intra_day_model5

2023-05-01

```
source("notebooks/intra_day/Functions.R")
df <- read.csv("data/merged_interpolated.csv")
names(df)[c(1, 2, 10)] <- c('dt', 'Temp', 'Demand')

# create timezone adjusted standard and daylight savings time
ACDT <- as.POSIXct(paste(df$dt, "+1000"), format="%F %T %z", tz="Australia/South")
ACST <- ACDT - ((as.POSIXlt(ACDT)$isdst > 0)*(60*60))
ACST.lt <- as.POSIXlt(ACST)
ACDT.lt <- as.POSIXlt(ACDT)

# time normalised to [0, 1) via proportion of minutes per day
df$Time <- (ACST.lt$hour*60 + ACST.lt$min) / (60*24)
df$DSTTime <- df$Time + ((ACDT.lt$isdst!=0) / 24)

df$dt <- as.POSIXct(df$dt, format="%F %T")

# choose year 2022
df <- df[ACST.lt$year == 122,]
df$Year <- ((1:nrow(df) - 1) %% (365 * 48)) / (365 * 48)
df$WtdTemp <- wtdtemp(df$DSTTime, df$Temp)

# extract weekdays
df.week <- df[ACDT.lt$wday %in% 1:5,]

## Split into regression data and out of sample test data.
train <- ((0*288)+1):(250*288)
fitdata <- df.week[train,]
#testdata <- df.week[-train,]

head(fitdata)
```

```
##                       dt  Temp cloud8 windk wdir humid rainmm radkjm2 pv_est
## 67039 2022-01-01 00:30:00 25.20    5.5   6.0  230    38      0       0      0
## 67040 2022-01-01 01:00:00 24.10    6.0   5.0  270    44      0       0      0
## 67041 2022-01-01 01:30:00 24.40    5.5   5.5  250    44      0       0      0
## 67042 2022-01-01 02:00:00 24.70    5.0   6.0  230    44      0       0      0
## 67043 2022-01-01 02:30:00 24.35    5.0   5.5  285    44      0       0      0
## 67044 2022-01-01 03:00:00 24.00    5.0   5.0  340    44      0       0      0
##       Demand total_load       Time    DSTTime          Year  WtdTemp
## 67039   1769   1702.227 0.00000000 0.04166667 0.000000e+00 21.52151
## 67040   1732   1693.678 0.02083333 0.06250000 5.707763e-05 21.09332
## 67041   1626   1595.355 0.04166667 0.08333333 1.141553e-04 21.07331
## 67042   1548   1521.352 0.06250000 0.10416667 1.712329e-04 21.05881
## 67043   1526   1495.105 0.08333333 0.12500000 2.283105e-04 20.91967
## 67044   1471   1449.107 0.10416667 0.14583333 2.853881e-04 20.81153
```

Fit and summarise Model 5.

```
library(mgcv)
```

```
## Loading required package: nlme
```

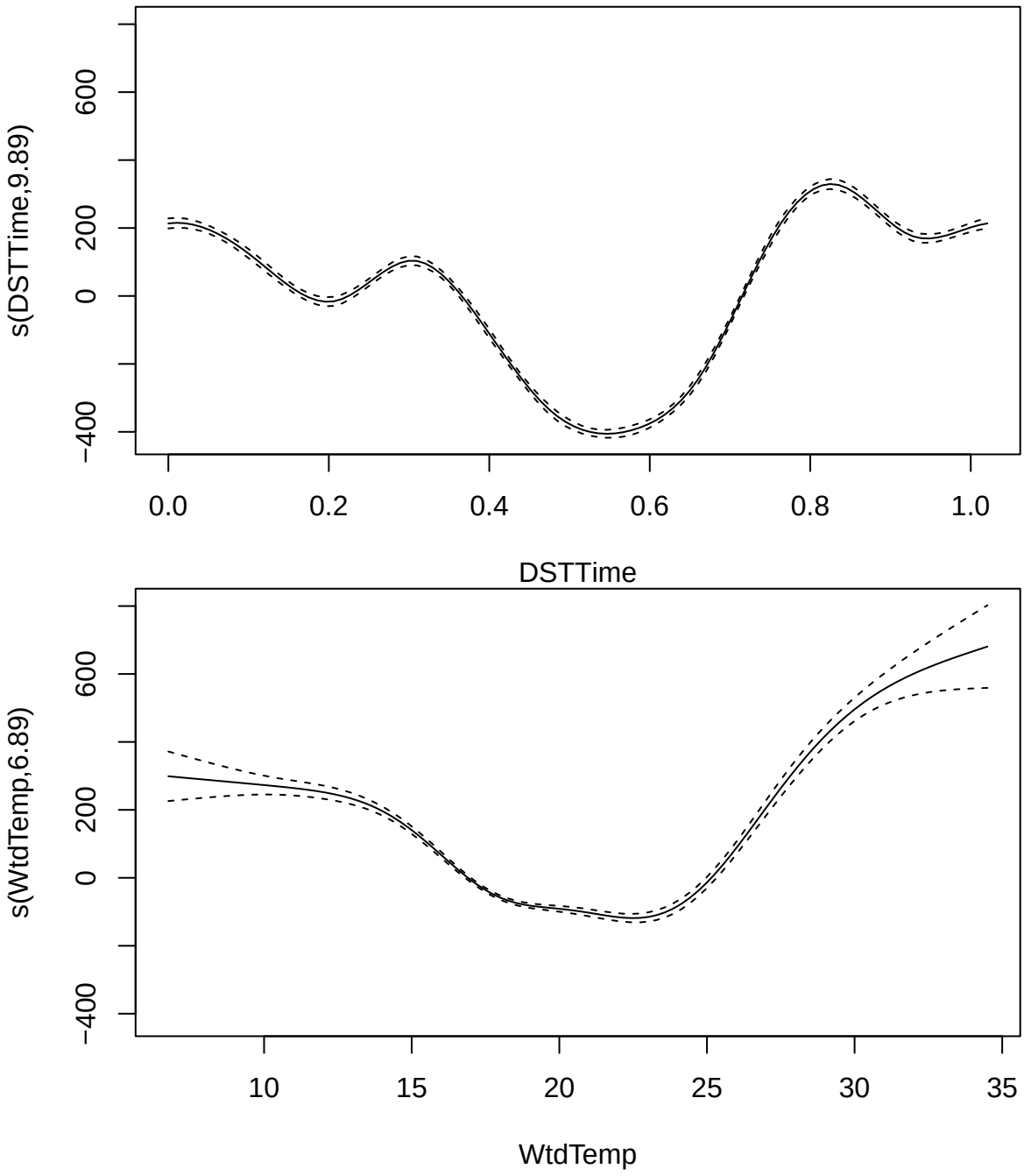
```
## This is mgcv 1.8-42. For overview type 'help("mgcv-package")'.
```

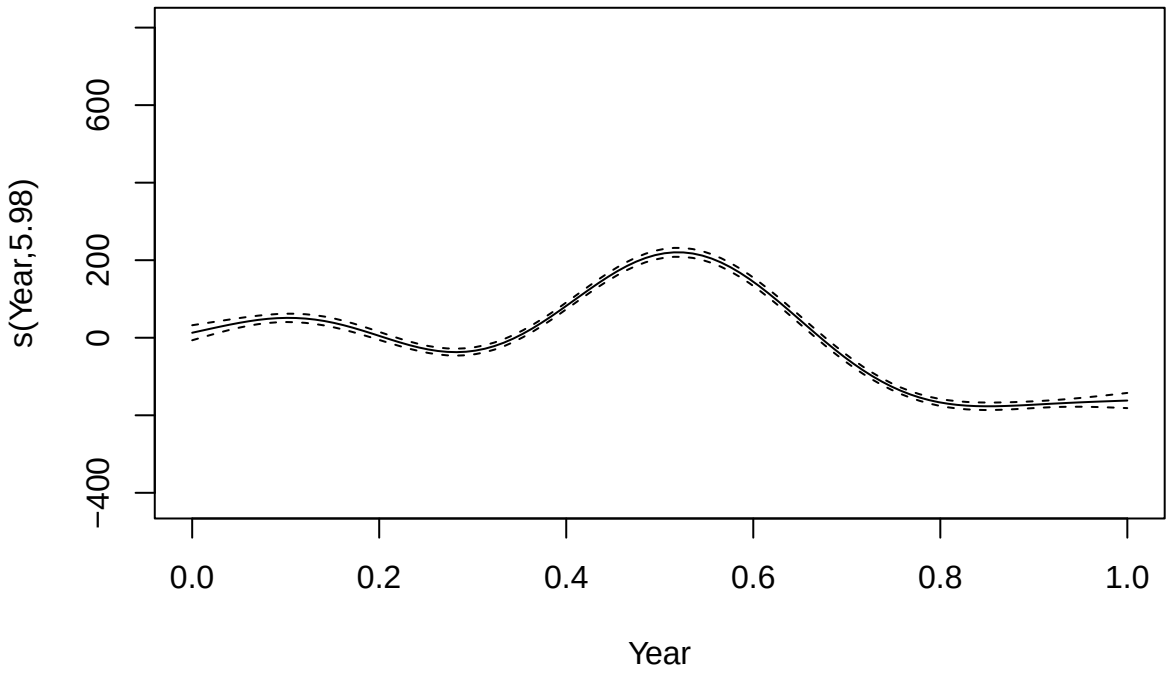
```
gamlwmod <- Demand ~  s(DSTTime, bs = "cc", k = 12) + s(WtdTemp, bs = "tp", k =8) + s(Year, bs = "tp", k
wtdyear <- gamm(gamlwmod,  data = fitdata)
print(summary(wtdyear$gam))
```

```
##
## Family: gaussian
## Link function: identity
##
## Formula:
## Demand ~ s(DSTTime, bs = "cc", k = 12) + s(WtdTemp, bs = "tp",
##     k = 8) + s(Year, bs = "tp", k = 7)
##
## Parametric coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 1320.925      1.917   689.2   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Approximate significance of smooth terms:
##              edf Ref.df      F p-value
## s(DSTTime) 9.894 10.000 1102.9  <2e-16 ***
## s(WtdTemp) 6.890  6.890  317.9  <2e-16 ***
## s(Year)    5.981  5.981  562.4  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-sq.(adj) =  0.687
##   Scale est. = 46014     n = 12528
```

Plot each smooth term.

```
plot(wtdyear$gam, all.terms=T)
```


s(DSTTime,9.89)
600
200
0
−400
0.0
0.2
0.4
0.6
0.8
1.0
DSTTime
s(WtdTemp,6.89)
600
200
0
−400
10
15
20
25
30
35
WtdTemp

