

Chapter 4

The GAM Procedure

Chapter Table of Contents

OVERVIEW	3
GETTING STARTED	3
SYNTAX	8
PROC GAM Statement	8
BY Statement	8
CLASS Statement	9
FREQ Statement	9
ID Statement	10
MODEL Statement	10
OUTPUT Statement	11
SCORE Statement	12
DETAILS	12
Nonparametric Regression	12
Additive Models and Generalized Additive Models	13
Back-Fitting and Local Scoring Algorithms	14
Smoothers	17
Selection of Smoothing Parameters	18
Distribution Family and Canonical Link	19
Forms of Additive Models	19
ODS Tables Produced by PROC GAM	20
EXAMPLES	20
Example 4.1 Generalized Additive Model with Binary Data	20
Example 4.2 Comparing PROC GAM with PROC TPSPLINE	27
REFERENCES	32

Chapter 4

The GAM Procedure

Overview

The GAM procedure is a new, experimental procedure that fits generalized additive models as those models are defined by Hastie and Tibshirani (1990). This procedure provides an array of powerful tools for data analysis, based on nonparametric regression and smoothing techniques.

Nonparametric regression relaxes the usual assumption of linearity and enables you to uncover structure in the relationship between the independent variables and the dependent variable that might otherwise be missed. The SAS System provides many procedures for nonparametric regression, such as the LOESS procedure for local regression and the TPSPLINE procedure for thin-plate smoothing splines. The generalized additive models fit by the GAM procedure combine

- an additive assumption (Stone 1985) that allows relatively many nonparametric relationships to be explored simultaneously with
- the distributional flexibility of generalized linear models (Nelder 1972)

Thus, you can use the GAM procedure when you have multiple independent variables whose effect you want to model nonparametrically, or when the dependent variable is not normally distributed. Refer to the “Nonparametric Regression” section on page 12 for more details on the form of generalized additive models.

The GAM procedure

- provides nonparametric estimates for additive models
- supports the use of multidimensional data
- supports multiple SCORE statements
- fits both generalized semiparametric additive models and generalized additive models
- enables you to choose a particular model by specifying the model degrees of freedom or smoothing parameter

Getting Started

The following example illustrates the use of the GAM procedure to explore in a nonparametrical way how two factors affect a response. The data come from a study (Sockett et al. 1987) of the factors affecting patterns of insulin-dependent diabetes

mellitus in children. The objective is to investigate the dependence of the level of serum C-peptide on various other factors in order to understand the patterns of residual insulin secretion. The response measurement is the logarithm of C-peptide concentration (pmol/ml) at diagnosis, and the predictor measurements are age and base deficit (a measure of acidity):

```

title 'Patterns of Diabetes';
data diabetes;
  input Age BaseDeficit CPeptide @@;
  logCP = log(CPeptide);
  datalines;
  5.2   -8.1  4.8   8.8  -16.1  4.1  10.5   -0.9  5.2
 10.6   -7.8  5.5  10.4  -29.0  5.0   1.8  -19.2  3.4
 12.7  -18.9  3.4  15.6  -10.6  4.9   5.8   -2.8  5.6
   1.9  -25.0  3.7   2.2   -3.1  3.9   4.8   -7.8  4.5
   7.9  -13.9  4.8   5.2   -4.5  4.9   0.9  -11.6  3.0
  11.8   -2.1  4.6   7.9   -2.0  4.8  11.5   -9.0  5.5
  10.6  -11.2  4.5   8.5   -0.2  5.3  11.1   -6.1  4.7
  12.8   -1.0  6.6  11.3   -3.6  5.1   1.0   -8.2  3.9
  14.5   -0.5  5.7  11.9   -2.0  5.1   8.1   -1.6  5.2
  13.8  -11.9  3.7  15.5   -0.7  4.9   9.8   -1.2  4.8
  11.0  -14.3  4.4  12.4   -0.8  5.2  11.1  -16.8  5.1
   5.1   -5.1  4.6   4.8   -9.5  3.9   4.2  -17.0  5.1
   6.9   -3.3  5.1  13.2   -0.7  6.0   9.9   -3.3  4.9
  12.5  -13.6  4.1  13.2   -1.9  4.6   8.9  -10.0  4.9
  10.8  -13.5  5.1
;
run;

```

The following statements perform the desired analysis. The PROC GAM statement invokes the procedure and specifies the `diabetes` data set as input. The MODEL statement specifies `logCP` as the response variable and requests that univariate BSPLINES with 4 degrees of freedom be used to model the effect of `Age` and `BaseDeficit`. The OUTPUT statement specifies that partial prediction curves are to be saved in the data set `estimates`:

```

title 'Patterns of Diabetes';
proc gam data=diabetes;
  model logCP = spline(age) spline(BaseDeficit);
  output out=estimates p;
run;

```

The results are shown in Figure 4.1 and Figure 4.2.

```

Patterns of Diabetes

The GAM Procedure
Dependent Variable: logCP
Smoothing Model Component: spline(Age) spline(BaseDeficit)

Iteration Summary and Fit Statistics

Final number of backfitting iterations          5
Final backfitting criterion                    5.542743E-10
Final residual sum of squares                  0.4180802183

Summary of Input Data Set

Number of Observations                        43
Number of Missing Observations                0
Distribution                                  Gaussian
Link Function                                 Identity

```

Figure 4.1. Summary Statistics

Figure 4.1 shows two tables. The first table summarizes the convergence criterion for back-fitting, and the second one summarizes the input data set and the distribution family used for the model.

```

Patterns of Diabetes

The GAM Procedure
Dependent Variable: logCP
Smoothing Model Component: spline(Age) spline(BaseDeficit)

Regression Model Analysis
Parameter Estimates

Parameter          Parameter          Standard
                   Estimate           Error      t Value      Pr > |t|

Intercept          1.48141           0.02588    57.24        <.0001
L_Age              0.01437           0.00437     3.28         0.0024
L_BaseDeficit      0.00807           0.00240     3.35         0.0020

Smoothing Model Analysis
Fit Statistics of Smoothing Components

Component          Smoothing          DF           GCV           No. of
                   Parameter           DF           GCV           Unique
                   Parameter           DF           GCV           Obs.

spline(Age)        0.995582           4.000000    0.011675     37
spline(BaseDeficit) 0.995299           4.000000    0.012437     39

Smoothing Model Analysis
Analysis of Deviance

Source              DF           Sum of
                   DF           Squares      F Value      Pr > F

spline(Age)         4.000000    0.150760     12.26        0.0155
spline(BaseDeficit) 4.000000    0.081272     6.61         0.1580

```

Figure 4.2. Analysis of Model

Figure 4.2 displays summary statistics for the model. It consists of three tables. The first is the Parameter Estimates table for the parametric part of the model. It indicates that the linear trends for both `Age` and `BaseDeficit` are highly significant with p -values of 0.0024 and 0.0020. The second table is the summary of smoothing components of the nonparametric part of the model. Since the GAM fit used the default $DF = 4$, the main point of this table is to present the smoothing parameter values that yield this DF for each component. Finally, the third table shows the Analysis of Deviance table for the nonparametric component of the model.

