

7.1, 7.3, 7.4

Timothy Hanson

Department of Statistics, University of South Carolina

Stat 770: Categorical Data Analysis

## 7.1 Alternative links in binary regression\*

There are three common links considered in binary regression: logistic, probit, and complimentary log-log. All three are written

$$\pi(\mathbf{x}) = F(\mathbf{x}'\beta).$$

- Logistic regression:  $F(x) = \frac{e^x}{1+e^x}$ .
- Probit regression:  $F(x) = \Phi(x)$  where  $\Phi(x) = \int_{-\infty}^x \frac{e^{-0.5z^2}}{\sqrt{2\pi}} dz$ .
- Complimentary log-log binary regression:  
 $F(x) = 1 - \exp\{-\exp(x)\}$ .

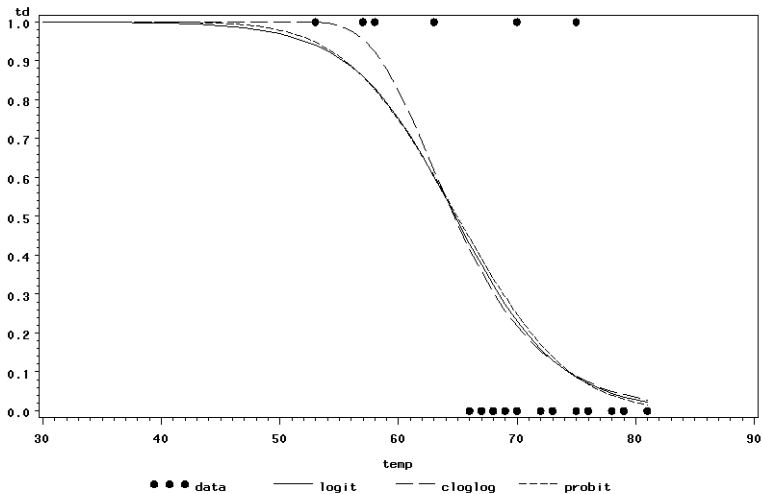
They differ primarily in the tails, but the logistic and probit links are symmetric in that rare and very common events are treated similarly in the tails. The CLL link approaches 1 faster than 0, so obtaining “rare event” status requires more extreme values of  $\mathbf{x}$  than reaching “likely event” status.

## O-ring data

Famous data set; your text has it in Table 5.14 (p. 199).  $N = 23$  space shuttle flights before the *Challenger* disaster in 1986. Temperature in Fahrenheit, and whether at least one primary O-ring suffered thermal distress.

```
data shut1; input temp td @@; datalines;
  66 0 70 1 69 0 68 0 67 0 72 0 73 0 70 0 57 1 63 1
  70 1 78 0 67 0 53 1 67 0 75 0 70 0 81 0 76 0 79 0
  75 1 76 0 58 1
;
data shut2;
  do i=1 to 50; temp=i+29; td=.; output; end;
data shut3; set shut1 shut2;
proc logistic descending data=shut3; model td = temp / link=logit;
  output out=shut4 p=p1;
proc logistic descending data=shut3; model td = temp / link=cloglog;
  output out=shut5 p=p2;
proc logistic descending data=shut3; model td = temp / link=probit;
  output out=shut6 p=p3;
data shut7; set shut4 shut5 shut6;
proc sort data=shut7; by temp;
goptions;
  symbol1 color=black value=dot interpol=none;
  symbol2 color=black value=none l=1 interpol=join;
  symbol3 color=black value=none l=2 interpol=join;
  symbol4 color=black value=none l=3 interpol=join;
  legend1 label=none value=('data' 'logit' 'cloglog' 'probit');
proc gplot data=shut7;
  plot td*temp p1*temp p2*temp p3*temp / overlay legend=legend1;
```

# Fits from different links



Statistic	logit	cloglog	probit
AIC	24.3	23.5	24.4
$\hat{\alpha}$	15.0	12.3	8.8
$\hat{\beta}$	-0.23	-0.20	-0.14

- Complimentary log-log chosen as “best” out of three according to AIC.
- Fitted cloglog model is

$$\hat{\pi}(\text{temp}) = 1 - \exp\{-\exp(12.3 - 0.2 \text{ temp})\}.$$

- H-L  $p$ -values are 0.21, 0.23, 0.22 respectively.

