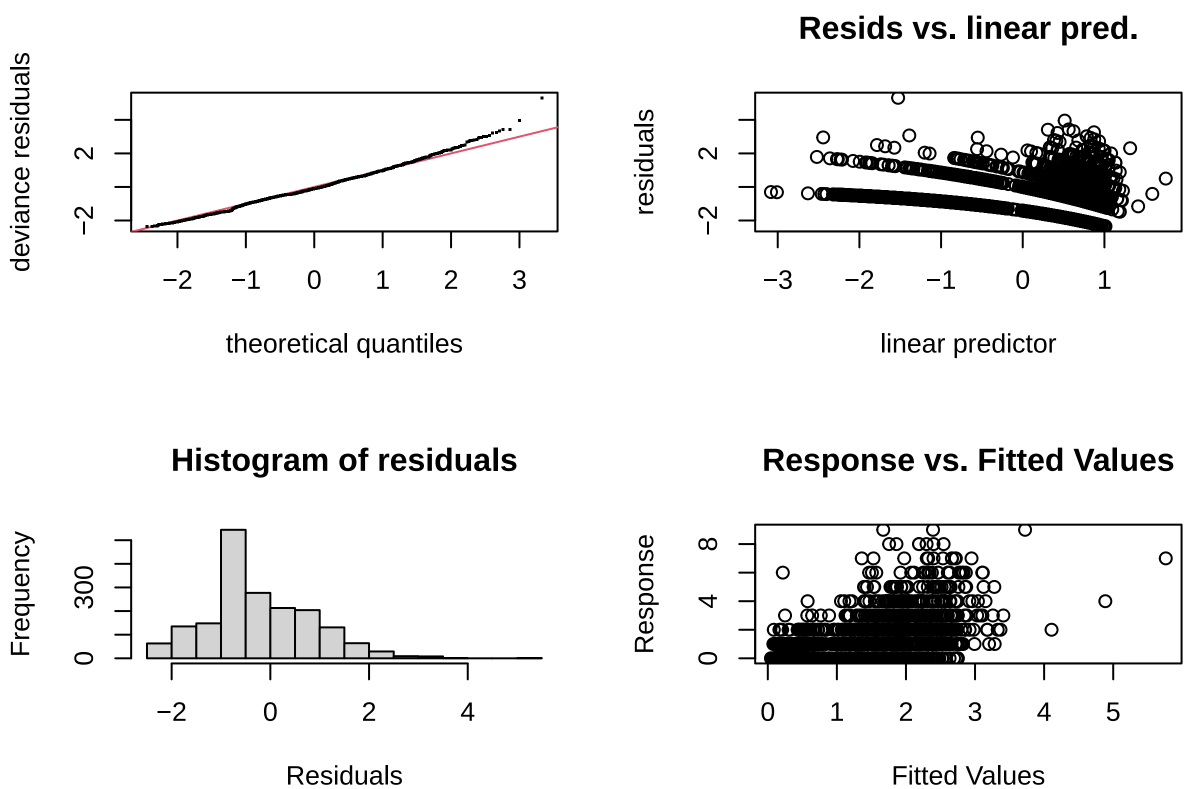

s(Rainfall,1)
5
0
−5
−10
0
1
2
3
4
5
6
Rainfall


s(Wind,1)
5
0
−5
−10
2
4
6
8
Wind


s(doy,12.81)
5
0
−5
−10
0
100
200
300
doy




```
gam.check(kb_sx_period)
```




```
##
## Method: REML   Optimizer: outer newton
## full convergence after 10 iterations.
## Gradient range [-0.0005808806,1.213109e-05]
## (score 2581.201 & scale 1).
## Hessian positive definite, eigenvalue range [5.918907e-05,2.48077].
## Model rank =  537 / 537
##
## Basis dimension (k) checking results. Low p-value (k-index<1) may
## indicate that k is too low, especially if edf is close to k'.
##
##                      k'    edf k-index p-value
## s(PM10):Period1    9.00   1.00    0.93   0.145
## s(PM10):Period2    9.00   1.00    0.93   0.195
## s(PM10):Period3    9.00   2.89    0.93   0.110
## s(NO2):Period1     9.00   1.85    0.94   0.210
## s(NO2):Period2     9.00   4.52    0.94   0.220
## s(NO2):Period3     9.00   2.73    0.94   0.170
## s(CO3):Period1     9.00   3.71    0.95   0.350
## s(CO3):Period2     9.00   1.81    0.95   0.365
## s(CO3):Period3     9.00   1.00    0.95   0.315
## s(SO2):Period1     9.00   1.00    0.91   0.035 *
## s(SO2):Period2     9.00   1.66    0.91   0.020 *
## s(SO2):Period3     9.00   1.77    0.91   0.015 *
## s(O3):Period1      9.00   3.70    1.00   0.970
## s(O3):Period2      9.00   1.00    1.00   0.935
```

```
## s(O3):Period3     9.00   1.00    1.00   0.960
## s(Mean)           9.00   2.32    0.91   0.010 **
## s(Rainfall)       9.00   1.00    0.90  <2e-16 ***
## s(Wind)           9.00   1.00    0.90   0.010 **
## s(doy)          363.00  12.81    0.96   0.480
## s(time)           9.00   8.18    0.89  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

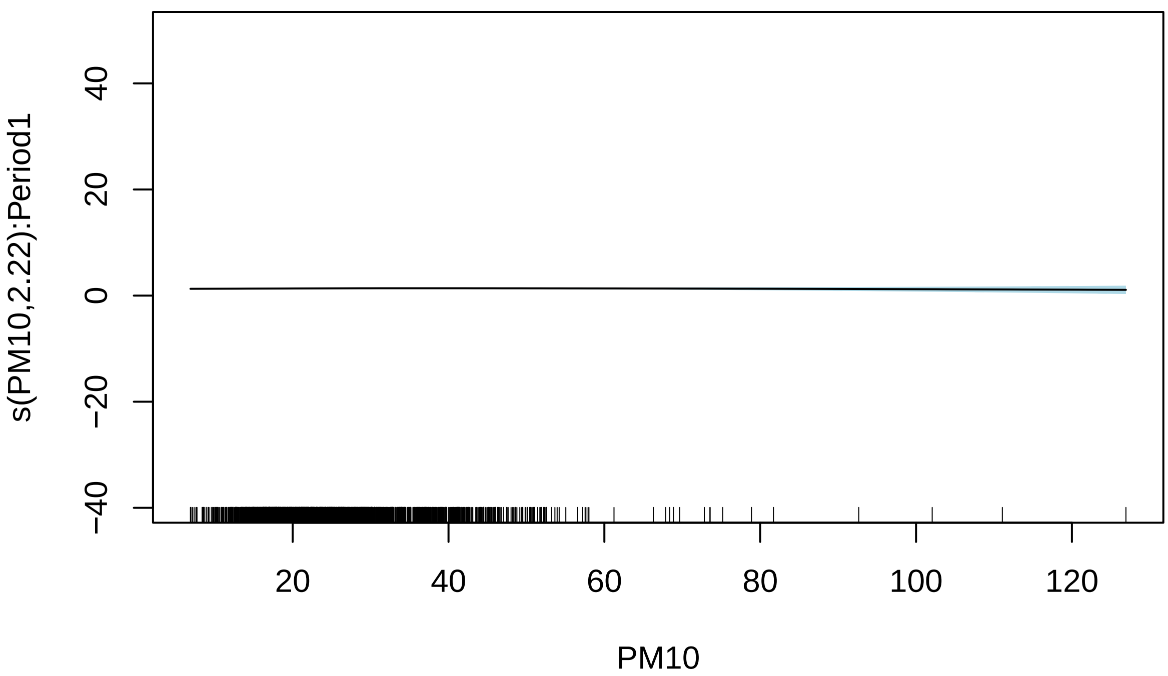
```
jb_URTI_period <- gam(URTI ~ s(PM10, by=Period) + s(NO2, by=Period) + s(CO3, by=Period) + s(SO2, by=Peri
summary(jb_URTI_period)
```

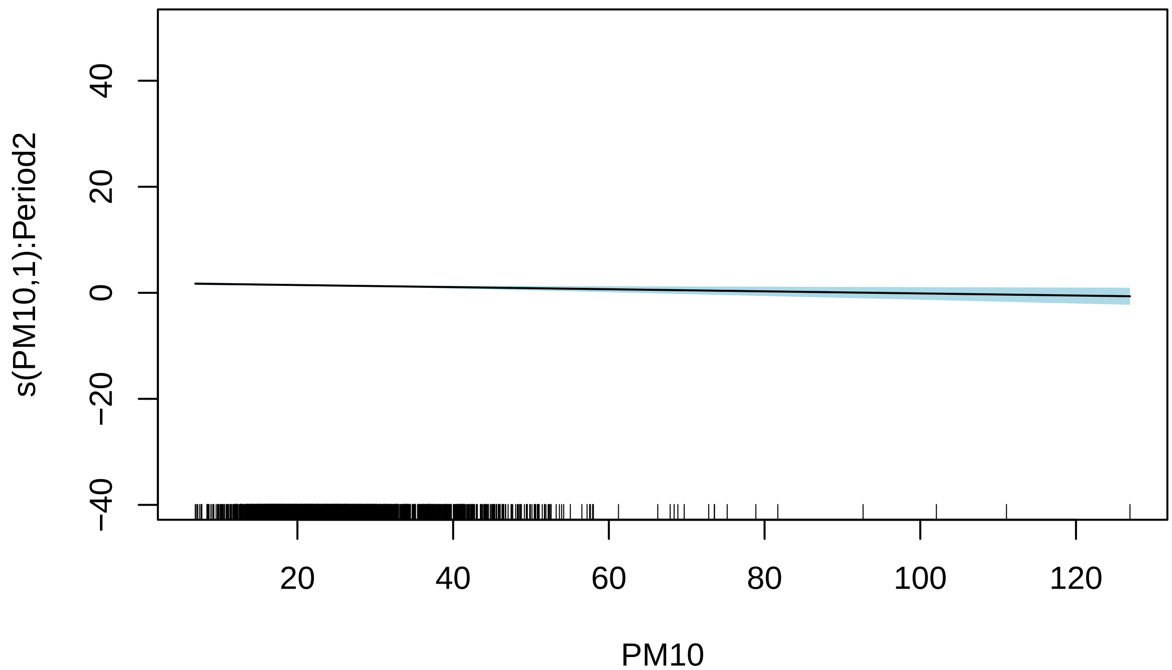
### JB URTI by Period

```
##
## Family: poisson
## Link function: log
##
## Formula:
## URTI ~ s(PM10, by = Period) + s(NO2, by = Period) + s(CO3, by = Period) +
##     s(SO2, by = Period) + s(O3, by = Period) + s(Mean) + s(Rainfall) +
##     s(Wind) + s(doy, bs = "cc", k = 365) + s(time, bs = "bs") +
##     Period
##
## Parametric coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  1.35307    0.06089  22.221  < 2e-16 ***
## Period2     -1.81615    0.20616  -8.809  < 2e-16 ***
## Period3     -1.22615    0.29257  -4.191 2.78e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Approximate significance of smooth terms:
##                    edf  Ref.df Chi.sq  p-value
## s(PM10):Period1  2.216   2.854  2.823 0.454525
## s(PM10):Period2  1.000   1.001  6.473 0.010990 *
## s(PM10):Period3  1.000   1.000  2.708 0.099940 .
## s(NO2):Period1   2.125   2.697  2.861 0.414996
## s(NO2):Period2   7.447   7.950 33.261  2.4e-05 ***
## s(NO2):Period3   1.003   1.006  0.498 0.484211
## s(CO3):Period1   4.161   5.050 15.375 0.009432 **
## s(CO3):Period2   1.182   1.344  6.603 0.015904 *
## s(CO3):Period3   1.587   1.928  4.079 0.120194
## s(SO2):Period1   1.939   2.460  2.057 0.516429
## s(SO2):Period2   3.148   3.782 12.057 0.057687 .
## s(SO2):Period3   1.801   2.031  1.984 0.337325
## s(O3):Period1    1.000   1.000  0.636 0.425130
## s(O3):Period2    1.001   1.002  2.928 0.087080 .
## s(O3):Period3    1.000   1.000  1.937 0.164056
## s(Mean)          2.369   3.035  6.842 0.079414 .
## s(Rainfall)      1.616   2.007  1.718 0.419464
```

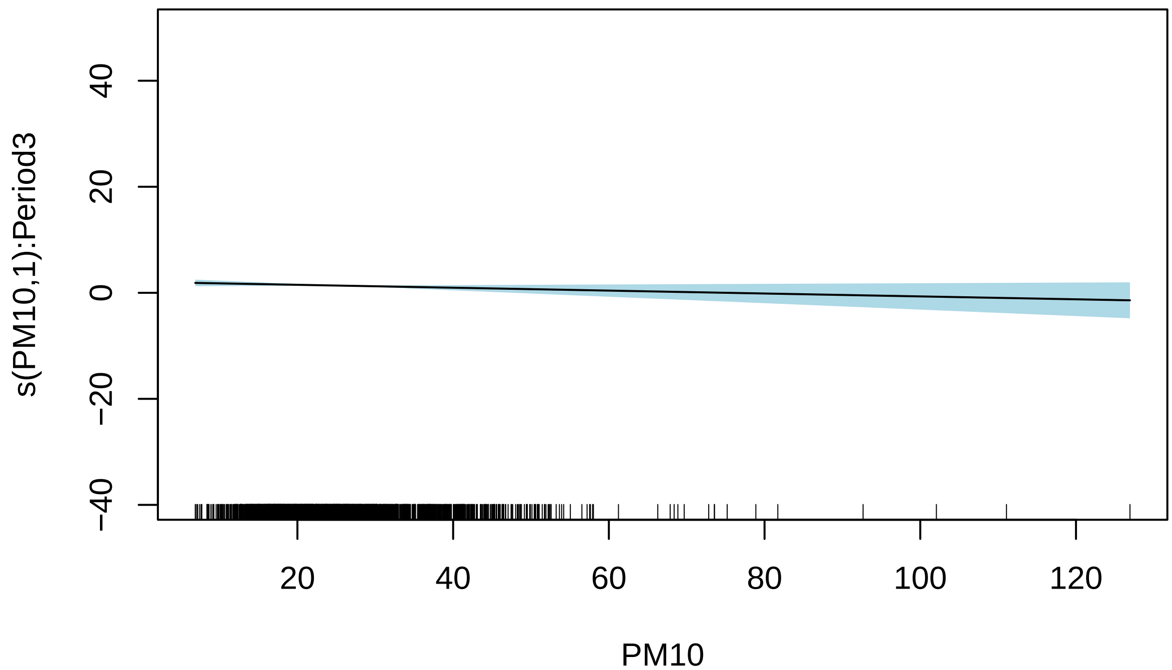
```
## s(Wind)          1.003   1.005  0.246 0.623475
## s(doy)          10.831 363.000 53.623  < 2e-16 ***
## s(time)          5.491   6.372 29.015 0.000155 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-sq.(adj) =  0.221   Deviance explained = 26.6%
## -REML = 3539.1  Scale est. = 1         n = 1827
```

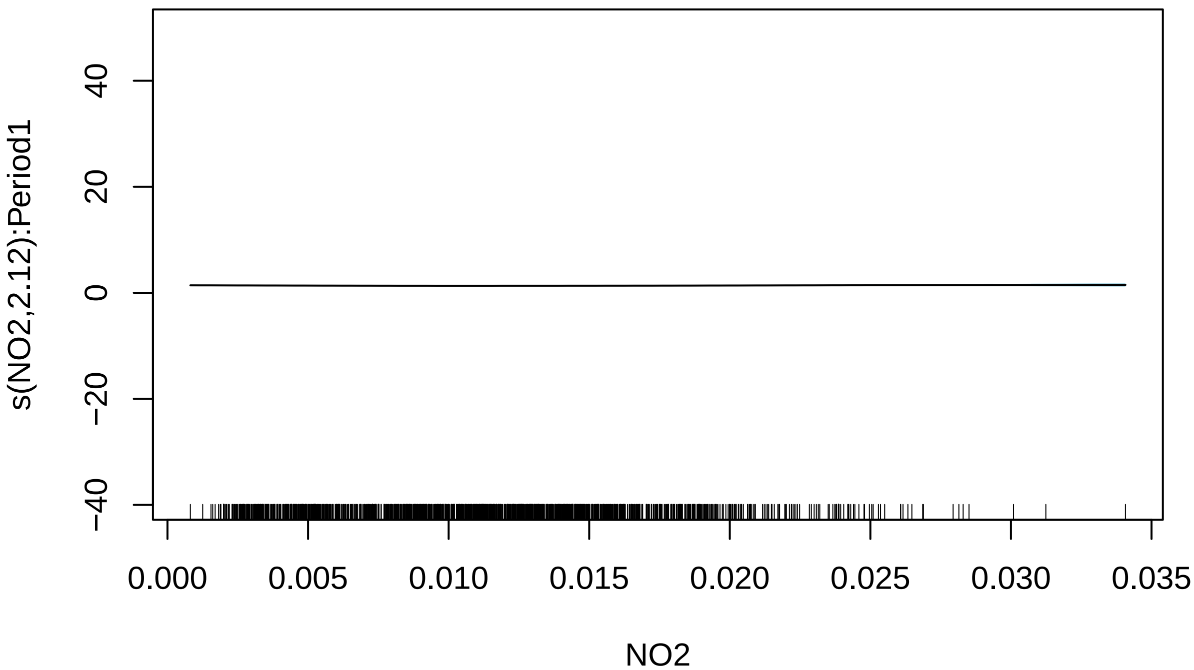
```
plot.gam(jb_URTI_period, shade = TRUE, shade.col = "lightblue", shift = coef(jb_URTI_period)[1],  seWit
```

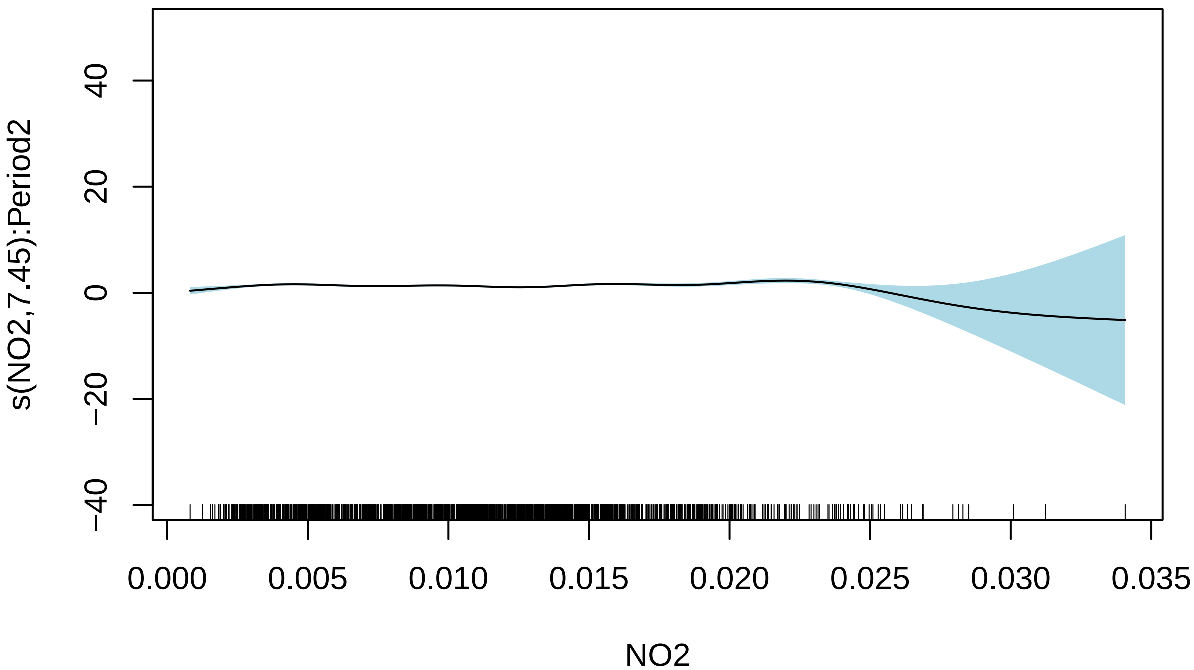




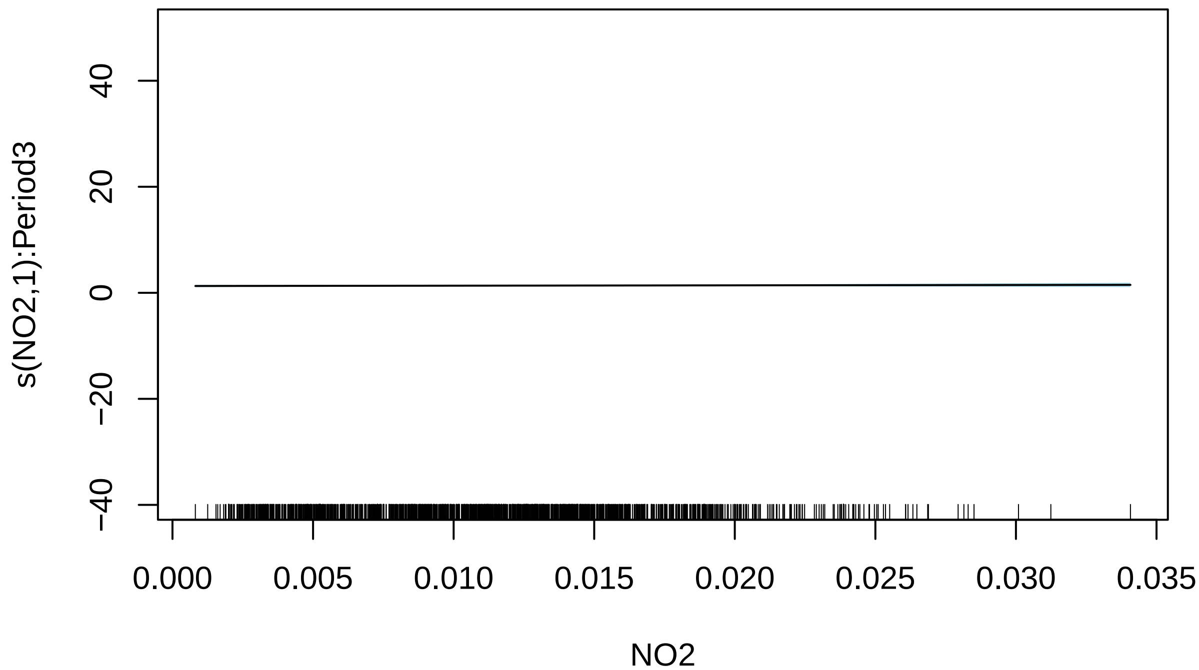
s(PM10,1):Period2
40
20
0
−20
−40
20
40
60
80
100
120
PM10

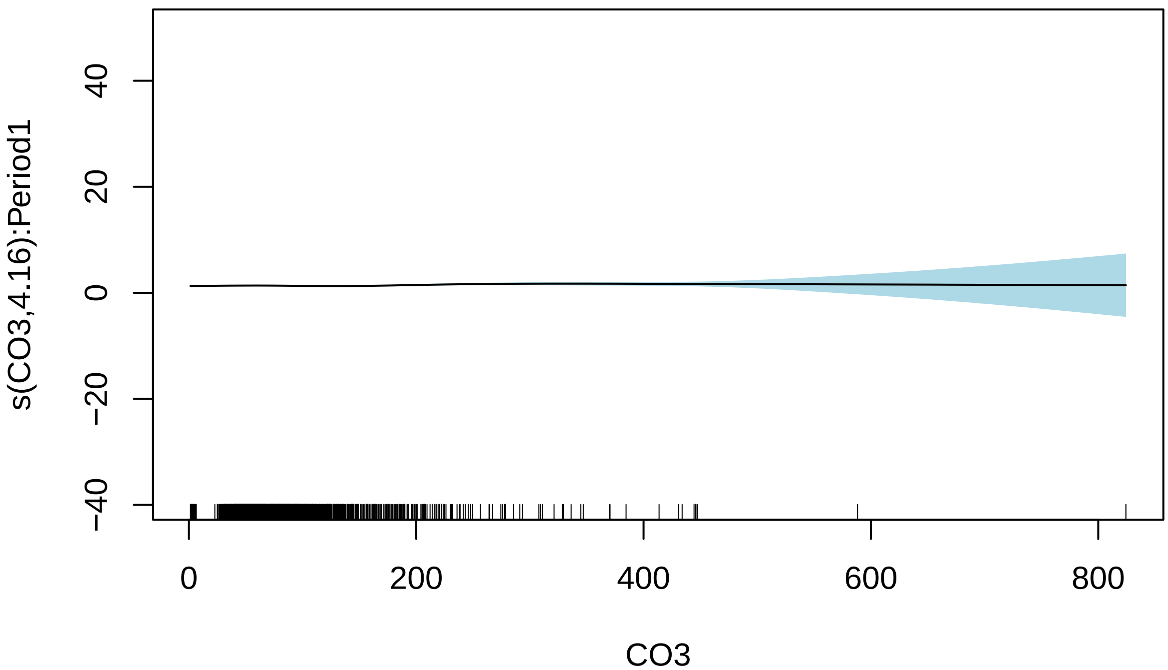

s(PM10,1):Period3
40
20
0
−20
−40
20
40
60
80
100
120
PM10


s(NO2,2.12):Period1
40
20
0
−20
−40
0.000
0.005
0.010
0.015
0.020
0.025
0.030
0.035
NO2

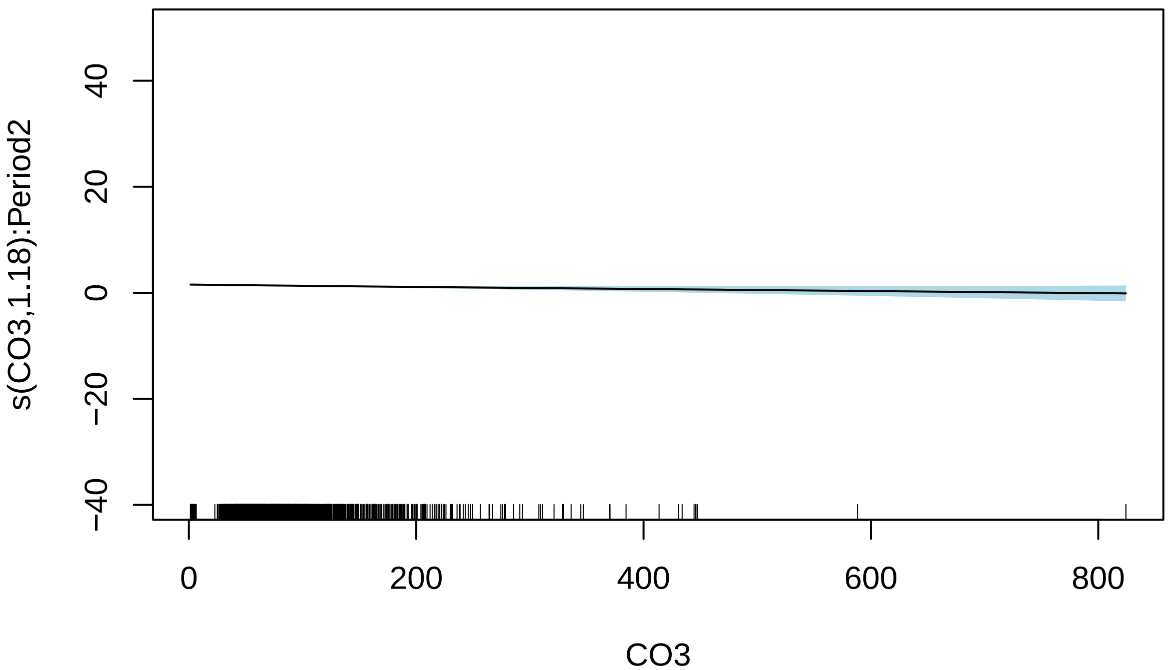


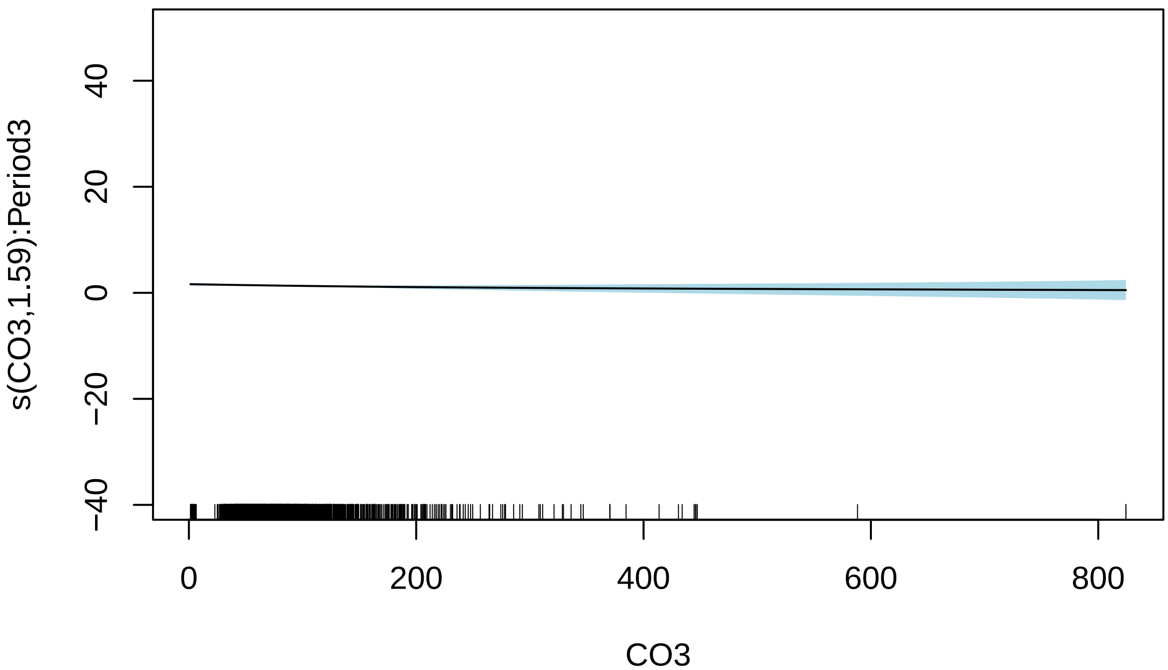
s(NO2,7.45):Period2
40
20
0
−20
−40
0.000
0.005
0.010
0.015
0.020
0.025
0.030
0.035
NO2


40
20
0
−20
−40
s(NO2,1):Period3
0.000
0.005
0.010
0.015
0.020
0.025
0.030
0.035
NO2

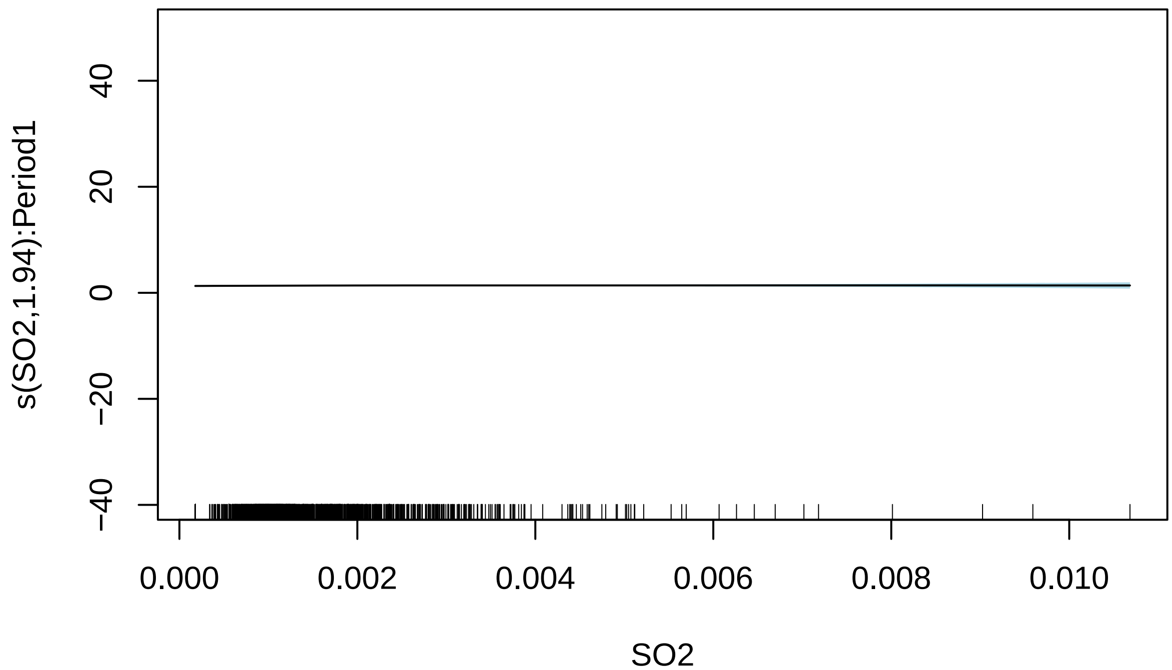


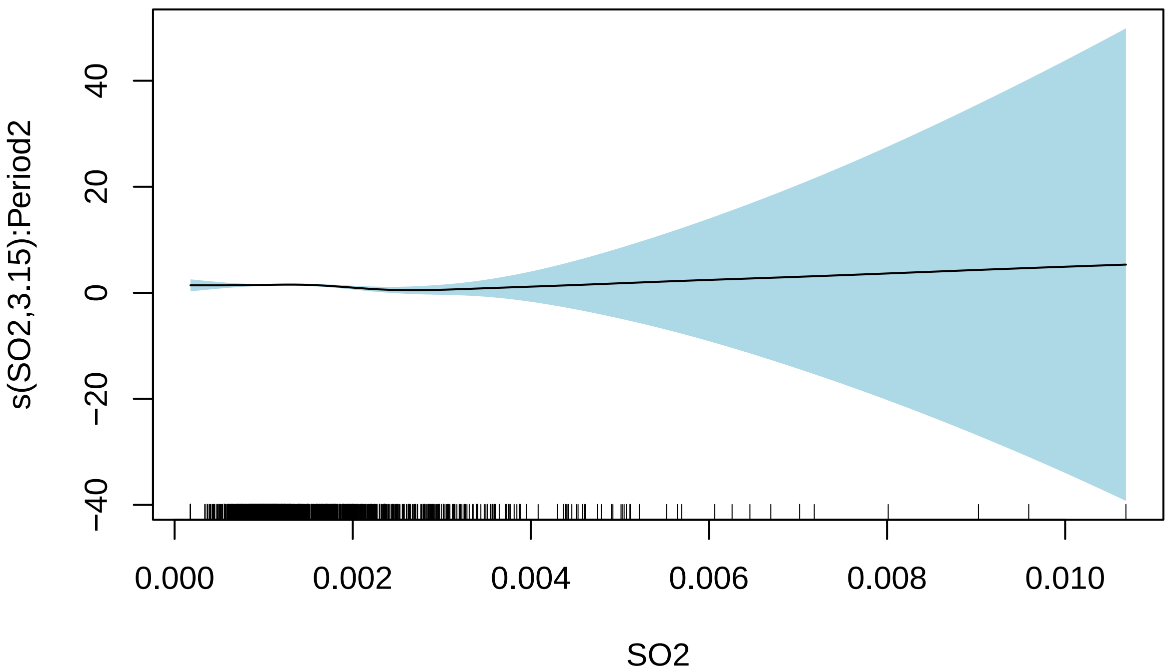
s(CO3,4.16):Period1
40
20
0
−20
−40
0
200
400
600
800
CO3


s(CO3,1.18):Period2
40
20
0
−20
−40
0
200
400
600
800
CO3

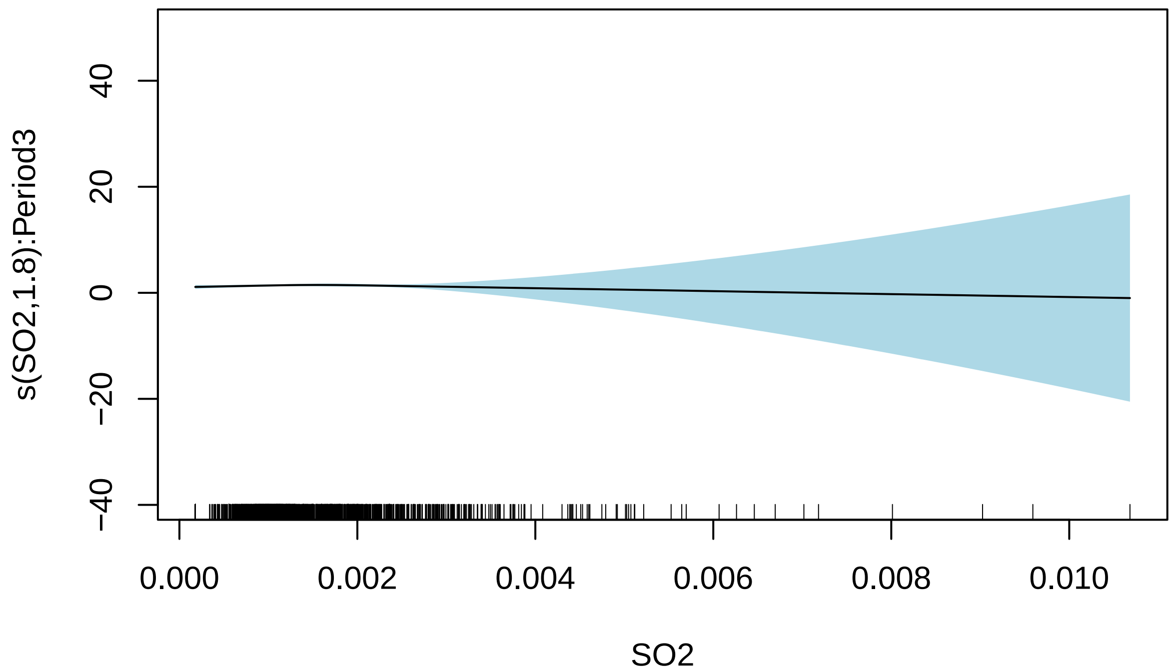


s(CO3,1.59):Period3
40
20
0
−20
−40
0
200
400
600
800
CO3

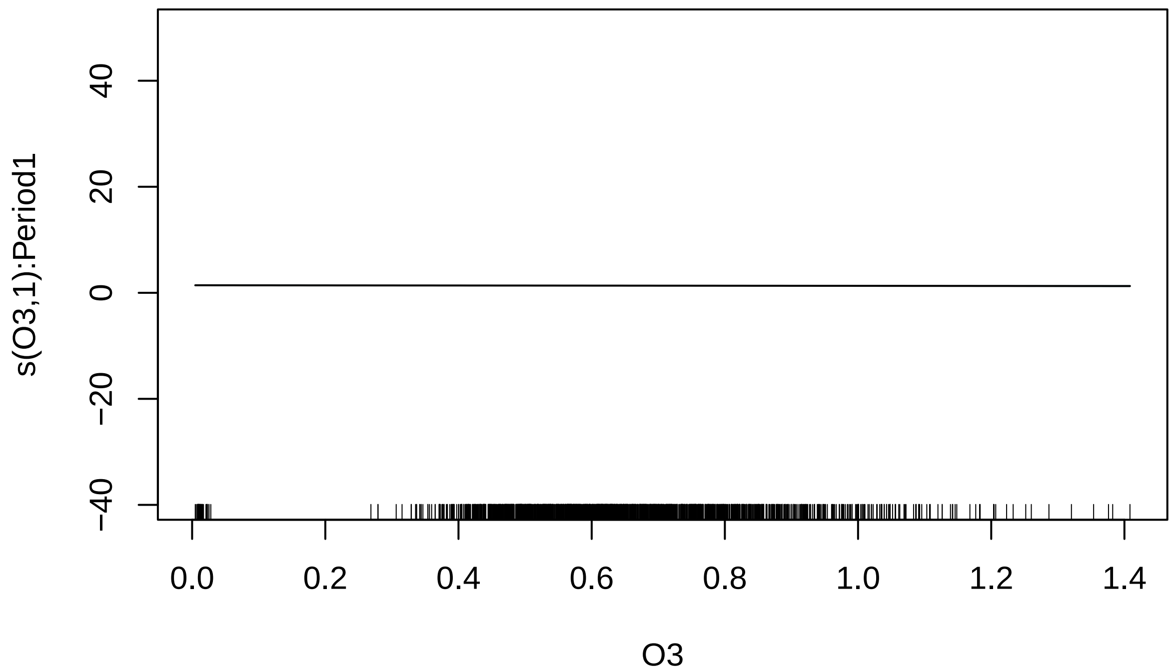

s(SO2,1.94):Period1
40
20
0
−20
−40
0.000
0.002
0.004
0.006
0.008
0.010
SO2



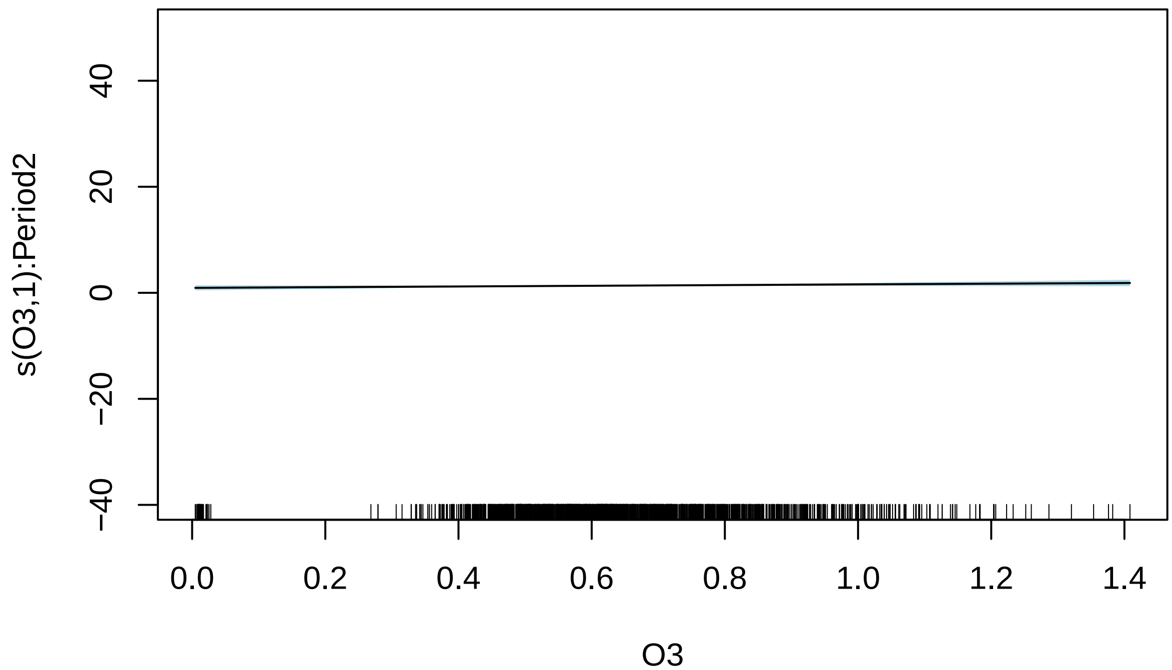
s(SO2,3.15):Period2
40
20
0
−20
−40
0.000
0.002
0.004
0.006
0.008
0.010
SO2

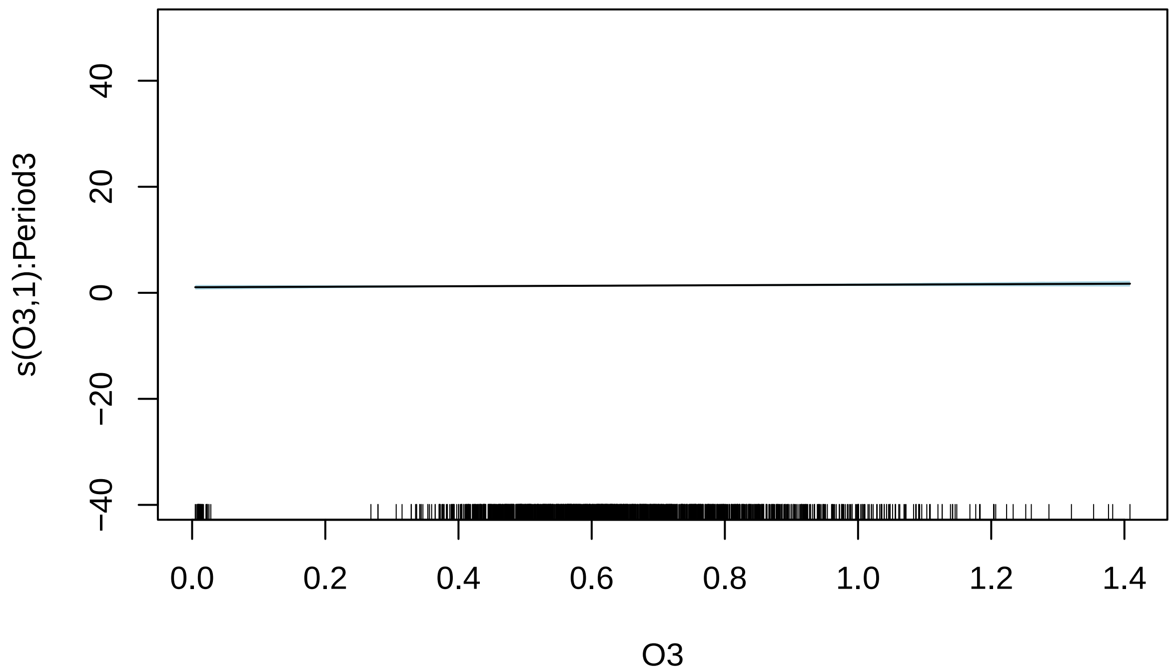


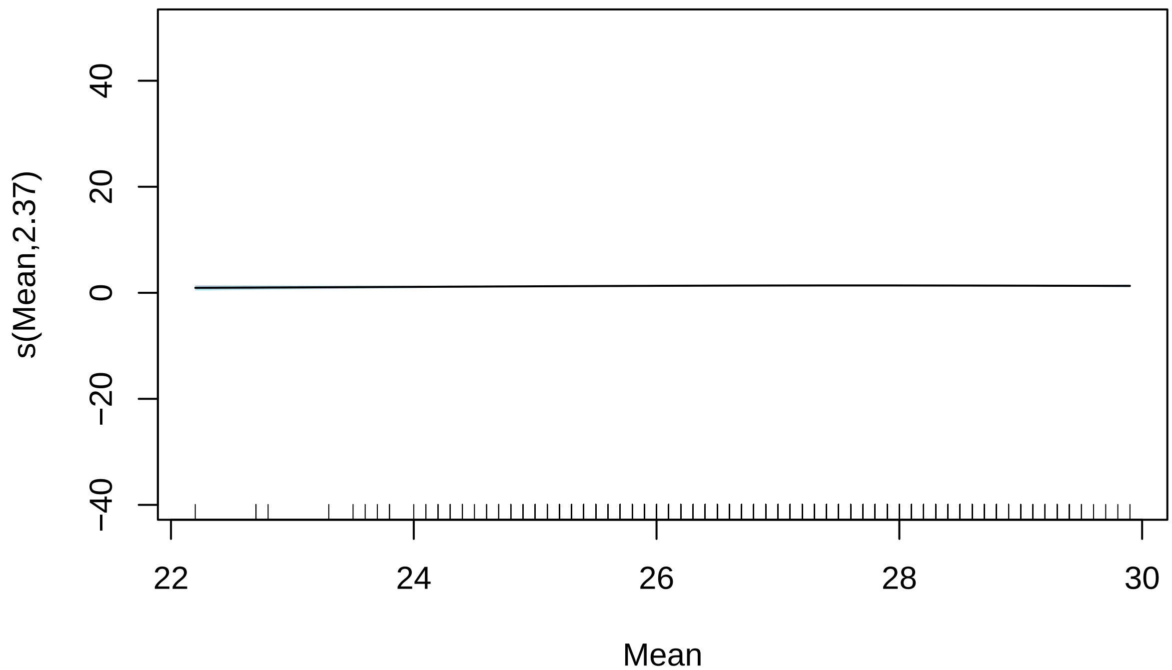
s(SO2,1.8):Period3
40
20
0
−20
−40
0.000
0.002
0.004
0.006
0.008
0.010
SO2

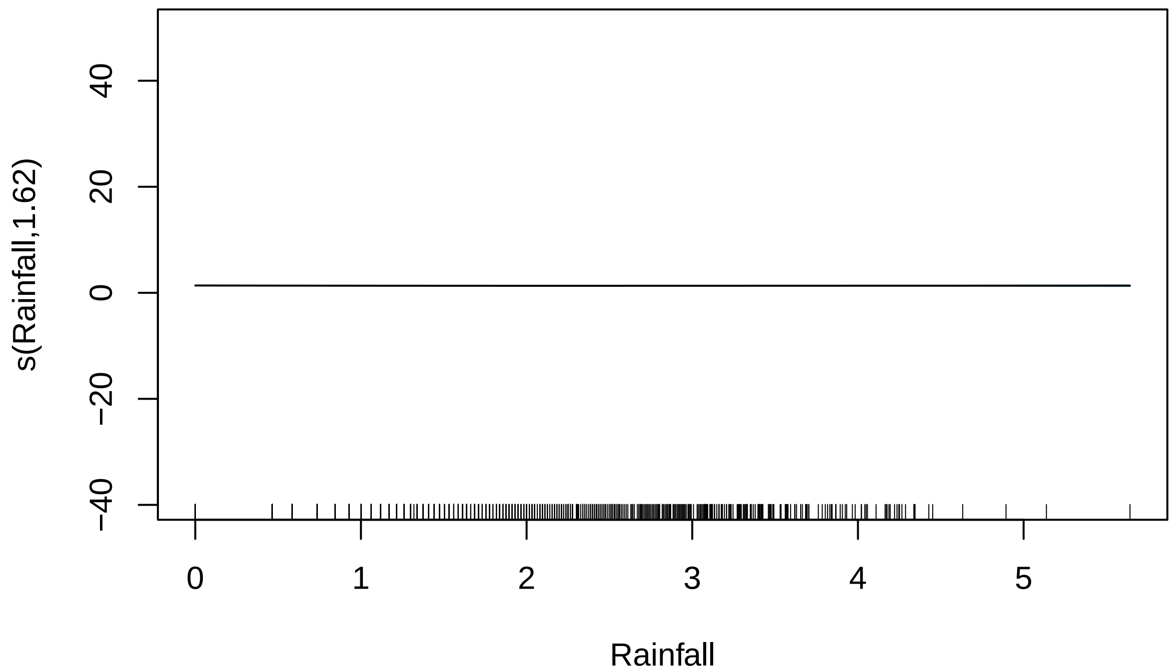


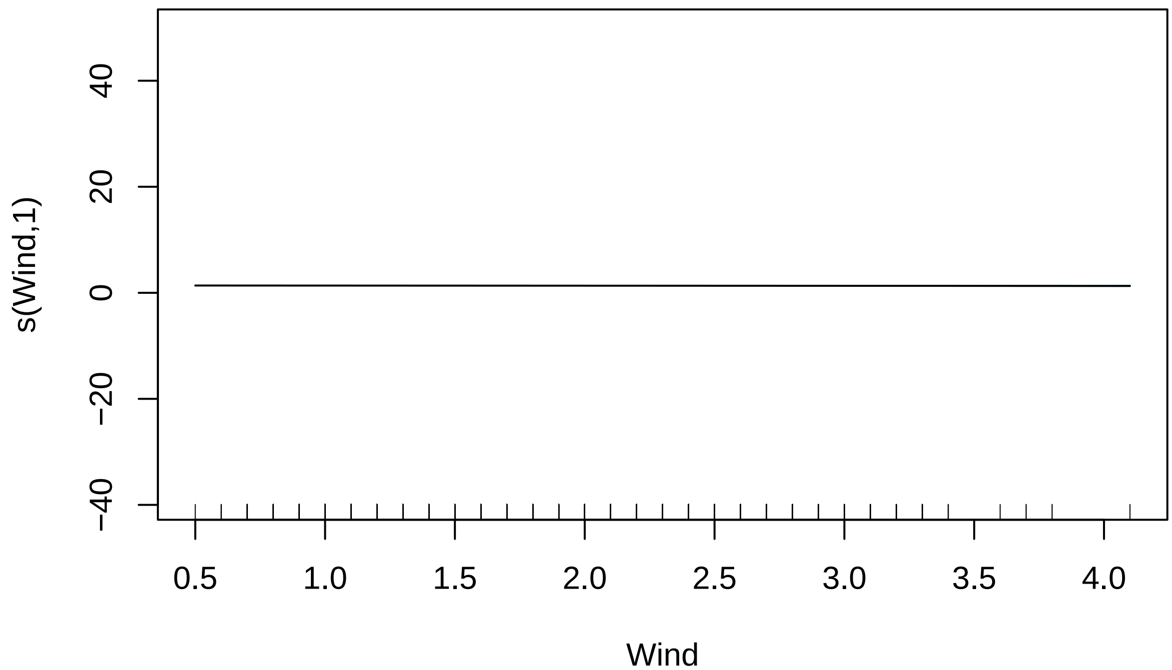
s(O3,1):Period1
40
20
0
−20
−40
0.0
0.2
0.4
0.6
0.8
1.0
1.2
1.4
O3

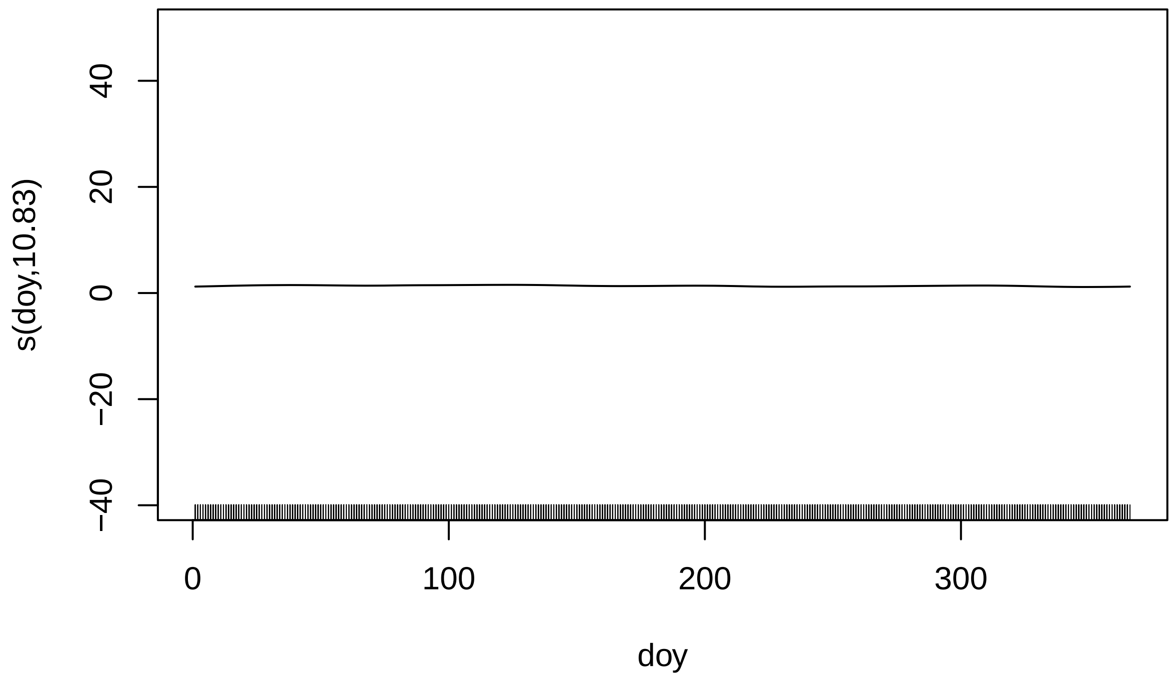

s(O3,1):Period2
40
20
0
−20
−40
0.0
0.2
0.4
0.6
0.8
1.0
1.2
1.4
O3


s(O3,1):Period3
40
20
0
−20
−40
0.0
0.2
0.4
0.6
0.8
1.0
1.2
1.4
O3

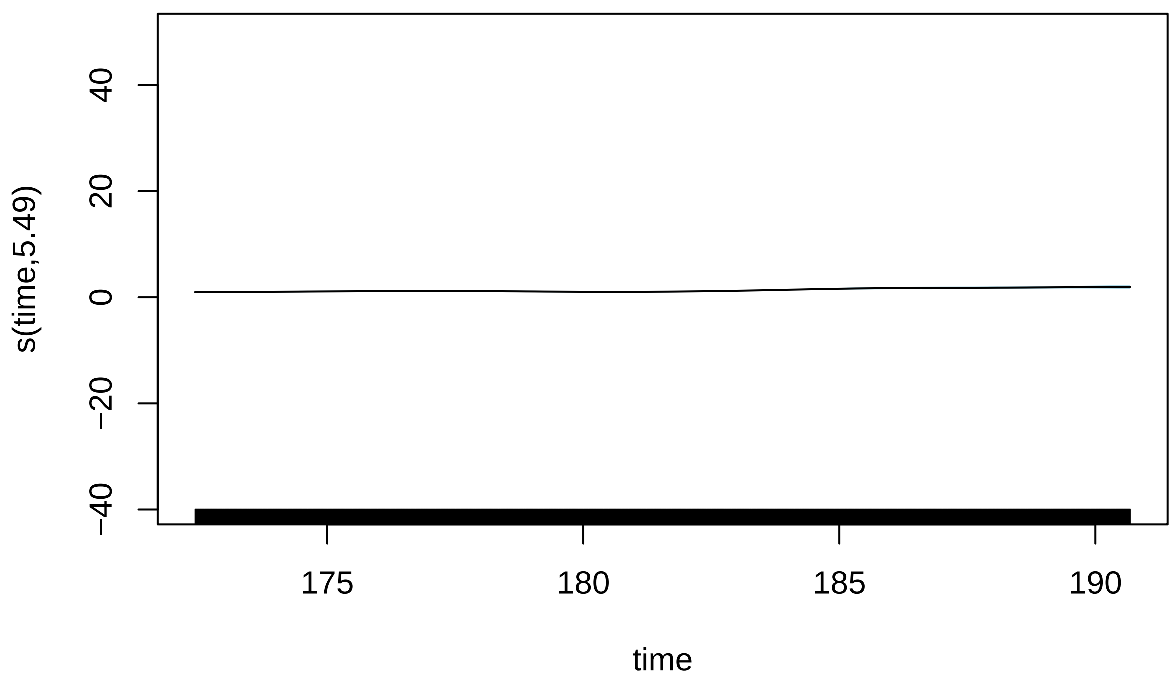

s(Mean,2.37)
40
20
0
−20
−40
22
24
26
28
30
Mean


40
20
0
−20
−40
s(Rainfall,1.62)
0
1
2
3
4
5
Rainfall

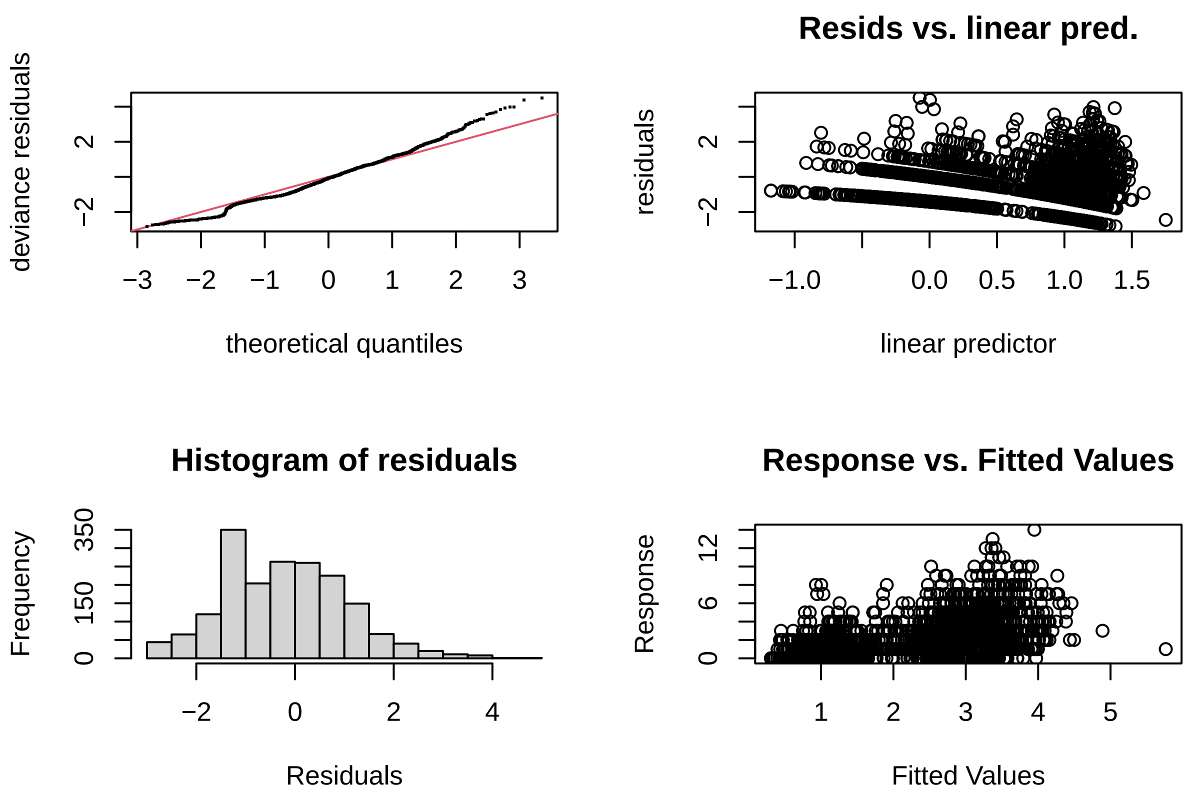

40
20
0
−20
−40
s(Wind,1)
0.5
1.0
1.5
2.0
2.5
3.0
3.5
4.0
Wind



s(doy,10.83)
40
20
0
−20
−40
0
100
200
300
doy




