STAT 705 Nonlinear regression via basis expansions

Timothy Hanson

Department of Statistics, University of South Carolina

Stat 705: Data Analysis II

Nonlinear regression

We have discussed parametric nonlinear models as well as completely nonparametric approaches to estimating $f(\cdot)$ in

$$Y_i = f(x_i) + \epsilon_i, \quad E(\epsilon_i) = 0, \quad corr(\epsilon_i, \epsilon_j) = \delta_{ij}.$$

Now we will discuss another approach to estimating $f(\cdot)$, assuming that $f(\cdot)$ belongs to a class of appropriately smooth functions.

This approach is termed *basis expansions*.

Ethanol data, R help file

Ethanol fuel was burned in a single-cylinder engine. For various settings of the engine compression and equivalence ratio, the emissions of nitrogen oxides were recorded. Specifically, $n = 88$ observations on

- NOx: Concentration of nitrogen oxides (NO and NO2) in micrograms/J.
- C: Compression ratio of the engine.
- E: Equivalence ratio – a measure of the richness of the air and ethanol fuel mixture.

Brinkman, N.D. (1981) Ethanol FuelA Single-Cylinder Engine Study of Efficiency and Exhaust Emissions. *SAE transactions*, 90, 14101424.

We will examine how NOx varies with the equivalence ratio E.

Ethanol data w/ LOWESS fit

```
library(lattice)
data(ethanol)
attach(ethanol)
plot(ethanol)
a=min(E); b=max(E)
x=seq(a,b,length=100)
n=length(E)

fit=loess(NOx~E)
pred=predict(fit,x,se=TRUE)
plot(x,pred$fit,type="l",xlab="equivalence ratio",ylab="NOx",main="Lowess Fit")
lines(x,pred$fit-1.96*pred$se.fit,lty=3)
lines(x,pred$fit+1.96*pred$se.fit,lty=3)
points(E,NOx)
```

Weierstrauss approximation theorem

$\boxed{\text{Weierstrauss approximation theorem}}$: Let $f : [a, b] \to \mathbb{R}$ be a continuous function. Then for each $\epsilon > 0$, there exists a polynomial $p_\epsilon : [a, b] \to \mathbb{R}$ such that $|f(x) - p_\epsilon(x)| < \epsilon$ for all $x \in [a, b]$.

How is this different than Taylor's theorem? Weierstrauss gives *uniform* convergence, not just pointwise, and Taylor's theorem requires more smoothness, e.g. at least one derivative.

Basically says that any function $f(\cdot)$, even rough ones, can be approximated well by smooth functions.

One such approximation is through the use of *basis splines*, or B-splines for short. A B-spline is a a function defined by polynomials over intervals that partition $[a, b]$, and can be written

$$f(x) = \sum_{j=1}^{J} \beta_j B_j(x),$$

where $B_j(x)$ are known, fixed basis functions.

Basis "mother"

A B-spline is a linear combination of basis functions. Quadratic B-spline basis function on $[0,3]$:

$$\phi(x) = \left\{ \begin{array}{ll} 0.5x^2 & 0 \leq x \leq 1 \\ 0.75 - (x-1.5)^2 & 1 \leq x \leq 2 \\ 0.5(3-x)^2 & 2 \leq x \leq 3 \\ 0 & \text{otherwise} \end{array} \right\}.$$

The basis functions $B_j(\cdot)$ are just shifted, shrunk/stretched versions of these.

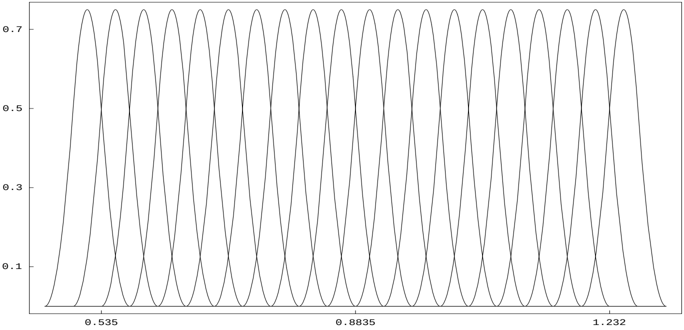
J basis functions

- Want J basis functions, typically $J = 20$.
- Without detail, jth basis function is

$$B_j(x) = \phi\left(\frac{x-a}{\Delta} + 3 - j\right), \quad \Delta = \frac{b-a}{J-2}.$$

- Here $a = \min\{x_1, x_2, \ldots, x_n\}$ and $b = \max\{x_1, x_2, \ldots, x_n\}$.
- So (a, b) is the range of the predictors in the data.
- For ethanol data, $x_i \in (0.535, 1.232)$.
- Next slide is $\{B_1(x), \ldots, B_{20}(x)\}$ for ethanol $x_i \in (0.535, 1.232)$.

$J = 20$ quadratic basis functions over $x_i \in (a, b)$



Here, x is the equivalence ratio E.