The `P` option in the `OUTPUT` statement puts the partial predictions for `Age` and `BaseDeficit` in the output data set. You can compute the entire partial prediction effect for each factor by adding the estimated linear terms to the respective partial predictions, as in the following statement:

```
data estimates; set estimates;
  P2_age      = P_age      + 0.01437*age;
  P2_BaseDeficit = P_BaseDeficit + 0.00807*BaseDeficit;
run;
```

Plotting the partial predictions is one way to explore the overall shape of the relationship between each factor and the response. First of all, the following statements set up the graphics options:

```
axis1 label=(angle=90 rotate=0) minor=none;
axis2 minor=none order=(0 to 16 by 4);
symbol1 color=red interpol=join value=none line=1;
symbol2 color=blue interpol=join value=none line=2;
```

Then the following statements plot the partial prediction curves for `Age` and `BaseDeficit`:

```
proc sort data=estimates;
  by age;

title;
proc gplot data=estimates;
  plot P_age *age = 1
       P2_Age*Age = 2 /overlay legend frame cframe=ligr
       name='gam1' vaxis=axis1 haxis=axis2;
run;

proc sort data=estimates;
  by BaseDeficit;

axis2 minor=none;
proc gplot data=estimates;
  plot P_BaseDeficit *BaseDeficit = 1
       P2_BaseDeficit*BaseDeficit = 2 /
       overlay legend frame cframe=ligr name='gam2'
       vaxis=axis1 haxis=axis2;
run;
```

Finally, the following statements redisplay the curves side by side for easy comparison:

```

goptions display;
proc greplay tc=tempcat nofs;
  igout gseg;
  tdef newtwo des='two plots of equal size'
  1/llx=0   lly=0
     ulx=0   uly=100
     urx=50  ury=100
     lrx=50  lry=0
  2/llx=50  lly=0
     ulx=50  uly=100
     urx=100 ury=100
     lrx=100 lry=0
  ;
  template newtwo;
  treplay 1:gaml
          2:gam2;
run; quit;

```

The resulting plots for each predictor with and without the linear term are shown in Figure 4.3.

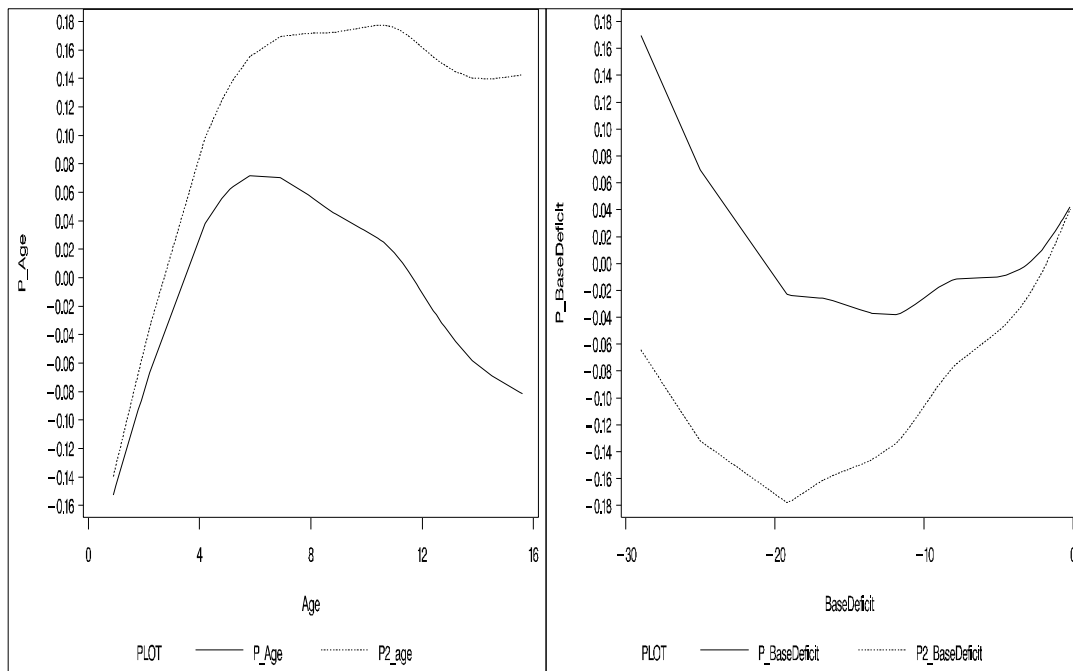


Figure 4.3. Partial Prediction for Each Predictor

Both plots show a strong quadratic pattern, with a possible indication of higher-order behavior. Further investigation is required to determine whether these patterns are real or not.

Syntax

```

PROC GAM < option > ;
  CLASS variables ;
  MODEL dependent = < PARAM(effects) >
                    smoothing effects < /options > ;
  SCORE data=SAS-data-set out=SAS-data-set ;
  OUTPUT < out=SAS-data-set > keyword < ...keyword > < /option > ;
  BY variables ;
  ID variables ;
  FREQ variable ;

```

The syntax of the GAM procedure is similar to that of other regression procedures in the SAS System. The PROC GAM and MODEL statements are required. The SCORE statement can appear multiple times; all other statements appear only once.

The syntax for PROC GAM is described in the following sections in alphabetical order after the description of the PROC GAM statement.

PROC GAM Statement

```
PROC GAM< option > ;
```

The PROC GAM statement invokes the procedure. You can specify the following option.

DATA=SAS-data-set

specifies the SAS data set to be read by PROC GAM. The default value is the most recently created data set.

BY Statement

```
BY variables ;
```

You can specify a BY statement with PROC GAM to obtain separate analyses on observations in groups defined by the BY variables. When a BY statement appears, the procedure expects the input data set to be sorted in order of the BY variables.

If your input data set is not sorted in ascending order, use one of the following alternatives:

- Sort the data using the SORT procedure with a similar BY statement.
- Specify the BY statement option NOTSORTED or DESCENDING in the BY statement for the GAM procedure. The NOTSORTED option does not mean that the data are unsorted but rather that the data are arranged in groups (ac-

ording to values of the BY variables) and that these groups are not necessarily in alphabetical or increasing numeric order.

- Create an index for the BY variables using the DATASETS procedure.

For more information on the BY statement, refer to the discussion in *SAS Language Reference: Concepts*. For more information on the DATASETS procedure, refer to the discussion in the *SAS Procedures Guide*.

CLASS Statement

CLASS *variables* ;

The CLASS statement names the classification variables to be used in the analysis. Typical class variables are TREATMENT, SEX, RACE, GROUP, and REPLICATION. If the CLASS statement is used, it must appear before the MODEL statement.

Classification variables can be either character or numeric. Class levels are determined from the formatted values of the CLASS variables. Thus, you can use formats to group values into levels. Refer to the discussion of the FORMAT procedure in the *SAS Procedures Guide*, and the discussions for the FORMAT statement and SAS formats in *SAS Language Reference: Dictionary*.

FREQ Statement

FREQ *variable* ;

The FREQ statement names a variable that provides frequencies for each observation in the DATA= data set. Specifically, if n is the value of the FREQ variable for a given observation, then that observation is used n times.

The analysis produced using a FREQ statement reflects the expanded number of observations. You can produce the same analysis (without the FREQ statement) by first creating a new data set that contains the expanded number of observations. For example, if the value of the FREQ variable is 5 for the first observation, the first five observations in the new data set are identical. Each observation in the old data set is replicated n_i times in the new data set, where n_i is the value of the FREQ variable for that observation.

If the value of the FREQ variable is missing or is less than 1, the observation is not used in the analysis. If the value is not an integer, only the integer portion is used.

ID Statement

ID *variables* ;

The variables in the ID statement are copied from the input data set to the OUT= data set. If you omit the ID statement, only the variables used in the MODEL statement and requested statistics are included in the output data set.

MODEL Statement

MODEL *dependent* = <PARAM(*effects*)> <*smoothing effects*> </options> ;

The MODEL statement specifies the dependent variable and the independent effects you want to use to model its values. Specify the independent parametric variables inside the parentheses of PARAM(). The parametric variables can be either CLASS variables or continuous variables. Any number of smoothing effects can be specified, as follows:

Smoothing Effect	Meaning
spline(variable, <df=number>)	fit smoothing spline with the variable and with DF=number
spline2(variable, variable, <df=number>)	fit bivariate thin-plate spline with DF=number

Both parametric effects and smoothing effects are optional, but at least one of them must be present.

If only parametric variables are present, PROC GAM fits a parametric linear model using the terms inside the parentheses of PARAM(). If only smoothing effects are present, PROC GAM fits a nonparametric additive model. If both types of effect are present, PROC GAM fits a semiparametric model using the parametric effects as the linear part of the model.

The following table shows how to specify various models for a dependent variable *y* and independent variables *x*, *x1*, and *x2*.

