SAS/STAT® 9.2 User's Guide

The GAM Procedure

(Book Excerpt)

SAS® Documentation

This document is an individual chapter from *SAS/STAT® 9.2 User's Guide*.

The correct bibliographic citation for the complete manual is as follows: SAS Institute Inc. 2008. *SAS/STAT® 9.2 User's Guide*. Cary, NC: SAS Institute Inc.

SAS Institute Inc., SAS Campus Drive, Cary, North Carolina 27513.

1st electronic book, March 2008
2nd electronic book, February 2009

Chapter 36

The GAM Procedure

Contents

Overview: GAM Procedure **1832**

Getting Started: GAM Procedure **1832**

Syntax: GAM Procedure **1836**

- PROC GAM Statement 1836
- BY Statement 1838
- CLASS Statement 1838
- FREQ Statement 1839
- MODEL Statement 1840
- OUTPUT Statement 1844
- SCORE Statement 1845

Details: GAM Procedure **1846**

- Missing Values 1846
- Nonparametric Regression 1846
- Additive Models and Generalized Additive Models 1847
- Backfitting and Local Scoring Algorithms 1848
- Smoothers 1850
- Selection of Smoothing Parameters 1851
- Confidence Intervals for Smoothers 1853
- Distribution Family and Canonical Link 1854
- Dispersion Parameter 1855
- Computational Resources 1855
- Forms of Additive Models 1858
- Estimates from PROC GAM 1858
- ODS Table Names 1859
- ODS Graphics 1859

Examples: GAM Procedure **1860**

- Example 36.1: Generalized Additive Model with Binary Data 1860
- Example 36.2: Poisson Regression Analysis of Component Reliability . . . 1869
- Example 36.3: Comparing PROC GAM with PROC LOESS 1875

References **1887**

Overview: GAM Procedure

The GAM procedure 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 the following:

- an additivity assumption (Stone 1985) that enables relatively many nonparametric relationships to be explored simultaneously
- the distributional flexibility of generalized linear models (Nelder and Wedderburn 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. See the section "Nonparametric Regression" on page 1846 for more details on the form of generalized additive models.

The GAM procedure does the following:

- 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
- supports graphical displays produced through ODS Graphics

Getting Started: GAM Procedure

The following example illustrates the use of the GAM procedure to explore in a nonparametric 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
;
```

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 smoothing splines with the default of 4 degrees of freedom be used to model the effect of Age and BaseDeficit.

```
title 'Patterns of Diabetes';
proc gam data=diabetes;
    model logCP = spline(Age) spline(BaseDeficit);
run;
```

The results are shown in Figure 36.1 and Figure 36.2.

Figure 36.1 Summary Statistics

```
                          Patterns of Diabetes

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

                       Summary of Input Data Set

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

Figure 36.1 *continued*

```
                      Iteration Summary and Fit Statistics

           Final Number of Backfitting Iterations                     5
           Final Backfitting Criterion                     5.542745E-10
           The Deviance of the Final Estimate              0.4180791724
```

Figure 36.1 shows two tables. The first table summarizes the input data set and the distributional family used for the model; the second table summarizes the convergence criterion for backfitting.

Figure 36.2 Analysis of Model

```
                            Regression Model Analysis
                              Parameter Estimates

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

     Intercept                   1.48141       0.05120      28.93      <.0001
     Linear(Age)                 0.01437       0.00437       3.28      0.0024
     Linear(BaseDeficit)         0.00807       0.00247       3.27      0.0025

                            Smoothing Model Analysis
                      Fit Summary for Smoothing Components

                                                                         Num
                            Smoothing                                 Unique
 Component                  Parameter            DF            GCV       Obs

 Spline(Age)                 0.995582      3.000000       0.011675        37
 Spline(BaseDeficit)         0.995299      3.000000       0.012437        39

                             Smoothing Model Analysis
                               Analysis of Deviance

                                              Sum of
  Source                          DF         Squares    Chi-Square    Pr > ChiSq

  Spline(Age)                3.00000        0.150761       12.2605        0.0065
  Spline(BaseDeficit)        3.00000        0.081273        6.6095        0.0854
```

Figure 36.2 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. The second table is the summary of smoothing components of the nonparametric part of the model. By default, each smoothing component has approximately 4 degrees of freedom (DF). For univariate spline components, one DF is taken up by the (parametric) linear part of the model, so the remaining approximate DF is 3, and the main point of this table is to present the smoothing parameter values that yield this DF for each component. Finally, the third table is the "Analysis of Deviance" table for the nonparametric component of the model.

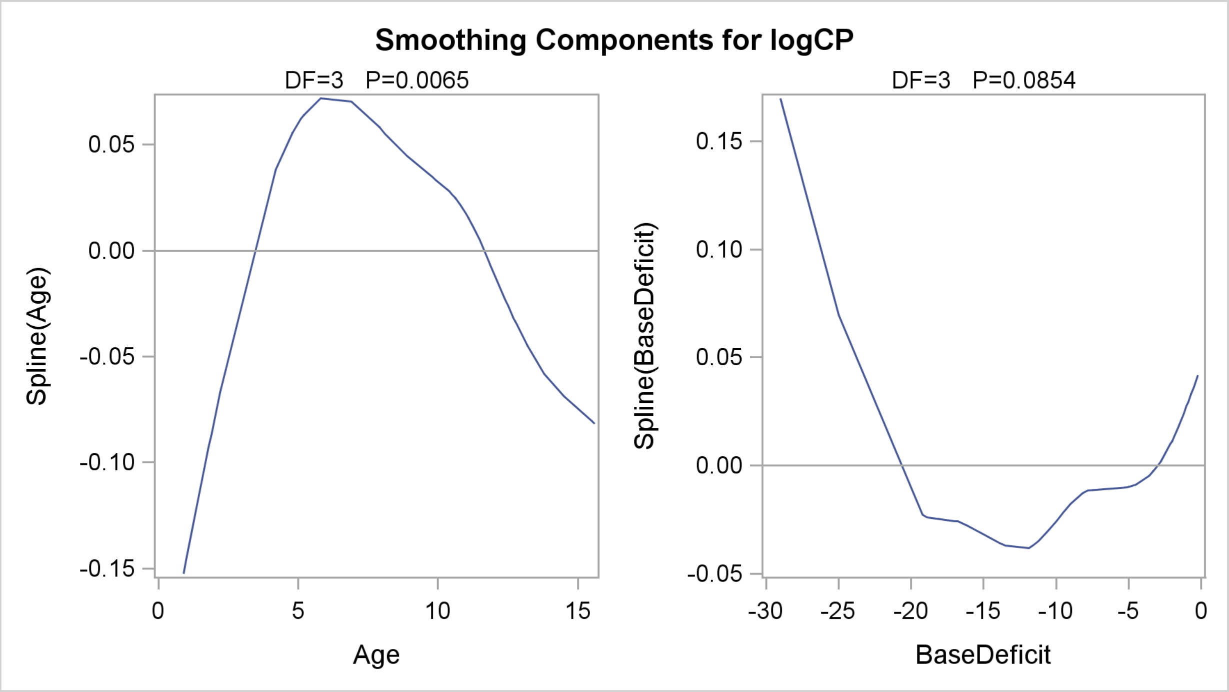
In order to explore the overall shape of the relationship between each factor and the response, use the graphics features of PROC GAM to plot the partial predictions, as in the following statements:

```
ods graphics on;

proc gam data=diabetes;
    model logCP = spline(Age) spline(BaseDeficit);
run;

ods graphics off;
```

These graphical displays are requested by specifying the ODS GRAPHICS statement. For general information about ODS Graphics, see Chapter 21, “Statistical Graphics Using ODS.” For specific information about the graphics available in the GAM procedure, see the section “ODS Graphics” on page 1859.

Both plots (Figure 36.3) 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.

Figure 36.3 Partial Predictions for Each Predictor



Syntax: GAM Procedure

The following statements are available in PROC GAM:

> **PROC GAM** < *options* > **;**
> **CLASS** *variable<(options)><variable<(options)>· · · ></options>* **;**
> **MODEL** *dependent</options> = < PARAM(effects) > < smoothing effects >< /options >* **;**
> **SCORE** *data=SAS-data-set out=SAS-data-set* **;**
> **OUTPUT** *< out=SAS-data-set > keyword=prefix < · · · keyword=prefix >* **;**
> **BY** *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** < *options* > **;**

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

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.

DESCENDING

DESC

reverses the sorting order of the classification variable (specified in the CLASS statement). If both the DESCENDING and ORDER= options are specified, PROC GAM orders the categories according to the ORDER= option and then reverses that order. This option has the same effect as the classification variable option DESCENDING in the CLASS statement and the response variable option DESCENDING in the MODEL statement.

ORDER=DATA | FORMATTED | FREQ | INTERNAL

specifies the sorting order for the levels of all classification variables (specified in the CLASS statement). This ordering determines which parameters in the model correspond to each level in the data. Note that the ORDER= option in the CLASS statement and the ORDER= response variable option in the MODEL statement override the ORDER= option in the PROC GAM statement.

PLOTS < **(***global-plot-options***)** > < **=** *plot-request* < **(***options***)** > >

PLOTS < **(***global-plot-options***)** > < **=(***plot-request* < **(***options***)** > < ⋯ *plot-request* < **(***options***)** > > **)** >

controls the plots produced through ODS Graphics. When you specify only one plot request, you can omit the parentheses around the plot request. Here are some examples:

```
      plots=all

      plots=components(commonaxes)

      plots(unpack)=components(commonaxes clm)
```

You must enable ODS Graphics before requesting plots as shown in the following example. With ODS Graphics enabled, the output graph by default is a panel of multiple plots of partial prediction curves of smoothing components, if PLOTS is not specified or no options are specified for PLOTS. For general information about ODS Graphics, see Chapter 21, "Statistical Graphics Using ODS."

```
   ods graphics on;

   proc gam data=test plots(unpack)=components(commonaxes clm);
      model z=spline(x) spline(y);
   run;

   ods graphics off;
```

Global Plot Options

The *global-plot-options* apply to all plots generated by the GAM procedure, unless altered by a *specific-plot-option*.

UNPACK

specifies that multiple smoothing component plots that are collected into graphics panels be displayed separately. Use this option if you want to access individual smoothing component plots within the panel.

Specific Plot Options

The following listing describes the specific plots and their options.

ALL

requests that all plots are produced.

NONE

suppresses all plots.

COMPONENTS | COMPONENT < **(***components-options***)** >

requests the SmoothingComponentPlot that displays a panel of smoothing component plots. The following *components-options* are available:

ADDITIVE requests that the additive component plots are produced for spline effects. The additive component plots combine the linear trend and the nonparametric prediction for each spline effect if its linear part is not specified explicitly in the MODEL statement.

CLM includes confidence limits in the smoothing component plots. By default, 95% confidence limits are produced, but you can change the significance level by specifying the ALPHA= option in the MODEL statement. Note that producing these limits can be computationally intensive for large data sets.

COMMONAXES specifies that smoothing component plots use a common vertical axis. This enables you to visually judge relative effect size.

UNPACK specifies that the smoothing components be displayed individually.

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 by 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 (according 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 by using the DATASETS procedure.

For more information about the BY statement, see in *SAS Language Reference: Concepts*. For more information about the DATASETS procedure, see the discussion in the *Base SAS Procedures Guide*.

CLASS Statement

CLASS *variable<(options)><variable<(options)>· · · ></options>* **;**

The CLASS statement names the classification variables to be used in the analysis. The CLASS statement must precede the MODEL statement. You can specify various *options* for each variable be enclosing them in parentheses after the variable name. You can also specify global *options* for the CLASS statement by placing them after a slash (/). Global *options* are applied to all the variables specified in the CLASS statement. If you specify more than one CLASS statement, the global *options* specified on any one CLASS statement apply to all CLASS statements. However, individual CLASS variable *options* override the global *options*.

DESCENDING

DESC

reverses the sorting order of the classification variable. If both the DESCENDING and ORDER= options are specified, PROC GAM orders the categories according to the ORDER= option and then reverses that order.

ORDER=DATA | FORMATTED | FREQ | INTERNAL

specifies the sorting order for the categories of categorical variables. This ordering determines which parameters in the model correspond to each level in the data. When the default ORDER=FORMATTED is in effect for numeric variables for which you have supplied no explicit format, the levels are ordered by their internal values. The following table shows how PROC GAM interprets values of the ORDER= option.

Value of ORDER=	Levels Sorted By
DATA	order of appearance in the input data set
FORMATTED	external formatted value, except for numeric variables with no explicit format, which are sorted by their unformatted (internal) value
FREQ	descending frequency count; levels with the most observations come first in the order
INTERNAL	unformatted value

By default, ORDER=FORMATTED. For FORMATTED and INTERNAL, the sort order is machine-dependent. For more information on sorting order, see the chapter on the SORT procedure in the *SAS Procedures Guide* and the discussion of BY-group processing in *SAS Language Reference: Concepts*.

TRUNCATE< *=n* >

specifies the length *n* of CLASS variable values to use in determining CLASS variable levels. If you specify TRUNCATE without the length *n*, the first 16 characters of the formatted values are used. When formatted values are longer than 16 characters, you can use this option to revert to the levels as determined in releases previous to SAS 9. The default is to use the full formatted length of the CLASS variable. The TRUNCATE option is available only as a global option.

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 by 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.

The FREQ statement is not available when a loess smoother is included in the model.

MODEL Statement

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

MODEL *event/trials=<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 classification variables or continuous variables. Classification variables must be declared with a CLASS statement. Interactions between variables can also be included as parametric effects. Multiple PARAM() statements are allowed in the MODEL statement. The syntax for the specification of effects is the same as for the GLM procedure (Chapter 39, "The GLM Procedure").

Only continuous variables can be specified in smoothing effects. Any number of smoothing effects can be specified, as follows:

Smoothing Effect	Meaning
SPLINE(variable <, DF=number>)	fit a smoothing spline with the variable and with DF=number
LOESS(variable <, DF=number>)	fit a local regression with the variable and with DF=number
SPLINE2(variable1, variable2 <,DF=number>)	fit a bivariate thin-plate smoothing spline with variable1 and variable2 and with DF=number

The number specified in the DF= option must be positive. If you specify neither the DF= option nor the METHOD=GCV in the MODEL statement, then the default used is DF=4. Note that for univariate spline components, a degree of freedom is removed by default to account for the linear portion of the model, so the value displayed in the "Fit Summary" and "Analysis of Deviance tables" will be one less than the value you specify.

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 by 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 by using the parametric effects as the linear part of the model.

Table 36.1 shows how to specify various models for a dependent variable y and independent variables x, x1, and x2.

Table 36.1 Syntax for Common GAM Models

Type of Model	**Syntax** for `model`	**Mathematical Form**
Parametric	`y=param(x1 x2)`	$E(Y) = \beta_0 + \beta_1 x_1 + \beta_2 x_2$
Nonparametric	`y=spline(x)`	$E(Y) = \beta_0 + \beta_1 x + s(x)$
Nonparametric	`y=loess(x)`	$E(Y) = \beta_0 + s(x)$
Semiparametric	`y=spline(x1) param(x2)`	$E(Y) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + s(x_1)$
Additive	`y=spline(x1) spline(x2)`	$E(Y) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + s_1(x_1) + s_2(x_2)$
Thin-plate spline	`y=spline2(x1,x2)`	$E(Y) = \beta_0 + s(x_1, x_2)$

Response Variable Options

Response variable options determine how the GAM procedure models probabilities for binary data.

You can specify the following options by enclosing them in parentheses after the response variable. See the section "CLASS Statement" on page 1838 for more detail.

DESCENDING

DESC

reverses the order of the response categories. If both the DESCENDING and ORDER= options are specified, PROC GAM orders the response categories according to the ORDER= option and then reverses that order.

EVENT='*category*' | *keyword*

specifies the event category for the binary response model. PROC GAM models the probability of the event category. You can specify the value (formatted, if a format is applied) of the event category in quotes, or you can specify one of the following keywords. The default is EVENT=FIRST.