## 7.2 Bayesian logistic regression

The Bayesian approach allows the addition of information for  $\beta$  in the form of a prior. If no information is available, the prior can be uninformative. Conditional means priors allow incorporation of probability of success for different covariate values (Bedrick, Christensen, and Johnson, 1997).

Bayesian approaches typically do not depend on asymptotics so they're valid for small sample sizes.

Inference usually obtained through Markov chain Monte Carlo. Yields Monte Carlo estimates of inferences of interest (odds ratios, etc.)

In SAS, can add BAYES statement to PROC GENMOD. Example coming up where Bayes approach handles complete separation in data.

## 7.3 Exact conditional logistic regression

Pages 265–270 describe a method to obtain exact small-sample inference for regression parameters. The basic idea involves conditioning on sufficient statistics of parameters you don't care about. This was also done to obtain Monte Carlo p-values using EXACT in PROC FREQ (Chapter 3).

Exact conditional logistic regression is appropriate when the data are sparse, i.e. either  $\sum_{i=1}^N y_i$ , or  $\sum_{i=1}^N (n_i - y_i)$  is small.

Without loss of generality, assume we have two predictors  $x_1$  and  $x_2$ . The logistic regression likelihood looks like

$$P(Y_1 = y_1, \dots, Y_n = y_n) = \frac{\exp \left[ \beta_0 \sum_{i=1}^N y_i + \beta_1 \sum_{i=1}^N y_i x_{i1} + \beta_2 \sum_{i=1}^N y_i x_{i2} \right]}{\prod_{i=1}^N \left[ 1 + \exp \left( \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} \right) \right]}$$

The sufficient statistic for each  $\beta_j$  is  $T_j = \sum_{i=1}^N y_i x_{ij}$  where  $x_{i0} = 1$  for the intercept.

## Exact conditional logistic regression

The likelihood for  $\beta_2$ , conditional  $T_0 = t_0$  and  $T_1 = t_1$  is given by

$$P(Y_1 = y_1, \dots, Y_N = y_N | T_0 = t_0, T_1 = t_1) = \frac{\exp(t_2 \beta_2)}{\sum_{S(t_0, t_1)} \exp(t_2^* \beta_2)}$$

where  $S(t_0, t_1) =$

$\left\{ (y_1^*, \dots, y_N^*) : \sum_{i=1}^N y_i^* x_{i0} = t_0 \text{ and } \sum_{i=1}^N y_i^* x_{i1} = t_1 \right\}$ . This is maximized to give the conditional estimate  $\tilde{\beta}_2$ . Further inference (e.g. hypothesis testing) requires  $P(T_2 = t_2 | T_0 = t_0, T_1 = t_1)$ . This is given on p. 267 as (7.7). More details are in a document posted on the course webpage, if you are interested.

Instead of one effect, we may be interested in two or more effects. We simply condition on the remaining effects to obtain a conditional likelihood of two or more effects, similar to the above.



## Exact conditional logistic regression

To test one or more effects, add an EXACT statement in PROC LOGISTIC, followed by a list variables you want to test dropping from the model. Options include JOINT (dropping more than one effect and each effect separately), JOINTONLY, ESTIMATE (=PARAM, ODDS, or BOTH), ALPHA, and ONESIDED.

If your data are not sparse, be prepared to wait for days – a Bayesian approach might be better.