```
gam.check(jb_URTI_period)
```




```
##
## Method: REML   Optimizer: outer newton
## full convergence after 15 iterations.
## Gradient range [-0.0003096198,0.0001142785]
## (score 3539.058 & scale 1).
## Hessian positive definite, eigenvalue range [4.169477e-05,1.466765].
## Model rank =  537 / 537
##
## Basis dimension (k) checking results. Low p-value (k-index<1) may
## indicate that k is too low, especially if edf is close to k'.
##
##                     k'   edf k-index p-value
## s(PM10):Period1   9.00  2.22    0.97   0.330
## s(PM10):Period2   9.00  1.00    0.97   0.355
## s(PM10):Period3   9.00  1.00    0.97   0.375
## s(NO2):Period1    9.00  2.12    0.99   0.830
## s(NO2):Period2    9.00  7.45    0.99   0.805
## s(NO2):Period3    9.00  1.00    0.99   0.810
## s(CO3):Period1    9.00  4.16    0.94   0.050 *
## s(CO3):Period2    9.00  1.18    0.94   0.035 *
## s(CO3):Period3    9.00  1.59    0.94   0.055 .
## s(SO2):Period1    9.00  1.94    0.96   0.195
## s(SO2):Period2    9.00  3.15    0.96   0.215
## s(SO2):Period3    9.00  1.80    0.96   0.200
## s(O3):Period1     9.00  1.00    0.98   0.580
## s(O3):Period2     9.00  1.00    0.98   0.520
```

```
## s(O3):Period3     9.00   1.00    0.98   0.585
## s(Mean)           9.00   2.37    0.96   0.240
## s(Rainfall)       9.00   1.62    0.94   0.070 .
## s(Wind)           9.00   1.00    0.90  <2e-16 ***
## s(doy)          363.00  10.83    0.93   0.015 *
## s(time)           9.00   5.49    0.86  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

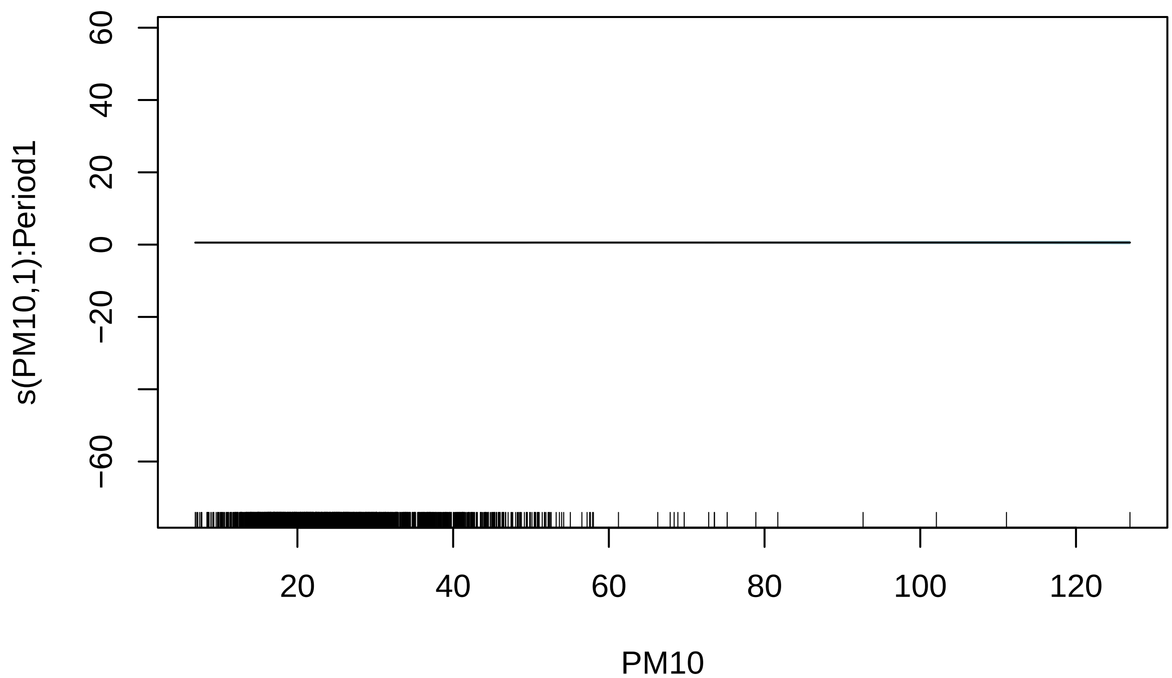
```
jb_pneumo_period <- gam(Pneumonia ~ s(PM10, by=Period) + s(NO2, by=Period) + s(CO3, by=Period) + s(SO2,
summary(jb_pneumo_period)
```

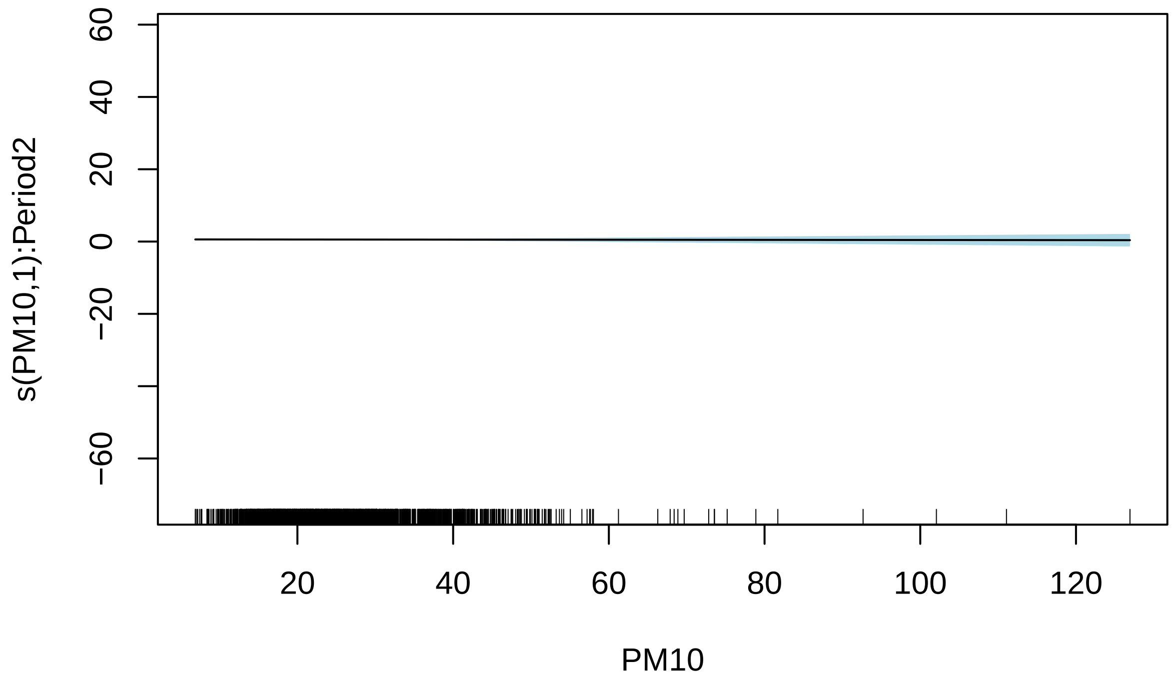
**KB influenza or pneumonia by period**

```
##
## Family: poisson
## Link function: log
##
## Formula:
## Pneumonia ~ s(PM10, by = Period) + s(NO2, by = Period) + s(CO3,
##     by = Period) + s(SO2, by = Period) + s(O3, by = Period) +
##     s(Mean) + s(Rainfall) + s(Wind) + s(doy, bs = "cc", k = 365) +
##     s(time, bs = "bs") + Period
##
## Parametric coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  0.55899    0.05411  10.332  < 2e-16 ***
## Period2     -1.13810    0.23629  -4.817 1.46e-06 ***
## Period3     -0.35880    0.29677  -1.209    0.227
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Approximate significance of smooth terms:
##                    edf  Ref.df Chi.sq  p-value
## s(PM10):Period1 1.001   1.001  0.004 0.954707
## s(PM10):Period2 1.000   1.001  0.042 0.839138
## s(PM10):Period3 1.001   1.002  0.651 0.420489
## s(NO2):Period1  1.558   1.938  1.718 0.366025
## s(NO2):Period2  4.546   5.514 22.774 0.000615 ***
## s(NO2):Period3  1.001   1.001  3.698 0.054513 .
## s(CO3):Period1  1.000   1.001  4.497 0.033979 *
## s(CO3):Period2  1.000   1.001  1.771 0.183328
## s(CO3):Period3  1.000   1.000  6.988 0.008211 **
## s(SO2):Period1  3.765   4.717  6.233 0.210452
## s(SO2):Period2  3.443   4.067 17.922 0.001510 **
## s(SO2):Period3  1.429   1.682  1.228 0.331621
## s(O3):Period1   4.256   5.357  7.470 0.260727
## s(O3):Period2   1.990   2.521  6.397 0.056909 .
## s(O3):Period3   1.000   1.000  0.379 0.538368
## s(Mean)         1.000   1.001  1.282 0.257591
## s(Rainfall)     1.440   1.758  1.289 0.391615
```

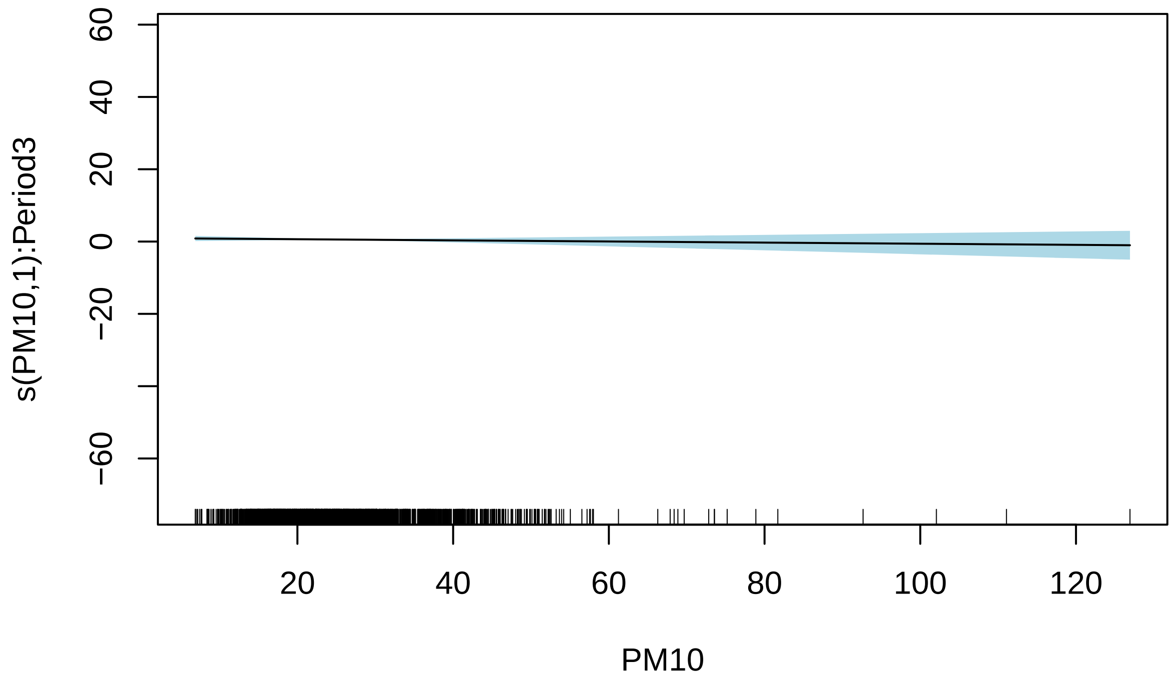
```
## s(Wind)          2.158   2.760  3.043 0.443944
## s(doy)           3.011 363.000 14.073 0.000774 ***
## s(time)          2.540   3.161 13.374 0.004548 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-sq.(adj) =   0.11   Deviance explained = 12.5%
## -REML = 2758.3  Scale est. = 1         n = 1827
```

