

Generalized Additive Models
oooooooooooo

Multiple Smoothers
oooooooooooooooooooo

Bivariate Splines
oooooooooooooooooooo

Module 5: Generalized Additive Models

BIOS 526

Reading

- From Wood, S. *Generalized Additive Models: An Introduction with R, 2nd Edition*: Chapter 4, 5.1.2, 5.3.1, 5.4, 5.5.1, 6.1.2, 6.10, 6.12.
- A nice resource for fitting GAMs:
https://wiki.qcbs.ca/r_workshop8.

Concepts

- Generalized additive models (logistic).
- Additive models with multiple covariates
- Using `gam()` in package `mgcv()` to fit semiparametric models with parametric (slopes) and nonparametric terms (functions)
- Bivariate splines.
- Interpreting non-linear effects.

Generalized Additive Models

Binary Outcome Example

Recall the example from Module 4:

Dataset: a cohort of live births from Georgia born in the year 2001 (N = 77,340).

Variables:

- ptb : indicator for whether the baby from pregnancy i was born preterm (< 37 weeks).
 - age : the mother's age at delivery.
 - $male$: indicator of the baby's sex (1 = male; 0=female).
 - $tobacco$: indicator for mother's tobacco use during pregnancy (1 = yes; 0 = no)

Previous analysis

```
### Fit logistic regression model  
> fit = glm(ptb~age + male+tobacco, data = dat, family = binomial(link='logit'))  
> summary(fit)
```

Call:

```
glm(formula = ptb ~ age + male + tobacco, family = binomial(link = "logit"),
    data = dat)
```

Deviance Residuals:

	Min	1Q	Median	3Q	Max
	-0.5159563	-0.4235975	-0.4102807	-0.4087970	2.2499946

Coefficients:

	Estimate	Std. Error	z value	Pr(> z)
(Intercept)	-2.421266448	0.063135527	-38.35030	< 2.22e-16 ***
age	-0.000629473	0.002159576	-0.29148	0.7706843
maleM	0.072365862	0.025867177	2.79759	0.0051485 **
tobacco	0.409649486	0.053462733	7.66234	1.8258e-14 ***

Signif. codes: 0 ‘****’ 0.001 ‘***’ 0.01 ‘**’ 0.05 ‘*’ 0.1 ‘.’ 1

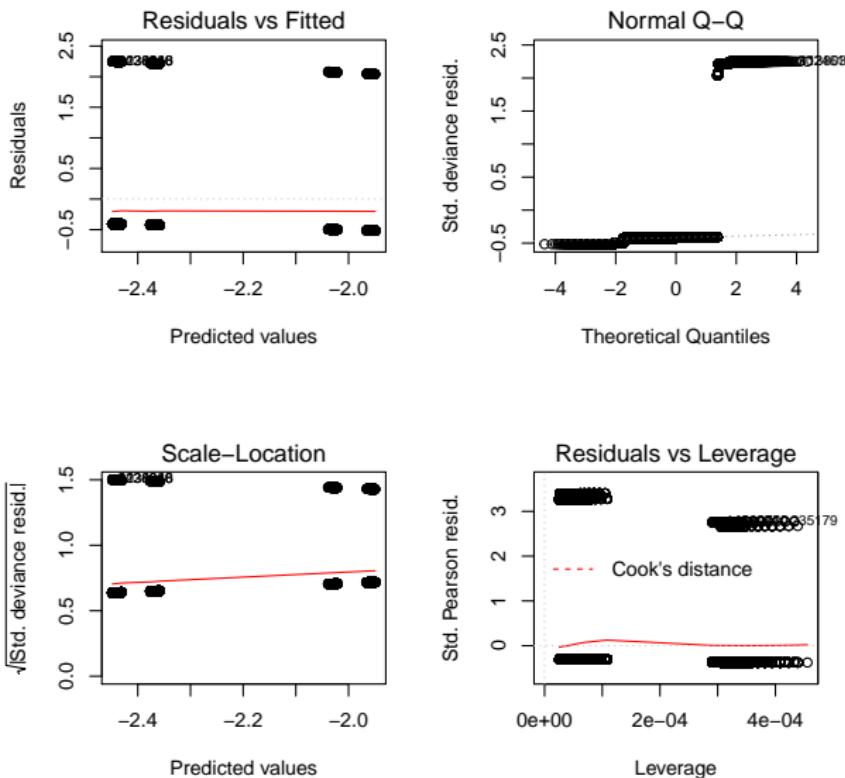
(Dispersion parameter for binomial family taken to be 1)

Null deviance: 44907.564 on 77339 degrees of freedom

Residual deviance: 44845.557 on 77336 degrees of freedom

AIC: 44853.557

The plot diagnostics are not very helpful with binary responses:



Generalized Additive Model

To account for non-linear age effect

$$ptb_i \sim \text{Bernoulli}(p_i)$$

$$\log \left(\frac{p_i}{1 - p_i} \right) = \beta_0 + \beta_1 male_i + \beta_2 tobacco_i + s(age_i),$$

where $s(\text{age}_i)$ is a **smooth** function of age. The above model is known as a **generalized additive model**.

GAMs: Generalized Additive Models.

Everything we learned about additive models is directly applicable in this setting!

GAM with logit link

Using the `mgcv:::gam` function:

```
> fit.gam = gam(ptb~s(age)+male+tobacco,family=binomial,data=dat)
> summary(fit.gam)
```

Family: binomial

Link function: logit

Formula:

```
ptb ~ s(age) + male + tobacco
```

Parametric coefficients:

	Estimate	Std. Error	z value	Pr(> z)
(Intercept)	-2.44226	0.01913	-127.647	< 2e-16 ***
maleM	0.07274	0.02588	2.811	0.00494 **
tobacco	0.39016	0.05356	7.284	3.24e-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(age)	3.314	4.146	70.17	3.85e-14 ***

Signif. codes: 0 ?***? 0.001 ?**? 0.01 ?*? 0.05 ?.? 0.1 ? ? 1

R-sq.(adj) = 0.00177 Deviance explained = 0.304%

UBRE = -0.42095 Scale est. = 1 n = 77340

GAM diagnostics

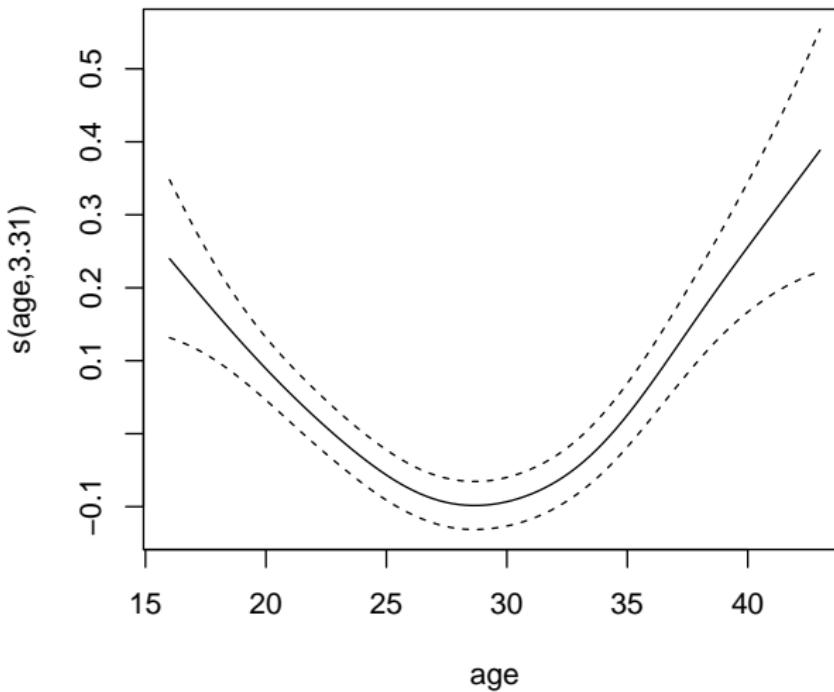
```
> gam.check(fit.gam)
```

```
Method: UBRE    Optimizer: outer newton
full convergence after 3 iterations.
Gradient range [9.816712095e-07,9.816712095e-07]
(score -0.4209494806 & scale 1).
Hessian positive definite, eigenvalue range [1.285936713e-05,1.285936713e-05].
Model rank = 12 / 12
```

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(age)	9.00	3.31	0.94
			0.66

Generalized Additive Model



Bayesian credible intervals

We are assuming the underlying function is smooth. We can formalize this as a prior in a Bayesian model. We won't get into details, but may return to this topic in M7.