## 7.3.6 Example with quasi-complete separation and sparsity

```
data promote;
input race$ month$ promoted total @@;
datalines;
black july      0 7
black august   0 7
black september 0 8
white july     4 20
white august   4 17
white september 2 15
;

proc logistic data=promote; class race month;
model promoted/total=race month;
exact race / alpha=0.05 estimate=both onesided;

proc logistic data=promote; class race month;
model promoted/total=race month / firth;

proc genmod data=promote; class race month;
model promoted/total=race month / dist=binom link=logit;
bayes coefprior=jeffries;
run;
```

# 'Regular' logistic regression

## Model Convergence Status

Quasi-complete separation of data points detected.

WARNING: The maximum likelihood estimate may not exist.

WARNING: The LOGISTIC procedure continues in spite of the above warning. Results shown are based on the last maximum likelihood iteration. Validity of the model fit is questionable.

## The LOGISTIC Procedure

WARNING: The validity of the model fit is questionable.

## Testing Global Null Hypothesis: BETA=0

Test	Chi-Square	DF	Pr > ChiSq
Likelihood Ratio	8.2663	3	0.0408
Score	5.3825	3	0.1458
Wald	0.5379	3	0.9105

## Type 3 Analysis of Effects

Effect	DF	Wald	
		Chi-Square	Pr > ChiSq
race	1	0.0027	0.9583
month	2	0.5351	0.7652

## Analysis of Maximum Likelihood Estimates

Parameter	DF	Estimate	Standard	Wald	Pr > ChiSq
			Error	Chi-Square	
Intercept	1	-1.8718	0.7596	6.0730	0.0137
race black	1	-12.6091	241.4	0.0027	0.9583
month august	1	0.6931	0.9507	0.5316	0.4659
month july	1	0.4855	0.9431	0.2650	0.6067

# Conditional exact logistic regression

The LOGISTIC Procedure

Exact Conditional Analysis

Conditional Exact Tests

Effect	Test	Statistic	--- p-Value ---	
			Exact	Mid
race	Score	4.5906	0.0563	0.0434
	Probability	0.0257	0.0563	0.0434

Exact Parameter Estimates

Parameter	Estimate	Standard Error	One-sided 95% Confidence Limits	One-sided p-Value
race black	-1.8813*	.	-Infinity -0.2491	0.0257

NOTE: \* indicates a median unbiased estimate.

Exact Odds Ratios

Parameter	Estimate	One-sided 95% Confidence Limits	One-sided p-Value
race black	0.152*	0 0.779	0.0257

NOTE: \* indicates a median unbiased estimate.

Race is significant using the small-sample exact approach. Race is also significant using a Bayesian approach fit via MCMC (coming up).

# Bayesian approach using Jeffreys' prior (FIRTH)

Uses Jeffreys' prior, but inferenced based on normal approximation.

Testing Global Null Hypothesis: BETA=0

Test	Chi-Square	DF	Pr > ChiSq
Likelihood Ratio	5.6209	3	0.1316
Score	4.4120	3	0.2203
Wald	3.1504	3	0.3690

Type 3 Analysis of Effects

Effect	DF	Wald Chi-Square	Pr > ChiSq
race	1	2.6869	0.1012
month	2	0.4464	0.7999

Analysis of Maximum Likelihood Estimates

Parameter	DF	Estimate	Standard Error	Wald Chi-Square	Pr > ChiSq
Intercept	1	-1.6891	0.6946	5.9133	0.0150
race	black	-2.3491	1.4331	2.6869	0.1012
month	august	0.5850	0.8770	0.4449	0.5047
month	july	0.3867	0.8703	0.1975	0.6568

Odds Ratio Estimates

Effect	Point Estimate	95% Wald Confidence Limits
race black vs white	0.095	0.006 1.584
month august vs septembe	1.795	0.322 10.014
month july vs septembe	1.472	0.267 8.105

# Bayesian approach using Jeffreys' prior

Uses Jeffreys' prior; inference from Markov chain Monte Carlo.