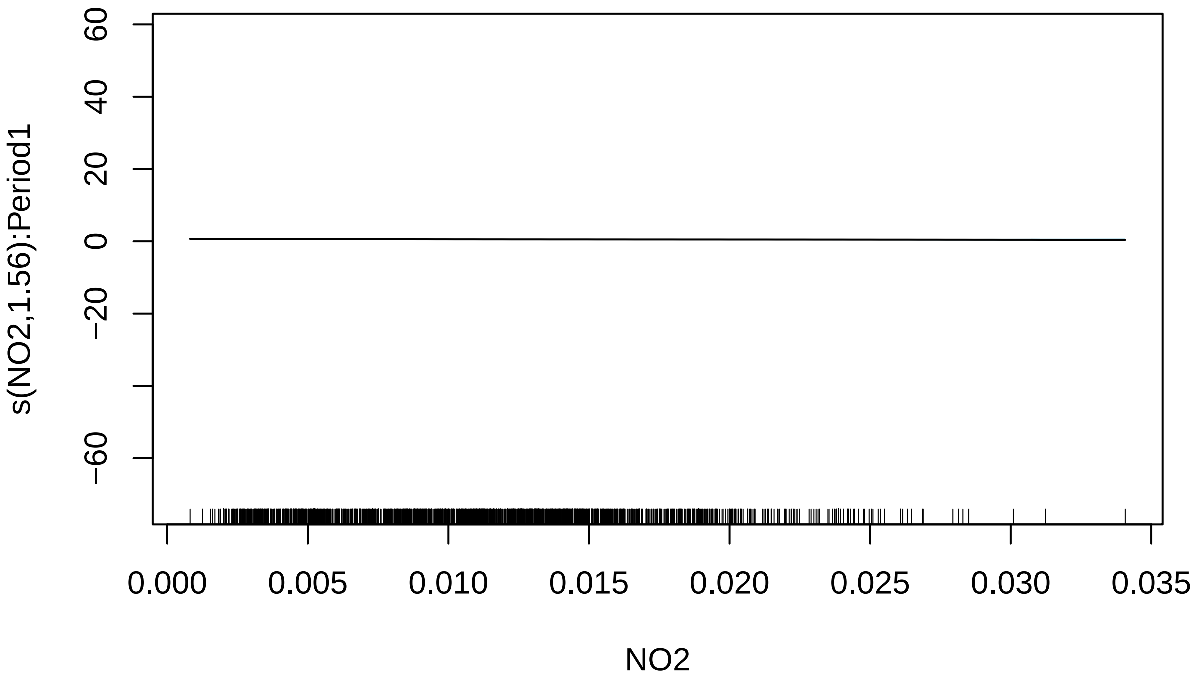
```
plot.gam(jb_pneumo_period, shade = TRUE, shade.col = "lightblue", shift = coef(jb_pneumo_period)[1],  se
```

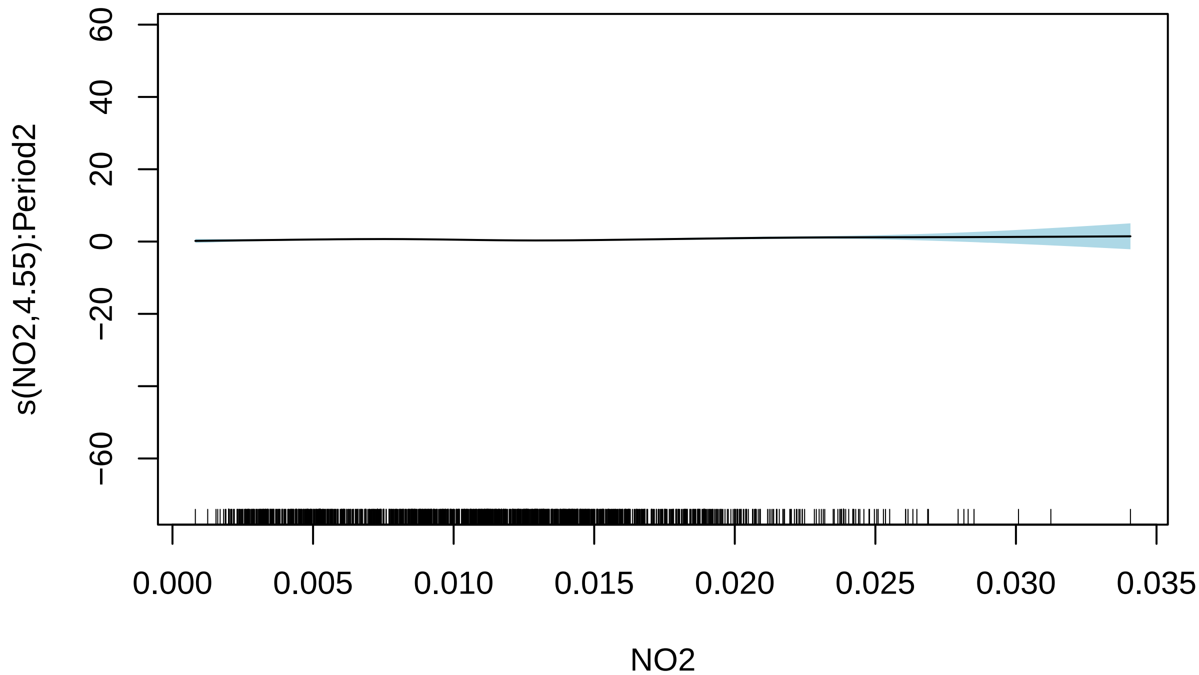


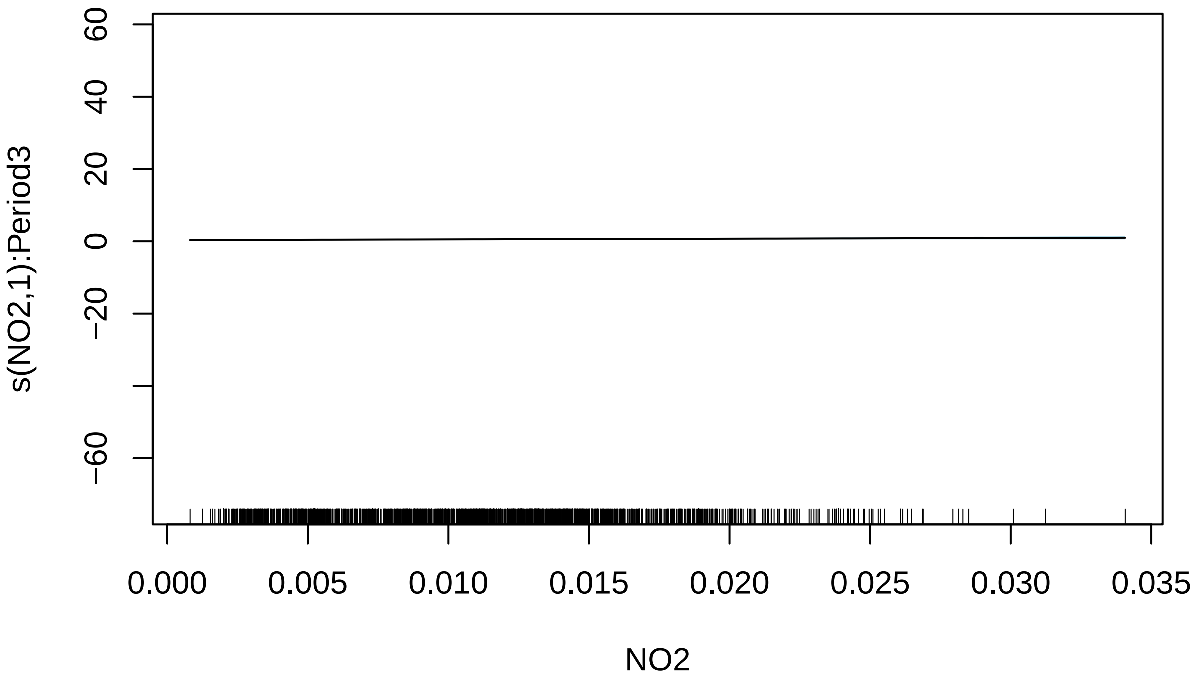


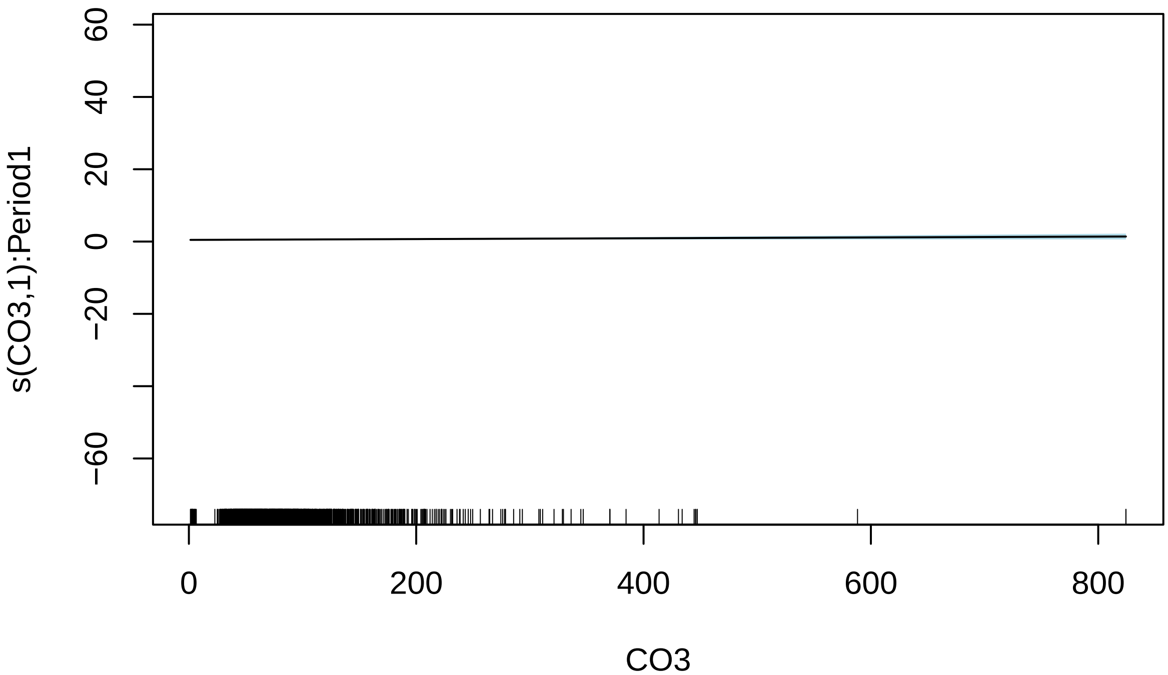
s(PM10,1):Period2
60
40
20
0
−20
−60
20
40
60
80
100
120
PM10


s(PM10,1):Period3
60
40
20
0
−20
−60
20
40
60
80
100
120
PM10

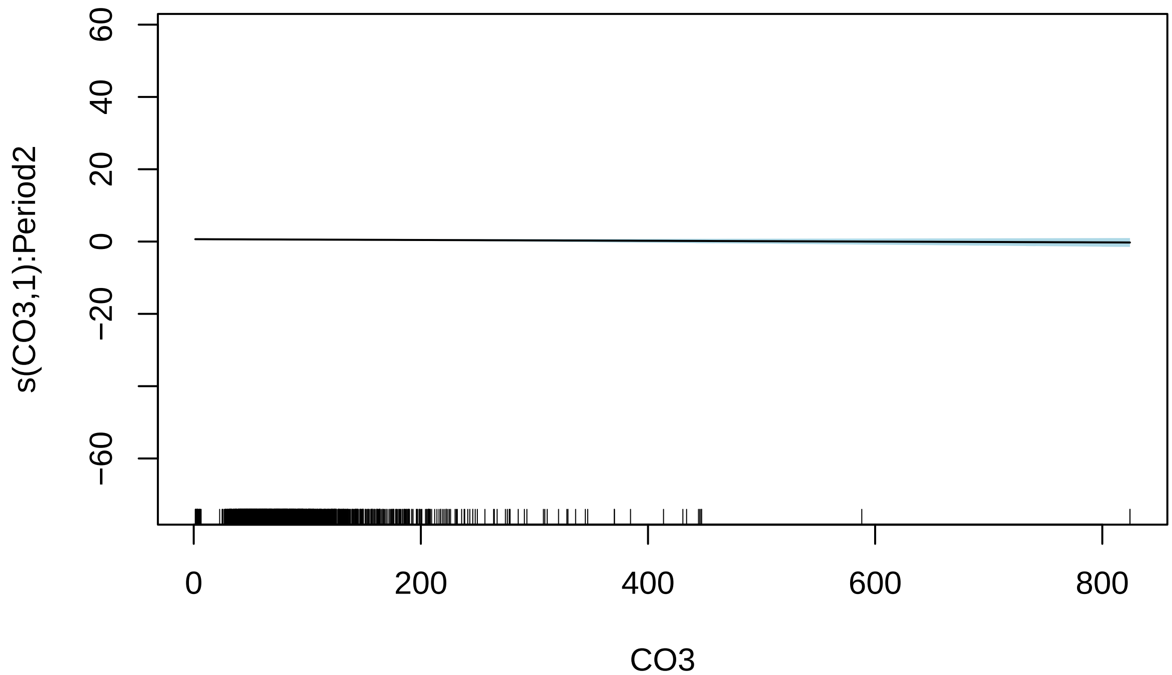

s(NO2,1.56):Period1
60
40
20
0
−20
−60
0.000
0.005
0.010
0.015
0.020
0.025
0.030
0.035
NO2

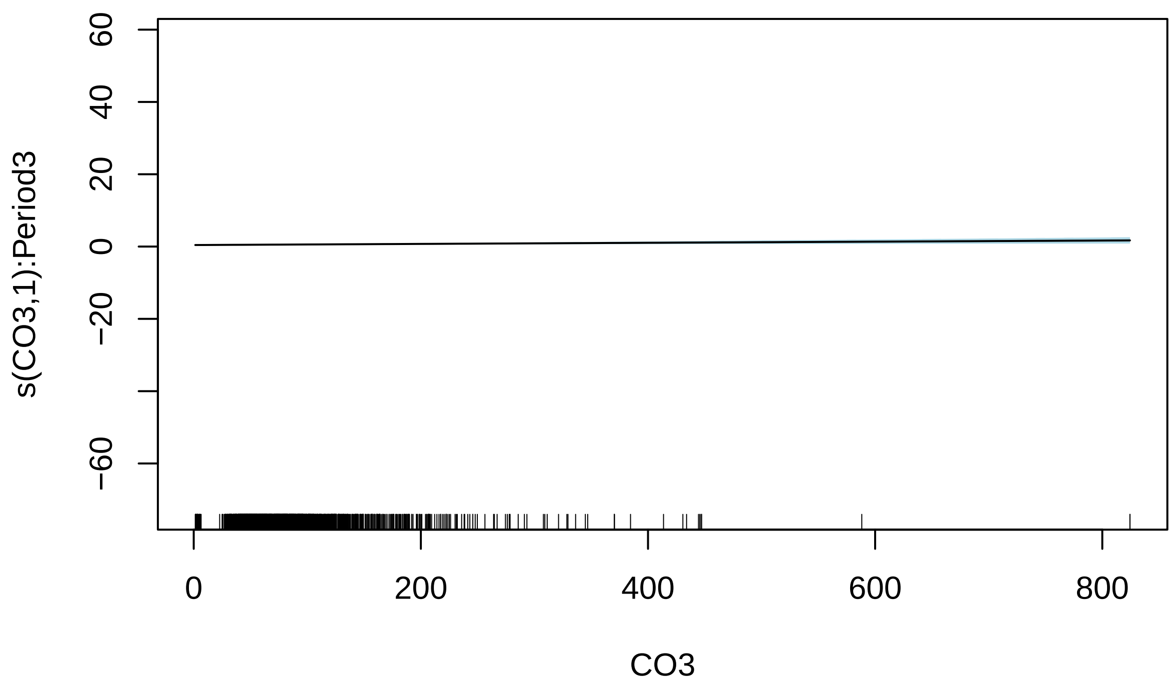

s(NO2,4.55):Period2
60
40
20
0
−20
−60
0.000
0.005
0.010
0.015
0.020
0.025
0.030
0.035
NO2

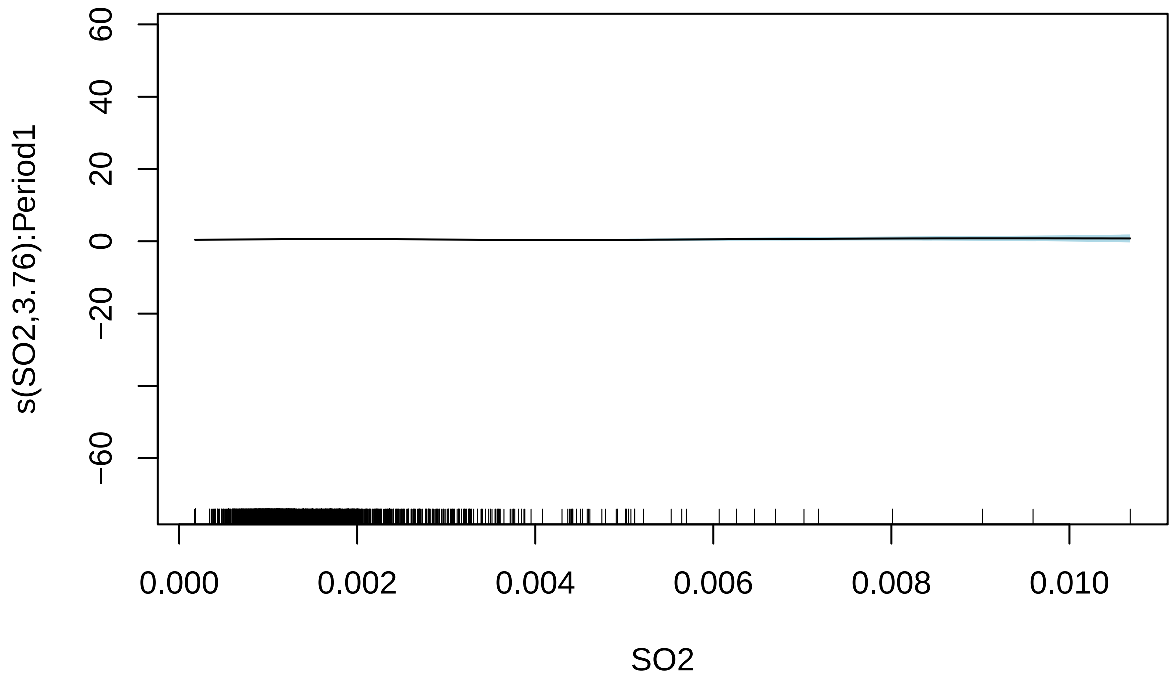

s(NO2,1):Period3
60
40
20
0
−20
−60
0.000
0.005
0.010
0.015
0.020
0.025
0.030
0.035
NO2

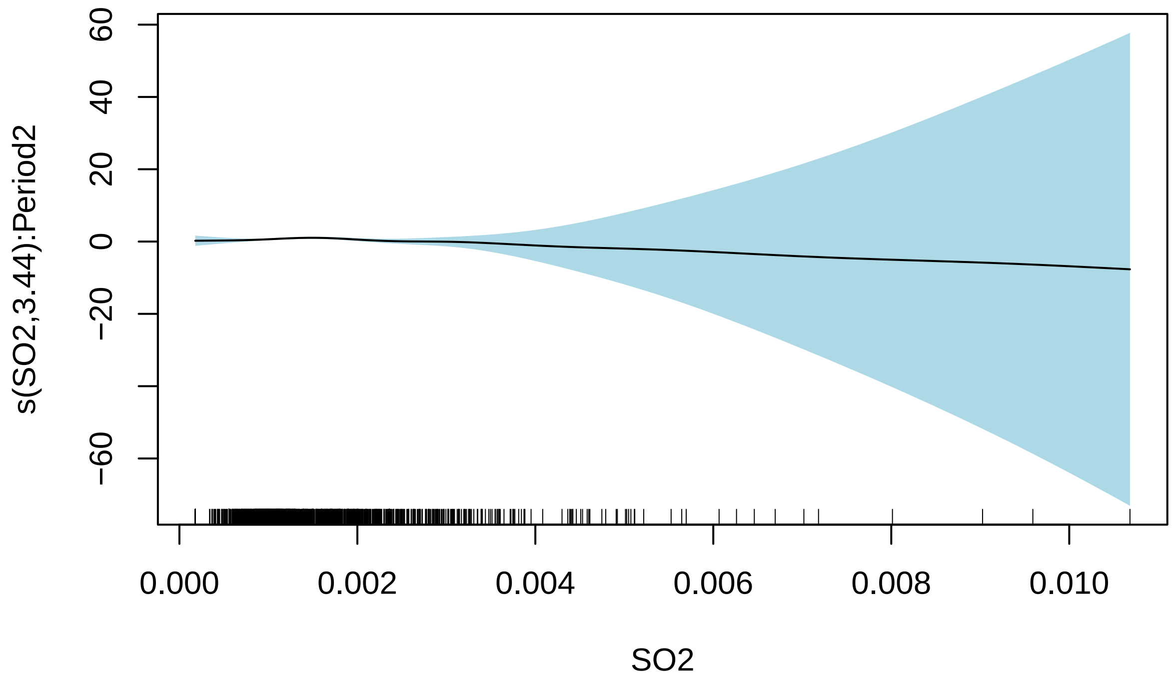


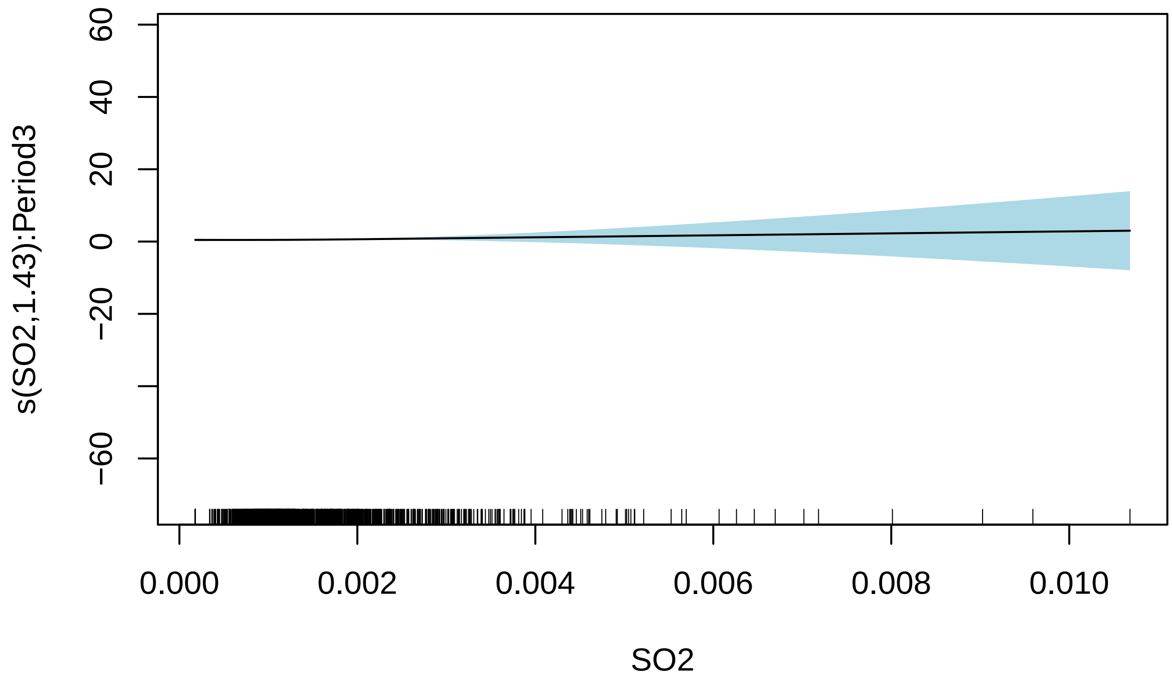
s(CO3,1):Period1
60
40
20
0
−20
−60
0
200
400
600
800
CO3

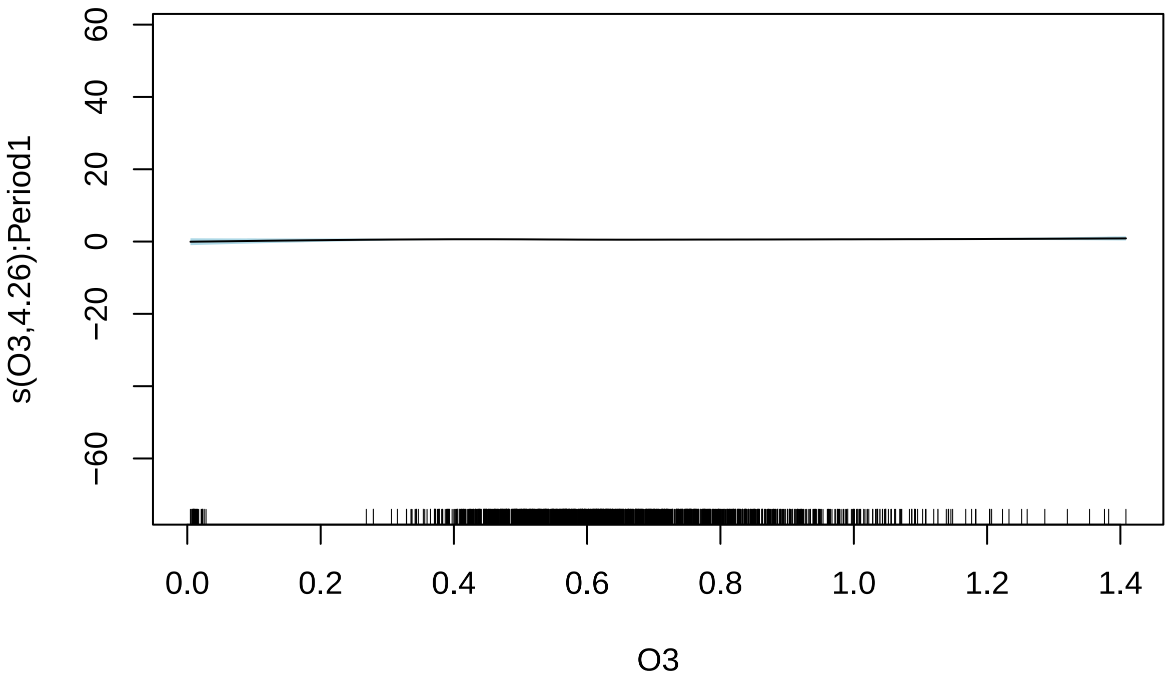

s(CO3,1):Period2
60
40
20
0
−20
−60
0
200
400
600
800
CO3


s(CO3,1):Period3
60
40
20
0
−20
−60
0
200
400
600
800
CO3

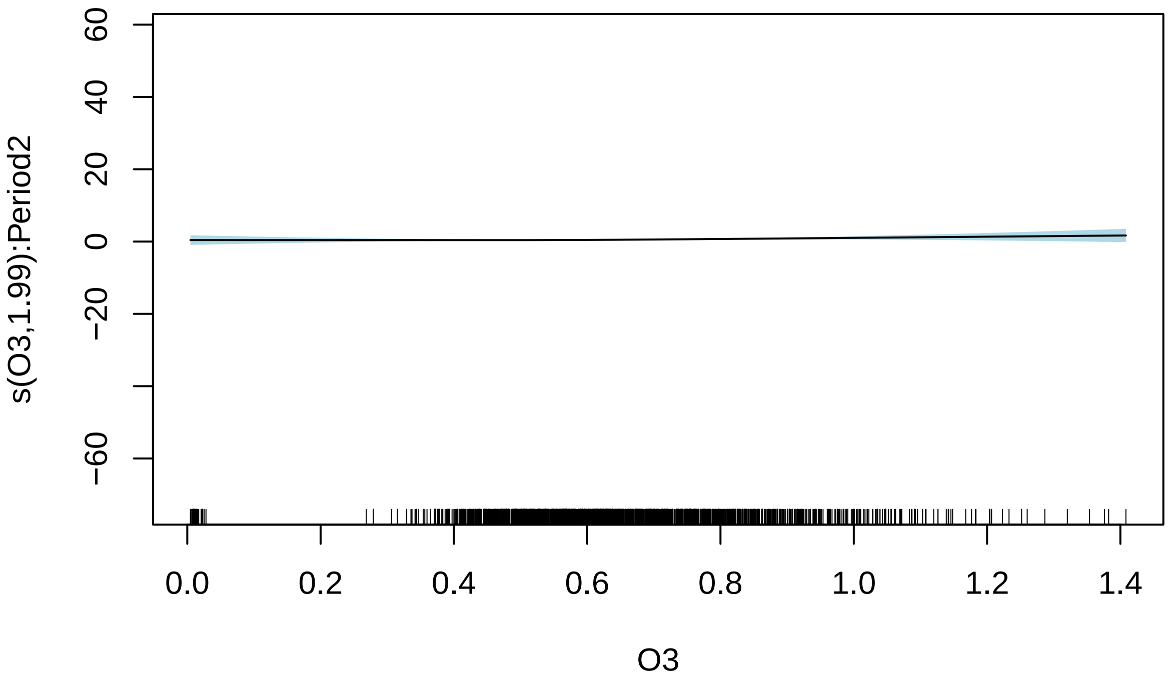

s(SO2,3.76):Period1
60
40
20
0
−20
−60
0.000
0.002
0.004
0.006
0.008
0.010
SO2


s(SO2,3.44):Period2
60
40
20
0
−20
−60
0.000
0.002
0.004
0.006
0.008
0.010
SO2

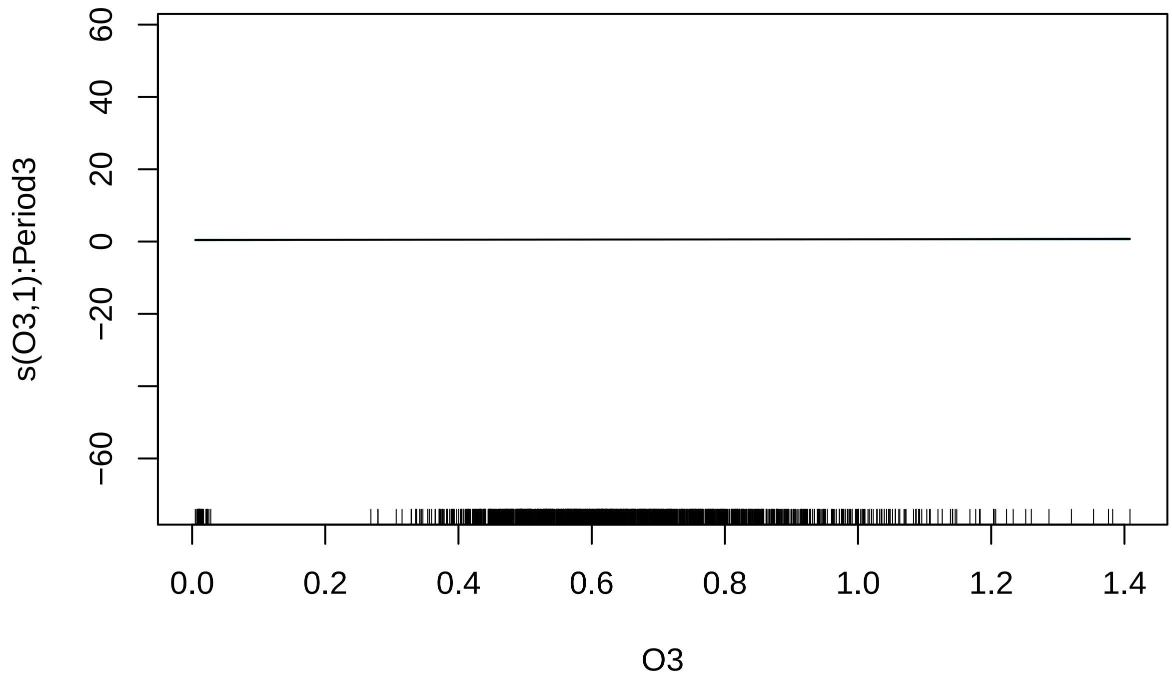

s(SO2,1.43):Period3
60
40
20
0
−20
−60
0.000
0.002
0.004
0.006
0.008
0.010
SO2

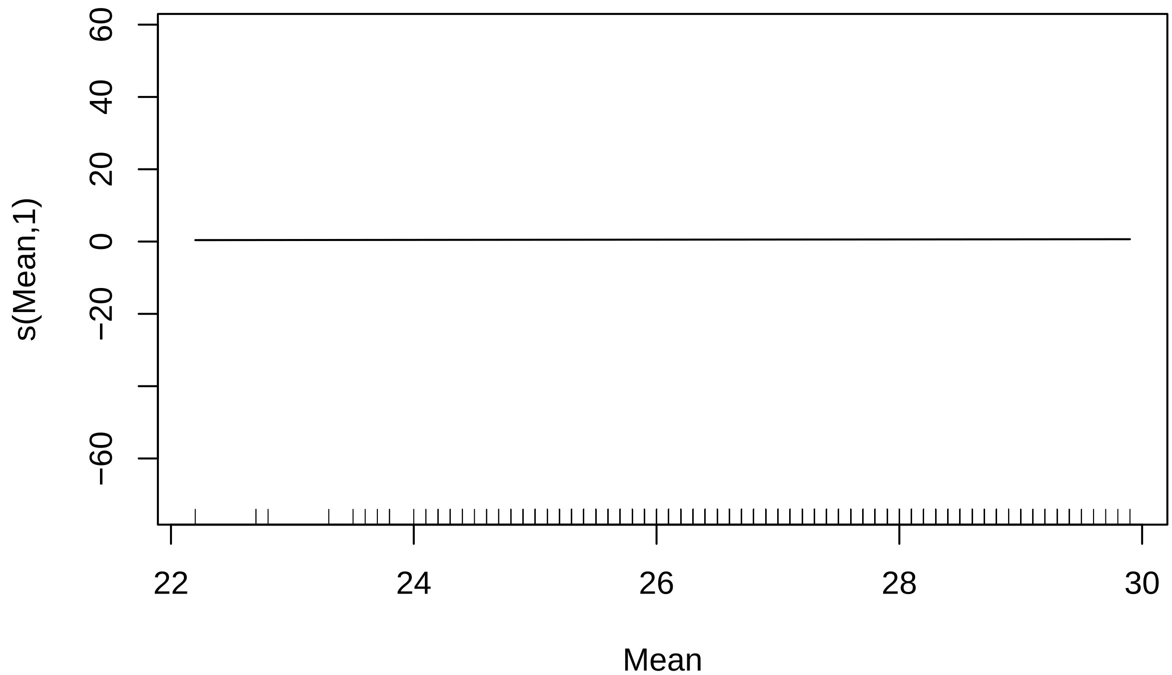


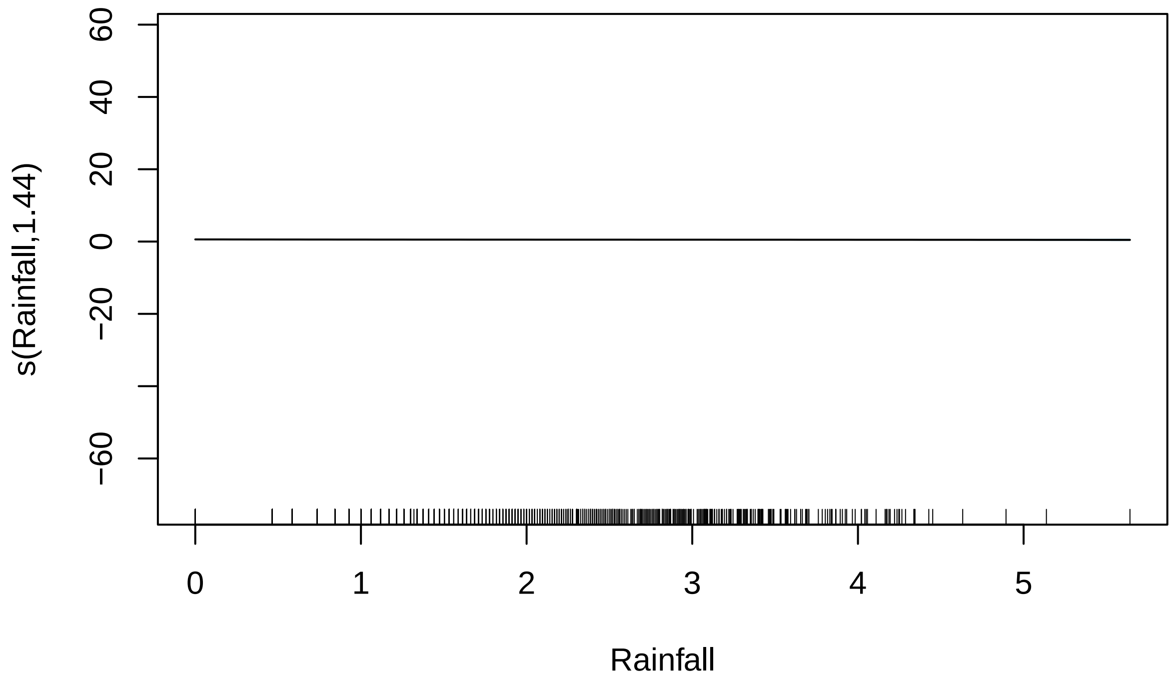
s(O3,4.26):Period1
60
40
20
0
−20
−60
0.0
0.2
0.4
0.6
0.8
1.0
1.2
1.4
O3