In Bayesian statistics, the parameters β are random variables. A ridge penalty corresponds to an improper Gaussian prior:

$$f_{\beta} \propto \exp(-\beta' \mathbf{B} \beta / 2).$$

For Gaussian data, this results in a posterior distribution

$$[\beta | \mathbf{y}, \lambda] \sim N \left\{ \hat{\beta}, \sigma^2 (\mathbf{X}' \mathbf{X} + \lambda \mathbf{B})^{-1} \right\}$$

For a general likelihood, we use the Fisher Information matrix (Hessian of the negative log likelihood at $\hat{\beta}$)

$$[\beta | \mathbf{y}, \lambda] \sim N \left\{ \hat{\beta}, \left(\hat{\mathcal{I}} + \lambda \mathbf{B} \right)^{-1} \right\}$$

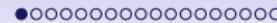
Multiply by ϕ for quasi-Poisson.

In particular, the "Bayesian credible intervals" plotted in `mgcv::gam` have Frequentist coverage probabilities. See Section 6.10 in Wood.

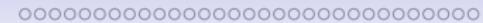
Generalized Additive Models



Multiple Smoothers



Bivariate Splines



Multiple Smoothers

Multiple Smooth Terms

Let's consider an additive model for two continuous variables, x_i and z_i :

$$y_i = \beta_0 + q_1(x_i) + q_2(z_i) + \epsilon_i, \quad \epsilon_i \stackrel{iid}{\sim} N(0, \sigma^2) \quad (1)$$

where $g_1(\cdot)$ and $g_2(\cdot)$ denote smooth relationships between the response y_i and predictors x_i and z_i .

Again, extends to generalized linear models (binomial, Poisson, etc).

We again express non-linear functions using basis functions:

$$g_1(x_i) = \sum_{m=1}^{M_1} \beta_{m1} b_{m1}(x_i) \quad g_2(z_i) = \sum_{m=1}^{M_2} \beta_{m2} b_{m2}(z_i)$$

Induce smoothing by penalizing regression coefficients.

Note we also use smoothers to control for **confounders** flexibly.

Associations between Mortality and Fine Particulate Matter

Fine particulate matter ($PM_{2.5}$):

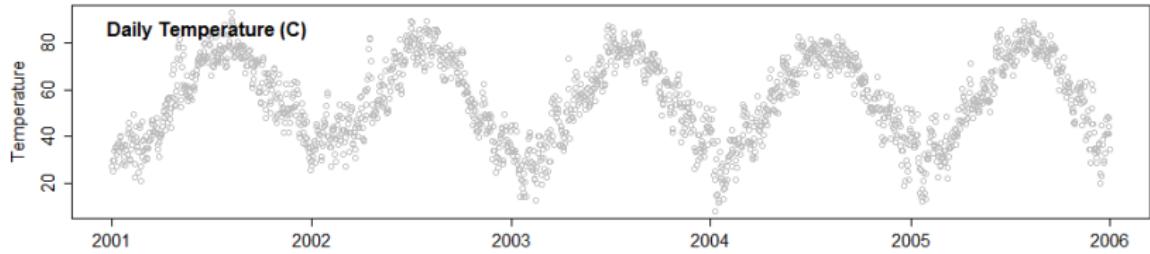
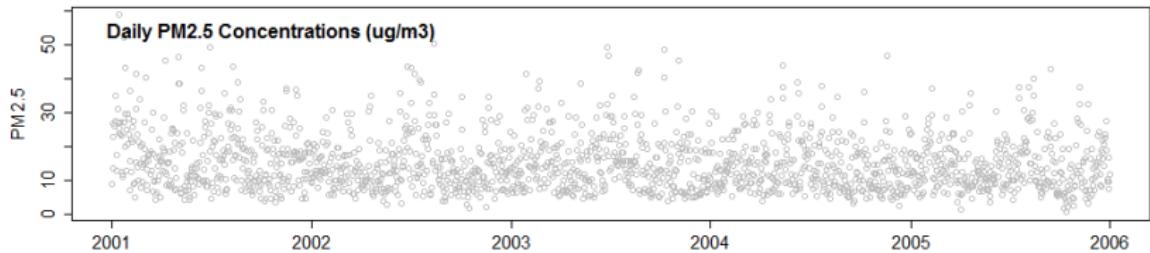
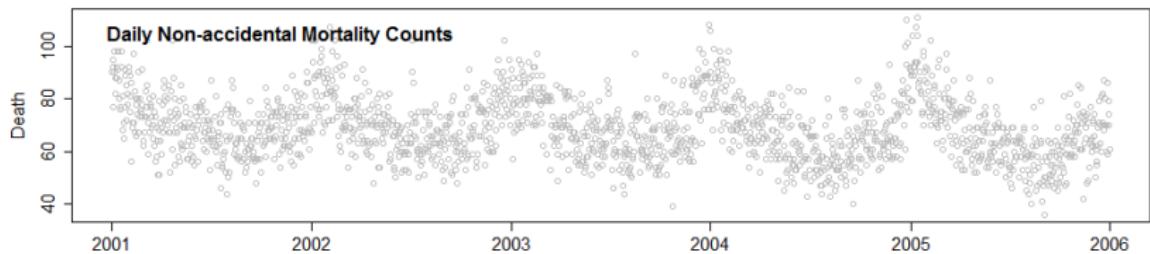
- represents a mixture of solid and liquid particles in the air that are less than $2.5 \mu m$ in diameter;
- mainly arises from combustion sources (power generation, vehicle, and industrial operations).

Scientific Question: what is the association between daily mortality counts and daily concentration of outdoor $PM_{2.5}$ air pollution?

Data Sources:

- Daily counts of non-accidental deaths (age ≥ 65) in the 5-county New York City area (2001-2005) obtained from the National Center for Health Statistics (CDC).
- Daily $PM_{2.5}$ concentrations from Environmental Protection Agency
- Daily meteorology conditions from the National Climatic Data Center (NOAA).

Associations between Mortality and Fine Particulate Matter



Time-Series Health Model

- We are interested in the association between **daily variation** in mortality counts and **daily variation** in exposure.
- In a time-series design we view population as the unit of analysis.
I.e. outcome = total mortality counts arising from the population.
- Confounders that vary smoothly in time can be easily controlled for by including smooths.

Time-Series Health Model

Let y_t denote the death count on day t , and x_t be the corresponding PM_{2.5} level.

There is typically a *temporal delay* between exposure and outcome.

Let's examine the association between daily mortality and **previous-day** exposure.

$$\log y_t = \beta_0 + g_1(x_{t-1}) + \text{confounders} + \epsilon_t, \quad \epsilon_t \stackrel{iid}{\sim} N(0, \sigma^2). \quad (2)$$

We are interested in $g_1(x_{t-1})$, the smooth of lagged PM_{2.5}. Here we model log-transformed death counts, which can improve normality of residuals.

Confounding factors to consider:

- Day of the week.
- Seasonality and long-term trends.
- Same-day temperature.
- Same-day dew point temperature.
- Previous day's temperature and dew point temperature.