FIRST

designates the first ordered category as the event.

LAST

designates the last ordered category as the event.

One of the most common sets of response levels is $\{0, 1\}$, with 1 representing the event for which the probability is to be modeled. Consider the example where Y takes the value 1 and 0 for event and nonevent, respectively, and X is the explanatory variable. To specify the value 1 as the event category, using the following MODEL statement:

```
model Y (event='1') = X;
```

ORDER=DATA | FORMATTED | FREQ | INTERNAL

specifies the sort order for the levels of the response variable. When ORDER=FORMATTED (the default) for numeric variables for which you have supplied no explicit format (that is, for which there is no corresponding FORMAT statement in the current PROC GAM run or in the DATA step that created the data set), the levels are ordered by their internal (numeric) value. If you specify the ORDER= option in the MODEL statement and the ORDER= option in the CLASS statement, the former takes precedence. The following table shows the interpretation of the ORDER= values:

Value of ORDER=	Levels Sorted By
DATA	order of appearance in the input data set
FORMATTED	external formatted value, except for numeric variables with no explicit format, which are sorted by their unformatted (internal) value
FREQ	descending frequency count; levels with the most observations come first in the order
INTERNAL	unformatted value

By default, ORDER=FORMATTED. For the FORMATTED and INTERNAL values, the sort order is machine-dependent.

For more information about sort order, see the chapter on the SORT procedure in the *Base SAS Procedures Guide* and the discussion of BY-group processing in *SAS Language Reference: Concepts*.

Model Options

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. See the section "OUTPUT Statement" on page 1844 for more information about the OUTPUT statement.

ANODEV=*type*

specifies the *type* of method to be used to produce the "Analysis of Deviance" table for smoothing effects. The available choices are as follows:

REFIT — specifies that PROC GAM perform χ^2 tests by fitting nested GAM models. This is the default choice if you do not specify the ANODEV= option. This choice requires fitting separate GAM models where one smoothing term is omitted from each model.

NOREFIT — specifies that PROC GAM perform approximate tests of smoothing effects. To test each smoothing effect, a weighted least squares model is fitted to the remaining parametric part of the model while keeping other nonlinear smoothers fixed. For details, see Hastie (1991). This choice requires only a single GAM fitting to be performed, which reduces the time of the procedure.

NONE requests that the procedure not produce the "Analysis of Deviance" table for smoothing effects.

DIST=*distribution-id*

LINK=*distribution-id*

specifies the distribution family used in the model. The choices for *distribution-id* are displayed in Table 36.2. See "Distribution Family and Canonical Link" on page 1854 for more information.

Table 36.2 Distribution Families for GAM Models

DIST=	Distribution	Response Data Type
GAUSSIAN | GAUS | NORM	Normal (Gaussian)	continuous variables
BINOMIAL | LOGI | BIN	Binomial	binary variables
POISSON | POIS | LOGL	Poisson	nonnegative discrete variables
GAMMA | GAMM	Gamma	positive continuous variables
IGAUSSIAN | IGAU | INVG	Inverse gaussian	positive continuous variables

Canonical link functions are used with those distributions. Although theoretically, alternative links are possible, with nonparametric regression models the final fit is relatively insensitive to the precise choice of link functions. Therefore, only the canonical link for each distribution family is implemented in PROC GAM. The loess smoother is not available for DIST=BINOMIAL when the number of trials is greater than 1.

EPSILON=*number*

specifies the convergence criterion for the backfitting algorithm. The default value is 1E−8.

EPSSCORE=*number*

specifies the convergence criterion for the local scoring algorithm. The default value is 1E−8.

ITPRINT

produces an iteration summary table for the smoothing effects when doing backfitting and local scoring.

MAXITER=*number*

specifies the maximum number of iterations for the backfitting algorithm. The default value is 50.

MAXITSCORE=*number*

specifies the maximum number of iterations for the local scoring algorithm. The default value is 100.

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. See the section "Selection of Smoothing Parameters" on page 1851 for more details on the GCV method.

OUTPUT Statement

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

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

All the variables in the original data set are included in the new data set, along with the variables created by specifying *keywords* in the OUTPUT statement. These new variables contain the values of a variety of statistics and diagnostic measures that are calculated for each observation in the data set. If no *keywords* are present, the OUT= data set contains only the original data set and predicted values. The predicted values include the linear predictor for the response and each smoothing term in the model. When you specify a distribution family with the DIST= or LINK= option in the MODEL statement, predicted response values after applying the inverse link function are also included. Predicted values are computed for observations with missing response values whose values of the specified explanatory variables are nonmissing, and whose values of the specified smoothing variables are within the smoothing ranges of the fitted model.

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 < =prefix>

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

PREDICTED	predicted values for each smoothing component and overall predicted values on the response scale at design points
LINP	linear prediction values on the link scale at design points
UCLM	upper confidence limits for each predicted smoothing component
LCLM	lower confidence limits for each predicted smoothing component
ADIAG	diagonal element of the hat matrix associated with the observation for each smoothing spline component
RESIDUAL	residual standardized by its weights
STD	standard deviation of the prediction for each smoothing component
ALL	all statistics in this list

The names of the new variables that contain the statistics are formed by concatenating the user supplied *prefix* and the corresponding variable names. If you do not specify a *prefix*, the names are formed by using default prefixes listed in the following table.

The default prefixes for the new variables are as follows:

Keyword	Prefix
PRED	P_
LINP	LINP_
UCLM	UCLM_
LCLM	LCLM_
ADIAG	ADIAG_
RESID	R_
STD	STD_ (for spline)
	STDP_ (for loess)

For example, suppose that you have a dependent variable y and an independent smoothing variable x, and you specify the keywords PRED=MyP_ and ADIAG=MyA_. In this case, in addition to the variables in the input data set, the output SAS data set will contain the variables MyP_y, MyP_x, and MyA_x. If the keywords PRED and ADIAG are specified without prefixes, the output SAS data set will contain the variables P_y, P_x, and ADIAG_x.

SCORE Statement

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

The SCORE statement calculates predicted values for a new data set. All the variables in the named data set are included in the OUT= data set, along with the predicted values. The predicted values consist of predicted responses after the inverse link function transformation, predicted values of all smoothing terms, and predicted values on the link scale. Predicted values are computed for observations with missing response values whose values of the specified explanatory variables are nonmissing, and whose values of the specified smoothing variables are within the smoothing ranges of the fitted model. The predicted variables use the same naming convention as the OUTPUT statement. If you have multiple data sets to predict, you can specify multiple SCORE statements.

The following options 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: GAM Procedure

Missing Values

PROC GAM excludes any observation with missing values for an explanatory variable or dependent variable when fitting a model. However, all observations in the input data set are included in the output data set with an OUTPUT statement.

Nonparametric Regression

Nonparametric regression relaxes the usual assumption of linearity and enables you to explore the data more flexibly, 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. These estimates contain information about the relationship between the dependent and independent variables, and the information 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 by 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 nonlinear 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 $X_1, X_2, \cdots, X_p$ 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 $X_1, X_2, \cdots, X_p$. The standard linear regression model assumes a linear form for the dependency of Y on X:

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_p X_p + \epsilon$$

where $E(\epsilon) = 0$ and $\mathrm{Var}(\epsilon) = \sigma^2$. Given a sample, estimates of $\beta_0, \beta_1, \cdots, \beta_p$ are usually obtained by the least squares method.

The additive model generalizes the linear model by modeling the dependency as

$$Y = s_0 + s_1(X_1) + s_2(X_2) + \cdots + s_p(X_p) + \epsilon$$

where $s_i(X), i = 1, 2, \ldots, p$, are smooth functions, $E(\epsilon) = 0$ and $\mathrm{Var}(\epsilon) = \sigma^2$.

In order to be estimable, the smooth functions s_i have to satisfy standardized conditions such as $E\left(s_i(X_i)\right) = 0$. 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 might 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.

Like 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 $X_1, X_2, \cdots, X_p$ by a link function g. The quantity

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

defines the additive component, where $s_1(\,), \cdots, s_p(\,)$ are smooth functions, and the relationship between μ and η is defined by $g(\mu) = \eta$. The most commonly used link function is 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 and visualizing the relationship between the dependent variable and the independent variables.

Backfitting and Local Scoring Algorithms

Much of the development and notation in this section follows Hastie and Tibshirani (1986). Consider the estimation of the smoothing terms $s_0, s_1(\,), \cdots, s_p(\,)$ in the additive model

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

where $E\left(s_i(X_i)\right) = 0$ for every i. 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 backfitting algorithm is a general algorithm that can fit an additive model with any regression-type fitting mechanisms.

Define the jth set of partial residuals 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 to estimate each smoothing function $s_j(\,)$ given estimates $\{\hat{s}_i(\,), i \neq j\}$ for all the others. The resulting iterative procedure is known as the backfitting algorithm (Friedman and Stuetzle 1981). The following formulation is taken from Hastie and Tibshirani (1986).

The Backfitting Algorithm

The unweighted form of the backfitting algorithm is as follows:

```
1. Initialization:
   $s_0 = E(Y), s_1^{(1)} = s_2^{(1)} = \cdots = s_p^{(1)} = 0, m = 0$
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:
   $RSS = \frac{1}{n}\left\| Y - s_0 - \sum_{j=1}^{p} s_j^{(m)}(X_j) \right\|^2$ fails to decrease, or satisfies the convergence criterion.
```

In the preceding notation, $s_j^{(m)}(\ )$ denotes the estimate of $s_j(\ )$ at the mth iteration. It can be shown that with many smoothers (including linear regression, univariate and bivariate splines, and combinations of these), RSS never increases at any step. This implies that the algorithm always converges (Hastie and Tibshirani, 1986). Note, however, that for distributions other than Gaussian, numerical instabilities with weights can cause convergence problems. Even when the algorithm converges, 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 backfitting algorithm has the same form as for the unweighted case, except that the smoothers are weighted. In PROC GAM, weights are used with non-Gaussian data in the local scoring procedure described later in this section.

The GAM procedure uses the following condition as the convergence criterion for the backfitting algorithm:

$$\frac{\sum_{j=1}^{n}\sum_{i=1}^{k}\left(s_i^{(m-1)}(x_j)-s_i^{(m)}(x_j)\right)^2}{1+\sum_{j=1}^{n}\sum_{i=1}^{k}\left(s_i^{(m-1)}(x_j)\right)^2}\le\epsilon$$

where $\epsilon = 10^{-8}$ by default; you can change this with the EPSILON= option in the MODEL statement.

The algorithm described so far fits only 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 the form $\alpha + \sum_j X_j\beta_j$ with the additive form $\alpha + \sum_j f_j(X_j)$. See "Generalized Linear Models Theory" on page 1962 in Chapter 37, "The GENMOD Procedure," for more information.

PROC GAM fits generalized additive models by using a modified form of adjusted dependent variable regression, as described for generalized linear models in McCullagh and Nelder (1989), with the additive predictor taking the role of the linear predictor. Hastie and Tibshirani (1986) call this the *local scoring algorithm*. Important components of this algorithm depend on the link function for each distribution, as shown in the following table.

Distribution	**Link**	**Adjusted Dependent (Z)**	**Weights (w)**
Normal	μ	y	1
Binomial	$\log\left(\frac{\mu}{n-\mu}\right)$	$\eta + (y-\mu)/n\mu(1-\mu)$	$n\mu(1-\mu)$
Gamma	$-1/\mu$	$\eta + (y-\mu)/\mu^2$	μ^2
Poisson	$\log(\mu)$	$\eta + (y-\mu)/\mu$	μ
Inverse gaussian	$1/\mu^2$	$\eta - 2(y-\mu)/\mu^3$	$\mu^3/4$

Once the distribution and hence these quantities are defined, the local scoring algorithm proceeds as follows.

The General Local Scoring Algorithm

1. **Initialization:**
   $s_i = g(E(y)), s_1^0 = s_2^0 = \cdots = s_p^0 = 0, m = 0$

2. **Iterate:**
   $m = m + 1$;
   Form the predictor η_i, mean μ_i, weights w_i, and adjusted dependent variable z_i based on the previous iteration:

$$\begin{aligned}
\eta_i^{(m-1)} &= s_0 + \sum_{j=1}^{p} s_j^{(m-1)}(x_{ij}) \\
\mu_i^{(m-1)} &= g^{-1}\left(\eta_i^{(m-1)}\right) \\
w_i &= \left(V_i^{(m-1)}\right)^{-1} \cdot \left(\frac{\partial \mu_i}{\partial \eta_i}\right)^2_{(m-1)} \\
z_i &= \eta_i^{(m-1)} + \left(y_i - \mu_i^{(m-1)}\right) \cdot \left(\frac{\partial \mu_i}{\partial \eta_i}\right)_{(m-1)}
\end{aligned}$$

   where $V_i^{(m-1)}$ is the variance of Y at $\mu_i^{(m-1)}$. Fit an additive model to Z by using the backfitting algorithm with weights W to obtain estimated functions $s_j^{(m)}(\,)$;

3. **Until:**
   The convergence criterion is satisfied or the deviance fails to decrease. The deviance is an extension to generalized linear models of the RSS; see "Goodness of Fit" on page 1968 in Chapter 37, "The GENMOD Procedure," for a definition.

The GAM procedure uses the following condition as the convergence criterion for local scoring:

$$\frac{\sum_{j=1}^{n} \sum_{i=1}^{k} w(x_i) \left(s_i^{(m-1)}(x_j) - s_i^{(m)}(x_j)\right)^2}{\sum_{j=1}^{n} \sum_{i=1}^{k} w(x_i) + \sum_{j=1}^{n} \sum_{i=1}^{k} w(x_i) \left(s_i^{(m-1)}(x_j)\right)^2} \le \epsilon^s$$

where $\epsilon^s = 10^{-8}$ by default; you can change this with the EPSSCORE= option in the MODEL statement.

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

Smoothers

A smoother is a tool for summarizing the trend of a response measurement Y as a function of one or more predictor measurements $X_1, \cdots, X_p$. 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 does not assume a rigid form for the dependence of Y on $X_1, \cdots, X_p$. 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

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

where λ is a fixed constant and $a \le x_1 \le \cdots \le x_n \le b$. 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 value $\lambda/(1+\lambda)$ is the *smoothing parameter*. When λ is large, the smoothing parameter is close to 1, producing a smoother curve; small values of λ, corresponding to smoothing parameters near 0, are apt to produce rougher curves, more nearly interpolating the data.

Local Regression (Experimental)

Local regression was proposed by Cleveland, Devlin, and Grosse (1988). The idea of local regression is that at a predictor x, the regression function $\eta(x)$ can be locally approximated by the value of a function in some specified parametric class. Such a local approximation is obtained by fitting a regression surface to the data points within a chosen neighborhood of the point x. A weighted least squares algorithm is used to fit linear functions of the predictors at the centers of neighborhoods. The radius of each neighborhood is chosen so that the neighborhood contains a specified percentage of the data points. The smoothing parameter for the local regression procedure, which controls the smoothness of the estimated curve, is the fraction of the data in each local neighborhood. Data points in a given local neighborhood are weighted by a smooth decreasing function of their distance from the center of the neighborhood. See Chapter 50, “The LOESS Procedure,” for more details.

Thin-Plate Smoothing Spline

The thin-plate smoothing spline is a multivariate version of the cubic 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). See Chapter 89, “The TPSPLINE Procedure,” 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

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

where $\hat{\eta}_\lambda^{(-i)}(x_i)$ indicates the fit at x_i, computed by leaving out the ith 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 ith diagonal element of $A(\lambda)$. 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} individually. 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

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

The GCV formula simply replaces the a_{ii} with $\mathrm{tr}(A(\lambda))/n$. 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. Specify the METHOD=GCV option in the MODEL statement in order to use the GCV function to choose the smoothing parameters.

