# Chapter 53
## The HPFMM Procedure

## Contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Overview: HPFMM Procedure</td>
<td>4180</td>
</tr>
<tr>
<td>Basic Features</td>
<td>4181</td>
</tr>
<tr>
<td>PROC HPFMM Contrasted with PROC FMM</td>
<td>4182</td>
</tr>
<tr>
<td>Assumptions</td>
<td>4183</td>
</tr>
<tr>
<td>Notation for the Finite Mixture Model</td>
<td>4183</td>
</tr>
<tr>
<td>Homogeneous Mixtures</td>
<td>4183</td>
</tr>
<tr>
<td>Special Mixtures</td>
<td>4184</td>
</tr>
<tr>
<td>Getting Started: HPFMM Procedure</td>
<td>4184</td>
</tr>
<tr>
<td>Mixture Modeling for Binomial Overdispersion: “Student,” Pearson, Beer, and Yeast</td>
<td>4184</td>
</tr>
<tr>
<td>Modeling Zero-Inflation: Is It Better to Fish Poorly or Not to Have Fished at All?</td>
<td>4190</td>
</tr>
<tr>
<td>Looking for Multiple Modes: Are Galaxies Clustered?</td>
<td>4198</td>
</tr>
<tr>
<td>Comparison with Roeder’s Method</td>
<td>4205</td>
</tr>
<tr>
<td>Syntax: HPFMM Procedure</td>
<td>4208</td>
</tr>
<tr>
<td>PROC HPFMM Statement</td>
<td>4208</td>
</tr>
<tr>
<td>BAYES Statement</td>
<td>4220</td>
</tr>
<tr>
<td>BY Statement</td>
<td>4230</td>
</tr>
<tr>
<td>CLASS Statement</td>
<td>4230</td>
</tr>
<tr>
<td>FREQ Statement</td>
<td>4231</td>
</tr>
<tr>
<td>ID Statement</td>
<td>4231</td>
</tr>
<tr>
<td>MODEL Statement</td>
<td>4231</td>
</tr>
<tr>
<td>Response Variable Options</td>
<td>4233</td>
</tr>
<tr>
<td>Model Options</td>
<td>4234</td>
</tr>
<tr>
<td>OUTPUT Statement</td>
<td>4239</td>
</tr>
<tr>
<td>PERFORMANCE Statement</td>
<td>4242</td>
</tr>
<tr>
<td>PROBMODEL Statement</td>
<td>4242</td>
</tr>
<tr>
<td>RESTRICT Statement</td>
<td>4244</td>
</tr>
<tr>
<td>WEIGHT Statement</td>
<td>4246</td>
</tr>
<tr>
<td>Details: HPFMM Procedure</td>
<td>4246</td>
</tr>
<tr>
<td>A Gentle Introduction to Finite Mixture Models</td>
<td>4246</td>
</tr>
<tr>
<td>The Form of the Finite Mixture Model</td>
<td>4246</td>
</tr>
<tr>
<td>Mixture Models Contrasted with Mixing and Mixed Models: Untangling the Terminology Web</td>
<td>4247</td>
</tr>
<tr>
<td>Overdispersion</td>
<td>4249</td>
</tr>
<tr>
<td>Log-Likelihood Functions for Response Distributions</td>
<td>4249</td>
</tr>
<tr>
<td>Bayesian Analysis</td>
<td>4256</td>
</tr>
<tr>
<td>Conjugate Sampling</td>
<td>4256</td>
</tr>
</tbody>
</table>
Overview: HPFMM Procedure

The HPFMM procedure is a high-performance counterpart of the FMM procedure that fits statistical models to data for which the distribution of the response is a finite mixture of univariate distributions—that is, each response comes from one of several random univariate distributions that have unknown probabilities. You can use PROC HPFMM to model the component distributions in addition to the mixing probabilities. For more precise definitions and a discussion of similar but distinct modeling methodologies, see the section “A Gentle Introduction to Finite Mixture Models” on page 4246.

The HPFMM procedure is designed to fit finite mixtures of regression models or finite mixtures of generalized linear models in which the covariates and regression structure can be the same across components or can be different. You can fit finite mixture models by maximum likelihood or Bayesian methods. Note that classical statistical models are a special case of the finite mixture models in which the distribution of the data has only a single component.

PROC HPFMM runs in either single-machine mode or distributed mode.

NOTE: Distributed mode requires SAS High-Performance Statistics.
Basic Features

The HPFMM procedure estimates the parameters in univariate finite mixture models and produces various statistics to evaluate parameters and model fit. The following list summarizes some basic features of the HPFMM procedure:

- maximum likelihood estimation for all models
- Markov chain Monte Carlo estimation for many models, including zero-inflated Poisson models
- many built-in link and distribution functions for modeling, including the beta, shifted $t$, Weibull, beta-binomial, and generalized Poisson distributions, in addition to many standard members of the exponential family of distributions
- specialized built-in mixture models such as the binomial cluster model (Morel and Nagaraj 1993; Morel and Neerchal 1997; Neerchal and Morel 1998)
- acceptance of multiple MODEL statements to build mixture models in which the model effects, distributions, or link functions vary across mixture components
- model-building syntax using CLASS and effect-based MODEL statements familiar from many other SAS/STAT procedures (for example, the GLM, GLIMMIX, and MIXED procedures)
- evaluation of sequences of mixture models when you specify ranges for the number of components
- simple syntax to impose linear equality and inequality constraints among parameters
- ability to model regression and classification effects in the mixing probabilities through the PROBMODEL statement
- ability to incorporate full or partially known component membership into the analysis through the PARTIAL= option in the PROC HPFMM statement
- OUTPUT statement that produces a SAS data set with important statistics for interpreting mixture models, such as component log likelihoods and prior and posterior probabilities
- ability to add zero-inflation to any model
- output data set with posterior parameter values for the Markov chain
- multithreading and distributed computing for high-performance optimization and Monte Carlo sampling

The HPFMM procedure uses ODS Graphics to create graphs as part of its output. For general information about ODS Graphics, see Chapter 21, “Statistical Graphics Using ODS.” For specific information about the statistical graphics available with the HPFMM procedure, see the PLOTS options in the PROC HPFMM statement.

Because the HPFMM procedure is a high-performance analytical procedure, it also does the following:

- enables you to run in distributed mode on a cluster of machines that distribute the data and the computations
Chapter 53: The HPFMM Procedure

- enables you to run in single-machine mode on the server where SAS is installed
- exploits all the available cores and concurrent threads, regardless of execution mode

For more information, see the section “Processing Modes” (Chapter 2, SAS/STAT User’s Guide: High-Performance Procedures).

PROC HPFMM Contrasted with PROC FMM


The HPFMM procedure is somewhat distinct from other high-performance analytical procedures in being very nearly a twin of its counterpart, PROC FMM. You can fit the same kinds of models and get the same kinds of tabular, graphical, and data set results from PROC HPFMM as from PROC FMM. The main difference is that PROC HPFMM was developed primarily to work in a distributed environment, and PROC FMM primarily for a single (potentially multithreaded) host.

PROC HPFMM and PROC FMM have several differences because of their respective underlying technology:

- The ORDER option that specifies the sort order for the levels of CLASS variables is not available in the PROC statement of the HPFMM procedure. Instead the HPFMM procedure makes this option available in the CLASS statement.
- The CLASS statement in the HPFMM procedure provides many more options than the CLASS statement in the FMM procedure.
- The PERFORMANCE statement in the HPFMM procedure includes a superset of the options that are available in the PERFORMANCE statement in the FMM procedure.
- The NOVAR option in the OUTPUT statement in the FMM procedure is not available in the OUTPUT statement of the HPFMM procedure.

The OUTPUT statement in PROC HPFMM produces observationwise statistics. However, as is customary for SAS high-performance analytical procedures, PROC HPFMM’s OUTPUT statement does not by default include the input and BY variables in the output data set. This is to avoid data duplication for large data sets. In order to include any input or BY variables in the output data set, you must list these variables in the ID statement. Furthermore, PROC HPFMM’s OUTPUT statement includes the predicted values of the response variable if you do not specify any output statistics.

In contrast, when you request that the posterior sample be saved to a SAS data by specifying the OUTPOST= option in the BAYES statement, PROC HPFMM includes the BY variables in the data set.
Assumptions

The HPFMM procedure makes the following assumptions in fitting statistical models:

- The number of components \( k \) in the finite mixture is known a priori and is not a parameter to be estimated.
- The parameters of the components are distinct a priori.
- The observations are uncorrelated.

Notation for the Finite Mixture Model

The general expression for the finite mixture model fitted with the HPFMM procedure is as follows:

\[
  f(y) = \sum_{j=1}^{k} \pi_j(z, \alpha_j) p_j(y; x'_j \beta_j, \phi_j)
\]

The number of components in the mixture is denoted as \( k \). The mixture probabilities \( \pi_j \) can depend on regressor variables \( z \) and parameters \( \alpha_j \). By default, the HPFMM procedure models these probabilities using a logit transform if \( k = 2 \) and as a generalized logit model if \( k > 2 \). The component distributions \( p_j \) can also depend on regressor variables in \( x_j \), regression parameters \( \beta_j \), and possibly scale parameters \( \phi_j \). Notice that the component distributions \( p_j \) are indexed by \( j \) since the distributions might belong to different families. For example, in a two-component model, you might model one component as a normal (Gaussian) variable and the second component as a variable with a \( t \) distribution with low degrees of freedom to manage overdispersion.

The mixture probabilities \( \pi_j \) satisfy \( \pi_j \geq 0 \), for all \( j \), and

\[
  \sum_{j=1}^{k} \pi_j(z, \alpha_j) = 1
\]

Homogeneous Mixtures

If the component distributions are of the same distributional form, the mixture is called homogeneous. In most applications of homogeneous mixtures, the mixing probabilities do not depend on regression parameters. The general model then simplifies to

\[
  f(y) = \sum_{j=1}^{k} \pi_j p(y; x'_j \beta_j, \phi_j)
\]
Since the component distributions depend on regression parameters $\beta_j$, this model is known as a homogeneous regression mixture. A homogeneous regression mixture assumes that the regression effects are the same across the components, although the HPFMM procedure does not impose such a restriction. If the component distributions do not contain regression effects, the model

$$f(y) = \sum_{j=1}^{k} \pi_j p(y; \mu_j, \phi_j)$$

is the homogeneous mixture model. A classical case is the estimation of a continuous density as a $k$-component mixture of normal distributions.

**Special Mixtures**

The HPFMM procedure enables you to fit several special mixture models. The Morel-Neerchal binomial cluster model (Morel and Nagaraj 1993; Morel and Neerchal 1997; Neerchal and Morel 1998) is a mixture of binomial distributions in which the success probabilities depend on the mixing probabilities.

Zero-inflated count models are obtained as two-component mixtures where one component is a classical count model—such as the Poisson or negative binomial model—and the other component is a distribution that is concentrated at zero. If the nondegenerate part of this special mixture is a zero-truncated model, the resulting two-component mixture is known as a hurdle model (Cameron and Trivedi 1998).

**Getting Started: HPFMM Procedure**

**Mixture Modeling for Binomial Overdispersion: “Student,” Pearson, Beer, and Yeast**

The following example demonstrates how you can model a complicated, two-component binomial mixture distribution, either with maximum likelihood or with Bayesian methods, with a few simple PROC HPFMM statements.

William Sealy Gosset, a chemist at the Arthur Guinness Son and Company brewery in Dublin, joined the statistical laboratory of Karl Pearson in 1906–1907 to study statistics. At first Gosset—who published all but one paper under the pseudonym “Student” because his employer forbade publications by employees after a co-worker had disclosed trade secrets—worked on the Poisson limit to the binomial distribution, using haemacytometer yeast cell counts. Gosset’s interest in studying small-sample (and limit) problems was motivated by the small sample sizes he typically saw in his work at the brewery.

Subsequently, Gosset’s yeast count data have been examined and revisited by many authors. In 1915, Karl Pearson undertook his own examination and realized that the variability in “Student’s” data exceeded that consistent with a Poisson distribution. Pearson (1915) bemoans the fact that if this were so, “it is certainly most unfortunate that such material should have been selected to illustrate Poisson’s limit to the binomial.”

Using a count of Gosset’s yeast cell counts on the 400 squares of a haemacytometer (Table 53.1), Pearson argues that a mixture process would explain the heterogeneity (beyond the Poisson).
Pearson fits various models to these data, chief among them a mixture of two binomial series
\[ v_1(p_1 + q_1)^\theta + v_2(p_2 + q_2)^\theta \]
where \( \theta \) is real-valued and thus the binomial series expands to
\[
(p + q)^\theta = \sum_{k=0}^{\infty} \frac{\Gamma(\theta + 1)}{\Gamma(k + 1)\Gamma(\theta - k + 1)} p^k q^{\theta - k}
\]

Pearson’s fitted model has \( \theta = 4.89997, v_1 = 356.986, v_2 = 43.014 \) (corresponding to a mixing proportion of \( 356.986/(43.014 + 356.986) = 0.892 \)), and estimated success probabilities in the binomial components of 0.1017 and 0.4514, respectively. The success probabilities indicate that although the data have about a 90% chance of coming from a distribution with small success probability of about 0.1, there is a 10% chance of coming from a distribution with a much larger success probability of about 0.45.

If \( \theta \) is an integer, the binomial series is the cumulative mass function of a binomial random variable. The value of \( \theta \) suggests that a suitable model for these data could also be constructed as a two-component mixture of binomial random variables as follows:

\[ f(y) = \pi \text{binomial}(5, \mu_1) + (1 - \pi) \text{binomial}(5, \mu_2) \]

The binomial sample size \( n=5 \) is suggested by Pearson’s estimate of \( \theta = 4.89997 \) and the fact that the largest cell count in Table 53.1 is 5.

The following DATA step creates a SAS data set from the data in Table 53.1.

```sas
data yeast;
  input count f;
  n = 5;
datalines;
  0 213
  1 128
  2 37
  3 18
  4 3
  5 1;
;
```

The two-component binomial model is fit with the HPFMM procedure with the following statements:

```sas
proc hpfmm data=yeast;
  model count/n = / k=2;
  freq f;
run;
```
Because the events/trials syntax is used in the MODEL statement, PROC HPFMM defaults to the binomial distribution. The K=2 option specifies that the number of components is fixed and known to be two. The FREQ statement indicates that the data are grouped; for example, the first observation represents 213 squares on the haemacytometer where no yeast cells were found.

The “Model Information” and “Number of Observations” tables in Figure 53.1 convey that the fitted model is a two-component homogeneous binomial mixture with a logit link function. The mixture is homogeneous because there are no model effects in the MODEL statement and because both component distributions belong to the same distributional family. By default, PROC HPFMM estimates the model parameters by maximum likelihood.