Data

```
> load ("NYC.RData")
> str(health)
'data.frame': 1826 obs. of 12 variables:
 $ date      : Date, format: "2001-01-01" "2001-01-02" "2001-01-03" ...
 $ alldaydeaths : int 171 198 179 169 201 182 167 167 193 159 ...
 $ age65plus   : int 122 146 133 128 145 141 126 116 142 124 ...
 $ cardioresp: int 103 106 109 90 120 101 102 101 115 101 ...
 $ cr65plus    : int 90 92 95 77 98 90 88 82 92 88 ...
 $ dow        : chr "Monday" "Tuesday" "Wednesday" "Thursday" ...
 $ pm25       : num [1:1826, 1] 8.72 13.39 22.9 26.76 25.89 ...
 ..- attr(*, "dimnames")=List of 2
 ... $ : chr "1" "2" "3" "4" ...
 ... $ : NULL
 $ Temp       : num 27.6 25.1 25.3 29.8 29.8 33.9 34.9 35.7 32.3 27.6 ...
 $ DpTemp     : num 14.2 11.8 13.1 15.9 20.5 26.7 22.2 31.2 24.4 11.7 ...
 $ rmTemp     : num 27.7 26.8 26 26.7 28.3 ...
 $ rmDpTemp   : num 17.5 14.3 13 13.6 16.5 ...
```

- $DpTemp$ = Dew point temperature.
 - rm denotes 3-day running mean of the current day and 2 days prior.
 - dow (day of week) is recorded as character.

Lagged PM_{2.5} Exposure

```

> fit = gam(log(cr65plus)^s(pm25.lag1)+fdow+s(date2, k = 100 )+s(Temp)\\
+ s(DpTemp)+s(rmTemp)+s(rmDpTemp), data = health)
> summary(fit)

```

Family: gaussian

Link function: identity

Formula:

```
log(cr65plus) ~ s(pm25.lag1) + fdow + s(date2, k = 100) + s(Temp) +
  s(DpTemp) + s(rmTemp) + s(rmDpTemp)
```

Parametric coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	4.205748765	0.007731497	543.97601	< 2.22e-16 ***
fdowMonday	0.036176289	0.010980266	3.29466	0.0010051 **
fdowSaturday	0.004256722	0.010918297	0.38987	0.6966825

1

Signif. codes: 0 ‘****’ 0.001 ‘***’ 0.01 ‘**’ 0.05 ‘.’ 0.1 ‘ ’ 1

Lagged PM_{2.5} Cont.

Approximate significance of smooth terms:

	edf	Ref.df	F	p-value	
s(pm25.lag1)	1.000040	1.000080	5.50991	0.01902030	*
s(date2)	60.137293	72.669727	6.29239	< 2.22e-16	***
s(Temp)	1.000002	1.000004	14.33175	0.00015864	***
s(DpTemp)	1.446148	1.787928	2.05017	0.09521592	.
s(rmTemp)	5.826145	7.051150	2.86630	0.00532417	**
s(rmDpTemp)	1.000004	1.000007	9.45312	0.00214032	**

Diagnostics

```
> gam.check(fit)
```

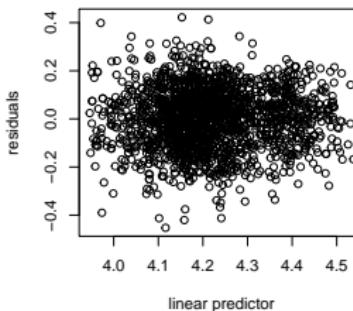
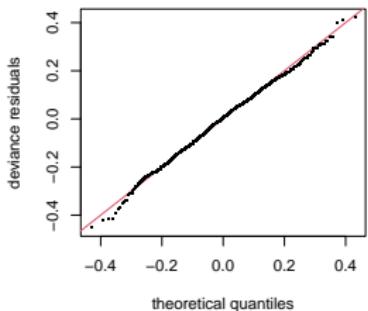
```
Method: GCV  Optimizer: magic
Smoothing parameter selection converged after 17 iterations.
The RMS GCV score gradient at convergence was 3.105686e-08 .
The Hessian was positive definite.
Model rank = 151 / 151
```

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'.

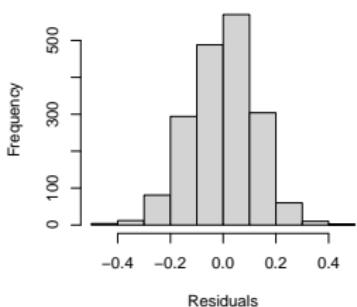
	k'	edf	k-index	p-value
s(pm25.lag1)	9.00	1.00	1.04	0.95
s(date2)	99.00	60.14	0.99	0.34
s(Temp)	9.00	1.00	1.00	0.45
s(DpTemp)	9.00	1.45	1.00	0.39
s(rmTemp)	9.00	5.83	1.04	0.94
s(rmDpTemp)	9.00	1.00	1.01	0.69

Model diagnostics

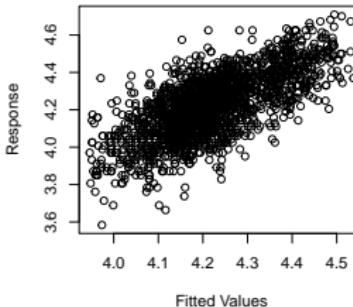
Resids vs. linear pred.



Histogram of residuals



Response vs. Fitted Values

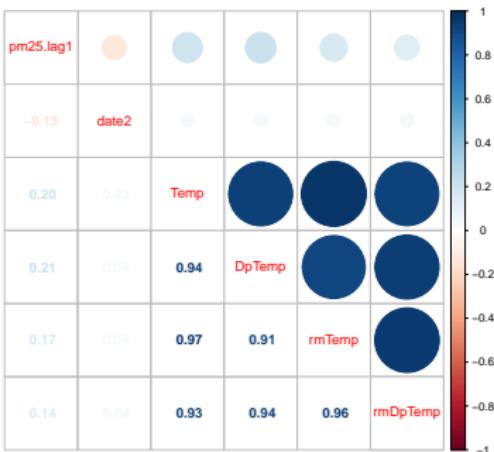


Correlations and VIF

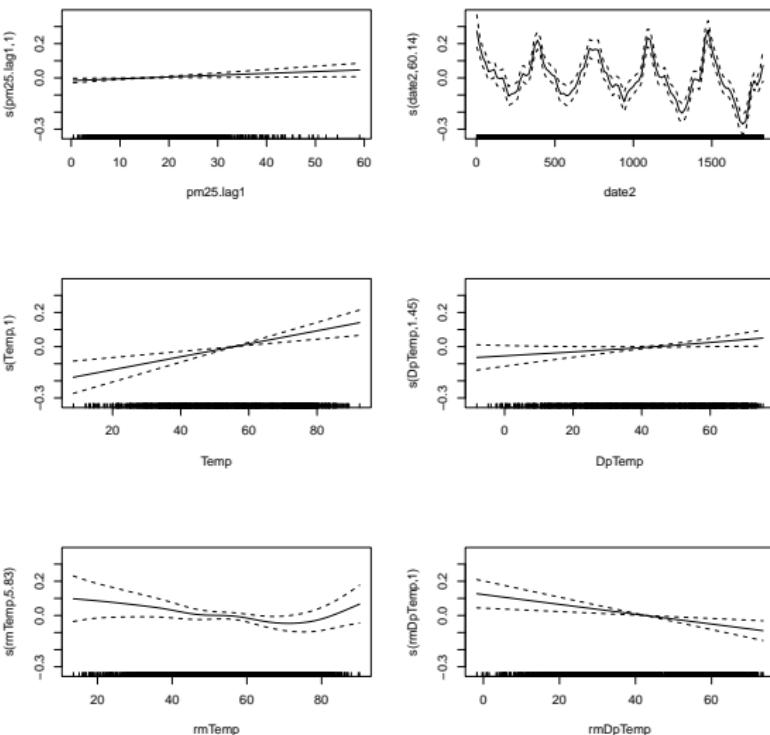
It is always good to check the correlations between variables.

VIFs are more complicated due to expanding covariates with a spline basis.

Here, we will be concerned if there are high correlations with PM2.5.lag1, which is our variable of scientific interest.