Degrees of Freedom

The estimated GAM model can be expressed as

$$\hat{\eta}(X) = \hat{s}_0 + \sum_{i=1}^{p} A_i(y,\lambda)Y$$

Because the weights are calculated based on previous iteration during the local scoring iteration, the matrices A_i might depend on Y for non-Gaussian data. However, for the final iteration, the A_i matrix for the spline smoothers has the same role as the projection matrix in linear regression; therefore, nonparametric degrees of freedom (DF) for the ith spline smoother can be defined as

$$DF(\text{spline smoother}) = \mathrm{tr}(A_i(y,\lambda))$$

For loess smoothers A_i is not symmetric and so is not a projection matrix. In this case PROC GAM uses

$$DF(\text{loess smoother}) = \text{tr}(A_i(y,\lambda)'A_i(y,\lambda))$$

The GAM procedure gives you the option of specifying the degrees of freedom for each individual smoothing component. If you choose a particular value for the degrees of freedom, then during every local scoring iteration the procedure will search for a corresponding smoothing parameter lambda that yields the specified value or comes as close as possible. The final estimate for the smoother during this local scoring iteration will be based on this lambda. Note that for univariate spline components, an additional degree of freedom is removed by default to account for the linear portion of the model, so the value displayed in the "Fit Summary" and "Analysis of Deviance" tables will be one less than the value you specify.

Confidence Intervals for Smoothers

In the GAM procedure, curvewise confidence intervals for smoothing splines and pointwise confidence intervals for loess are provided in the output data set.

Curvewise Confidence Interval for Smoothing Spline

Viewing the spline model as a Bayesian model, Wahba (1983) proposes Bayesian confidence intervals for smoothing spline estimates as follows:

$$\hat{s}_\lambda(x_i) \pm z_{\alpha/2}\sqrt{\hat{\sigma}^2 a_{ii}(\lambda)}$$

where $a_{ii}(\lambda)$ is the ith diagonal element of the $A(\lambda)$ matrix and $z_{\alpha/2}$ is the $1-\alpha/2$ quantile of the standard normal distribution. The confidence intervals are interpreted as intervals "across the function" as opposed to pointwise intervals.

Suppose that you fit a spline estimate to experimental data that consists of a true function f and a random error term, ϵ_i. In repeated experiments, it is likely that about $100(1-\alpha)\%$ of the confidence intervals cover the corresponding true values, although some values are covered every time and other values are not covered by the confidence intervals most of the time. This effect is more pronounced when the true response curve or surface has small regions of particularly rapid change.

Pointwise Confidence Interval for Loess Smoothers

As defined in Cleveland, Devlin, and Grosse (1988), a standardized residual for a loess smoother follows a t distribution with ρ degrees of freedom, where ρ is called the "lookup degrees of freedom," defined as

$$\rho = \delta_1^2/\delta_2$$

where $\delta_1 = \operatorname{tr}\left((I - A(\lambda))^T (I - A(\lambda))\right)$ and $\delta_2 = \operatorname{tr}\left((I - A(\lambda))^T (I - A(\lambda))\right)^2$. Therefore an approximate pointwise confidence interval at x_i is

$$\hat{s}_\lambda(x_i) \pm t_{\alpha/2,\rho}\hat{\sigma}(x_i)$$

where $t_{\alpha/2,\rho}$ is the $1-\alpha/2$ quantile of the t distribution with ρ degrees of freedom and $\hat{\sigma}(x_i)$ is the estimate of the standard deviation.

Distribution Family and Canonical Link

In general, there is not just one reasonable link function for a given response variable distribution. For parametric models, the choice of link function can lead to substantively different estimates and tests. However, the inherent flexibility of nonparametric models makes them less likely to be sensitive to the precise choice of link function. Thus, for simplicity and computational efficiency, the GAM procedure uses only the canonical link for each distribution, as discussed in the following sections.

The Gaussian Model

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

The Binomial Model

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

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

By default, PROC GAM models the probability of response level with *lower ordered value*. Ordered values are assigned to response levels in ascending sorted order and are displayed in the “Response Profiles” table. For binary data, if your event category has a higher Ordered Value, then by default the nonevent is modeled. The effect of modeling the nonevent is to change the signs of the estimated coefficients for linear terms in the model for the event. You can change which probability is modeled by specifying the EVENT=, DESCENDING, or ORDER= response variable options in the MODEL statement.

The Poisson Model

The link function for the Poisson model is the log function. Assuming that the mean of the Poisson distribution is $\mu(x)$, the dependence of $\mu(x)$ and independent variables $x_1, \cdots, x_k$ is

$$g(\mu) = \log(\mu) = \eta$$

The Gamma Model

Let the mean of the gamma distribution be $\mu(x)$. The canonical link function for the gamma distribution is $-1/\mu(x)$. Therefore, the relationship between $\mu(x)$ and the independent variables $x_1, \cdots, x_k$ is

$$g(\mu) = -\frac{1}{\mu} = \eta$$

The Inverse Gaussian Model

Let the mean of the inverse gaussian distribution be $\mu(x)$. The canonical link function for inverse gaussian distribution is $1/\mu^2$. Therefore, the relationship between $\mu(x)$ and the independent variables $x_1, \cdots, x_k$ is

$$g(\mu) = \frac{1}{\mu^2} = \eta$$

Dispersion Parameter

Continuous distributions in the exponential family (Gaussian, gamma, and inverse gaussian) have a dispersion parameter that can be estimated by the scaled deviance. For these continuous response distributions, PROC GAM incorporates this dispersion parameter estimate into standard errors of the parameter estimates, prediction standard errors of spline components, and chi-square statistics. The discrete distributions used in GAM (binomial and Poisson) do not have a dispersion parameter. For more details on the distributions, dispersion parameter, and deviance, see "Generalized Linear Models Theory" on page 1962 in Chapter 37, "The GENMOD Procedure."

Computational Resources

Since PROC GAM implements a doubly iterative method (inner backfitting iterations within each local scoring iteration), data are accessed multiple times in performing a fit. To expedite the data access, PROC GAM keeps the data used in the analysis in memory.

Let

n = number of observations used in the analysis
p_r = number of parametric variables
p_s = number of univariate spline smoothers
p_l = number of loess smoothers
p_b = number of bivariate thin-plate spline smoothers
$p = p_r + p_s + p_l + p_b$
$p_n = p_s + p_l + p_b$
m = maximum number of iterations for the backfitting algorithm

In addition to the space to store the data ($8np$ bytes), the minimum working space (in bytes) needed for fitting a model using PROC GAM is

$$(16+8p_r)(n+2p_r)+(160+48p+16p_s+8p_b+8p_l)n+8p+32p_b+32p_s+8m+8n+(4n+4)p_s+4$$

plus $80 + 120n + 8n^2 + 8p_b$ bytes of memory for fitting bivariate thin-plate smoothing spline variables and $48n + 16p_l$ bytes of memory for fitting loess variables. If confidence limits are requested, additional memory is required.

It is difficult to provide accurate estimates of the time required to fit a GAM model. Both the backfitting algorithm and the local scoring algorithm are iterative techniques whose convergence rates depend on the particular data being analyzed. Furthermore, the time required depends on the types of smoothers that you specify, as well as on the inferential information you request.

You can estimate the time required for problems with a larger number of observations by observing the time required for smaller problems and then using the following growth rules (obtained using by simulations) that show that the time required grows proportionally with the following:

- n^3 when at least one bivariate thin-plate spline is used
- $n^{3/2}$ when only loess smoothers are used
- n when only univariate smoothing splines are used

For additive models (models with Gaussian response distribution) with a fixed number of observations, the time required is roughly proportional to $p_n^{3/2}$. For generalized additive models (models with non-Gaussian distributions), the computation time grows more rapidly as p_n increases. This is harder to quantify as it depends on the distribution family and the number of iterations required for the local scoring algorithm to converge.

Figure 36.4 Feasible Problem Sizes for Different Smoothers

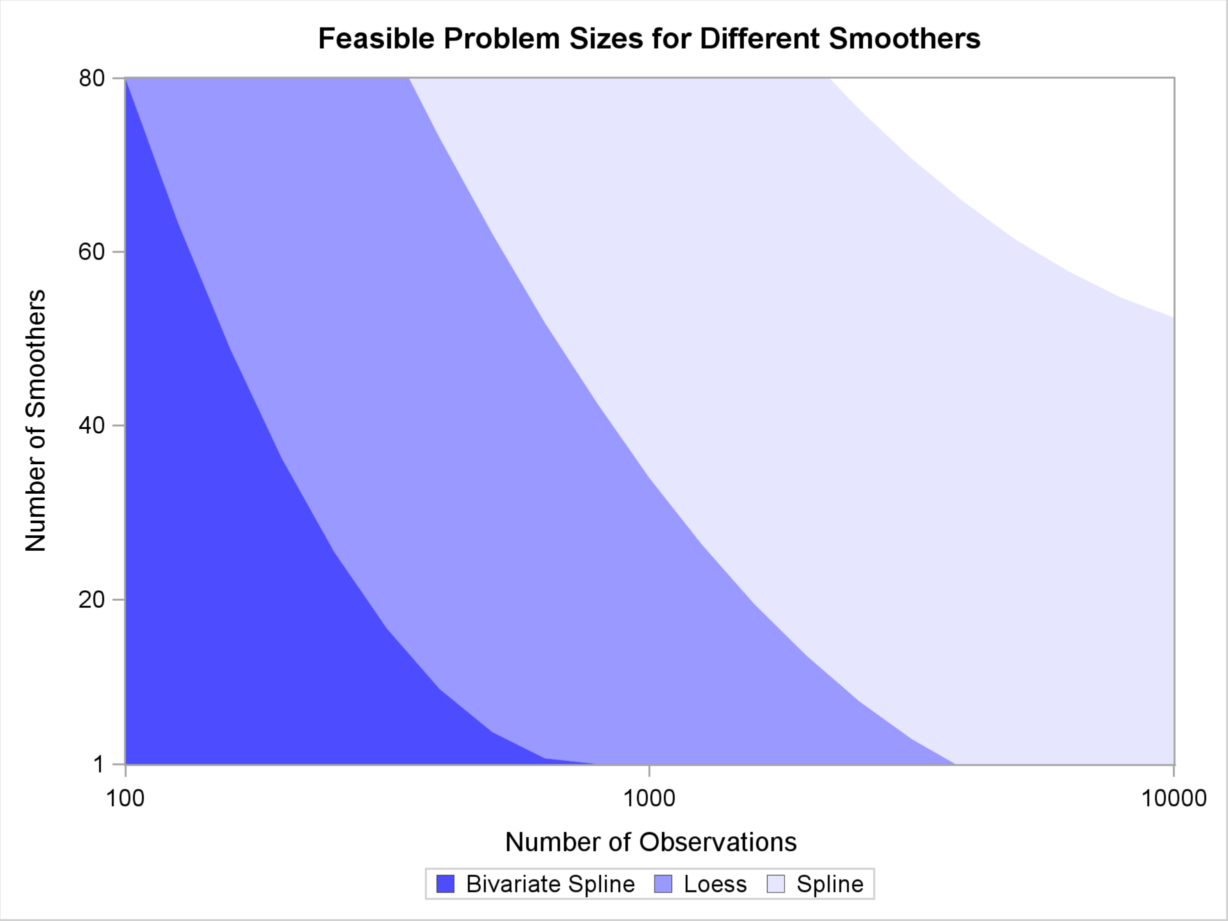


Figure 36.4 shows a rough estimation of feasible sizes for the smoothers that you can use, as a function of the number of observations and number of smoothing components. This figure depicts the regions where you can expect a single fit of an additive model to finish within a few minutes on a typical Pentium 4 system.

Note that the times reflected in Figure 36.4 are based on fitting additive models (no local scoring iterations) when no analysis of deviance or confidence limits are computed. The time required for fitting generalized additive models grows proportionally with the number of the local scoring iterations. Furthermore, analysis of deviance (if you do not request the fast approximations with the ANODEV option) requires fitting multiple GAM models as each smoothing component is omitted sequentially, and so the time estimates need to be multiplied by the number of smoothing components when analysis of deviance is performed. Finally computation of confidence limits for each individual smoother increases the time required, especially when loess smoothers are utilized.

For univariate spline smoothers, subject to the aforementioned caveats, problems that correspond to all shaded regions in Figure 36.4 can be completed within a few minutes. For univariate loess smoothers, the two darkest regions are feasible. For bivariate spline smoothers, problems that correspond to only the darkest shading can be completed in the order of a few minutes. The problems that correspond to the upper right unshaded region might be possible, but they require long computation times.

Forms of Additive Models

Suppose that y is a continuous variable and x1 and x2 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:

$$\eta(x_1, x_2) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + s_1(x_1) + s_2(x_2)$$

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 x1 and an unknown functional relation with x2:

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

If you want to fit a model containing a functional two-way interaction between x1 and x2, 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.

Estimates from PROC GAM

PROC GAM provides the capability to fit both nonparametric and semiparametric models. So that you can better understand the underlying trend of any given factor, PROC GAM separates the linear trend from any general nonparametric trend during the fitting as well as in the final report. This makes it easy to determine whether the significance of a smoothing variable is associated with a simple linear trend or a more complicated pattern.

For example, suppose you want to fit a semiparametric model as

$$y = \alpha_0 + \alpha_1 z + f_1(x_1) + f_2(x_2)$$

The GAM estimate for this model is

$$y = \widehat{\alpha_0} + \widehat{\alpha_1} z + \widehat{\beta_1} x_1 + \widehat{\beta_2} x_2 + \widehat{s_1}(x_1) + \widehat{s_2}(x_2)$$

where $\widehat{s_1}$ and $\widehat{s_2}$ are linear-adjusted nonparametric estimates of the f_1 and f_2 effects. The p-values for $\widehat{\alpha_0}$, $\widehat{\alpha_1}$, $\widehat{\beta_1}$, and $\widehat{\beta_2}$ are reported in the parameter estimates table. $\widehat{\beta_1}$ and $\widehat{\beta_2}$ are the estimates labeled Linear(x1) and Linear(x2) in the table. The p-values for $\widehat{s_1}$ and $\widehat{s_2}$ are reported in the analysis of deviance table.

Only $\widehat{s_1}$, $\widehat{s_2}$, and $\hat{y}$ are output to the output data set, with the corresponding variable names P_x1, P_x2, and P_y. For Gaussian data, the complete marginal prediction for variable x1 is:

$$\widehat{\beta_1}x_1 + P_{x_1}$$

If the additive component plots are requested by the ADDITIVE option, the additive component for variable x2 is obtained by:

$$\widehat{\beta_2}(x_2 - \bar{x_2}) + P_{x_2}$$

ODS Table Names

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 about ODS, see Chapter 20, "Using the Output Delivery System."

Table 36.3 ODS Tables Produced by PROC GAM

ODS Table Name	Description	Statement	Option
ANODEV	Analysis of deviance table for smoothing variables	PROC	default
ClassSummary	Summary of classification variables	PROC	default
ConvergenceStatus	Convergence status of the local scoring algorithm	PROC	default
InputSummary	Input data summary	PROC	default
IterHistory	Iteration history table	MODEL	ITPRINT
IterSummary	Iteration summary	PROC	default
FitSummary	Fit parameters and fit summary	PROC	default
ParameterEstimates	Parameter estimation for regression variables	PROC	default
ResponseProfile	Frequency counts for binary models	MODEL	DIST=BINOMIAL

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.

ODS Graphics

This section describes the use of ODS for creating statistical graphs with the GAM procedure. To request these graphs you must specify the ODS GRAPHICS statement in addition to the options indicated in Table 36.4. For more information about the ODS GRAPHICS statement, see Chapter 21, "Statistical Graphics Using ODS."

When ODS Graphics is in effect, the GAM procedure by default produces plots of the partial predictions for each nonparametric predictor in the model. Use the PLOTS option in the PROC GAM statement to control aspects of these plots.

ODS Graph Names

PROC GAM assigns a name to each graph it creates by using ODS. You can use these names to reference the graphs when using ODS. The names are listed in Table 36.4.