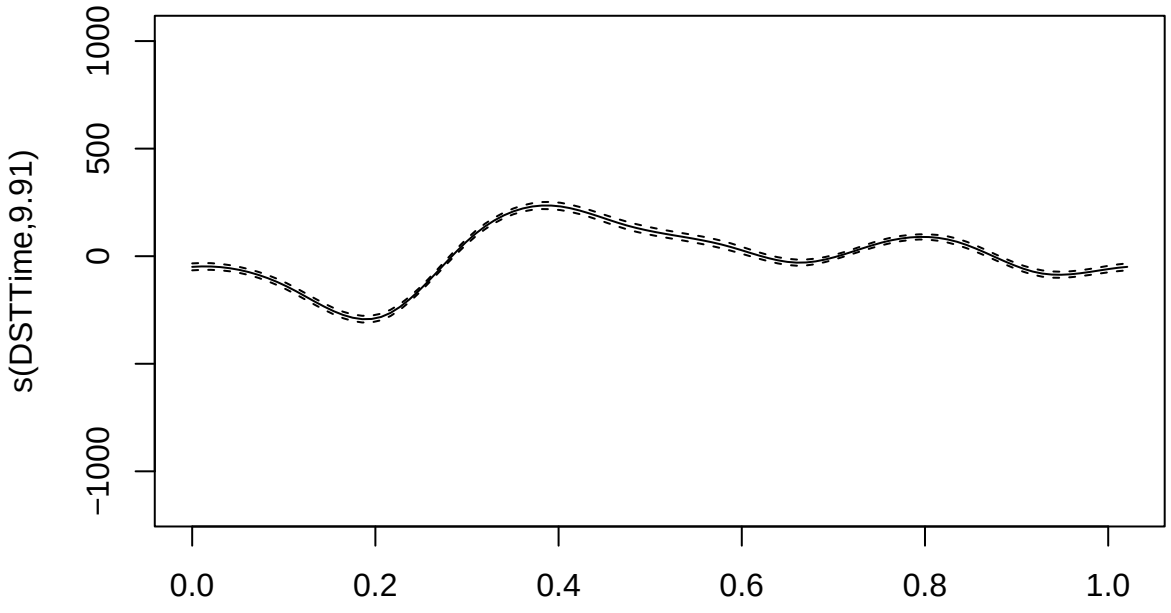


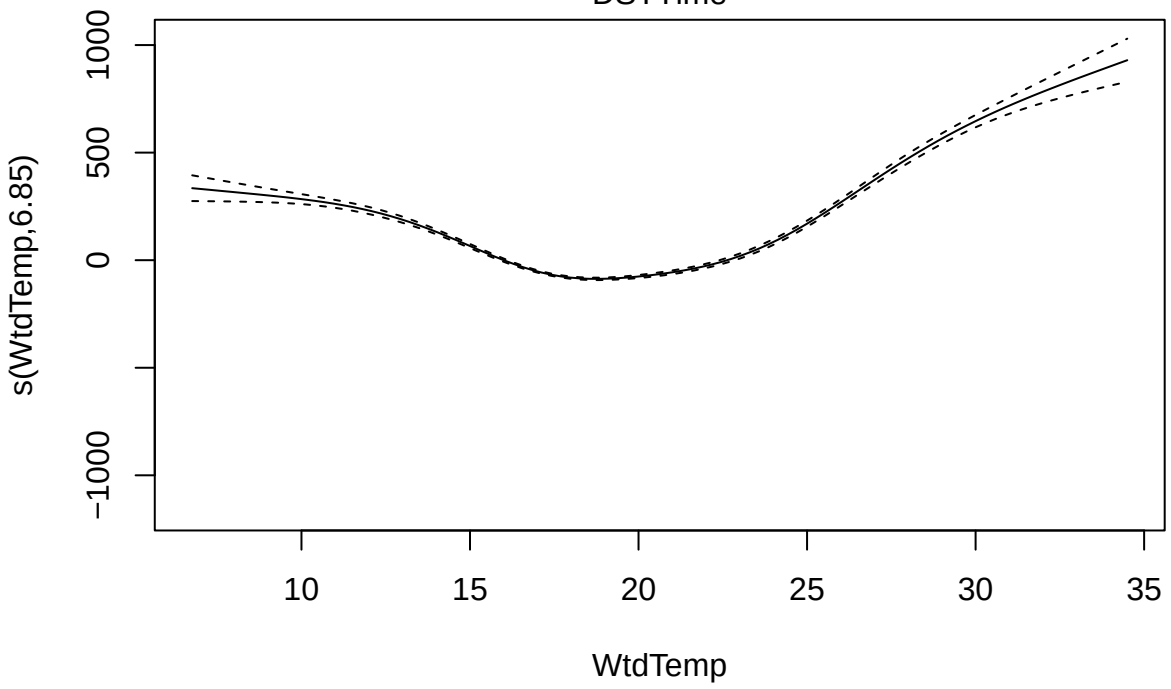
Fit again with smoothed radkjm2.

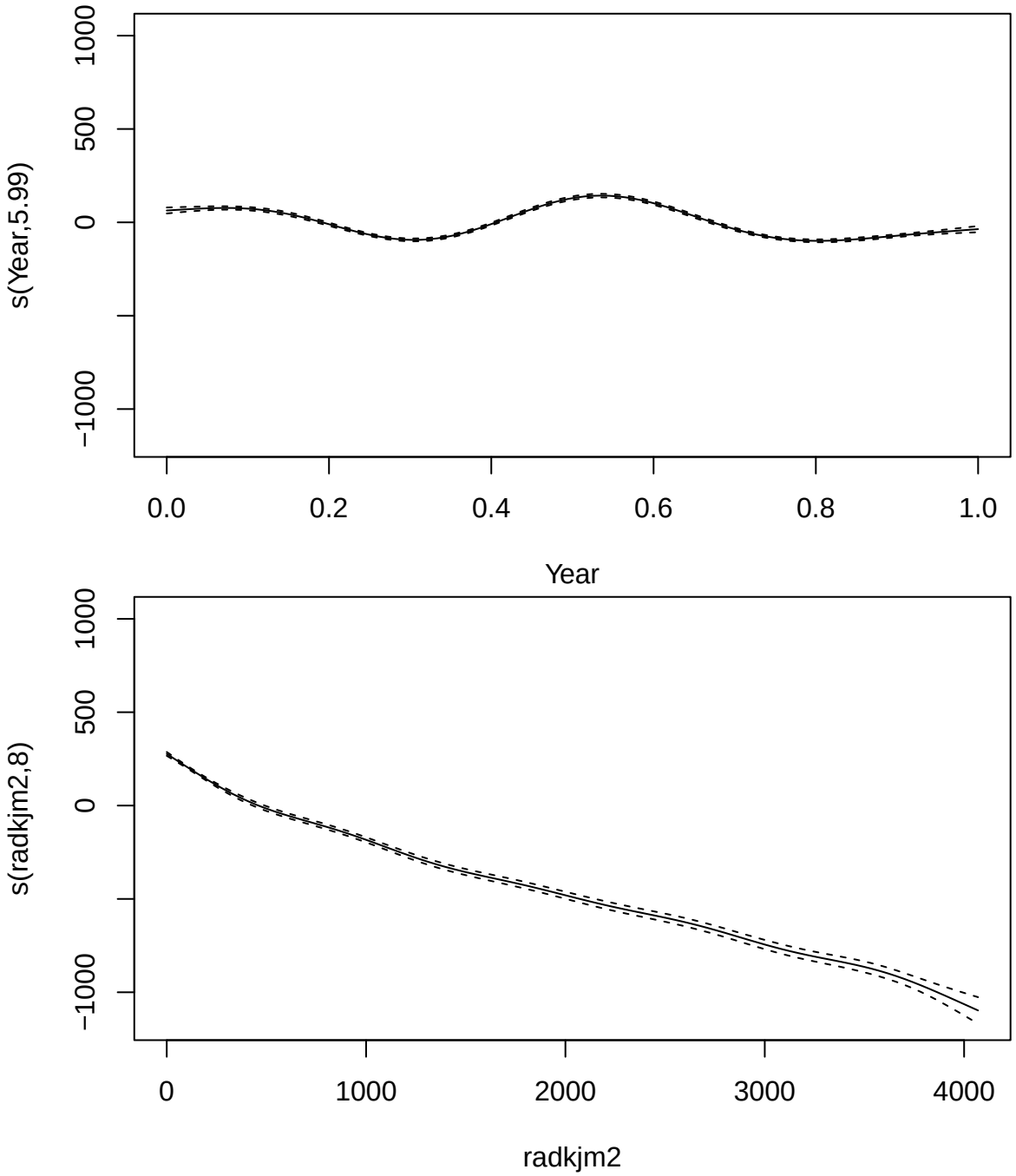
```
gamlwmod_weather <- Demand ~  s(DSTTime, bs = "cc", k = 12) + s(WtdTemp, bs = "tp", k =8) + s(Year, bs =
wtdyear_weather <- gamm(gamlwmod_weather,  data = fitdata)
print(summary(wtdyear_weather$gam))
```

```
##
## Family: gaussian
## Link function: identity
##
## Formula:
## Demand ~ s(DSTTime, bs = "cc", k = 12) + s(WtdTemp, bs = "tp",
##     k = 8) + s(Year, bs = "tp", k = 7) + s(radkjm2)
##
## Parametric coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 1320.925      1.575   838.8   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Approximate significance of smooth terms:
##              edf Ref.df     F p-value
## s(DSTTime) 9.911 10.000 306.5  <2e-16 ***
## s(WtdTemp) 6.848  6.848 625.9  <2e-16 ***
## s(Year)    5.986  5.986 340.6  <2e-16 ***
## s(radkjm2) 7.998  7.998 753.1  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-sq.(adj) =  0.789
##   Scale est. = 31063     n = 12528
```

```
plot(wtdyear_weather$gam, all.terms = T)
```


s(DSTTime,9.91)
1000
500
0
−1000
0.0
0.2
0.4
0.6
0.8
1.0
DSTTime


s(WtdTemp,6.85)
1000
500
0
−1000
10
15
20
25
30
35
WtdTemp



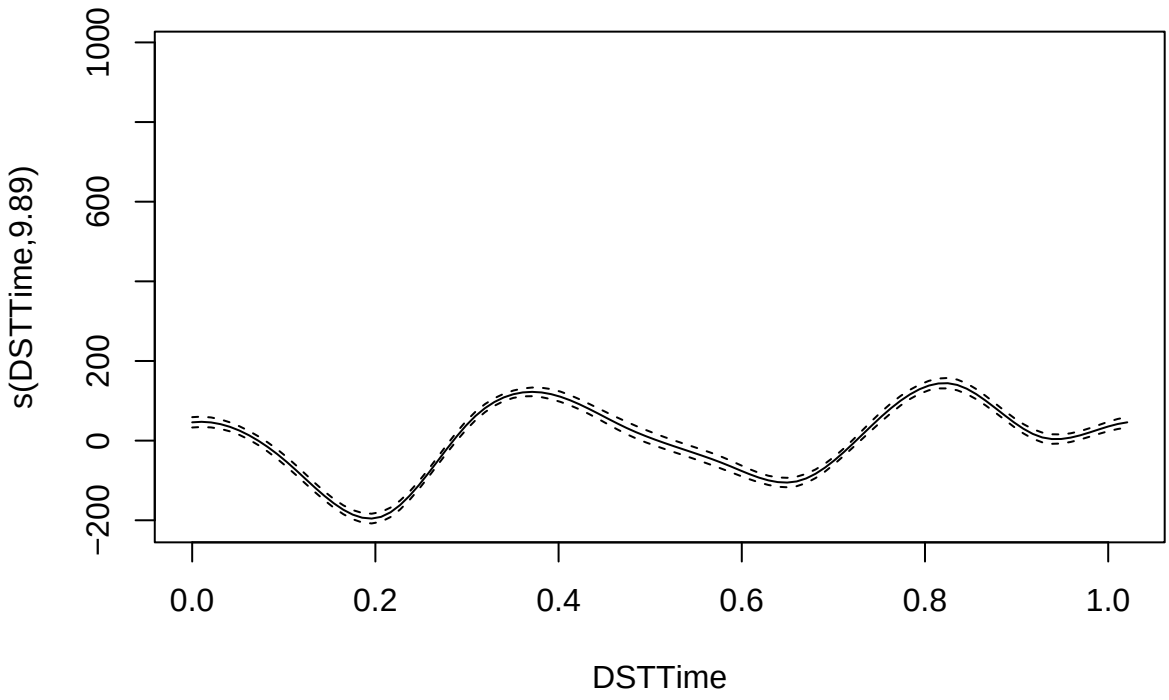
Fit again with all weather variables.