Interpretation



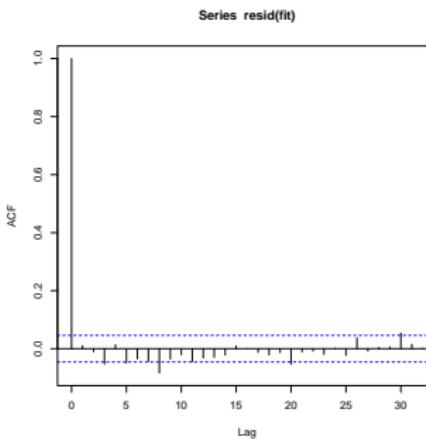
Note on time series data

We should analyze the residuals for evidence of autocorrelation.

If residuals are correlated, e.g., nearby residuals more similar than distant residuals, inference is not valid, and p-values tend to be too small.

ACF: $\text{corr}(r_t, r_{t-l})$

```
acf(resid(fit))
```



Interpretation of model

- Daily counts of non-accidental mortality were positively associated with previous-day PM_{2.5} level among those 65 or above.
- Approximately linear on log scale.

```
> newd <- health[1, ] # grab any row
> newd$pm25.lag1 <- 15 - 1e-05 # subtract some small number
> y1 <- predict(fit, newd)
> newd$pm25.lag1 <- 15 + 1e-05 # add some small number
> y2 <- predict(fit, newd)
> (y2 - y1)/2e-05
      1
0.001054248
```

- We estimated a 0.00105 increase in log death count per unit ($\mu\text{g}/\text{m}^3$) increase in PM_{2.5} level in the previous day.
- 10 unit increase in PM_{2.5} level from the previous day increased daily death counts by $100 * (e^{(0.00105 \times 10)} - 1)\% = 1.06\%$.

Example of invalid model

To see the impact of autocorrelation, we remove date.

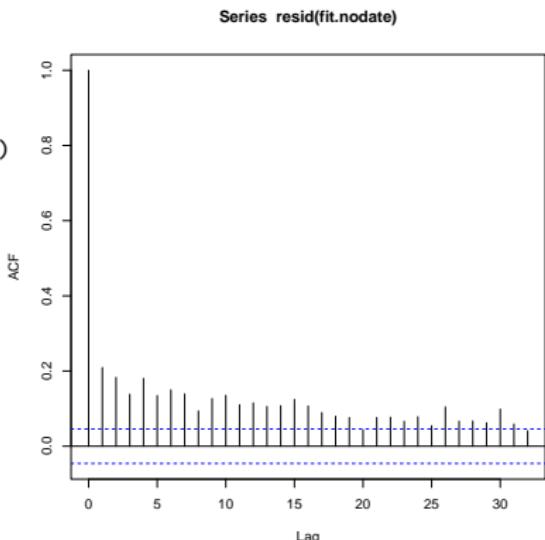
```

fit.nodate = gam(log(cr65plus)~s(pm25.lag1)
+fdow+s(Temp)+ s(DpTemp)
+s(rmTemp)+s(rmDpTemp). data = health)

```

Now, we no longer properly account for the dependence between nearby observations.

This leads to significant autocorrelation ($> 2 * 1/\sqrt{n}$)



Example of invalid model, continued

```
> summary(fit.nodate)
```

Family: gaussian

Link function: identity

Formula:

```
log(cr65plus) ~ s(pm25.lag1) + fdow + s(Temp) + s(DpTemp) + s(rmTemp) +
  s(rmDpTemp)
```

Parametric coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	4.2196867	0.0086125	489.950	<2e-16 ***
fdowFriday	-0.0149662	0.0121799	-1.229	0.2193
...				

Approximate significance of smooth terms:

	edf	Ref.df	F	p-value
s(pm25.lag1)	1.000	1.000	47.599	<2e-16 ***
s(Temp)	1.683	2.177	4.480	0.0103 *
s(DpTemp)	1.000	1.000	1.406	0.2359
s(rmTemp)	7.705	8.572	6.781	<2e-16 ***
s(rmDpTemp)	2.111	2.762	1.906	0.1233

Signif. codes: 0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

R-sq.(adj) = 0.337 Deviance explained = 34.4%
 GCV = 0.019388 Scale est. = 0.01917 n = 1825

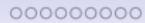
Autocorrelation in residuals

One person's mean structure is another person's correlation structure.

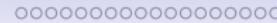
If you don't have covariates to model the correlation structure, you can specify correlated errors.

This is beyond the scope of this course, but a nice tutorial is available at
[https://petolau.github.io/
Analyzing-double-seasonal-time-series-with-GAM-in-R/](https://petolau.github.io/Analyzing-double-seasonal-time-series-with-GAM-in-R/)

Generalized Additive Models



Multiple Smoothers



Bivariate Splines



Bivariate Splines

Bivariate Splines

Under an additive model framework, we can also consider smooth effects of two variables jointly:

$$y_i = f(x_i, z_i) + \epsilon_i, \quad \epsilon_i \sim N(0, \sigma^2).$$

We can think of $f(x_i, z_i)$ as a **surface**.

We can use 2-dimensional splines to model $f(x_i, z_i)$.

Let $\mathbf{s}_i = (x_i, z_i)$ be some pair of covariate values, and let $\mathbf{k}_m = (x_m, z_m)$ denote the m^{th} knot in the domain of x_i and z_i . We can express the smooth function as

$$f(x_i, z_i) = \beta_0 + \sum_{m=1}^M \beta_m b_m(\mathbf{s}_i; \mathbf{k}_m).$$

Note that $b_m(\cdot, \cdot)$ is a basis function that maps $\mathbb{R} \times \mathbb{R} \rightarrow \mathbb{R}$.

Thin-plate Spline

One popular bivariate basis function uses [thin-plate splines](#), which extends to $\mathbf{s}_i \in \mathbb{R}^d$ and $\partial^l g$ penalties. We consider $d = 2$ and $l = 2$:

$$f(\mathbf{s}_i) = \beta_0 + \beta_1 x_t + \beta_2 z_t + \sum_{m=1}^M \beta_{2+m} b_m(\mathbf{s}_i; \mathbf{k}_m)$$

using the radial basis:

$$b_m(\mathbf{s}_i; \mathbf{k}_m) = \|\mathbf{s}_i - \mathbf{k}_m\|^2 \log(\|\mathbf{s}_i - \mathbf{k}_m\|).$$

Here, $\|\mathbf{s}_i - \mathbf{k}_m\|$ is the Euclidean distance between the covariate \mathbf{s}_i and the knot location \mathbf{k}_m .

The radial basis kernel is $r^2 \log r$.

The thin-plate spline is sensitive to the scale of each variable, but invariant to rotation (isotropic).

It is best for variables measured on the same scale (e.g. geographical distance).

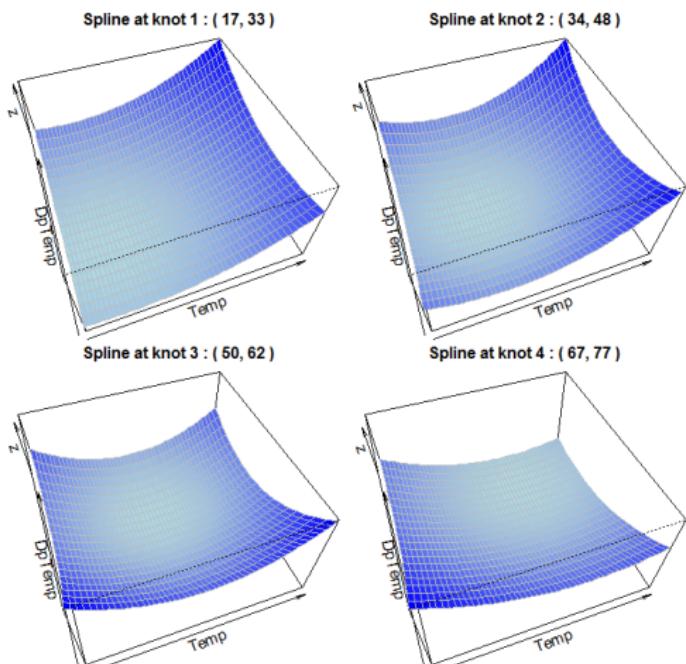
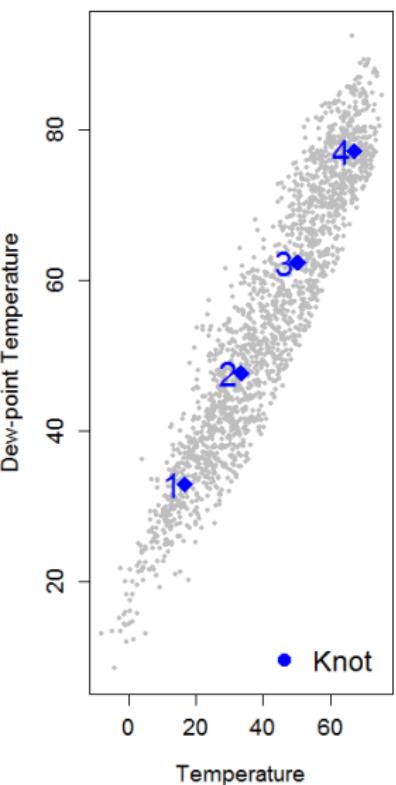
Thin-plate Spline, cont.