Table 4.1. Syntax for Common GAM Models

Type of Model	Syntax	Mathematical Form
Parametric	model <i>y</i> = param(<i>x</i>);	$E(y) = \beta_0 + \beta_1 x$
Nonparametric	model <i>y</i> = spline(<i>x</i>);	$E(y) = \beta_0 + s(x_2)$
Semiparametric	model <i>y</i> = param(<i>x1</i>) spline(<i>x2</i>);	$E(y) = \beta_0 + \beta_1 x_1 + s(x_2)$
Additive	model <i>y</i> = spline(<i>x1</i>) spline(<i>x2</i>);	$E(y) = \beta_0 + s_1(x_1) + s_2(x_2)$
Thin-plate spline	model <i>y</i> = spline(<i>x1</i> , <i>x2</i>);	$E(y) = \beta_0 + s(x_1, x_2)$

You can specify the following options in the MODEL statement.

ALPHA=number

specifies the significance level α of the confidence limits on the final nonparametric component estimates when you request confidence limits to be included in the output data set. Specify *number* as a value between 0 and 1. The default value is 0.05. Refer to the “OUTPUT Statement” section on page 11 for more information on the OUTPUT statement.

DIST=distribution-id

specifies the distribution family used in the model. The *distribution-id* can be GAUSSIAN or LOGISTIC. The canonical link is used with those distributions. Although theoretically, alternative links are possible, with nonparametric models the final fit is relatively insensitive to the precise choice of link function. Therefore, only the canonical link for each distribution family is implemented in PROC GAM.

EPSILON=number

specifies the convergence criterion for the back-fitting algorithm.

ITPRINT

produces an iteration table for the smoothing effects.

MAXITER=number

specifies the maximum number of iterations for the back-fitting algorithm.

METHOD=GCV

specifies that the value of the smoothing parameter should be selected by generalized cross validation. If you specify both METHOD=GCV and the DF= option for the smoothing effects, the user-specified DF= is used, and the METHOD=GCV option is ignored. Refer to the “Selection of Smoothing Parameters” section on page 18 for more details on the GCV method.

OUTPUT Statement

OUTPUT *OUT=SAS-data-set* < *keyword* . . . *keyword* > ;

The OUTPUT statement creates a new SAS data set containing diagnostic measures calculated after fitting the model.

You can request a variety of diagnostic measures that are calculated for each observation in the data set. The new data set contains the variables specified in the MODEL statement in addition to the requested variables. If no *keyword* is present, the data set contains only the predicted values.

Details on the specifications in the OUTPUT statement are as follows.

OUT=SAS-data-set

specifies the name of the new data set to contain the diagnostic measures. This specification is required.

keyword

specifies the statistics to include in the output data set. The keywords and the statistics they represent are as follows:

PRED	predicted values
ADIAG	diagonal element of the hat matrix associated with the observation

The names of the new variables that contain the statistics are formed by using a prefix of one or more characters that identify the statistic, followed by an underscore (`_`), followed by the dependent variable name.

For example, suppose that you have a dependent variable `y`, and you specify the keywords `PRED` and `ADIAG`. In this case, the output SAS data set will contain the variables `P_y` and `ADIAG_y`.

SCORE Statement

SCORE *DATA=SAS-data-set* *OUT=SAS-data-set* ;

The `SCORE` statement calculates predicted values for a new data set. If you have multiple data sets to predict, you can specify multiple `SCORE` statements. You must use a `SCORE` statement for each data set.

The following keywords must be specified in the `SCORE` statement.

DATA=SAS-data-set

specifies an input SAS data set containing all the variables included in independent effects in the `MODEL` statement. The predicted response is computed for each observation in the `SCORE DATA=` data set.

OUT=SAS-data-set

specifies the name of the SAS data set to contain the predictions.

Details

Nonparametric Regression

Nonparametric regression relaxes the usual assumption of linearity and enables you to explore the data visually, uncovering structure in the data that might otherwise be missed.

However, many forms of nonparametric regression do not perform well when the number of independent variables in the model is large. The sparseness of data in this setting causes the variances of the estimates to be unacceptably large unless the sample size is extremely large. The problem of rapidly increasing variance for increasing dimensionality is sometimes referred to as the “curse of dimensionality.” Interpretability is another problem with nonparametric regression based on kernel and smoothing spline estimates. The information these estimates contain about the relationship between the dependent and independent variables is often difficult to comprehend.

To overcome these difficulties, Stone (1985) proposed additive models. These models estimate an additive approximation to the multivariate regression function. The

benefits of an additive approximation are at least twofold. First, since each of the individual additive terms is estimated using a univariate smoother, the curse of dimensionality is avoided, at the cost of not being able to approximate universally. Second, estimates of the individual terms explain how the dependent variable changes with the corresponding independent variables.

To extend the additive model to a wide range of distribution families, Hastie and Tibshirani (1990) proposed generalized additive models. These models enable the mean of the dependent variable to depend on an additive predictor through a non-linear link function. The models permit the response probability distribution to be any member of the exponential family of distributions. Many widely used statistical models belong to this general class; they include additive models for Gaussian data, nonparametric logistic models for binary data, and nonparametric log-linear models for Poisson data.

Additive Models and Generalized Additive Models

This section describes the methodology and the fitting procedure behind generalized additive models.

Let Y be a response random variable and X_1, X_2, \dots, X_p be a set of predictor variables. A regression procedure can be viewed as a method for estimating the expected value of Y given the values of X_1, X_2, \dots, X_p . The standard linear regression model assumes a linear form for the conditional expectation

$$E(Y|X_1, X_2, \dots, X_p) = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \dots + \beta_p X_p$$

Given a sample, estimates of $\beta_0, \beta_1, \dots, \beta_p$ are usually obtained by the least squares method.

The additive model generalizes the linear model by modeling the conditional expectation as

$$E(Y|X_1, X_2, \dots, X_p) = s_0 + s_1(X_1) + s_2(X_2) + \dots + s_p(X_p)$$

where $s_i(X), i = 1, 2, \dots, p$ are smooth functions.

In order to be estimable, the smooth functions s_i have to satisfy standardized conditions such as $E s_j(X_j) = 0$. These functions are not given a parametric form but instead are estimated in a nonparametric fashion.

While traditional linear models and additive models can be used in most statistical data analysis, there are types of problems for which they are not appropriate. For example, the normal distribution may not be adequate for modeling discrete responses such as counts or bounded responses such as proportions.

Generalized additive models address these difficulties, extending additive models to many other distributions besides just the normal. Thus, generalized additive models can be applied to a much wider range of data analysis problems.

Similar to generalized linear models, generalized additive models consist of a random component, an additive component, and a link function relating the two components. The response Y , the random component, is assumed to have exponential family density

$$f_Y(y; \theta; \phi) = \exp \left\{ \frac{y\theta - b(\theta)}{a(\phi)} + c(y, \phi) \right\}$$

where θ is called the natural parameter and ϕ is the scale parameter. The mean of the response variable μ is related to the set of covariates X_1, X_2, \dots, X_p by $g(\mu) = \eta$. Here, η is defined as

$$\eta = s_0 + \sum_{i=1}^p s_i(X_i)$$

where $s_1(\cdot), \dots, s_p(\cdot)$ are smooth functions, the quantity η is the linear component, and $g(\cdot)$ is the link function. The most commonly used link for a given f is called the canonical link, for which $\eta = \theta$.

Generalized additive models and generalized linear models can be applied in similar situations, but they serve different analytic purposes. Generalized linear models emphasize estimation and inference for the parameters of the model, while generalized additive models focus on exploring data nonparametrically. Generalized additive models are more suitable for exploring the data set and visualizing the relationship between the dependent variable and the independent variables.

Back-Fitting and Local Scoring Algorithms

Consider the estimation of the smoothing terms $s_0, s_1(\cdot), \dots, s_p(\cdot)$ in the additive model

$$\mu(X) = s_0 + \sum_{i=1}^p s_i(X_i)$$

where $E[s_j(X_j)] = 0$ for every j . Since the algorithm for additive models is the basis for fitting generalized additive models, the algorithm for additive models is discussed first.

Many ways are available to approach the formulation and estimation of additive models. The back-fitting algorithm is a general algorithm that can fit an additive model using any regression-type fitting mechanisms.