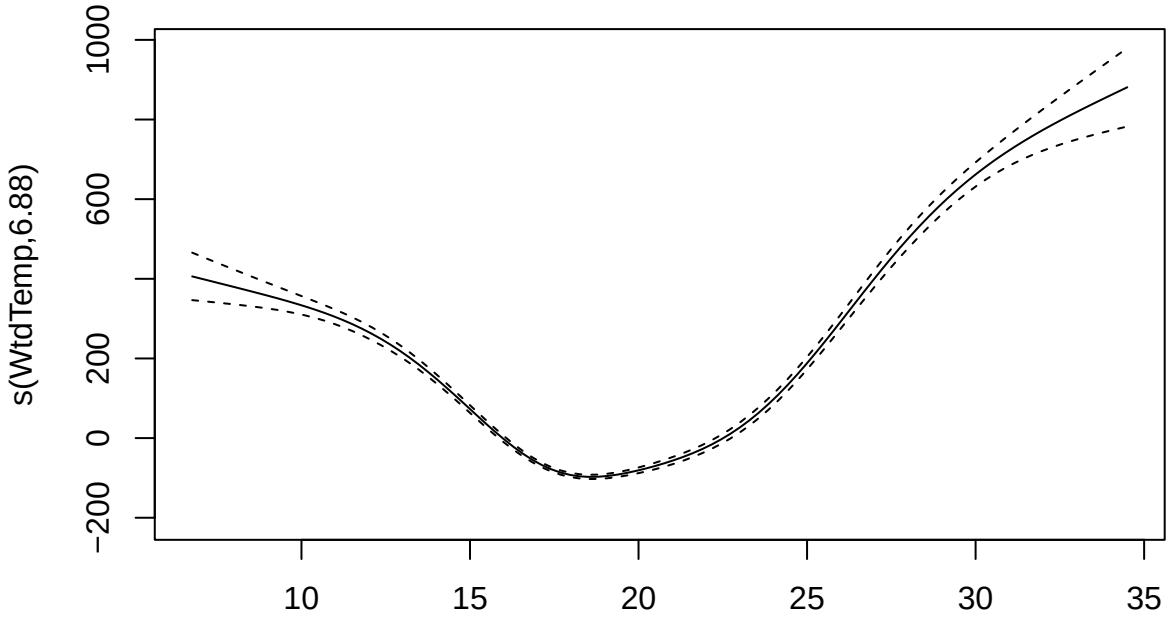
```
gamlwmod_weather <- Demand ~  s(DSTTime, bs = "cc", k = 12) + s(WtdTemp, bs = "tp", k =8) + s(Year, bs =
wtdyear_weather <- gamm(gamlwmod_weather,  data = fitdata)
print(summary(wtdyear_weather$gam))
```

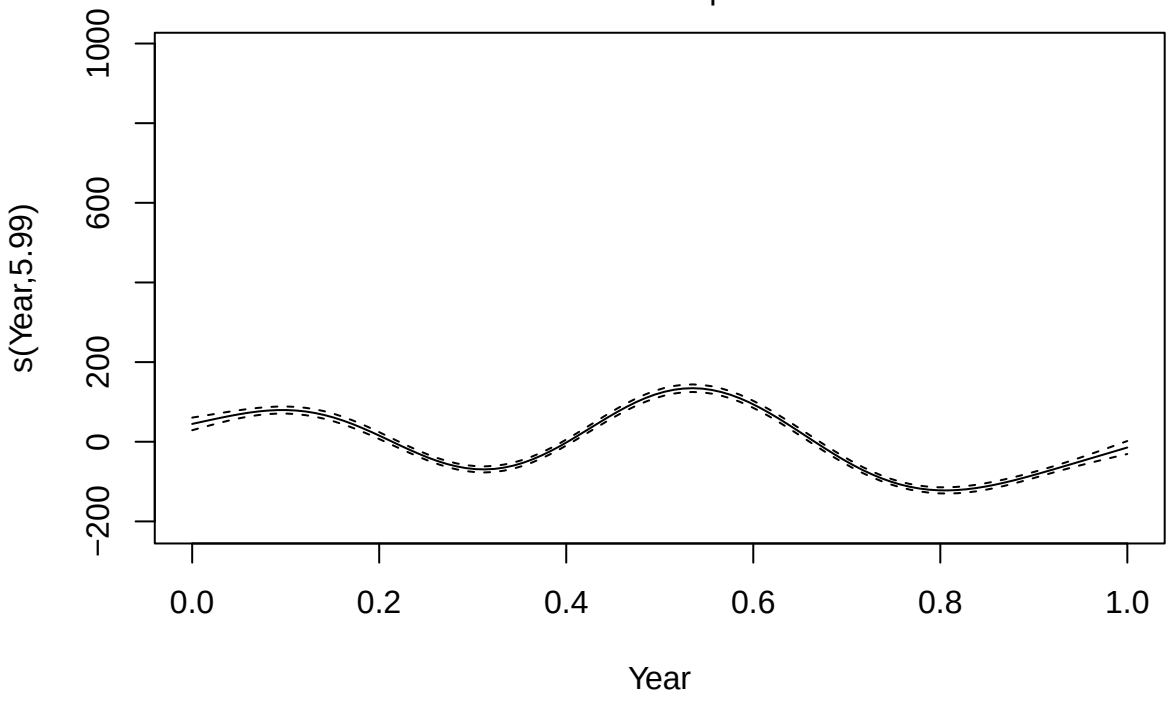
```
##
## Family: gaussian
## Link function: identity
##
## Formula:
## Demand ~ s(DSTTime, bs = "cc", k = 12) + s(WtdTemp, bs = "tp",
```

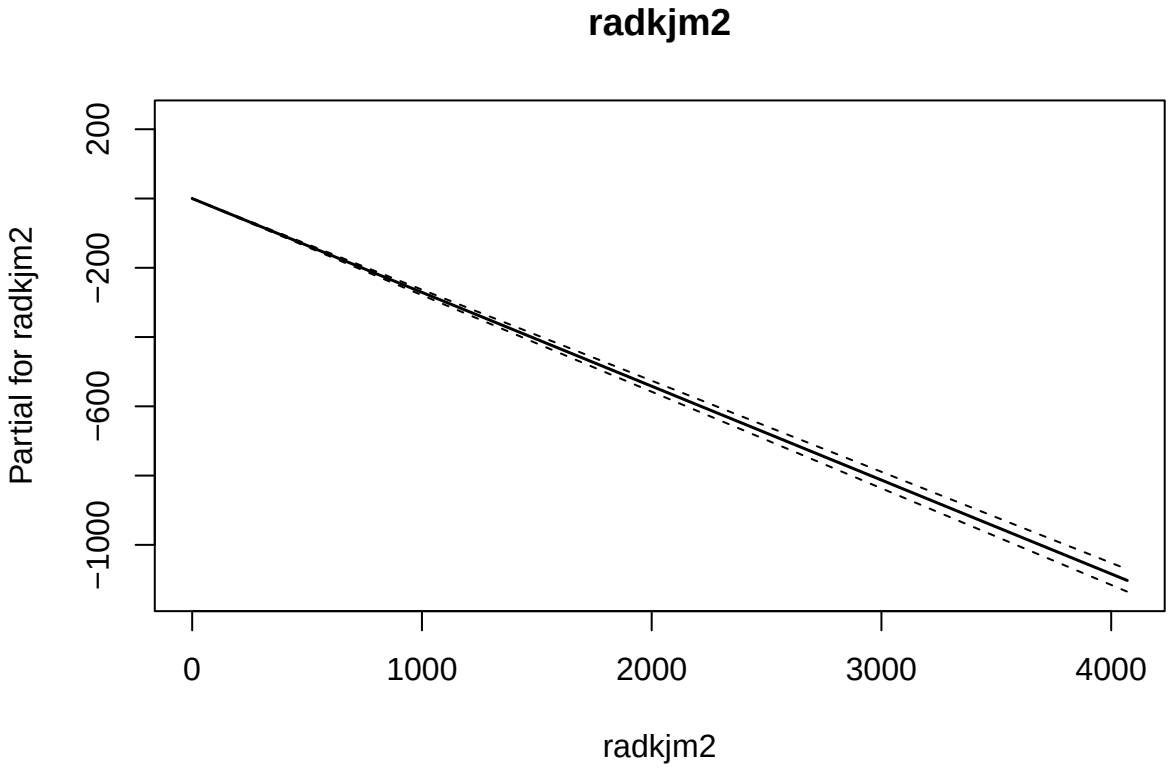
```
##     k = 8) + s(Year, bs = "tp", k = 7) + radkjm2 + cloud8 + windk +
##     wdir + humid + rainmm
##
## Parametric coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.298e+03  1.114e+01 116.549  < 2e-16 ***
## radkjm2     -2.710e-01  3.975e-03 -68.164  < 2e-16 ***
## cloud8       1.145e+01  7.822e-01  14.642  < 2e-16 ***
## windk        6.523e+00  4.051e-01  16.103  < 2e-16 ***
## wdir        -1.473e-01  1.914e-02  -7.694 1.53e-14 ***
## humid        1.731e+00  1.455e-01  11.895  < 2e-16 ***
## rainmm       1.659e+01  4.808e+00   3.450 0.000562 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Approximate significance of smooth terms:
##              edf Ref.df     F p-value
## s(DSTTime) 9.893 10.000 280.0  <2e-16 ***
## s(WtdTemp) 6.885  6.885 695.0  <2e-16 ***
## s(Year)    5.985  5.985 351.6  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-sq.(adj) =  0.797
##   Scale est. = 29835     n = 12528
```

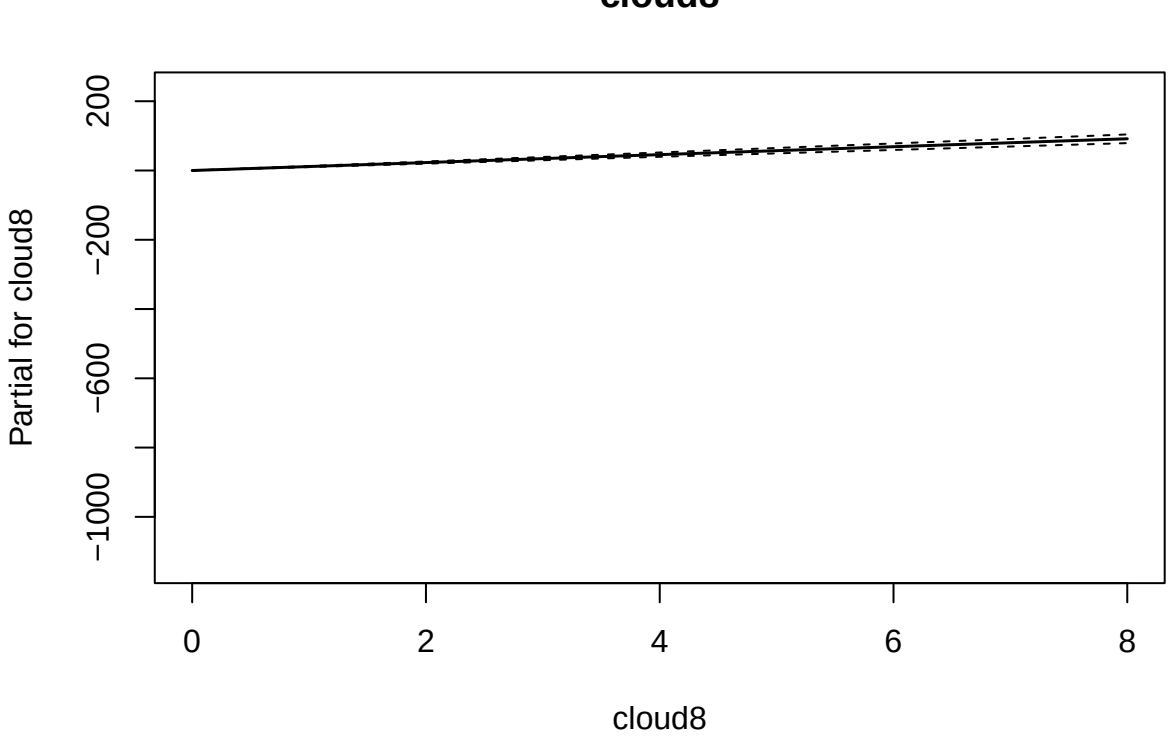
```
plot(wtdyear_weather$gam, all.terms = T)
```