It can be shown that the thin-plate spline function minimizes

$$\sum_{i=1}^n \{y_i - f(x_i, z_i)\}^2 + \lambda \int \left(\frac{\partial^2 f(x, z)}{\partial x^2} \right)^2 + 2 \left(\frac{\partial^2 f(x, z)}{\partial x \partial z} \right)^2 + \left(\frac{\partial^2 f(x, z)}{\partial z^2} \right)^2 dx dz .$$

More information is in Wood 2017, pages 215-221, and references therein.

Thin-plate Spline



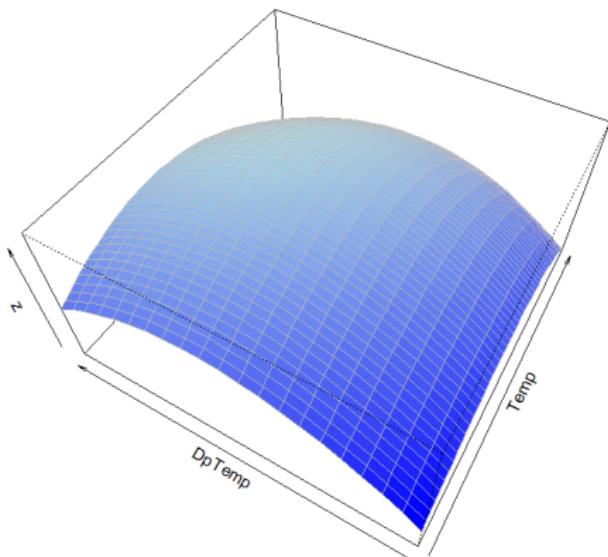
Generalized Additive Models
oooooooooooo

Multiple Smoothers
oooooooooooooooooooo

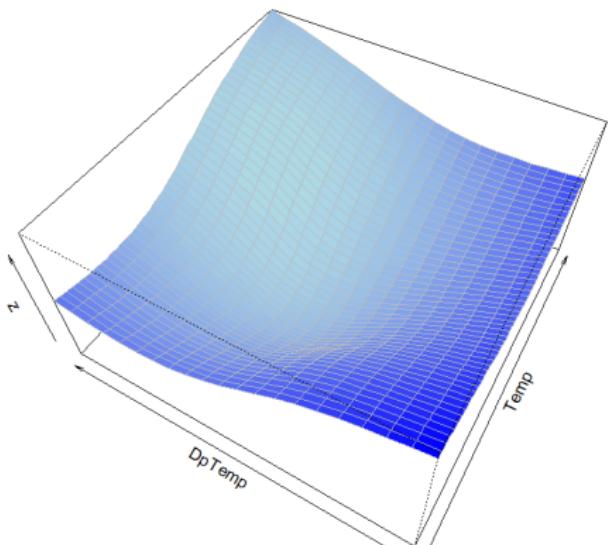
Bivariate Splines
oooooooo●oooooooooooooooooooo

Thin-plate Spline

Basis coefficients = [-2,1,1,-2]



Basis coefficients = [1,-1,-1,1]



Thin-plate regression spline

`mgcv::gam` uses thin-plate **regression** splines as the default for smoothers of a single variable (as well as two variables).

This is implemented in a ‘knot-free’ manner.

This is the general idea.

For $d = 2, l = 2$:

1. Construct the $n \times n$ matrix \mathbf{E} from $\|\mathbf{s}_i - \mathbf{s}_{i'}\|^2 \log(\|\mathbf{s}_i - \mathbf{s}_{i'}\|)$.
2. Use the singular value decomposition to find a low rank representation, e.g., k leading singular vectors, and use this in place of \mathbf{X} in the penalized objective function.
3. In practice, there are some additional things to worry about to make $\mathbf{1}$ (for the intercept), \mathbf{x} and \mathbf{z} in the null space of the penalty.
4. Then estimate the β_0, β_1 for \mathbf{x} , β_2 for \mathbf{z} (unpenalized) and β_3, \dots, β_k (penalized), which dramatically reduces computation costs.

The formulas for the general case (d dimensions and l th derivative) get a bit complicated; see 5.5 in Wood.

Joint Effects of Temperature and Dew point Temperature

To define a bivariate smoother, simply specify $s(var_1, var_2)$ in the equation formula.

Default in `mgcv:gam` is to use a thin-plate regression spline.

For this demonstration, let's look at the joint effects of same-day temperature and dew point temperature on log mortality, controlling only for time trends.

```
> fit1 = gam(log(alldeaths)~s(date2, k=100) + s(Temp, DpTemp, k = 20), data = health)
> summary (fit1)
```

Parametric coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	4.219108	0.002953	1429	<2e-16 ***

Approximate significance of smooth terms:

	edf	Ref.df	F	p-value
s(date2)	69.527	82.144	8.501	< 2e-16 ***
s(Temp,DpTemp)	5.938	8.032	6.407	2.96e-08 ***

R-sq.(adj) = 0.45 Deviance explained = 47.3%

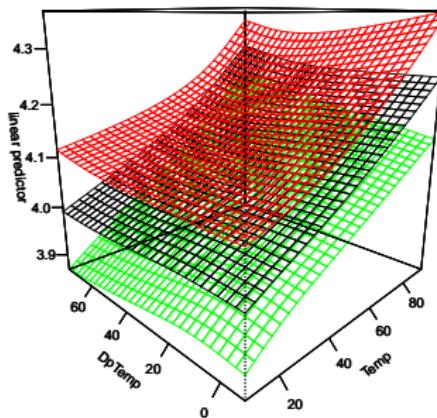
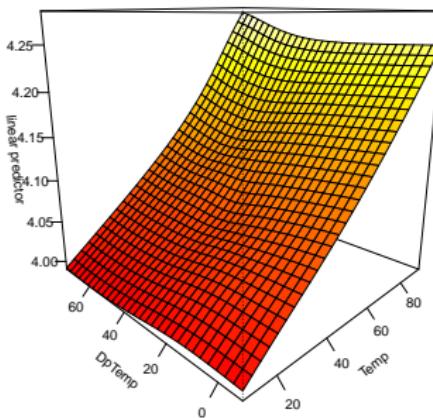
GCV score = 0.016624 Scale est. = 0.015928 n = 1826

You should also use `gam.check` (note shown here)



Joint Effects of Temperature and Dew Point Temperature

Perspective Plots from Thin-Plate Spline



red/green are \pm 2 s.e.