## Bayesian Analysis

### Model Information

Burn-In Size	2000
MC Sample Size	10000
Thinning	1
Sampling Algorithm	ARMS
Distribution	Binomial
Link Function	Logit

### Fit Statistics

DIC (smaller is better)	16.131
pD (effective number of parameters)	3.645

### Posterior Summaries

Parameter	N	Mean	Standard Deviation	Percentiles		
				25%	50%	75%
Intercept	10000	-1.8560	0.7482	-2.3170	-1.7943	-1.3319
raceblack	10000	-3.4542	1.9101	-4.5220	-3.1403	-2.0504
monthaugust	10000	0.6772	0.9428	0.0332	0.6543	1.2855
monthjuly	10000	0.4642	0.9365	-0.1802	0.4351	1.0652

### Posterior Intervals

Parameter	Alpha	Equal-Tail Interval		HPD Interval	
Intercept	0.050	-3.4978	-0.5705	-3.3326	-0.4128
raceblack	0.050	-8.0198	-0.5685	-7.5265	-0.2661
monthaugust	0.050	-1.1122	2.5991	-1.1378	2.5499
monthjuly	0.050	-1.2837	2.4032	-1.2698	2.4152

# Hierarchical model building

"When using a polynomial regression model as an approximation to the true regression function, statisticians will often fit a second-order or third-order model and then explore whether a lower-order model is adequate...With the hierarchical approach, if a polynomial term of a given order is retained, then all related terms of lower order are also retained in the model. Thus, one would not drop the quadratic term of a predictor variable but retain the cubic term in the model. Since the quadratic term is of lower order, it is viewed as providing more basic information about the shape of the response function; the cubic term is of higher order and is viewed as providing refinements in the specification of the shape of the response function." – *Applied Statistical Linear Models* by Neter, Kutner, Nachtsheim, and Wasserman.

"It is not usually sensible to consider a model with interaction but not the main effects that make up the interaction." – *Categorical Data Analysis* by Agresti.

"Consider the relationship between the terms  $\beta_1x$  and  $\beta_2x^2$ . To fit the term  $\beta_0 + \beta_2x^2$  without including  $\beta_1x$  implies that the maximum (or minimum) of the response occurs at  $x = 0$ ...ordinarily there is no reason to suppose that the turning point of the response is at a specified point in the  $x$ -scale, so that the fitting of  $\beta_2x^2$  without the linear term is usually unhelpful.

A further example, involving more than one covariate, concerns the relation between a cross-term such as  $\beta_{12}x_1x_2$  and the corresponding linear terms  $\beta_{11}x_1$  and  $\beta_{22}x_2$ . To include the former in a model formula without the latter two is equivalent to assuming the point  $(0, 0)$  is a col or saddle-point of the response surface. Again, there is usually no reason to postulate such a property for the origin, so that the linear terms must be included with the cross-term." – *Generalized Linear Models* by McCullagh and Nelder.

# Polynomial approximation to unknown surface

Real model

$$\text{logit}(\pi_i) = f(x_{i1}, x_{i2}).$$

First order approximation to  $f(x_1, x_2)$  about some  $(\bar{x}_1, \bar{x}_2)$ :

$$\begin{aligned} f(x_1, x_2) &= f(\bar{x}_1, \bar{x}_2) + \frac{\partial f(\bar{x}_1, \bar{x}_2)}{\partial x_1}(x_1 - \bar{x}_1) + \frac{\partial f(\bar{x}_1, \bar{x}_2)}{\partial x_2}(x_2 - \bar{x}_2) \\ &\quad + \text{HOT.} \\ &= \left[ f(\bar{x}_1, \bar{x}_2) - \bar{x}_1 \frac{\partial f(\bar{x}_1, \bar{x}_2)}{\partial x_1} - \bar{x}_2 \frac{\partial f(\bar{x}_1, \bar{x}_2)}{\partial x_2} \right] \\ &\quad + \left[ \frac{\partial f(\bar{x}_1, \bar{x}_2)}{\partial x_1} \right] x_1 + \left[ \frac{\partial f(\bar{x}_1, \bar{x}_2)}{\partial x_2} \right] x_2 + \text{HOT} \\ &= \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \text{HOT} \end{aligned}$$

$\text{logit}(\pi) = \beta_0 + \beta_1 x_1 + \beta_2 x_2$  is an approximation to unknown, infinite-dimensional  $f(x_1, x_2)$  characterized by  $(\beta_0, \beta_1, \beta_2)$ .