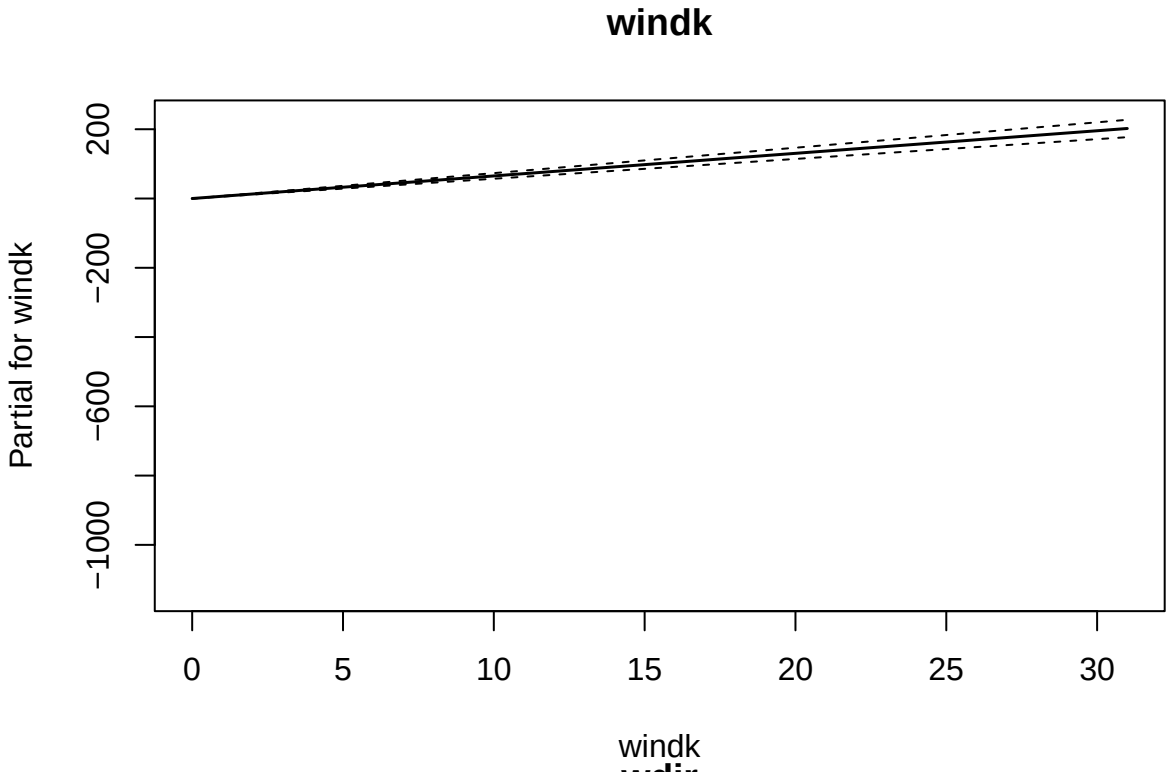


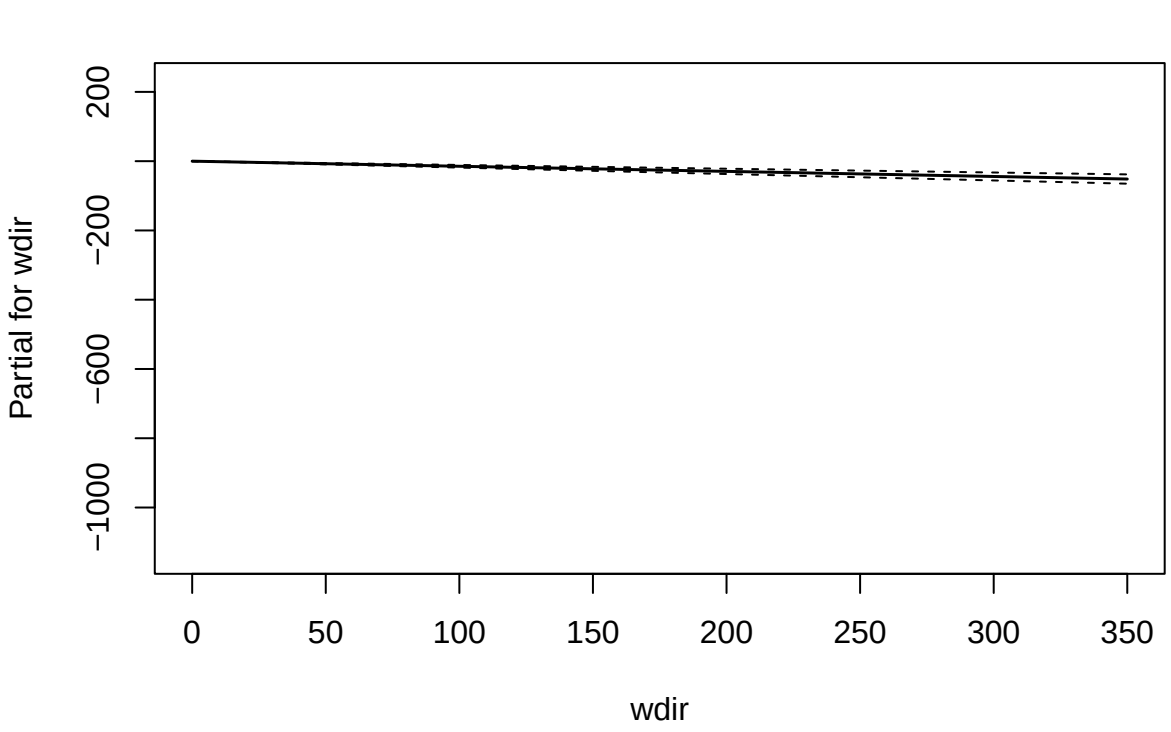

s(WtdTemp,6.88)
1000
600
200
0
−200
10
15
20
25
30
35
WtdTemp

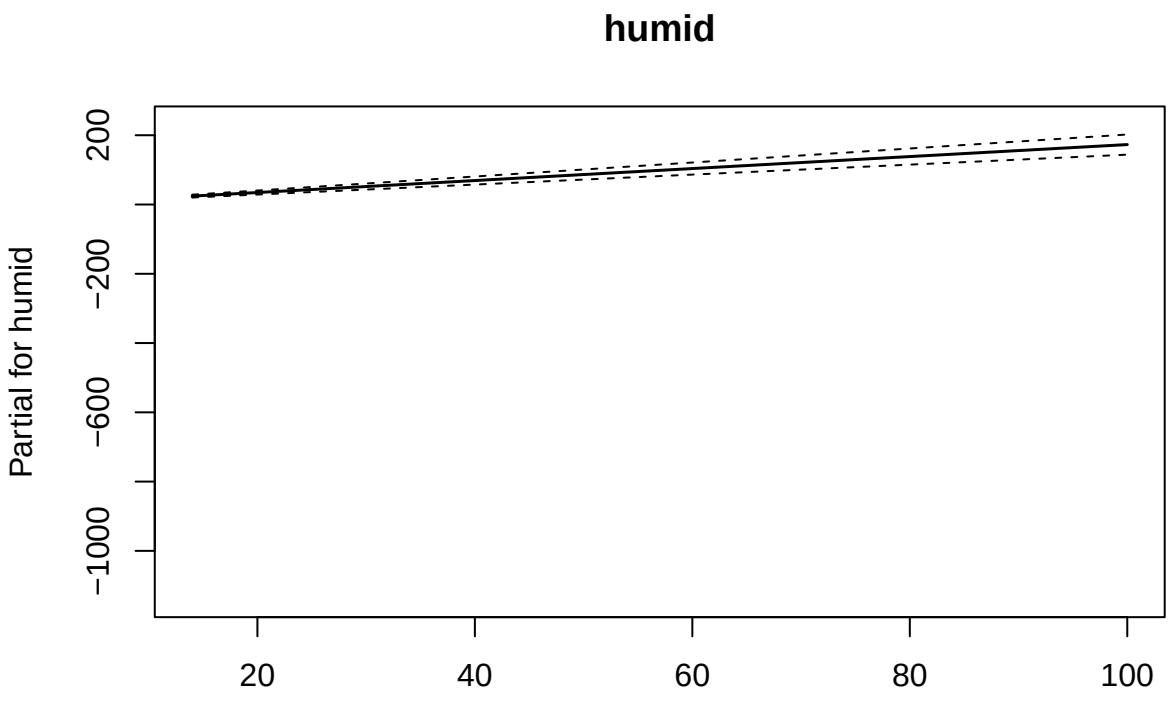

s(Year,5.99)
1000
600
200
0
−200
0.0
0.2
0.4
0.6
0.8
1.0
Year

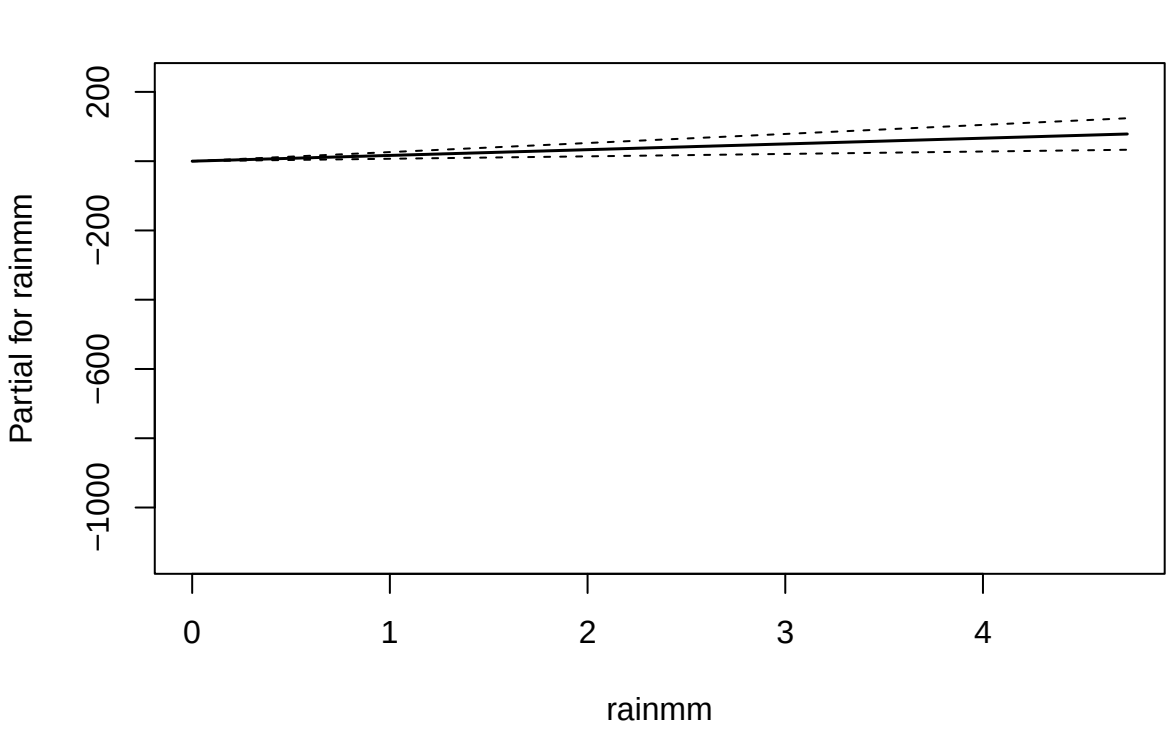

radkjm2
Partial for radkjm2
200
−200
−600
−1000
0
1000
2000
3000
4000
radkjm2


cloud8
Partial for cloud8
200
−200
−600
−1000
0
2
4
6
8
cloud8


windk
Partial for windk
200
−200
−600
−1000
0
5
10
15
20
25
30
windk


wdir
Partial for wdir
200
−200
−600
−1000
0
50
100
150
200
250
300
350
wdir





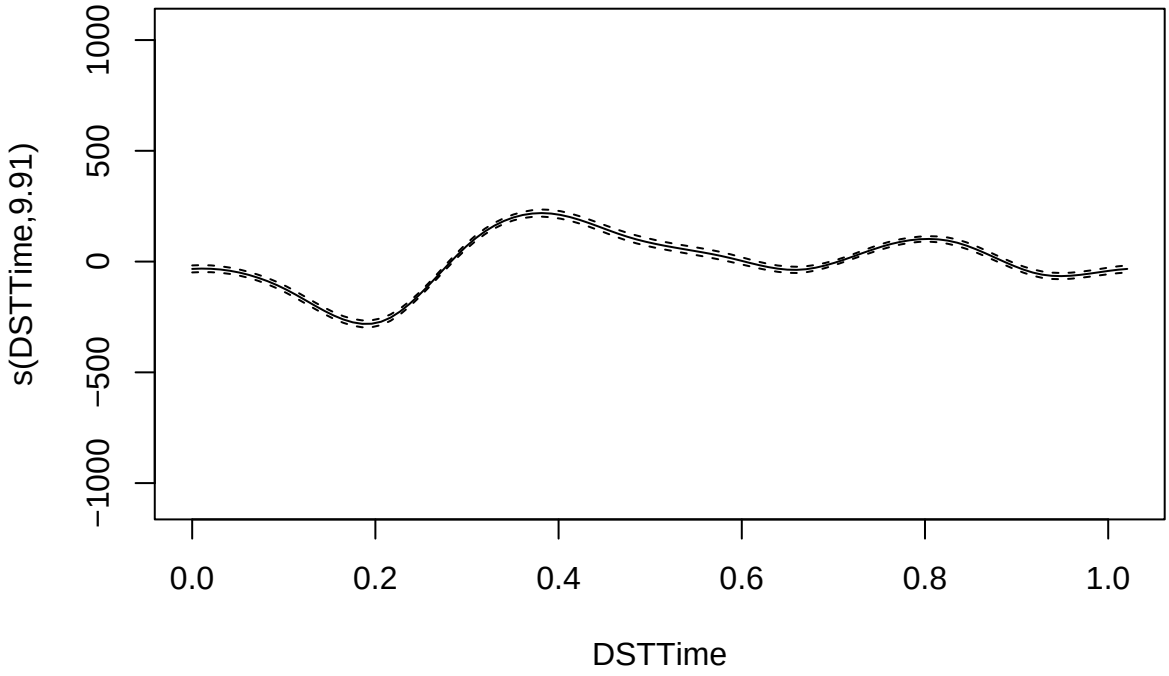
Fit again with all weather variables, with radkjm2 smoothed.