Although only six observations are read from the data set, the data represent 400 observations (squares on the haemacytometer). Since a constant binomial sample size of 5 is assumed, the data represent 273 successes (finding a yeast cell) out of 2,000 Bernoulli trials.

![Figure 53.1 Model Information for Yeast Cell Model](image)

The estimated intercepts (on the logit scale) for the two binomial means are –2.2316 and –0.2974, respectively. These values correspond to binomial success probabilities of 0.09695 and 0.4262, respectively (Figure 53.2). The two components mix with probabilities 0.8799 and $0.8799 = 0.1201$. These values are generally close to the values found by Pearson (1915) using infinite binomial series instead of binomial mass functions.

![Figure 53.2 Maximum Likelihood Estimates](image)
To obtain fitted values and other observationwise statistics under the stipulated two-component model, you can add the OUTPUT statement to the previous PROC HPFMM run. The following statements request componentwise predicted values and the posterior probabilities:

```plaintext
proc hpfmm data=yeast;
   model count/n = / k=2;
   freq f;
   id f n;
   output out=hpfmmout pred(components) posterior;
run;

data hpfmmout;
   set hpfmmout;
   PredCount_1 = post_1 * f;
   PredCount_2 = post_2 * f;
run;
proc print data=hpfmmout;
run;
```

The DATA step following the PROC HPFMM step computes the predicted cell counts in each component (Figure 53.3). Note that the predicted means in the components, 0.48476 and 2.13099, are close to the values determined by Pearson (0.4983 and 2.2118), as are the predicted cell counts.

<table>
<thead>
<tr>
<th>Obs</th>
<th>F</th>
<th>N</th>
<th>Pred_1</th>
<th>Pred_2</th>
<th>Post_1</th>
<th>Post_2</th>
<th>PredCount_1</th>
<th>PredCount_2</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>213</td>
<td>5</td>
<td>0.096951</td>
<td>0.42620</td>
<td>0.98606</td>
<td>0.01394</td>
<td>210.030</td>
<td>2.9698</td>
</tr>
<tr>
<td>2</td>
<td>128</td>
<td>5</td>
<td>0.096951</td>
<td>0.42620</td>
<td>0.91089</td>
<td>0.08911</td>
<td>116.594</td>
<td>11.4058</td>
</tr>
<tr>
<td>3</td>
<td>37</td>
<td>5</td>
<td>0.096951</td>
<td>0.42620</td>
<td>0.59638</td>
<td>0.40362</td>
<td>22.066</td>
<td>14.9341</td>
</tr>
<tr>
<td>4</td>
<td>18</td>
<td>5</td>
<td>0.096951</td>
<td>0.42620</td>
<td>0.17598</td>
<td>0.82402</td>
<td>3.168</td>
<td>14.8323</td>
</tr>
<tr>
<td>5</td>
<td>3</td>
<td>5</td>
<td>0.096951</td>
<td>0.42620</td>
<td>0.02994</td>
<td>0.97006</td>
<td>0.090</td>
<td>2.9102</td>
</tr>
<tr>
<td>6</td>
<td>1</td>
<td>5</td>
<td>0.096951</td>
<td>0.42620</td>
<td>0.00444</td>
<td>0.99556</td>
<td>0.004</td>
<td>0.9956</td>
</tr>
</tbody>
</table>

Gosset, who was interested in small-sample statistical problems, investigated the use of prior knowledge in mathematical-statistical analysis—for example, deriving the sampling distribution of the correlation coefficient after having assumed a uniform prior distribution for the coefficient in the population (Aldrich 1997). Pearson also was not opposed to using prior information, especially uniform priors that reflect “equal distribution of ignorance.” Fisher, on the other hand, would not have any of it: the best estimator in his opinion is obtained by a criterion that is absolutely independent of prior assumptions about probabilities of particular values. He objected to the insinuation that his derivations in the work on the correlation were deduced from Bayes theorem (Fisher 1921).
The preceding analysis of the yeast cell count data uses maximum likelihood methods that are free of prior assumptions. The following analysis takes instead a Bayesian approach, assuming a beta prior distribution for the binomial success probabilities and a uniform prior distribution for the mixing probabilities. The changes from the previous run of PROC HPFMM are the addition of the ODS GRAPHICS, PERFORMANCE, and BAYES statements and the SEED=12345 option.

```plaintext
ods graphics on;
proc hpfmm data=yeast seed=12345;
  model count/n = / k=2;
  freq f;
  performance nthreads=2;
  bayes;
run;
ods graphics off;
```

When ODS Graphics is enabled, PROC HPFMM produces diagnostic trace plots for the posterior samples. Bayesian analyses are sensitive to the random number seed and thread count; the SEED= and NTHREADS= options in the PERFORMANCE statement ensure consistent results for the purposes of this example. The SEED=12345 option in the PROC HPFMM statement determines the random number seed for the random number generator that the analysis used. The NTHREADS=2 option in the PERFORMANCE statement sets the number of threads to be used by the procedure to two. The BAYES statement requests a Bayesian analysis.

The “Bayes Information” table in Figure 53.4 provides basic information about the Markov chain Monte Carlo sampler. Because the model is a homogeneous mixture, the HPFMM procedure applies an efficient conjugate sampling algorithm with a posterior sample size of 10,000 samples after a burn-in size of 2,000 samples. The “Prior Distributions” table displays the prior distribution for each parameter along with its mean and variance and the initial value in the chain. Notice that in this situation all three prior distributions reduce to a uniform distribution on \((0, 1)\).

**Figure 53.4** Basic Information about MCMC Sampler

<table>
<thead>
<tr>
<th>Bayes Information</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Sampling Algorithm</td>
<td>Conjugate</td>
</tr>
<tr>
<td>Data Augmentation</td>
<td>Latent Variable</td>
</tr>
<tr>
<td>Initial Values of Chain</td>
<td>Data Based</td>
</tr>
<tr>
<td>Burn-In Size</td>
<td>2000</td>
</tr>
<tr>
<td>MC Sample Size</td>
<td>10000</td>
</tr>
<tr>
<td>MC Thinning</td>
<td>1</td>
</tr>
<tr>
<td>Parameters in Sampling</td>
<td>3</td>
</tr>
<tr>
<td>Mean Function Parameters</td>
<td>2</td>
</tr>
<tr>
<td>Scale Parameters</td>
<td>0</td>
</tr>
<tr>
<td>Mixing Prob Parameters</td>
<td>1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Prior Distributions</th>
<th>Distribution</th>
<th>Mean</th>
<th>Variance</th>
<th>Initial Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 Success Probability</td>
<td>Beta(1, 1)</td>
<td>0.5000</td>
<td>0.08333</td>
<td>0.1365</td>
</tr>
<tr>
<td>2 Success Probability</td>
<td>Beta(1, 1)</td>
<td>0.5000</td>
<td>0.08333</td>
<td>0.1365</td>
</tr>
<tr>
<td>1 Probability</td>
<td>Dirichlet(1, 1)</td>
<td>0.5000</td>
<td>0.08333</td>
<td>0.6180</td>
</tr>
</tbody>
</table>
The HPFMM procedure produces a log note for this model, indicating that the sampled quantities are not the linear predictors on the logit scale, but are the actual population parameters (on the data scale):

**NOTE:** Bayesian results for this model (no regressor variables, non-identity link) are displayed on the data scale, not the linked scale. You can obtain results on the linked (=linear) scale by requesting a Metropolis-Hastings sampling algorithm.

The trace panel for the success probability in the first binomial component is shown in Figure 53.5. Note that the first component in this Bayesian analysis corresponds to the second component in the MLE analysis. The graphics in this panel can be used to diagnose the convergence of the Markov chain. If the chain has not converged, inferences cannot be made based on quantities derived from the chain. You generally look for the following:

- a smooth unimodal distribution of the posterior estimates in the density plot displayed on the lower right
- good mixing of the posterior samples in the trace plot at the top of the panel (good mixing is indicated when the trace traverses the support of the distribution and appears to have reached a stationary distribution)

**Figure 53.5** Trace Panel for Success Probability in First Component
The autocorrelation plot in Figure 53.5 shows fairly high and sustained autocorrelation among the posterior estimates. While this is generally not a problem, you can affect the degree of autocorrelation among the posterior estimates by running a longer chain and thinning the posterior estimates; see the NMC= and THIN= options in the BAYES statement.

Both the trace plot and the density plot in Figure 53.5 are indications of successful convergence.

Figure 53.6 reports selected results that summarize the 10,000 posterior samples. The arithmetic means of the success probabilities in the two components are 0.0917 and 0.3974, respectively. The posterior mean of the mixing probability is 0.8312. These values are similar to the maximum likelihood parameter estimates in Figure 53.2 (after swapping components).

**Figure 53.6** Summaries for Posterior Estimates

<table>
<thead>
<tr>
<th>Component Parameter</th>
<th>N</th>
<th>Mean</th>
<th>Standard Deviation</th>
<th>25</th>
<th>50</th>
<th>75</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 Success Probability</td>
<td>10000</td>
<td>0.0917</td>
<td>0.0168</td>
<td>0.0830</td>
<td>0.0938</td>
<td>0.1027</td>
</tr>
<tr>
<td>2 Success Probability</td>
<td>10000</td>
<td>0.3974</td>
<td>0.0871</td>
<td>0.3379</td>
<td>0.3957</td>
<td>0.4556</td>
</tr>
<tr>
<td>1 Probability</td>
<td>10000</td>
<td>0.8312</td>
<td>0.1045</td>
<td>0.7986</td>
<td>0.8578</td>
<td>0.8984</td>
</tr>
</tbody>
</table>

Note that the standard errors in Figure 53.2 are not comparable to those in Figure 53.6, since the standard errors for the MLEs are expressed on the logit scale and the Bayes estimates are expressed on the data scale. You can add the METROPOLIS option in the BAYES statement to sample the quantities on the logit scale.

The “Posterior Intervals” table in Figure 53.6 displays 95% credible intervals (equal-tail intervals and intervals of highest posterior density). It can be concluded that the component with the higher success probability contributes less than 40% to the process.

### Modeling Zero-Inflation: Is It Better to Fish Poorly or Not to Have Fished at All?

The following example shows how you can use PROC HPFMM to model data with more zero values than expected.

Many count data show an excess of zeros relative to the frequency of zeros expected under a reference model. An excess of zeros leads to overdispersion since the process is more variable than a standard count data model. Different mechanisms can lead to excess zeros. For example, suppose that the data are generated from two processes with different distribution functions—one process generates the zero counts, and the other process generates nonzero counts. In the vernacular of Cameron and Trivedi (1998), such a model is called a hurdle model. With a certain probability—the probability of a nonzero count—a hurdle is crossed, and events are being generated. Hurdle models are useful, for example, to model the number of doctor visits.
per year. Once the decision to see a doctor has been made—the hurdle has been overcome—a certain number of visits follow.

Hurdle models are closely related to zero-inflated models. Both can be expressed as two-component mixtures in which one component has a degenerate distribution at zero and the other component is a count model. In a hurdle model, the count model follows a zero-truncated distribution. In a zero-inflated model, the count model has a nonzero probability of generating zeros. Formally, a zero-inflated model can be written as

$$\Pr(Y = y) = \pi p_1 + (1 - \pi) p_2(y, \mu)$$

$$p_1 = \begin{cases} 1 & y = 0 \\ 0 & \text{otherwise} \end{cases}$$

where $p_2(y, \mu)$ is a standard count model with mean $\mu$ and support $y \in \{0, 1, 2, \ldots\}$.

The following data illustrates the use of a zero-inflated model. In a survey of park attendees, randomly selected individuals were asked about the number of fish they caught in the last six months. Along with that count, the gender and age of each sampled individual was recorded. The following DATA step displays the data for the analysis:

```plaintext
data catch;
  input gender $ age count @@;
  datalines;
    F  54  18  M  37  0  F  48  12  M  27  0
    M  55  0   M  32  0  F  49  12  F  45  11
    M  39  0   F  34  1  F  50  0   M  52  4
    M  33  0   M  32  0  F  23  1   F  17  0
    F  44  5   M  44  0  F  26  0   F  30  0
    F  38  0   F  38  0  F  52  18  M  23  1
    F  23  0   M  32  0  F  33  3   M  26  0
    F  46  8   M  45  5  M  51  10  F  48  5
    F  31  2   F  25  1   M  22  0   M  41  0
    M  19  0   M  23  0  M  31  1   M  17  0
    F  21  0   F  44  7   M  28  0   M  47  3
    M  23  0   F  29  3   F  24  0   M  34  1
    F  19  0   F  35  2   M  39  0   M  43  6
  ;
```

At first glance, the prevalence of zeros in the DATA set is apparent. Many park attendees did not catch any fish. These zero counts are made up of two populations: attendees who do not fish and attendees who fish poorly. A zero-inflation mechanism thus appears reasonable for this application since a zero count can be produced by two separate distributions.

The following statements fit a standard Poisson regression model to these data. A common intercept is assumed for men and women, and the regression slope varies with gender.

```plaintext
proc hpfmm data=catch;
    class gender;
    model count = gender*age / dist=Poisson;
run;
```

Figure 53.7 displays information about the model and data set. The “Model Information” table conveys that the model is a single-component Poisson model (a Poisson GLM) and that parameters are estimated by maximum likelihood. There are two levels in the CLASS variable gender, with females preceding males.
Chapter 53: The HPFMM Procedure

Figure 53.7 Model Information and Class Levels in Poisson Regression

<table>
<thead>
<tr>
<th>The HPFMM Procedure</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Model Information</strong></td>
</tr>
<tr>
<td>Data Set</td>
</tr>
<tr>
<td>Response Variable</td>
</tr>
<tr>
<td>Type of Model</td>
</tr>
<tr>
<td>Distribution</td>
</tr>
<tr>
<td>Components</td>
</tr>
<tr>
<td>Link Function</td>
</tr>
<tr>
<td>Estimation Method</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Class Level Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Class</td>
</tr>
<tr>
<td>gender</td>
</tr>
</tbody>
</table>

Number of Observations Read 52
Number of Observations Used 52

The “Fit Statistics” and “Parameter Estimates” tables from the maximum likelihood estimation of the Poisson GLM are shown in Figure 53.8. If the model is not overdispersed, the Pearson statistic should roughly equal the number of observations in the data set minus the number of parameters. With n=52, there is evidence of overdispersion in these data.