# Polynomial approximation to unknown surface

Now let  $\mathbf{x} = (x_1, x_2)$  and

$$f(\mathbf{x}) = f(\bar{\mathbf{x}}) + Df(\bar{\mathbf{x}})(\mathbf{x} - \bar{\mathbf{x}}) + \frac{1}{2}(\mathbf{x} - \bar{\mathbf{x}})'D^2f(\bar{\mathbf{x}})(\mathbf{x} - \bar{\mathbf{x}}) + \text{HOT}.$$

This similarly reduces to

$$f(x_1, x_2) = \beta_0 + \beta_1x_1 + \beta_2x_2 + \beta_3x_1^2 + \beta_4x_2^2 + \beta_5x_1x_2 + \text{HOT},$$

where  $(\beta_0, \beta_1, \beta_2, \beta_3, \beta_4, \beta_5)$  correspond to various (unknown) partial derivatives of  $f(x_1, x_2)$ . Depending on the shape of the true (unknown)  $f(x_1, x_2)$ , some or many of the terms in the approximation  $\text{logit}(\pi) = \beta_0 + \beta_1x_1 + \beta_2x_2 + \beta_3x_1^2 + \beta_4x_2^2 + \beta_5x_1x_2$  may be unnecessary.

We work backwards via Wald tests *hierarchically* getting rid of HOT first to get at more general trends/shapes, e.g. the first order approximation.

This directly relates to generalized additive models (GAM) where instead we approximate

$$f(x_1, x_2) = \beta_0 + g_1(x_1) + g_2(x_2),$$

where  $g_1(\cdot)$  and  $g_2(\cdot)$  are modeled flexibly (often using splines).

A linear additive model is a special case where  $g_j(x) = \beta_j x$  yielding  $f(x_1, x_2) = \beta_0 + \beta_1 x_1 + \beta_2 x_2$ .

## 7.4.9 Generalized additive models

Consider a linear regression problem:

$$Y_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + e_i,$$

where  $e_1, \dots, e_n \stackrel{iid}{\sim} N(0, \sigma^2)$ .

Diagnostics (residual plots, added variable plots) might indicate poor fit of the basic model above. Remedial measures might include transforming the response, transforming one or both predictors, or both. One also might consider adding quadratic terms and/or an interaction term.

Note: we only consider transforming *continuous* predictors!

# Transformations of predictors

When considering a transformation of one predictor, an added variable plot can suggest a transformation (e.g.  $\log(x)$ ,  $1/x$ ) that might work *if the other predictor is “correctly” specified*.

In general, a transformation is given by a function  $g(x)$ . Say we decide that  $x_{i1}$  should be log-transformed and the reciprocal of  $x_{i2}$  should be used. Then the resulting model is

$$Y_i = \beta_0 + \beta_1 \log(x_{i1}) + \beta_2/x_{i2} + e_i = \beta_0 + g_1(x_{i1}) + g_2(x_{i2}) + e_i,$$

where  $g_1(x)$  and  $g_2(x)$  are two functions of  $\beta_1$  and  $\beta_2$ , respectively.

## One method for “nonparametric regression”

Here we are specifying forms for  $g_1(x)$  and  $g_2(x)$  based on exploratory data analysis, but we could from the outset specify *models* for  $g_1(x)$  and  $g_2(x)$  that are rich enough to capture interesting and predictively useful aspects of how the predictors affect the response and *estimate these functions from the data*.

This is an example of “nonparametric regression,” which ironically involves the inclusion of *lots* of parameters rather than fewer.