Tensor Product

Another way to obtain a 2-D spline is by construction. First consider the effect of a variable x_i specified by M basis functions $b_m(x_i)$

$$f(x_i) = \sum_{m=0}^M \beta_m b_{m,x}(x_i).$$

Now assume $f(x_i, z_i)$ is created by allowing each spline coefficient to vary with the second variable z_i :

$$f(x_i, z_i) = \sum_{m=0}^M \beta_m(z_i) b_{m,x}(x_i).$$

We can express $\beta_m(z_i)$ also as a smooth function of z_i using N basis functions $b_{n,z}(z_i)$:

$$f(x_i, z_i) = \sum_{m=0}^M \sum_{n=0}^N \alpha_{m,n} b_{n,z}(z_i) b_{m,x}(x_i).$$

This is equivalent to expressing $f(x_i, z_i)$ as all pairwise basis functions of x_i and z_i :

$$f(x_i, z_i) = \sum_{m=0}^M \sum_{n=0}^N \alpha_{m,n} b_{n,z}(z_i) b_{m,x}(x_i).$$

Generalized Additive Models
oooooooooooo

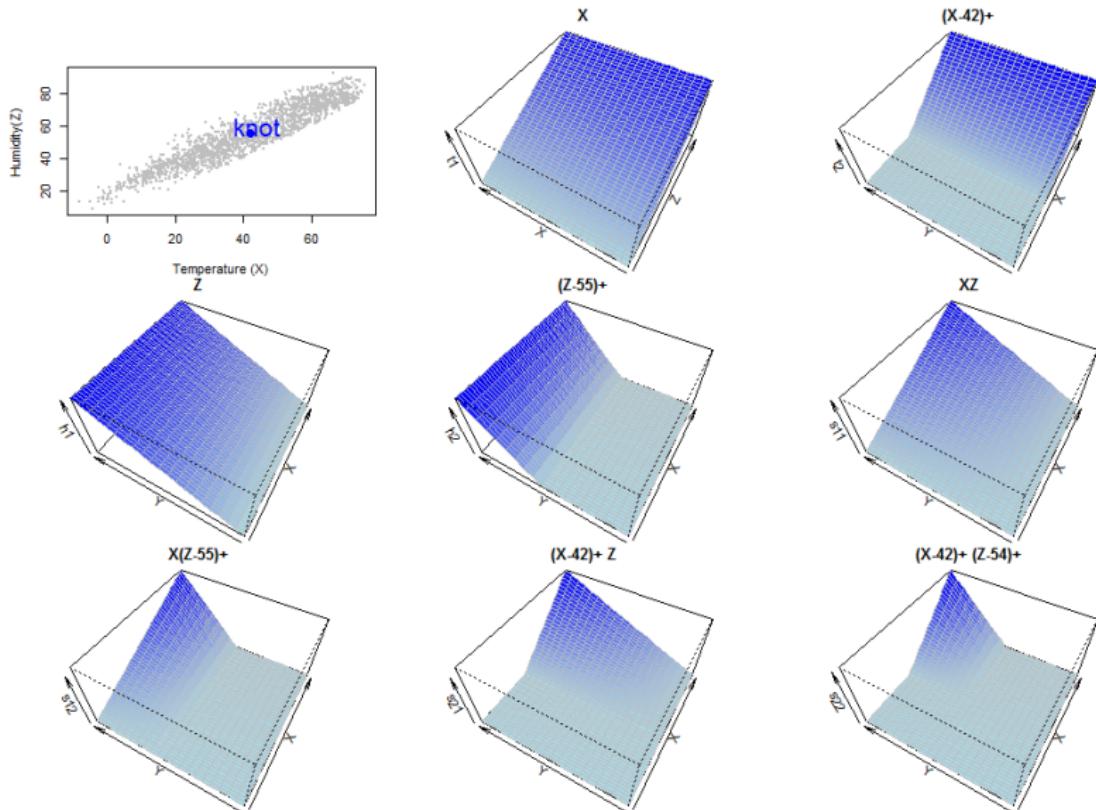
Multiple Smoothers
oooooooooooooooooooo

Bivariate Splines
oooooooooooo●oooooooooooooooooooo

Tensor Product

Example with 1 knot: write out all pairwise interactions

Tensor Product Example



Tensor Product

The roughness penalty optimized by the tensor product is

$$\sum_{i=1}^n \{y_i - f(x_i, z_i)\}^2 + \int \lambda_x \left(\frac{\partial^2 f(x, z)}{\partial x^2} \right)^2 + \lambda_z \left(\frac{\partial^2 f(x, z)}{\partial z^2} \right)^2 dx dz .$$

- Allows penalization in each variable dimension by assigning a smoothing parameter for each marginal smooth effect.
- Choice of basis function and knots also do not need to be the same for each covariate.
- Provides a recipe for constructing flexible multivariate joint effects.
- Often used for modeling interactions where the degree of smoothness may not be the same for all covariates.

Fitting Tensor Product

A bivariate smoother using tensor product is specified by `te(var_1, var_2, bs='cr')` in the equation formula.

Recommended when scales differ.

The option `bs = 'cr'` specifies using cubic spline functions for each covariate .

```
> fit1 = gam (log(alldeaths) ~ s(date2, k = 100) +
+              te(Temp, DpTemp, k = 20, bs = "cr") , data = health)

> summary (fit1)

Parametric coefficients:
                        Estimate Std. Error t value Pr(>|t|)
(Intercept) 4.219108    0.002952    1429    <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

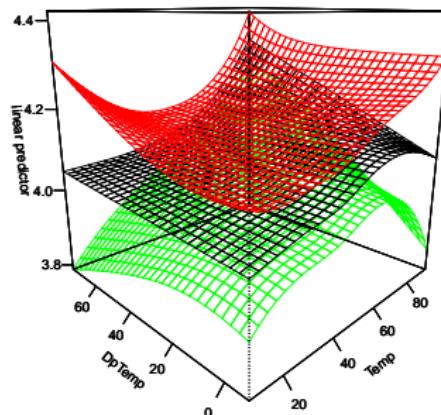
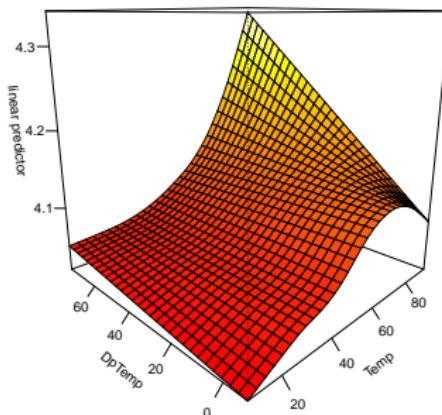
Approximate significance of smooth terms:
            edf Ref.df      F p-value
s(date2)     68.353 81.039  8.343 <2e-16 ***
te(Temp,DpTemp) 6.003  7.391 18.134 <2e-16 ***

R-sq.(adj) =  0.45   Deviance explained = 47.3%
GCV score = 0.016601  Scale est. = 0.015916 n = 1826
```



Joint Effects of Temperature and Dew Point Temperature

Perspective Plots from Cubic Tensor Product Spline

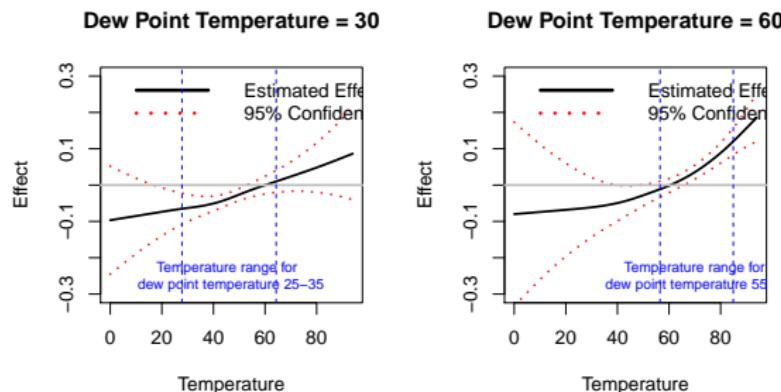


red/green are +/- 2 s.e.

Marginal Effects

By fitting an effect surface, we can obtain **slices** with respect to a covariate and examine the **conditional** effect of another covariate.

For example, we can look at the non-linear effect of temperature when dew point temperature = 30 versus 60.



PM_{2.5} and Temperature

Let's revisit the PM_{2.5} analysis. An important research question is whether there is an interaction between air pollution and temperature.