Figure 53.8 Fit Results in Poisson Regression

<table>
<thead>
<tr>
<th>Fit Statistics</th>
</tr>
</thead>
<tbody>
<tr>
<td>-2 Log Likelihood</td>
</tr>
<tr>
<td>AIC (Smaller is Better)</td>
</tr>
<tr>
<td>AICC (Smaller is Better)</td>
</tr>
<tr>
<td>BIC (Smaller is Better)</td>
</tr>
<tr>
<td>Pearson Statistic</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Parameter Estimates for Poisson Model</th>
</tr>
</thead>
<tbody>
<tr>
<td>Effect</td>
</tr>
<tr>
<td>Intercept</td>
</tr>
<tr>
<td>age*gender</td>
</tr>
<tr>
<td>age*gender</td>
</tr>
</tbody>
</table>

Suppose that the cause of overdispersion is zero-inflation of the count data. The following statements fit a zero-inflated Poisson model.

```latex
proc hpfmm data=catch;
    class gender;
    model count = gender*age / dist=Poisson;
    model + / dist=Constant;
run;
```
There are two `MODEL` statements, one for each component of the mixture. Because the distributions are different for the components, you cannot specify the mixture model with a single `MODEL` statement. The first `MODEL` statement identifies the response variable for the model (`count`) and defines a Poisson model with intercept and gender-specific slopes. The second `MODEL` statement uses the continuation operator ("+") and adds a model with a degenerate distribution by using `DIST=CONSTANT`. Because the mass of the constant is placed by default at zero, the second `MODEL` statement adds a zero-inflation component to the model. It is sufficient to specify the response variable in one of the `MODEL` statements; you use the "=" sign in that statement to separate the response variable from the model effects.

**Figure 53.9** displays the “Model Information” and “Optimization Information” tables for this run of the HPFMM procedure. The model is now identified as a zero-inflated Poisson (ZIP) model with two components, and the parameters continue to be estimated by maximum likelihood. The “Optimization Information” table shows that there are four parameters in the optimization (compared to three parameters in the Poisson GLM model). The four parameters correspond to three parameters in the mean function (intercept and two gender-specific slopes) and the mixing probability.

**Figure 53.9** Model and Optimization Information in the ZIP Model

<table>
<thead>
<tr>
<th>Model Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data Set</td>
</tr>
<tr>
<td>Response Variable</td>
</tr>
<tr>
<td>Type of Model</td>
</tr>
<tr>
<td>Components</td>
</tr>
<tr>
<td>Estimation Method</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Optimization Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Optimization Technique</td>
</tr>
<tr>
<td>Parameters in Optimization</td>
</tr>
<tr>
<td>Mean Function Parameters</td>
</tr>
<tr>
<td>Scale Parameters</td>
</tr>
<tr>
<td>Mixing Prob Parameters</td>
</tr>
</tbody>
</table>

Results from fitting the ZIP model by maximum likelihood are shown in **Figure 53.10**. The –2 log likelihood and the information criteria suggest a much-improved fit over the single-component Poisson model (compare **Figure 53.10** to **Figure 53.8**). The Pearson statistic is reduced by factor 2 compared to the Poisson model and suggests a better fit than the standard Poisson model.

**Figure 53.10** Maximum Likelihood Results for the ZIP model

<table>
<thead>
<tr>
<th>Fit Statistics</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>-2 Log Likelihood</td>
<td>145.6</td>
</tr>
<tr>
<td>AIC (Smaller is Better)</td>
<td>153.6</td>
</tr>
<tr>
<td>AICC (Smaller is Better)</td>
<td>154.5</td>
</tr>
<tr>
<td>BIC (Smaller is Better)</td>
<td>161.4</td>
</tr>
<tr>
<td>Pearson Statistic</td>
<td>43.467</td>
</tr>
<tr>
<td>Effective Parameters</td>
<td>4</td>
</tr>
<tr>
<td>Effective Components</td>
<td>2</td>
</tr>
</tbody>
</table>
The number of effective parameters and components shown in Figure 53.8 equals the values from Figure 53.9. This is not always the case because components can collapse (for example, when the mixing probability approaches zero or when two components have identical parameter estimates). In this example, both components and all four parameters are identifiable. The Poisson regression and the zero process mix, with a probability of approximately 0.6972 attributed to the Poisson component.

The HPFMM procedure enables you to fit some mixture models by Bayesian techniques. The following statements add the BAYES statement to the previous PROC HPFMM statements:

```sas
proc hpfmm data=catch seed=12345;
  class gender;
  model count = gender*age / dist=Poisson;
  model + / dist=constant;
  performance nthreads=2;
  bayes;
run;
```

The “Model Information” table indicates that the model parameters are estimated by Markov chain Monte Carlo techniques, and it displays the random number seed (Figure 53.11). This is useful if you did not specify a seed to identify the seed value that reproduces the current analysis. The “Bayes Information” table provides basic information about the Monte Carlo sampling scheme. The sampling method uses a data augmentation scheme to impute component membership and then the Gamerman (1997) algorithm to sample the component-specific parameters. The 2,000 burn-in samples are followed by 10,000 Monte Carlo samples without thinning.
The “Prior Distributions” table identifies the prior distributions, their parameters for the sampled quantities, and their initial values. The prior distribution of parameters associated with model effects is a normal distribution with mean 0 and variance 1,000. The prior distribution for the mixing probability is a Dirichlet(1, 1), which is identical to a uniform distribution (Figure 53.11). Since the second mixture component is a degeneracy at zero with no associated parameters, it does not appear in the “Prior Distributions” table in Figure 53.11.
Figure 53.12 displays descriptive statistics about the 10,000 posterior samples. Recall from Figure 53.10 that the maximum likelihood estimates were –3.5215, 0.1216, 0.1056, and 0.6972, respectively. With this choice of prior, the means of the posterior samples are generally close to the MLEs in this example. The “Posterior Intervals” table displays 95% intervals of equal-tail probability and 95% intervals of highest posterior density (HPD) intervals.

**Figure 53.12** Posterior Summaries and Intervals in the ZIP Model

<table>
<thead>
<tr>
<th>Component</th>
<th>Effect</th>
<th>gender</th>
<th>N</th>
<th>Mean</th>
<th>Standard Deviation</th>
<th>25</th>
<th>50</th>
<th>75</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Intercept</td>
<td></td>
<td>10000</td>
<td>-3.5456</td>
<td>0.6463</td>
<td>-3.9822</td>
<td>-3.5286</td>
<td>-3.1082</td>
</tr>
<tr>
<td>1</td>
<td>age*gender F</td>
<td></td>
<td>10000</td>
<td>0.1219</td>
<td>0.0135</td>
<td>0.1129</td>
<td>0.1216</td>
<td>0.1310</td>
</tr>
<tr>
<td>1</td>
<td>age*gender M</td>
<td></td>
<td>10000</td>
<td>0.1057</td>
<td>0.0140</td>
<td>0.0962</td>
<td>0.1053</td>
<td>0.1148</td>
</tr>
<tr>
<td>1</td>
<td>Probability</td>
<td></td>
<td>10000</td>
<td>0.6923</td>
<td>0.0950</td>
<td>0.6280</td>
<td>0.6960</td>
<td>0.7589</td>
</tr>
</tbody>
</table>

**Figure 53.12** Posterior Intervals

<table>
<thead>
<tr>
<th>Component</th>
<th>Effect</th>
<th>gender</th>
<th>Alpha</th>
<th>Equal-Tail Interval</th>
<th>HPD Interval</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Intercept</td>
<td></td>
<td>0.050</td>
<td>-4.8615 –2.3246</td>
<td>-4.8979 –2.3808</td>
</tr>
<tr>
<td>1</td>
<td>age*gender F</td>
<td></td>
<td>0.050</td>
<td>0.0955 0.1490</td>
<td>0.0963 0.1495</td>
</tr>
<tr>
<td>1</td>
<td>age*gender M</td>
<td></td>
<td>0.050</td>
<td>0.0784 0.1338</td>
<td>0.0786 0.1339</td>
</tr>
<tr>
<td>1</td>
<td>Probability</td>
<td></td>
<td>0.050</td>
<td>0.4999 0.8683</td>
<td>0.5139 0.8799</td>
</tr>
</tbody>
</table>

You can generate trace plots for the posterior parameter estimates by enabling ODS Graphics:

```r
ods graphics on;
ods select TADPanel;
proc hpfmm data=catch seed=12345;
   class gender;
   model count = gender*age / dist=Poisson;
   model + / dist=constant;
   performance nthreads=2;
   bayes;
run;
ods graphics off;
```
A separate trace panel is produced for each sampled parameter, and the panels for the gender-specific slopes are shown in Figure 53.13. There is good mixing in the chains: the modest autocorrelation that diminishes after about 10 successive samples. By default, the HPFMM procedure transfers the credible intervals for each parameter from the “Posterior Intervals” table to the trace plot and the density plot in the trace panel.

**Figure 53.13** Trace Panels for Gender-Specific Slopes
Looking for Multiple Modes: Are Galaxies Clustered?

Mixture modeling is essentially a generalized form of one-dimensional cluster analysis. The following example shows how you can use PROC HPFMM to explore the number and nature of Gaussian clusters in univariate data.

Roeder (1990) presents data from the Corona Borealis sky survey with the velocities of 82 galaxies in a narrow slice of the sky. Cosmological theory suggests that the observed velocity of each galaxy is proportional to its distance from the observer. Thus, the presence of multiple modes in the density of these velocities could indicate a clustering of the galaxies at different distances.

The following DATA step recreates the data set in Roeder (1990). The computed variable \( v \) represents the measured velocity in thousands of kilometers per second.
Analysis of potentially multimodal data is a natural application of finite mixture models. In this case, the modeling is complicated by the question of the variance for each of the components. Using identical variances for each component could obscure underlying structure, but the additional flexibility granted by component-specific variances might introduce spurious features.

You can use PROC HPFMM to prepare analyses for equal and unequal variances and use one of the available fit statistics to compare the resulting models. You can use the model selection facility to explore models with varying numbers of mixture components—say, from three to seven as investigated in Roeder (1990). The following statements select the best unequal-variance model using Akaike’s information criterion (AIC), which has a built-in penalty for model complexity:

```
title2 "Three to Seven Components, Unequal Variances";
ods graphics on;
proc hpfmm data=galaxies criterion=AIC;
   model v = / kmin=3 kmax=7;
   ods exclude IterHistory OptInfo ComponentInfo;
run;
```

The KMIN= and KMAX= options indicate the smallest and largest number of components to consider. The ODS GRAPHICS and ODS SELECT statements request a density plot. The output for unequal variances is shown in Figure 53.14 and Figure 53.15.
Chapter 53: The HPFMM Procedure

Figure 53.14 Model Selection for Galaxy Data Assuming Unequal Variances

HPFMM Analysis of Galaxies Data
Three to Seven Components, Unequal Variances

The HPFMM Procedure

<table>
<thead>
<tr>
<th>Model Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data Set</td>
</tr>
<tr>
<td>Response Variable</td>
</tr>
<tr>
<td>Type of Model</td>
</tr>
<tr>
<td>Distribution</td>
</tr>
<tr>
<td>Min Components</td>
</tr>
<tr>
<td>Max Components</td>
</tr>
<tr>
<td>Link Function</td>
</tr>
<tr>
<td>Estimation Method</td>
</tr>
</tbody>
</table>

Component Evaluation for Mixture Models

<table>
<thead>
<tr>
<th>Components Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>Model ID</td>
</tr>
<tr>
<td>----------</td>
</tr>
<tr>
<td>1</td>
</tr>
<tr>
<td>2</td>
</tr>
<tr>
<td>3</td>
</tr>
<tr>
<td>4</td>
</tr>
<tr>
<td>5</td>
</tr>
</tbody>
</table>

The model with 3 components (ID=1) was selected as 'best' based on the AIC statistic.

Fit Statistics

-2 Log Likelihood: 407.0
AIC (Smaller is Better): 423.0
AICC (Smaller is Better): 424.9
BIC (Smaller is Better): 442.2
Pearson Statistic: 82.0001
Effective Parameters: 8
Effective Components: 3

Parameter Estimates for Normal Model

<table>
<thead>
<tr>
<th>Component</th>
<th>Parameter</th>
<th>Estimate</th>
<th>Standard Error</th>
<th>z Value</th>
<th>Pr &gt;</th>
<th>( z )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Intercept</td>
<td>9.7101</td>
<td>0.1597</td>
<td>60.80</td>
<td>&lt;.0001</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>Intercept</td>
<td>33.0444</td>
<td>0.5322</td>
<td>62.09</td>
<td>&lt;.0001</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Intercept</td>
<td>21.4039</td>
<td>0.2597</td>
<td>82.41</td>
<td>&lt;.0001</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>Variance</td>
<td>0.1785</td>
<td>0.09542</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>Variance</td>
<td>0.8496</td>
<td>0.6937</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Variance</td>
<td>4.8567</td>
<td>0.8098</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Figure 53.14  continued

Parameter Estimates for Mixing Probabilities

| Component | Mixing Probability | GLogit(Prob) | Standard Error | z Value | Pr > |z| |
|-----------|--------------------|-------------|----------------|---------|-------|-----|
| 1         | 0.0854             | -2.3308     | 0.3959         | -5.89   | <0.001 |
| 2         | 0.0366             | -3.1781     | 0.5893         | -5.39   | <0.001 |
| 3         | 0.8781             | 0           |                |         |       |     |

Figure 53.15  Density Plot for Best (Three-Component) Model Assuming Unequal Variances
This example uses the AIC for model selection. Figure 53.16 shows the AIC and other model fit criteria for each of the fitted models.

To require that the separate components have identical variances, add the `EQUATE=SCALE` option in the `MODEL` statement:

```plaintext
  title2 "Three to Seven Components, Equal Variances";
  proc hpfmm data=galaxies criterion=AIC gconv=0;
    model v = / kmin=3 kmax=7 equate=scale;
  run;
```

The `GCONV=` convergence criterion is turned off in this PROC HPFMM run to avoid the early stoppage of the iterations when the relative gradient changes little between iterations. Turning the criterion off usually ensures that convergence is achieved with a small absolute gradient of the objective function.

The output for equal variances is shown in Figure 53.17 and Figure 53.18.
Looking for Multiple Modes: Are Galaxies Clustered?