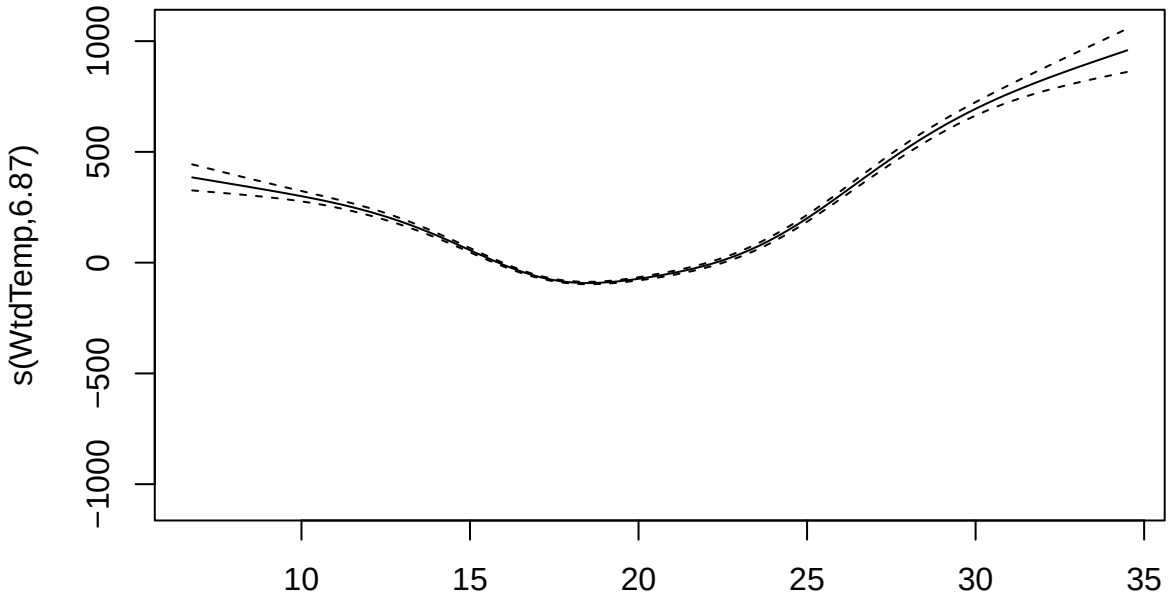
```
gamlwmod_weather <- Demand ~  s(DSTTime, bs = "cc", k = 12) + s(WtdTemp, bs = "tp", k =8) + s(Year, bs =
wtdyear_weather <- gamm(gamlwmod_weather,  data = fitdata)
print(summary(wtdyear_weather$gam))
```

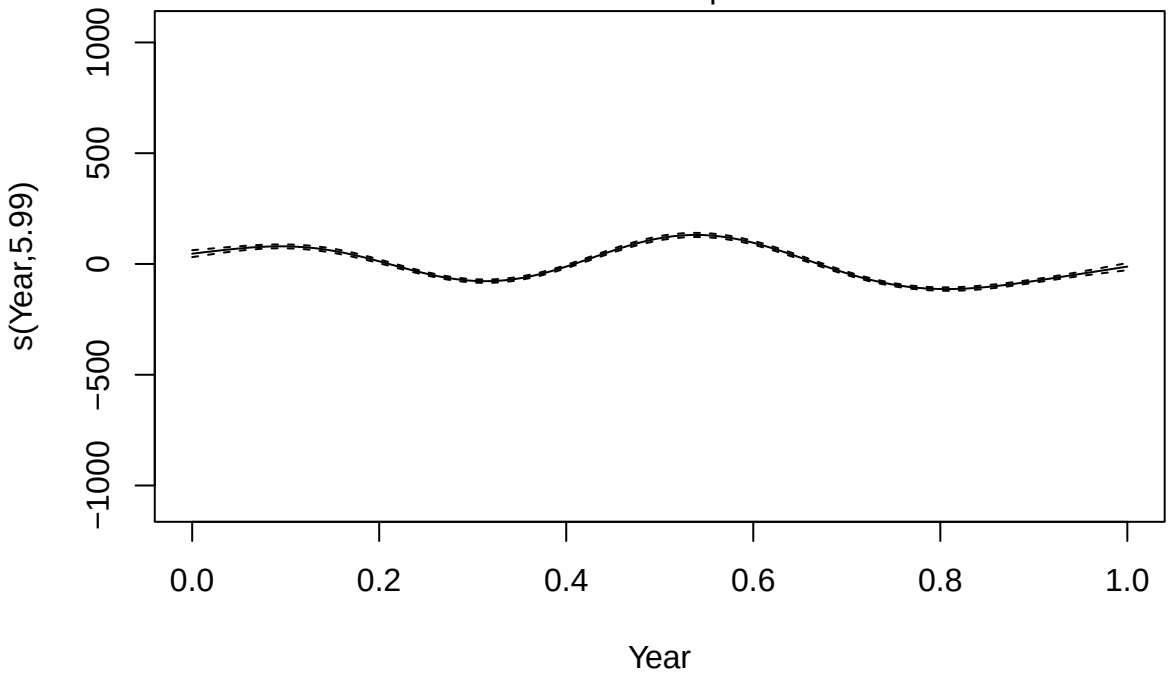
```
##
```

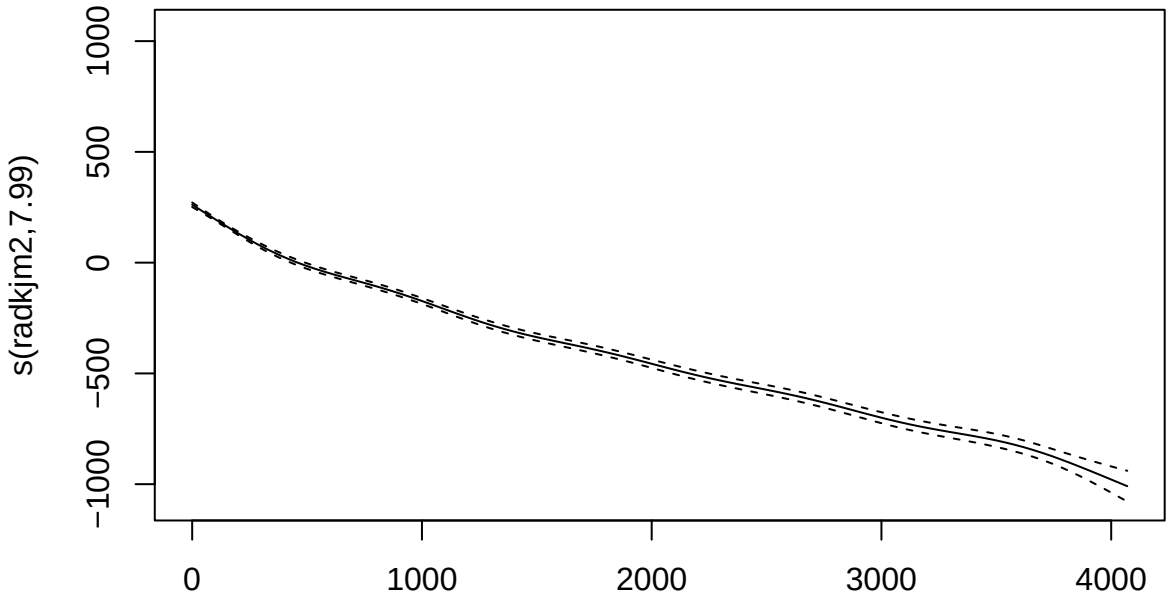
```
## Family: gaussian
## Link function: identity
##
## Formula:
## Demand ~ s(DSTTime, bs = "cc", k = 12) + s(WtdTemp, bs = "tp",
##     k = 8) + s(Year, bs = "tp", k = 7) + s(radkjm2) + cloud8 +
##     windk + wdir + humid + rainmm
##
## Parametric coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) 1105.34960   10.63713 103.914  < 2e-16 ***
## cloud8        10.53372    0.77441  13.602  < 2e-16 ***
## windk          7.16962    0.40166  17.850  < 2e-16 ***
## wdir          -0.15233    0.01891  -8.055  8.7e-16 ***
## humid          1.81408    0.14385  12.611  < 2e-16 ***
## rainmm         7.17144    4.77590   1.502    0.133
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Approximate significance of smooth terms:
##              edf Ref.df     F p-value
## s(DSTTime) 9.914 10.000 319.7  <2e-16 ***
## s(WtdTemp) 6.867  6.867 681.6  <2e-16 ***
## s(Year)    5.986  5.986 337.5  <2e-16 ***
## s(radkjm2) 7.989  7.989 638.5  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-sq.(adj) =  0.802
##   Scale est. = 29054     n = 12528
```

```
plot(wtdyear_weather$gam, all.terms = T)
```