Table 36.4 ODS Graphics Produced by PROC GAM

ODS Graph Name	Plot Description	PLOTS= Option
SmoothingComponentPlot	panel of multiple partial prediction curves	COMPONENTS
SmoothingComponentPlot	unpacked partial prediction curves	COMPONENTS(UNPACK)

By default, partial prediction plots for each component are displayed in panels containing at most six plots. If you specify more than six smoothing components, multiple panels are used. Use the PLOTS(UNPACK) option in the PROC GAM statement to display these plots individually.

Examples: GAM Procedure

Example 36.1: Generalized Additive Model with Binary Data

This 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. (1994). 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 (Age), 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 data=kyphosis descending;
    model Kyphosis = Age StartVert NumVert/link=logit dist=binomial;
run;
```

The GENMOD analysis of the independent variable effects is shown in Output 36.1.1. Based on these results, the only significant factor is StartVert with a log odds ratio of -0.1972. The variable NumVert has a p-value of 0.0904 with a log odds ratio of 0.3031.

Output 36.1.1 GENMOD Analysis: Partial Output

```
                      Comparing PROC GAM with PROC GENMOD

                              The GENMOD Procedure

                 Analysis Of Maximum Likelihood Parameter Estimates

                                Standard       Wald 95%            Wald
 Parameter  DF  Estimate      Error   Confidence Limits  Chi-Square  Pr > ChiSq

 Intercept   1   -1.2497     1.2424    -3.6848    1.1853        1.01      0.3145
 Age         1    0.0061     0.0055    -0.0048    0.0170        1.21      0.2713
 StartVert   1   -0.1972     0.0657    -0.3260   -0.0684        9.01      0.0027
 NumVert     1    0.3031     0.1790    -0.0477    0.6540        2.87      0.0904
 Scale       0    1.0000     0.0000     1.0000    1.0000

NOTE: The scale parameter was held fixed.
```

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 (event='1') = spline(Age      ,df=3)
                                spline(StartVert,df=3)
                                spline(NumVert  ,df=3) / dist=binomial;
run;
```

The MODEL statement requests an additive model with a univariate smoothing spline for each term. The response variable option EVENT= chooses Kyphosis= 1 (presence) as the event so that the probability of presence of kyphosis is modeled. The option "DIST=BINOMIAL" with binary responses specifies a logistic model. Each term is fit by using a univariate smoothing spline with three degrees of freedom. Of these three degrees of freedom, one is taken up by the linear portion of the fit and two are left for the nonlinear spline portion. Although this might seem to be an unduly modest amount of flexibility, it is better to be conservative with a data set this small.

Output 36.1.2 and Output 36.1.3 list the output from PROC GAM.

Output 36.1.2 Summary Statistics

```
                        Comparing PROC GAM with PROC GENMOD

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

                              Summary of Input Data Set

                    Number of Observations                   83
                    Number of Missing Observations            0
                    Distribution                       Binomial
                    Link Function                         Logit

                                  Response Profile

                       Ordered                        Total
                         Value        Kyphosis    Frequency

                             1               0           65
                             2               1           18

NOTE: PROC GAM is modeling the probability that Kyphosis=1. One way to change
      this to model the probability that Kyphosis=0 is to specify the response
      variable option EVENT='0'.


                      Iteration Summary and Fit Statistics

           Number of local scoring iterations                          9
           Local scoring convergence criterion               2.6635648E-9
           Final Number of Backfitting Iterations                      1
           Final Backfitting Criterion                       5.2326574E-9
           The Deviance of the Final Estimate                46.610922438
```

Output 36.1.3 Model Fit Statistics

```
                             Regression Model Analysis
                                Parameter Estimates

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

      Intercept                   -2.01533        1.45620      -1.38      0.1706
      Linear(Age)                  0.01213        0.00794       1.53      0.1308
      Linear(StartVert)           -0.18615        0.07628      -2.44      0.0171
      Linear(NumVert)              0.38347        0.19102       2.01      0.0484
```

Output 36.1.3 *continued*

```
                           Smoothing Model Analysis
                    Fit Summary for Smoothing Components

                                                                    Num
                        Smoothing                                 Unique
Component               Parameter          DF             GCV       Obs

Spline(Age)              0.999996    2.000000      328.512831        66
Spline(StartVert)        0.999551    2.000000      317.646685        16
Spline(NumVert)          0.921758    2.000000       20.144056        10

                            Smoothing Model Analysis
                              Analysis of Deviance

                                           Sum of
 Source                       DF          Squares    Chi-Square    Pr > ChiSq

 Spline(Age)             2.00000        10.494369       10.4944        0.0053
 Spline(StartVert)       2.00000         5.494968        5.4950        0.0641
 Spline(NumVert)         2.00000         2.184518        2.1845        0.3355
```

The critical part of the GAM results is the "Analysis of Deviance" table, shown in Output 36.1.3. For each smoothing effect in the model, this table gives a χ^2 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 is highly significant, the effect of StartVert is nearly significant, and the effect of NumVert is insignificant at the 5% level.

PROC GAM can also perform approximate analysis of deviance for smoothing effects by using the ANODEV=NOREFIT option, as in the following statements:

```
title 'PROC GAM with Approximate Analysis of Deviance';
proc gam data=kyphosis;
   model Kyphosis (event='1') = spline(Age      ,df=3)
                                spline(StartVert,df=3)
                                spline(NumVert  ,df=3) /
                                    dist=binomial anodev=norefit;
run;
```

Output 36.1.4 Approximate Analysis of Deviance Table

```
                  PROC GAM with Approximate Analysis of Deviance

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

                            Smoothing Model Analysis
                         Approximate Analysis of Deviance

           Source                        DF    Chi-Square    Pr > ChiSq

           Spline(Age)              2.00000        7.0888        0.0289
           Spline(StartVert)        2.00000        5.0431        0.0803
           Spline(NumVert)          2.00000        2.2471        0.3251
```

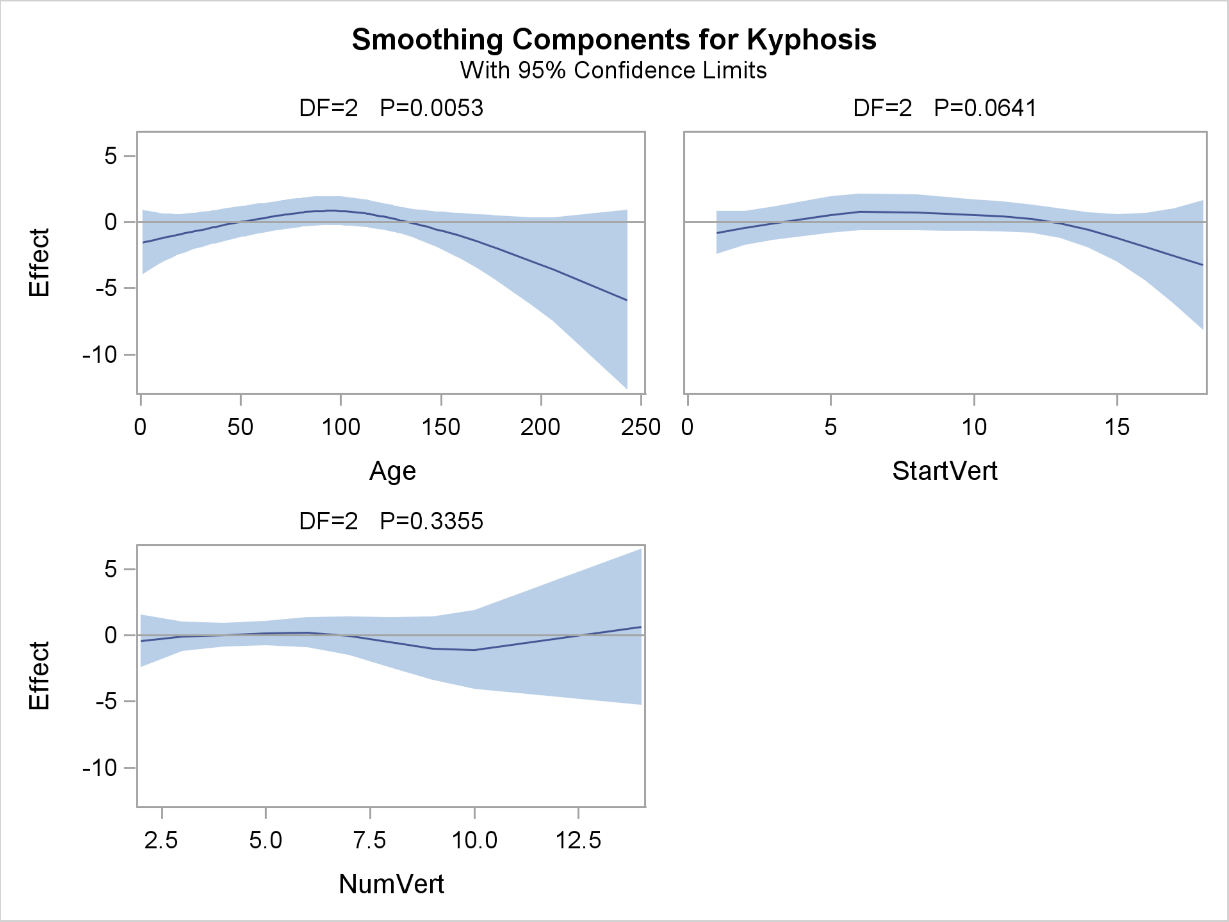
The “Approximate Analysis of Deviance” table shown in Output 36.1.4 yields similar conclusions to those of the “Analysis of Deviance” table (Output 36.1.3). In addition to the GAM fit that uses all the specified smoothing effects, the default ANODEV=REFIT option requires *p* additional GAM fits to be performed for *p* smoothing effects. In each of these fits, a submodel is fit by omitting one smoothing term from the model. By contrast, the ANODEV=NOREFIT option keeps the nonparametric terms fixed and requires a weighted least squares fit for only the parametric part of the model. Hence, GAM with the ANODEV=NOREFIT option is computationally inexpensive and is useful for obtaining approximate analysis of deviance results for models with many smoothing effects.

Plots of predictions against predictor can be used to investigate why PROC GAM and PROC GENMOD produce different results. The following statements use ODS Graphics to produce plots of the individual smoothing components. The CLM suboption for the PLOTS option adds a curvewise Bayesian confidence band to each smoothing component, while the COMMONAXES suboption forces all three smoothing component plots to share the same vertical axis limits, allowing a visual judgment of nonparametric effect size.

```
ods graphics on;

proc gam data=kyphosis plots=components(clm commonaxes);
    model Kyphosis (event='1') = spline(Age      ,df=3)
                                 spline(StartVert,df=3)
                                 spline(NumVert  ,df=3) / dist=binomial;
run;
```

For general information about ODS Graphics, see Chapter 21, “Statistical Graphics Using ODS.” For specific information about the graphics available in the GAM procedure, see the section “ODS Graphics” on page 1859. The smoothing component plots are displayed in Output 36.1.5.

Output 36.1.5 Partial Prediction for Each Predictor



The plots show that the partial predictions corresponding to both Age and StartVert have a 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 rerun the analysis without this point, to explore how much it affects the conclusions. You can do this by simply including a WHERE clause when specifying the data set for the GAM procedure, as in the following statements:

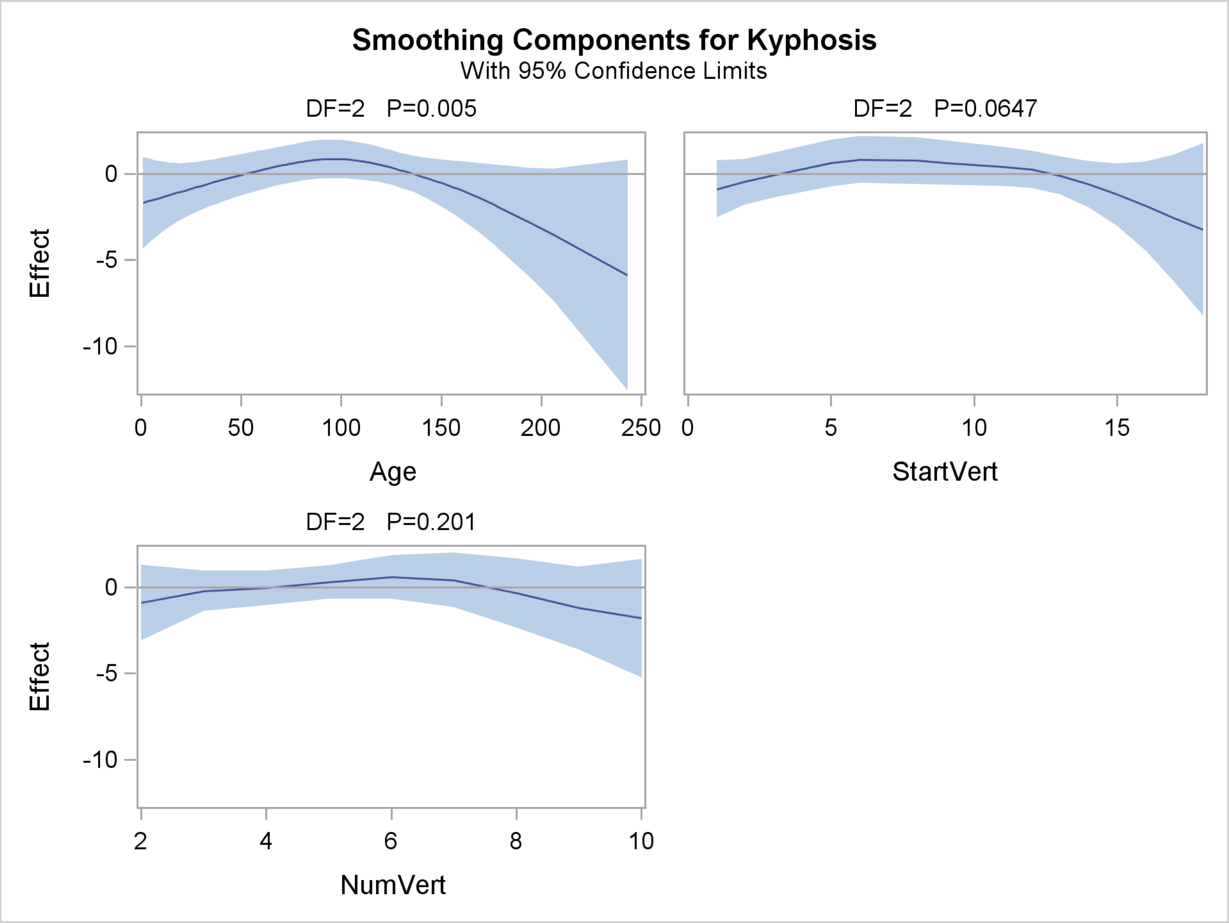
```
proc gam data=kyphosis(where=(NumVert^=14)) plots=components(clm commonaxes);
    model Kyphosis (event='1') = spline(Age      ,df=3)
                                 spline(StartVert,df=3)
                                 spline(NumVert  ,df=3) / dist=binomial;
run;

ods graphics off;
```

The analysis of deviance table from this reanalysis is shown in Output 36.1.6, and Output 36.1.7 shows the recomputed partial predictor plots.

Output 36.1.6 Analysis after Removing NumVert=14

```
                PROC GAM with Approximate Analysis of Deviance

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

                          Smoothing Model Analysis
                            Analysis of Deviance

                                             Sum of
 Source                       DF            Squares    Chi-Square    Pr > ChiSq

 Spline(Age)             2.00000          10.587556       10.5876        0.0050
 Spline(StartVert)       2.00000           5.477094        5.4771        0.0647
 Spline(NumVert)         2.00000           3.209089        3.2091        0.2010
```

Output 36.1.7 Analysis after Removing NumVert=14



Removing data point NumVert=14 has little effect on either the analysis of deviance results or the estimated curves for StartVert and NumVert. But the removal has a noticeable effect on the variable NumVert, whose curve now also seems quadratic, though it is much less pronounced than for the other two variables.