Define the partial residual as

$$R_j = Y - s_0 - \sum_{k \neq j} s_k(X_k)$$

Then $E(R_j|X_j) = s_j(X_j)$. This observation provides a way for estimating each smoothing function $s_j(\cdot)$ given estimates $\{\hat{s}_i(\cdot), i \neq j\}$ for all the others. The resulting iterative procedure is known as the back-fitting algorithm (Friedman and Stuetzle 1981).

The Back-Fitting Algorithm

1. **Initialization:**

$$s_0 = E(Y), s_1^1 = s_2^1 = \dots = s_p^1 = 0, m = 0.$$

2. **Iterate:**

$$m = m + 1$$

for $j = 1$ to p do:

$$R_j = Y - s_0 - \sum_{k=1}^{j-1} s_k^m(X_k) - \sum_{k=j+1}^p s_k^{m-1}(X_k)$$

$$S_j^m = E(R_j|X_j).$$

3. **Until:**

$$RSS = \text{Avg}(Y - s_0 - \sum_{j=1}^p s_j^m(X_j))^2 \text{ fails to decrease.}$$

In the above notation, $s_j^m(\cdot)$ denotes the estimate of $s_j(\cdot)$ at the m th iteration. It can be shown that RSS never increases at any step, which implies that the algorithm always converges. However, the individual functions need not be unique, since dependence among the covariates can lead to more than one representation for the same fitted surface.

A weighted back-fitting algorithm has the same form as for the unweighted case, except that the smoothers are weighted. The weights might represent the relative precision of each observation or might arise as part of another iterative procedure. For example, weights are used in the local scoring procedure described later in this section.

The algorithm so far described fits just additive models. The algorithm for generalized additive models is a little more complicated. Generalized additive models extend generalized linear models in the same manner that additive models extend linear regression models, that is, by replacing form $\alpha + \sum_j X_j \beta_j$ with the additive form $\alpha + \sum_j f_j(X_j)$. Thus, it is helpful to review the iteratively reweighted least-square procedure for computing the maximum likelihood estimates in a generalized linear model.

For generalized linear models, the maximum likelihood estimate of β is defined by the score equations

$$\sum_{i=1}^n x_{ij} \left(\frac{\partial \mu_i}{\partial \eta_i} \right) V_i^{-1}(y_i - \mu_i), j = 0, 1, \dots, p$$

where V_i is the variance matrix for Y_i . The Fisher scoring procedure is the standard method for solving these equations. It involves a Newton-Raphson algorithm using the expected (as opposed to the observed) information matrix. An equivalent procedure that is convenient for this problem is called dependent variable regression and is a form of iteratively reweighted least squares. Given a current coefficient vector

β^0 , with corresponding linear predictor η^0 and fitted values μ^0 , construct the adjusted dependent variable

$$z_i = \eta_i^0 + (y_i - \mu_i^0) \left(\frac{\partial \eta_i}{\partial \mu_i} \right)_0$$

Define weights w_i by

$$w_i^{-1} = \left(\frac{\partial \eta_i}{\partial \mu_i} \right)_0^2 V_i^0$$

The algorithm proceeds by regressing z_i on x with weight w_i to obtain a revised estimate β . Then a new μ^0 and η^0 are computed, new z_i s are computed, and the process is repeated until the change in the deviance

$$D(y; \hat{\mu}) = 2(l(\mu_{max}; y) - l(\hat{\mu}'y))$$

is sufficiently small.

Some adjusted dependent variables and weights for commonly used models are listed in the following table.

Distribution	Link	Adjusted Dependent(Z)	Weights(w)
Normal	identity	y	1
Bin(n, μ)	logit	$\eta + (y - \mu)/n\mu(1 - \mu)$	$n\mu(1 - \mu)$
Gamma	log	$\eta + (y - \mu)/\mu$	1
Poisson	log	$\eta + (y - \mu)/\mu$	μ

Generalized additive models differ from generalized linear models in that an additive predictor replaces the linear predictor. Estimation of the additive terms is accomplished by replacing the weighted linear regression in the adjusted dependent variable regression by the weighted back-fitting algorithm for fitting a weighted additive mode. This results in the algorithm described below as the *local scoring algorithm*. The name “local scoring” derives from the fact that local averaging is used to generalize the Fisher scoring procedure.

The General Local Scoring Algorithm

1. **Initialization:**

$$s_i = g(E(y)), s_1^0 = s_2^0 = \dots = s_p^0 = 0, m = 0.$$

2. **Iterate:**

$$m = m + 1$$

Form the adjusted dependent variable, predictor, and mean based on the previous iteration

$$Z = \eta^{m-1} + (Y - \mu^{m-1})(\partial \eta / \partial \mu^{m-1})$$

$$\eta^{m-1} = s_0 + \sum_{j=1}^p s_j^{m-1}(X_j)$$

$$\mu^{m-1} = g^{-1}(\eta^{m-1}).$$

Form the weights

$$w_i = (\partial\mu^{m-1}/\partial\eta^{m-1})^2 V_i^{-1}.$$

Fit an additive model to Z using the back-fitting algorithm with weights W to obtain estimated functions $s_j^m(\cdot)$.

3. **Until:**

$\text{Avg}(D(Y, \mu^m))$ fails to decrease, where $\text{Avg}(D(Y, \mu^m))$ is an average of the deviance of estimate μ^m .

The estimating procedure for generalized additive models consists of two loops. Inside each step of the local scoring algorithm (outer loop), a weighted back-fitting algorithm (inner loop) is used until convergence, that is, until the RSS fails to decrease. Then, based on the estimates from this weighted back-fitting algorithm, a new set of weights is calculated and the next iteration of the scoring algorithm starts. The scoring algorithm stops when the deviance of the estimates ceases to decrease.

Smoothers

A smoother is a tool for summarizing the trend of a response measurement Y as a function of one or more predictor measurements X_1, \dots, X_p . It produces an estimate of the trend that is less variable than Y itself. An important property of a smoother is its nonparametric nature. It doesn't assume a rigid form for the dependence of Y on X_1, \dots, X_p . This section gives a brief overview of the smoothers that can be used with the GAM procedure.

Cubic Smoothing Spline

A smoothing spline is the solution to the following optimization problem: among all functions $\eta(x)$ with two continuous derivatives, find one that minimizes the penalized least square

$$\sum_{i=1}^n (y_i - \eta(x_i))^2 + \lambda \int_a^b (\eta''(t))^2 dt$$

where λ is a fixed constant, and $a \leq x_1 \leq \dots \leq x_n \leq b$. The first term measures closeness to the data while the second term penalizes curvature in the function. It can be shown that there exists an explicit, unique minimizer, and that minimizer is a natural cubic spline with knots at the unique values of x_i .

The parameter λ is the smoothing parameter. Large values of λ produce smoother curves while smaller values produce wiggly curves.

Thin-Plate Smoothing Spline

The theoretical foundations for the thin-plate smoothing spline are described in Duchon (1976, 1977) and Meinguet (1979). Further results and applications are given in Wahba and Wendelberger (1980). Refer to "The TPSPLINE Procedure" in *SAS/STAT User's Guide, Version 8* for more details.

Selection of Smoothing Parameters

CV and GCV

The smoothers discussed here have a single smoothing parameter. In choosing the smoothing parameter, cross validation can be used. Cross validation works by leaving points (x_i, y_i) out one at a time, estimating the squared residual for smooth function at x_i based on the remaining $n - 1$ data points, and choosing the smoother to minimize the sum of those squared residuals. This mimics the use of training and test samples for prediction. The cross validation function is defined as

$$CV(\lambda) = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{\eta}_{\lambda}^{-i}(x_i))^2$$

where $\hat{\eta}_{\lambda}^{-i}(x_i)$ indicates the fit at x_i , computed by leaving out the i th data point. The quantity $nCV(\lambda)$ is sometimes called the prediction sum of squares or *PRESS* (Allen 1974).