B-spline fit in R

```
library(splines)
X=bs(E,df=20,degree=2) # slightly different than what’s in the notes
Xp=bs(x,df=20,degree=2)

fit=lm(NOx~X-1) # fit model, create 95% CI’s
lo=x; hi=x
for(i in 1:100){
 lo[i]=Xp[i,]%*%fit$coef-qt(0.975,n-20)*Xp[i,]%*%summary(fit)$cov%*%Xp[i,]
 hi[i]=Xp[i,]%*%fit$coef+qt(0.975,n-20)*Xp[i,]%*%summary(fit)$cov%*%Xp[i,]
}

# fit B-spline model, display fitted regression line and 95% pointwise CI’s
par(mfrow=c(1,1))
plot(E,NOx,main="B-spline, J=20")
lines(x,Xp%*%fit$coef)
lines(x,lo,lty=3)
lines(x,hi,lty=3)
```

Penalized fitting

- This is an example of overfitting! One fix is to reduce J, say from 20 to 10.
- Another is to penalize the spline for being too "wiggly", but leave J at 20.
- Penalized least squares criterion is

$$\underbrace{\sum_{i=1}^{n}(y_i - f(x_i))^2}_{\text{makes } f(x_i) \text{ close to } y_i} + \underbrace{\lambda \int_a^b [f''(x)]^2 dx}_{\text{bigger } \lambda \Rightarrow \text{ less wiggly } f(x)} .$$

Penalized fitting

- Maximize

$$\mathcal{L}(\boldsymbol{\beta}) = 0.5n\log(\sigma^{-2}) - 0.5\sigma^{-2}\sum_{i=1}^{n}\left(y_i - \sum_{j=1}^{J}\beta_j\phi_j(x_i)\right)^2,$$

  subject to penalty

$$\int_a^b |f_j''(x)|^2 dx \le c.$$

- Equivalent to maximizing penalized log-likelihood (Eilers and Marx, 1996).
- Accomplished in `smooth.spline` in R, e.g. `lines(smooth.spline(E,NOx))`.

Fourier cosine representation theorem

For a continuous function $f : [a, b] \to \mathbb{R}$ such that $\int_a^b f(x)dx = 0$,

$$f(x) = \sum_{j=1}^{\infty} a_j \ \cos\left\{\frac{j\pi(x-a)}{b-a}\right\},$$

where

$$a_j = \frac{2}{b-a}\int_a^b f(x)\cos\left\{\frac{j\pi(x-a)}{b-a}\right\}.$$

Note that this representation implies $f'(a) = f'(b) = 0$.

Lenk (1999) and Efromovich (1999) propose Bayesian and frequentist models for $f(\cdot)$ with J terms

$$f(x) = \underbrace{\mu + \beta_0 x}_{\text{linear part}} + \underbrace{\sum_{j=1}^{J-2} \beta_j \cos\left\{\frac{j\pi(x-a)}{b-a}\right\}}_{\text{“wiggly” part integrates to zero}} .$$

A type of penalized likelihood can be used to shrink more oscillatory components toward zero.

Cosine expansion with $J = 12$ for ethanol data

```
# first four basis functions for NOx vs. E
par(mfrow=c(2,2))
plot(x,cos(1*pi*(x-a)/(b-a)),main="cosine basis j=1",type="l")
plot(x,cos(2*pi*(x-a)/(b-a)),main="cosine basis j=2",type="l")
plot(x,cos(3*pi*(x-a)/(b-a)),main="cosine basis j=3",type="l")
plot(x,cos(4*pi*(x-a)/(b-a)),main="cosine basis j=4",type="l")

# create design matrix and matrix for predictions
X=matrix(rep(0,n*12),ncol=12)
Xp=matrix(rep(0,100*12),ncol=12)
for(i in 1:n){
 X[i,1]=1; X[i,2]=E[i]
 for(j in 3:12){X[i,j]=cos((j-2)*pi*(E[i]-a)/(b-a))}
}
fit=lm(NOx~X-1) # fit model, create CI’s
lo=x; hi=x
for(i in 1:100){
 Xp[i,1]=1; Xp[i,2]=x[i]
 for(j in 3:12){Xp[i,j]=cos((j-2)*pi*(x[i]-a)/(b-a))}
 lo[i]=Xp[i,]%*%fit$coef-qt(0.975,n-12)*Xp[i,]%*%summary(fit)$cov%*%Xp[i,]
 hi[i]=Xp[i,]%*%fit$coef+qt(0.975,n-12)*Xp[i,]%*%summary(fit)$cov%*%Xp[i,]
}

# display fitted regression line and 95% pointwise CI’s
par(mfrow=c(1,1))
plot(E,NOx,main="Cosine basis expansion, J=12")
lines(x,Xp%*%fit$coef)
lines(x,lo,lty=3)
lines(x,hi,lty=3)
```

Comments

- Many other bases: Legendre polynomials, wavelets, other types of splines (e.g. thin plate), fractional polynomials, etc.
- Any basis results in simple linear model in the expansion coefficients. Often need to penalize coefficients in some way to get estimates that are not too “wiggly” and overfit the data.
- Lots of good notes out there on the web, e.g. Emily Fox at U. Washington.
- Can built additive models by considering separate expansion for each predictor; can also consider pairwise interaction *surfaces*, available in SAS PROC GAM. Penalized B-splines used in BayesX for Windows (free program for fitting GAMM's with spatial structure).
- Just meant to illustrate the basic idea here. Worthy of an entire course.