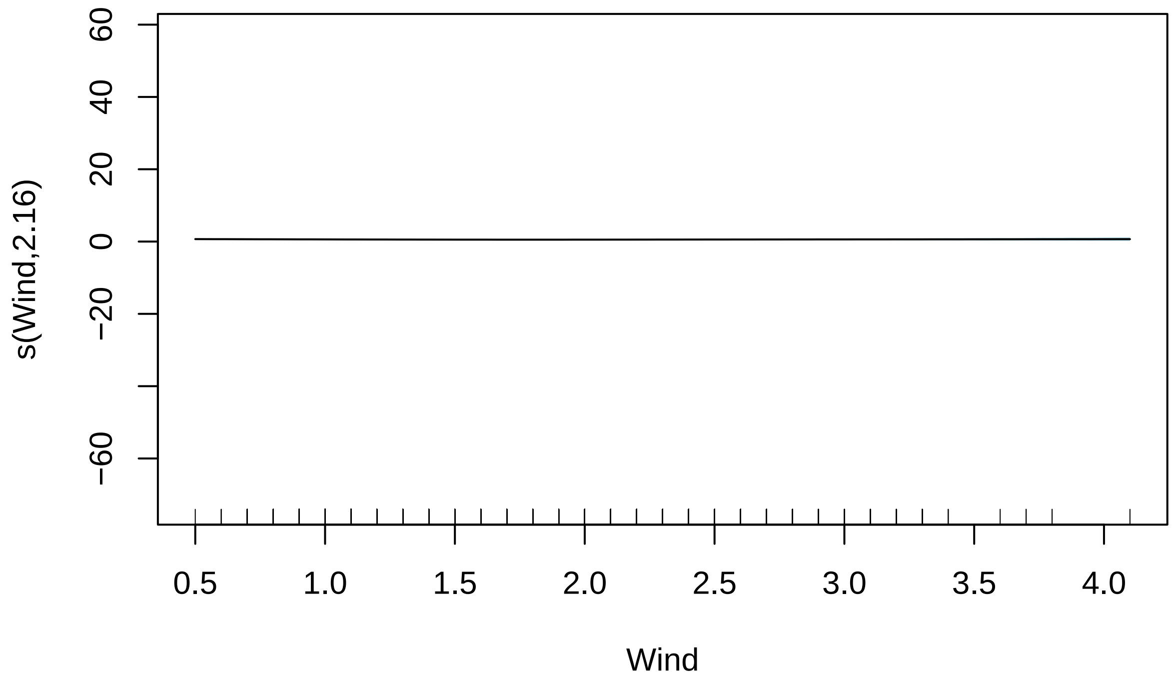


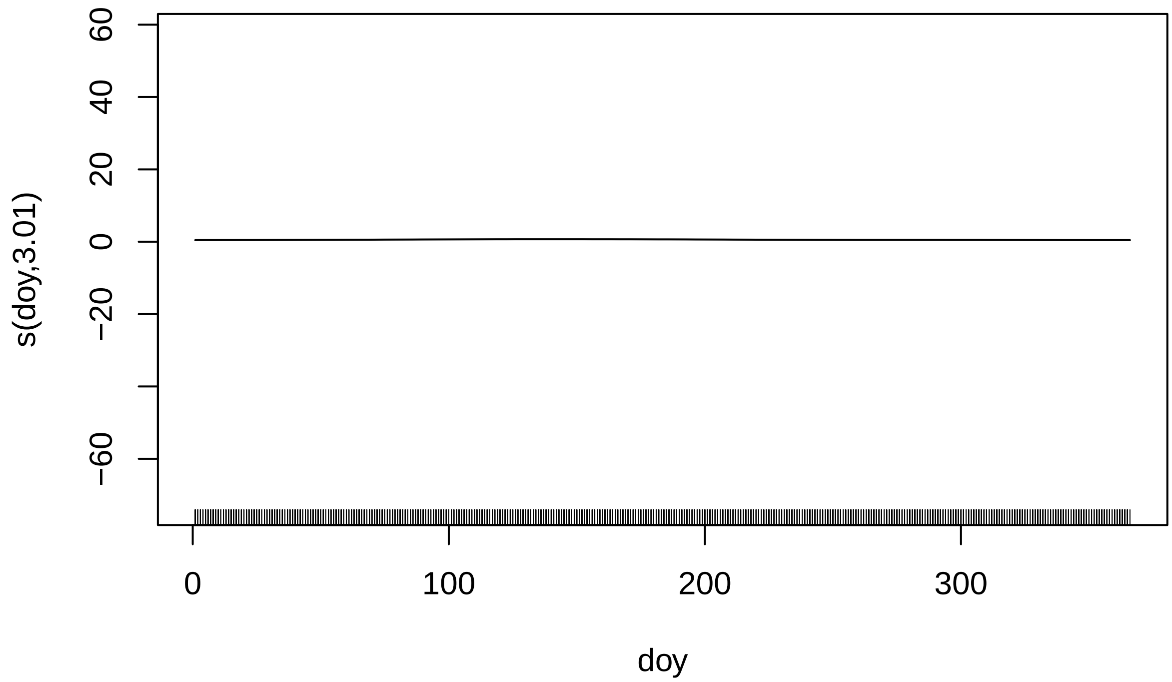
s(O3,1.99):Period2
60
40
20
0
−20
−60
0.0
0.2
0.4
0.6
0.8
1.0
1.2
1.4
O3


s(O3,1):Period3
60
40
20
0
−20
−60
0.0
0.2
0.4
0.6
0.8
1.0
1.2
1.4
O3

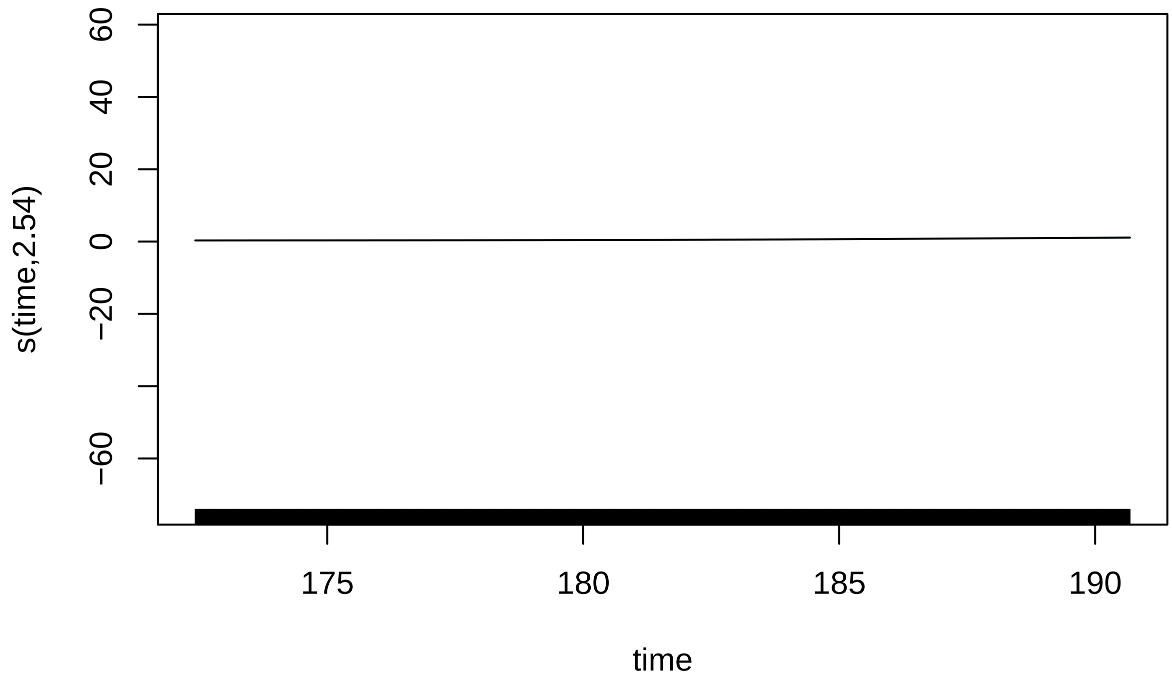

s(Mean,1)
60
40
20
0
−20
−60
22
24
26
28
30
Mean


s(Rainfall,1.44)
60
40
20
0
−20
−60
0
1
2
3
4
5
Rainfall

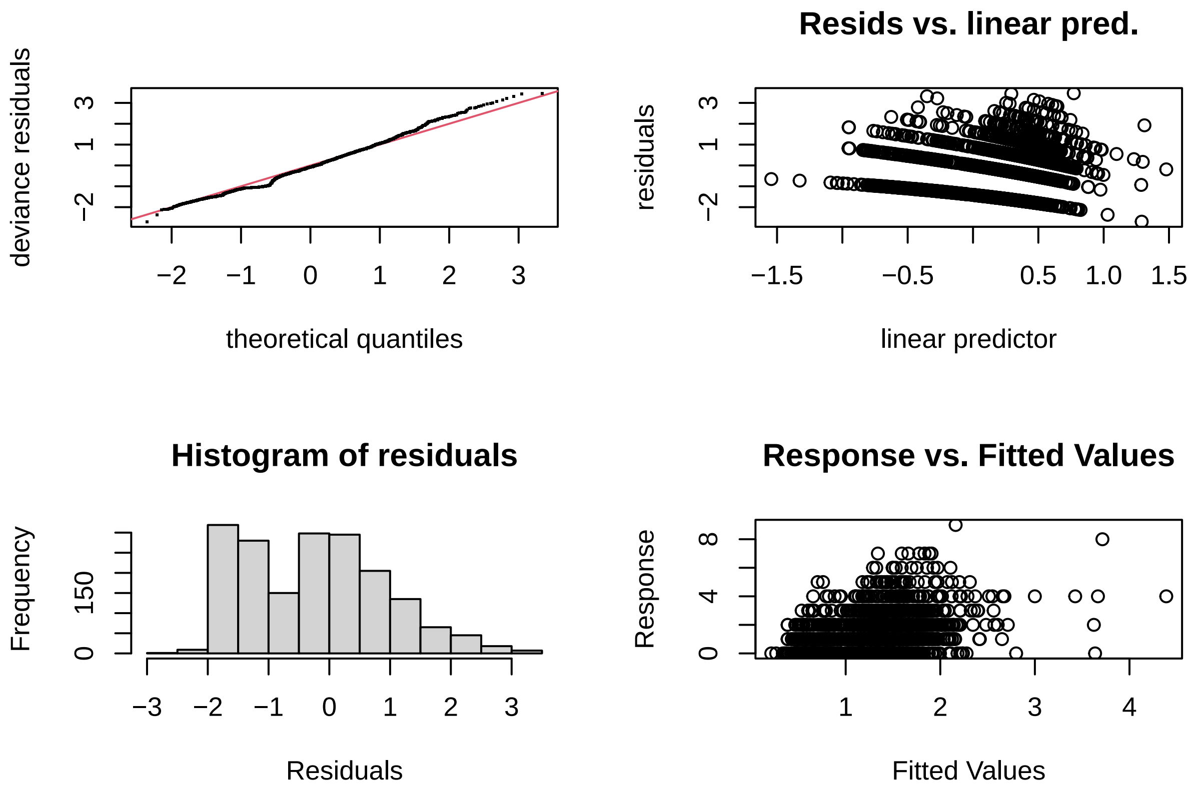

s(Wind,2.16)
60
40
20
0
−20
−60
0.5
1.0
1.5
2.0
2.5
3.0
3.5
4.0
Wind



s(doy,3.01)
60
40
20
0
−20
−60
0
100
200
300
doy




```
gam.check(jb_pneumo_period)
```




```
##
## Method: REML   Optimizer: outer newton
## full convergence after 10 iterations.
## Gradient range [-0.0002210864,9.350133e-06]
## (score 2758.252 & scale 1).
## Hessian positive definite, eigenvalue range [5.104145e-05,1.094411].
## Model rank =  537 / 537
##
## Basis dimension (k) checking results. Low p-value (k-index<1) may
## indicate that k is too low, especially if edf is close to k'.
##
##                      k'   edf k-index p-value
## s(PM10):Period1    9.00  1.00    0.95   0.375
## s(PM10):Period2    9.00  1.00    0.95   0.295
## s(PM10):Period3    9.00  1.00    0.95   0.340
## s(NO2):Period1     9.00  1.56    0.96   0.505
## s(NO2):Period2     9.00  4.55    0.96   0.500
## s(NO2):Period3     9.00  1.00    0.96   0.425
## s(CO3):Period1     9.00  1.00    0.94   0.150
## s(CO3):Period2     9.00  1.00    0.94   0.160
## s(CO3):Period3     9.00  1.00    0.94   0.140
## s(SO2):Period1     9.00  3.76    0.98   0.745
## s(SO2):Period2     9.00  3.44    0.98   0.740
## s(SO2):Period3     9.00  1.43    0.98   0.730
## s(O3):Period1      9.00  4.26    0.97   0.715
## s(O3):Period2      9.00  1.99    0.97   0.695
```

```
## s(O3):Period3     9.00   1.00    0.97   0.720
## s(Mean)           9.00   1.00    0.95   0.395
## s(Rainfall)       9.00   1.44    0.94   0.140
## s(Wind)           9.00   2.16    0.92   0.020 *
## s(doy)          363.00   3.01    0.97   0.695
## s(time)           9.00   2.54    0.91   0.005 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

### 3.3 Johor Bahru analysis

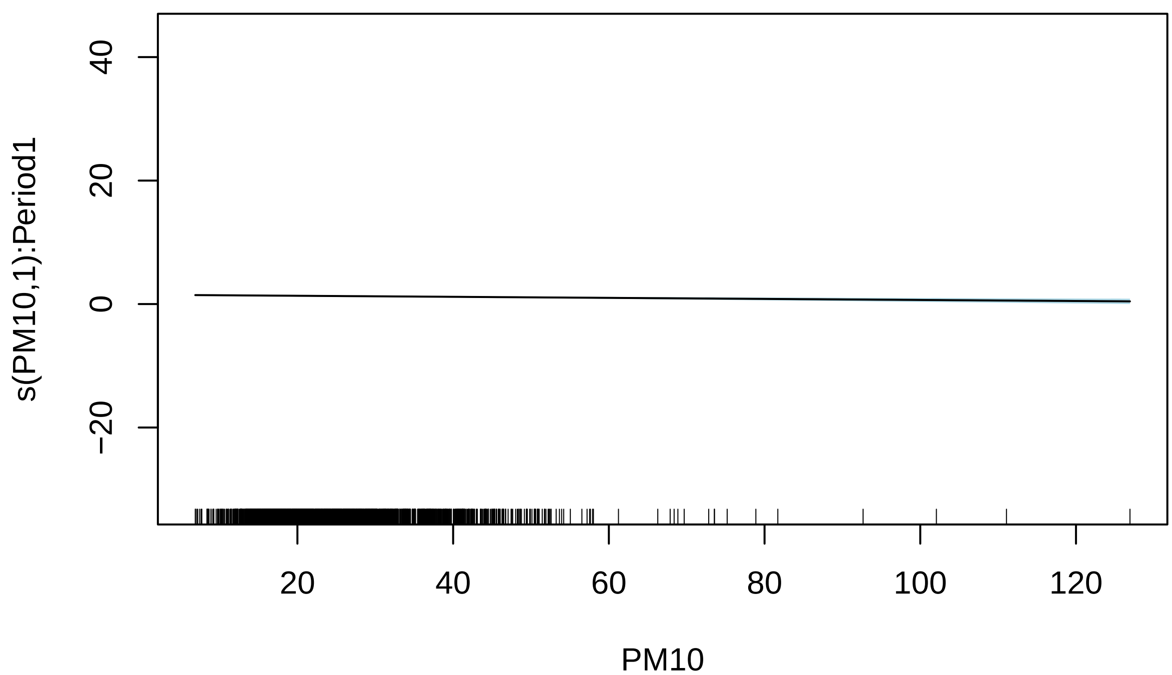
```
jb_CLD_period <- gam(CLD ~ s(PM10, by=Period) + s(NO2, by=Period) + s(CO3, by=Period) + s(SO2, by=Perio
summary(jb_CLD_period)
```

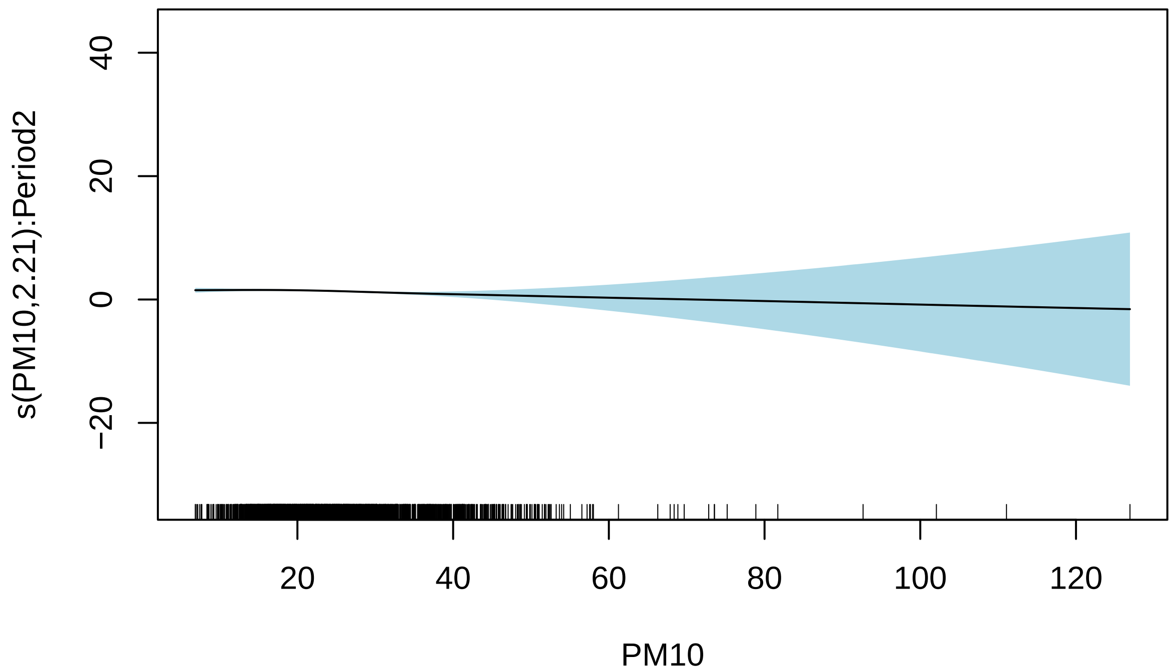
### JB CLD by period

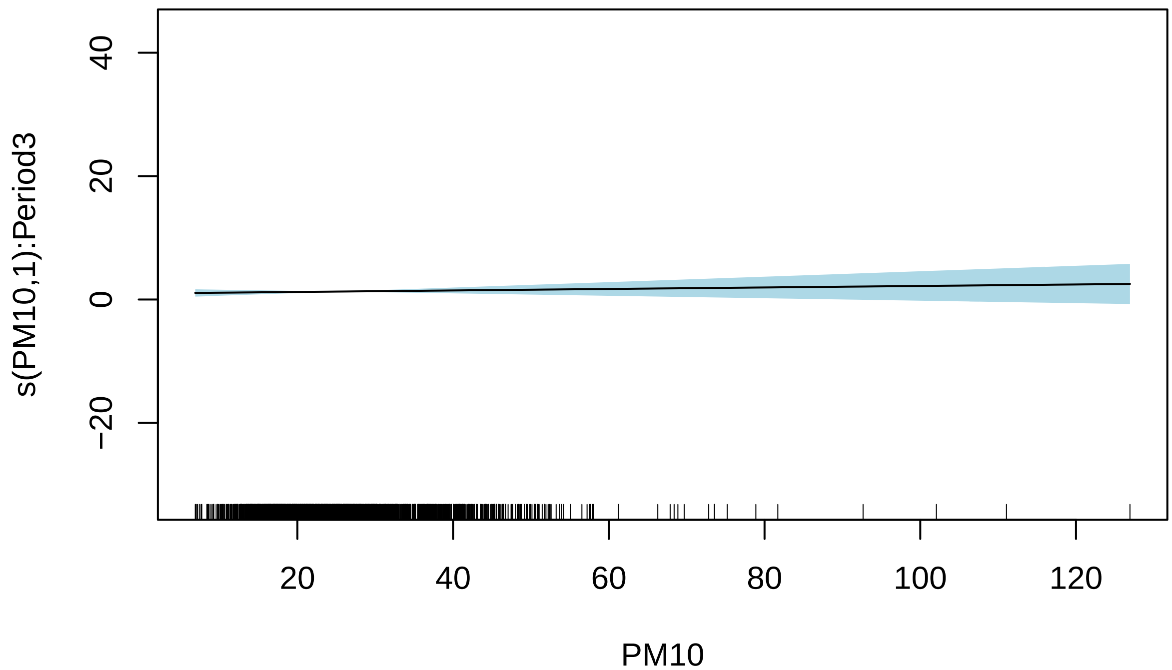
```
##
## Family: poisson
## Link function: log
##
## Formula:
## CLD ~ s(PM10, by = Period) + s(NO2, by = Period) + s(CO3, by = Period) +
##     s(SO2, by = Period) + s(O3, by = Period) + s(Mean) + s(Rainfall) +
##     s(Wind) + s(doy, bs = "cc", k = 365) + s(time, bs = "bs") +
##     Period
##
## Parametric coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  1.29339    0.07985  16.197  < 2e-16 ***
## Period2     -1.68835    0.23719  -7.118 1.09e-12 ***
## Period3     -1.16142    0.31733  -3.660 0.000252 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Approximate significance of smooth terms:
##                    edf  Ref.df  Chi.sq  p-value
## s(PM10):Period1  1.001   1.002  15.239 9.57e-05 ***
## s(PM10):Period2  2.212   2.778  10.042 0.027651 *
## s(PM10):Period3  1.000   1.000   0.579 0.446915
## s(NO2):Period1   1.139   1.262   0.020 0.950539
## s(NO2):Period2   1.522   1.882   4.165 0.080669 .
## s(NO2):Period3   1.001   1.003   1.546 0.213865
## s(CO3):Period1   1.001   1.001   0.007 0.938612
## s(CO3):Period2   2.413   3.128   3.411 0.354454
## s(CO3):Period3   1.196   1.356   1.944 0.318420
## s(SO2):Period1   2.439   3.076   5.214 0.178981
## s(SO2):Period2   3.069   3.676   8.544 0.157926
## s(SO2):Period3   1.001   1.001   2.071 0.150387
## s(O3):Period1    4.421   5.549  11.434 0.061607 .
## s(O3):Period2    2.722   3.415   5.903 0.110777
## s(O3):Period3    3.141   3.734  20.746 0.000324 ***
```