All of the smoothers fit by the GAM procedure can be formulated as a linear combination of the sample responses

$$\hat{\eta}(x) = A(\lambda)Y$$

for some matrix $A(\lambda)$, which depends on λ . (The matrix $A(\lambda)$ depends on x and the sample data, as well, but this dependence is suppressed in the preceding equation.) Let a_{ii} be the diagonal elements of the $A(\lambda)$. Then the *CV* function can be expressed as

$$CV(\lambda) = \frac{1}{n} \sum_{i=1}^n \left(\frac{(y_i - \hat{\eta}_{\lambda}(x_i))}{1 - a_{ii}} \right)^2$$

In most cases, it is very time consuming to compute the quantity a_{ii} . To solve this computational problem, Wahba (1990) has proposed the generalized cross validation function (*GCV*) that can be used to solve a wide variety of problems involving selection of a parameter to minimize the prediction risk.

The *GCV* function is defined as

$$GCV(\lambda) = \frac{\sum_{i=1}^n (y_i - \hat{\eta}_{\lambda}(x_i))^2}{(n - \text{tr}(A(\lambda)))^2}$$

The *GCV* formula simply replaces the a_{ii} with $\text{tr}(A(\lambda))/n$. Therefore, it can be viewed as a weighted version of *CV*. In most of the cases of interest, *GCV* is closely related to *CV* but much easier to compute. The GAM procedure uses the *GCV* function as the criterion for choosing the smoothing parameters.

The A matrix has the same role as the projection matrix in linear regression; therefore, nonparametric degrees of freedom (DF) for the model can be defined as $\text{tr}(A)$.

Distribution Family and Canonical Link

For each distribution, more than one link can exist. Different link functions may result in a slight difference in estimates for parametric models. However, the difference will be less pronounced for nonparametric models because of the flexibility of nonparametric model forms. To simplify the calculation, the GAM procedure uses the canonical link.

The GAM procedure can fit the data from the Gaussian and binomial distributions:

The Gaussian Model

With this model, the link function is the identity function, and the generalized additive model is the additive model.

The Logistic Model

A binomial response model assumes that the proportion of successes Y is such that Y has a $Bin(n(x), p(x))$ distribution. The $Bin(n(x), p(x))$ refers to the binomial distribution with parameters $n(x)$ and $p(x)$. Often the data are binary, in which case $n(x) = 1$. The canonical link is

$$g(p(x)) = \log \frac{p(x)}{1 - p(x)} = \eta(x)$$

Forms of Additive Models

Suppose that y is a continuous variable and x_1 and x_2 are two explanatory variables of interest. To fit an additive model, you can use a MODEL statement similar to that used in many regression procedures in the SAS system:

```
model y = spline(x1) spline(x2);
```

This model statement requires the procedure to fit the following model:

$$f(x_1, x_2) = \text{Intercept} + s_1(x_1) + s_2(x_2)$$

where the $s_i()$ terms denote nonparametric spline functions of the respective explanatory variables.

The GAM procedure can fit semiparametric models. The following MODEL statement assumes a linear relation with x_1 and an unknown functional relation with x_2 :

```
model y = param(x1) spline(x2);
```

If you want to fit a model containing a functional two-way interaction between x_1 and x_2 , you can use the following MODEL statement:

```
model y = spline2(x1,x2);
```

In this case, the GAM procedure fits a model equivalent to that of PROC TPSPLINE.

ODS Tables Produced by PROC GAM

PROC GAM assigns a name to each table it creates. You can use these names to reference the table when using the Output Delivery System (ODS) to select tables and create output data sets. These names are listed in the following table. For more information on ODS, refer to “Using the Output Delivery System” in *SAS/STAT User’s Guide, Version 8*.

Table 4.2. ODS Tables Produced by PROC GAM

ODS Table Name	Description	Statement	Option
ANOVA	ANOVA table for parametric fit	PROC	default
ANODEV	Analysis of Deviance table for smoothing variables	PROC	default
ClassSummary	Summary of class variables	PROC	default
DataSummary	Data summary	PROC	default
IterSummary	Iteration summary	PROC	default
FitSummary	Fit parameters and fit summary	PROC	default
ParameterEstimates	Parameter estimation for regression variables	PROC	default
Iteration	Iteration history table	MODEL	ITPRINT

By referring to the names of such tables, you can use the ODS OUTPUT statement to place one or more of these tables in output data sets.

Examples

Example 4.1. Generalized Additive Model with Binary Data

The following example illustrates the capabilities of the GAM procedure and compares it to the GENMOD procedure.

The data used in this example are based on a study by Bell et al. (1989). Bell and his associates studied the result of multiple-level thoracic and lumbar laminectomy, a corrective spinal surgery commonly performed on children. The data in the study consist of retrospective measurements on 83 patients. The specific outcome of interest is the presence (1) or absence (0) of kyphosis, defined as a forward flexion of the spine of at least 40 degrees from vertical. The available predictor variables are Age in months at time of the operation, the starting of vertebrae levels involved in the operation (StartVert), and the number of levels involved (NumVert). The goal of this analysis is to identify risk factors for kyphosis. PROC GENMOD can be used to investigate the relationship among kyphosis and the predictors. The following DATA step creates the data kyphosis:

```

title 'Comparing PROC GAM with PROC GENMOD';
data kyphosis;
  input Age StartVert NumVert Kyphosis @@;
  datalines;
71 5 3 0      158 14 3 0      128 5 4 1
2 1 5 0      1 15 4 0      1 16 2 0

```

```

61 17 2 0      37 16 3 0      113 16 2 0
59 12 6 1      82 14 5 1      148 16 3 0
18 2 5 0       1 12 4 0      243 8 8 0
168 18 3 0     1 16 3 0      78 15 6 0
175 13 5 0     80 16 5 0     27 9 4 0
22 16 2 0     105 5 6 1      96 12 3 1
131 3 2 0     15 2 7 1      9 13 5 0
12 2 14 1     8 6 3 0      100 14 3 0
4 16 3 0     151 16 2 0     31 16 3 0
125 11 2 0    130 13 5 0     112 16 3 0
140 11 5 0    93 16 3 0      1 9 3 0
52 6 5 1     20 9 6 0      91 12 5 1
73 1 5 1     35 13 3 0     143 3 9 0
61 1 4 0     97 16 3 0     139 10 3 1
136 15 4 0   131 13 5 0     121 3 3 1
177 14 2 0   68 10 5 0      9 17 2 0
139 6 10 1   2 17 2 0      140 15 4 0
72 15 5 0   2 13 3 0      120 8 5 1
51 9 7 0    102 13 3 0    130 1 4 1
114 8 7 1   81 1 4 0      118 16 3 0
118 16 4 0  17 10 4 0    195 17 2 0
159 13 4 0  18 11 4 0    15 16 5 0
158 15 4 0  127 12 4 0   87 16 4 0
206 10 4 0  11 15 3 0    178 15 4 0
157 13 3 1  26 13 7 0    120 13 2 0
42 6 7 1    36 13 4 0
;

proc genmod;
  model Kyphosis = Age StartVert NumVert
    / link=logit dist=binomial;
run;

```

Output 4.1.1. GENMOD Analysis: Partial Output

Comparing PROC GAM with PROC GENMOD							
The GENMOD Procedure							
Analysis Of Parameter Estimates							
Parameter	DF	Estimate	Standard Error	Wald 95% Confidence Limits		Chi-Square	Pr > ChiSq
Intercept	1	1.2497	1.2424	-1.1853	3.6848	1.01	0.3145
Age	1	-0.0061	0.0055	-0.0170	0.0048	1.21	0.2713
StartVert	1	0.1972	0.0657	0.0684	0.3260	9.01	0.0027
NumVert	1	-0.3031	0.1790	-0.6540	0.0477	2.87	0.0904
Scale	0	1.0000	0.0000	1.0000	1.0000		

NOTE: The scale parameter was held fixed.

The GENMOD analysis of the independent variable effects is shown in Output 4.1.1. Based on these results, the only significant factor is **StartVert** with odds ratio of -0.1972 . The variable **NumVert** has a p -value of 0.0904 with odds ratio of 0.3031.