An important difference between the first analysis of these data with GENMOD and the subsequent analysis with GAM is that GAM indicates that age is a significant predictor of kyphosis. The difference is due to the fact that the GENMOD model includes only a linear effect in Age whereas the GAM model allows a more complex relationship, which the plots indicate is nearly quadratic. 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 statements fit 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 36.1.8.

```
title 'Comparing PROC GAM with PROC GENMOD';
proc genmod data=kyphosis(where=(NumVert^=14)) descending;
            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 36.1.8 Joint Linear and Quadratic Tests

```
                    Comparing PROC GAM with PROC GENMOD

                           The GENMOD Procedure

                             Contrast Results

                                  Chi-
            Contrast        DF    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 36.2: Poisson Regression Analysis of Component Reliability

In this example, the number of maintenance repairs on a complex system are modeled as realizations of Poisson random variables. The system under investigation has a large number of components, which occasionally break down and are replaced or repaired. During a four-year period, the system was observed to be in a state of steady operation, meaning that the rate of operation remained approximately constant. A monthly maintenance record is available for that period, which tracks the number of components removed for maintenance each month. The data are listed in the following statements, which create a SAS data set:

```
title 'Analysis of Component Reliability';
data equip;
    input year month removals @@;
datalines;
1987   1  2 1987   2  4 1987   3  3
1987   4  3 1987   5  3 1987   6  8
1987   7  2 1987   8  6 1987   9  3
1987  10  9 1987  11  4 1987  12 10
1988   1  4 1988   2  6 1988   3  4
1988   4  4 1988   5  3 1988   6  5
1988   7  3 1988   8  4 1988   9  5
1988  10  3 1988  11  6 1988  12  3
1989   1  2 1989   2  6 1989   3  1
1989   4  5 1989   5  5 1989   6  4
1989   7  2 1989   8  2 1989   9  2
1989  10  5 1989  11  1 1989  12 10
1990   1  3 1990   2  8 1990   3 12
1990   4  7 1990   5  3 1990   6  2
1990   7  4 1990   8  3 1990   9  0
1990  10  6 1990  11  6 1990  12  6
;
```

For planning purposes, it is of interest to understand the long- and short-term trends in the maintenance needs of the system. Over the long term, it is suspected that the quality of new components and repair work improves over time, so the number of component removals would tend to decrease from year to year. It is not known whether the robustness of the system is affected by seasonal variations in the operating environment, but this possibility is also of interest.

Because the maintenance record is in the form of counts, the number of removals are modeled as realizations of Poisson random variables. Denote by λ_{ij} the unobserved component removal rate for year i and month j. Since the data were recorded at regular intervals (from a system operating at a constant rate), each λ_{ij} is assumed to be a function of year and month only.

A preliminary two-way analysis is performed by using PROC GENMOD to make broad inferences on repair trends. A log-link is specified for the model

$$\log \lambda_{ij} \quad = \quad \mu + \alpha_i^Y + \alpha_j^M$$

where μ is a grand mean, α_i^Y is the effect of the ith year, and α_j^M is the effect of the jth month. In the following statements, the CLASS statement declares the variables year and month as categorical. Type III sum of squares are requested to test whether there is an overall effect of year and/or month.

```
title2 'Two-way model';
proc genmod data=equip;
    class year month;
    model removals=year month / dist=Poisson link=log type3;
run;
```

Output 36.2.1 displays the listed Type III statistics for the fitted model. With the test for year effects yielding a p-value of 0.4527, there is no evidence of a long-term trend in maintenance rates. Apparently, the quality of new or repaired components did not change between 1987 and 1990. However, the test for monthly trends does yield a small p-value of 0.0321, indicating that seasonal trends are just barely significant at the $\alpha = 0.05$ level.

Output 36.2.1 PROC GENMOD Listing for Type III Analysis

```
                    Analysis of Component Reliability
                              Two-way model

                          The GENMOD Procedure

                    LR Statistics For Type 3 Analysis

                                               Chi-
              Source              DF        Square     Pr > ChiSq

              year                 3          2.63         0.4527
              month               11         21.12         0.0321
```

The Type III tests indicate that the year term can be dropped from the model. The focus of the analysis is now on identifying the form of the underlying seasonal trend, which is a task that PROC GAM is especially suited for. PROC GAM will be used to fit both a reduced categorical model, with year eliminated, and a nonparametric spline model. Although PROC GENMOD also has the capability to fit categorical models, as demonstrated earlier, PROC GAM will be used here to fit both models for a better comparison.

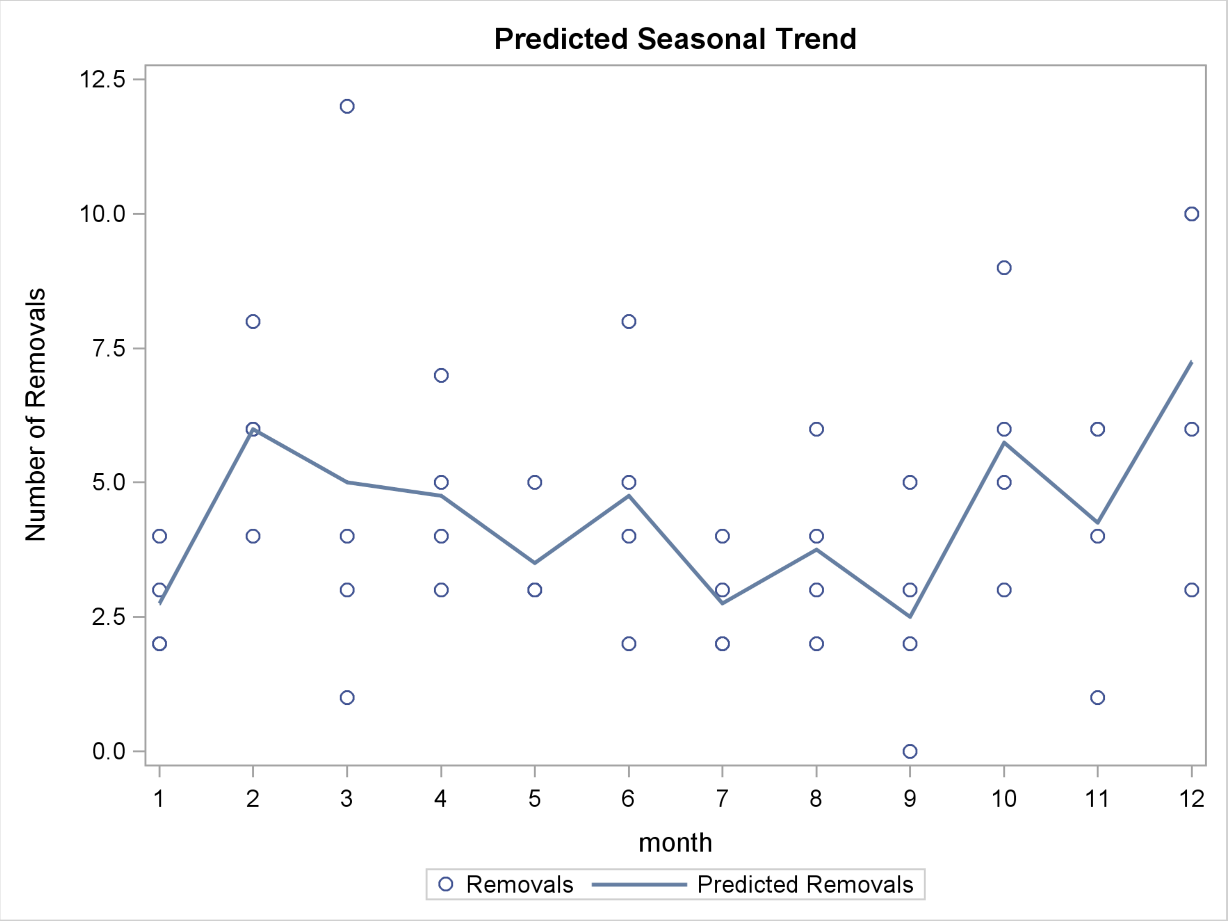
The following PROC GAM statements specify the reduced categorical model. For this part of the analysis, a CLASS statement is again used to specify that month is a categorical variable. In the follow-up, the seasonal effect will be treated as a nonparametric function of month.

```
title2 'One-way model';
proc gam data=equip;
     class month;
     model removals=param(month) / dist=Poisson;
     output out=est p;
run;
```

The following statements use the SGPLOT procedure to generate a plot of the estimated seasonal trend. The plot is displayed in Output 36.2.2.

```
proc sort data=est;by month;run;
proc sgplot data=est;
    title "Predicted Seasonal Trend";
    yaxis label="Number of Removals";
    xaxis integer values=(1 to 12);
    scatter x=Month y=Removals / name="points"
                                 legendLabel="Removals";
    series  x=Month y=p_Removals / name="line"
                                   legendLabel="Predicted Removals"
                                   lineattrs = GRAPHFIT;
    discretelegend "points" "line";
run;
```

Output 36.2.2 Predicted Seasonal Trend from a Parametric Model Fit Using a CLASS Statement



The predicted repair rates shown in Output 36.2.2 form a jagged seasonal pattern. Ignoring the month-to-month fluctuations, which are difficult to explain and can be artifacts of random noise, the general removal rate trend starts by increasing at the beginning of the year; then the trend flattens out in February and then decreases through August; and then it flattens out again in September and begins an increasing trend that continues throughout the rest of the year.

One advantage of nonparametric regression is its ability to highlight general trends in the data, such as those described earlier, and to attribute local fluctuations to unexplained random noise. The nonparametric regression model used by PROC GAM specifies that the underlying removal rates λ_j are of the form

$$\log \lambda_j = \beta_0 + \beta_1 Month_j + s(Month_j)$$

where β_1 is a linear coefficient and $s(\,)$ is a nonparametric regression function. β_1 and $s(\,)$ define the linear and nonparametric parts, respectively, of the seasonal trend.

The following statements request that PROC GAM fit a cubic spline model to the monthly repair data. The output listing is displayed in Output 36.2.3 and Output 36.2.4.

```
title 'Analysis of Component Reliability';
title2 'Spline model';
proc gam data=equip;
    model removals=spline(month) / dist=Poisson method=gcv;
run;
```

The METHOD=GCV option is used to determine an appropriate level of smoothing. The keywords LCLM and UCLM in the OUTPUT statement request that lower and upper 95% confidence bounds on each $s(Month_j)$ be included in the output data set.

Output 36.2.3 PROC GAM Listing for Cubic Spline Regression Using the METHOD=GCV Option

```
                    Analysis of Component Reliability
                              Spline model

                           The GAM Procedure
                      Dependent Variable: removals
               Smoothing Model Component(s): spline(month)

                        Summary of Input Data Set

                Number of Observations                  48
                Number of Missing Observations           0
                Distribution                       Poisson
                Link Function                          Log

                  Iteration Summary and Fit Statistics

       Number of local scoring iterations                        5
       Local scoring convergence criterion           7.241527E-12
       Final Number of Backfitting Iterations                    1
       Final Backfitting Criterion                   1.710339E-11
       The Deviance of the Final Estimate            56.901543546
```

Output 36.2.4 Model Fit Statistics

```
                          Regression Model Analysis
                            Parameter Estimates

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

      Intercept            1.34594         0.14509       9.28      <.0001
      Linear(month)        0.02274         0.01893       1.20      0.2362

                          Smoothing Model Analysis
                    Fit Summary for Smoothing Components

                                                                       Num
                     Smoothing                                      Unique
   Component         Parameter              DF             GCV         Obs

   Spline(month)      0.901512        1.879980        0.115848          12

                           Smoothing Model Analysis
                             Analysis of Deviance

                                           Sum of
    Source                    DF          Squares    Chi-Square    Pr > ChiSq

    Spline(month)        1.87998         8.877764        8.8778        0.0103
```

Notice in the listing in Output 36.2.4 that the DF value chosen for the nonlinear portion of the spline by minimizing GCV is about 1.88, which is smaller than the default value of 3. This indicates that the spline model of the seasonal trend is relatively simple. As indicated by the “Analysis of Deviance” table, it is a significant feature of the data: the table lists a *p*-value of 0.0103 for the hypothesis of no seasonal trend. Note also that the “Parameter Estimates” table lists a *p*-value of 0.2362 for the hypothesis of no linear factor in the seasonal trend.

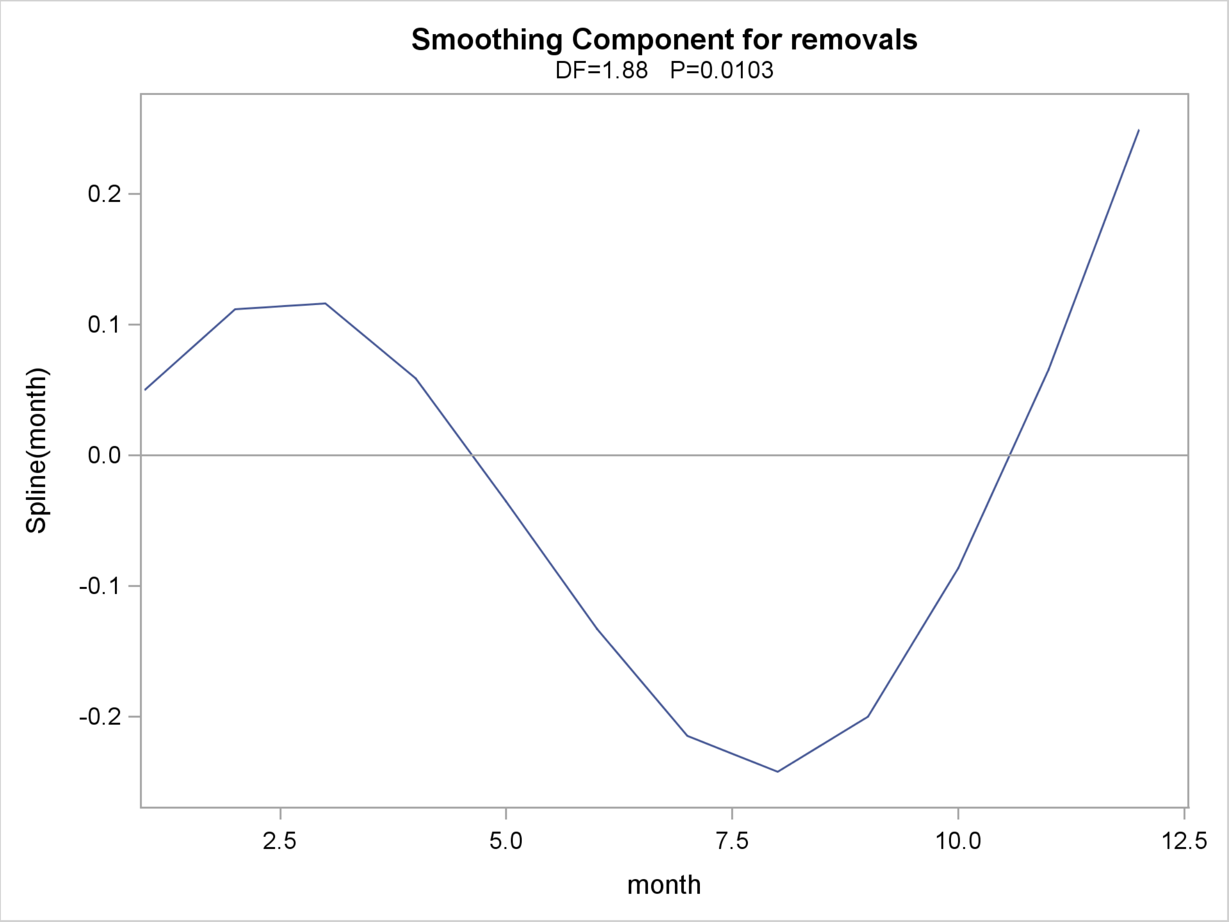
The following statements use ODS Graphics statement to plot the smoothing component for the effect of Month on predicted repair rates:

```
ods graphics on;

proc gam data=equip;
    model removals=spline(month) / dist=Poisson method=gcv;
run;
```

For general information about ODS graphics, see Chapter 21, “Statistical Graphics Using ODS.” For specific information about the graphics available in the GAM procedure, see the section “ODS Graphics” on page 1859. The smoothing component plot is displayed in Output 36.2.5.