```
## s(Mean)           1.163   1.307   0.572 0.501561
## s(Rainfall)       1.000   1.000  10.868 0.000979 ***
## s(Wind)           5.099   6.146  36.322  < 2e-16 ***
## s(doy)           16.179 363.000  82.027  < 2e-16 ***
## s(time)           8.447   8.858 173.916  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-sq.(adj) =  0.277   Deviance explained = 30.1%
## -REML = 3507.8  Scale est. = 1         n = 1827
```

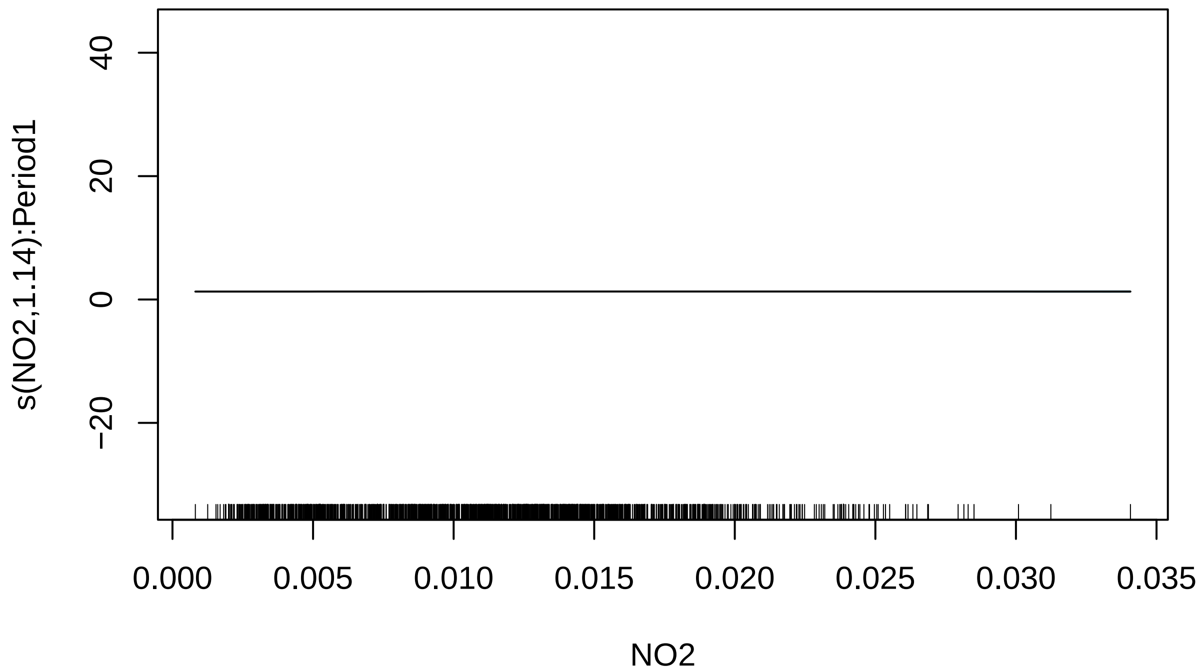
```
plot.gam(jb_CLD_period, shade = TRUE, shade.col = "lightblue", shift = coef(jb_CLD_period)[1],  seWithM
```

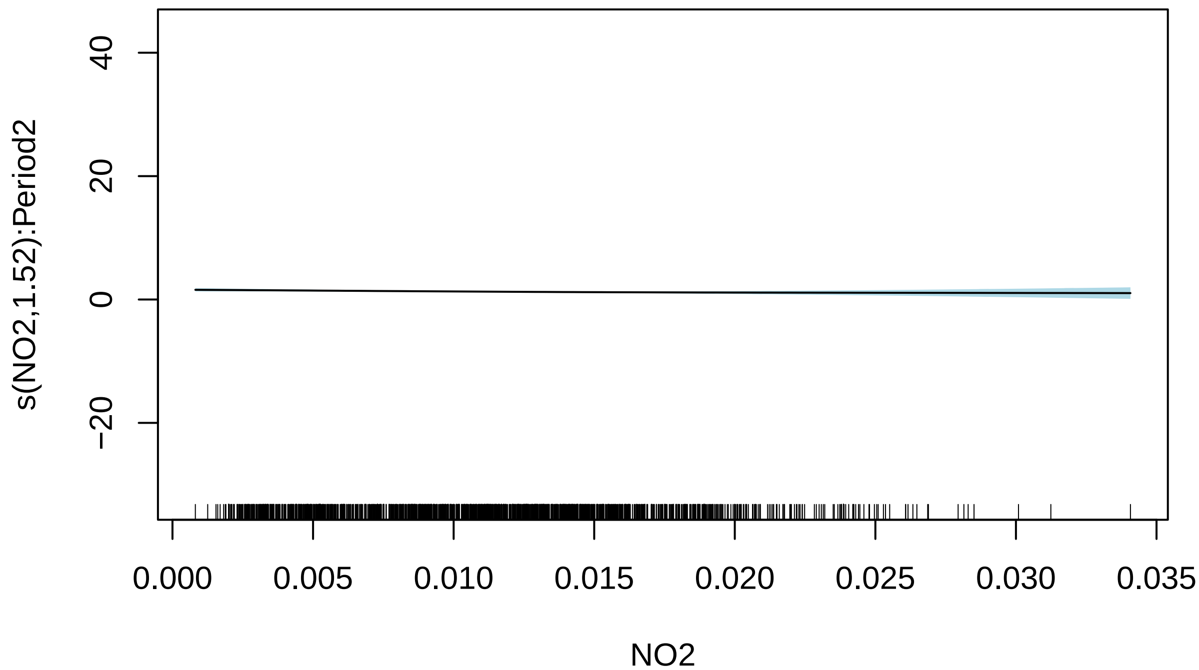


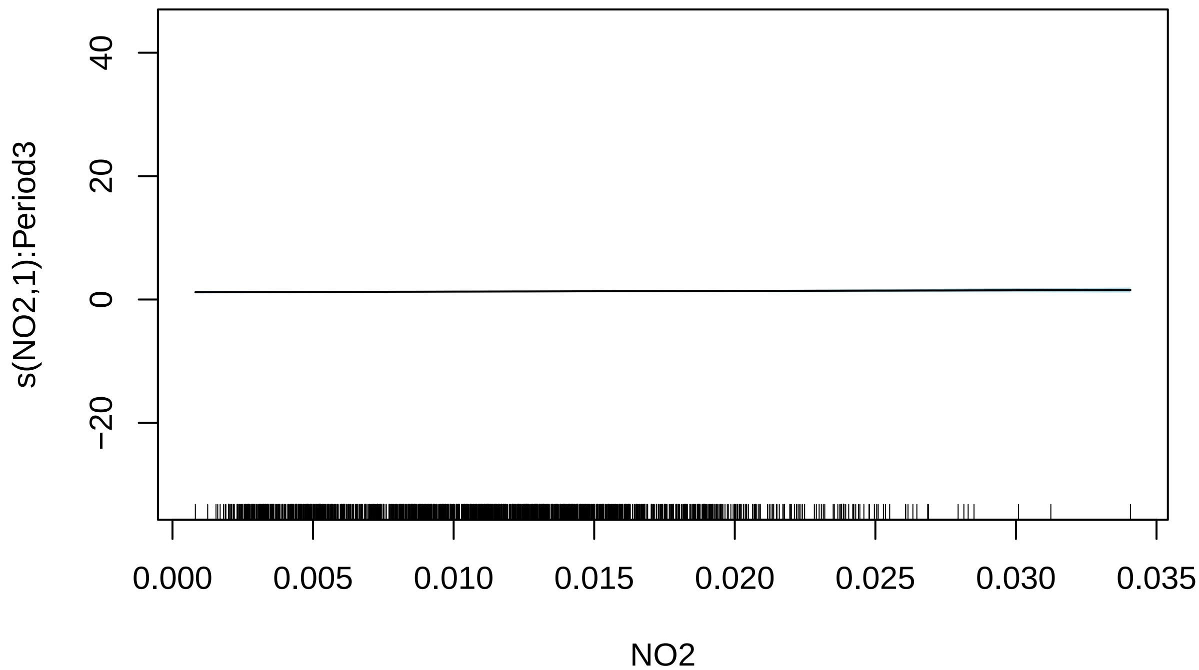

s(PM10,2.21):Period2
40
20
0
−20
20
40
60
80
100
120
PM10

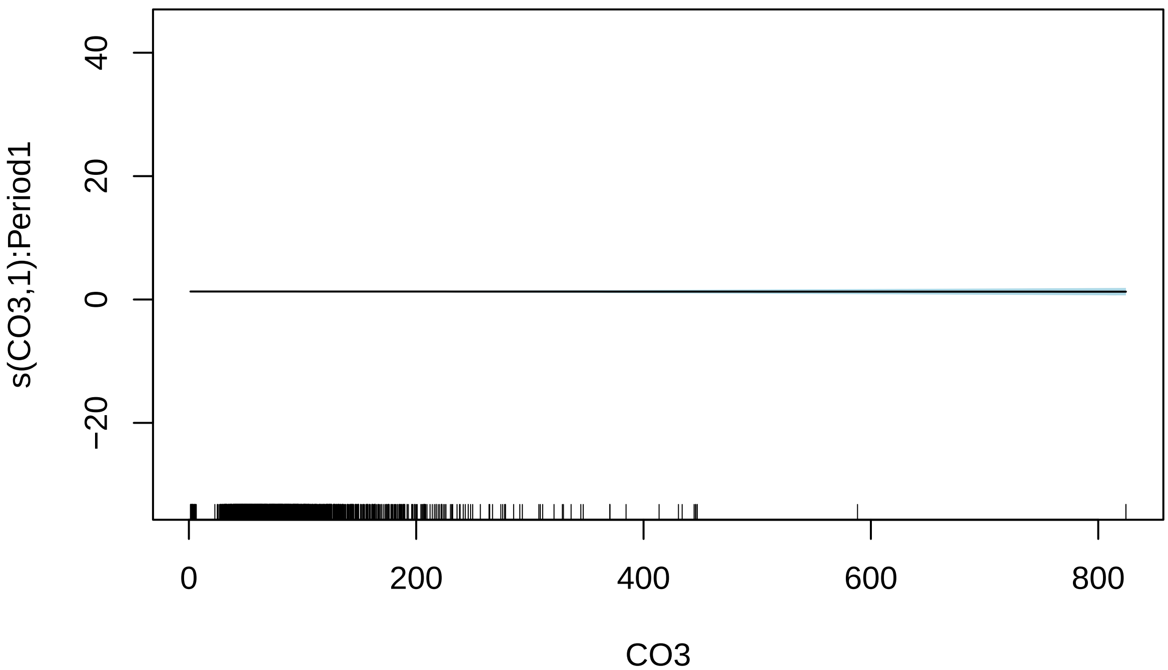


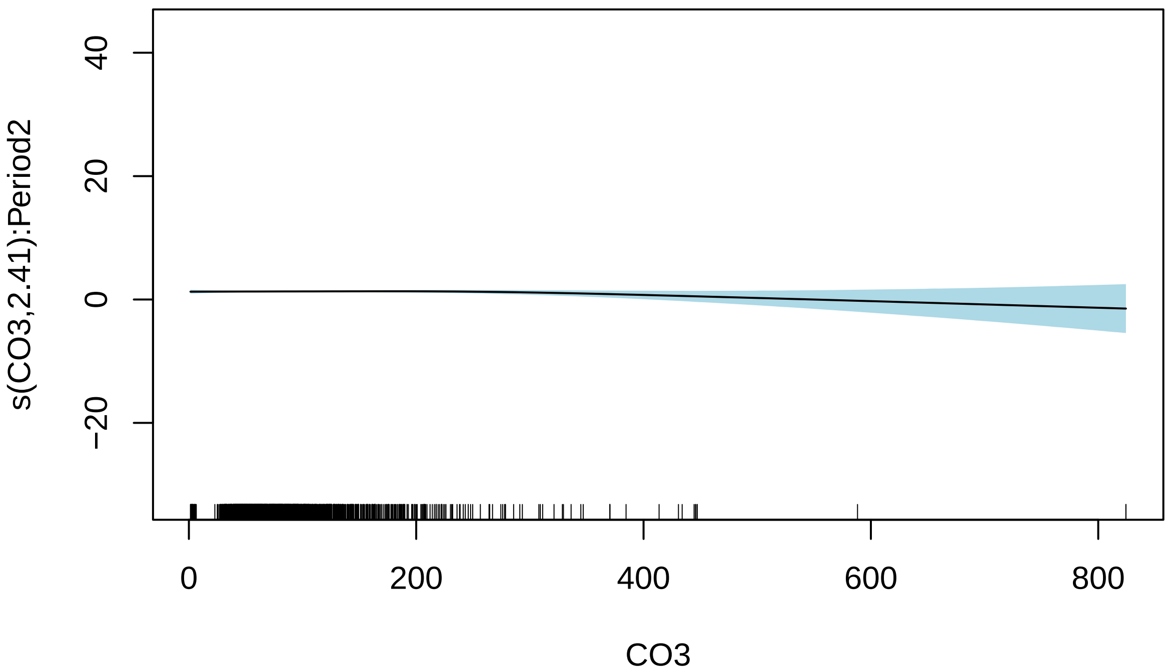
s(PM10,1):Period3
40
20
0
−20
20
40
60
80
100
120
PM10

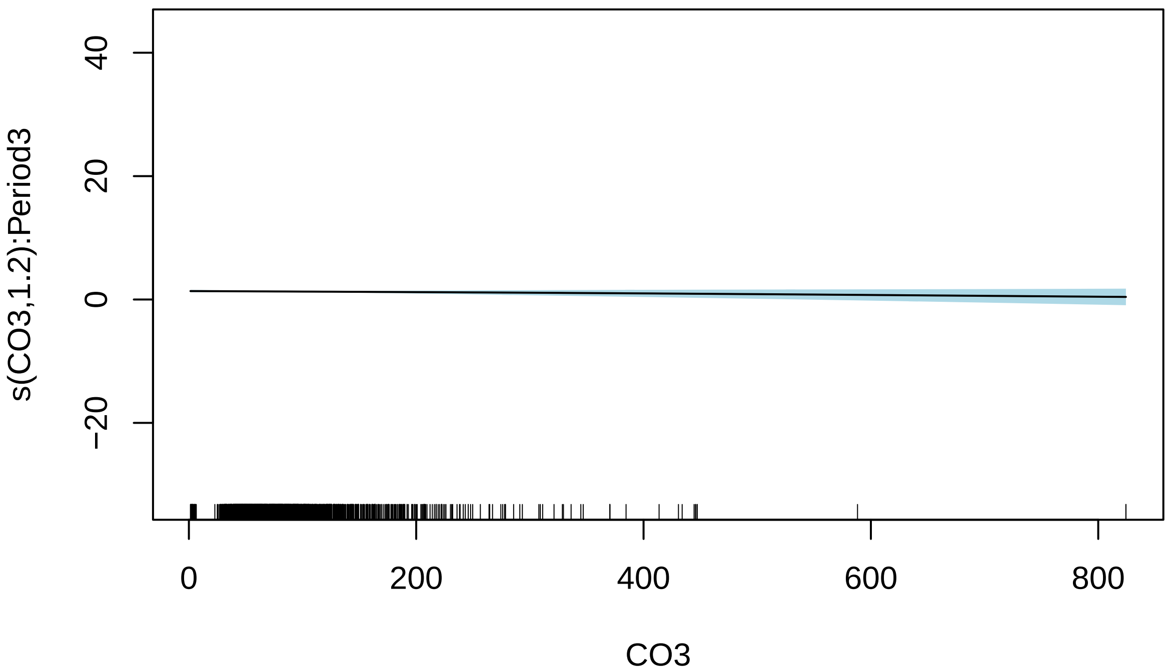

s(NO2,1.14):Period1
40
20
0
−20
0.000
0.005
0.010
0.015
0.020
0.025
0.030
0.035
NO2

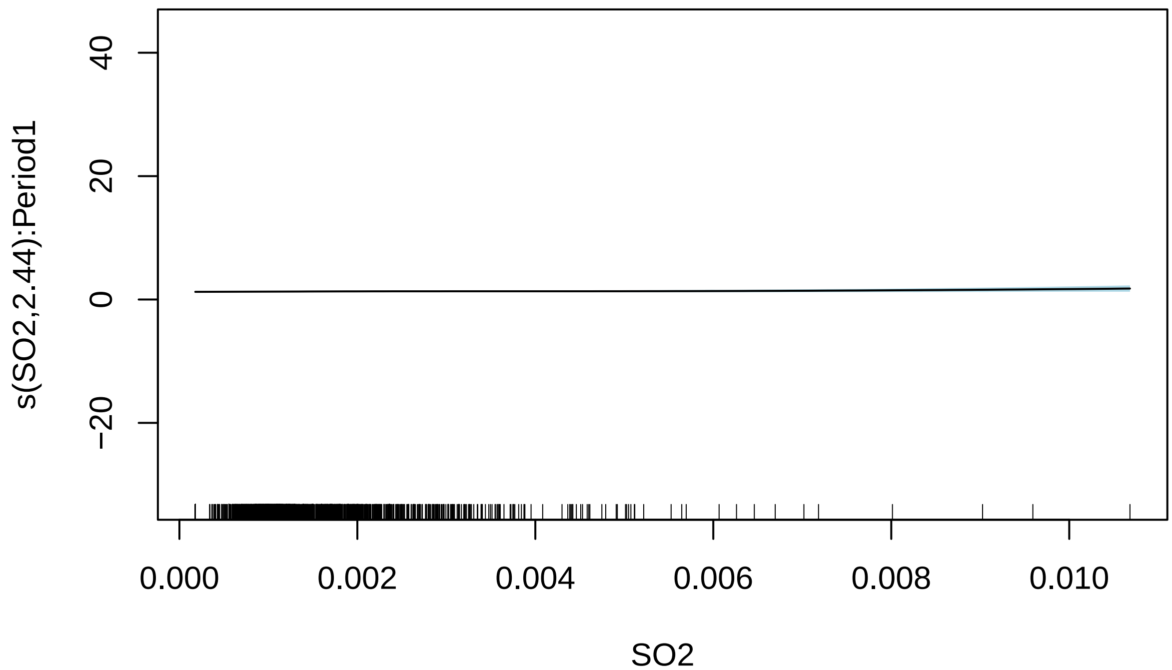

40
20
0
−20
s(NO2,1.52):Period2
0.000
0.005
0.010
0.015
0.020
0.025
0.030
0.035
NO2


s(NO2,1):Period3
40
20
0
−20
0.000
0.005
0.010
0.015
0.020
0.025
0.030
0.035
NO2

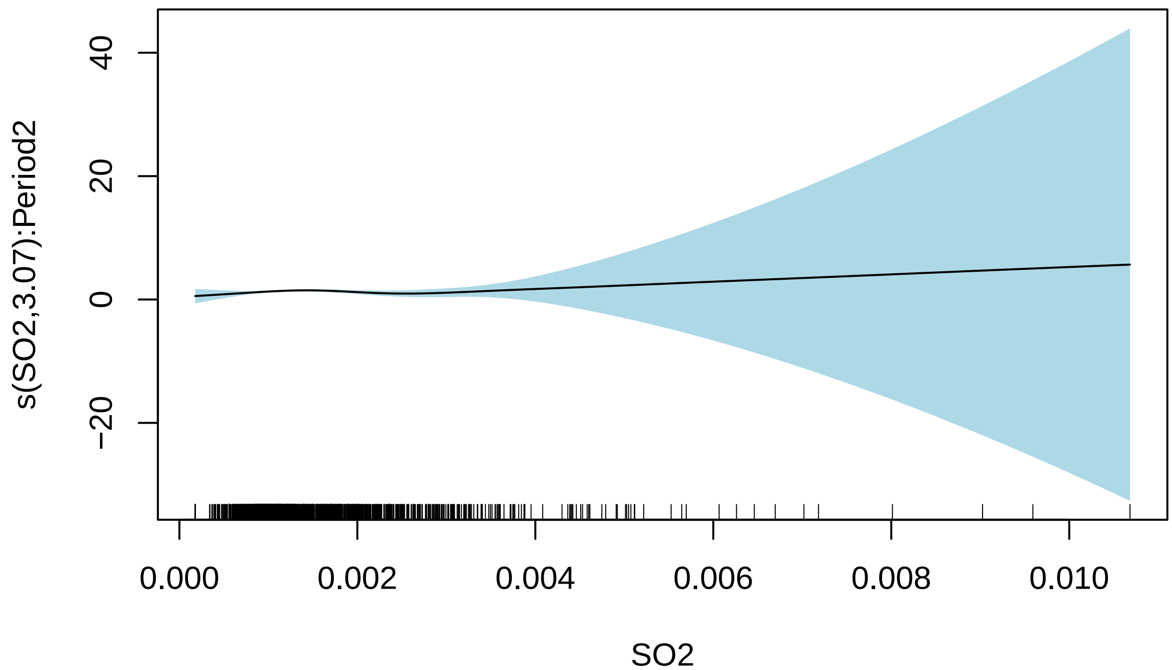

s(CO3,1):Period1
40
20
0
−20
0
200
400
600
800
CO3


s(CO3,2.41):Period2
40
20
0
−20
0
200
400
600
800
CO3

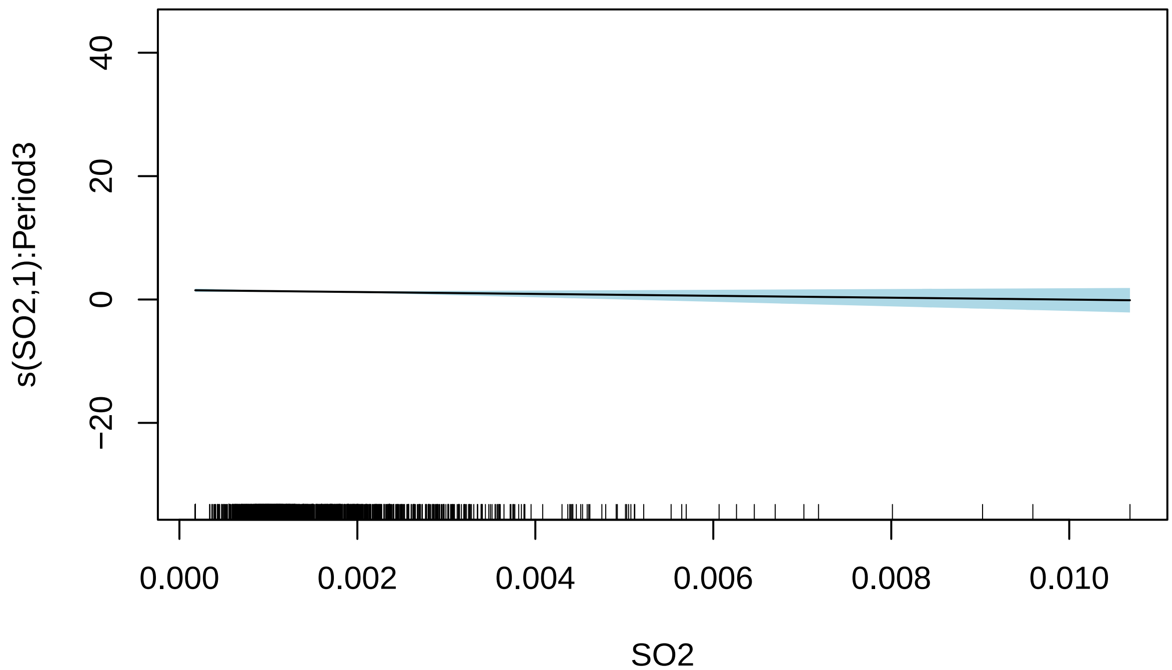

s(CO3,1.2):Period3
40
20
0
−20
0
200
400
600
800
CO3



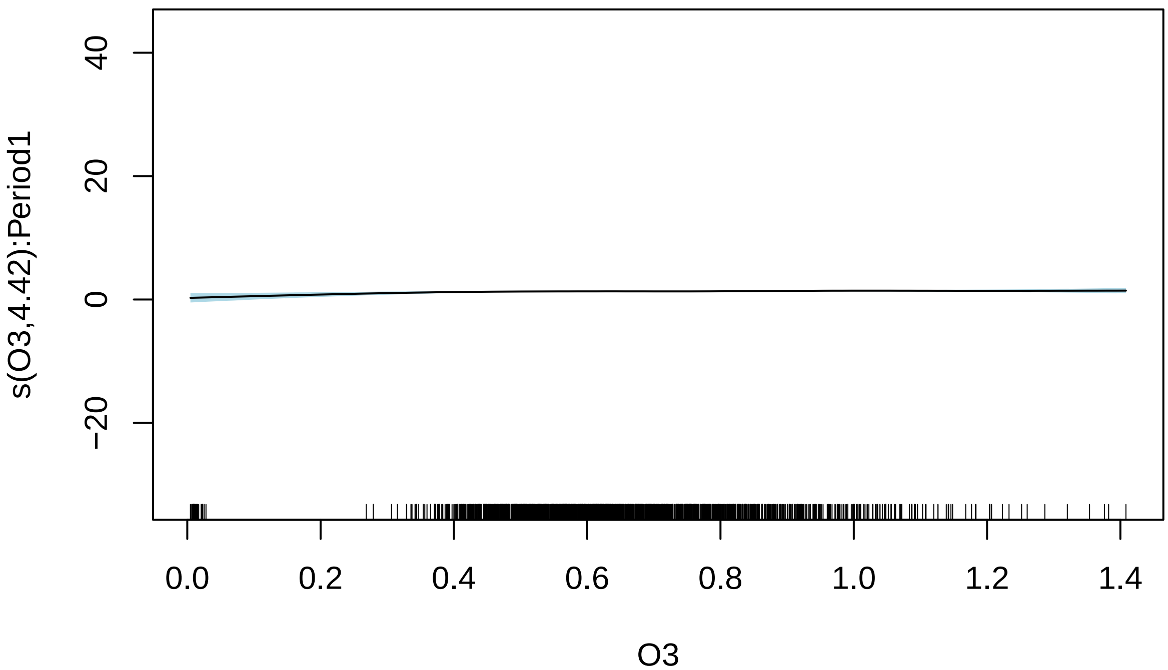
40
20
0
−20
s(SO2,2.44):Period1
0.000
0.002
0.004
0.006
0.008
0.010
SO2



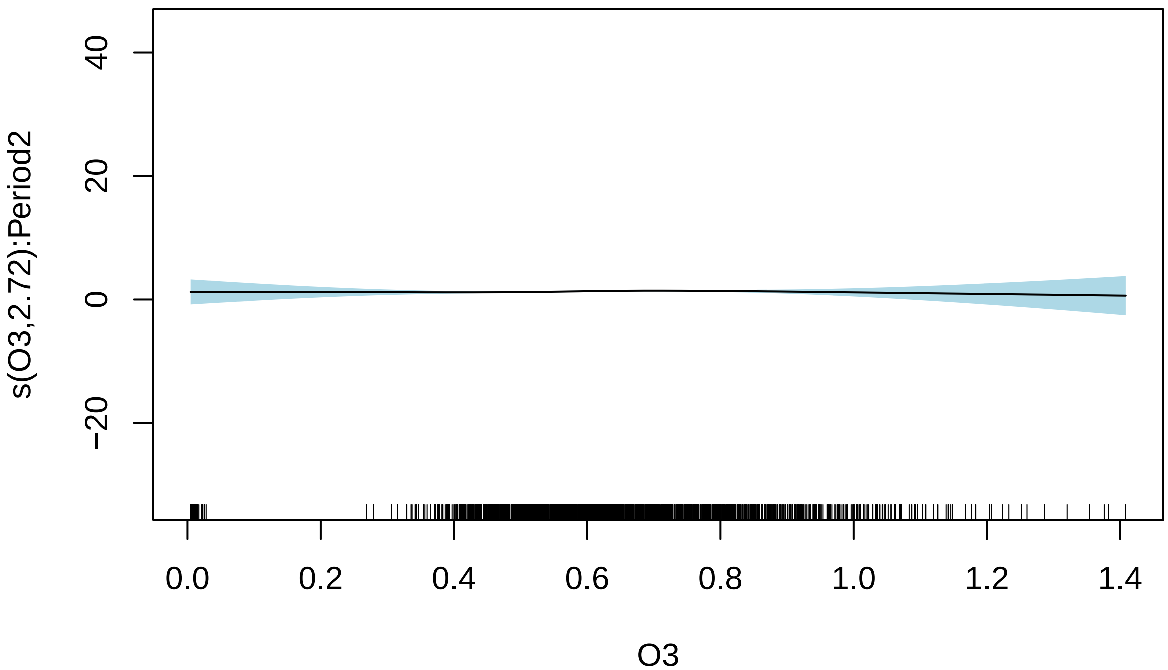
40
20
0
−20
s(SO2,3.07):Period2
0.000
0.002
0.004
0.006
0.008
0.010
SO2



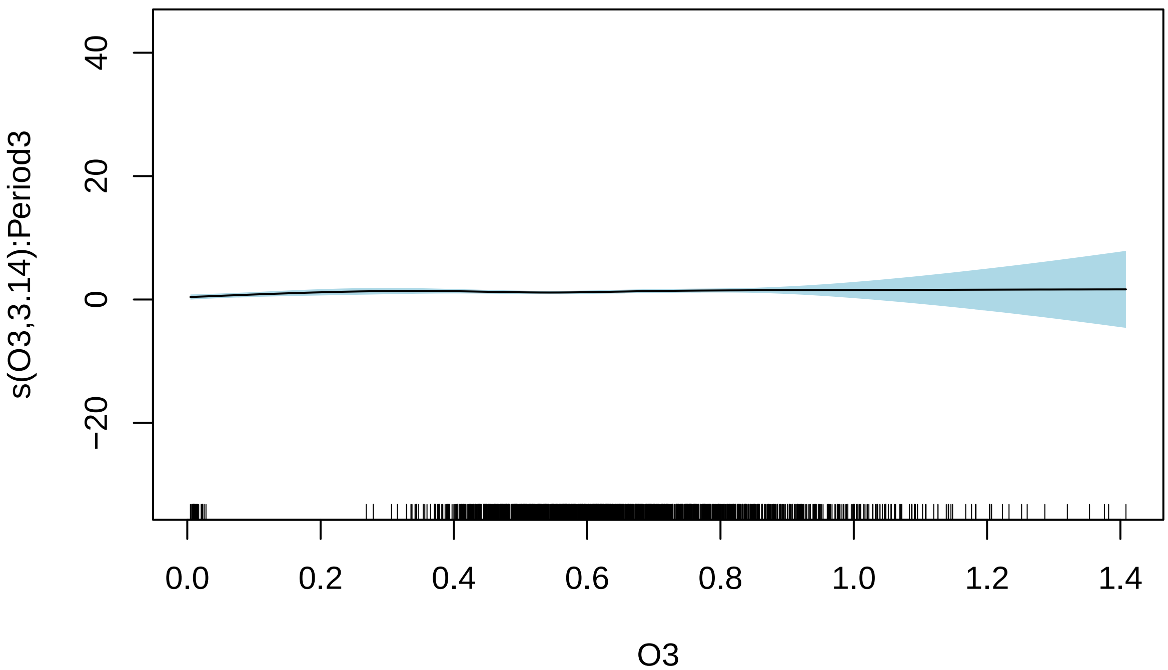
40
20
0
−20
s(SO2,1):Period3
0.000
0.002
0.004
0.006
0.008
0.010
SO2

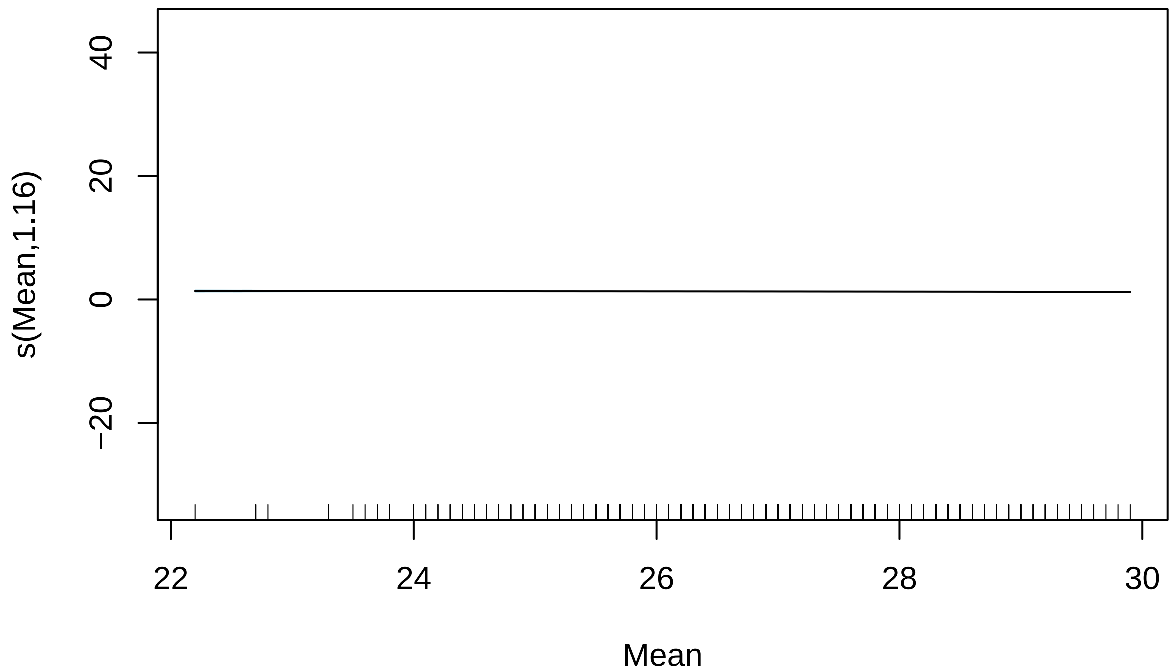


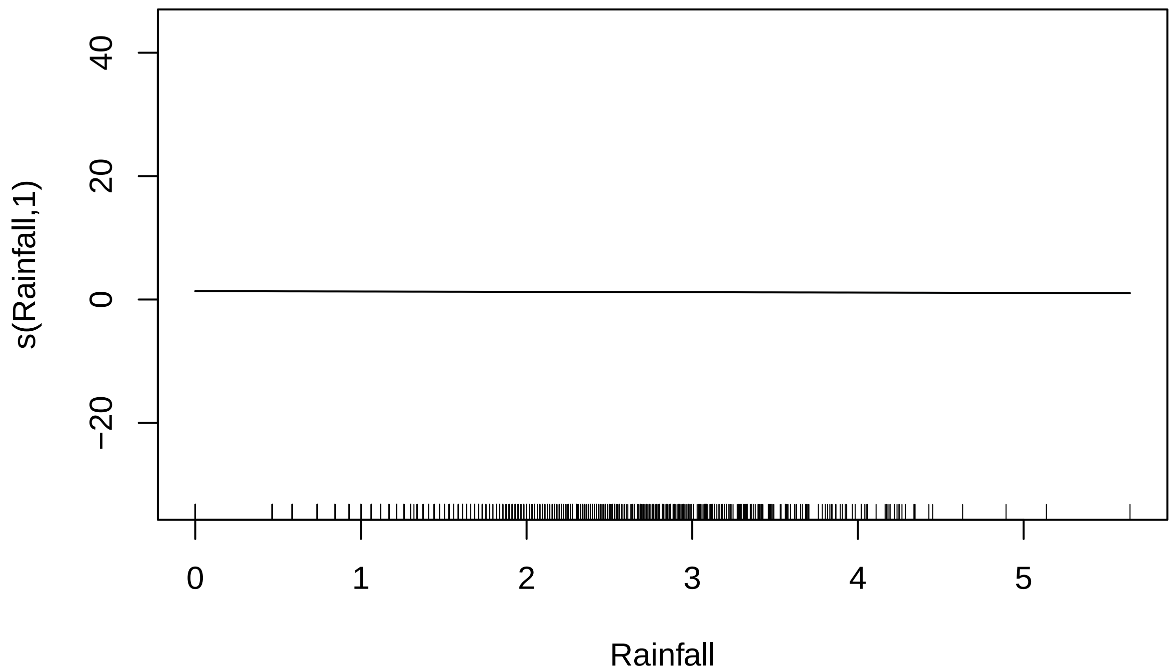
s(O3,4.42):Period1
40
20
0
−20
0.0
0.2
0.4
0.6
0.8
1.0
1.2
1.4
O3



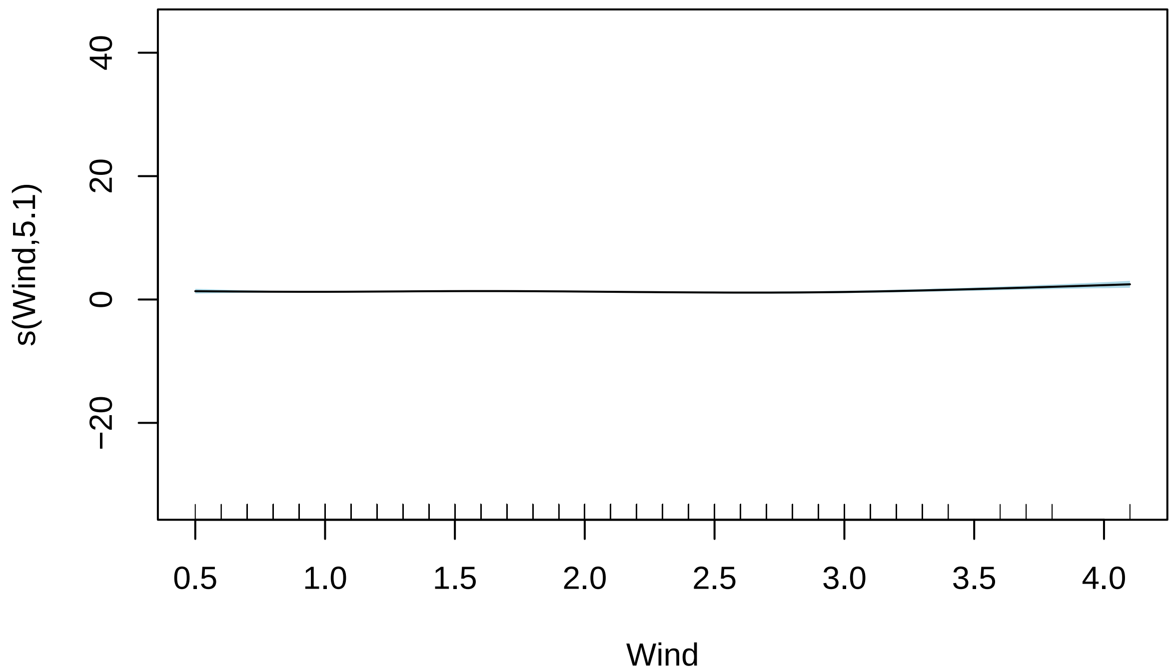
s(O3,2.72):Period2
40
20
0
−20
0.0
0.2
0.4
0.6
0.8
1.0
1.2
1.4
O3

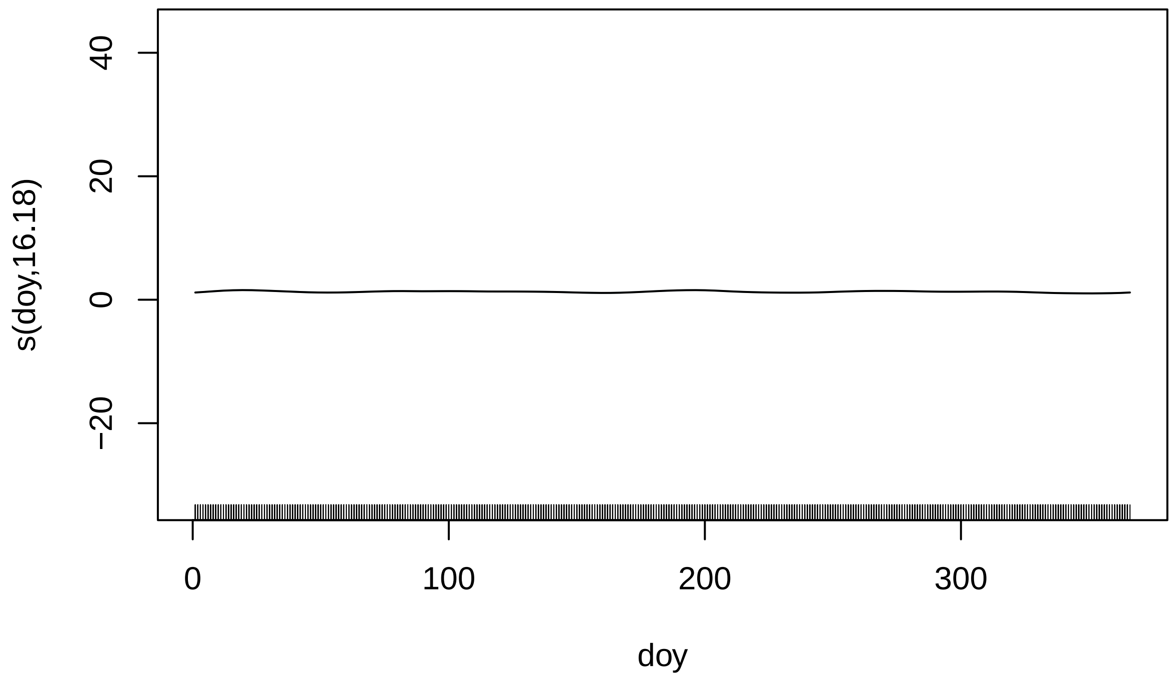

40
20
0
−20
s(O3,3.14):Period3
0.0
0.2
0.4
0.6
0.8
1.0
1.2
1.4
O3

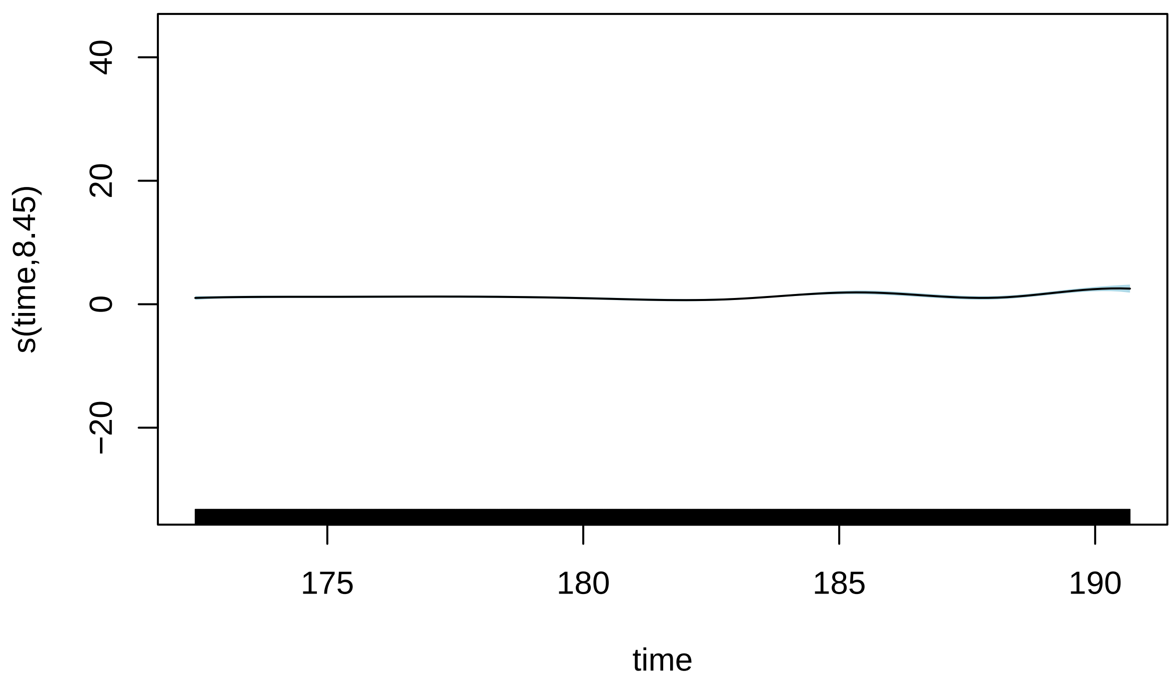

s(Mean,1.16)
40
20
0
−20
22
24
26
28
30
Mean



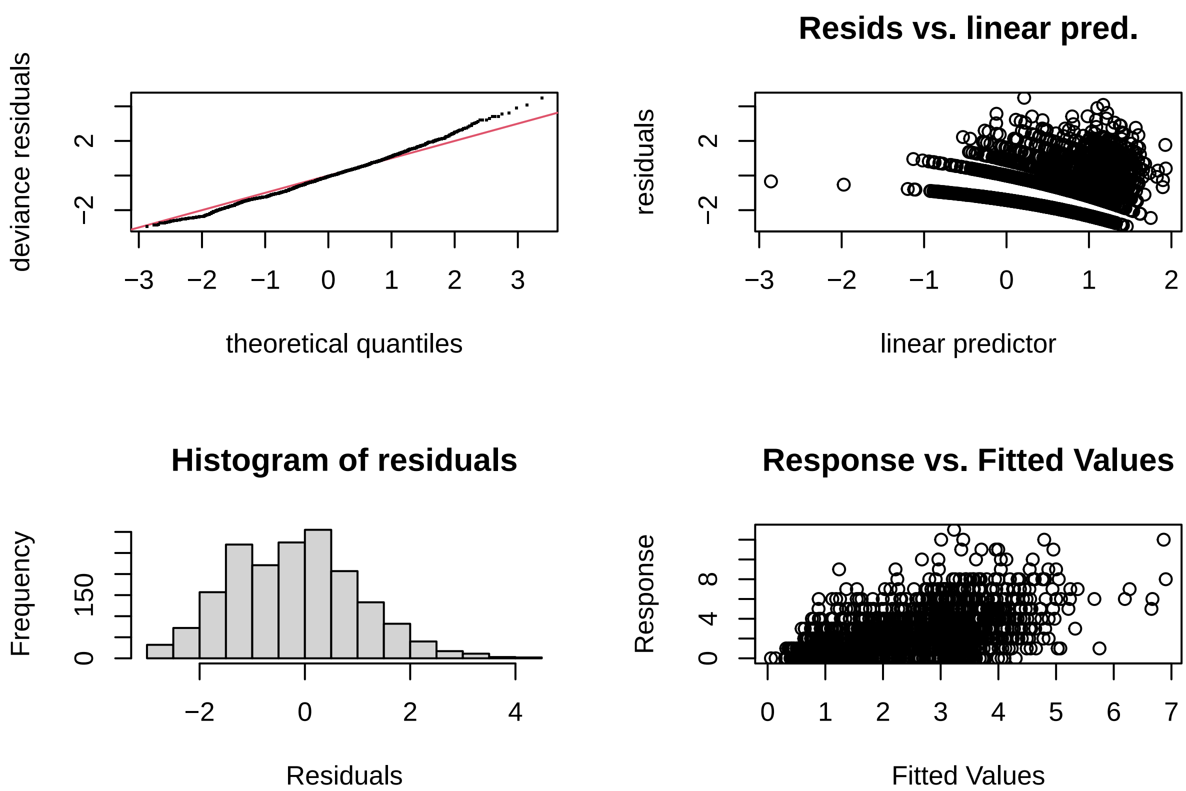
40
20
0
−20
s(Rainfall,1)
0
1
2
3
4
5
Rainfall


s(Wind,5.1)
40
20
0
−20
0.5
1.0
1.5
2.0
2.5
3.0
3.5
4.0
Wind


40
20
0
−20
s(doy,16.18)
0
100
200
300
doy




```
gam.check(jb_CLD_period)
```




```
##
## Method: REML   Optimizer: outer newton
## full convergence after 11 iterations.
## Gradient range [-0.0003259048,0.0001096532]
## (score 3507.784 & scale 1).
## Hessian positive definite, eigenvalue range [2.145479e-05,3.705023].
## Model rank =  537 / 537
##
## Basis dimension (k) checking results. Low p-value (k-index<1) may
## indicate that k is too low, especially if edf is close to k'.
##
##                     k'   edf k-index p-value
## s(PM10):Period1   9.00  1.00    0.98   0.535
## s(PM10):Period2   9.00  2.21    0.98   0.560
## s(PM10):Period3   9.00  1.00    0.98   0.640
## s(NO2):Period1    9.00  1.14    0.96   0.335
## s(NO2):Period2    9.00  1.52    0.96   0.385
## s(NO2):Period3    9.00  1.00    0.96   0.335
## s(CO3):Period1    9.00  1.00    1.03   0.995
## s(CO3):Period2    9.00  2.41    1.03   0.995
## s(CO3):Period3    9.00  1.20    1.03   0.995
## s(SO2):Period1    9.00  2.44    0.93   0.045 *
## s(SO2):Period2    9.00  3.07    0.93   0.025 *
## s(SO2):Period3    9.00  1.00    0.93   0.020 *
## s(O3):Period1     9.00  4.42    0.99   0.705
## s(O3):Period2     9.00  2.72    0.99   0.760
```

```
## s(O3):Period3     9.00   3.14    0.99   0.730
## s(Mean)           9.00   1.16    0.91   0.015 *
## s(Rainfall)       9.00   1.00    0.94   0.100 .
## s(Wind)           9.00   5.10    0.94   0.050 *
## s(doy)          363.00  16.18    0.92   0.005 **
## s(time)           9.00   8.45    0.84  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

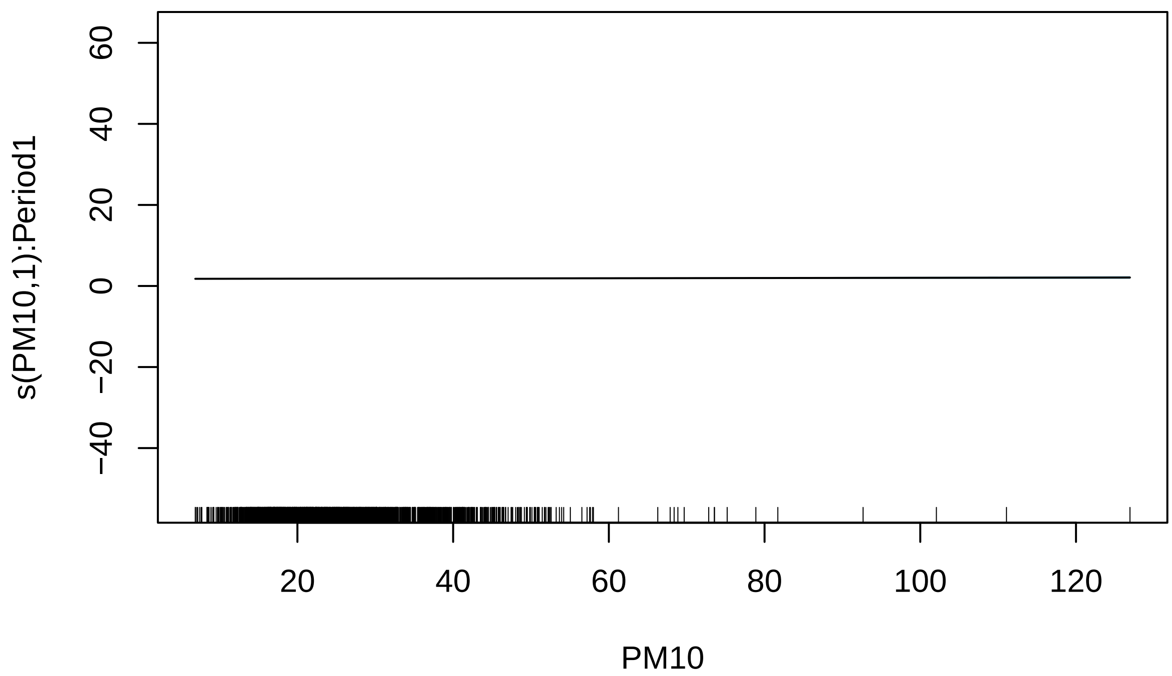
```
jb_LRTI_period <- gam(LRTI ~ s(PM10, by=Period) + s(NO2, by=Period) + s(CO3, by=Period) + s(SO2, by=Per
summary(jb_LRTI_period)
```