The GENMOD procedure assumes a strict linear relationship between the response and the predictors. The following SAS statements use PROC GAM to investigate a less restrictive model, with moderately flexible spline terms for each of the predictors:

```

title 'Comparing PROC GAM with PROC GENMOD';
proc gam data=kyphosis;
  model Kyphosis=spline(Age,df=3) spline(StartVert,df=3)
        spline(NumVert,df=3) /dist = logist;
  output out=estimate p;
run;

```

The MODEL statement requests an additive model using a univariate BSPLINE for each term. The option `dist=logist` specifies a logistic model. Each term is fitted using a smoothing spline with three degrees of freedom. Although this might seem to be an unduly modest amount of flexibility, it is better to be conservative with a data set this small. An output data set `estimate` containing predicted values is requested by the OUTPUT statement.

Output 4.1.2 and Output 4.1.3 list the output from PROC GAM.

Output 4.1.2. Summary Statistics

Comparing PROC GAM with PROC GENMOD	
The GAM Procedure	
Dependent Variable: Kyphosis	
Smoothing Model Component: spline(Age) spline(StartVert) spline(NumVert)	
Iteration Summary and Fit Statistics	
Number of local score iterations	9
Local score convergence criterion	3.9786363E-9
Final number of backfitting iterations	1
Final backfitting criterion	5.2326788E-9
Final residual sum of squares	46.610928346
Summary of Input Data Set	
Number of Observations	83
Number of Missing Observations	0
Distribution	Binomial
Link Function	Logit

Output 4.1.3. Analysis of Model

```

Comparing PROC GAM with PROC GENMOD

The GAM Procedure
Dependent Variable: Kyphosis
Smoothing Model Component: spline(Age) spline(StartVert) spline(NumVert)

Regression Model Analysis
Parameter Estimates

Parameter          Parameter          Standard
                   Estimate          Error          t Value          Pr > |t|

Intercept          -2.01545          0.74274          -2.71           0.0083
L_Age              0.01213          0.00622          1.95           0.0552
L_StartVert       -0.18615          0.06061          -3.07           0.0030
L_NumVert          0.38347          0.15264          2.51           0.0142

Smoothing Model Analysis
Fit Statistics of Smoothing Components

Component          Smoothing          DF          GCV          No. of
                   Parameter          Unique
                   Obs.

spline(Age)        0.999996          3.000000          328.513619          66
spline(StartVert) 0.999551          3.000000          317.647039          16
spline(NumVert)    0.921758          3.000000          20.144078          10

Smoothing Model Analysis
Analysis of Deviance

Source          DF          Sum of
                   Squares          F Value          Pr > F

spline(Age)        3.000000          10.494366          16.44          0.0009
spline(StartVert) 3.000000          5.494965          8.61          0.0135
spline(NumVert)    3.000000          2.184514          3.42          0.3311
    
```

The critical part of the GAM results is the Analysis of Deviance table, shown in Output 4.1.3. For each smoothing effect in the model, this table gives an F -test comparing the deviance between the full model and the model without this variable. In this case, the analysis of deviance results indicates that the effect of Age and StartVert are highly significant, while the effect of NumVert is insignificant. Plots of predictions against predictor can be used to investigate why PROC GAM and PROC GENMOD produce different results.

The GAM statement requests an output data set of predicted values to be created. Since the estimate of the generalized additive model is the sum of functional estimates of individual predictors, plus a constant, the output data set will contain a column of partial prediction for each predictor. If requested, a Bayesian confidence interval or a point-wise standard-error band, as defined in Hastie and Tibshirani (1990), can be produced in the output data set.

Using the following statements, the data set estimate is plotted in Output 4.1.4:

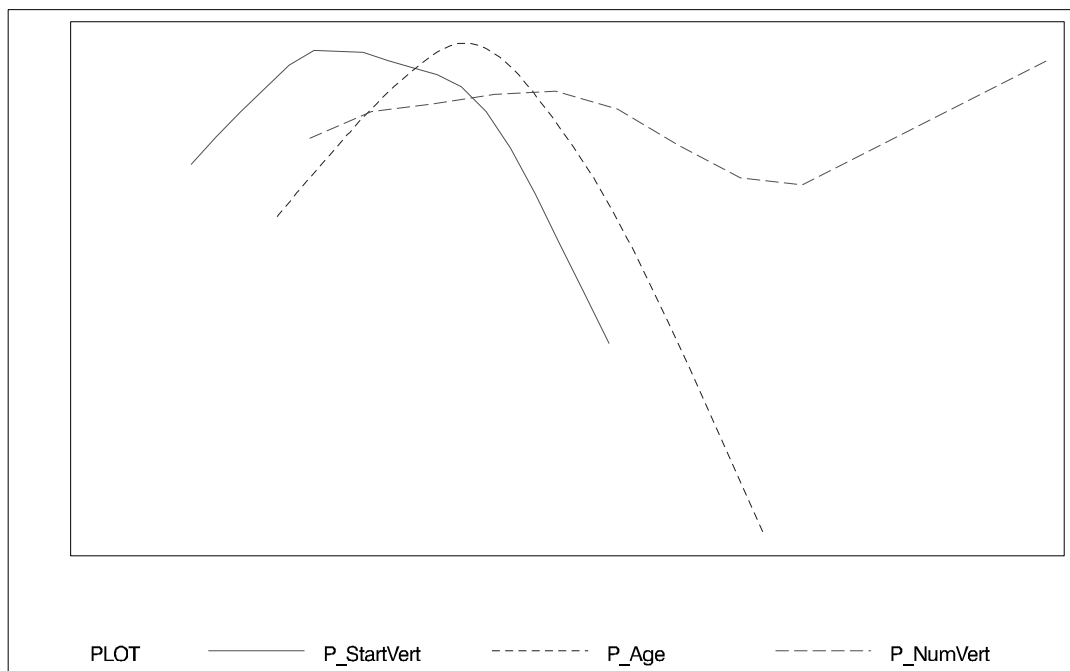
```

proc sort data=estimate(keep=StartVert P_StartVert)
  out =StartVert;
  by StartVert;
proc sort data=estimate(keep=Age      P_Age      )
  out =Age;
  by Age;
proc sort data=estimate(keep=NumVert  P_NumVert  )
  out =NumVert;
  by NumVert;
data Plot; merge StartVert Age NumVert;
proc standard m=0 s=1 data=Plot out=Plot;
  var StartVert Age NumVert;
run;

legend1 frame cframe=ligr cborder=black label=none
  position=center;
axis1 label=(angle=90 rotate=0 " ") minor=none
  value=NONE major=NONE;
axis2 minor=none label=(" ") major=NONE value=NONE;
symbol1 color=red   interpol=join value=none line=1;
symbol2 color=blue  interpol=join value=none line=2;
symbol3 color=green interpol=join value=none line=3;
proc gplot data=Plot;
  title;
  plot P_StartVert*StartVert=1
      P_Age      *Age      =2
      P_NumVert  *NumVert  =3 / overlay legend frame
      cframe=ligr vaxis=axis1 haxis=axis2;
run;