Figure 53.17 Model Selection for Galaxy Data Assuming Equal Variances

**HPFMM Analysis of Galaxies Data**
Three to Seven Components, Equal Variances

The HPFMM Procedure

**Model Information**

- **Data Set**: WORK.GALAXIES
- **Response Variable**: v
- **Type of Model**: Homogeneous Mixture
- **Distribution**: Normal
- **Min Components**: 3
- **Max Components**: 7
- **Link Function**: Identity
- **Estimation Method**: Maximum Likelihood

**Component Evaluation for Mixture Models**

<table>
<thead>
<tr>
<th>Model ID</th>
<th>Total</th>
<th>Eff. Total</th>
<th>Eff. -2 Log L</th>
<th>AIC</th>
<th>AICC</th>
<th>BIC</th>
<th>Pearson</th>
<th>Max Gradient</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>3</td>
<td>3</td>
<td>6</td>
<td>6</td>
<td>478.74</td>
<td>490.74</td>
<td>491.86</td>
<td>505.18</td>
</tr>
<tr>
<td>2</td>
<td>4</td>
<td>4</td>
<td>8</td>
<td>8</td>
<td>416.49</td>
<td>432.49</td>
<td>434.47</td>
<td>451.75</td>
</tr>
<tr>
<td>3</td>
<td>5</td>
<td>5</td>
<td>10</td>
<td>10</td>
<td>416.49</td>
<td>436.49</td>
<td>439.59</td>
<td>460.56</td>
</tr>
<tr>
<td>4</td>
<td>6</td>
<td>6</td>
<td>12</td>
<td>12</td>
<td>416.49</td>
<td>440.49</td>
<td>445.02</td>
<td>469.37</td>
</tr>
<tr>
<td>5</td>
<td>7</td>
<td>7</td>
<td>14</td>
<td>14</td>
<td>416.49</td>
<td>444.49</td>
<td>450.76</td>
<td>478.19</td>
</tr>
</tbody>
</table>

The model with 4 components (ID=2) was selected as 'best' based on the AIC statistic.

**Fit Statistics**

- **-2 Log Likelihood**: 416.5
- **AIC (Smaller is Better)**: 432.5
- **AICC (Smaller is Better)**: 434.5
- **BIC (Smaller is Better)**: 451.7
- **Pearson Statistic**: 82.0000
- **Effective Parameters**: 8
- **Effective Components**: 4

**Parameter Estimates for Normal Model**

<table>
<thead>
<tr>
<th>Component</th>
<th>Parameter</th>
<th>Estimate</th>
<th>Standard Error</th>
<th>z Value</th>
<th>Pr &gt;</th>
<th>Pr &gt; [z]</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Intercept</td>
<td>23.5058</td>
<td>0.3460</td>
<td>67.93</td>
<td>&lt;.0001</td>
<td>&lt;.0001</td>
</tr>
<tr>
<td>2</td>
<td>Intercept</td>
<td>33.0440</td>
<td>0.7610</td>
<td>43.42</td>
<td>&lt;.0001</td>
<td>&lt;.0001</td>
</tr>
<tr>
<td>3</td>
<td>Intercept</td>
<td>20.0086</td>
<td>0.3029</td>
<td>66.06</td>
<td>&lt;.0001</td>
<td>&lt;.0001</td>
</tr>
<tr>
<td>4</td>
<td>Intercept</td>
<td>9.7103</td>
<td>0.4981</td>
<td>19.50</td>
<td>&lt;.0001</td>
<td>&lt;.0001</td>
</tr>
<tr>
<td>1</td>
<td>Variance</td>
<td>1.7354</td>
<td>0.3905</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>Variance</td>
<td>1.7354</td>
<td>0.3905</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Variance</td>
<td>1.7354</td>
<td>0.3905</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Variance</td>
<td>1.7354</td>
<td>0.3905</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Not surprisingly, the two variance specifications produce different optimal models. The unequal variance specification favors a three-component model while the equal variance specification favors a four-component model. Comparison of the AIC fit statistics, 423.0 and 432.5, indicates that the three-component, unequal variance model provides the best overall fit.
Comparison with Roeder’s Method

It is important to note that Roeder’s original analysis proceeds in a different manner than the finite mixture modeling presented here. The technique presented by Roeder first develops a “best” range of scale parameters based on a specific criterion. Roeder then uses fixed scale parameters taken from this range to develop optimal equal-scale Gaussian mixture models.

You can reproduce Roeder’s point estimate for the density by specifying a five-component Gaussian mixture. In addition, use the `EQUATE=SCALE` option in the `MODEL` statement and a `RESTRICT` statement fixing the first component’s scale parameter at 0.9025 (Roeder’s $h = 0.95$, scale $= h^2$). The combination of these options produces a mixture of five Gaussian components, each with variance 0.9025. The following statements conduct this analysis:

```
title2 "Five Components, Equal Variances = 0.9025";
proc hpfmm data=galaxies;
    model v = / k=5 equate=scale;
    restrict int 0 (scale 1) = 0.9025;
run;
ods graphics off;
```

The output is shown in Figure 53.19 and Figure 53.20.

**Figure 53.19** Reproduction of Roeder’s Five-Component Analysis of Galaxy Data

**HPFMM Analysis of Galaxies Data**

**Five Components, Equal Variances = 0.9025**

**The HPFMM Procedure**

<table>
<thead>
<tr>
<th>Model Information</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Data Set</td>
<td>WORK.GALAXIES</td>
</tr>
<tr>
<td>Response Variable</td>
<td>v</td>
</tr>
<tr>
<td>Type of Model</td>
<td>Homogeneous Mixture</td>
</tr>
<tr>
<td>Distribution</td>
<td>Normal</td>
</tr>
<tr>
<td>Components</td>
<td>5</td>
</tr>
<tr>
<td>Link Function</td>
<td>Identity</td>
</tr>
<tr>
<td>Estimation Method</td>
<td>Maximum Likelihood</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Fit Statistics</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>-2 Log Likelihood</td>
<td>412.2</td>
</tr>
<tr>
<td>AIC (Smaller is Better)</td>
<td>430.2</td>
</tr>
<tr>
<td>AICC (Smaller is Better)</td>
<td>432.7</td>
</tr>
<tr>
<td>BIC (Smaller is Better)</td>
<td>451.9</td>
</tr>
<tr>
<td>Pearson Statistic</td>
<td>82.5549</td>
</tr>
<tr>
<td>Effective Parameters</td>
<td>9</td>
</tr>
<tr>
<td>Effective Components</td>
<td>5</td>
</tr>
</tbody>
</table>

**Linear Constraints at Solution**

<table>
<thead>
<tr>
<th>Constraint</th>
<th>Active</th>
</tr>
</thead>
<tbody>
<tr>
<td>k = 1</td>
<td></td>
</tr>
<tr>
<td>Variance</td>
<td>0.90</td>
</tr>
</tbody>
</table>
**Figure 53.19** continued

### Parameter Estimates for Normal Model

| Component | Parameter | Estimate | Standard Error | z Value | Pr > |z| |
|-----------|-----------|----------|----------------|---------|------|---|
| 1         | Intercept | 26.3266  | 0.7778         | 33.85   | <.001|   |
| 2         | Intercept | 33.0443  | 0.5485         | 60.25   | <.001|   |
| 3         | Intercept | 9.7101   | 0.3591         | 27.04   | <.001|   |
| 4         | Intercept | 23.0295  | 0.2294         | 100.38  | <.001|   |
| 5         | Intercept | 19.7187  | 0.1784         | 110.55  | <.001|   |
| 1         | Variance  | 0.9025   | 0              |         |      |   |
| 2         | Variance  | 0.9025   | 0              |         |      |   |
| 3         | Variance  | 0.9025   | 0              |         |      |   |
| 4         | Variance  | 0.9025   | 0              |         |      |   |
| 5         | Variance  | 0.9025   | 0              |         |      |   |

### Parameter Estimates for Mixing Probabilities

| Component | Mixing Probability | GLogit(Prob) | Standard Error | z Value | Pr > |z| |
|-----------|--------------------|-------------|----------------|---------|------|---|
| 1         | 0.0397             | -2.4739     | 0.7084         | -3.49   | 0.0005|   |
| 2         | 0.0366             | -2.5544     | 0.6016         | -4.25   | <.001|   |
| 3         | 0.0854             | -1.7071     | 0.4141         | -4.12   | <.001|   |
| 4         | 0.3678             | -0.2466     | 0.2699         | -0.91   | 0.3609|   |
| 5         | 0.4706             | 0           |                |         |      |   |
Figure 53.20 Density Plot for Roeder’s Analysis
Syntax: HPFMM Procedure

The following statements are available in the HPFMM procedure:

```
PROC HPFMM < options > ;
  BAYES bayes-options ;
  BY variables ;
  CLASS variables ;
  FREQ variable ;
  ID variables ;
  MODEL response< (response-options) >= < effects > < / model-options > ;
  MODEL events/trials = < effects > < / model-options > ;
  MODEL + < effects > < / model-options > ;
  OUTPUT < OUT=SAS-data-set >
     < keyword < (keyword-options) > = < name > . . .
     < keyword < (keyword-options) > = < name > < / options > > ;
  PERFORMANCE performance-options ;
  PROBMODEL < effects > < / probmodel-options > ;
  RESTRICT < 'label' > constraint-specification < , . . . , constraint-specification >
     < operator < value > > < / option > ;
  WEIGHT variable ;
```

The PROC HPFMM statement and at least one MODEL statement is required. The CLASS, RESTRICT and MODEL statements can appear multiple times. If a CLASS statement is specified, it must precede the MODEL statements. The RESTRICT statements must appear after the MODEL statements.

PROC HPFMM Statement

```
PROC HPFMM < options > ;
```

The PROC HPFMM statement invokes the HPFMM procedure. Table 53.2 summarizes the options available in the PROC HPFMM statement. These and other options in the PROC HPFMM statement are then described fully in alphabetical order.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Basic Options</td>
<td></td>
</tr>
<tr>
<td>DATA=</td>
<td>Specifies the input data set</td>
</tr>
<tr>
<td>EXCLUSION=</td>
<td>Specifies how the procedure responds to support violations in the data</td>
</tr>
<tr>
<td>NAMELEN=</td>
<td>Specifies the length of effect names</td>
</tr>
<tr>
<td>SEED=</td>
<td>Specifies the random number seed for analyses that require random number draws</td>
</tr>
<tr>
<td>Option</td>
<td>Description</td>
</tr>
<tr>
<td>----------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td><strong>Displayed Output</strong></td>
<td></td>
</tr>
<tr>
<td>COMPONENTINFO</td>
<td>Displays information about the mixture components</td>
</tr>
<tr>
<td>CORR</td>
<td>Displays the asymptotic correlation matrix of the maximum likelihood parameter estimates or the empirical correlation matrix of the Bayesian posterior estimates</td>
</tr>
<tr>
<td>COV</td>
<td>Displays the asymptotic covariance matrix of the maximum likelihood parameter estimates or the empirical covariance matrix of the Bayesian posterior estimates</td>
</tr>
<tr>
<td>COVI</td>
<td>Displays the inverse of the covariance matrix of the parameter estimates</td>
</tr>
<tr>
<td>FITDETAILS</td>
<td>Displays fit information for all examined models</td>
</tr>
<tr>
<td>ITDETAILS</td>
<td>Adds estimates and gradients to the “Iteration History” table</td>
</tr>
<tr>
<td>NOCLPRINT</td>
<td>Suppresses the “Class Level Information” table completely or partially</td>
</tr>
<tr>
<td>NOITPRINT</td>
<td>Suppresses the “Iteration History Information” table</td>
</tr>
<tr>
<td>NOPRINT</td>
<td>Suppresses tabular and graphical output</td>
</tr>
<tr>
<td>PARMSTYLE=</td>
<td>Specifies how parameters are displayed in ODS tables</td>
</tr>
<tr>
<td>PLOTS</td>
<td>Produces ODS statistical graphics</td>
</tr>
<tr>
<td><strong>Computational Options</strong></td>
<td></td>
</tr>
<tr>
<td>CRITERION=</td>
<td>Specifies the criterion used in model selection</td>
</tr>
<tr>
<td>NOCENTER</td>
<td>Prevents centering and scaling of the regressor variables</td>
</tr>
<tr>
<td>PARTIAL=</td>
<td>Specifies a variable that defines a partial classification</td>
</tr>
<tr>
<td><strong>Options Related to Optimization</strong></td>
<td></td>
</tr>
<tr>
<td>ABSCONV=</td>
<td>Tunes an absolute function convergence criterion</td>
</tr>
<tr>
<td>ABSFCONV=</td>
<td>Tunes an absolute function difference convergence criterion</td>
</tr>
<tr>
<td>ABSGCONV=</td>
<td>Tunes the absolute gradient convergence criterion</td>
</tr>
<tr>
<td>FCONV=</td>
<td>Specifies a relative function convergence criterion that is based on a relative change of the function value</td>
</tr>
<tr>
<td>FCONV2=</td>
<td>Specifies a relative function convergence criterion that is based on a predicted reduction of the objective function</td>
</tr>
<tr>
<td>GCONV=</td>
<td>Tunes the relative gradient convergence criterion</td>
</tr>
<tr>
<td>MAXITER=</td>
<td>Specifies the maximum number of iterations in any optimization</td>
</tr>
<tr>
<td>MAXFUNC=</td>
<td>Specifies the maximum number of function evaluations in any optimization</td>
</tr>
<tr>
<td>MAXTIME=</td>
<td>Specifies the upper limit of CPU time in seconds for any optimization</td>
</tr>
<tr>
<td>MINITER=</td>
<td>Specifies the minimum number of iterations in any optimization</td>
</tr>
<tr>
<td>TECHNIQUE=</td>
<td>Selects the optimization technique</td>
</tr>
</tbody>
</table>
Table 53.2  continued

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SINGULAR</td>
<td>Tunes general singularity criterion</td>
</tr>
<tr>
<td>SINGLES</td>
<td>Tunes singularity for the residual variance</td>
</tr>
<tr>
<td>SINGCHOL</td>
<td>Tunes singularity for Cholesky decompositions</td>
</tr>
<tr>
<td>INVALIDLOGL</td>
<td>Tunes the value assigned to an invalid component log likelihood</td>
</tr>
</tbody>
</table>

You can specify the following options in the PROC HPFMM statement.

**ABSCONV=r**

**ABSTOL=r**

specifies an absolute function convergence criterion. For minimization, the termination criterion is $f(\boldsymbol{\psi}^{(k)}) \leq r$, where $\boldsymbol{\psi}$ is the vector of parameters in the optimization and $f(\cdot)$ is the objective function. The default value of $r$ is the negative square root of the largest double-precision value, which serves only as a protection against overflows.

**ABSFCONV=r**

**ABSFOTOL=r**

specifies an absolute function difference convergence criterion. For all techniques except NMSIMP, the termination criterion is a small change of the function value in successive iterations:

$$|f(\boldsymbol{\psi}^{(k-1)}) - f(\boldsymbol{\psi}^{(k)})| \leq r$$

Here, $\boldsymbol{\psi}$ denotes the vector of parameters that participate in the optimization, and $f(\cdot)$ is the objective function. The same formula is used for the NMSIMP technique, but $\boldsymbol{\psi}^{(k)}$ is defined as the vertex with the lowest function value, and $\boldsymbol{\psi}^{(k-1)}$ is defined as the vertex with the highest function value in the simplex. The default value is $r=0$.