For this demonstration, we consider previous-day PM_{2.5} level and 3-day moving average of temperature (rmTemp).

We use thin-plate splines, which are much faster to fit than tensor; pm25.lag1 and rmTemp are very roughly on same scale.

Bivariate smooth for rmTemp and PM2.5

```
> fit = gam(log(alldeaths)~s(pm25.lag1, rmTemp)+factor(dow)+s(date2, k = 75)+  
           s(DpTemp)+s(rmDpTemp), data = health)  
> summary (fit)
```

Parametric coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	4.2191876	0.0077934	541.380	<2e-16 ***
factor(dow)Friday	-0.0142658	0.0110132	-1.295	0.1954
factor(dow)Monday	0.0222996	0.0110146	2.025	0.0431 *
factor(dow)Saturday	-0.0092121	0.0110082	-0.837	0.4028
factor(dow)Thursday	0.0011288	0.0110248	0.102	0.9185

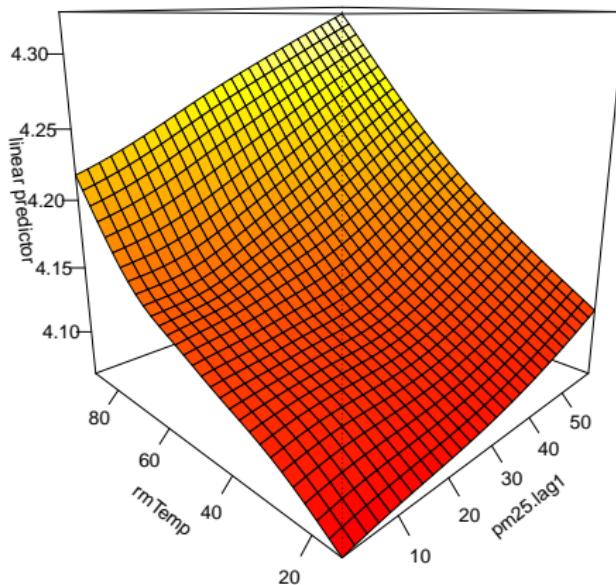
Approximate significance of smooth terms:

	edf	Ref.df	F	p-value
s(pm25.lag1,rmTemp)	5.858	8.381	4.048	6.35e-05 ***
s(date2)	57.808	66.508	6.721	< 2e-16 ***
s(DpTemp)	1.362	1.633	25.075	8.11e-10 ***
s(rmDpTemp)	1.000	1.000	35.085	3.79e-09 ***

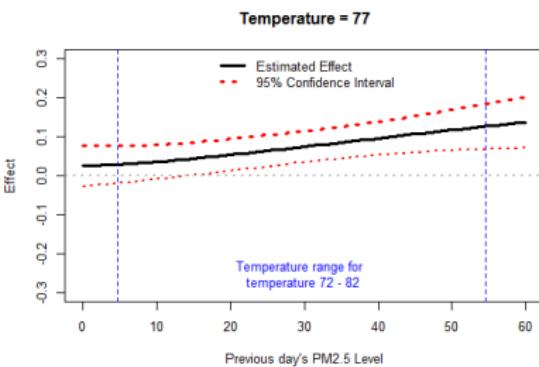
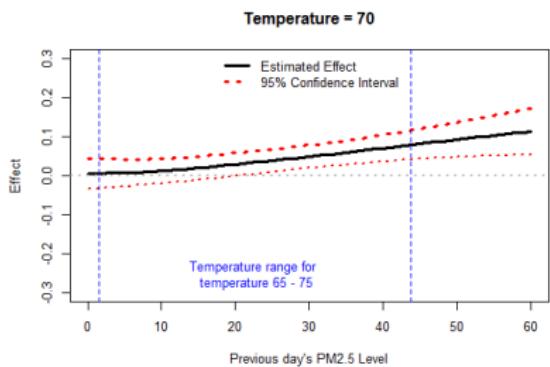
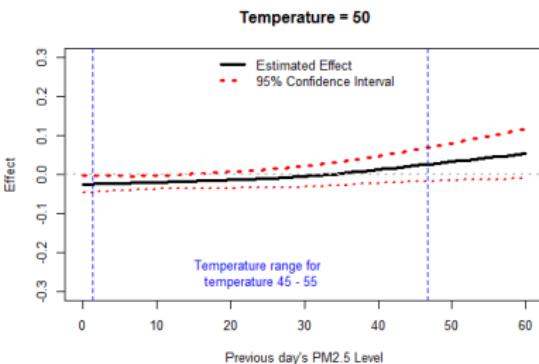
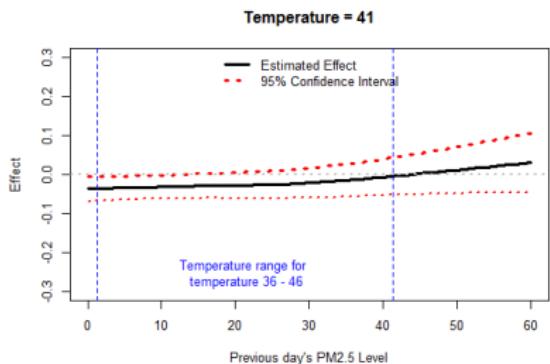
R-sq.(adj) = 0.456 Deviance explained = 47.8%
 GCV score = 0.016379 Scale est. = 0.015724 n = 1825

Bivariate smooth for rmTemp and PM_{2.5}

Effect of temperature is more pronounced than PM_{2.5}.



PM_{2.5} Effect Modification by Temperature?



Extracting Effects from Smooth Function

- Let's say we are interested in the difference in $f(x_i)$ when x_i increases from value a to value b :

$$\hat{f}(b) - \hat{f}(a) = \sum_{m=1}^M \hat{\beta}_m b_m(b) - \sum_{m=1}^M \hat{\beta}_m b_m(a) = \sum_{m=1}^M \hat{\beta}_m \{b_m(b) - b_m(a)\}.$$

The covariance matrix is given by

$Cov(\mathbf{B}'_{b-a}\hat{\beta}) = \mathbf{B}'_{b-a}Cov(\hat{\beta})\mathbf{B}_{b-a}$, where

$$\mathbf{B}_{b-a} = [b_1(b) - b_1(a), b_2(b) - b_2(a), \dots, b_M(b) - b_M(a)]'.$$

- However $b_m()$'s are basis functions and $b_m(b) - b_m(a) \neq b_m(b - a)$. We will need to construct the new covariate vector \mathbf{B} ourselves.

Extracting Effects from a Smooth Function

We want to estimate the effect of a $10 \mu\text{g}/\text{m}^3$ increase in PM_{2.5} levels from 40 to $50 \mu\text{g}/\text{m}^3$ when temperature = 70F.

```
> X1 = predict(fit, data.frame(pm25.lag1 = 40, rmTemp=70,
+                               DpTemp=0, rmDpTemp = 0, dow = "Sunday", date2=0),type= "lpmatrix")
> X2 = predict(fit, data.frame(pm25.lag1 = 50, rmTemp=70,
+                               DpTemp=0, rmDpTemp = 0, dow = "Sunday", date2=0),type= "lpmatrix")

> X.diff = X2 - X1
> dim (X.diff)
[1] 1 128
> Est = X.diff %*% coef (fit) ## Estimate
> se = sqrt( X.diff %*% vcov (fit) %*% t(X.diff) ) ## Standard Error
> Est; se
      [,1]
1 0.02213508
      1
1 0.01096592
```

So our estimate is 0.022 (95%CI 0.001, 0.044).

The same effect of a 10-unit increase in PM_{2.5} levels from 20 to $30 \mu\text{g}/\text{m}^3$ is 0.020 (95%CI 0.004, 0.036).

These two estimates are very similar because the PM_{2.5} effect appears to be quite linear.

Inference in models with smoothing splines

For parametric terms, the inference is identical to purely parametric model.