```

Output 4.1.4. Partial Prediction for Each Predictor



The plot shows that the partial predictions corresponding to both Age and StartVert have a strong quadratic pattern, while NumVert has a more complicated but weaker pattern. However, in the plot for NumVert, notice that about half the vertical range of the function is determined by the point at the upper extreme. It would be a good idea, therefore, to re-run the analysis without this point, to see how much it affects the conclusions. You can do this simply by including a WHERE clause when specifying the data set for the GAM procedure, as in the following code:

```

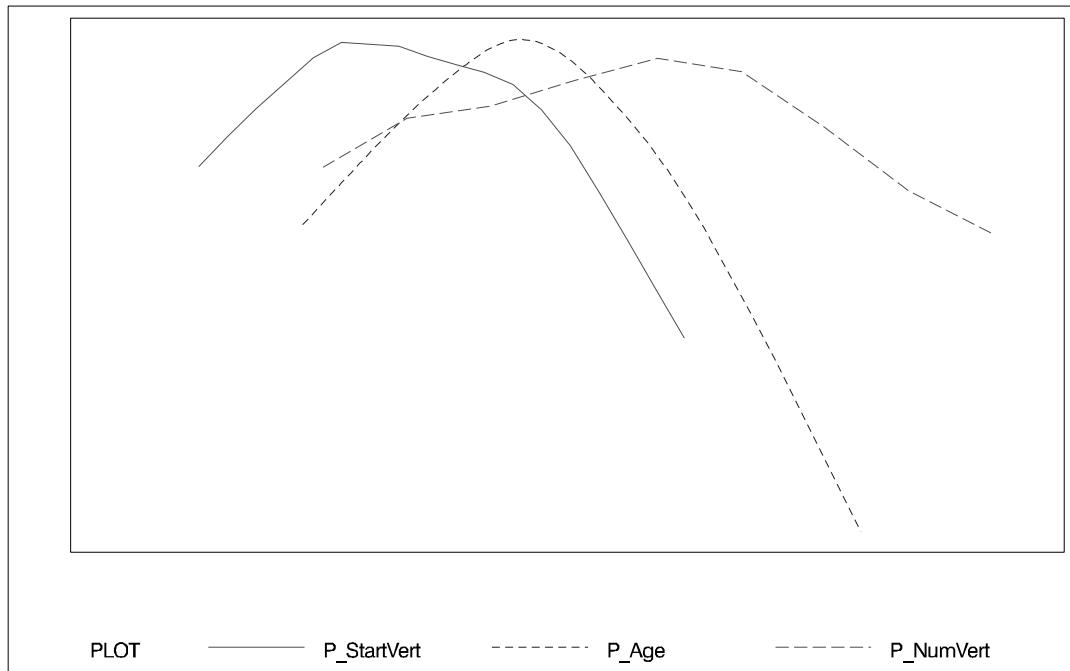
title 'Comparing PROC GAM with PROC GENMOD';
proc gam data=kyphosis(where=(NumVert^=14));
  model Kyphosis=spline(Age,df=3) spline(StartVert,df=3)
           spline(NumVert, df=3) /dist = logist;
  output out=estimate p;
run;

```

The analysis of deviance table from this re-analysis is shown in Output 4.1.5, and Output 4.1.6 shows the re-computed partial predictor plots.

Output 4.1.5. Analysis After Removing NumVert=14

Comparing PROC GAM with PROC GENMOD				
The GAM Procedure				
Dependent Variable: Kyphosis				
Smoothing Model Component: spline(Age) spline(StartVert) spline(NumVert)				
Smoothing Model Analysis				
Analysis of Deviance				
Source	DF	Sum of Squares	F Value	Pr > F
spline(Age)	3.000000	10.587568	16.90	0.0007
spline(StartVert)	3.000000	5.477104	8.74	0.0126
spline(NumVert)	3.000000	3.209100	5.12	0.1630

Output 4.1.6. Partial Prediction After Removing NumVert=14

After removing data point NumVert=14, the predictors `age` and `StartVert` are still significant and the variable `NumVert` still insignificant. Based on the plot in Output 4.1.6, the removed point has almost no effect on estimates of curve shape for variables `Age` and `StartVert`. But the removal has a dramatic effect on the variable `NumVert`: this curve for this variable `NumVert` now also seems quadratic, though it is much less pronounced than for the other two variables.

Having used the GAM procedure to discover an appropriate form of the dependence of `Kyphosis` on each of the three independent variables, you can use the GENMOD procedure to fit and assess the corresponding parametric model. The following code fits a GENMOD model with quadratic terms for all three variables, including tests for the joint linear and quadratic effects of each variable. The resulting contrast tests are shown in Output 4.1.7.

```

title 'Comparing PROC GAM with PROC GENMOD';
proc genmod data=kyphosis(where=(NumVert^=14));
  model kyphosis = Age      Age      *Age
                  StartVert StartVert*StartVert
                  NumVert  NumVert  *NumVert
                  /link=logit dist=binomial;
  contrast 'Age'      Age      1, Age*Age      1;
  contrast 'StartVert' StartVert 1, StartVert*StartVert 1;
  contrast 'NumVert'  NumVert  1, NumVert*NumVert  1;
run;

```

Output 4.1.7. Joint Linear and Quadratic Tests

Comparing PROC GAM with PROC GENMOD				
The GENMOD Procedure				
Contrast Results				
Contrast	DF	Chi-Square	Pr > ChiSq	Type
Age	2	13.63	0.0011	LR
StartVert	2	15.41	0.0005	LR
NumVert	2	3.56	0.1684	LR

The results for the quadratic GENMOD model are now quite consistent with the GAM results.

From this example, you can see that PROC GAM is very useful in visualizing the data and detecting the nonlinearity among the variables.

Example 4.2. Comparing PROC GAM with PROC TPSPLINE

This example compares the GAM procedure with the TPSPLINE procedure, another nonparametric procedure that fits a smooth surface to multivariate data. It does not assume additivity of the model and uses very general basis functions for model fitting, making the TPSPLINE procedure much slower than the GAM procedure. For more details about the TPSPLINE procedure, refer to “The TPSPLINE Procedure” in *SAS/STAT User’s Guide, Version 8*.

The data used here is also analyzed in “The TPSPLINE Procedure” in *SAS/STAT User’s Guide, Version 8*. It presents age-adjusted melanoma incidences for 37 years from the Connecticut Tumor Registry (Houghton, Flannery, and Viola 1980):

```

title 'Comparing PROC GAM with PROC TPSPLINE';
data melanoma;
  input year incidences @@;
  datalines;
1936 0.9 1937 0.8 1938 0.8 1939 1.3
1940 1.4 1941 1.2 1942 1.7 1943 1.8
1944 1.6 1945 1.5 1946 1.5 1947 2.0
1948 2.5 1949 2.7 1950 2.9 1951 2.5
1952 3.1 1953 2.4 1954 2.2 1955 2.9
1956 2.5 1957 2.6 1958 3.2 1959 3.8
1960 4.2 1961 3.9 1962 3.7 1963 3.3
1964 3.7 1965 3.9 1966 4.1 1967 3.8
1968 4.7 1969 4.4 1970 4.8 1971 4.8
1972 4.8
;
run;

```

The variable incidences records the number of melanoma cases per 100,000 people for the years 1936 to 1972.

Four to five degrees of freedom for each nonparametric term in a generalized additive model fits most data well. However, to select DF more objectively you can use the GCV option to minimize the generalized cross validation function, as shown in the following PROC GAM code:

```
proc gam data=melanoma;  
  model incidences = spline(year) /method = GCV;  
  output out=gam p;  
run;
```

The results are listed in Output 4.2.1 and Output 4.2.2.

Output 4.2.1. Summary Statistics

Comparing PROC GAM with PROC TPSPLINE	
The GAM Procedure	
Dependent Variable: incidences	
Smoothing Model Component: spline(year)	
Iteration Summary and Fit Statistics	
Final number of backfitting iterations	2
Final backfitting criterion	0
Final residual sum of squares	1.2242517494
Summary of Input Data Set	
Number of Observations	37
Number of Missing Observations	0
Distribution	Gaussian
Link Function	Identity

Output 4.2.2. Analysis of Model

```

Comparing PROC GAM with PROC TPSPLINE

The GAM Procedure
Dependent Variable: incidences
Smoothing Model Component: spline(year)

Regression Model Analysis
Parameter Estimates

Parameter          Parameter          Standard
                   Estimate          Error      t Value      Pr > |t|
Intercept          -212.69706          7.00491    -30.36       <.0001
L_year             0.11029             0.00358     30.77       <.0001

Smoothing Model Analysis
Fit Statistics of Smoothing Components

Component          Smoothing          DF              GCV              No. of
                   Parameter          DF              GCV              Unique
                   Parameter          DF              GCV              Obs.
spline(year)       0.634903          13.414936      0.088803         37

Smoothing Model Analysis
Analysis of Deviance

Source              DF              Sum of
                   DF              Squares      F Value      Pr > F
spline(year)       13.414936      2.736763     50.49       <.0001
    
```