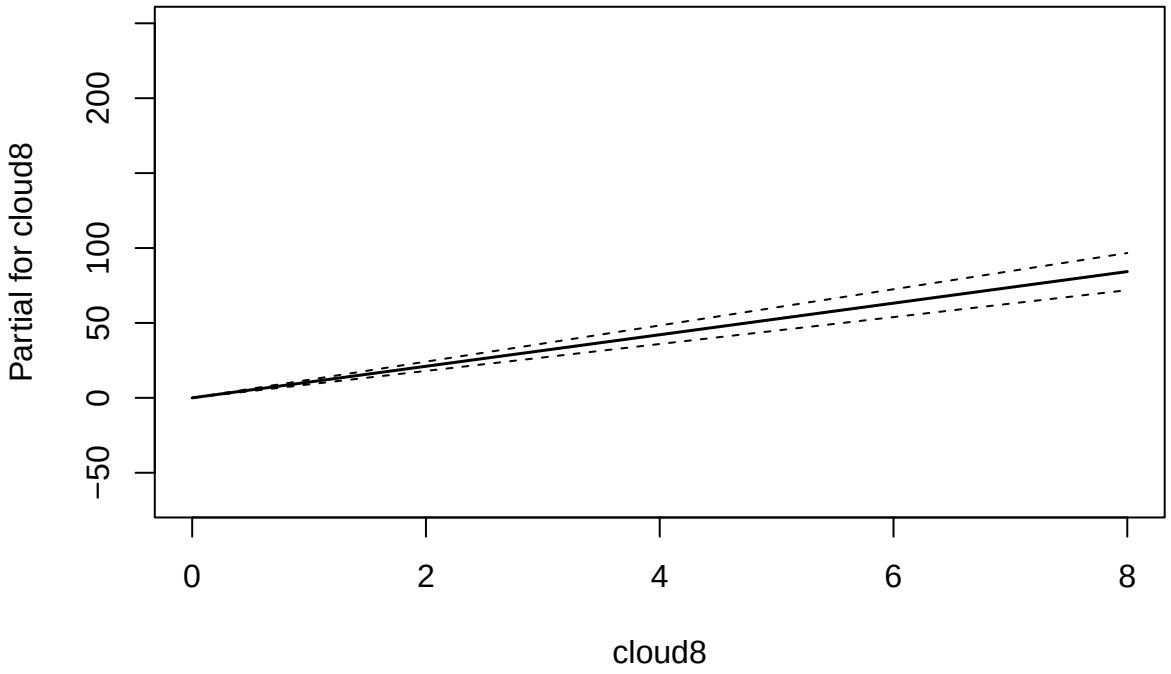


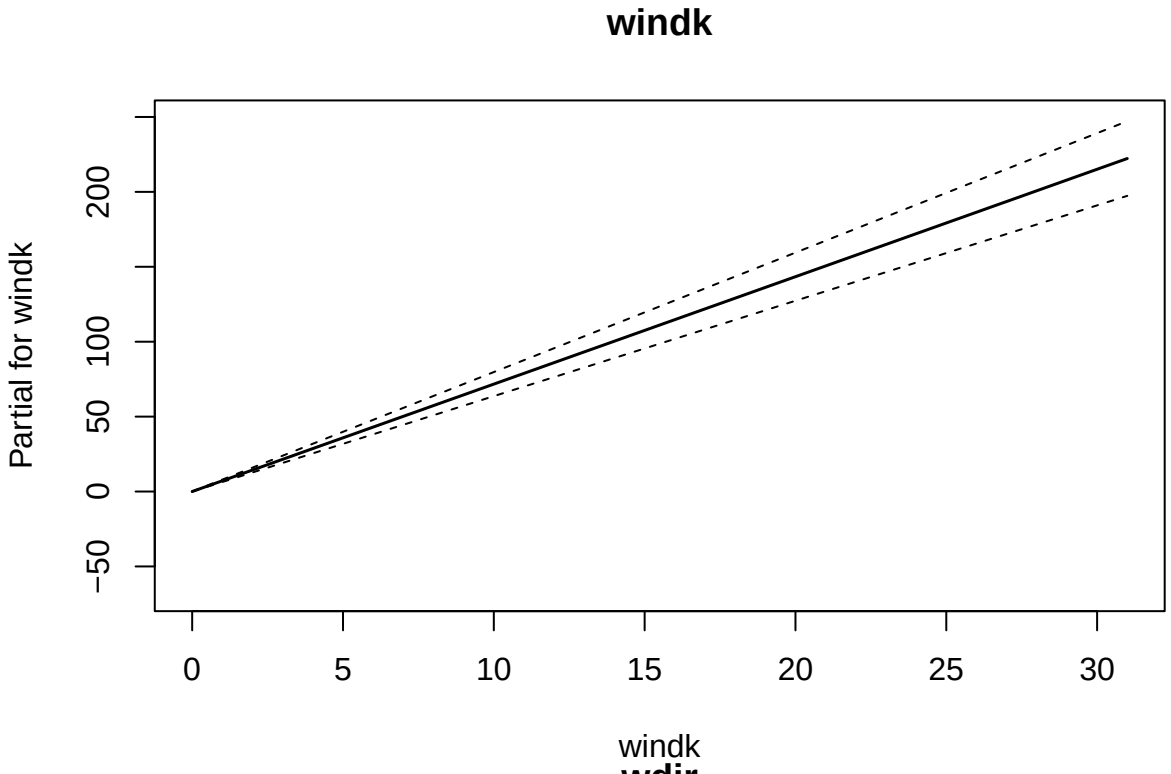

s(WtdTemp,6.87)
1000
500
0
−500
−1000
10
15
20
25
30
35
WtdTemp

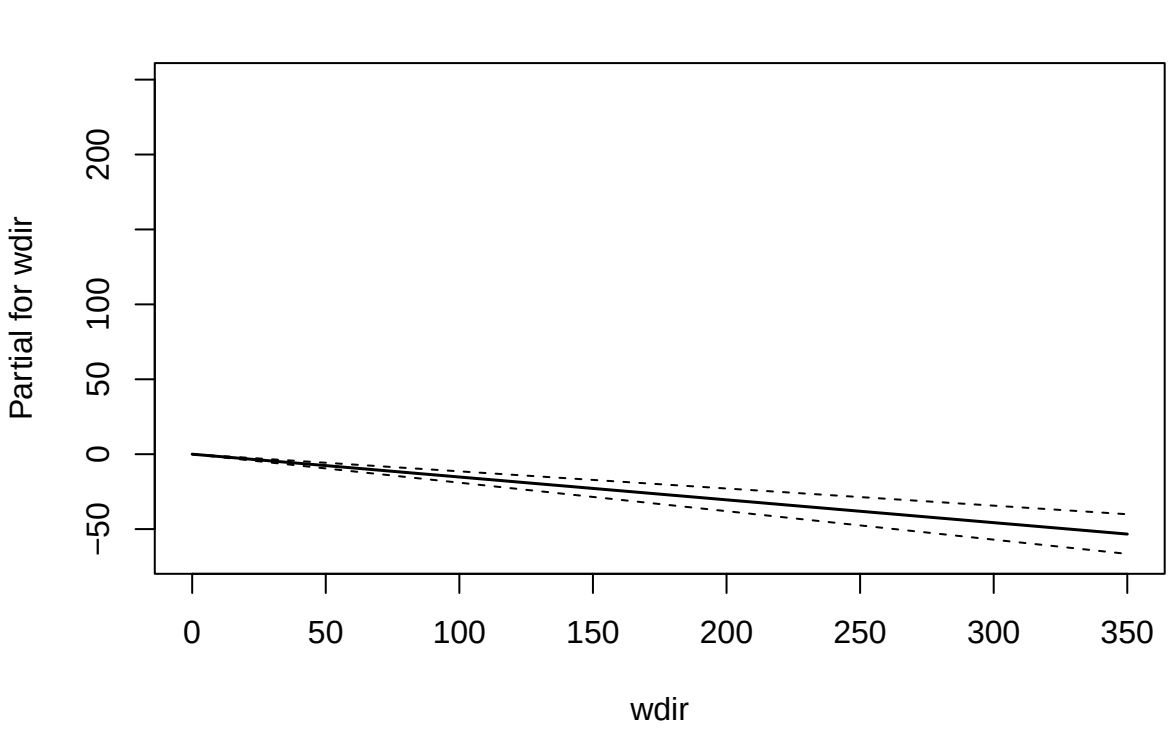

s(Year,5.99)
1000
500
0
−500
−1000
0.0
0.2
0.4
0.6
0.8
1.0
Year

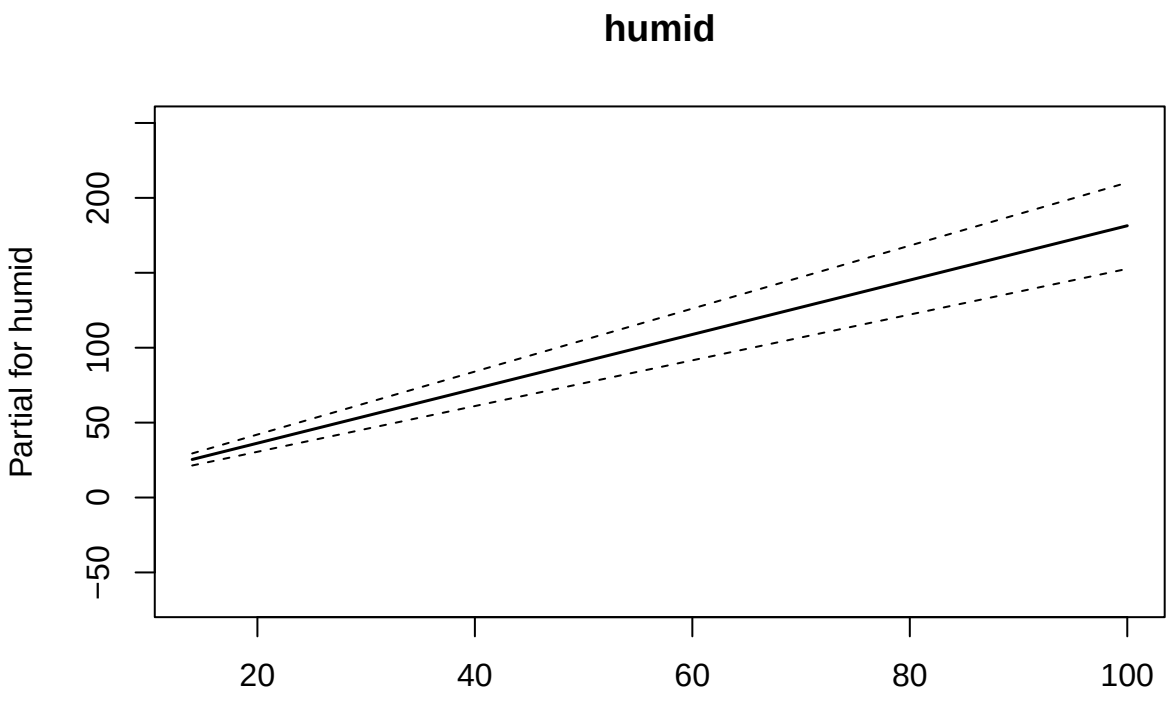

s(radkjm2,7.99)
1000
500
0
−500
−1000
0
1000
2000
3000
4000
radkjm2

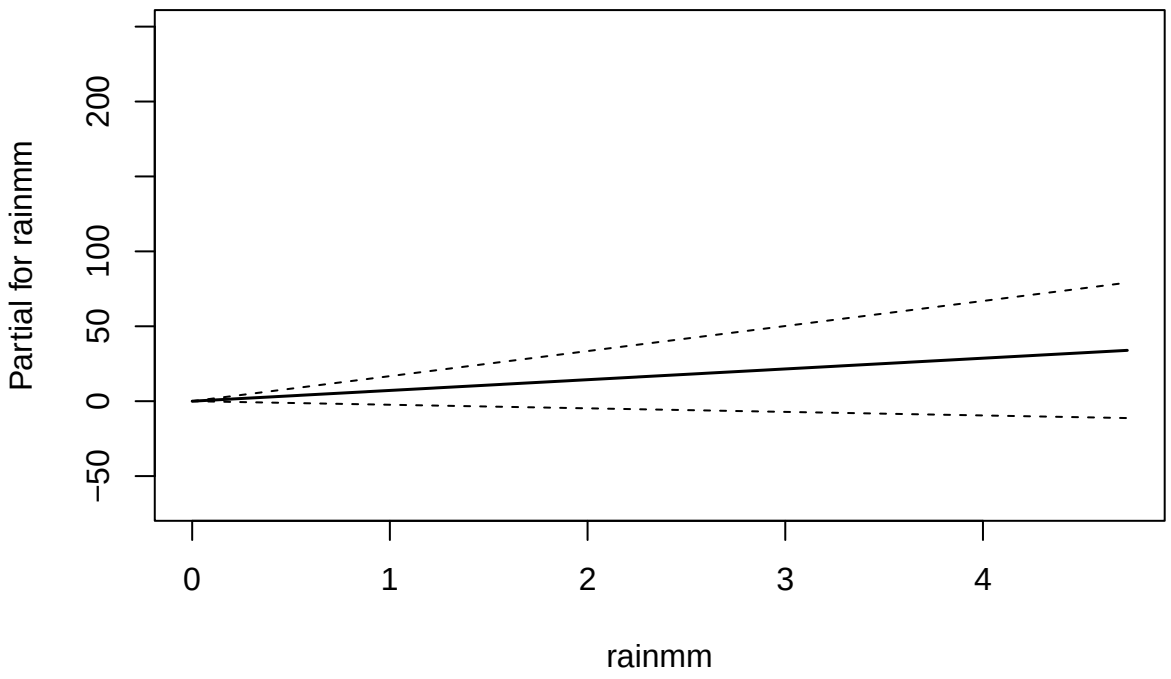
cloud8


Partial for cloud8
200
100
50
0
−50
0
2
4
6
8
cloud8


windk
Partial for windk
−50
0
50
100
200
0
5
10
15
20
25
30
windk


wdir
Partial for wdir
−50
0
50
100
200
0
50
100
150
200
250
300
350
wdir





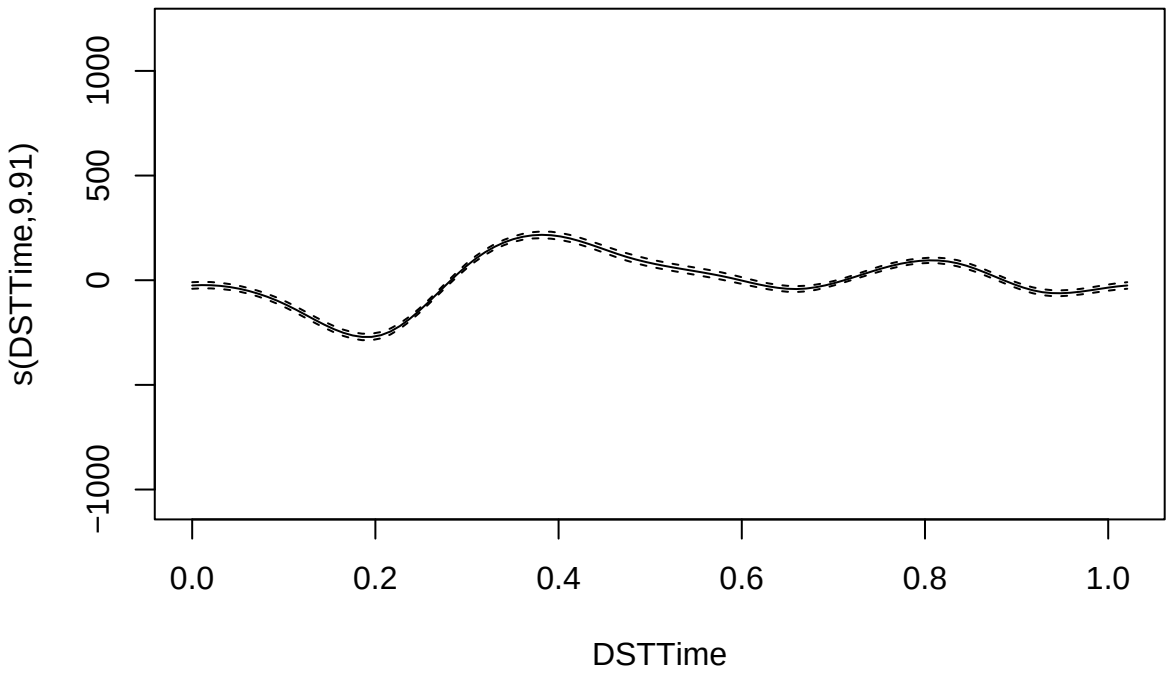
Fit again with all weather variables smoothed.

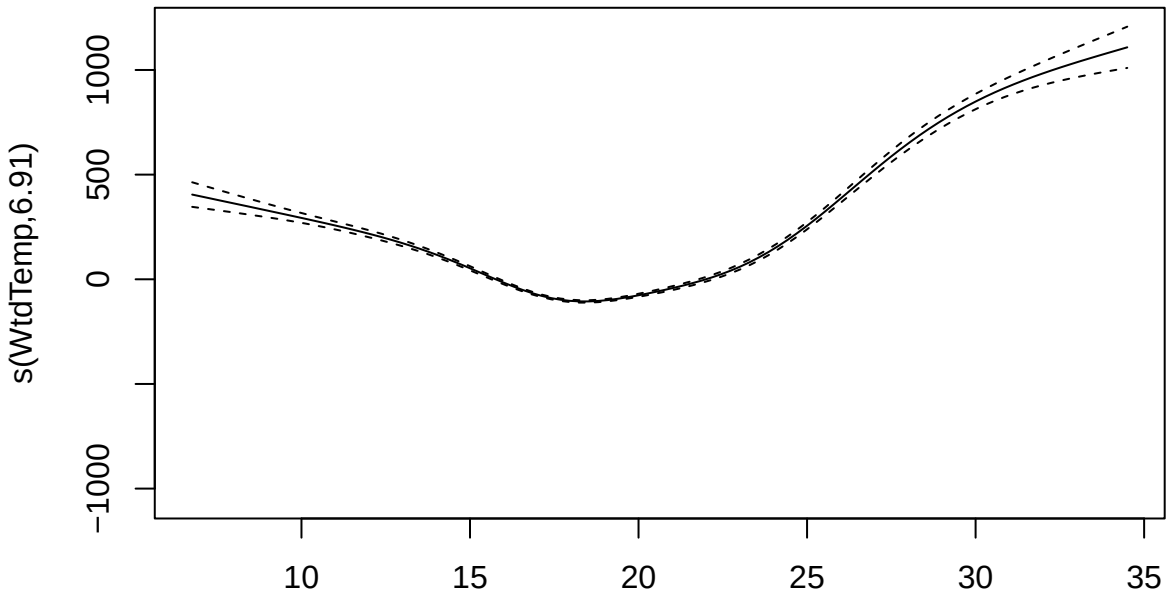
```
gamlwmod_weather <- Demand ~  s(DSTTime, bs = "cc", k = 12) + s(WtdTemp, bs = "tp", k =8) + s(Year, bs =
wtdyear_weather <- gamm(gamlwmod_weather,  data = fitdata)
print(summary(wtdyear_weather$gam))
```

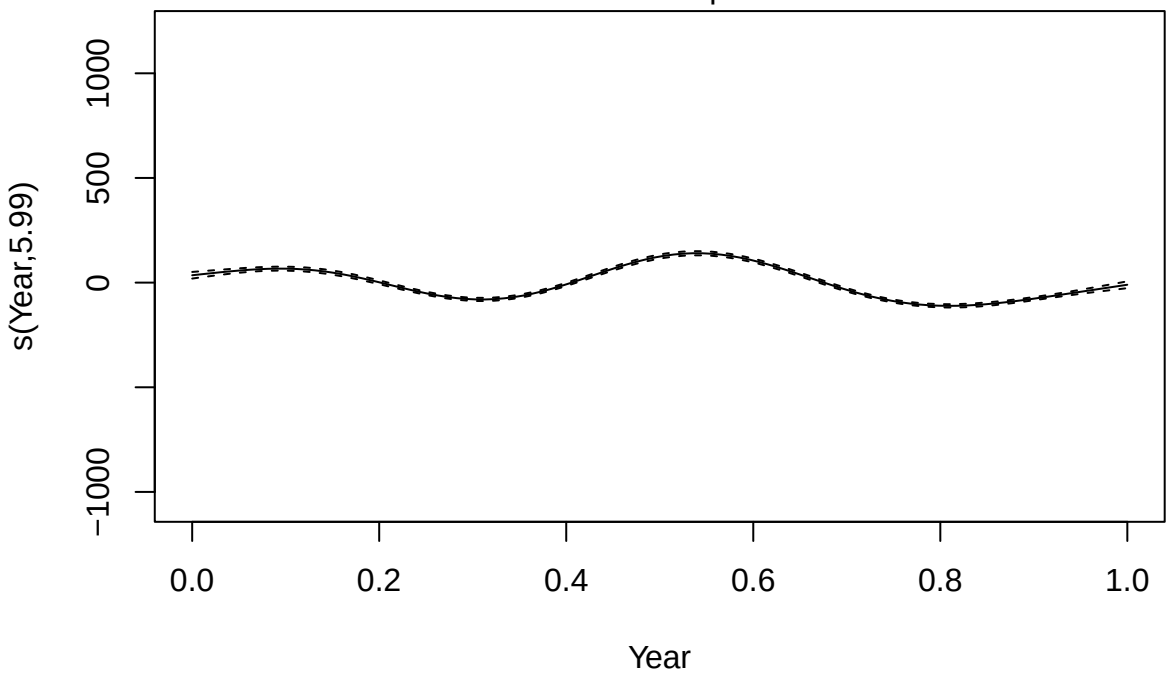
```
##
```