**ABSGCONV=r**

**ABSGTOL=r**

specifies an absolute gradient convergence criterion. The termination criterion is a small maximum absolute gradient element:

$$\max_j |g_j(\boldsymbol{\psi}^{(k)})| \leq r$$

Here, $\boldsymbol{\psi}$ denotes the vector of parameters that participate in the optimization, and $g_j(\cdot)$ is the gradient of the objective function with respect to the $j$th parameter. This criterion is not used by the NMSIMP technique. The default value is $r=1E-5$.

**COMPONENTINFO**

**COMPINFO**

**CINFO**

produces a table with additional details about the fitted model components.
**PROC HPFMM Statement**

- **COV**
  produces the covariance matrix of the parameter estimates. For maximum likelihood estimation, this matrix is based on the inverse (projected) Hessian matrix. For Bayesian estimation, it is the empirical covariance matrix of the posterior estimates. The covariance matrix is shown for all parameters, even if they did not participate in the optimization or sampling.

- **COVI**
  produces the inverse of the covariance matrix of the parameter estimates. For maximum likelihood estimation, the covariance matrix is based on the inverse (projected) Hessian matrix. For Bayesian estimation, it is the empirical covariance matrix of the posterior estimates. This matrix is then inverted by sweeping, and rows and columns that correspond to linear dependencies or singularities are zeroed.

- **CORR**
  produces the correlation matrix of the parameter estimates. For maximum likelihood estimation this matrix is based on the inverse (projected) Hessian matrix. For Bayesian estimation, it is based on the empirical covariance matrix of the posterior estimates.

- **CRITERION=keyword**
  **CRIT=keyword**
  specifies the criterion by which the HPFMM procedure ranks models when multiple models are evaluated during maximum likelihood estimation. You can choose from the following keywords to rank models:

  - **AIC** based on Akaike’s information criterion
  - **AICC** based on the bias-corrected AIC criterion
  - **BIC** based on the Bayesian information criterion
  - **GRADIENT** based on the largest element of the gradient (in absolute value)
  - **LOGL | LL** based on the mixture log likelihood
  - **PEARSON** based on the Pearson statistic

  The default is CRITERION=BIC.

- **DATA=SAS-data-set**
  names the SAS data set to be used by PROC HPFMM. The default is the most recently created data set.

- **EXCLUSION=NONE | ANY | ALL**
  **EXCLUDE=NONE | ANY | ALL**
  specifies how the HPFMM procedure handles support violations of observations. For example, in a mixture of two Poisson variables, negative response values are not possible. However, in a mixture of a Poisson and a normal variable, negative values are possible, and their likelihood contribution to the Poisson component is zero. An observation that violates the support of one component distribution of the model might be a valid response with respect to one or more other component distributions. This requires some nuanced handling of support violations in mixture models.

  The default exclusion technique, EXCLUSION=ALL, removes an observation from the analysis only if it violates the support of all component distributions. The other extreme, EXCLUSION=NONE, permits an observation into the analysis regardless of support violations. EXCLUSION=ANY removes observations from the analysis if the response violates the support of any component distributions. In the single-component case, EXCLUSION=ALL and EXCLUSION=ANY are identical.
**FCONV=**

FCONV= specifies a relative function convergence criterion that is based on the relative change of the function value. For all techniques except NMSIMP, PROC HPFMM terminates when there is a small relative change of the function value in successive iterations:

\[
\frac{|f(\psi^{(k)}) - f(\psi^{(k-1)})|}{|f(\psi^{(k-1)})|} \leq r
\]

Here, \(\psi\) denotes the vector of parameters that participate in the optimization, and \(f(\cdot)\) is the objective function. The same formula is used for the NMSIMP technique, but \(\psi^{(k)}\) is defined as the vertex with the lowest function value, and \(\psi^{(k-1)}\) is defined as the vertex with the highest function value in the simplex.

The default is \(r = 10^{-\text{FDIGITS}}\), where FDIGITS is by default \(-\log_{10}(\epsilon)\), and \(\epsilon\) is the machine precision.

**FTOL=**

FTOL= specifies a relative function convergence criterion that is based on the predicted reduction of the objective function. For all techniques except NMSIMP, the termination criterion is a small predicted reduction

\[
df^{(k)} \approx f(\theta^{(k)}) - f(\theta^{(k)} + s^{(k)})
\]

of the objective function. The predicted reduction

\[
df^{(k)} = -g^{(k)} s^{(k)} - \frac{1}{2} s^{(k)} H^{(k)} s^{(k)}
\]

\[
= -\frac{1}{2} s^{(k)} g^{(k)}
\]

\[
\leq r
\]

is computed by approximating the objective function \(f\) by the first two terms of the Taylor series and substituting the Newton step:

\[
s^{(k)} = -[H^{(k)}]^{-1} g^{(k)}
\]

For the NMSIMP technique, the termination criterion is a small standard deviation of the function values of the \(n + 1\) simplex vertices \(\theta_l^{(k)}\), \(l = 0, \ldots, n,\)

\[
\sqrt{\frac{1}{n+1} \sum_l \left[ f(\theta_l^{(k)}) - \bar{f}(\theta^{(k)}) \right]^2} \leq r
\]

where \(\bar{f}(\theta^{(k)}) = \frac{1}{n+1} \sum_l f(\theta_l^{(k)})\). If there are \(n_{\text{act}}\) boundary constraints active at \(\theta^{(k)}\), the mean and standard deviation are computed only for the \(n + 1 - n_{\text{act}}\) unconstrained vertices.

The default value is \(r = 1\times10^{-6}\) for the NMSIMP technique and \(r = 0\) otherwise.
FITDETAILS requests that the “Optimization Information,” “Iteration History,” and “Fit Statistics” tables be produced for all optimizations when models with different number of components are evaluated. For example, the following statements fit a binomial regression model with up to three components and produces fit and optimization information for all three:

```latex
proc hpfmm fitdetails;
    model y/n = x / kmax=3;
run;
```

Without the FITDETAILS option, only the “Fit Statistics” table for the selected model is displayed.

In Bayesian estimation, the FITDETAILS option displays the following tables for each model that the procedure fits: “Bayes Information,” “Iteration History,” “Prior Information,” “Fit Statistics,” “Posterior Summaries,” “Posterior Intervals,” and any requested diagnostics tables. The “Iteration History” table appears only if the BAYES statement includes the INITIAL=MLE option.

Without the FITDETAILS option, these tables are listed only for the selected model.

GCONV=r
GTOL=r

specifies a relative gradient convergence criterion. For all techniques except CONGRA and NMSIMP, the termination criterion is a small normalized predicted function reduction:

\[
g(\psi^{(k)}) / |f(\psi^{(k)})| \leq r
\]

Here, \(\psi\) denotes the vector of parameters that participate in the optimization, \(f(\cdot)\) is the objective function, and \(g(\cdot)\) is the gradient. For the CONGRA technique (where a reliable Hessian estimate \(H\) is not available), the following criterion is used:

\[
\frac{\|g(\psi^{(k)})\|_2^2 + s(\psi^{(k)})^2}{\|g(\psi^{(k)}) - g(\psi^{(k-1)})\|_2 |f(\psi^{(k)})|} \leq r
\]

This criterion is not used by the NMSIMP technique. The default value is \(r=1E^{-8}\).

HESSIAN displays the Hessian matrix of the model. This option is not available for Bayesian estimation.

INVALIDLOGL=r

specifies the value assumed by the HPFMM procedure if a log likelihood cannot be computed (for example, because the value of the response variable falls outside of the response distribution’s support). The default value is \(-1E20\).

ITDETAILS adds parameter estimates and gradients to the “Iteration History” table. If the HPFMM procedure centers or scales the model variables (or both), the parameter estimates and gradients reported during the iteration refer to that scale. You can suppress centering and scaling with the NOCENTER option.
**MAXFUNC=**\( n \)  
**MAXFU=**\( n \)

specifies the maximum number of function calls in the optimization process. The default values are as follows, depending on the optimization technique:

- TRUREG, NRRIDG, and NEWRAP: 125
- QUANEW and DBLDOG: 500
- CONGRA: 1000
- NMSIMP: 3000

The optimization can terminate only after completing a full iteration. Therefore, the number of function calls that are actually performed can exceed the number that is specified by the MAXFUNC= option. You can choose the optimization technique with the TECHNIQUE= option.

**MAXITER=**\( n \)  
**MAXIT=**\( n \)

specifies the maximum number of iterations in the optimization process. The default values are as follows, depending on the optimization technique:

- TRUREG, NRRIDG, and NEWRAP: 50
- QUANEW and DBLDOG: 200
- CONGRA: 400
- NMSIMP: 1000

These default values also apply when \( n \) is specified as a missing value. You can choose the optimization technique with the TECHNIQUE= option.

**MAXTIME=**\( r \)

specifies an upper limit of \( r \) seconds of CPU time for the optimization process. The time is checked only at the end of each iteration. Therefore, the actual run time might be longer than the specified time. By default, CPU time is not limited.

**MINITER=**\( n \)  
**MINIT=**\( n \)

specifies the minimum number of iterations. The default value is 0. If you request more iterations than are actually needed for convergence to a stationary point, the optimization algorithms can behave strangely. For example, the effect of rounding errors can prevent the algorithm from continuing for the required number of iterations.

**NAMELEN=**\( \text{number} \)

specifies the length to which long effect names are shortened. The default and minimum value is 20.

**NOCENTER**

requests that regressor variables not be centered or scaled. By default the HPFMM procedure centers and scales columns of the \( X \) matrix if the models contain intercepts. If NOINT options in MODEL statements are in effect, the columns of \( X \) are scaled but not centered. Centering and scaling can help with the stability of estimation and sampling algorithms. The HPFMM procedure does not produce a table of the centered and scaled coefficients and provides no user control over the type of centering and scaling that is applied. The NOCENTER option turns any centering and scaling off and processes the raw values of the continuous variables.
NOCLPRINT<=number>
suppresses the display of the “Class Level Information” table if you do not specify number. If you specify number, the values of the classification variables are displayed for only those variables whose number of levels is less than number. Specifying a number helps to reduce the size of the “Class Level Information” table if some classification variables have a large number of levels.

NOITPRINT
suppresses the display of the “Iteration History Information” table.

NOPRINT
suppresses the normal display of tabular and graphical results. The NOPRINT option is useful when you want to create only one or more output data sets with the procedure. This option temporarily disables the Output Delivery System (ODS); see Chapter 20, “Using the Output Delivery System,” for more information.

PARMSTYLE=EFFECT | LABEL
specifies the display style for parameters and effects. The HPFMM procedure can display parameters in two styles:

- The EFFECT style (which is used by the MIXED and GLIMMIX procedure, for example) identifies a parameter with an “Effect” column and adds separate columns for the CLASS variables in the model.
- The LABEL style creates one column, named Parameter, that combines the relevant information about a parameter into a single column. If your model contains multiple CLASS variables, the LABEL style might use space more economically.

The EFFECT style is the default for models that contain effects; otherwise the LABEL style is used (for example, in homogeneous mixtures). You can change the display style with the PARMSTYLE= option. Regardless of the display style, ODS output data sets that contain information about parameter estimates contain columns for both styles.

PARTIAL=variable
MEMBERSHIP=variable
specifies a variable in the input data set that identifies component membership. You can specify missing values for observations whose component membership is undetermined; this is known as a partial classification (McLachlan and Peel 2000, p. 75). For observations with known membership, the likelihood contribution is no longer a mixture. If observation $i$ is known to be a member of component $m$, then its log likelihood contribution is

$$\log \left\{ \pi_m(z, \alpha_m) \ p_m(y; x'_m \beta_m, \phi_m) \right\}$$

Otherwise, if membership is undetermined, it is

$$\log \left\{ \sum_{j=1}^{k} \pi_j(z, \alpha_j) \ p_j(y; x'_j \beta_j, \phi_j) \right\}$$

The variable specified in the PARTIAL= option can be numeric or character. In case of a character variable, the variable must appear in the CLASS statement. If the PARTIAL= variable appears in the
**CLASS** statement, the membership assignment is made based on the levelized values of the variable, as shown in the “Class Level Information” table. Invalid values of the PARTIAL= variable are ignored.

In a model in which label switching is a problem, the switching can sometimes be avoided by assigning just a few observations to categories. For example, in a three-component model, switches might be prevented by assigning the observation with the smallest response value to the first component and the observation with the largest response value to the last component.

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

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

controls the plots produced through ODS Graphics.

ODS Graphics must be enabled before plots can be requested. For example:

```plaintext
ods graphics on;
proc hpfmm data=yeast seed=12345;
   model count/n = / k=2;
   freq f;
   performance nthreads=2;
   bayes;
run;
ods graphics off;
```

For more information about enabling and disabling ODS Graphics, see the section “Enabling and Disabling ODS Graphics” on page 615 in Chapter 21, “Statistical Graphics Using ODS.”

**Global Plot Options**

The **global-plot-options** apply to all relevant plots generated by the HPFMM procedure. The **global-plot-options** supported by the HPFMM procedure are as follows:

**UNPACKPANEL**

**UNPACK**

displays each graph separately. (By default, some graphs can appear together in a single panel.)

**ONLY**

produces only the specified plots. This option is useful if you do not want the procedure to generate all default graphics, but only the ones specified.
Specific Plot Options

The following listing describes the specific plots and their options.

ALL
requests that all plots appropriate for the analysis be produced.

NONE
requests that no ODS graphics be produced.

DENSITY < (density-options)>
requests a plot of the data histogram and mixture density function. This graphic is a default graphic in models without effects in the MODEL statements and is available only in these models. Furthermore, all distributions involved in the mixture must be continuous. You can specify the following density-options to modify the plot:

CUMULATIVE
CDF
displays the histogram and densities in cumulative form.

NBINS=n
BINS=n
specifies the number of bins in the histogram; \( n \) is greater than or equal to 0. By default, the HPFMM procedure computes a suitable bin width and number of bins, based on the range of the response and the number of usable observations. The option has no effect for binary data.

NOCOMPONENTS
NOCOMP
suppresses the component densities from the plot. If the component densities are displayed, they are scaled so that their sum equals the mixture density at any point on the graph. In single-component models, this option has no effect.

NODENSITY
NODENS
suppresses the computation of the mixture density (and the component densities if the COMPONENTS suboption is specified). If you specify the NOHISTOGRAM and the NODENSITY option, no graphic is produced.