Based on the summary of the model, the final model has a DF = 13.414936 and the nonparametric trend is highly significant. Note that this DF is much greater than the default value of 4, indicating that there is a great deal of structure in the yearly incidence rates of melanoma. A prediction plot should reveal the nature of this structure:

```

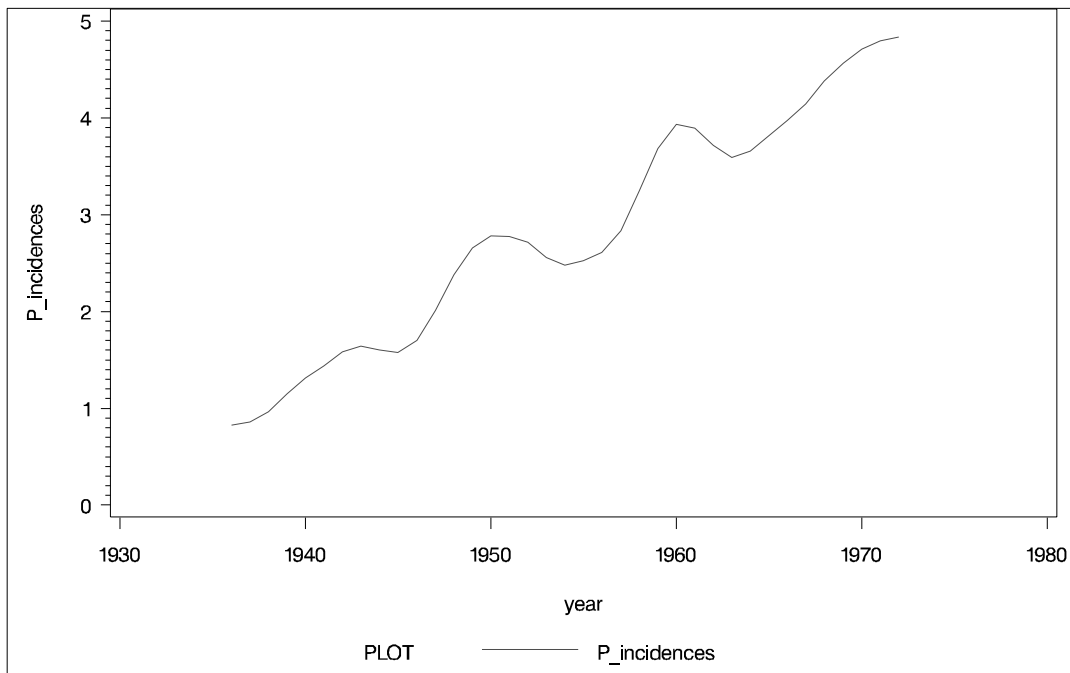
legend1 frame cframe=ligr cborder=black label=none
       position=center;
axis1  label=(angle=90 rotate=0);
axis2  minor=none;
symbol1 color=red interpol=join value=none line=1;

proc sort data=gam; by year;
proc gplot data=gam;
  title;
  plot p_incidences*year = 1 /overlay legend
       frame cframe=ligr vaxis=axis1 haxis=axis2;
run;
    
```

Output 4.2.3 shows the predicted melanoma rate over time. Two features stand out on this plot:

- Melanoma incidence on the whole increases over the period of the study.
- A strong periodic effect is evident, with a period of a little more than a decade. This has been attributed to the 11-year sunspot cycle: the more sunspots there are, the more melanoma cases there are likely to be.

Output 4.2.3. Predicted Melanoma Incidence Rates By Year



Since PROC TPSPLINE also fits a nonparametric model, PROC TPSPLINE and PROC GAM fits should be very similar for this univariate case. The following code produces the TPSPLINE analysis shown in Output 4.2.4:

```

title 'Comparing PROC GAM with PROC GENMOD';
proc tpspline data=melanoma;
  model incidences = (year);
  output out=tpspline p;
run;

```

Output 4.2.4. Analysis from PROC TPSPLINE

Comparing PROC GAM with PROC TPSPLINE	
The TPSPLINE Procedure	
Dependent Variable: incidences	
Summary of Input Data Set	
Number of Non-Missing Observations	37
Number of Missing Observations	0
Unique Smoothing Design Points	37
Summary of Final Model	
Number of Regression Variables	0
Number of Smoothing Variables	1
Order of Derivative in the Penalty	2
Dimension of Polynomial Space	2
Summary Statistics of Final Estimation	
log10(n*Lambda)	-0.0607
Smoothing Penalty	0.5171
Residual SS	1.2243
Tr(I-A)	22.5852
Model DF	14.4148
Standard Deviation	0.2328

The TPSPLINE model analysis shows that the DF for the model is 14.4148. This is consistent with the GAM model because the DF value in GAM excludes the degree of freedom of the linear L_{year} term. The OUTPUT statements in the PROC GAM and PROC TPSPLINE code create gam and tpspline data sets containing the predicted values for the respective procedures. You can use the following code to look at the values for the two procedures side by side:

```
data both; merge gam      (rename=(p_incidences=gam      ))
                tpspline(rename=(p_incidences=tpspline));
proc print data=both;
  var year gam tpspline;
run;
```

The results, the first ten of which are displayed in Output 4.2.5, show that PROC GAM and PROC TPSPLINE give essentially the same predictions for this problem.

Output 4.2.5. Melanoma Predictions for First Ten Years

Comparing PROC GAM with PROC TPSPLINE				
	Obs	year	gam	tpspline
	1	1936	0.82425	0.82424
	2	1937	0.85580	0.85580
	3	1938	0.96379	0.96379
	4	1939	1.15046	1.15046
	5	1940	1.31044	1.31044
	6	1941	1.43881	1.43881
	7	1942	1.58218	1.58218
	8	1943	1.64382	1.64382
	9	1944	1.60148	1.60148
	10	1945	1.57498	1.57499

References

- Allen, D.M. (1974), "The relationship between variable selection and data augmentation and a method of prediction," *Technometrics*, 16, 125–127.
- Bell, D., Walker, J., O'Connor, G., Orrel, J. and Tibshirani, R. (1989), "Spinal Deformation Following Multi-Level Thoracic and Lumbar Laminectomy in Children." Submitted for publication.
- Duchon, J. (1976), "Fonctions-Spline et Esperances Conditionnelles de Champs Gaussiens," *Ann. Sci. Univ. Clermont Ferrand II Math.*, 14, 19–27.
- Duchon, J. (1977), "Splines Minimizing Rotation-Invariant Semi-Norms in Sobolev Spaces," in *Constructive Theory of Functions of Several Variables*, eds. W. Schempp and K. Zeller, New York: Springer-Verlag, 85–100.
- Hastie, T.J. and Tibshirani, R.J. (1990), *Generalized Additive Models*, New York: Chapman and Hall.
- Houghton, A. N., Flannery, J. and Viola, M. V. (1980), "Malignant Melanoma in Connecticut and Denmark," *International Journal of Cancer*, 25, 95–104.
- Meinguet, J. (1979), "Multivariate Interpolation at Arbitrary Points Made Simple," *Journal of Applied Mathematics and Physics (ZAMP)*, 30, 292–304.
- Nelder, J.A. and Wedderburn, R.W.M. (1972) "Generalized Linear models" *Journal of the Royal Statistical Society, Series A*, 135, 370–384.
- SAS Institute Inc. (1999), *SAS/STAT User's Guide, Version 8*, Cary, NC: SAS Institute Inc.
- Socketk, E.B., Daneman, D., Clarson, C., and Ehrich, R.M. (1987), "Factors Affecting and Patterns of Residual Insulin Secretion During the First Year of Type I (Insulin Dependent) Diabetes Mellitus in Children," *Diabet*, 30, 453–459.
- Stone, C.J. (1985), "Additive Regression and Other Nonparametric Models," *Annals of Statistics*, 13, 689–705.
- Wahba, G. (1990), *Spline Models for Observational Data*, Philadelphia: Society for Industrial and Applied Mathematics.

- Wahba, G. and Wendelberger, J. (1980), "Some New Mathematical Methods for Variational Objective Analysis Using Splines and Cross Validation," *Monthly Weather Review*, 108, 1122–1145.