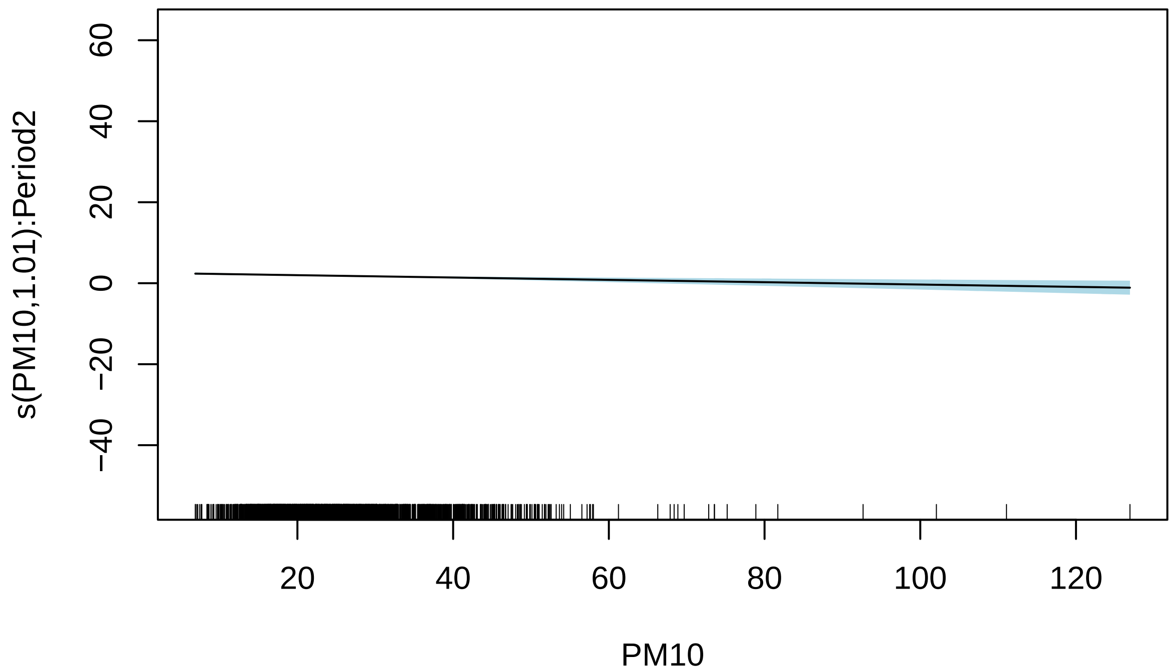
**JB LRTI by period**

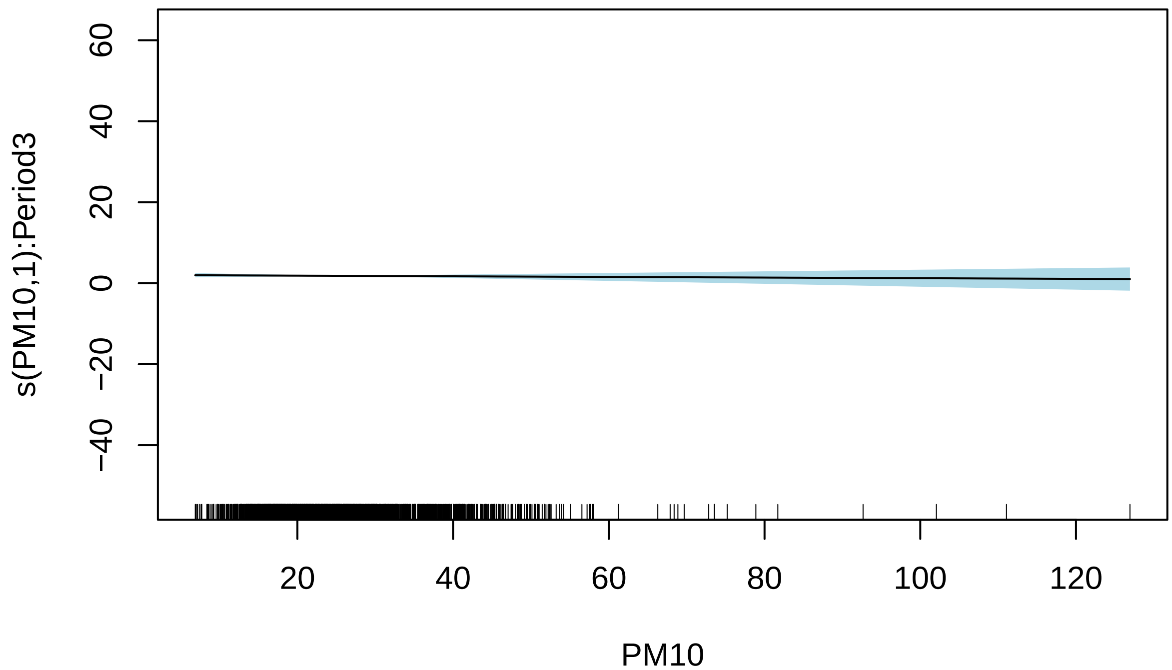
```
##
## Family: poisson
## Link function: log
##
## Formula:
## LRTI ~ s(PM10, by = Period) + s(NO2, by = Period) + s(CO3, by = Period) +
##     s(SO2, by = Period) + s(O3, by = Period) + s(Mean) + s(Rainfall) +
##     s(Wind) + s(doy, bs = "cc", k = 365) + s(time, bs = "bs") +
##     Period
##
## Parametric coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  1.82018    0.08647  21.051  < 2e-16 ***
## Period2     -2.73624    0.27286 -10.028  < 2e-16 ***
## Period3     -2.39548    0.32425  -7.388 1.49e-13 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Approximate significance of smooth terms:
##                    edf  Ref.df  Chi.sq  p-value
## s(PM10):Period1  1.001   1.003   1.974 0.160263
## s(PM10):Period2  1.011   1.021  14.759 0.000149 ***
## s(PM10):Period3  1.000   1.001   0.317 0.573866
## s(NO2):Period1   1.001   1.001   0.067 0.796734
## s(NO2):Period2   1.524   1.881   2.584 0.333405
## s(NO2):Period3   1.001   1.003   0.006 0.947910
## s(CO3):Period1   1.960   2.448   2.925 0.295761
## s(CO3):Period2   3.025   3.806   8.595 0.059082 .
## s(CO3):Period3   1.000   1.001   2.662 0.102907
## s(SO2):Period1   1.649   2.076   0.934 0.616670
## s(SO2):Period2   3.567   4.138  18.256 0.000923 ***
## s(SO2):Period3   1.903   2.119   4.308 0.090864 .
## s(O3):Period1    3.476   4.447   6.874 0.199254
## s(O3):Period2    3.932   4.772  58.997  < 2e-16 ***
## s(O3):Period3    1.000   1.000   8.295 0.003977 **
## s(Mean)          1.000   1.000   6.104 0.013503 *
## s(Rainfall)      2.277   2.849   4.104 0.334666
```

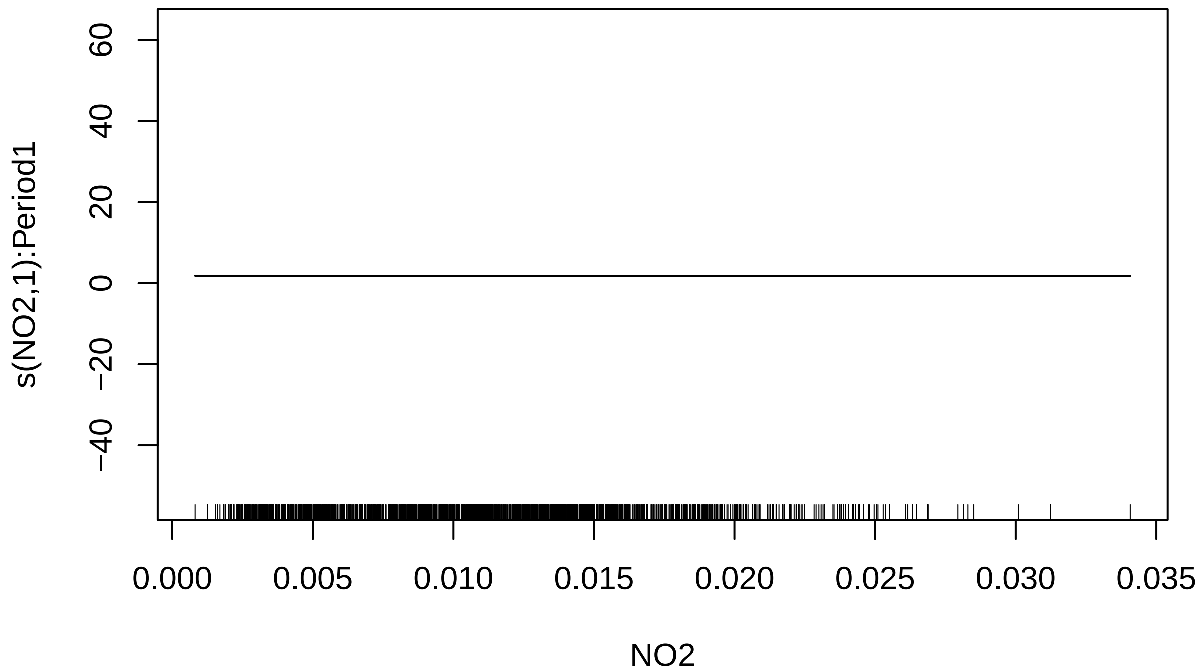
```
## s(Wind)          2.607   3.323  11.377 0.013674 *
## s(doy)          18.727 363.000 111.296  < 2e-16 ***
## s(time)          7.837   8.503 186.745  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-sq.(adj) =  0.338   Deviance explained = 37.1%
## -REML = 3653.1  Scale est. = 1         n = 1827
```

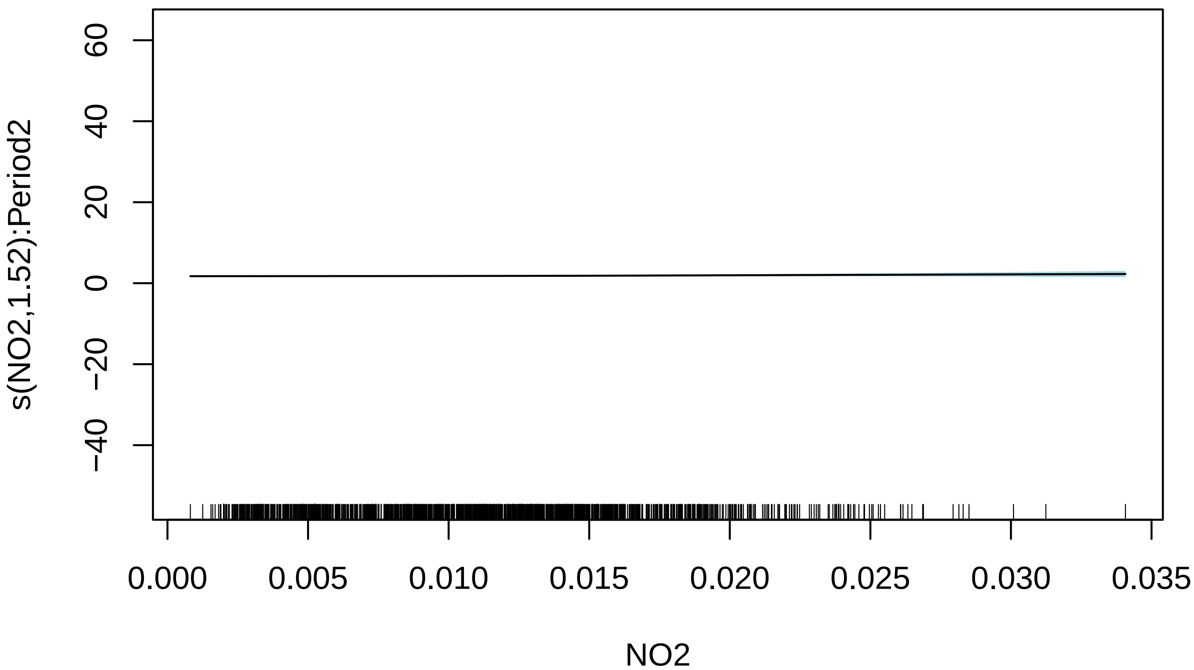
```
plot.gam(jb_LRTI_period, shade = TRUE, shade.col = "lightblue", shift = coef(jb_LRTI_period)[1],  seWit
```



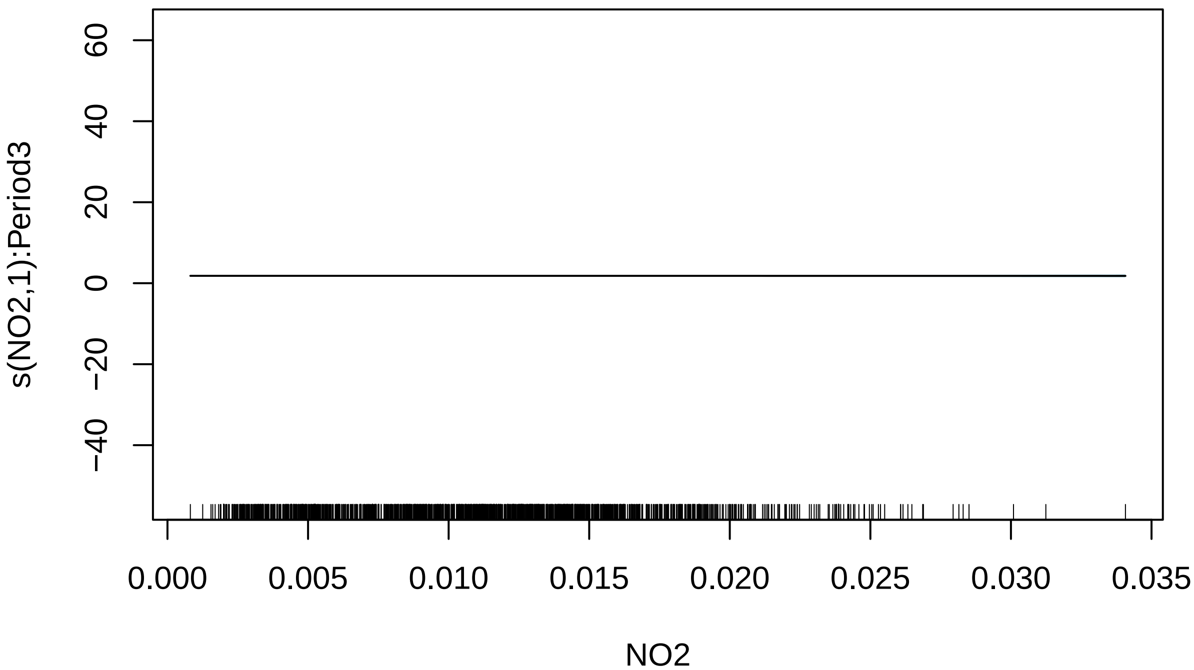

s(PM10,1.01):Period2
60
40
20
0
−20
−40
20
40
60
80
100
120
PM10

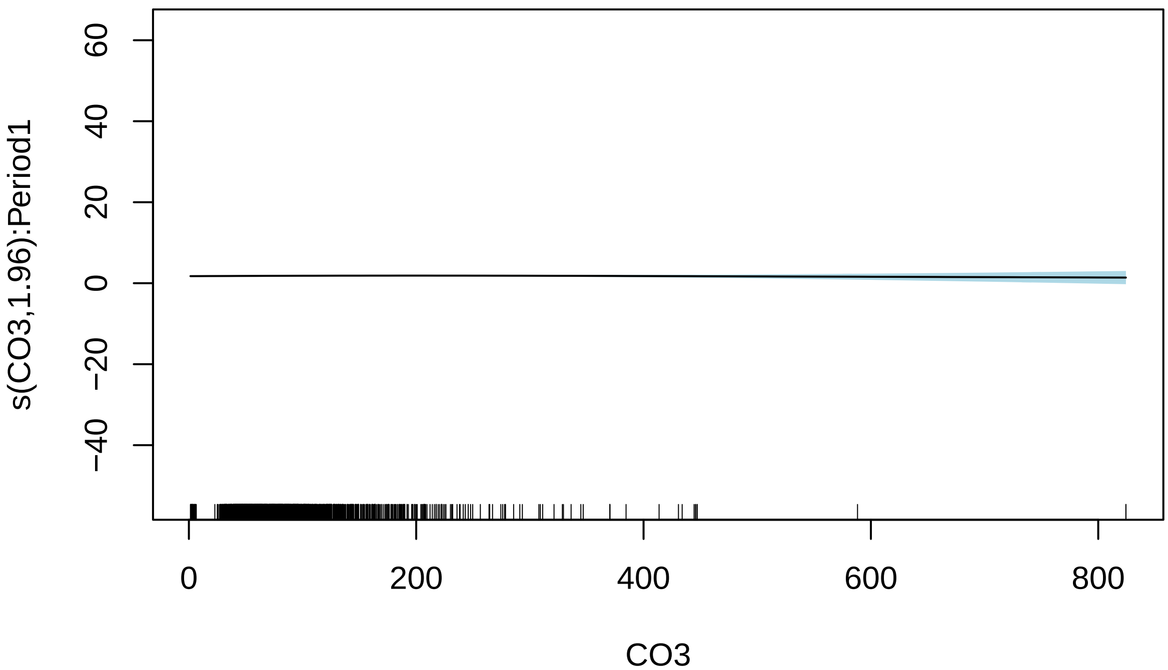

s(PM10,1):Period3
60
40
20
0
−20
−40
20
40
60
80
100
120
PM10


s(NO2,1):Period1
60
40
20
0
−20
−40
0.000
0.005
0.010
0.015
0.020
0.025
0.030
0.035
NO2

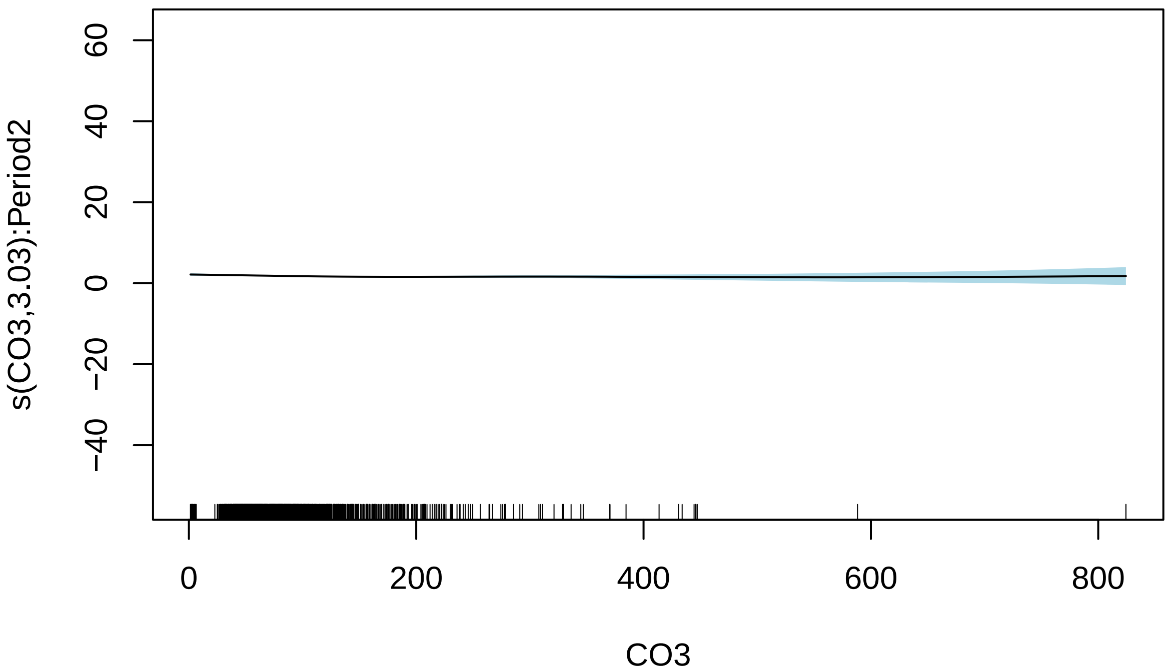


s(NO2,1.52):Period2
60
40
20
0
−20
−40
0.000
0.005
0.010
0.015
0.020
0.025
0.030
0.035
NO2

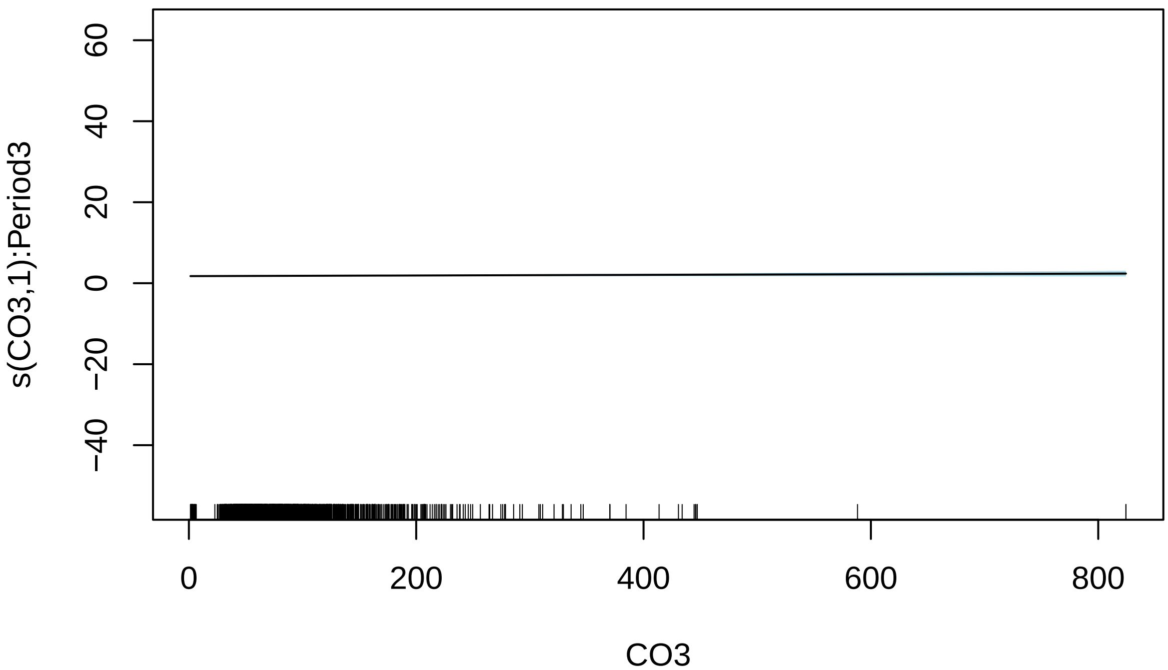

s(NO2,1):Period3
60
40
20
0
−20
−40
0.000
0.005
0.010
0.015
0.020
0.025
0.030
0.035
NO2



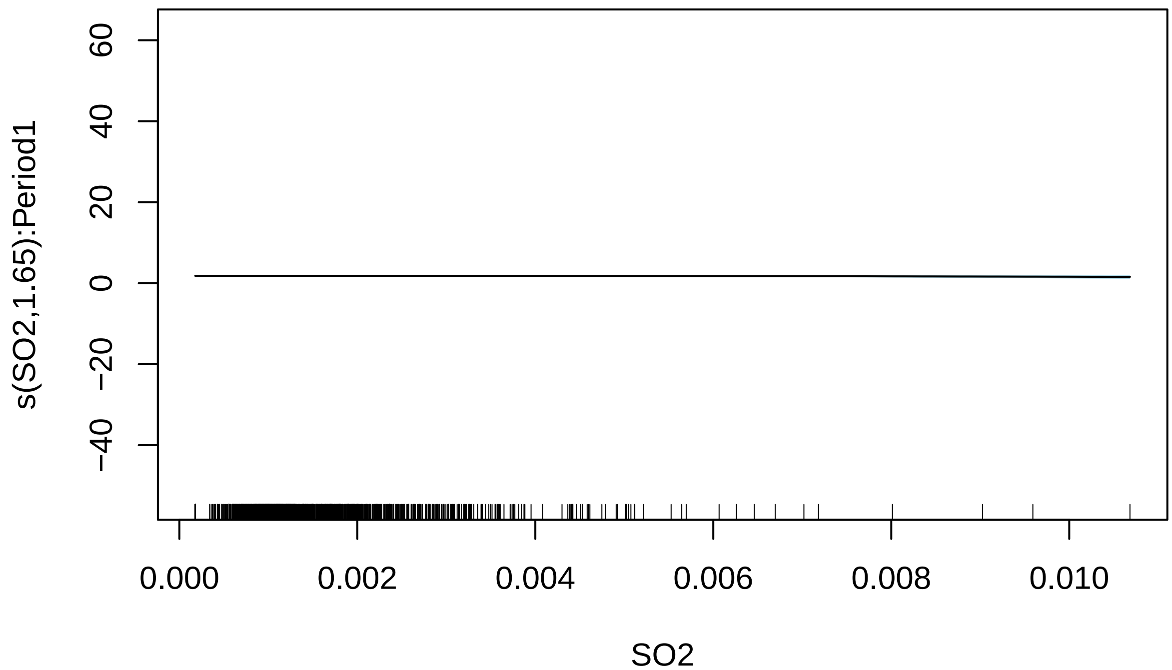
s(CO3,1.96):Period1
60
40
20
0
−20
−40
0
200
400
600
800
CO3

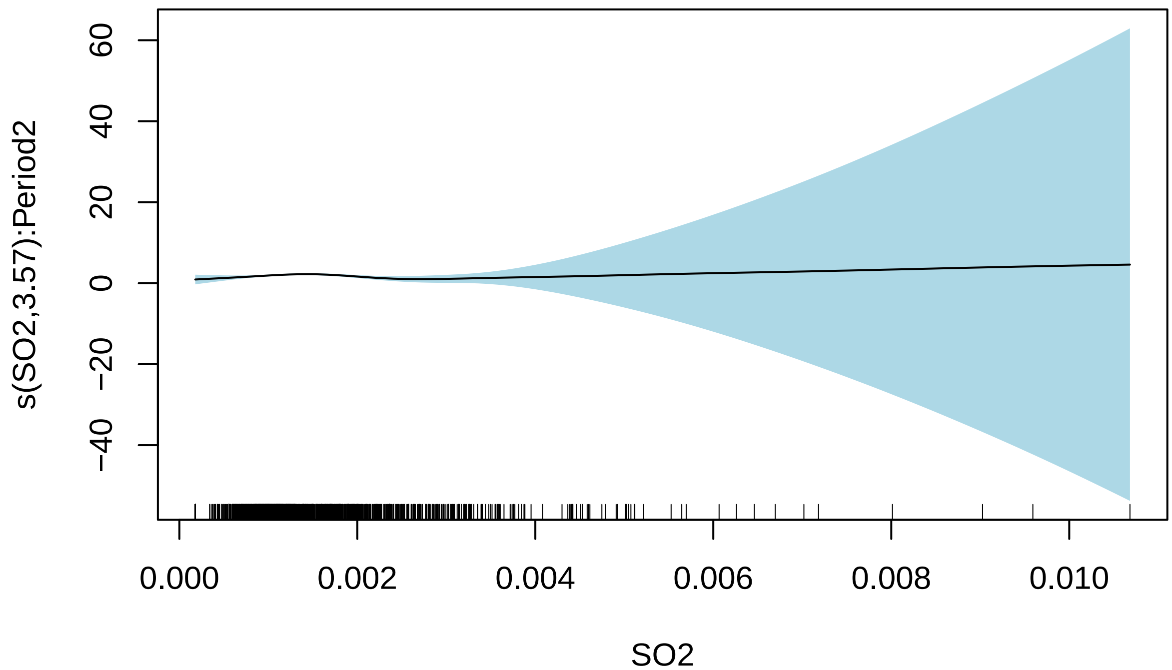


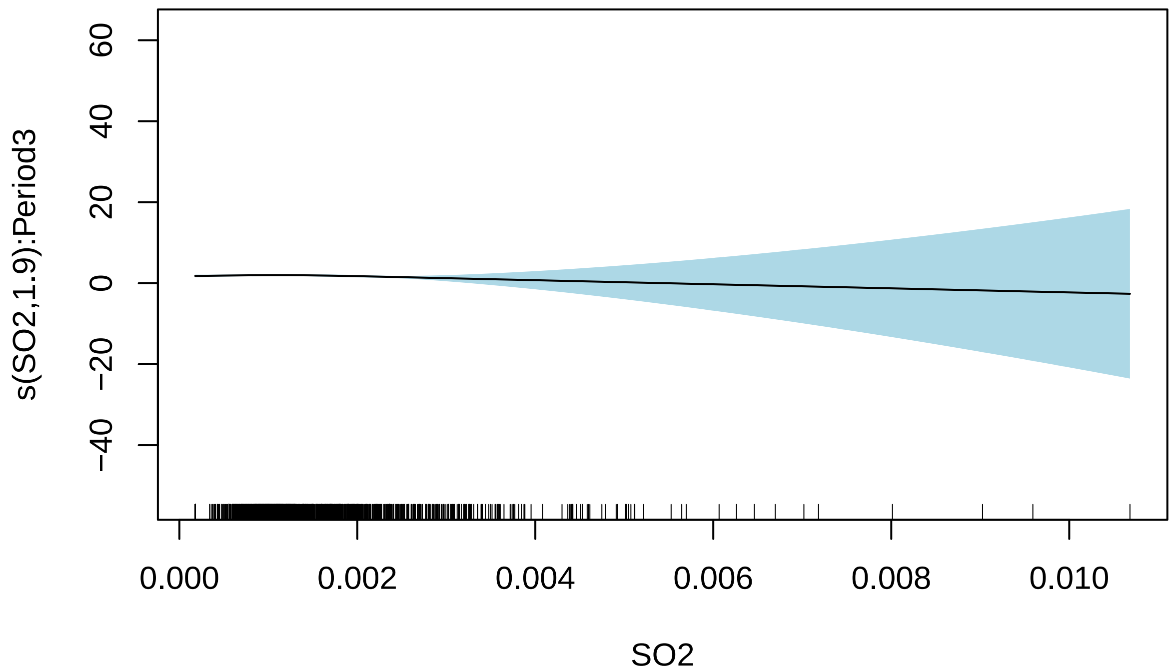
s(CO3,3.03):Period2
60
40
20
0
−20
−40
0
200
400
600
800
CO3



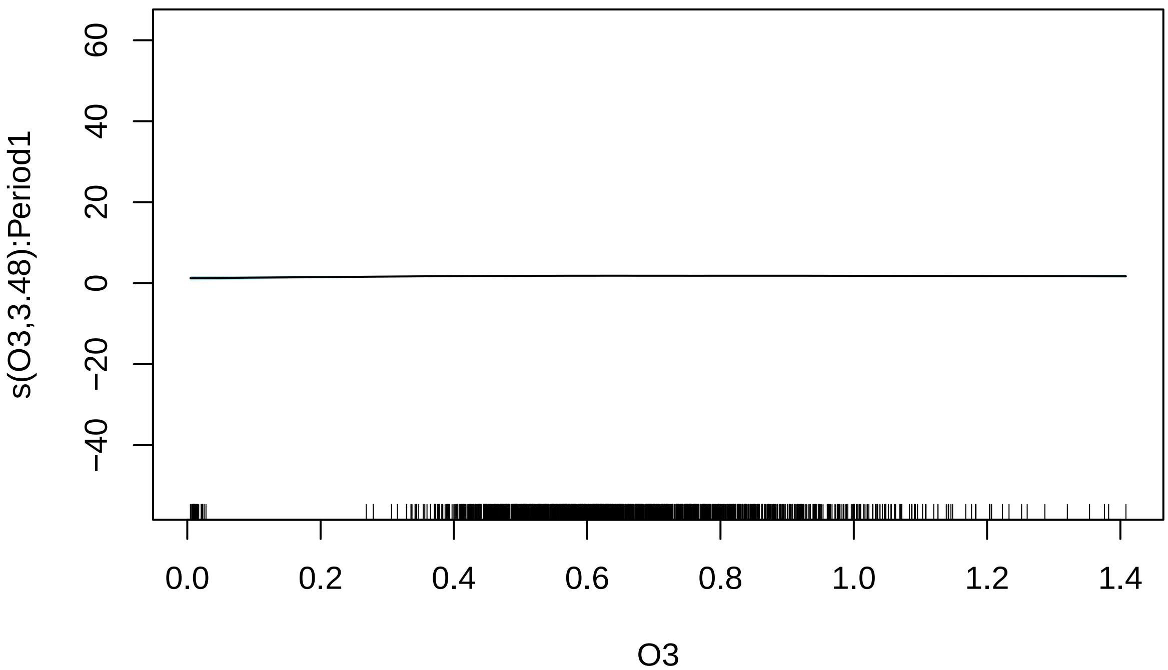
s(CO3,1):Period3
60
40
20
0
−20
−40
0
200
400
600
800
CO3


s(SO2,1.65):Period1
60
40
20
0
−20
−40
0.000
0.002
0.004
0.006
0.008
0.010
SO2

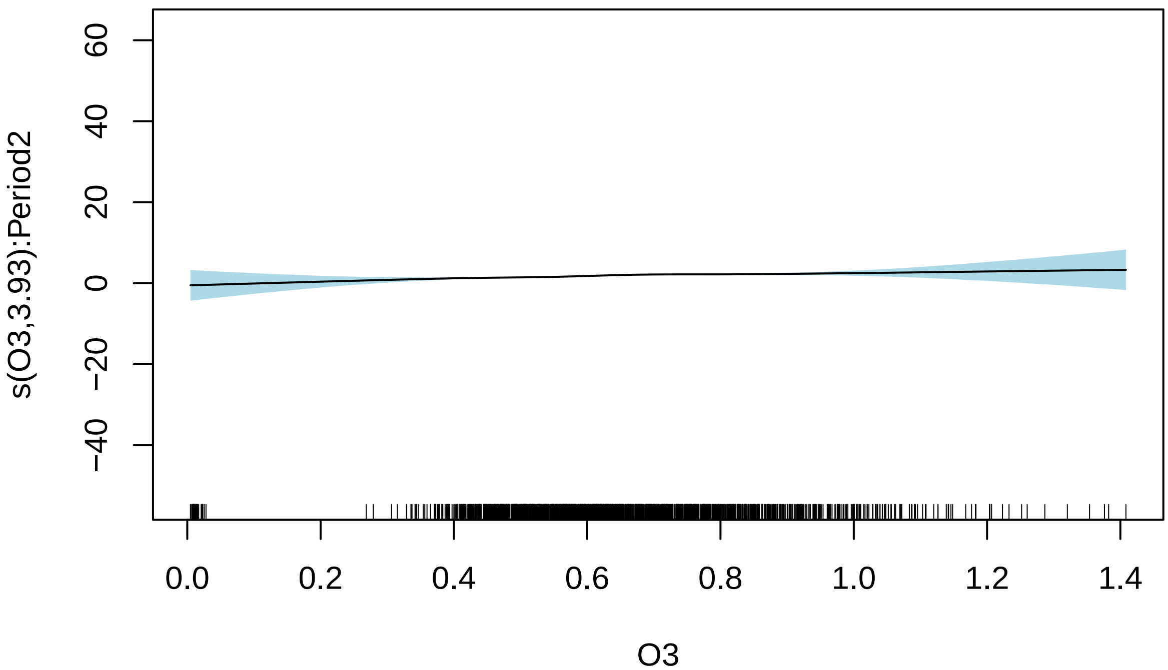

s(SO2,3.57):Period2
60
40
20
0
−20
−40
0.000
0.002
0.004
0.006
0.008
0.010
SO2

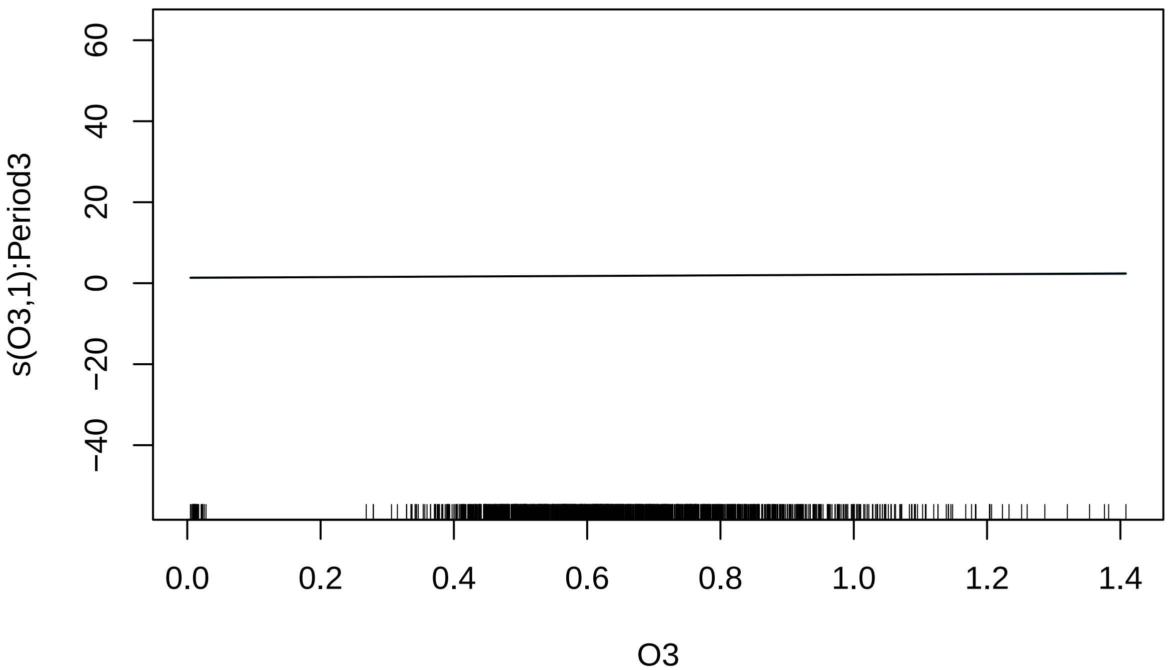


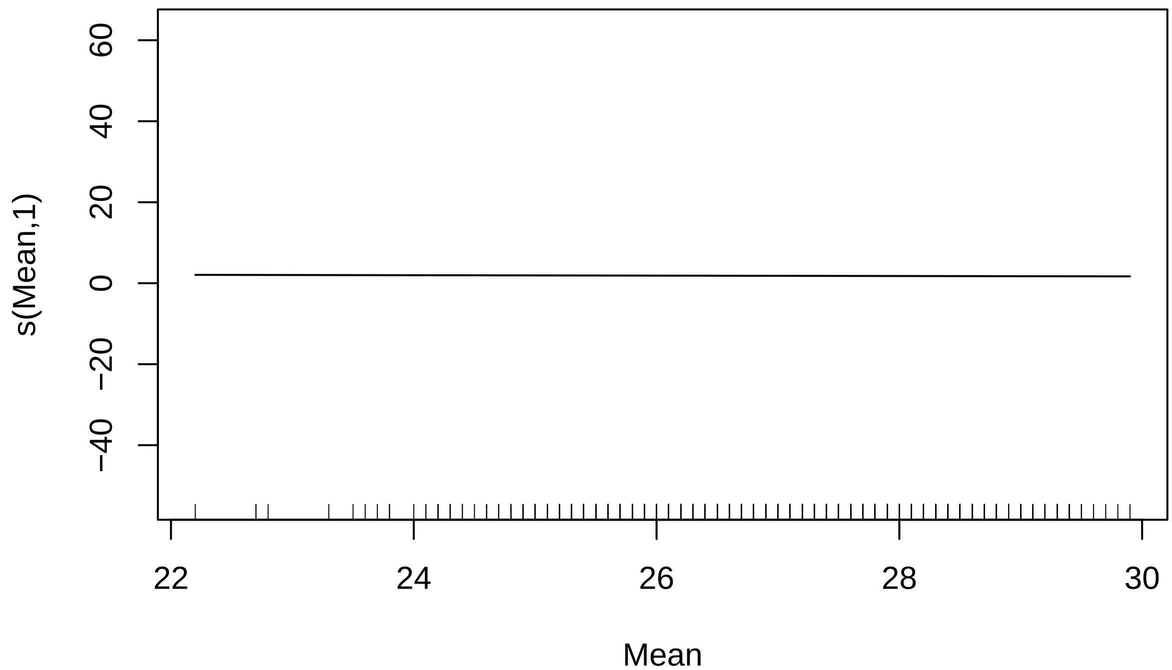
s(SO2,1.9):Period3
60
40
20
0
−20
−40
0.000
0.002
0.004
0.006
0.008
0.010
SO2