# Additive model for normal-errors regression

For simple regression data  $\{(x_i, y_i)\}_{i=1}^n$ , a cubic spline smoother  $g(x)$  minimizes

$$\sum_{i=1}^n (y_i - g(x_i))^2 + \lambda \int_{-\infty}^{\infty} g''(x)^2 dx.$$

Good fit is achieved by minimizing the sum of squares  $\sum_{i=1}^n (y_i - g(x_i))^2$ . The  $\int_{-\infty}^{\infty} g''(x)^2 dx$  term measures how wiggly  $g(x)$  is and  $\lambda \geq 0$  is how much we will penalize  $g(x)$  for being wiggly.

So the spline trades off between goodness of fit and wiggleness.

Although not obvious, the solution to this minimization is a cubic spline: a piecewise cubic polynomial with the pieces joined at the unique  $x_j$  values.

Hastie and Tibshirani (1986, 1990) point out that the meaning of  $\lambda$  depends on the units  $x_i$  is measured in, but that  $\lambda$  can be picked to yield an “effective degrees of freedom”  $df$  or an “effective number of parameters” being used in  $g(x)$ . Then the complexity of  $g(x)$  is equivalent to  $df$ -degree polynomial, but with the coefficients “spread out” more yielding a more flexible function that fits data better.

Alternatively,  $\lambda$  can be picked through cross validation, by minimizing

$$CV(\lambda) = \sum_{i=1}^n (y_i - g_{\lambda}^{-i}(x_i))^2.$$

Both options are available in SAS.

## Generalized additive model

We don't have  $\{(x_i, y_i)\}_{i=1}^n$  where  $y_1, \dots, y_n$  are continuous, but rather  $\{(\mathbf{x}_i, y_i)\}_{i=1}^n$  where  $y_i$  is categorical (e.g. Bernoulli) or Poisson. The generalized additive model (GAM) is given by

$$h\{E(Y_i)\} = \beta_0 + g_1(x_{i1}) + \dots + g_p(x_{ip}),$$

for  $p$  predictor variables.  $Y_i$  is a member of an exponential family such as binomial, Poisson, normal, etc.  $h$  is a link function.

Each of  $g_1(x), \dots, g_p(x)$  are modeled via cubic smoothing splines, each with their own smoothness parameters  $\lambda_1, \dots, \lambda_p$  either specified as  $df_1, \dots, df_p$  or estimated through cross-validation. The model is fit through "backfitting." See Hastie and Tibshirani (1990) or the SAS documentation for details.



# Fit of GAM to O-ring space shuttle data

```
data shut1;
  input temp td @@;
  datalines;
66 0 70 1 69 0 68 0 67 0 72 0 73 0 70 0 57 1 63 1 70 1 78 0 67 0
53 1 67 0 75 0 70 0 81 0 76 0 79 0 75 1 76 0 58 1
;
ods html; ods graphics on;
proc gam plots(clm) data=shut1;
  model td = spline(temp) / dist=binomial;
run; quit; ods graphics off; ods html close;
```

## Output:

```
                The GAM Procedure
      Dependent Variable: td
Smoothing Model Component(s): spline(temp)
```

### Summary of Input Data Set

Number of Observations	23
Number of Missing Observations	0
Distribution	Binomial
Link Function	Logit

# Output from PROC GAM

## Iteration Summary and Fit Statistics

Number of local score iterations	15
Local score convergence criterion	5.925073E-10
Final Number of Backfitting Iterations	1
Final Backfitting Criterion	8.5164609E-9
The Deviance of the Final Estimate	12.445020758

The local score algorithm converged.

### Regression Model Analysis Parameter Estimates

Parameter	Parameter Estimate	Standard Error	t Value	Pr >  t
Intercept	5.18721	14.01486	0.37	0.7156
Linear(temp)	-0.08921	0.19693	-0.45	0.6560

### Smoothing Model Analysis Fit Summary for Smoothing Components

Component	Smoothing Parameter	DF	GCV	Num Unique Obs
Spline(temp)	0.999976	3.000000	136344	16