You can use the t-statistics from `summary(fit)`.

```
> fit.full = gam(log(cr65plus)~s(pm25.lag1, rmTemp)+fdow+s(date2, k = 100)+s(DpTemp)+s(rmDpTemp))  
> summary(fit.full)
```

Family: gaussian

Link function: identity

Formula:

```
log(cr65plus) ~ s(pm25.lag1, rmTemp) + fdow + s(date2, k = 100) +  
    s(DpTemp) + s(rmDpTemp)
```

Parametric coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	4.219093	0.007784	541.997	<2e-16 ***
fdowFriday	-0.014166	0.011001	-1.288	0.1980
fdowMonday	0.022352	0.011002	2.032	0.0423 *
fdowSaturday	-0.009111	0.010995	-0.829	0.4074
fdowThursday	0.001261	0.011012	0.114	0.9089
fdowTuesday	-0.001689	0.011002	-0.153	0.8780
fdowWednesday	0.000470	0.011012	0.043	0.9660

Signif. codes: 0 ?***? 0.001 ?**? 0.01 ?*? 0.05 ?.? 0.1 ? ? 1

Inference in models with smoothing splines

For overall effect of fdow, use anova:

```
> anova(fit.full)

Family: gaussian
Link function: identity

Formula:
log(cr65plus) ~ s(pm25.lag1, rmTemp) + fdow + s(date2, k = 100) +
  s(DpTemp) + s(rmDpTemp)

Parametric Terms:
  df      F p-value
fdow  6 2.154  0.0448
```

Inference in models with smoothing splines

For smoothed terms, the inference is approximate because the distribution of the test statistics is impacted by the penalization.

The idea is to jointly test the significance of the $\hat{\beta}_j$ for the coefficients corresponding to the spline of the j th smooth term.

There is some discussion of these approximate p-values in
`?summary.gam`. For detailed discussion, see Wood 2017 p.304.

Approximate inference for smooth effects:

Approximate significance of smooth terms:

	edf	Ref.df	F	p-value	
s(pm25.lag1,rmTemp)	5.433	7.738	4.246	6.30e-05	***
s(date2)	62.047	74.627	6.118	< 2e-16	***
s(DpTemp)	1.210	1.379	29.260	2.89e-09	***
s(rmDpTemp)	1.000	1.000	34.655	4.68e-09	***

R-sq.(adj) = 0.458 Deviance explained = 48%

GCV = 0.016377 Scale est. = 0.015688 n = 1825

Inference in models with smoothing splines

Should we use a bivariate spline for pm25.lag1 and rmTemp, or two univariate splines? Don't do this:

```
> anova(fit.reduced2, fit.full, test='F')
Analysis of Deviance Table

Model 1: log(cr65plus) ~ s(pm25.lag1) + s(rmTemp) + s(date2, k = 100) +
  fdow + s(DpTemp) + s(rmDpTemp)
Model 2: log(cr65plus) ~ s(pm25.lag1, rmTemp) + fdow + s(date2, k = 100) +
  s(DpTemp) + s(rmDpTemp)
Resid. Df Resid. Dev          Df     Deviance F Pr(>F)
1 1734.9095 27.353616
2 1733.2552 27.428321 1.6542684 -0.074705162
```

What happened here?

Models are not nested.

Using tensor splines, we can construct in a special way to make nested. Then we can recast the problem as testing for an interaction.

Testing for an interaction

To test for an interaction between pm25.lag1 and rmTemp, we can construct the tensor spline in a special way such that the univariate splines are in the null space of the penalty of the tensor spline. See pages 243 and the example on page 343-346.

```
fit.full.tensor = gam(log(cr65plus)~s(pm25.lag1,bs='cr')+s(rmTemp,bs='cr')  
+ti(pm25.lag1, rmTemp,bs='cr')+fdow+s(date2, k = 100)+s(DpTemp)+s(rmDpTemp), data = health)  
summary(fit.full.tensor)  
anova(fit.full.tensor)
```

Testing for an interaction

```
> anova(fit.full.tensor)
```

Family: gaussian

Link function: identity

Formula:

```
log(cr65plus) ~ s(pm25.lag1, bs = "cr") + s(rmTemp, bs = "cr") +
  ti(pm25.lag1, rmTemp, bs = "cr") + fdow + s(date2, k = 100) +
  s(DpTemp) + s(rmDpTemp)
```

Parametric Terms:

df	F	p-value
fdow	6	2.24
		0.0371

Approximate significance of smooth terms:

	edf	Ref.df	F	p-value
s(pm25.lag1)	1.000	1.001	4.147	0.0418
s(rmTemp)	5.301	6.480	2.524	0.0171
ti(pm25.lag1,rmTemp)	1.000	1.000	3.463	0.0629
s(date2)	59.917	72.436	6.143	<2e-16
s(DpTemp)	1.000	1.000	41.860	<2e-16
s(rmDpTemp)	1.000	1.000	31.651	<2e-16

$H_0: f_{xz}(x_i, z_i)$ does not improve model fit over $f_x(x_i)$ and $f_z(z_i)$.
 p=0.06.

Inference in models with smoothing splines

Let's refit the model with pm25.lag1 as a linear term.

```
> fit.reduced3 = gam(log(cr65plus)~pm25.lag1+s(rmTemp)
+ s(date2, k = 100)+fdow+s(DpTemp)+s(rmDpTemp), data = health)
> summary(fit.reduced3)
```

Family: gaussian

Link function: identity

Formula:

```
log(cr65plus) ~ pm25.lag1 + s(rmTemp) + s(date2, k = 100) + fdow +
  s(DpTemp) + s(rmDpTemp)
```

Parametric coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	4.203e+00	1.018e-02	412.689	<2e-16 ***
pm25.lag1	1.073e-03	4.508e-04	2.381	0.0174 *
fdowFriday	-1.478e-02	1.099e-02	-1.345	0.1789
fdowMonday	2.249e-02	1.098e-02	2.048	0.0407 *
fdowSaturday	-9.026e-03	1.098e-02	-0.822	0.4110
fdowThursday	5.425e-04	1.100e-02	0.049	0.9607
fdowTuesday	-1.433e-03	1.098e-02	-0.130	0.8962
fdowWednesday	9.336e-05	1.100e-02	0.008	0.9932

...

R-sq.(adj) = 0.459 Deviance explained = 48.2%

GCV = 0.016324 Scale est. = 0.015642 n = 1825

GAM or linear?

It may seem like we can test for whether or not to include a smooth term versus linear term using anova. However, the anova test can produce funny results when the EDF is approximately one, as the change in DF is very small. So, I suggest not doing this:

```
> anova(fit.reduced2,fit.reduced3,test='F')
Analysis of Deviance Table

Model 1: log(cr65plus) ~ s(pm25.lag1) + s(rmTemp) + s(date2, k = 100) +
  fdow + s(DpTemp) + s(rmDpTemp)
Model 2: log(cr65plus) ~ pm25.lag1 + s(rmTemp) + s(date2, k = 100) + fdow +
  s(DpTemp) + s(rmDpTemp)
Resid. Df Resid. Dev      Df   Deviance      F Pr(>F)
1     1734.9    27.354
2     1734.9    27.353 0.016157 0.00029882 1.1824 0.0325 *
---
Signif. codes:  0 ?***? 0.001 ?**? 0.01 ?*? 0.05 ?.? 0.1 ? ? 1
```

GAM or linear?

When you have an effect that is close to linear, your interpretation is very similar whether or not you use a gam, and when EDF=1, then the approximate p values are equivalent to refitting with a linear term.

To determine whether or not to include a smooth or linear effect, I suggest looking at the EDF. If it is greater than 1, than it seems reasonable to use a smooth term.

There is not really any disadvantage to modeling with a smooth term, except for some extra work we need to do for interpretation.

One approach to test for a non-linear effect is to construct a special spline that separates the linear and non-linear parts:

[https://stats.stackexchange.com/questions/449641/
is-there-a-hypothesis-test-that-tells-us-whether-we-should-use-](https://stats.stackexchange.com/questions/449641/is-there-a-hypothesis-test-that-tells-us-whether-we-should-use-)