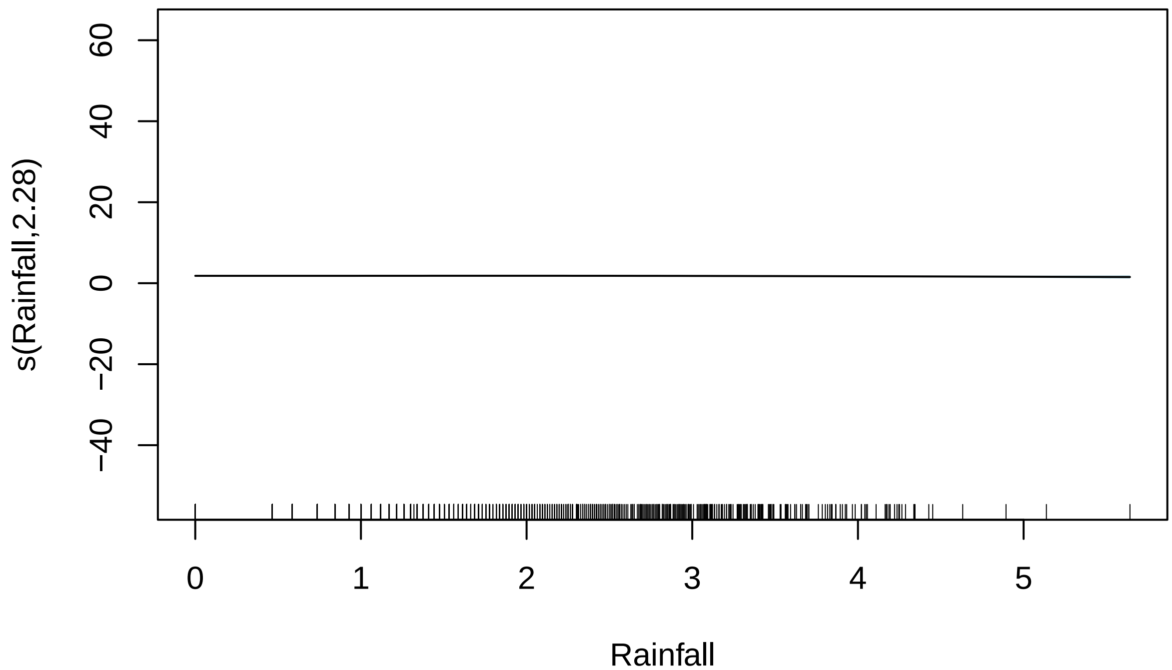
s(O3,3.48):Period1
60
40
20
0
−20
−40
0.0
0.2
0.4
0.6
0.8
1.0
1.2
1.4
O3

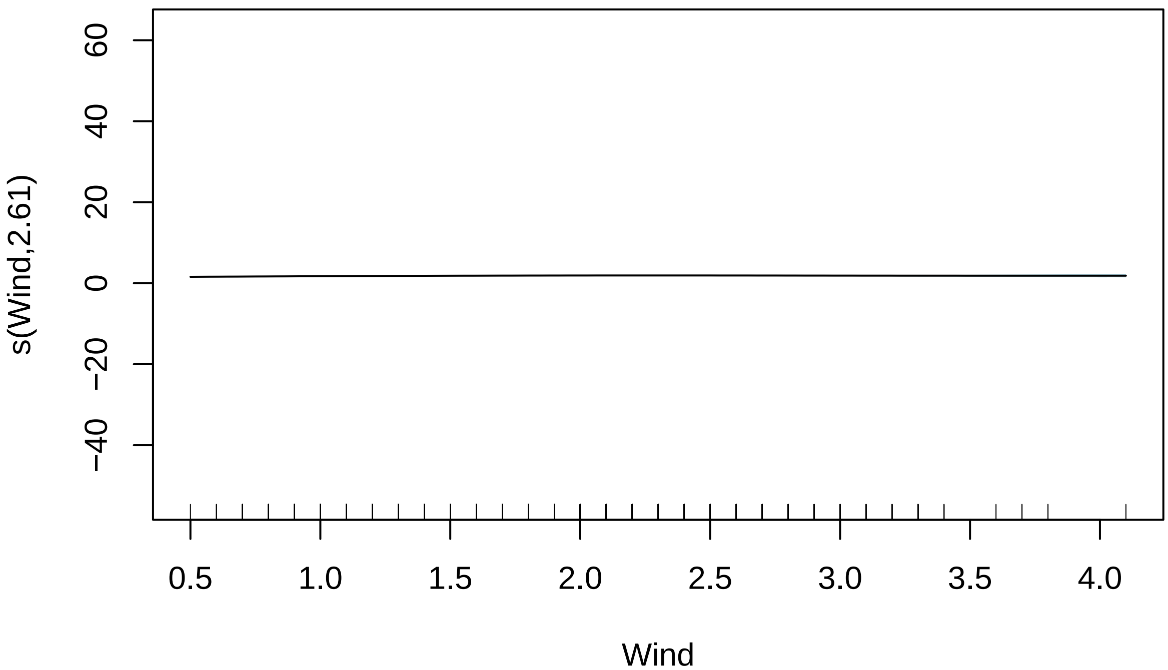

s(O3,3.93):Period2
60
40
20
0
−20
−40
0.0
0.2
0.4
0.6
0.8
1.0
1.2
1.4
O3

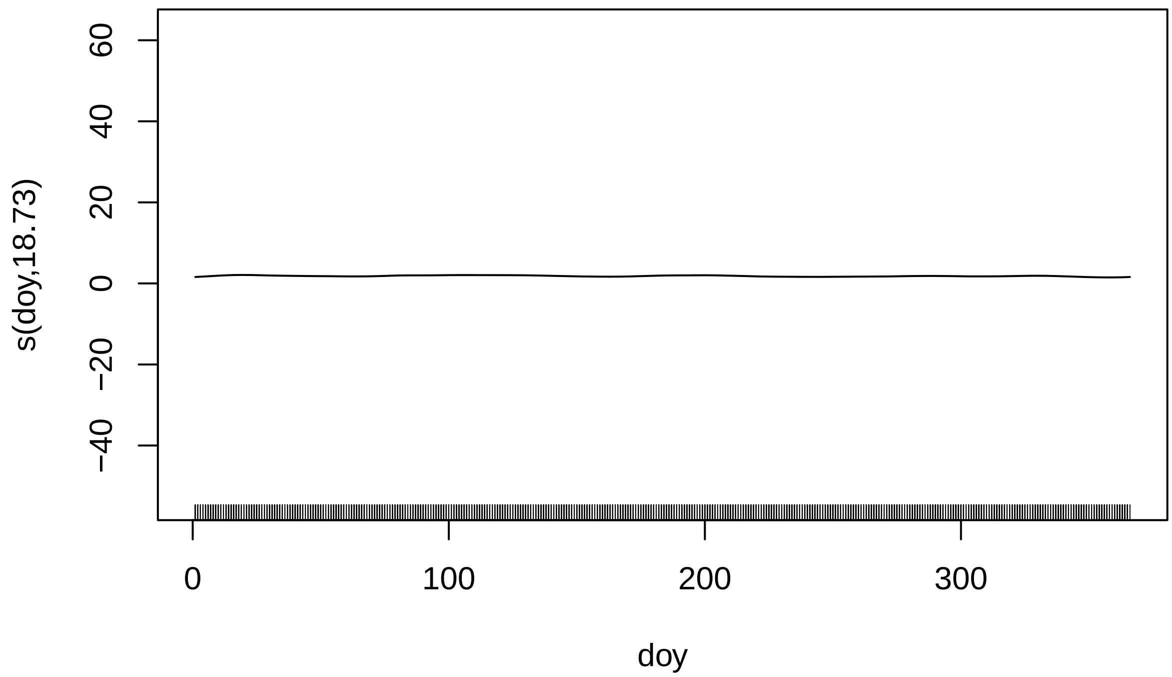

s(O3,1):Period3
60
40
20
0
−20
−40
0.0
0.2
0.4
0.6
0.8
1.0
1.2
1.4
O3

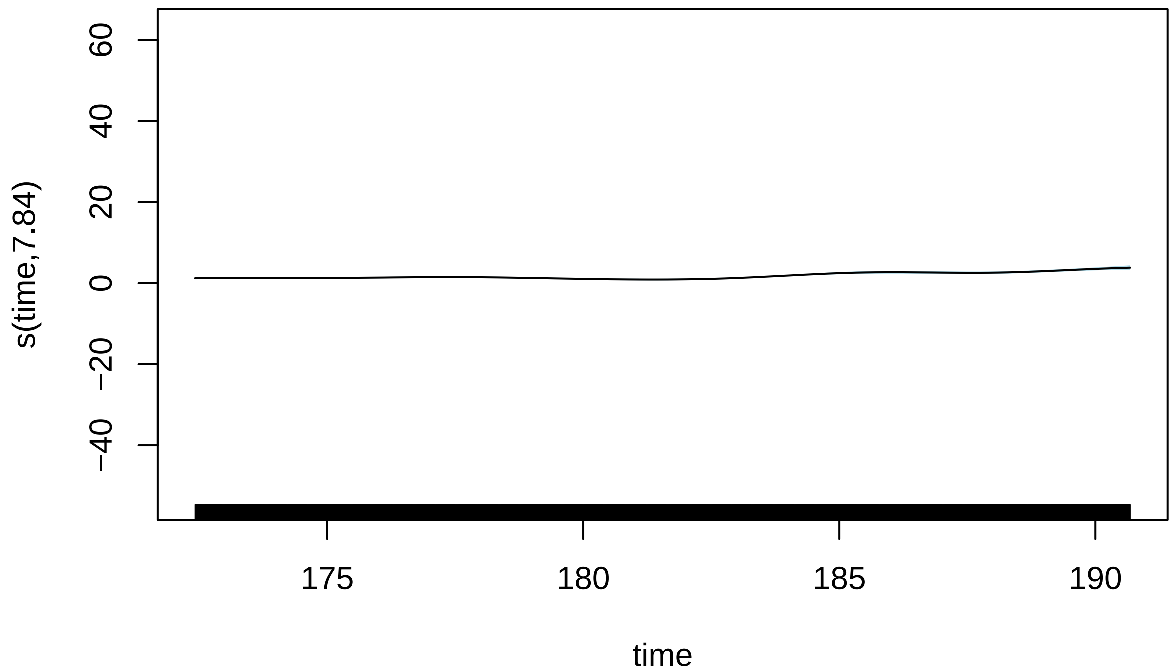


s(Mean,1)
60
40
20
0
−20
−40
22
24
26
28
30
Mean

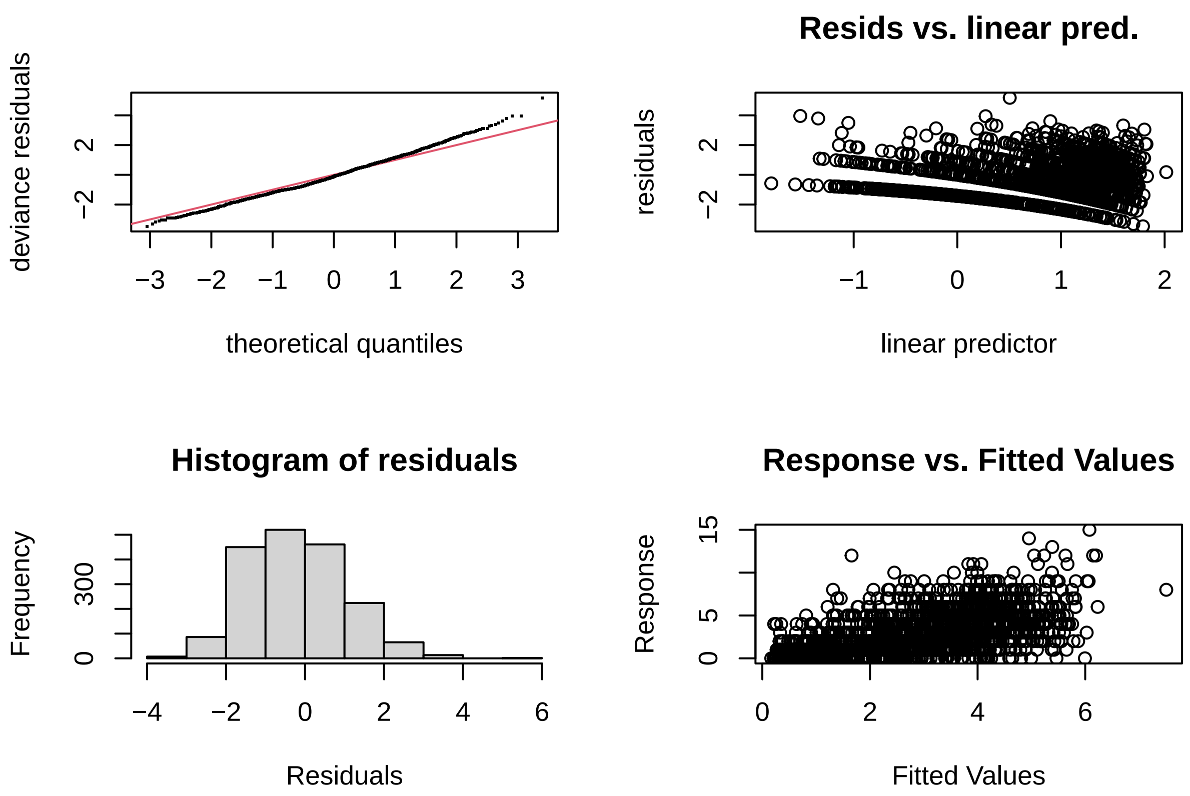

s(Rainfall,2.28)
60
40
20
0
−20
−40
0
1
2
3
4
5
Rainfall


s(Wind,2.61)
60
40
20
0
−20
−40
0.5
1.0
1.5
2.0
2.5
3.0
3.5
4.0
Wind


s(doy,18.73)
60
40
20
0
−20
−40
0
100
200
300
doy




```
gam.check(jb_LRTI_period)
```




```
##
## Method: REML   Optimizer: outer newton
## full convergence after 10 iterations.
## Gradient range [-0.0003575688,0.00026311]
## (score 3653.113 & scale 1).
## Hessian positive definite, eigenvalue range [3.130071e-05,2.913879].
## Model rank =  537 / 537
##
## Basis dimension (k) checking results. Low p-value (k-index<1) may
## indicate that k is too low, especially if edf is close to k'.
##
##                     k'    edf k-index p-value
## s(PM10):Period1   9.00   1.00    1.02    0.99
## s(PM10):Period2   9.00   1.01    1.02    0.98
## s(PM10):Period3   9.00   1.00    1.02    0.98
## s(NO2):Period1    9.00   1.00    0.97    0.44
## s(NO2):Period2    9.00   1.52    0.97    0.40
## s(NO2):Period3    9.00   1.00    0.97    0.41
## s(CO3):Period1    9.00   1.96    0.99    0.75
## s(CO3):Period2    9.00   3.03    0.99    0.71
## s(CO3):Period3    9.00   1.00    0.99    0.73
## s(SO2):Period1    9.00   1.65    0.91  <2e-16 ***
## s(SO2):Period2    9.00   3.57    0.91  <2e-16 ***
## s(SO2):Period3    9.00   1.90    0.91    0.01 **
## s(O3):Period1     9.00   3.48    0.94    0.07 .
## s(O3):Period2     9.00   3.93    0.94    0.08 .
```

```
## s(O3):Period3     9.00   1.00    0.94    0.12
## s(Mean)           9.00   1.00    0.96    0.21
## s(Rainfall)       9.00   2.28    0.93    0.06 .
## s(Wind)           9.00   2.61    0.90  <2e-16 ***
## s(doy)          363.00  18.73    0.95    0.14
## s(time)           9.00   7.84    0.84  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

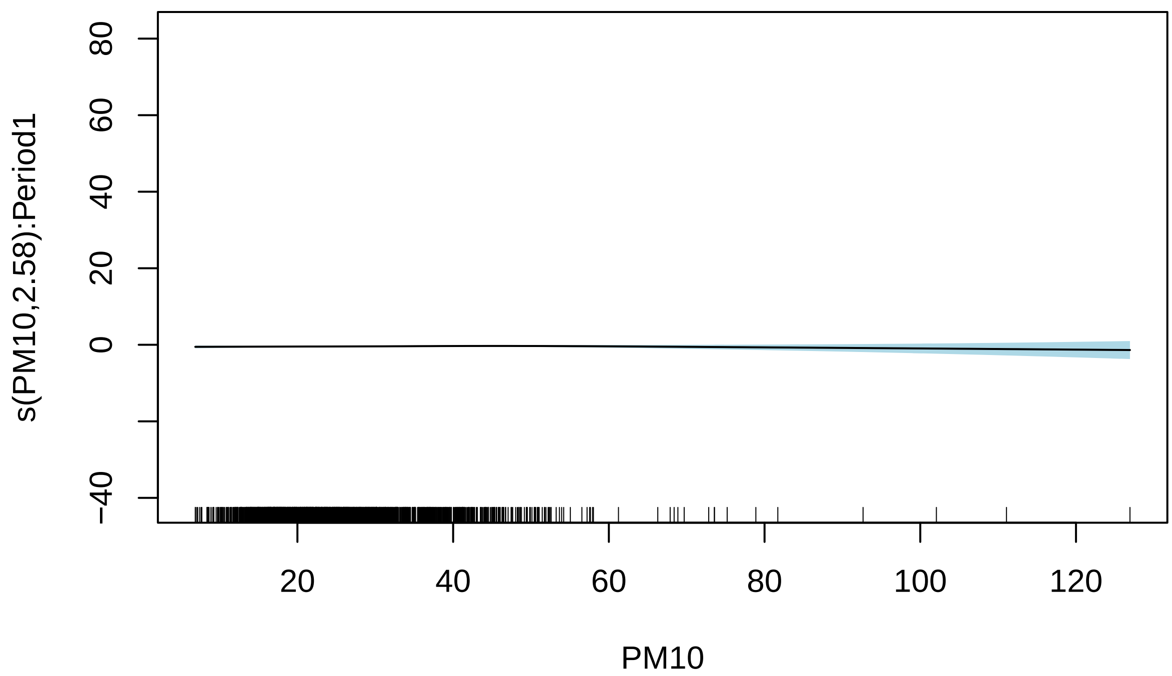
```
jb_sx_period <- gam(Symptoms_Signs ~ s(PM10, by=Period) + s(NO2, by=Period) + s(CO3, by=Period) + s(SO2
summary(jb_sx_period)
```

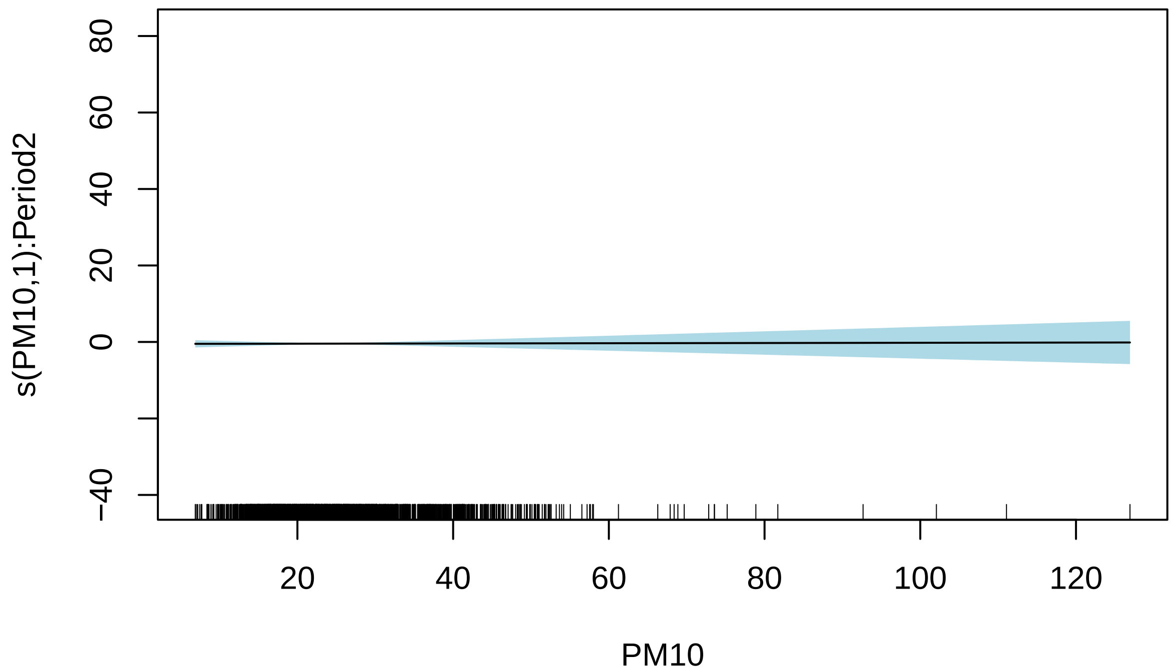
**JB SX by period**