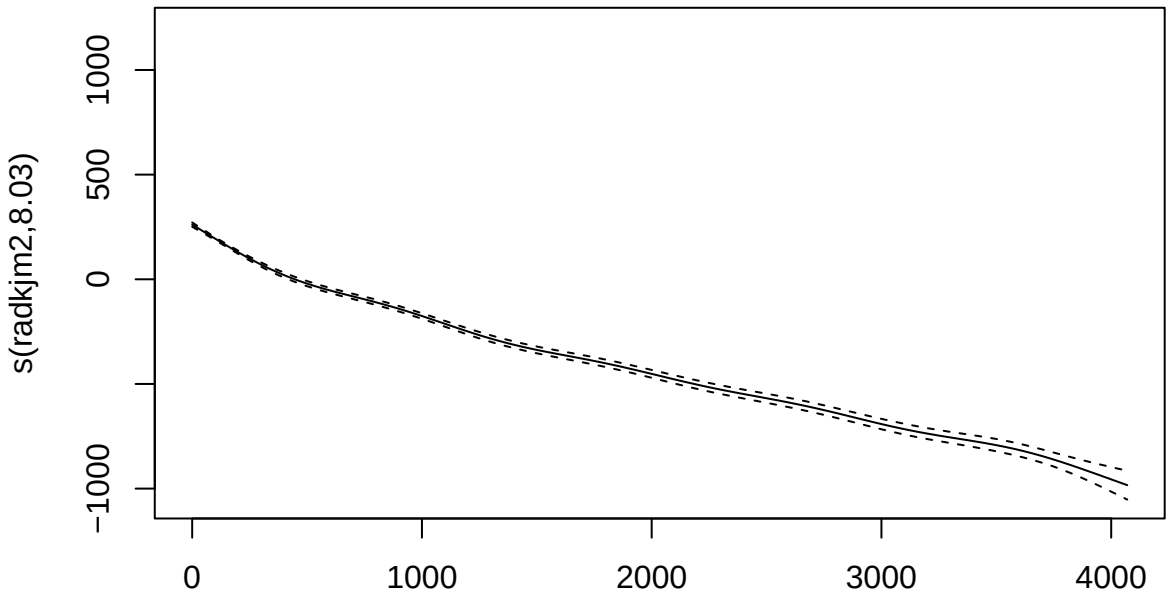
```
## Family: gaussian
## Link function: identity
##
## Formula:
## Demand ~ s(DSTTime, bs = "cc", k = 12) + s(WtdTemp, bs = "tp",
##     k = 8) + s(Year, bs = "tp", k = 7) + s(radkjm2) + s(cloud8) +
##     s(windk) + s(wdir) + s(humid) + s(rainmm)
##
## Parametric coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 1320.925      1.494   884.4   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Approximate significance of smooth terms:
##               edf Ref.df      F p-value
## s(DSTTime) 9.914 10.000 312.01  <2e-16 ***
## s(WtdTemp) 6.908  6.908 607.33  <2e-16 ***
## s(Year)    5.986  5.986 323.74  <2e-16 ***
## s(radkjm2) 8.025  8.025 594.21  <2e-16 ***
## s(cloud8)  1.000  1.000 187.29  <2e-16 ***
## s(windk)   4.260  4.260  67.75  <2e-16 ***
## s(wdir)    8.035  8.035  20.84  <2e-16 ***
## s(humid)   6.757  6.757  71.32  <2e-16 ***
## s(rainmm)  5.224  5.224  11.84  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-sq.(adj) =   0.81
##   Scale est. = 27948     n = 12528
```