NOLABEL
suppresses the component identification with labels. By default, the HPFMM procedure labels component densities in the legend of the plot. If you do not specify a model label with the LABEL= option in the MODEL statement, an identifying label is constructed from the parameter estimates that are associated with the component. In this case the parameter values are not necessarily the mean and variance of the distribution; the values used to identify the densities on the plot are chosen to simplify linking between graphical and tabular results.
NOHISTOGRAM

NOHIST
suppresses the computation of the histogram of the raw values. If you specify the NOHISTOGRAM and the NODENSITY option, no graphic is produced.

NPOINTS=n
N=n
specifies the number of values used to compute the density functions; \( n \) is greater than or equal to 0. The default is \( N=200 \).

WIDTH=value
BINWIDTH=value
specifies the bin width for the histogram. The \( value \) is specified in units of the response variable and must be positive. The option has no effect for binary data.

TRACE < (tadpanel-options)>
requests a trace panel with posterior diagnostics for a Bayesian analysis. If a BAYES statement is present, the trace panel plots are generated by default, one for each sampled parameter. You can specify the following tadpanel-options to modify the graphic:

BOX
BOXPLOT
replaces the autocorrelation plot with a box plot of the posterior sample.

SMOOTH=None | Mean | Spline
adds a reference estimate to the trace plot. By default, SMOOTH=None. SMOOTH=MEAN uses the arithmetic mean of the trace as the reference. SMOOTH=SPLINE adds a penalized B-spline.

REFERENCE= reference-style
adds vertical reference lines to the density plot, trace plot, and box plot. The available options for the reference-style are:

NONE
suppresses the reference lines

EQT
requests equal-tail intervals

HPD
requests intervals of highest posterior density. The level for the credible or HPD intervals is chosen based on the “Posterior Interval Statistics” table.

PERCENTILES (or PERC) for percentiles. Up to three percentiles can be displayed, as based on the “Posterior Summary Statistics” table.

The default is REFERENCE=EQT.

UNPACK
unpacks the panel graphic and displays its elements as separate plots.
CRITERIONPANEL <(critpanel-options)> 
requests a plot for comparing the model fit criteria for different numbers of components. This 
plot is available only if you also specify the KMAX option in at least one MODEL statement. The plot includes different criteria, depending on whether you are using maximum likelihood or 
Bayesian estimation. You can specify the following critpanel-option to modify the plot:

UNPACK
unpacks the panel plot and displays its elements as separate plots, one for each fit criterion.

SEED=n  
determines the random number seed for analyses that depend on a random number stream. If you 
do not specify a seed or if you specify a value less than or equal to zero, the seed is generated from 
reading the time of day from the computer clock. The largest possible value for the seed is $2^{31} - 1$. The seed value is reported in the “Model Information” table.

You can use the SYSRAN and SYSRANEND macro variables after a PROC HPFMM run to 
query the initial and final seed values. However, using the final seed value as the starting seed for a 
subsequent analysis does not continue the random number stream where the previous analysis left off. The SYSRANEND macro variable provides a mechanism to pass on seed values to ensure that the sequence of random numbers is the same every time you run an entire program.

Analyses that use the same (nonzero) seed are not completely reproducible if they are executed with a 
different number of threads since the random number streams in separate threads are independent. You 
can control the number of threads used by the HPFMM procedure with system options or through the 
PERFORMANCE statement in the HPFMM procedure.

SINGCHOL=number  
tunes the singularity criterion in Cholesky decompositions. The default is $1E4$ times the machine 
epsilon; this product is approximately $1E-12$ on most computers.

SINGRES=number  
sets the tolerance for which the residual variance or scale parameter is considered to be zero. The 
default is $1E4$ times the machine epsilon; this product is approximately $1E-12$ on most computers.

SINGULAR=number  
tunes the general singularity criterion applied by the HPFMM procedure in sweeps and inversions. The 
default is $1E4$ times the machine epsilon; this product is approximately $1E-12$ on most computers.

TECHNIQUE=keyword
TECH=keyword  
specifies the optimization technique to obtain maximum likelihood estimates. You can choose from the 
following techniques by specifying the appropriate keyword:

CONGRA performs a conjugate-gradient optimization.
DBLDOG performs a version of double-dogleg optimization.
NEWWRAP performs a Newton-Raphson optimization combining a line-search algorithm with 
ridging.
NMSIMP performs a Nelder-Mead simplex optimization.
NONE performs no optimization.
NRRIDG performs a Newton-Raphson optimization with ridging.
QUANEW performs a dual quasi-Newton optimization.
TRUREG performs a trust-region optimization.

The default is TECH=QUANEW.

For more details about these optimization methods, see the section “Choosing an Optimization Algorithm” on page 4261.

**ZERO PROB=** *number*

tunes the threshold (a value between 0 and 1) below which the HPFMM procedure considers a component mixing probability to be zero. This affects the calculation of the number of effective components. The default is the square root of the machine epsilon; this is approximately 1E–8 on most computers.

---

**BAYES Statement**

```
BAYES bayes-options ;
```

The BAYES statement requests that the parameters of the model be estimated by Markov chain Monte Carlo sampling techniques. The HPFMM procedure can estimate by maximum likelihood the parameters of all models supported by the procedure. Bayes estimation, on the other hand, is available for only a subset of these models.

In Bayesian analysis, it is essential to examine the convergence of the Markov chains before you proceed with posterior inference. With ODS Graphics turned on, the HPFMM procedure produces graphs at the end of the procedure output; these graphs enable you to visually examine the convergence of the chain. Inferences cannot be made if the Markov chain has not converged.

The output produced for a Bayesian analysis is markedly different from that for a frequentist (maximum likelihood) analysis for the following reasons:

- Parameter estimates do not have the same interpretation in the two analyses. Parameters are fixed unknown constants in the frequentist context and random variables in a Bayesian analysis.
- The results of a Bayesian analysis are summarized through chain diagnostics and posterior summary statistics and intervals.
- The HPFMM procedure samples the mixing probabilities in Bayesian models directly, rather than mapping them onto a logistic (or other) scale.

The HPFMM procedure applies highly specialized sampling algorithms in Bayesian models. For single-component models without effects, a conjugate sampling algorithm is used where possible. For models in the exponential family that contain effects, the sampling algorithm is based on Gamerman (1997). For the normal and $t$ distributions, a conjugate sampler is the default sampling algorithm for models with and without effects. In multi-component models, the sampling algorithm is based on latent variable sampling through data augmentation (Frühwirth-Schnatter 2006) and the Gamerman or conjugate sampler. Because of this specialization, the options for controlling the prior distributions of the parameters are limited.
Table 53.3 summarizes the bayes-options available in the BAYES statement. The full assortment of options is then described in alphabetical order.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Options Related to Sampling</strong></td>
<td></td>
</tr>
<tr>
<td>INITIAL=</td>
<td>Specifies how to construct initial values</td>
</tr>
<tr>
<td>NBI=</td>
<td>Specifies the number of burn-in samples</td>
</tr>
<tr>
<td>NMC=</td>
<td>Specifies the number of samples after burn-in</td>
</tr>
<tr>
<td>METROPOLIS</td>
<td>Forces a Metropolis-Hastings sampling algorithm even if conjugate sampling is possible</td>
</tr>
<tr>
<td>OUTPOST=</td>
<td>Generates a data set that contains the posterior estimates</td>
</tr>
<tr>
<td>THIN=</td>
<td>Controls the thinning of the Markov chain</td>
</tr>
<tr>
<td><strong>Specification of Prior Information</strong></td>
<td></td>
</tr>
<tr>
<td>MIXPRIORPARMS</td>
<td>Specifies the prior parameters for the Dirichlet distribution of the mixing probabilities</td>
</tr>
<tr>
<td>BETAPRIORPARMS=</td>
<td>Specifies the parameters of the normal prior distribution for individual parameters in the ( \beta ) vector</td>
</tr>
<tr>
<td>MUPRIORPARMS=</td>
<td>Specifies the parameters of the prior distribution for the means in homogeneous mixtures without effects</td>
</tr>
<tr>
<td>PHIPRIORPARMS=</td>
<td>Specifies the parameters of the inverse gamma prior distribution for the scale parameters in homogeneous mixtures</td>
</tr>
<tr>
<td>PRIOROPTIONS</td>
<td>Specifies additional options used in the determination of the prior distribution</td>
</tr>
<tr>
<td><strong>Posterior Summary Statistics and Convergence Diagnostics</strong></td>
<td></td>
</tr>
<tr>
<td>DIAGNOSTICS=</td>
<td>Displays convergence diagnostics for the Markov chain</td>
</tr>
<tr>
<td>STATISTICS</td>
<td>Displays posterior summary information for the Markov chain</td>
</tr>
<tr>
<td><strong>Other Options</strong></td>
<td></td>
</tr>
<tr>
<td>ESTIMATE=</td>
<td>Specifies which estimate is used for the computation of OUTPUT statistics and graphics</td>
</tr>
<tr>
<td>TIMEINC=</td>
<td>Specifies the time interval to report on sampling progress (in seconds)</td>
</tr>
</tbody>
</table>

You can specify the following bayes-options in the BAYES statement.

**BETAPRIORPARMS=pair-specification**

**BETAPRIORPARMS(pair-specification ... pair-specification)**

specifies the parameters for the normal prior distribution of the parameters that are associated with model effects \( (\beta) \). The *pair-specification* is of the form \( (a, b) \), and the values \( a \) and \( b \) are the mean and variance of the normal distribution, respectively. This option overrides the PRIOROPTIONS option.

The form of the BETAPRIORPARMS with an equal sign and a single pair is used to specify one pair of prior parameters that applies to all components in the mixture. In the following example, the two intercepts and the two regression coefficients all have a \( N(0, 100) \) prior distribution:
Chapter 53: The HPFMM Procedure

```
proc hpfmm;
  model y = x / k=2;
    bayes betapriorparms=(0,100);
run;
```

You can also provide a list of pairs to specify different sets of prior parameters for the various regression parameters and components. For example:

```
proc hpfmm;
  model y = x/ k=2;
    bayes betapriorparms( (0,10) (0,20) (.,.) (3,100) );
run;
```

The simple linear regression in the first component has a $N(0, 10)$ prior for the intercept and a $N(0, 20)$ prior for the slope. The prior for the intercept in the second component uses the HPFMM default, whereas the prior for the slope is $N(3, 100)$.

**DIAGNOSTICS=**ALL | NONE | (keyword-list)

**DIAG=**ALL | NONE | (keyword-list)

controls the computation of diagnostics for the posterior chain. You can request all posterior diagnostics by specifying DIAGNOSTICS=ALL or suppress the computation of posterior diagnostics by specifying DIAGNOSTICS=NONE. The following **keywords** enable you to select subsets of posterior diagnostics; the default is DIAGNOSTICS=(AUTOCORR).

**AUTOCORR** < (LAGS= numeric-list) >

computes for each sampled parameter the autocorrelations of lags specified in the LAGS= list. Elements in the list are truncated to integers, and repeated values are removed. If the LAGS= option is not specified, autocorrelations are computed by default for lags 1, 5, 10, and 50. See the section “Autocorrelations” on page 155 in Chapter 7, “Introduction to Bayesian Analysis Procedures,” for details.

**ESS**

computes an estimate of the effective sample size (Kass et al. 1998), the correlation time, and the efficiency of the chain for each parameter. See the section “Effective Sample Size” on page 155 in Chapter 7, “Introduction to Bayesian Analysis Procedures,” for details.

**GEWEKE** < (geweke-options) >

computes the Geweke spectral density diagnostics (Geweke 1992), which are essentially a two-sample $t$ test between the first $f_1$ portion and the last $f_2$ portion of the chain. The default is $f_1 = 0.1$ and $f_2 = 0.5$, but you can choose other fractions by using the following **geweke-options**: 

**FRAC1=value**

specifies the fraction $f_1$ for the first window.

**FRAC2=value**

specifies the fraction $f_2$ for the second window.

See the section “Geweke Diagnostics” on page 149 in Chapter 7, “Introduction to Bayesian Analysis Procedures,” for details.
HEIDELBERGER < (Heidel-options) >

HEIDEL < (Heidel-options) >

computes the Heidelberger and Welch diagnostic (which consists of a stationarity test and a half-width test) for each variable. The stationary diagnostic test tests the null hypothesis that the posterior samples are generated from a stationary process. If the stationarity test is passed, a half-width test is then carried out. See the section “Heidelberger and Welch Diagnostics” on page 151 in Chapter 7, “Introduction to Bayesian Analysis Procedures,” for more details.

These diagnostics are not performed by default. You can specify the DIAGNOSTICS=HEIDELBERGER option to request these diagnostics, and you can also specify suboptions, such as DIAGNOSTICS=HEIDELBERGER(EPS=0.05), as follows:

**SALPHA=value**

specifies the $\alpha$ level ($0 < \alpha < 1$) for the stationarity test. By default, SALPHA=0.05.

**HALPHA=value**

specifies the $\alpha$ level ($0 < \alpha < 1$) for the half-width test. By default, HALPHA=0.05.

**EPS=value**

specifies a small positive number $\epsilon$ such that if the half-width is less than $\epsilon$ times the sample mean of the retaining iterates, the half-width test is passed. By default, EPS=0.1.

MCERROR

MCSE

computes an estimate of the Monte Carlo standard error for each sampled parameter. See the section “Standard Error of the Mean Estimate” on page 156 in Chapter 7, “Introduction to Bayesian Analysis Procedures,” for details.

**MAXLAG=n**

specifies the largest lag used in computing the effective sample size and the Monte Carlo standard error. Specifying this option implies the ESS and MCERROR options. The default is MAXLAG=250.

RAFTERY < (Raftery-options) >

RL < (Raftery-options) >

computes the Raftery and Lewis diagnostics, which evaluate the accuracy of the estimated quantile ($\hat{\theta}_Q$ for a given $Q \in (0, 1)$) of a chain. $\hat{\theta}_Q$ can achieve any degree of accuracy when the chain is allowed to run for a long time. The algorithm stops when the estimated probability $\hat{P}_Q = \Pr(\theta \leq \hat{\theta}_Q)$ reaches within $\pm R$ of the value $Q$ with probability $S$; that is, $\Pr(Q - R \leq \hat{P}_Q \leq Q + R) = S$. See the section “Raftery and Lewis Diagnostics” on page 152 in Chapter 7, “Introduction to Bayesian Analysis Procedures,” for more details. The Raftery-options enable you to specify $Q$, $R$, $S$, and a precision level $\epsilon$ for a stationary test.