```
##
## Family: poisson
## Link function: log
##
## Formula:
## Symptoms_Signs ~ s(PM10, by = Period) + s(NO2, by = Period) +
##     s(CO3, by = Period) + s(SO2, by = Period) + s(O3, by = Period) +
##     s(Mean) + s(Rainfall) + s(Wind) + s(doy, bs = "cc", k = 365) +
##     s(time, bs = "bs") + Period
##
## Parametric coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -0.4332     0.1370  -3.161  0.00157 **
## Period2      -2.0724     0.4227  -4.902 9.47e-07 ***
## Period3       0.4464     0.8149   0.548  0.58382
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Approximate significance of smooth terms:
##                    edf  Ref.df Chi.sq  p-value
## s(PM10):Period1 2.581   3.345  3.016  0.40797
## s(PM10):Period2 1.001   1.002  0.011  0.91830
## s(PM10):Period3 1.754   2.133  7.301  0.04419 *
## s(NO2):Period1  2.471   3.139  9.243  0.03025 *
## s(NO2):Period2  1.559   1.938  0.474  0.79560
## s(NO2):Period3  1.359   1.630  1.128  0.60332
## s(CO3):Period1  2.422   3.044  5.033  0.17217
## s(CO3):Period2  1.001   1.001  0.162  0.68784
## s(CO3):Period3  1.982   2.612  0.522  0.91835
## s(SO2):Period1  1.001   1.001  1.374  0.24153
## s(SO2):Period2  1.000   1.001  0.018  0.89475
## s(SO2):Period3  1.000   1.001  0.085  0.77088
## s(O3):Period1   3.168   4.065  5.013  0.28622
## s(O3):Period2   2.335   2.958  3.333  0.29588
## s(O3):Period3   1.000   1.001  0.792  0.37383
## s(Mean)         2.088   2.679  3.722  0.29888
## s(Rainfall)     2.660   3.301 12.657  0.00748 **
```

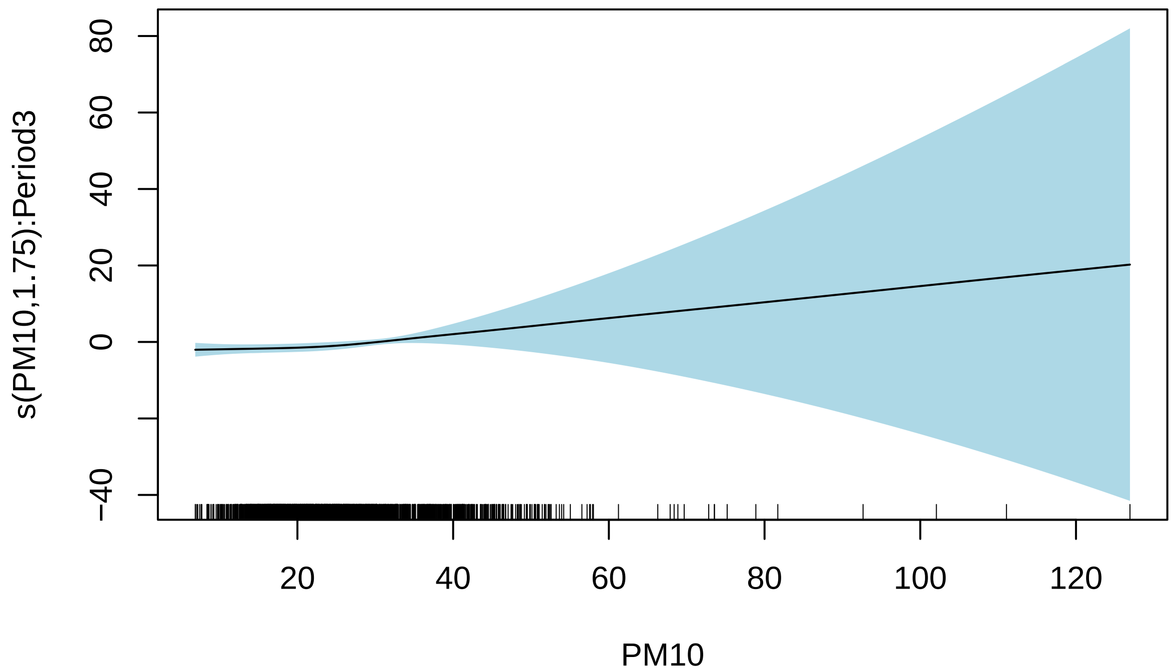
```
## s(Wind)         1.541   1.924  1.002  0.62640
## s(doy)          6.864 363.000 51.163  < 2e-16 ***
## s(time)         3.761   4.563 29.251 9.69e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-sq.(adj) =  0.211   Deviance explained = 30.5%
## -REML = 1631.3  Scale est. = 1         n = 1827
```

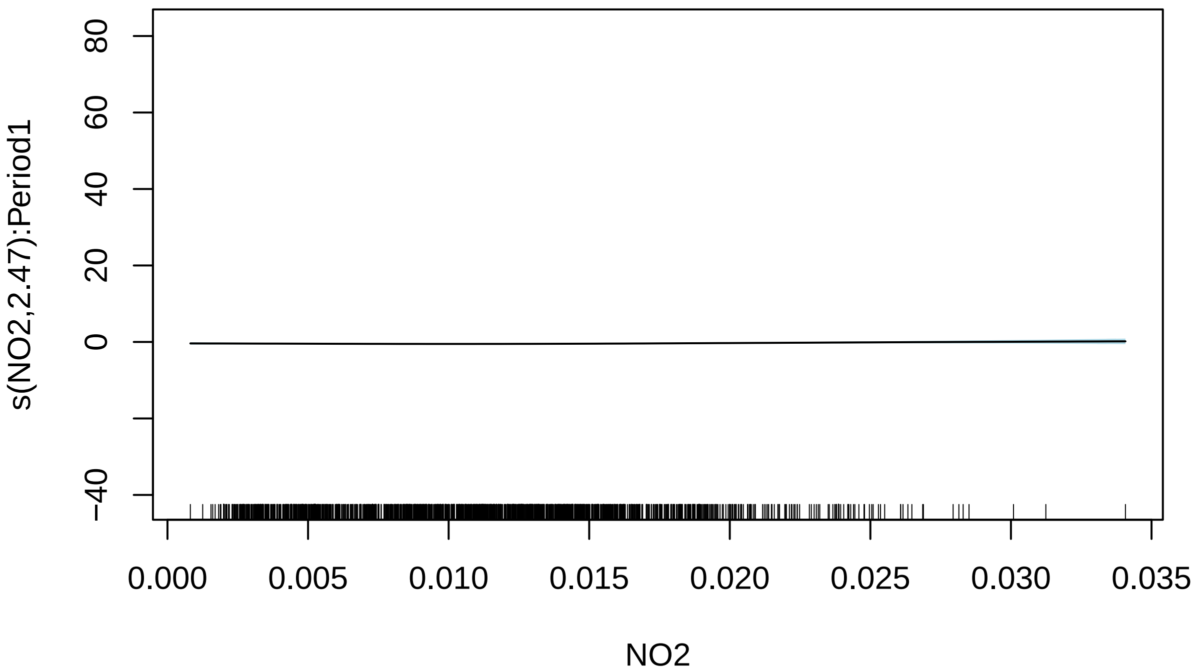
```r
plot.gam(jb_sx_period, shade = TRUE, shade.col = "lightblue", shift = coef(jb_sx_period)[1],  seWithMean
```

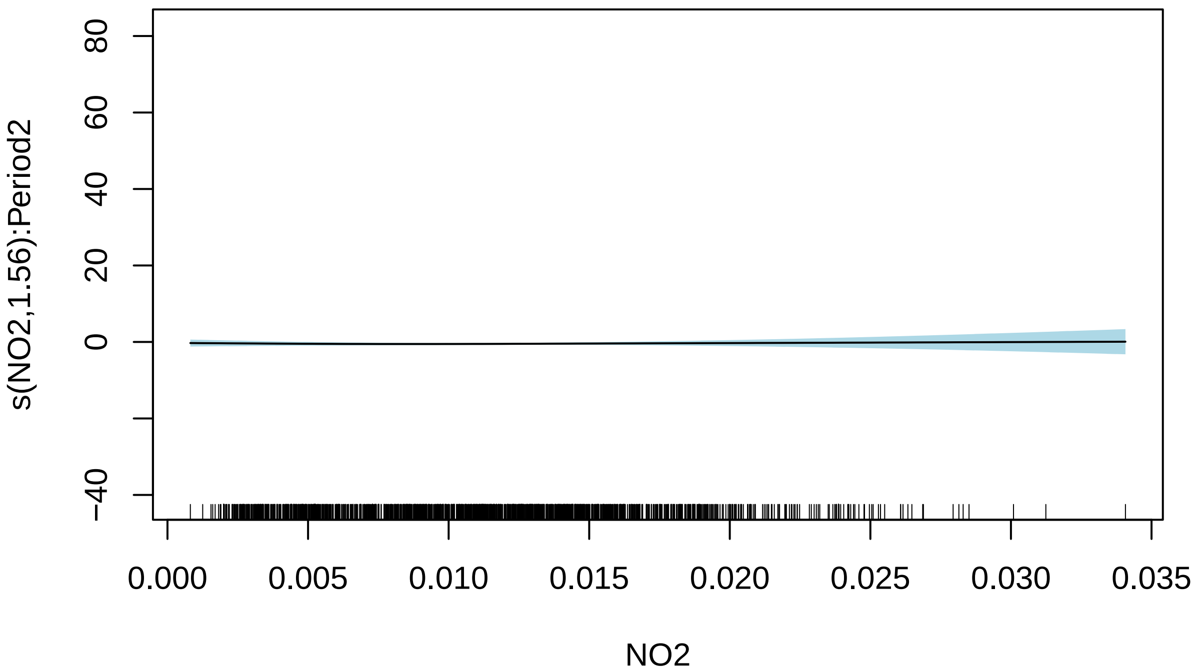




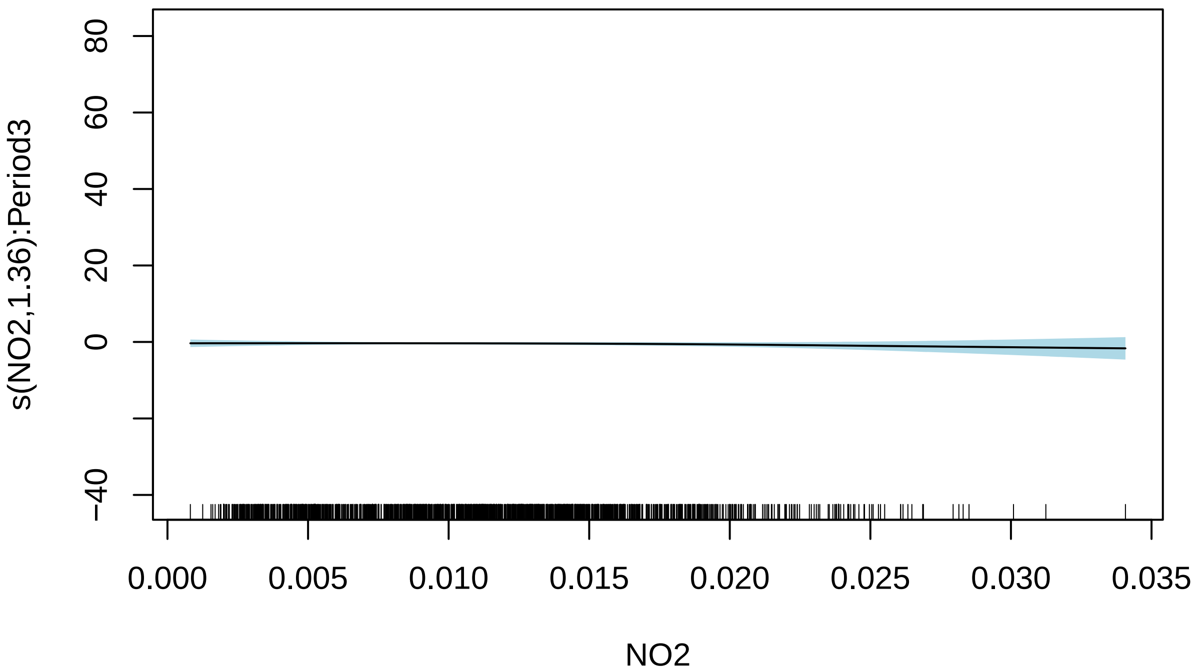
s(PM10,1):Period2
80
60
40
20
0
−40
20
40
60
80
100
120
PM10

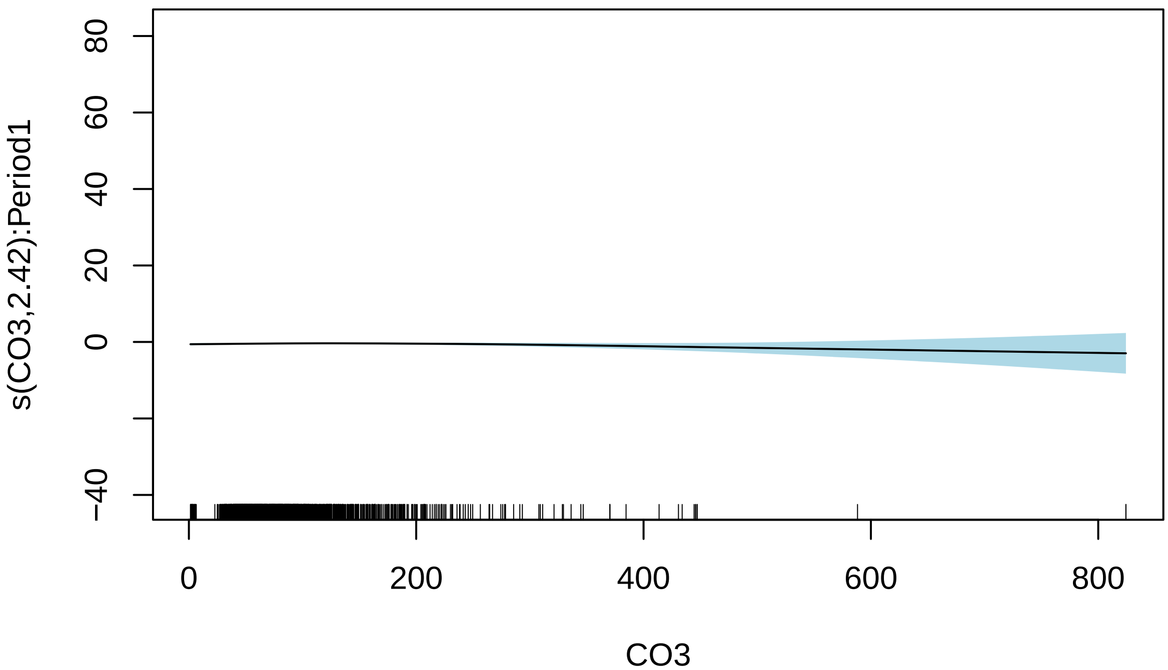

s(PM10,1.75):Period3
80
60
40
20
0
−40
20
40
60
80
100
120
PM10

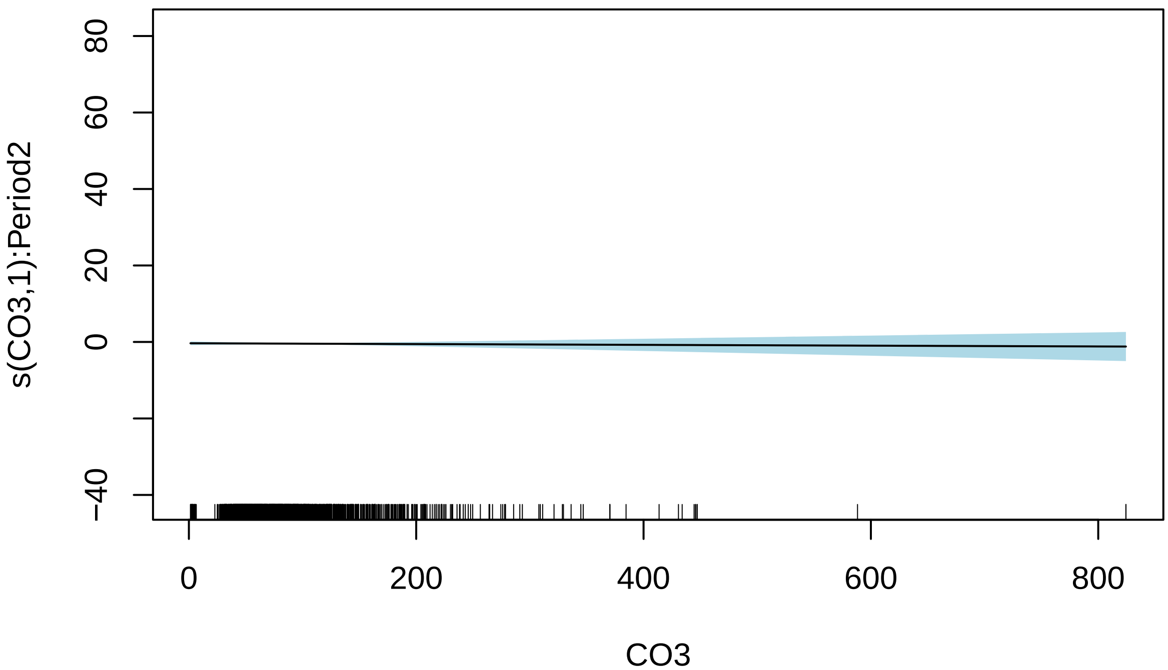

s(NO2,2.47):Period1
80
60
40
20
0
−40
0.000
0.005
0.010
0.015
0.020
0.025
0.030
0.035
NO2



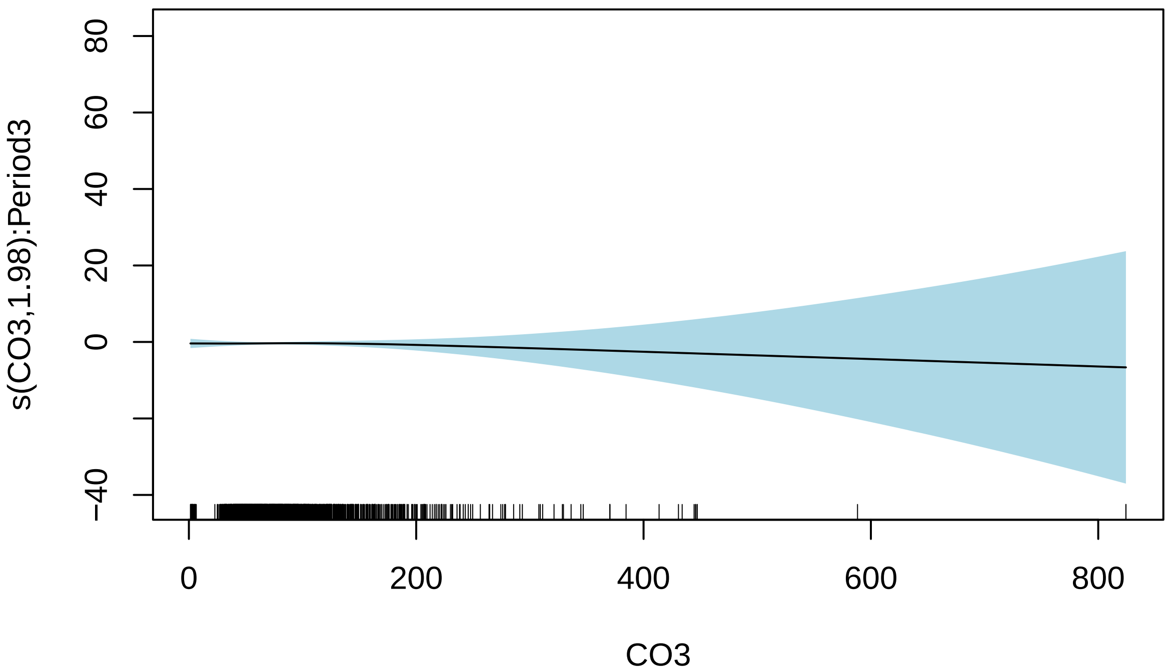
s(NO2,1.56):Period2
80
60
40
20
0
−40
0.000
0.005
0.010
0.015
0.020
0.025
0.030
0.035
NO2


s(NO2,1.36):Period3
80
60
40
20
0
−40
0.000
0.005
0.010
0.015
0.020
0.025
0.030
0.035
NO2

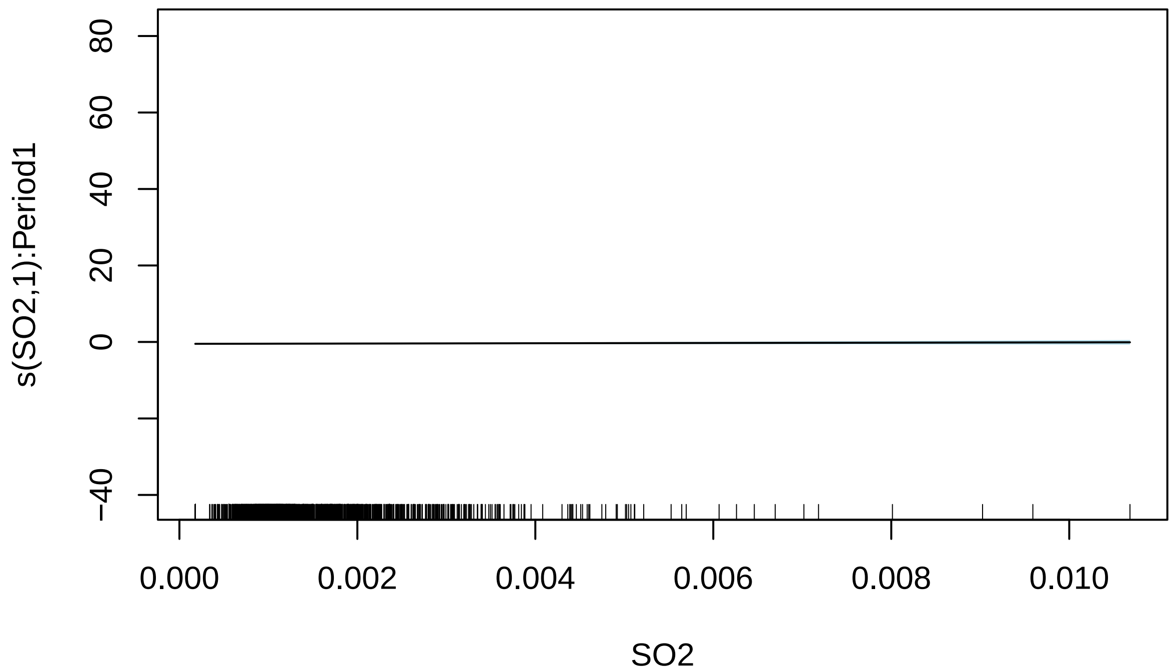

s(CO3,2.42):Period1
80
60
40
20
0
−40
0
200
400
600
800
CO3

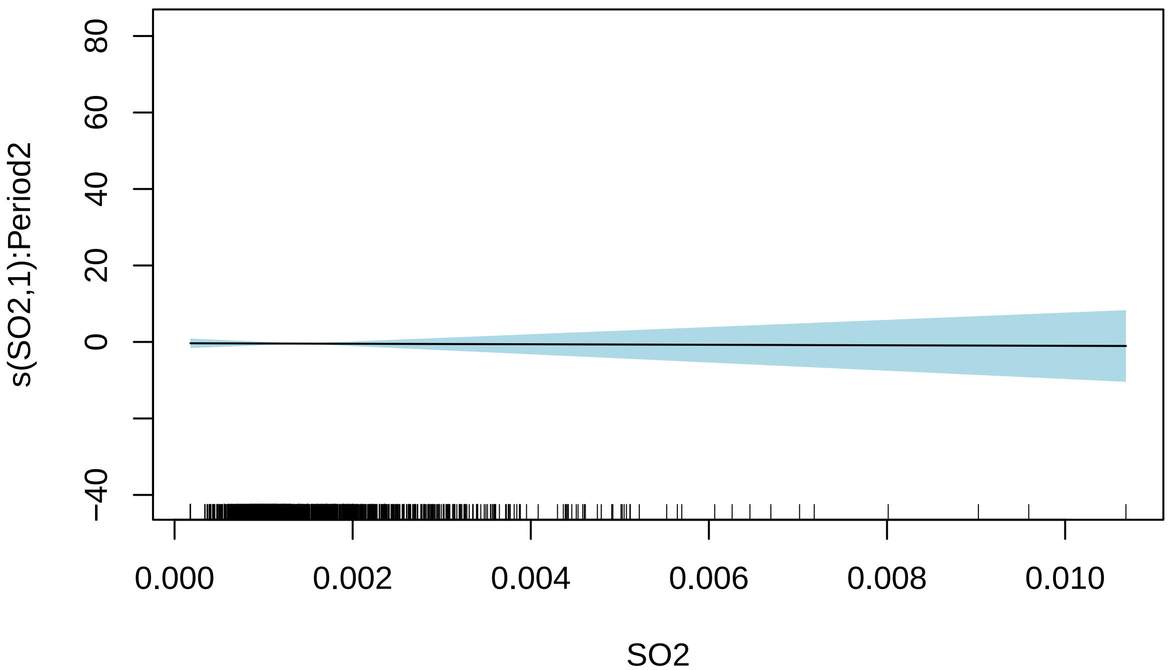


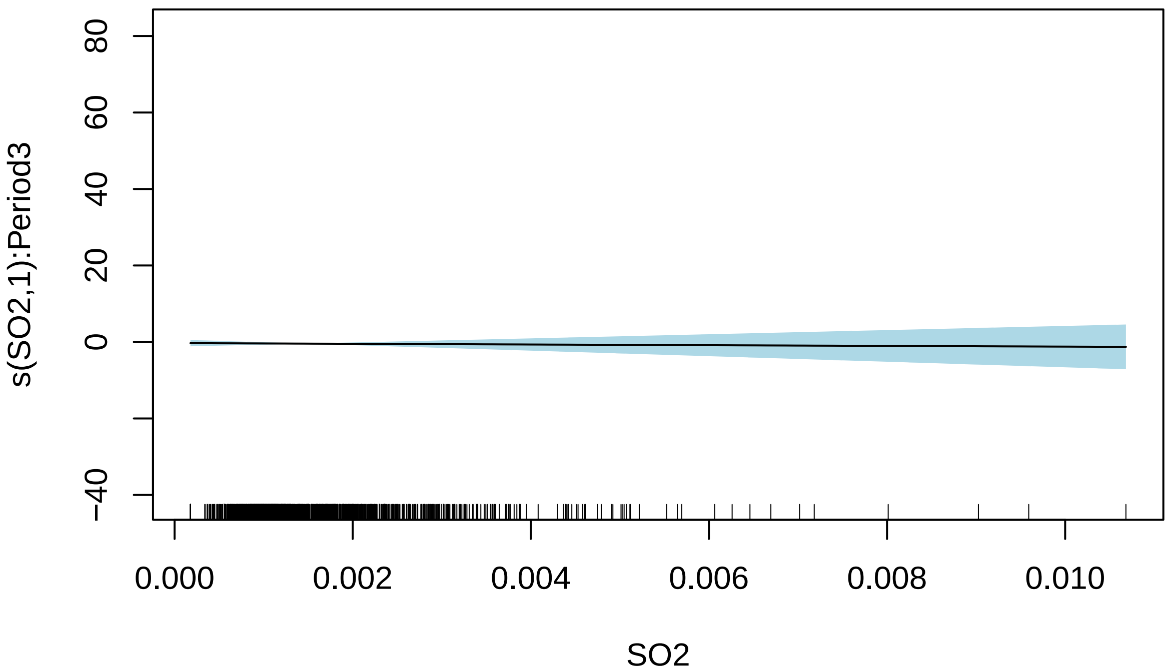
s(CO3,1):Period2
80
60
40
20
0
−40
0
200
400
600
800
CO3



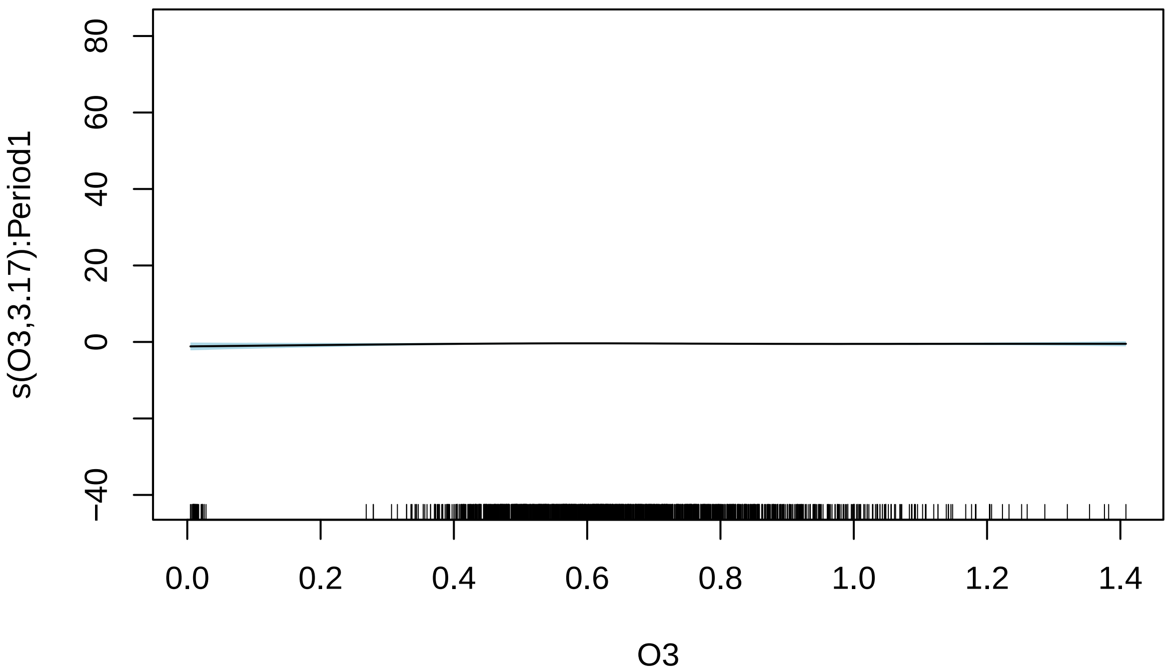
s(CO3,1.98):Period3
80
60
40
20
0
−40
0
200
400
600
800
CO3

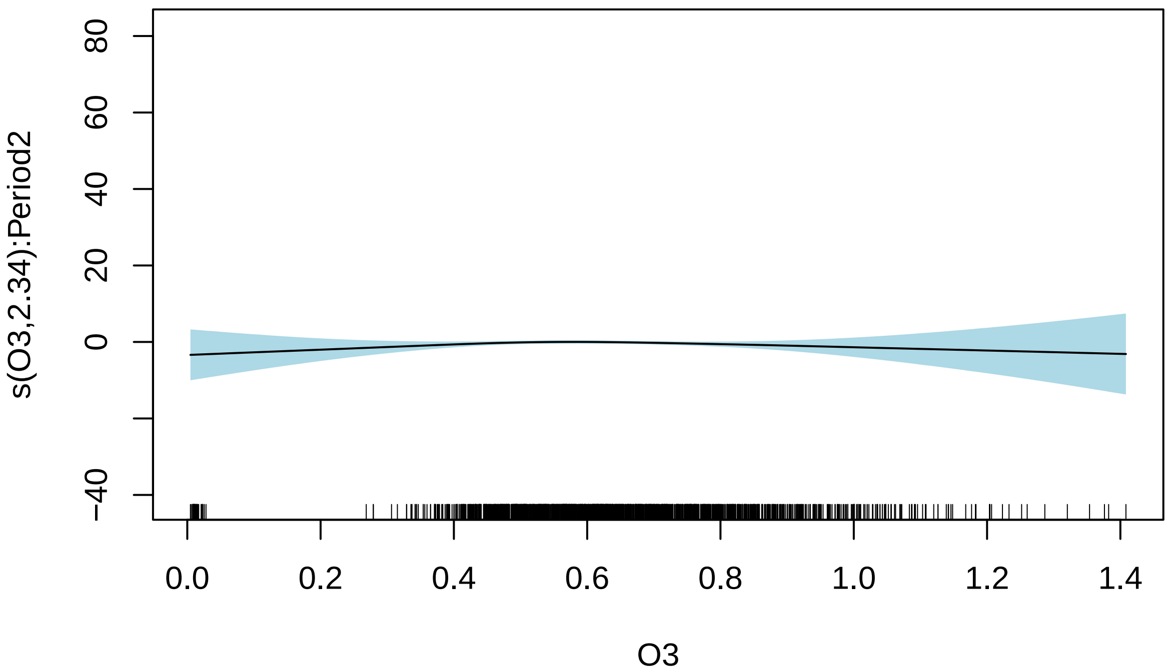

s(SO2,1):Period1
80
60
40
20
0
−40
0.000
0.002
0.004
0.006
0.008
0.010
SO2

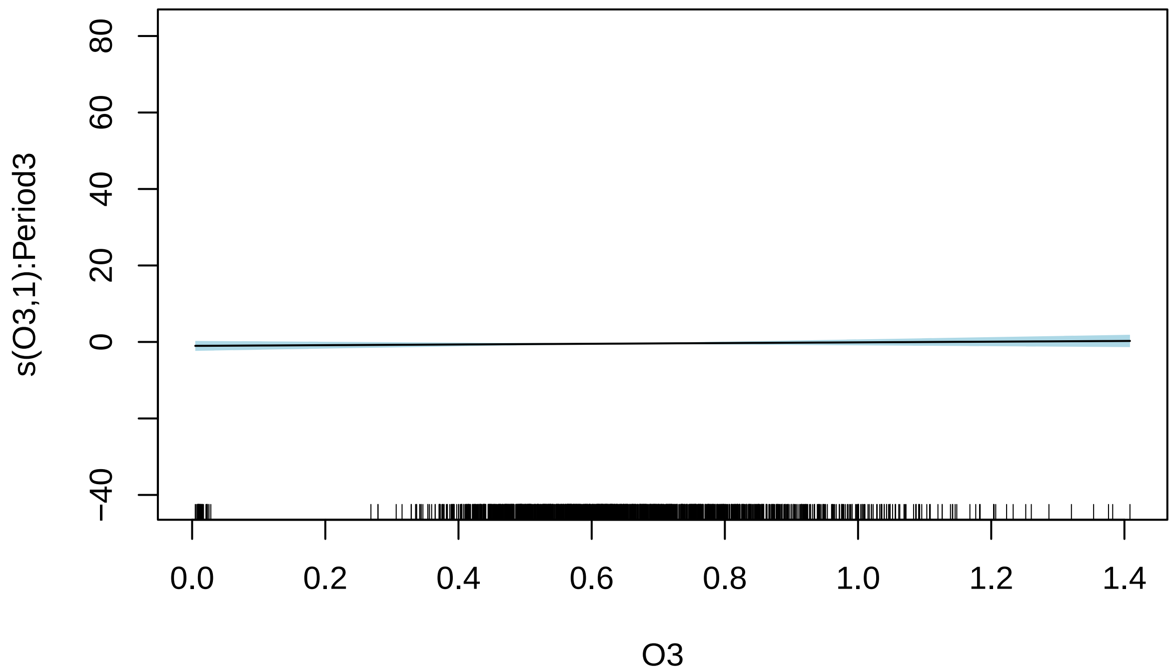

s(SO2,1):Period2
80
60
40
20
0
−40
0.000
0.002
0.004
0.006
0.008
0.010
SO2

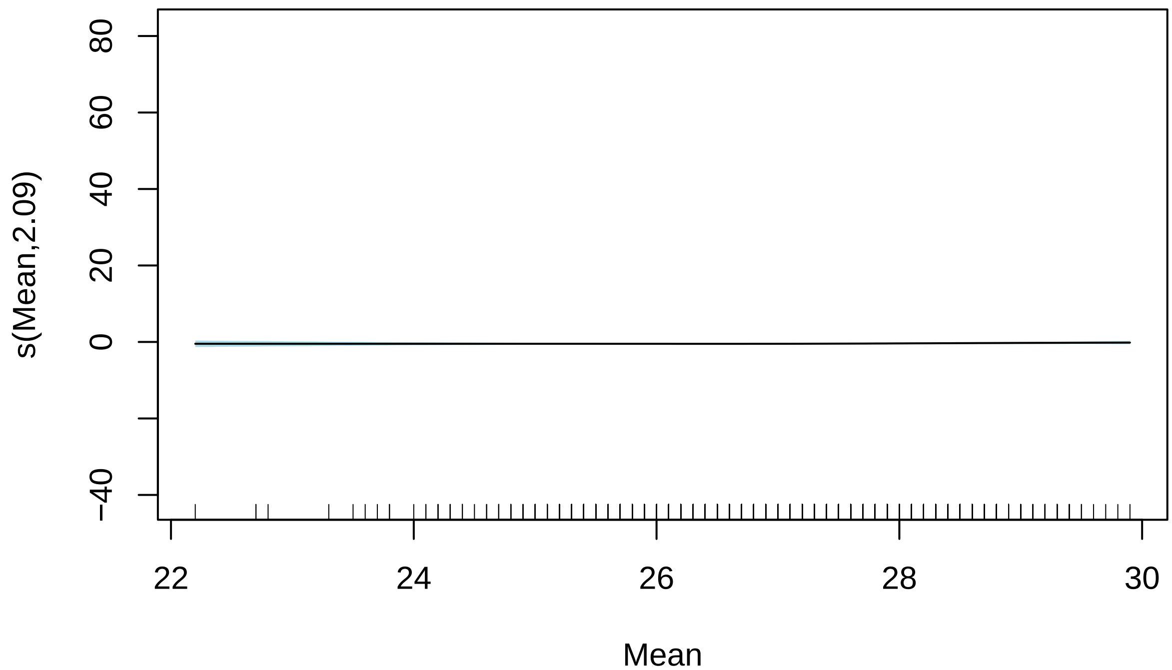


80
60
40
20
0
−40
s(SO2,1):Period3
0.000
0.002
0.004
0.006
0.008
0.010
SO2

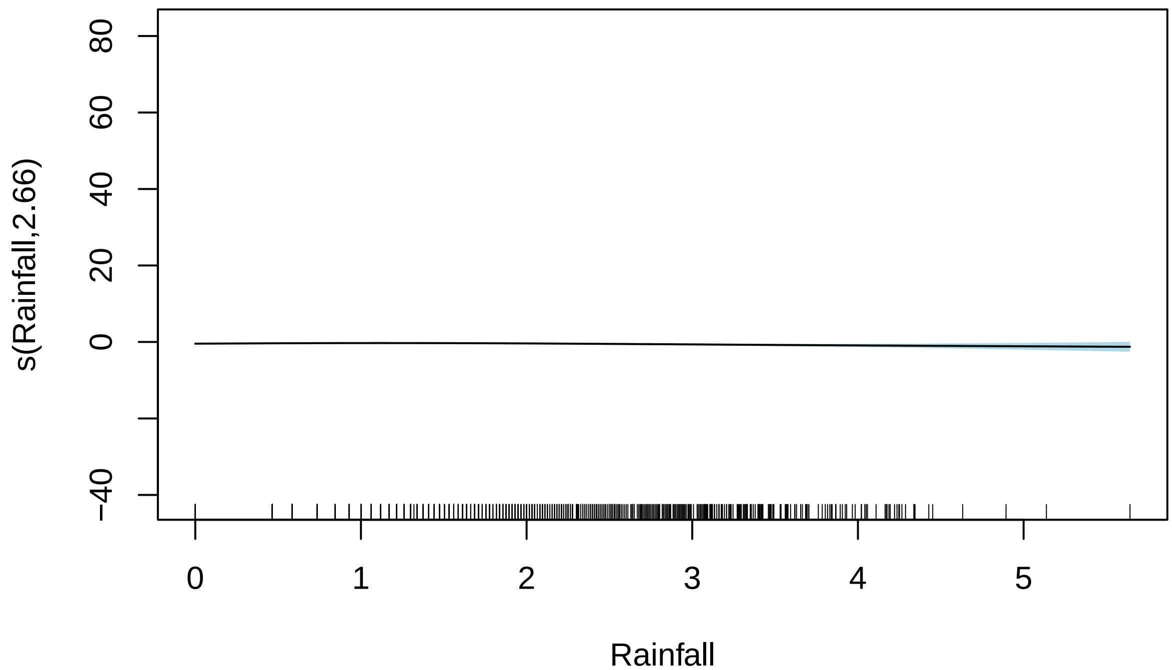

s(O3,3.17):Period1
80
60
40
20
0
−40
0.0
0.2
0.4
0.6
0.8
1.0
1.2
1.4
O3

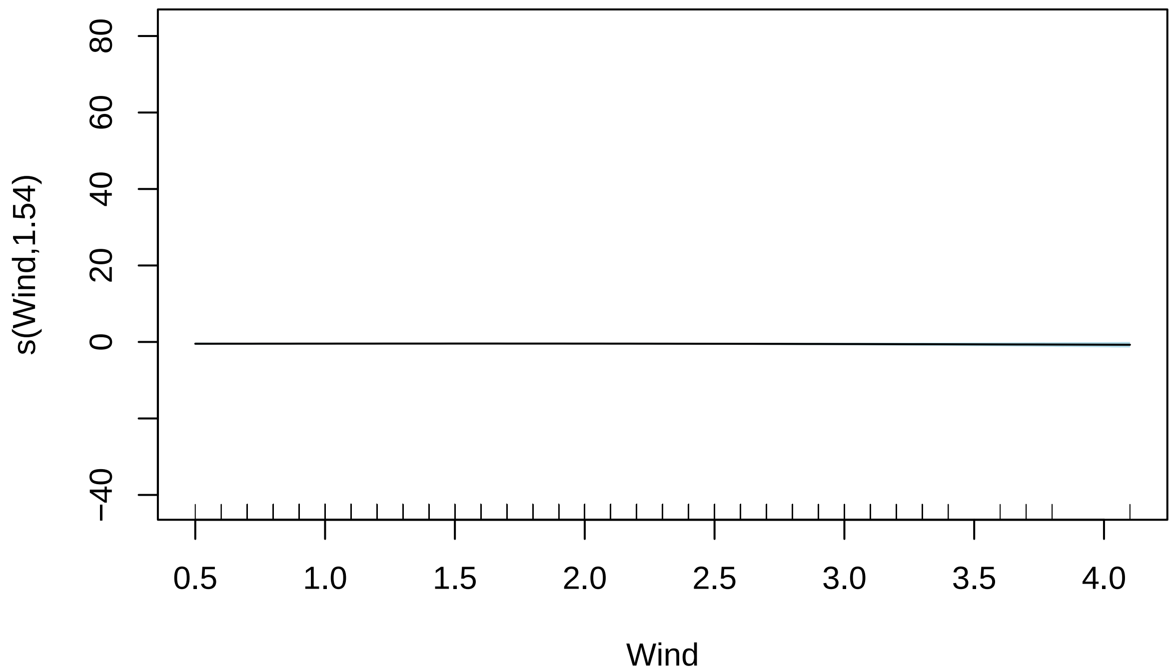

s(O3,2.34):Period2
80
60
40
20
0
−40
0.0
0.2
0.4
0.6
0.8
1.0
1.2
1.4
O3

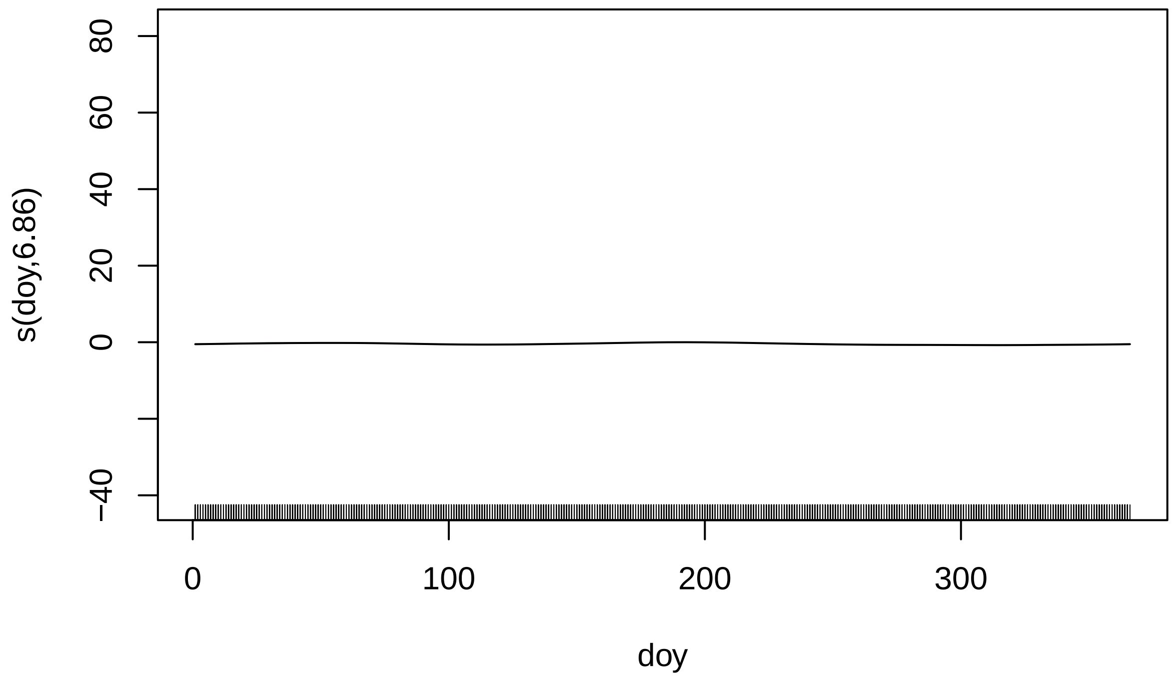

s(O3,1):Period3
80
60
40
20
0
−40
0.0
0.2
0.4
0.6
0.8
1.0
1.2
1.4
O3



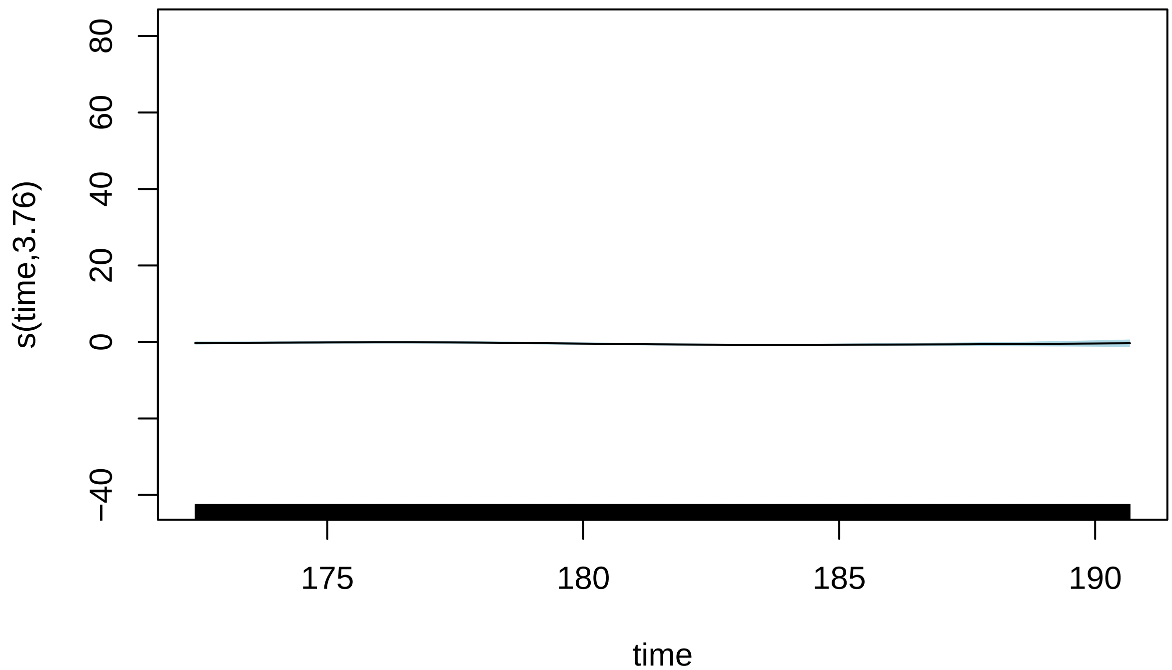
s(Mean,2.09)
80
60
40
20
0
−40
22
24
26
28
30
Mean


s(Rainfall,2.66)
80
60
40
20
0
−40
0
1
2
3
4
5
Rainfall

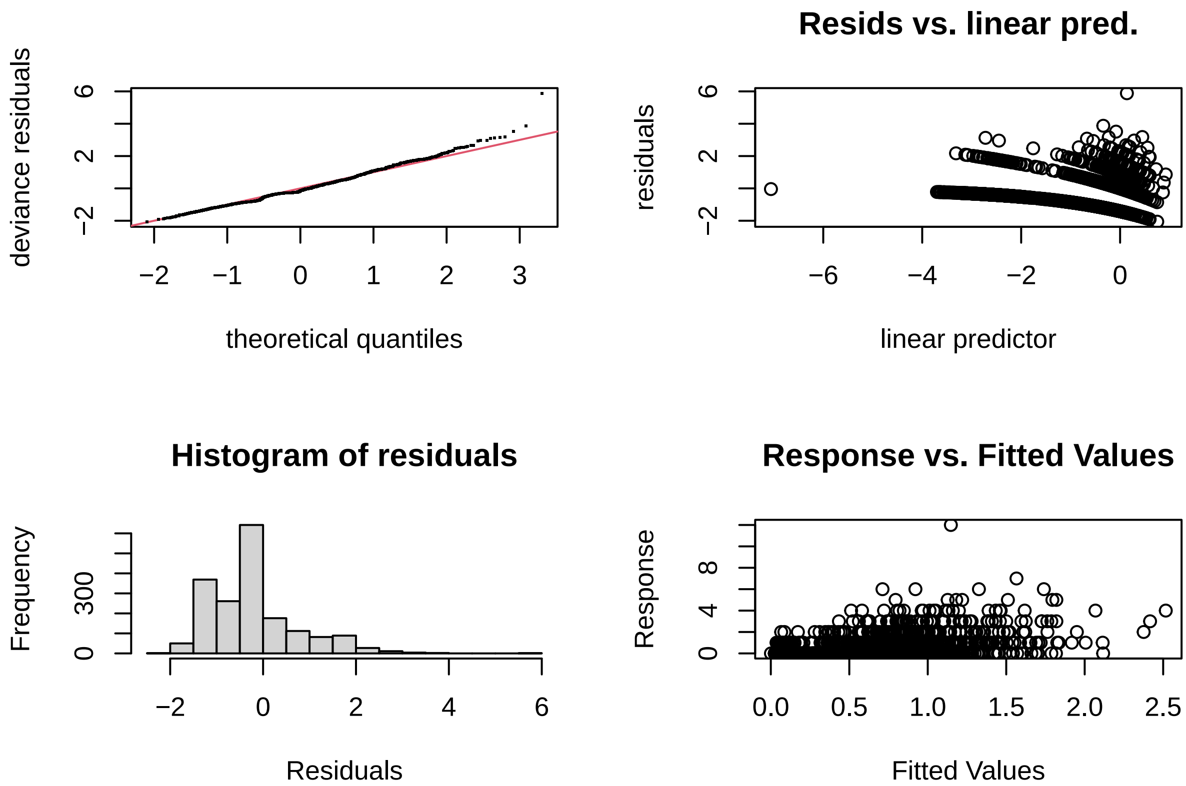

s(Wind,1.54)
80
60
40
20
0
−40
0.5
1.0
1.5
2.0
2.5
3.0
3.5
4.0
Wind



s(doy,6.86)
80
60
40
20
0
−40
0
100
200
300
doy




```
gam.check(jb_sx_period)
```




```
##
## Method: REML   Optimizer: outer newton
## full convergence after 8 iterations.
## Gradient range [-0.0002195525,3.626507e-05]
## (score 1631.318 & scale 1).
## Hessian positive definite, eigenvalue range [0.0001377561,1.982931].
## Model rank =  537 / 537
##
## Basis dimension (k) checking results. Low p-value (k-index<1) may
## indicate that k is too low, especially if edf is close to k'.
##
##                       k'    edf k-index p-value
## s(PM10):Period1    9.00   2.58    0.96    0.90
## s(PM10):Period2    9.00   1.00    0.96    0.92
## s(PM10):Period3    9.00   1.75    0.96    0.85
## s(NO2):Period1     9.00   2.47    0.92    0.28
## s(NO2):Period2     9.00   1.56    0.92    0.17
## s(NO2):Period3     9.00   1.36    0.92    0.25
## s(CO3):Period1     9.00   2.42    0.95    0.88
## s(CO3):Period2     9.00   1.00    0.95    0.84
## s(CO3):Period3     9.00   1.98    0.95    0.84
## s(SO2):Period1     9.00   1.00    0.95    0.71
## s(SO2):Period2     9.00   1.00    0.95    0.71
## s(SO2):Period3     9.00   1.00    0.95    0.77
## s(O3):Period1      9.00   3.17    0.91    0.21
## s(O3):Period2      9.00   2.34    0.91    0.20
```

```
## s(O3):Period3     9.00   1.00    0.91    0.20
## s(Mean)           9.00   2.09    0.96    0.90
## s(Rainfall)       9.00   2.66    0.94    0.64
## s(Wind)           9.00   1.54    0.91    0.19
## s(doy)          363.00   6.86    0.95    0.80
## s(time)           9.00   3.76    0.91    0.15
```