In Output 36.2.5, it is apparent that the pattern of repair rates follows the general pattern observed in Output 36.2.2. However, the plot in Output 36.2.5 is much cleaner because the month-to-month fluctuations are smoothed out to reveal the broader seasonal trend.

Output 36.2.5 Predicted Seasonal Trend from a Cubic Spline Model

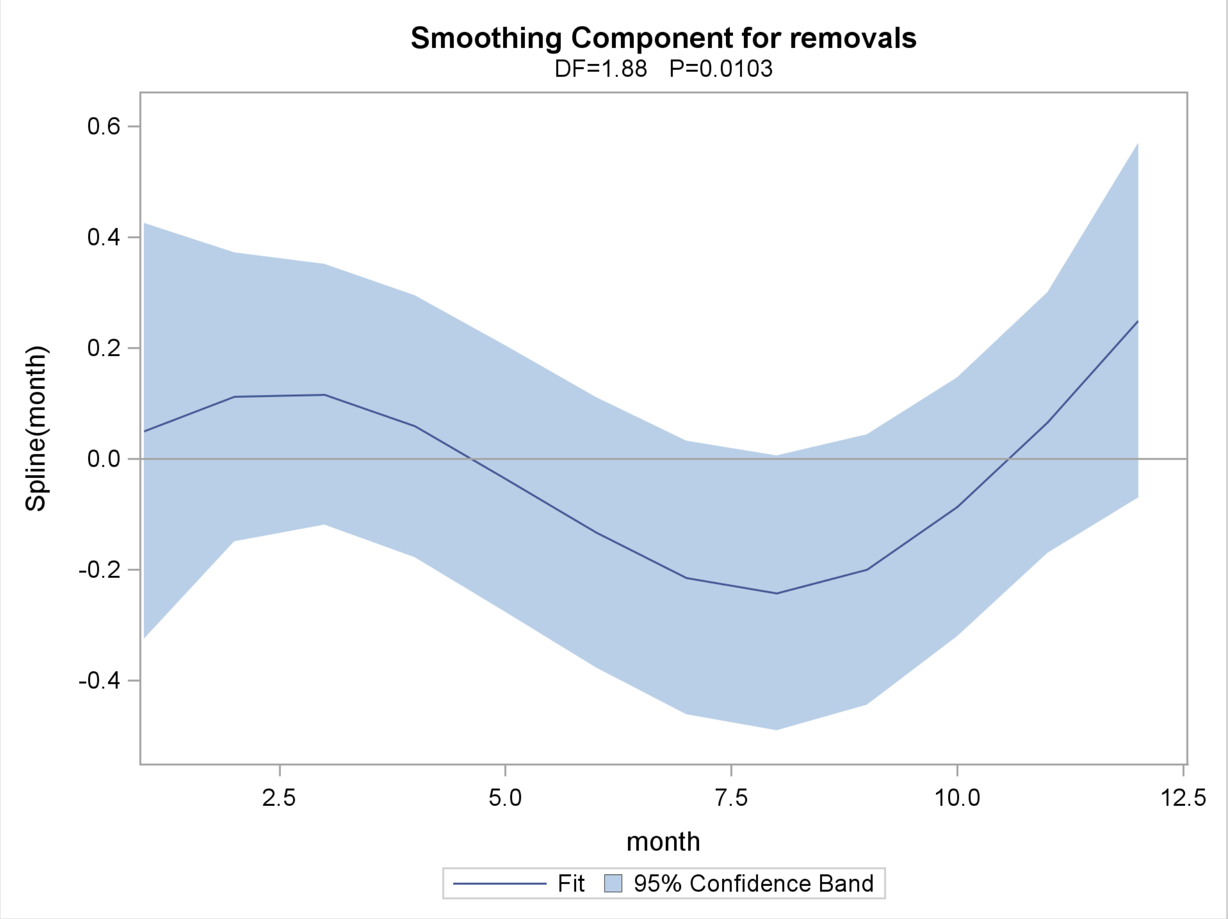


You can use the PLOTS(CLM) option to added a 95% confidence band to the plot of $s(Month_j)$, as in the following statements:

```
proc gam data=equip plots=components(clm);
    model removals=spline(month) / dist=Poisson method=gcv;
run;

ods graphics off;
```

The plot is displayed in Output 36.2.6.

Output 36.2.6 Estimated Nonparametric Factor of Seasonal Trend, Along with 95% Confidence Bounds



The small *p*-value in Output 36.2.1 of 0.0321 for the hypothesis of no seasonal trend indicates that the data exhibit significant seasonal structure. However, Output 36.2.6 is a graphical illustration of a degree of indistinctness in that structure. For instance, the horizontal reference line at zero is entirely within the 95% confidence band—that is, the estimated nonlinear part of the trend is relatively flat. Thus, despite evidence of seasonality based on the parametric model, it is difficult to narrow down its significant effects to a specific part of the year.

Example 36.3: Comparing PROC GAM with PROC LOESS

In an analysis of simulated data from a hypothetical chemistry experiment, additive nonparametric regression performed by PROC GAM is compared to the unrestricted multidimensional procedure of PROC LOESS.

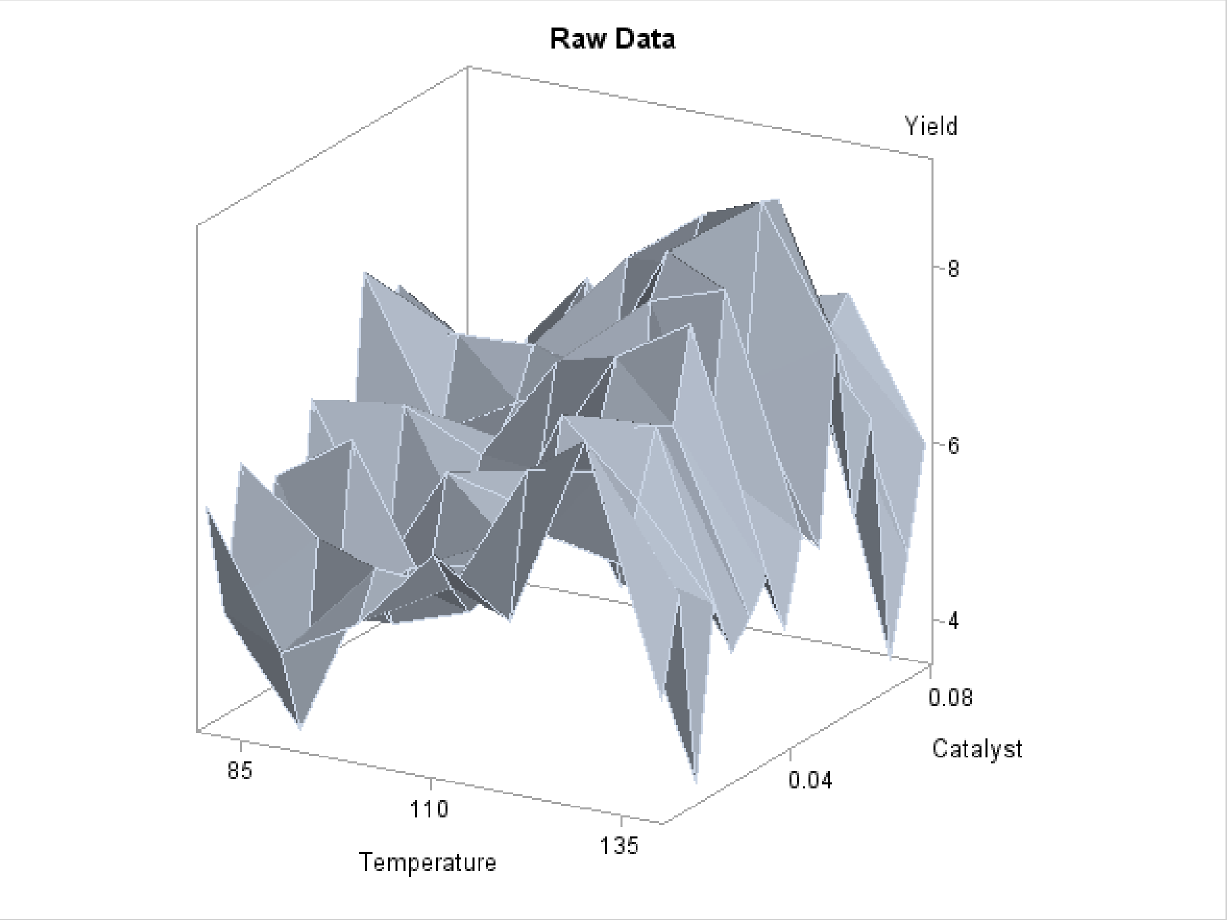
In each repetition of the experiment, a catalyst is added to a chemical solution, thereby inducing synthesis of a new material. The data are measurements of the temperature of the solution, the amount of catalyst added, and the yield of the chemical reaction. The following statements read and plots the raw data.

```
data ExperimentA;
    format Temperature f4.0 Catalyst f6.3 Yield f8.3;
    input Temperature Catalyst Yield @@;
datalines;
80  0.005 6.039  80 0.010 4.719  80 0.015 6.301
80  0.020 4.558  80 0.025 5.917  80 0.030 4.365
80  0.035 6.540  80 0.040 5.063  80 0.045 4.668
80  0.050 7.641  80 0.055 6.736  80 0.060 7.255
80  0.065 5.515  80 0.070 5.260  80 0.075 4.813
80  0.080 4.465  90 0.005 4.540  90 0.010 3.553
90  0.015 5.611  90 0.020 4.586  90 0.025 6.503
90  0.030 4.671  90 0.035 4.919  90 0.040 6.536
90  0.045 4.799  90 0.050 6.002  90 0.055 6.988
90  0.060 6.206  90 0.065 5.193  90 0.070 5.783
90  0.075 6.482  90 0.080 5.222 100 0.005 5.042
100 0.010 5.551 100 0.015 4.804 100 0.020 5.313
100 0.025 4.957 100 0.030 6.177 100 0.035 5.433
100 0.040 6.139 100 0.045 6.217 100 0.050 6.498
100 0.055 7.037 100 0.060 5.589 100 0.065 5.593
100 0.070 7.438 100 0.075 4.794 100 0.080 3.692
110 0.005 6.005 110 0.010 5.493 110 0.015 5.107
110 0.020 5.511 110 0.025 5.692 110 0.030 5.969
110 0.035 6.244 110 0.040 7.364 110 0.045 6.412
110 0.050 6.928 110 0.055 6.814 110 0.060 8.071
110 0.065 6.038 110 0.070 6.295 110 0.075 4.308
110 0.080 7.020 120 0.005 5.409 120 0.010 7.009
120 0.015 6.160 120 0.020 7.408 120 0.025 7.123
120 0.030 7.009 120 0.035 7.708 120 0.040 5.278
120 0.045 8.111 120 0.050 8.547 120 0.055 8.279
120 0.060 8.736 120 0.065 6.988 120 0.070 6.283
120 0.075 7.367 120 0.080 6.579 130 0.005 7.629
130 0.010 7.171 130 0.015 5.997 130 0.020 6.587
130 0.025 7.335 130 0.030 7.209 130 0.035 8.259
130 0.040 6.530 130 0.045 8.400 130 0.050 7.218
130 0.055 9.167 130 0.060 9.082 130 0.065 7.680
130 0.070 7.139 130 0.075 7.275 130 0.080 7.544
140 0.005 4.860 140 0.010 5.932 140 0.015 3.685
140 0.020 5.581 140 0.025 4.935 140 0.030 5.197
140 0.035 5.559 140 0.040 4.836 140 0.045 5.795
140 0.050 5.524 140 0.055 7.736 140 0.060 5.628
140 0.065 6.644 140 0.070 3.785 140 0.075 4.853
140 0.080 6.006
;
```

```
proc sort data=ExperimentA;
    by Temperature Catalyst;
run;

proc template;
    define statgraph surface;
    dynamic _X _Y _Z _T;
    begingraph;
        entrytitle _T;
        layout overlay3d/
             xaxisopts=(linearopts=(tickvaluesequence=
                        (start=85 end=135 increment=25)))
             yaxisopts=(linearopts=(tickvaluesequence=
                        (start=0 end=0.08 increment=0.04)))
             rotate=30 cube=false;
        surfaceplotparm x=_X y=_Y z=_Z;
        endlayout;
    endgraph;
    end;
run;

ods graphics on;

proc sgrender data=ExperimentA template=surface;
    dynamic _X='Temperature' _Y='Catalyst' _Z='Yield' _T='Raw Data';
run;
```

The plot is displayed in Output 36.3.1. A surface fitted to the plot of Output 36.3.1 by PROC LOESS will be of a very general (and flexible) type, since the procedure requires only weak assumptions about the structure of the dependencies among the data. PROC GAM, on the other hand, makes stronger structural assumptions by restricting the fitted surface to an additive form. These differences will be demonstrated in this example.

Output 36.3.1 Surface Plot of Yield by Temperature and Amount of Catalyst



The following statements request that both PROC LOESS and PROC GAM fit surfaces to the data:

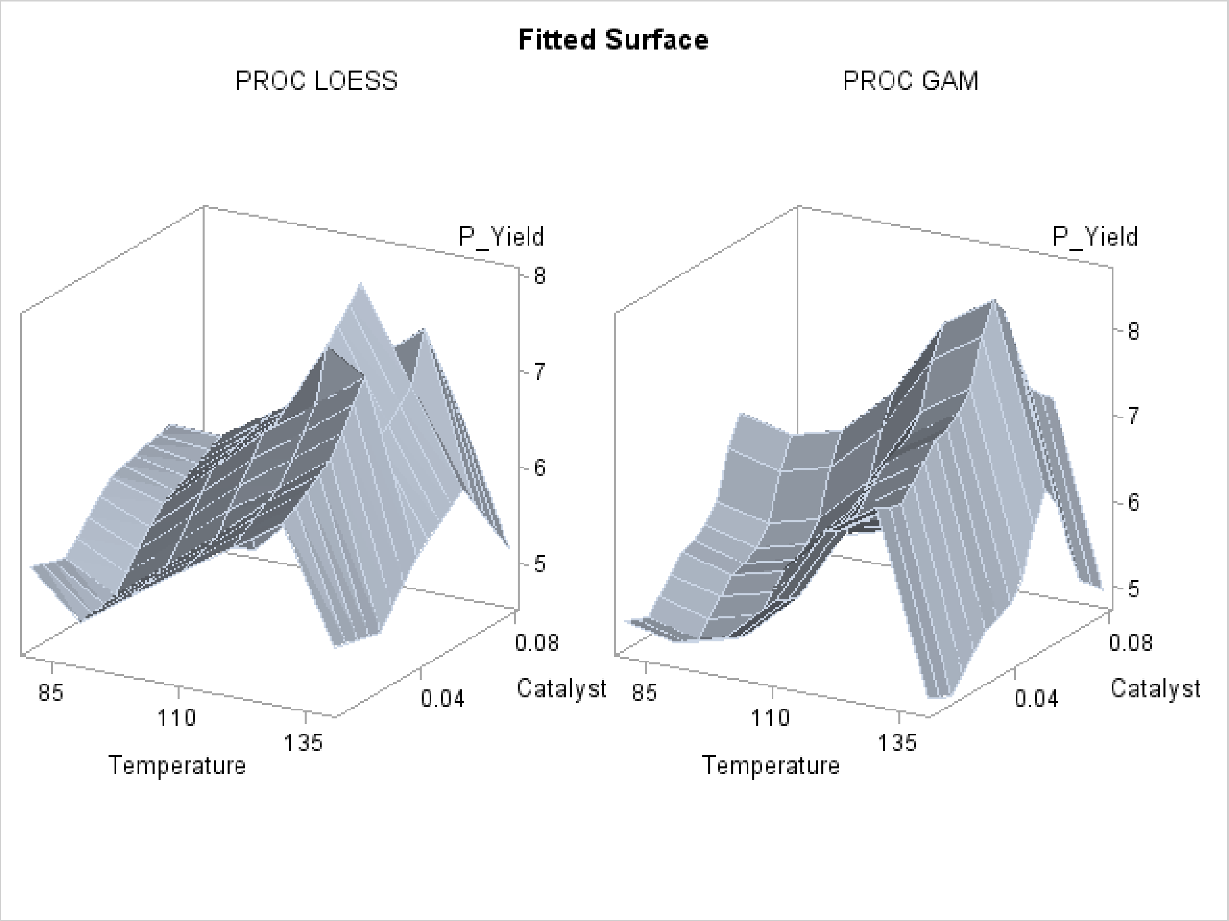
```
ods output ScoreResults=PredLOESS;
proc loess data=ExperimentA;
    model Yield = Temperature Catalyst
                  / scale=sd select=gcv degree=2;
    score;
run;

proc gam data=PredLoess;
    model Yield = loess(Temperature) loess(Catalyst) / method=gcv;
    output out=PredGAM p=Gam_p_;
run;
```

In both cases the smoothing parameter was chosen as the value that minimizes GCV. This is performed automatically by PROC LOESS and PROC GAM.