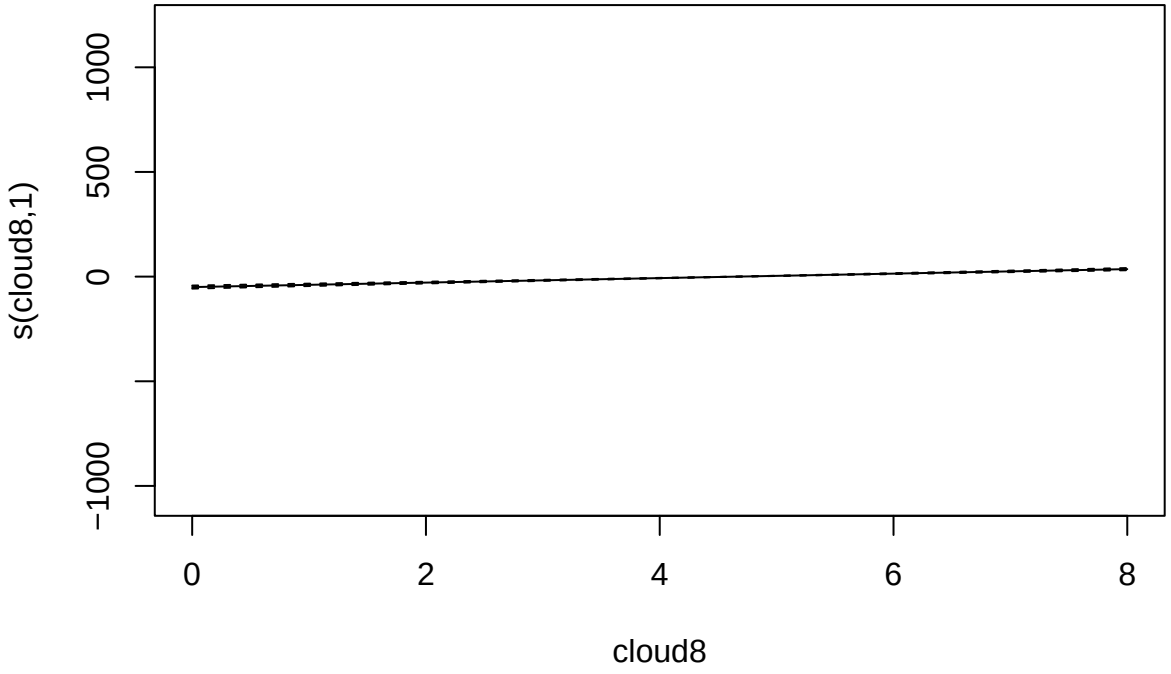
```
plot(wtdyear_weather$gam, all.terms = T)
```

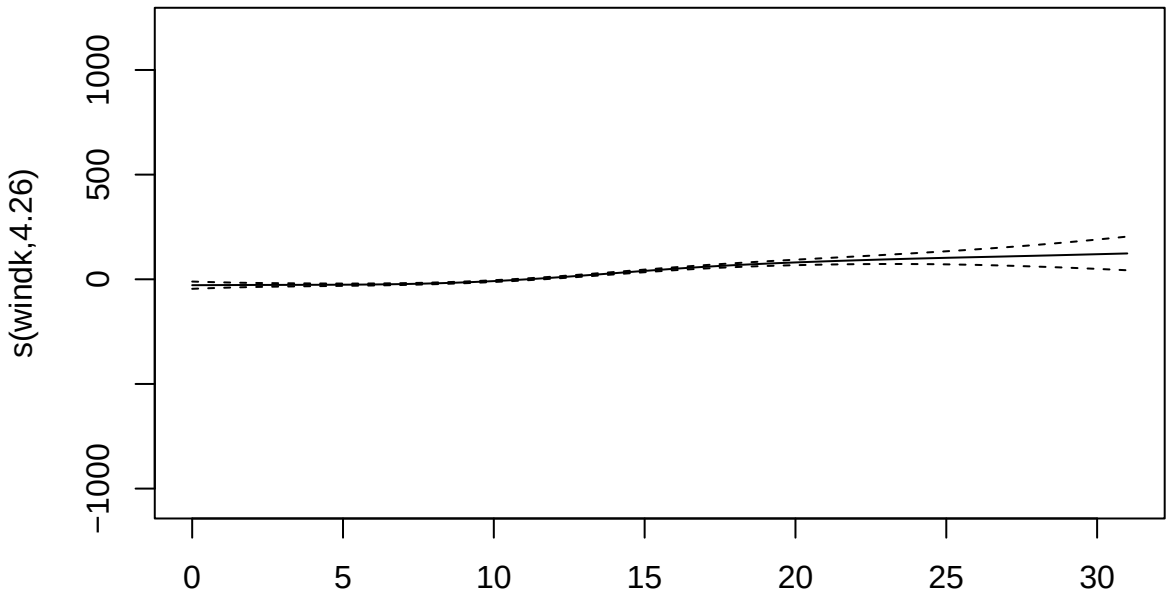


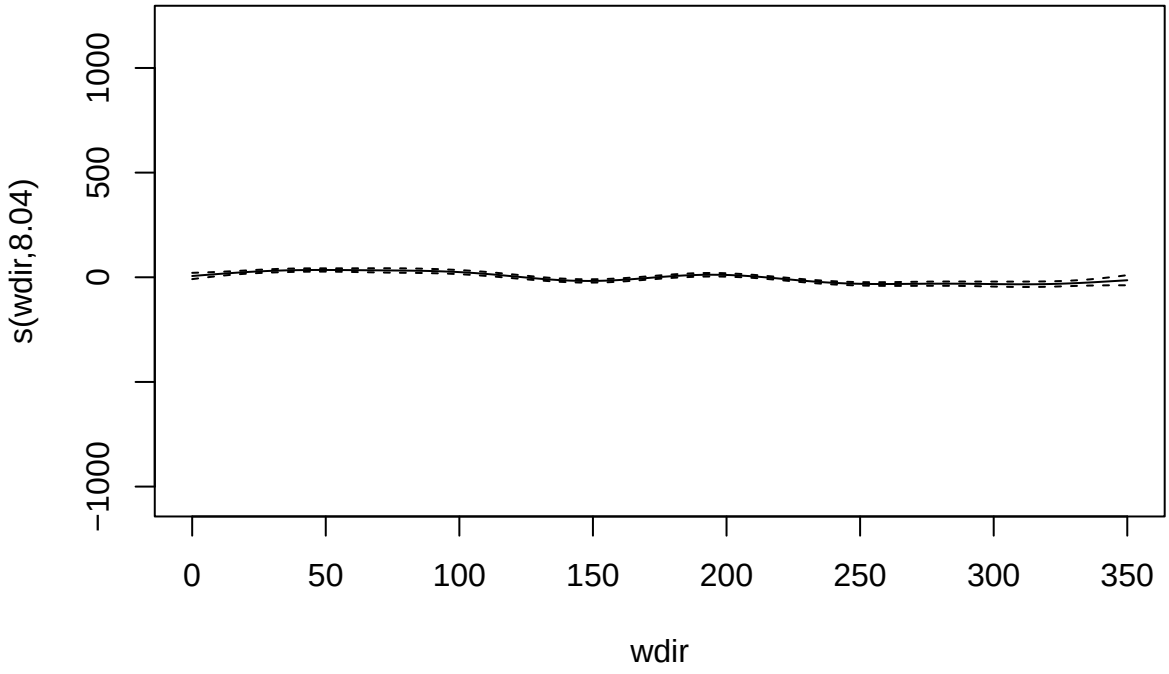

s(WtdTemp,6.91)
1000
500
0
−1000
10
15
20
25
30
35
WtdTemp

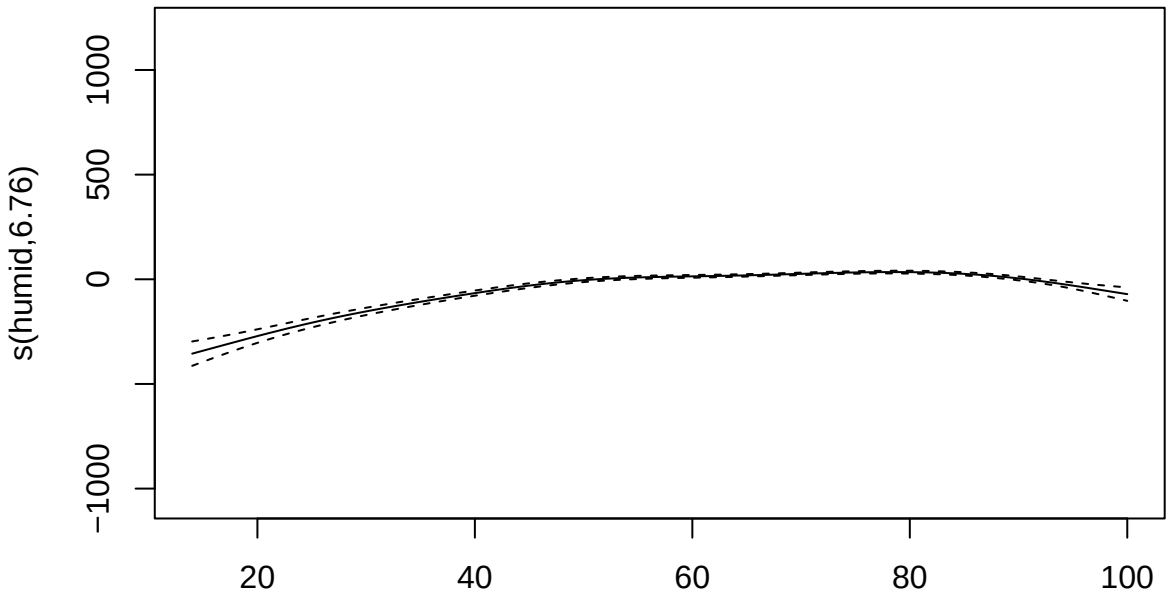

s(Year,5.99)
1000
500
0
−1000
0.0
0.2
0.4
0.6
0.8
1.0
Year

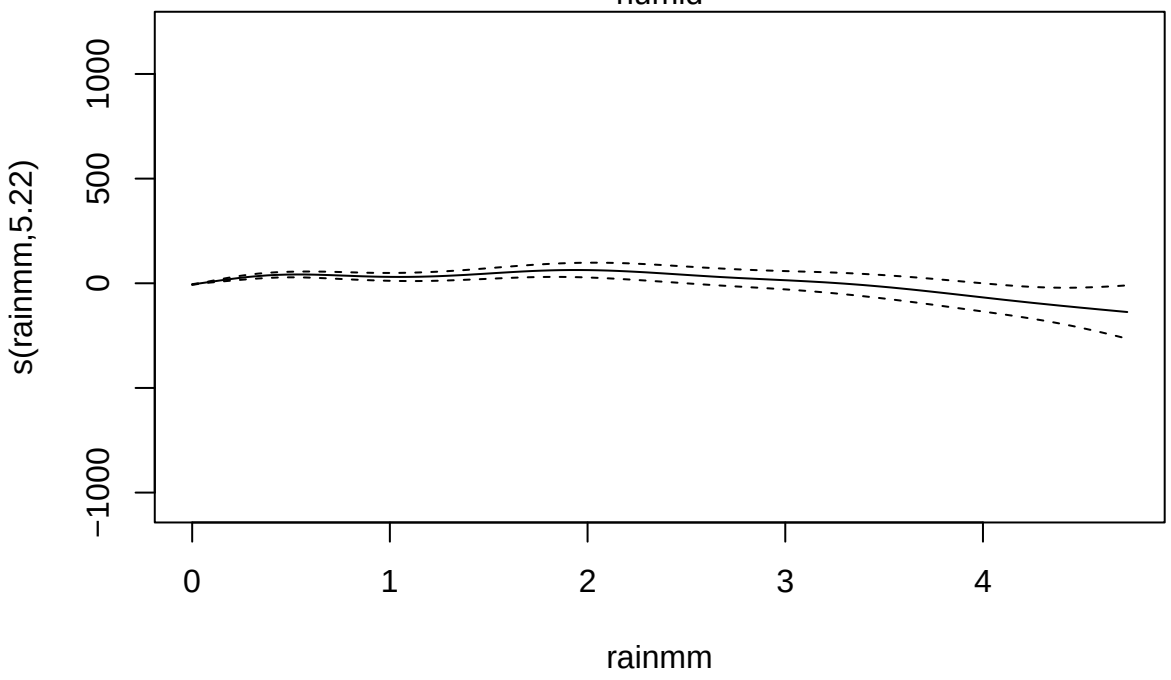

s(radkjm2,8.03)
1000
500
0
−1000
0
1000
2000
3000
4000
radkjm2


s(cloud8,1)
1000
500
0
−1000
0
2
4
6
8
cloud8


s(windk,4.26)
1000
500
0
−1000
0
5
10
15
20
25
30
windk


s(wdir,8.04)
1000
500
0
−1000
0
50
100
150
200
250
300
350
wdir


s(humid,6.76)
1000
500
0
−1000
20
40
60
80
100
humid


s(rainmm,5.22)
1000
500
0
−1000
0
1
2
3
4
rainmm