These diagnostics are not performed by default. You can specify the DIAGNOSTICS=RAFTERY option to request these diagnostics, and you can also specify suboptions, such as DIAGNOSTICS=RAFTERY(QUANTILE=0.05), as follows:
QUANTILE=value
Q=value

specifies the order (a value between 0 and 1) of the quantile of interest. By default, QUANTILE=0.025.

ACCURACY=value
R=value

specifies a small positive number as the margin of error for measuring the accuracy of estimation of the quantile. By default, ACCURACY=0.005.

PROB=value
S=value

specifies the probability of attaining the accuracy of the estimation of the quantile. By default, PROB=0.95.

EPS=value

specifies the tolerance level (a small positive number between 0 and 1) for the stationary test. By default, EPS=0.001.

MIXPRIORPARMS=K
MIXPRIORPARMS(value-list)

specifies the parameters used in constructing the Dirichlet prior distribution for the mixing parameters. If you specify MIXPRIORPARMS=K, the parameters of the k-dimensional Dirichlet distribution are a vector that contains the number of components in the model (k), whatever that might be. You can specify an explicit list of parameters in value-list. If the MIXPRIORPARMS option is not specified, the default Dirichlet parameter vector is a vector of length k of ones. This results in a uniform prior over the unit simplex; for k=2, this is the uniform distribution. See the section “Prior Distributions” on page 4258 for the distribution function of the Dirichlet as used by the HPFMM procedure.

ESTIMATE=MEAN | MAP

determines which overall estimate is used, based on the posterior sample, in the computation of OUTPUT statistics and certain ODS graphics. By default, the arithmetic average of the (thinned) posterior sample is used. If you specify ESTIMATE=MAP, the parameter vector is used that corresponds to the maximum log posterior density in the posterior sample. In any event, a message is written to the SAS log if postprocessing results depend on a summary estimate of the posterior sample.

INITIAL=DATA | MLE | MODE | RANDOM

determines how initial values for the Markov chain are obtained. The default when a conjugate sampler is used is INITIAL=DATA, in which case the HPFMM procedure uses the same algorithm to obtain data-dependent starting values as it uses for maximum likelihood estimation. If no conjugate sampler is available or if you use the METROPOLIS option to explicitly request that it not be used, then the default is INITIAL=MLE, in which case the maximum likelihood estimates are used as the initial values. If the maximum likelihood optimization fails, the HPFMM procedure switches to the default INITIAL=DATA.
The options INITIAL=MODE and INITIAL=RANDOM use the mode and random draws from the prior distribution, respectively, to obtain initial values. If the mode does not exist or if it falls on the boundary of the parameter space, the prior mean is used instead.

**METROPOLIS**
requests that the HPFMM procedure use the Metropolis-Hastings sampling algorithm based on Gamerman (1997), even in situations where a conjugate sampler is available.

**MUPRIORPARMS**=

specifies the parameters for the means in homogeneous mixtures without regression coefficients. The pair-specification is of the form \((a, b)\), where \(a\) and \(b\) are the two parameters of the prior distribution, optionally delimited with a comma. The actual distribution of the parameter is implied by the distribution selected in the MODEL statement. For example, it is a normal distribution for a mixture of normals, a gamma distribution for a mixture of Poisson variables, a beta distribution for a mixture of binary variables, and an inverse gamma distribution for a mixture of exponential variables. This option overrides the PRIOROPTIONS option.

The parameters correspond as follows:

- **Beta:** The parameters correspond to the \(\alpha\) and \(\beta\) parameters of the beta prior distribution such that its mean is \(\mu = \alpha / (\alpha + \beta)\) and its variance is \(\mu (1 - \mu) / (\alpha + \beta + 1)\).

- **Normal:** The parameters correspond to the mean and variance of the normal prior distribution.

- **Gamma:** The parameters correspond to the \(\alpha\) and \(\beta\) parameters of the gamma prior distribution such that its mean is \(\mu = \alpha / \beta\) and its variance is \(\alpha / \beta^2\).

- **Inverse gamma:** The parameters correspond to the \(\alpha\) and \(\beta\) parameters of the inverse gamma prior distribution such that its mean is \(\mu = \beta / (\alpha - 1)\) and its variance is \(\mu^2 / (\alpha - 2)\).

The two techniques for specifying the prior parameters with the MUPRIORPARMS option are as follows:

- Specify an equal sign and a single pair of values:

  ```
  proc hpfmm seed=12345;
  model y = / k=2;
  bayes mupriorparms=(0,50);
  run;
  ```
Specify a list of parameter pairs within parentheses:

```csharp
proc hpfmm seed=12345;
  model y = / k=2;
    bayes mupriorparms( (.,.) (1.4,10.5));
run;
```

If you specify an invalid value (outside of the parameter space for the prior distribution), the HPFMM procedure chooses the default value and writes a message to the SAS log. If you want to use the default values for a particular parameter, you can also specify missing values in the `pair-specification`. For example, the preceding list specification assigns default values for the first component and uses the values 1.4 and 10.5 for the mean and variance of the normal prior distribution in the second component. The first example assigns a $N(0, 50)$ prior distribution to the means in both components.

- `NBI=n` specifies the number of burn-in samples. During the burn-in phase, chains are not saved. The default is NBI=2000.

- `NMC=n` specifies the number of Monte Carlo samples after the burn-in. Samples after the burn-in phase are saved unless they are thinned with the THIN= option. The default is NMC=10000.

- `SAMPLE=n` specifies the number of Monte Carlo samples after the burn-in. Samples after the burn-in phase are saved unless they are thinned with the THIN= option. The default is NMC=10000.

- `OUTPOST<(outpost-options)>=data-set` requests that the posterior sample be saved to a SAS data set. In addition to variables that contain log likelihood and log posterior values, the OUTPOST data set contains variables for the parameters. The variable names for the parameters are generic (Parm_1, Parm_2, …, Parm_p). The labels of the parameters are descriptive and correspond to the “Parameter Mapping” table that is produced when the OUTPOST= option is in effect.

You can specify the following `outpost-options` in parentheses:

- `LOGPRIOR` adds the value of the log prior distribution to the data set.

- `NONSINGULAR | NONSING | COMPRESS` eliminates parameters that correspond to singular columns in the design matrix (and were not sampled) from the posterior data set. This is the default.

- `SINGULAR | SING` adds columns of zeros to the data set in positions that correspond to singularities in the model or to parameters that were not sampled for other reasons. By default, these columns of zeros are not written to the posterior data set.

- `PHIPRIORPARMS=pair-specification`
- `PHIPRIORPARMS(pair-specification ... pair-specification)` specifies the parameters for the inverse gamma prior distribution of the scale parameters ($\phi$’s) in the model. The `pair-specification` is of the form $(a, b)$, and the values are chosen such that the prior distribution has mean $\mu = b/(a - 1)$ and variance $\mu^2/(a - 2)$.

The form of the PHIPRIORPARMS with an equal sign and a single pair is used to specify one pair of prior parameters that applies to all components in the mixture. For example:
```sas
proc hpfmm seed=12345;
  model y = / k=2;
    bayes phipriorparms=(2.001,1.001);
run;
```

The form with a list of pairs is used to specify different prior parameters for the scale parameters in different components. For example:

```sas
proc hpfmm seed=12345;
  model y = / k=2;
    bayes phipriorparms( (. ,1.001) (3.001,2.001) );
run;
```

If you specify an invalid value (outside of the parameter space for the prior distribution), the HPFMM procedure chooses the default value and writes a message to the SAS log. If you want to use the default values for a particular parameter, you can also specify missing values in the `pair-specification`. For example, the preceding list specification assigns default values for the first component $a$ prior parameter and uses the value 1.001 for the $b$ prior parameter. The second pair assigns 3.001 and 2.001 for the $a$ and $b$ prior parameters, respectively.

`PRIOROPTIONS <= > (prior-options)`

`PRIOROPTS <= > (prior-options)`

specifies options related to the construction of the prior distribution and the choice of their parameters. Some `prior-options` apply only in particular models. The `BETAPIORPARMS=` and `MUPRIORPARMS=` options override this option.

You can specify the following `prior-options`:

**CONDITIONAL | COND**

chooses a conditional prior specification for the homogeneous normal and $t$ distribution response components. The default prior specification in these models is an independence prior where the mean of the $h$th component has prior $\mu_h \sim N(a, b)$. The conditional prior is characterized by $\mu_h \sim N(a, \sigma^2_h / b)$.

**DEPENDENT | DEP**

chooses a data-dependent prior for the homogeneous models without effects. The prior parameters $a$ and $b$ are chosen as follows, based on the distribution in the `MODEL` statement:

- Binary and binomial: $a = \bar{y} / (1 - \bar{y})$, $b=1$, and the prior distribution for the success probability is beta($a, b$).
- Poisson: $a = 1$, $b = 1 / \bar{y}$, and the prior distribution for $\mu$ is gamma($a, b$). See Frühwirth-Schnatter (2006, p. 280) and Viallefont, Richardson, and Greene (2002).
- Exponential: $a = 3$, $b = 2\bar{y}$, and the prior distribution for $\mu$ is inverse gamma with parameters $a$ and $b$. 

Normal and $t$: Under the default independence prior, the prior distribution for $\mu$ is $N(\tilde{y}, fs^2)$ where $f$ is the variance factor from the VAR= option and

$$s^2 = \frac{1}{n} \sum_{i=1}^{n} (y_i - \tilde{y})^2$$

Under the default conditional prior specification, the prior for $\mu_h$ is $N(a, \sigma^2_h/b)$ where $a = \tilde{y}$ and $b = 2.6/(\max\{y\} - \min\{y\})$. The prior for the scale parameter is inverse gamma with parameters 1.28 and 0.36$s^2$. For further details, see Raftery (1996) and Frühwirth-Schnatter (2006, p. 179).

**VAR=*/

specifies the variance for normal prior distributions. The default is VAR=1000. This factor is used, for example, in determining the prior variance of regression coefficients or in determining the prior variance of means in homogeneous mixtures of $t$ or normal distributions (unless a data-dependent prior is used).

**MLE=<r>**

specifies that the prior distribution for regression variables be based on a multivariate normal distribution centered at the MLEs and whose dispersion is a multiple $r$ of the asymptotic MLE covariance matrix. The default is MLE=10. In other words, if you specify PRIOROPTIONS(MLE), the HPFMM procedure chooses the prior distribution for the regression variables as $N(\hat{\beta}, 10\text{Var}(\hat{\beta}))$ where $\hat{\beta}$ is the vector of maximum likelihood estimates. The prior for the scale parameter is inverse gamma with parameters 1.28 and 0.36$s^2$ where

$$s^2 = \frac{1}{n} \sum_{i=1}^{n} (y_i - \tilde{y})^2$$

For further details, see Raftery (1996) and Frühwirth-Schnatter (2006, p. 179). If you specify PRIOROPTIONS(MLE) for the regression parameters, then the data-dependent prior is used for the scale parameter; see the PRIOROPTIONS(DEPENDENT) option above.

The MLE option is not available for mixture models in which the parameters are estimated directly on the data scale, such as homogeneous mixture models or mixtures of distributions without model effects for which a conjugate sampler is available. By using the METROPOLIS option, you can always force the HPFMM procedure to abandon a conjugate sampler in favor of a Metropolis-Hastings sampling algorithm to which the MLE option applies.

**STATISTICS <(global-options)> = ALL | NONE | keyword | (keyword-list)**

controls the number of posterior statistics produced. Specifying STATISTICS=ALL is equivalent to specifying STATISTICS=(SUMMARY INTERVAL). To suppress the computation of posterior statistics, specify STATISTICS=NONE. The default is STATISTICS=(SUMMARY INTERVAL). See the section “Summary Statistics” on page 156 in Chapter 7, “Introduction to Bayesian Analysis Procedures,” for more details.
The *global-options* include the following:

**ALPHA=numeric-list**
controls the coverage levels of the equal-tail credible intervals and the credible intervals of highest posterior density (HPD) credible intervals. The ALPHA= values must be between 0 and 1. Each ALPHA= value produces a pair of $(1 - \alpha)\%$ equal-tail and HPD credible intervals for each sampled parameter. The default is ALPHA=0.05, which results in 95\% credible intervals for the parameters.

**PERCENT=numeric-list**
requests the percentile points of the posterior samples. The values in numeric-list must be between 0 and 100. The default is PERCENT=(25 50 75), which yields for each parameter the 25th, 50th, and 75th percentiles, respectively.

The list of *keywords* includes the following:

**SUMMARY**
produces the means, standard deviations, and percentile points for the posterior samples. The default is to produce the 25th, 50th, and 75th percentiles; you can modify this list with the global PERCENT= option.

**INTERVAL**
produces equal-tail and HPD credible intervals. The default is to produce the 95\% equal-tail credible intervals and 95\% HPD credible intervals, but you can use the ALPHA= *global-option* to request credible intervals for any probabilities.

**THIN=n**
**THINNING=n**
controls the thinning of the Markov chain after the burn-in. Only one in every $k$ samples is used when THIN=k, and if NBI=$n_0$ and NMC=$n$, the number of samples kept is

$$\left\lfloor \frac{n_0 + n}{k} \right\rfloor - \left\lfloor \frac{n_0}{k} \right\rfloor$$

where $[a]$ represents the integer part of the number $a$. The default is THIN=1—that is, all samples are kept after the burn-in phase.

**TIMEINC=n**
specifies a time interval in seconds to report progress during the burn-in and sampling phase. The time interval is approximate, since the minimum time interval in which the HPFMM procedure can respond depends on the multithreading configuration.
**BY Statement**

```
BY variables ;
```

You can specify a BY statement with PROC HPFMM to obtain separate analyses of observations in groups that are 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 you specify more than one BY statement, only the last one specified is used.

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 NOTSORTED or DESCENDING option in the BY statement for the HPFMM 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 on the BY variables by using the DATASETS procedure (in Base SAS software).

For more information about BY-group processing, see the discussion in *SAS Language Reference: Concepts*. For more information about the DATASETS procedure, see the discussion in the *SAS Visual Data Management and Utility Procedures Guide*.

**CLASS Statement**

```
CLASS variable < (options)> . . . < variable < (options)> > < / global-options > ;
```

The CLASS statement names the classification variables to be used as explanatory variables in the analysis. The CLASS statement must precede the MODEL statement.