The following statements generate plots of the predicted yields, which are displayed in Output 36.3.2:

```
proc template;
    define statgraph surface1;
    begingraph;
        entrytitle "Fitted Surface";
        layout lattice/columns=2;
            layout
            overlay3d/xaxisopts=(linearopts=(tickvaluesequence=
                                 (start=85 end=135 increment=25)))
                      yaxisopts=(linearopts=(tickvaluesequence=
                                 (start=0 end=0.08 increment=0.04)))
                      rotate=30 cube=0
                      zaxisopts=(label="P_Yield");
                entry "PROC LOESS"/location=outside valign=top
                                   textattrs=graphlabeltext;
                surfaceplotparm x=Temperature y=Catalyst z=p_Yield;
            endlayout;
            layout
            overlay3d/xaxisopts=(linearopts=(tickvaluesequence=
                                 (start=85 end=135 increment=25)))
                      yaxisopts=(linearopts=(tickvaluesequence=
                                 (start=0 end=0.08 increment=0.04)))
                      rotate=30 cube=0
                      zaxisopts=(label="P_Yield");
                entry "PROC GAM"/location=outside valign=top
                                 textattrs=graphlabeltext;
                surfaceplotparm x=Temperature y=Catalyst z=Gam_p_Yield;
            endlayout;
        endlayout;
    endgraph;
    end;
run;

proc sgrender data=PredGAM template=surface1;
run;
```

Output 36.3.2 Fitted Regression Surfaces



Though both PROC LOESS and PROC GAM use the statistical technique loess, it is apparent from Output 36.3.2 that the manner in which it is applied is very different. By smoothing out the data in local neighborhoods, PROC LOESS essentially fits a surface to the data in pieces, one neighborhood at a time. The local regions are treated independently, so separate areas of the fitted surface are only weakly related. PROC GAM imposes additive structure, requiring that cross sections of the fitted surface always have the same shape and thereby relating regions that have a common value of the same individual regressor variable. Under that restriction, the loess technique need not be applied to the entire multidimensional scatter plot, but only to one-dimensional cross sections of the data.

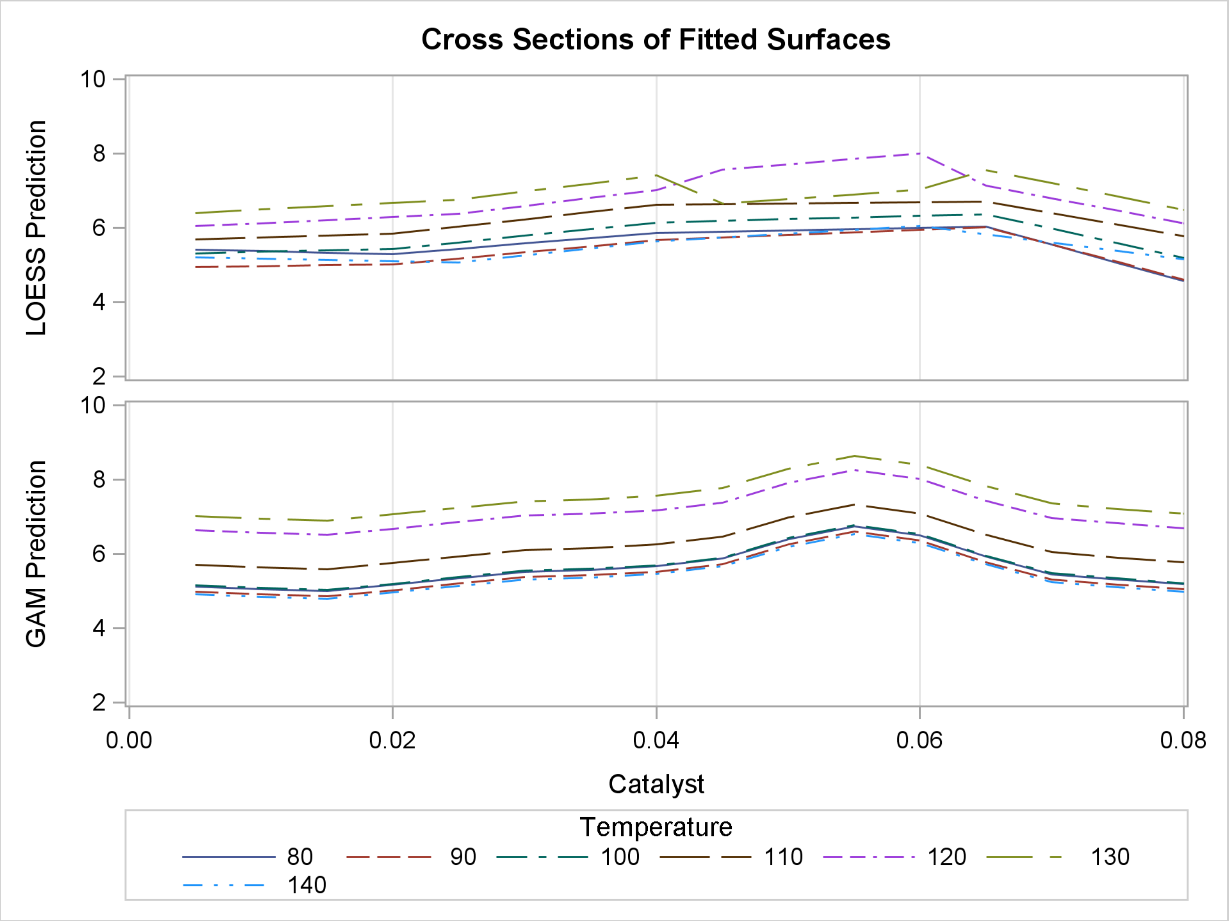
The advantage of using additive model fitting is that its statistical power is directed toward univariate smoothing, and so it is able to discern the finer details of any underlying structure in the data. Regression data can be very sparse when viewed in the context of multidimensional space, even when every individual set of regressor values densely covers its range. This is the familiar curse of dimensionality. Sparse data greatly restrict the effectiveness of nonparametric procedures, but additive model fitting, when appropriate, is one way to overcome this limitation.

To examine these properties, you can use ODS Graphics to generate plots of cross sections of unrestricted (PROC LOESS) and additive (PROC GAM) fitted surfaces for the variable Catalyst, as shown in the following statements:

```
proc template;
    define statgraph projection;
    begingraph;
        entrytitle "Cross Sections of Fitted Surfaces";
        layout lattice/rows=2 columndatarange=unionall
                       columngutter=10;
            columnAxes;
                columnAxis / display=all griddisplay=auto_on;
            endColumnAxes;

            layout overlay/
                xaxisopts=(display=none)
                yaxisopts=(label="LOESS Prediction"
                linearopts=(viewmin=2 viewmax=10));
                seriesplot x=Catalyst y=p_Yield /
                   group=temperature
                   name="Temperature";
            endlayout;

            layout overlay/
                xaxisopts=(display=none)
                yaxisopts=(label="GAM Prediction"
                linearopts=(viewmin=2 viewmax=10));
                seriesplot x=Catalyst y=Gam_p_Yield /
                   group=temperature
                   name="Temperature";
            endlayout;

            columnheaders;
                discreteLegend "Temperature" / title = "Temperature";
            endcolumnheaders;

        endlayout;
    endgraph;
    end;
run;

proc sgrender data=PredGAM template=projection;
run;
```

The plots are displayed in Output 36.3.3.

Output 36.3.3 Cross Sections of Fitted Regression Surfaces



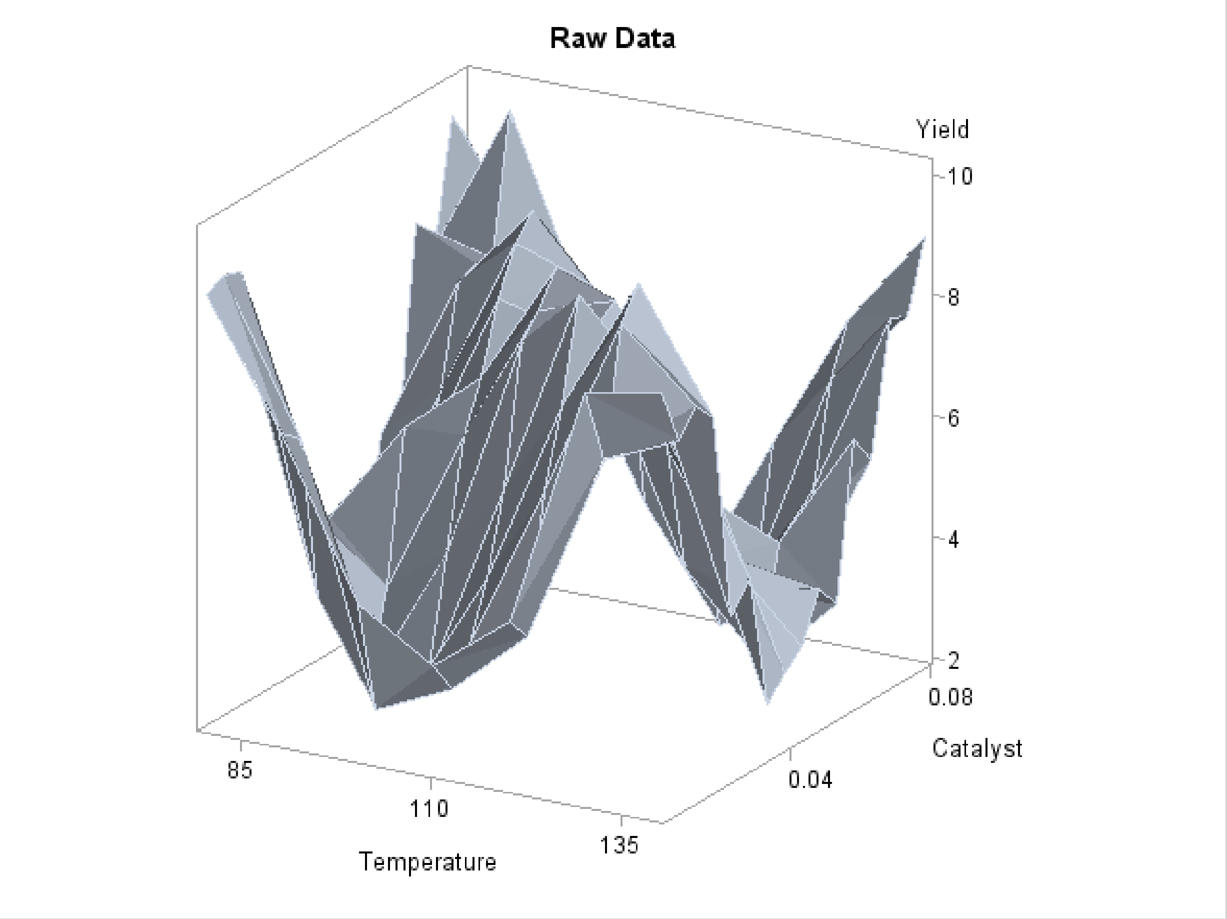
Notice that the cross sections in the top panel (PROC LOESS) of Output 36.3.3 have varying shapes, while every cross section in the bottom panel (PROC GAM) is the same curve shifted vertically. This illustrates precisely the kind of structural differences that distinguish additive models. A second important comparison to make between Output 36.3.2 and Output 36.3.3 is the level of detail in the fitted regression surfaces. Cross sections of the PROC LOESS surface are rather flat, but those of the additive surface have a clear shape. In particular, the ridge near Catalyst=0.055 is only vaguely evident in the PROC LOESS surface, but it is plainly revealed by the additive procedure.

For an example of a situation where unrestricted multidimensional fitting is preferred over additive regression, consider the following simulated data from a similar experiment. The following statements create another SAS data set and plot.

```
data ExperimentB;
    format Temperature f4.0 Catalyst f6.3 Yield f8.3;
    input Temperature Catalyst Yield @@;
datalines;
80  0.005  9.115  80 0.010  9.275  80 0.015  9.160
80  0.020  7.065  80 0.025  6.054  80 0.030  4.899
80  0.035  4.504  80 0.040  4.238  80 0.045  3.232
80  0.050  3.135  80 0.055  5.100  80 0.060  4.802
80  0.065  8.218  80 0.070  7.679  80 0.075  9.669
80  0.080  9.071  90 0.005  7.085  90 0.010  6.814
90  0.015  4.009  90 0.020  4.199  90 0.025  3.377
90  0.030  2.141  90 0.035  3.500  90 0.040  5.967
90  0.045  5.268  90 0.050  6.238  90 0.055  7.847
90  0.060  7.992  90 0.065  7.904  90 0.070 10.184
90  0.075  7.914  90 0.080  6.842 100 0.005  4.497
100 0.010  2.565 100 0.015  2.637 100 0.020  2.436
100 0.025  2.525 100 0.030  4.474 100 0.035  6.238
100 0.040  7.029 100 0.045  8.183 100 0.050  8.939
100 0.055  9.283 100 0.060  8.246 100 0.065  6.927
100 0.070  7.062 100 0.075  5.615 100 0.080  4.687
110 0.005  3.706 110 0.010  3.154 110 0.015  3.726
110 0.020  4.634 110 0.025  5.970 110 0.030  8.219
110 0.035  8.590 110 0.040  9.097 110 0.045  7.887
110 0.050  8.480 110 0.055  6.818 110 0.060  7.666
110 0.065  4.375 110 0.070  3.994 110 0.075  3.630
110 0.080  2.685 120 0.005  4.697 120 0.010  4.268
120 0.015  6.507 120 0.020  7.747 120 0.025  9.412
120 0.030  8.761 120 0.035  8.997 120 0.040  7.538
120 0.045  7.003 120 0.050  6.010 120 0.055  3.886
120 0.060  4.897 120 0.065  2.562 120 0.070  2.714
120 0.075  3.141 120 0.080  5.081 130 0.005  8.729
130 0.010  7.460 130 0.015  9.549 130 0.020 10.049
130 0.025  8.131 130 0.030  7.553 130 0.035  6.191
130 0.040  6.272 130 0.045  4.649 130 0.050  3.884
130 0.055  2.522 130 0.060  4.366 130 0.065  3.272
130 0.070  4.906 130 0.075  6.538 130 0.080  7.380
140 0.005  8.991 140 0.010  8.029 140 0.015  8.417
140 0.020  8.049 140 0.025  4.608 140 0.030  5.025
140 0.035  2.795 140 0.040  3.123 140 0.045  3.407
140 0.050  4.183 140 0.055  3.750 140 0.060  6.316
140 0.065  5.799 140 0.070  7.992 140 0.075  7.835
140 0.080  8.985
;

proc sort data=ExperimentB;
    by Temperature Catalyst;
run;

proc sgrender data=ExperimentB template=surface;
    dynamic _X=’Temperature’ _Y=’Catalyst’ _Z=’Yield’ _T=’Raw Data’;
run;
```

A plot of the raw data is displayed in Output 36.3.4.