### Smoothing Model Analysis Analysis of Deviance

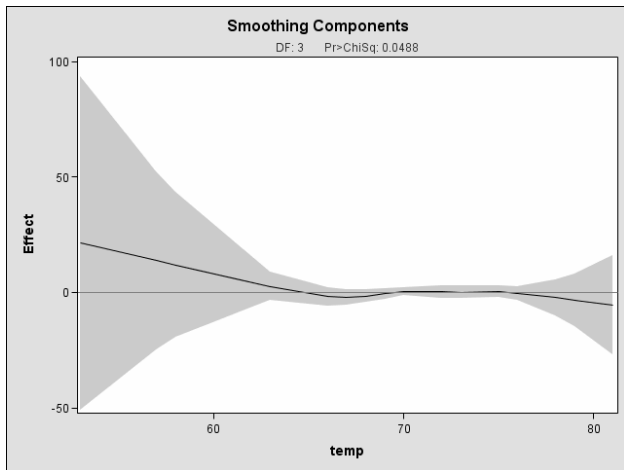
Source	DF	Sum of Squares	Chi-Square	Pr > ChiSq
Spline(temp)	3.00000	7.870171	7.8702	0.0488

The Analysis of Deviance table gives a  $\chi^2$ -test from comparing the deviance between the full model and the model with this variable dropped – here the intercept model *plus a linear effect in temperature*. We see that temperature effect is significantly nonlinear at the 5% level. The default  $df = 3$  corresponds to a smoothing spline with the complexity of a cubic polynomial.

The following plot was obtained from the `plots(c1m)` statement. The plot has the estimated smoothing spline function with the linear effect subtracted out. The plot includes a 95% curvewise Bayesian confidence band. We visually inspect where this band does not include zero to get an idea of where significant nonlinearity occurs. This plot can suggest simpler transformations of predictor variables than use of the full-blown smoothing spline. Model is

$$\text{logit}(\pi_i) = \beta_0 + \beta_1 x_i + \tilde{g}_1(x_i) = \beta_0 + g_1(x_i)$$

# Estimation of $\tilde{g}_1(\cdot)$ , “wiggly part” of $g_1(\cdot)$



The band basically includes zero for most temperature values; at a few points it comes close to not including zero.

The plot spans the range of temperature values in the data set and becomes highly variable at the ends. Do you think extrapolation is a good idea using GAMs?

We want to predict the probability of a failure at 39 degrees. I couldn't get GAM to predict beyond the range of predictor values.

The package `gam` was written by Trevor Hastie (one of the inventors of GAM) and (in your instructor's opinion) is easier to use and gives nicer output than SAS PROC GAM.

A subset of the `kyphosis` data set is given on p. 199. Kyphosis is severe forward flexion of the spine following spinal surgery. We will run the following code in class:

```
library(gam); data(kyphosis)
?kyphosis
fit=gam(Kyphosis~s(Age)+s(Number)+s(Start),family=binomial(link=logit),
  data=kyphosis)
par(mfrow=c(2,2))
plot(fit,se=TRUE)
summary(fit)
```

# More R examples

```
# Challenger O-ring data
td=c(0,1,0,0,0,0,0,0,1,1,1,0,0,1,0,0,0,0,0,0,1,0,1)
te=c(66,70,69,68,67,72,73,70,57,63,70,78,67,53,67,75,70,81,76,79,75,76,58)
fit=gam(td~s(te),family=binomial(link=logit))
plot(fit,se=TRUE)
summary(fit)
fit$coeff
```

```
# example with linear log-odds
# parametric part significant, nonparametric not significant
x=rnorm(1000,0,2); p=exp(x)/(1+exp(x)); y=rbinom(1000,1,p)
plot(x,y)
fit=gam(y~s(x),family=binomial(link=logit))
plot(fit,se=TRUE)
summary(fit)
fit$coef
```

```
# example with quadratic log-odds
# parametric part not be significant, nonparametric significant
p=exp(x^2)/(1+exp(x^2)); y=rbinom(1000,1,p)
plot(x,y)
fit=gam(y~s(x),family=binomial(link=logit))
plot(fit,se=TRUE)
summary(fit)
```