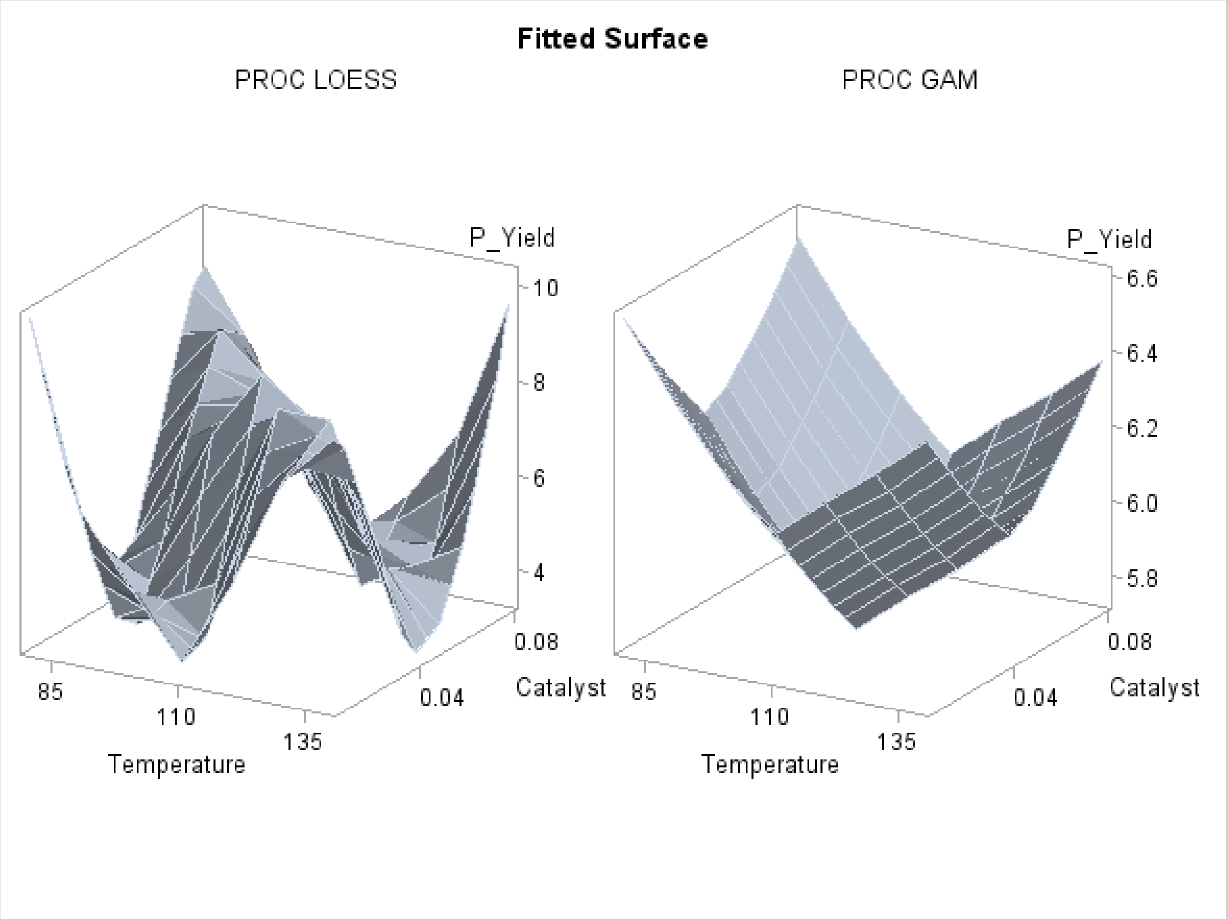
Output 36.3.4 Raw Data from Experiment B



Though the surface displayed in Output 36.3.4 is quite jagged, a distinct feature of the plot is a large ridge that runs diagonally across its surface. One would expect that the ridge would appear in the fitted regression surface of an appropriate nonparametric procedure. Nevertheless, between PROC LOESS and PROC GAM, only PROC LOESS is able to capture this significant feature.

The SAS program for fitting the new data is essentially the same as that for the data set from the first experiment. As in Output 36.3.2, multivariate and additive fitted surfaces for these data are displayed in Output 36.3.5.

Output 36.3.5 Fitted Regression Surfaces



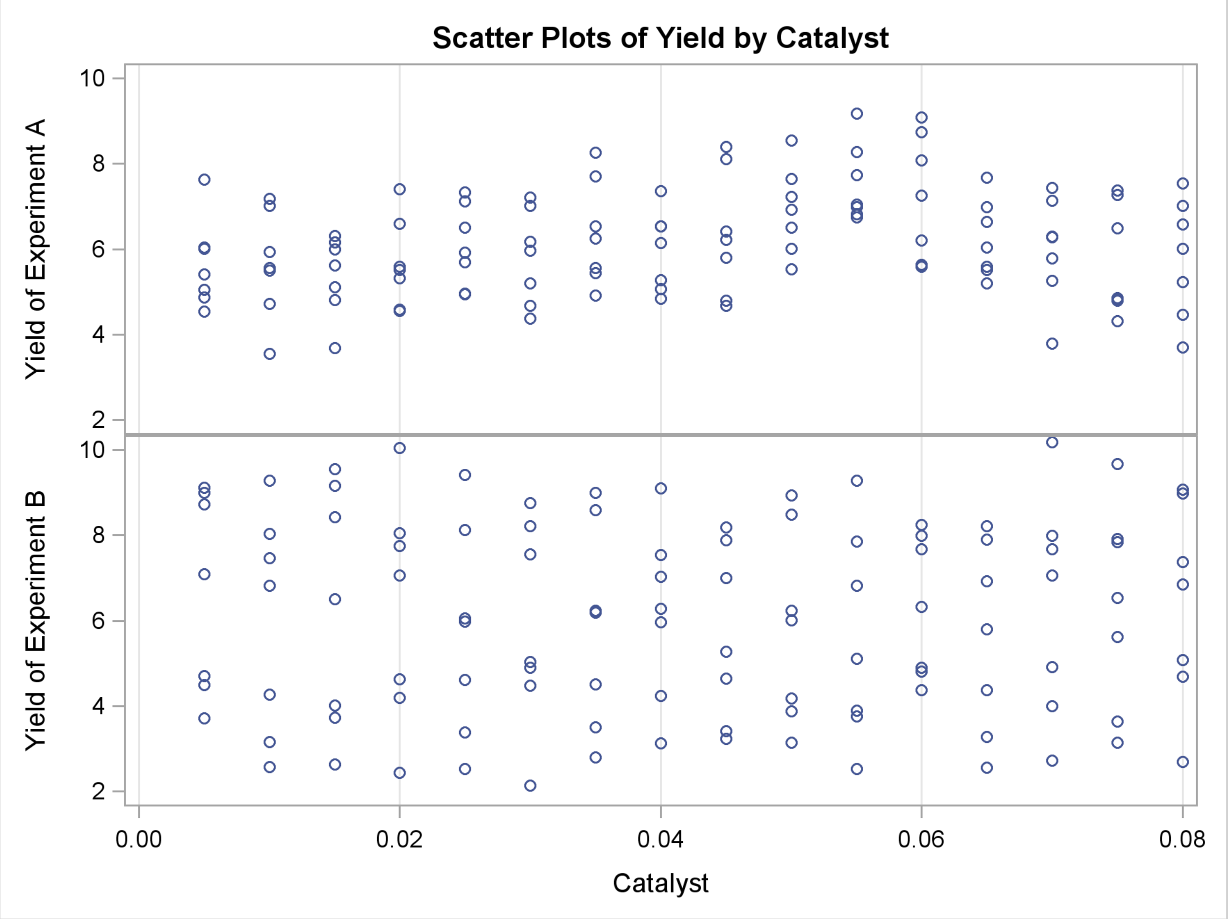
It is clear from Output 36.3.5 that the results of PROC LOESS and PROC GAM are completely different. While the plot in the left panel resembles the raw data plot in Output 36.3.4, the plot in the right panel is essentially featureless.

To understand what is happening, compare the scatter plots of Yield by Catalyst for the two data sets in this example. These are generated by the following statements and displayed in Output 36.3.6.

```
data PredGAM;
    set PredGAM;
        rename Yield=Yield_a;
run;

data PredGAMb;
    set PredGAMb;
    set PredGAM(keep=Yield_a);
run;

proc template;
    define statgraph scatter2;
    dynamic _X _Y1 _Y2;
    begingraph;
        entrytitle "Scatter Plots of Yield by Catalyst";
        layout lattice/rows=2 columndatarange=unionall
                      rowdatarange=unionall
                      columngutter=15;
            columnAxes;
                columnAxis / display=all griddisplay=auto_on;
            endColumnAxes;

            layout overlay/
                xaxisopts=(display=none)
                yaxisopts=(label="Yield of Experiment A"
                linearopts=(viewmin=2 viewmax=10));
                scatterplot x=_X y=_Y1;
            endlayout;

            layout overlay/
                xaxisopts=(display=none)
                yaxisopts=(label="Yield of Experiment B"
                linearopts=(viewmin=2 viewmax=10));
                scatterplot x=_X y=_Y2;
            endlayout;

        endlayout;
    endgraph;
    end;
run;

proc sgrender data=PredGAMb template=scatter2;
    dynamic _X='Catalyst' _Y1='Yield_a' _Y2='Yield';
run;

ods graphics off;
```

The top panel of Output 36.3.6 hints at the same kind of structure exhibited in the fitted cross sections of Output 36.3.3. In PROC GAM, the additive model component corresponding to Catalyst is fit to a similar scatter plot, with the partial residuals computed in the backfitting algorithm, so it is able to capture the trend seen here. In contrast, when the second data set is viewed from the perspective of Output 36.3.6, the diagonal ridge apparent in Output 36.3.4 is washed out, and no clear structure shows up in the scatter plot. As a result, the additive model fit produced by PROC GAM is relatively featureless.

Output 36.3.6 Scatter Plots of Yield by Catalyst



References

Allen, D. M. (1974), “The Relationship between Variable Selection and Data Augmentation and a Method of Prediction,” *Technometrics*, 16, 125–127.

Bell, D. F., Walker J. L., O’Connor, G., and Tibshirani, R. J. (1994), “Spinal Deformity after Multiple-Level Cervical Laminectomy in Children.” *Spine*, 19, 406–411.

Cleveland, W. S., Devlin, S. J., and Grosse, E. (1988), “Regression by Local Fitting,” *Journal of Econometrics,* 37, 87–114.

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*, ed. W. Schempp and K. Zeller, New York: Springer-Verlag, 85–100.

Friedman, J. H. and Stuetzle, W. (1981), "Projection Pursuit Regression," *Journal of the American Statistical Association*, 76, 817–823.

Hastie, T. J. (1991), "Generalized Additive Models," in *Statistical Models in S*, ed. J. M. Chambers and T. J. Hastie, Pacific Grove: Wadsworth & Brooks/Cole Advanced Books & Software, 249–307.

Hastie, T. J. and Tibshirani, R. J. (1986), "Generalized Additive Models (with discussion)," *Statistical Science*, 1, 297–318.

Hastie, T. J. and Tibshirani, R. J. (1990), *Generalized Additive Models*, New York: Chapman & Hall.

Houghton, A. N., Flannery, J., and Viola, M. V. (1980), "Malignant Melanoma in Connecticut and Denmark," *International Journal of Cancer*, 25, 95–104.

McCullagh, P. and Nelder, J. A. (1989), *Generalized Linear Models*, Second Edition, London: Chapman & Hall.

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 Language Reference: Concepts, Version 8*, Cary, NC: SAS Institute Inc.

SAS Institute Inc. (1999), *SAS Language Reference: Dictionary, Version 8*, Cary, NC: SAS Institute Inc.

SAS Institute Inc. (1999), *SAS Procedures Guide, Version 8*, Cary, NC: SAS Institute Inc.

SAS Institute Inc. (2004), *SAS/STAT 9.1 User's Guide*, Cary, NC: SAS Institute Inc.

Sockett, 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," *Diabetologia*, 30, 453–459.

Stone, C. J. (1985), "Additive Regression and Other Nonparametric Models," *Annals of Statistics*, 13, 689–705.

Wahba, G. (1983), "Bayesian 'Confidence Intervals' for the Cross Validated Smoothing Spline," *Journal of the Royal Statistical Society, Series B*, 45, 133–150.

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.

Subject Index

GAM procedure
  comparing PROC GAM with PROC LOESS, 1875
  Estimates from PROC GAM, 1858
  generalized additive model with binary data, 1860
  graphics, 1838
  ODS graph names, 1860
  ODS table names, 1859
  Poisson regression analysis of component reliability, 1869
  response level ordering, 1841
  response variable options, 1841

ODS graph names
  GAM procedure, 1860

response level ordering
  GAM procedure, 1841
response variable options
  GAM procedure, 1841
reverse response level ordering
  GAM procedure, 1841

Syntax Index

ADDITIVE option
    PROC GAM statement, 1838
all option
    PROC GAM statement, 1837
ALPHA= option
    MODEL statement (GAM), 1842
ANODEV= option
    MODEL statement (GAM), 1842

BY statement
    GAM procedure, 1838

CLASS statement
    GAM procedure, 1838
CLM option
    PROC GAM statement, 1838
COMMONAXES option
    PROC GAM statement, 1838
components option
    PROC GAM statement, 1837

DATA= option
    PROC GAM statement, 1836
    SCORE statement (GAM), 1845
DESCENDING option
    CLASS statement (GAM), 1839
    MODEL statement, 1841
    PROC GAM statement, 1836
DIST = option
    MODEL statement (GAM), 1843

EPSILON = option
    MODEL statement (GAM), 1843
EPSSCORE = option
    MODEL statement (GAM), 1843

FREQ statement
    GAM procedure, 1839

GAM procedure, 1836
    syntax, 1836
GAM procedure, BY statement, 1838
GAM procedure, CLASS statement, 1838
    DESCENDING option, 1839
    ORDER= option, 1839
    TRUNCATE option, 1839
GAM procedure, FREQ statement, 1839
GAM procedure, MODEL statement, 1840
    ALPHA= option, 1842
    ANODEV= option, 1842
    DESCENDING option, 1841
    DIST= option, 1843
    EPSILON= option, 1843
    EPSSCORE= option, 1843
    ITPRINT option, 1843
    MAXITER= option, 1843
    MAXITSCORE= option, 1843
    METHOD= option, 1843
    ORDER= option, 1842
GAM procedure, OUTPUT statement, 1844
    OUT= option, 1844
GAM procedure, PROC GAM statement, 1836
    ADDITIVE option, 1838
    all option, 1837
    CLM option, 1838
    COMMONAXES option, 1838
    component option, 1837
    DATA= option, 1836
    DESCENDING option, 1836
    none option, 1837
    ORDER option, 1836
    plots(unpack) option, 1837
    PLOTS= option, 1837
    UNPACKPANELS option, 1838
GAM procedure, SCORE statement, 1845
    DATA= option, 1845
    OUT= option, 1845

ITPRINT option
    MODEL statement (GAM), 1843

MAXITER = option
    MODEL statement (GAM), 1843
MAXITSCORE = option
    MODEL statement (GAM), 1843
METHOD= option
    MODEL statement (GAM), 1843
MODEL statement
    GAM procedure, 1840

none option
    PROC GAM statement, 1837

ORDER= option
    CLASS statement (GAM), 1839
    MODEL statement, 1842
    PROC GAM statement, 1836
OUT= option

    OUTPUT statement (GAM), 1844
    SCORE statement (GAM), 1845
OUTPUT statement
    GAM procedure, 1844

PLOTS= option
    PROC GAM statement, 1837
PROC GAM statement, *see* GAM procedure

SCORE statement, GAM procedure, 1845

TRUNCATE option
    CLASS statement (GAM), 1839

unpack option
    PROC GAM statement, 1837
UNPACKPANELS option
    PROC GAM statement, 1838

Your Turn

We welcome your feedback.

- If you have comments about this book, please send them to **`yourturn@sas.com`**. Include the full title and page numbers (if applicable).
- If you have comments about the software, please send them to **`suggest@sas.com`**.