The CLASS statement for SAS high-performance analytical procedures is documented in the section “CLASS Statement” (Chapter 3, *SAS/STAT User's Guide: High-Performance Procedures*). The HPFMM procedure also supports the following *global-option* in the CLASS statement:

**UPCASE**

uppercases the values of character-valued CLASS variables before levelizing them. For example, if the UPCASE option is in effect and a CLASS variable can take the values ‘a,’ ‘A,’ and ‘b,’ then ‘a’ and ‘A’ represent the same level and the CLASS variable is treated as having only two values: ‘A’ and ‘B.’
**FREQ Statement**

```
FREQ variable;
```

The `variable` in the FREQ statement identifies a numeric variable in the data set that contains the frequency of occurrence for each observation. SAS high-performance analytical procedures that support the FREQ statement treat each observation as if it appeared `f` times, where `f` is the value of the FREQ variable for the observation. If the frequency value is not an integer, it is truncated to an integer. If the frequency value is less than 1 or missing, the observation is not used in the analysis. When the FREQ statement is not specified, each observation is assigned a frequency of 1.

**ID Statement**

```
ID variables;
```

The ID statement lists one or more variables from the input data set that are transferred to output data sets created by SAS high-performance analytical procedures, provided that the output data set produces one or more records per input observation.

For more information about the common ID statement in SAS high-performance analytical procedures, see the section “ID Statement” (Chapter 3, SAS/STAT User’s Guide: High-Performance Procedures).

**MODEL Statement**

```
MODEL response <(response-options)> = <effects> </model-options>;
MODEL events/trials = <effects> </model-options>;
MODEL + <effects> </model-options>;
```

The MODEL statement defines elements of the mixture model, such as the model effects, the distribution, and the link function. At least one MODEL statement is required. You can specify more than one MODEL statement. Each MODEL statement identifies one or more components of a mixture. For example, if components differ in their distributions, link functions, or regressor variables, then you can use separate MODEL statements to define the components. If the finite mixture model is homogeneous—in the sense that all components share the same regressors, distribution, and link function—then you can specify the mixture model with a single MODEL statement by using the K= option.

An intercept is included in each model by default. It can be removed with the NOINT option.

The dependent variable can be specified by using either the `response` syntax or the `events/trials` syntax. The `events/trials` syntax is specific to models for binomial-type data. A binomial(`n, π`) variable is the sum of `n` independent Bernoulli trials with event probability `π`. Each Bernoulli trial results in either an event or a nonevent (with probability `1 − π`). The value of the second variable, `trials`, gives the number `n` of Bernoulli trials. The value of the first variable, `events`, is the number of events out of `n`. The values of both `events` and `(trials−events)` must be nonnegative, and the value of `trials` must be positive. Other distributions that allow the `events/trials` syntax are the beta-binomial distribution and the binomial cluster model.
If the *events/trials* syntax is used, the HPFMM procedure defaults to the binomial distribution. If you use the *response* syntax, the procedure defaults to the normal distribution unless the response variable is a character variable or listed in the *CLASS* statement.

The HPFMM procedure supports a continuation-style syntax in MODEL statements. Since a mixture has only one response variable, it is sufficient to specify the response variable in one MODEL statement. Other MODEL statements can use the continuation symbol “+” before the specification of effects. For example, the following statements fit a three-component binomial mixture model:

```sql
class A;
model y/n = x / k=2;
model + A;
```

The first MODEL statement uses the “=” sign to separate response from effect information and specifies the response variable by using the *events/trials* syntax. This determines the distribution as binomial. This MODEL statement adds two components to the mixture models with different intercepts and regression slopes. The second MODEL statement adds another component to the mixture where the mean is a function of the classification main effect for variable *A*. The response is also binomial; it is a continuation from the previous MODEL statement.

There are two sets of options in the MODEL statement. The *response-options* determine how the HPFMM procedure models probabilities for binary data. The *model-options* control other aspects of model formation and inference. Table 53.4 summarizes the *response-options* and *model-options* available in the MODEL statement. These are subsequently discussed in detail in alphabetical order by option category.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Response Variable Options</strong></td>
<td></td>
</tr>
<tr>
<td>DESCENDING</td>
<td>Reverses the order of response categories</td>
</tr>
<tr>
<td>EVENT=</td>
<td>Specifies the event category in binary models</td>
</tr>
<tr>
<td>ORDER=</td>
<td>Specifies the sort order for the response variable</td>
</tr>
<tr>
<td>REFERENCE=</td>
<td>Specifies the reference category in categorical models</td>
</tr>
<tr>
<td><strong>Model Building</strong></td>
<td></td>
</tr>
<tr>
<td>DIST=</td>
<td>Specifies the response distribution</td>
</tr>
<tr>
<td>LINK=</td>
<td>Specifies the link function</td>
</tr>
<tr>
<td>K=</td>
<td>Specifies the number of mixture components</td>
</tr>
<tr>
<td>KMAX=</td>
<td>Specifies the maximum number of mixture components</td>
</tr>
<tr>
<td>KMIN=</td>
<td>Specifies the minimum number of mixture components</td>
</tr>
<tr>
<td>KRESTART</td>
<td>Requests that the starting values for each analysis be determined separately instead of sequentially</td>
</tr>
<tr>
<td>NOINT</td>
<td>Excludes fixed-effect intercept from model</td>
</tr>
<tr>
<td>OFFSET=</td>
<td>Specifies the offset variable for linear predictor</td>
</tr>
<tr>
<td><strong>Statistical Computations and Output</strong></td>
<td></td>
</tr>
<tr>
<td>ALPHA=α</td>
<td>Determines the confidence level ((1 - \alpha))</td>
</tr>
<tr>
<td>CL</td>
<td>Displays confidence limits for fixed-effects parameter estimates</td>
</tr>
<tr>
<td>EQUATE=</td>
<td>Imposes simple equality constraints on parameters in this model</td>
</tr>
<tr>
<td>LABEL=</td>
<td>Identifies the model</td>
</tr>
</tbody>
</table>
Table 53.4  continued

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PARMS</td>
<td>Provides starting values for the parameters in this model</td>
</tr>
</tbody>
</table>

**Response Variable Options**

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

You can specify the following *response-options* by enclosing them in parentheses after the *response* variable. The default is ORDER=FORMATTED.

**DESCENDING**

**DESC**

reverses the order of the response categories. If both the DESCENDING and ORDER= options are specified, PROC HPFMM 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 HPFMM 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*:

**FIRST**

designates the first ordered category as the event. This is the default.

**LAST**

designates the last ordered category as the event.

**ORDER=order-type**

specifies the sort order for the levels of the response variable. You can specify the following values for order-type:

**DATA**

sorts the levels by order of appearance in the input data set.

**FORMATTED**

sorts the levels by external formatted value, except for numeric variables with no explicit format, which are sorted by their unformatted (internal) value.

**FREQ**

sorts the levels by descending frequency count; levels with the most observations come first in the order.

**INTERNAL**

sorts the levels by unformatted value.
Chapter 53: The HPFMM Procedure

**FREQDATA**
sorts the levels by order of descending frequency count, and within counts by order of appearance in the input data set when counts are tied.

**FREQFORMATTED**
sorts the levels by order of descending frequency count, and within counts by formatted value (as above) when counts are tied.

**FREQINTERNAL**
sorts the levels by order of descending frequency count, and within counts by unformatted value when counts are tied.

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 HPFMM 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.

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*.

**REFERENCE=’category’ | keyword**

**REF=’category’ | keyword**
specifies the reference category for categorical models. For the binary response model, specifying one response category as the reference is the same as specifying the other response category as the event category. You can specify the value (formatted if a format is applied) of the reference category in quotes, or you can specify one of the following *keywords*:

**FIRST**
designates the first ordered category as the reference category.

**LAST**
designates the last ordered category as the reference category. This is the default.

**Model Options**

**ALPHA=** *number*
requests that confidence intervals be constructed for each of the parameters with confidence level 1 – *number*. The value of *number* must be between 0 and 1; the default is 0.05.

**CL**
requests that confidence limits be constructed for each of the parameter estimates. The confidence level is 0.95 by default; this can be changed with the **ALPHA=** option.
**DISTRIBUTION=keyword**

**DIST=keyword**

specifies the probability distribution for a mixture component.

If you specify the DIST= option and you do not specify a link function with the LINK= option, a default link function is chosen according to Table 53.5. If you do not specify a distribution, the HPFMM procedure defaults to the normal distribution for continuous response variables and to the binary distribution for classification or character variables, unless the *events/trial* syntax is used in the MODEL statement. If you choose the *events/trial* syntax, the HPFMM procedure defaults to the binomial distribution.

Table 53.5 lists keywords that you can specify for the DISTRIBUTION= option and the corresponding default link functions. For generalized linear models with these distributions, you can find expressions for the log-likelihood functions in the section “Log-Likelihood Functions for Response Distributions” on page 4249.

### Table 53.5  Keyword Values of the DIST= Option

<table>
<thead>
<tr>
<th>keyword</th>
<th>Alias</th>
<th>Distribution</th>
<th>Default Link Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>BETA</td>
<td>Beta</td>
<td>BETA</td>
<td>Logit</td>
</tr>
<tr>
<td>BETABINOMIAL</td>
<td>BETABIN</td>
<td>Beta-binomial</td>
<td>Logit</td>
</tr>
<tr>
<td>BINARY</td>
<td>BERNULLI</td>
<td>Binary</td>
<td>Logit</td>
</tr>
<tr>
<td>BINOMIAL</td>
<td>BIN</td>
<td>Binomial</td>
<td>Logit</td>
</tr>
<tr>
<td>BINOMCLUSTER</td>
<td>BINOMCLUS</td>
<td>Binomial cluster</td>
<td>Logit</td>
</tr>
<tr>
<td>CONSTANT &lt; (c) &gt;</td>
<td>DEGENERATE</td>
<td>Degenerate</td>
<td>N/A</td>
</tr>
<tr>
<td>EXPONENTIAL</td>
<td>EXPO</td>
<td>Exponential</td>
<td>Log</td>
</tr>
<tr>
<td>FOLDEDNORMAL</td>
<td>FNORMAL</td>
<td>Folded normal</td>
<td>Identity</td>
</tr>
<tr>
<td>GAMMA</td>
<td>GAM</td>
<td>Gamma</td>
<td>Log</td>
</tr>
<tr>
<td>GAUSSIAN</td>
<td>NORMAL</td>
<td>Normal</td>
<td>Identity</td>
</tr>
<tr>
<td>GENPOISSON</td>
<td>GPOISSON</td>
<td>Generalized Poisson</td>
<td>Log</td>
</tr>
<tr>
<td>GEOMETRIC</td>
<td>GEOM</td>
<td>Geometric</td>
<td>Log</td>
</tr>
<tr>
<td>INVGAUSS</td>
<td>IGAUSSIAN, IG</td>
<td>Inverse Gaussian</td>
<td>Inverse squared (power(–2))</td>
</tr>
<tr>
<td>LOGNORMAL</td>
<td>LOGN</td>
<td>Lognormal</td>
<td>Identity</td>
</tr>
<tr>
<td>NEGBINOMIAL</td>
<td>NEGBIN, NB</td>
<td>Negative binomial</td>
<td>Log</td>
</tr>
<tr>
<td>POISSON</td>
<td>POI</td>
<td>Poisson</td>
<td>Log</td>
</tr>
<tr>
<td>T &lt; (ν) &gt;</td>
<td>STUDENT &lt; (ν) &gt;</td>
<td>t</td>
<td>Identity</td>
</tr>
<tr>
<td>TRUNCEXPO &lt; (a,b) &gt;</td>
<td>TEXPO &lt; (a,b) &gt;</td>
<td>Truncated exponential</td>
<td>Log</td>
</tr>
<tr>
<td>TRUNCLOGN &lt; (a,b) &gt;</td>
<td>TLOGN &lt; (a,b) &gt;</td>
<td>Lognormal</td>
<td>Identity</td>
</tr>
<tr>
<td>TRUNCNEXNBIN</td>
<td>TNEGBIN, TNB</td>
<td>Negative binomial</td>
<td>Log</td>
</tr>
<tr>
<td>TRUNCNNORMAL &lt; (a,b) &gt;</td>
<td>TNORMAL &lt; (a,b) &gt;</td>
<td>Truncated normal</td>
<td>Identity</td>
</tr>
<tr>
<td>TRUNCPOISSON</td>
<td>TPOISSON, TPOI</td>
<td>Truncated Poisson</td>
<td>Log</td>
</tr>
<tr>
<td>UNIFORM &lt; (a,b) &gt;</td>
<td>UNIF &lt; (a,b) &gt;</td>
<td>Uniform</td>
<td>N/A</td>
</tr>
<tr>
<td>WEIBULL</td>
<td></td>
<td>Weibull</td>
<td>Log</td>
</tr>
</tbody>
</table>

Note that the PROC HPFMM default link for the gamma or exponential distribution is not the canonical
The binomial cluster model is a two-component model described in Morel and Nagaraj (1993); Morel and Neerchal (1997); Neerchal and Morel (1998). See Example 53.1 for an application of the binomial cluster model in a teratological experiment.

If the events/trials syntax is used, the default distribution is the binomial and only the following choices are available: DIST=BINOMIAL, DIST=BETABINOMIAL, and DIST=BINOMCLUSTER. The trials variable is ignored for all other distributions. This enables you to fit models in which some components have a binomial or binomial-like distribution. For example, suppose that variable n is a binomial denominator and variable logn is its logarithm. Then the following statements model a two-component mixture of a binomial and Poisson count model:

```plaintext
model y/n = ;
model + / dist=Poisson offset=logn;
```

The OFFSET= option is used in the second MODEL statement to specify that the Poisson counts refer to different base counts, since the trial variable n is ignored in the second model.

If DIST=BINOMIAL is specified without the events/trials syntax, then n=1 is used for the default number of trials.

DIST=TRUNCNEGBIN and DIST=TRUNCPOISSON are zero-truncated versions of DIST=NEGBINOMIAL and DIST=POISSON, respectively—that is, only the value of 0 is excluded from the support.

For DIST=TRUNCEXPO, DIST=TRUNCLOGN, and DIST=TRUNCNORMAL, you must specify the lower (a) and upper (b) truncation points of the distribution. For example:

```plaintext
DIST=TRUNCEXPO< (a,b) >
DIST=TRUNCLOGN< (a,b) >
DIST=TRUNCNORMAL< (a,b) >
```

Each of these distributions is the conditional version of its corresponding nontruncated distribution that is confined to the support \([a, b]\) (inclusive). You can specify a missing value (.) for either \(a\) or \(b\) to truncate only on the other side; that is, \(a=\) indicates a right-truncated distribution, and \(b=\) indicates a left-truncated distribution.

For several distribution specifications you can provide additional optional parameters to further define the distribution. These optional parameters are listed in the following:

**CONSTANT< (c) >** The number \(c\) specifies the value where the mass is concentrated. The default is DIST=CONSTANT(0), so you can add zero-inflation to any model by adding a MODEL statement with DIST=CONSTANT.

**T< (v) >** The number \(v\) specifies the degrees of freedom for the (shifted) \(t\) distribution. The default is DIST=T(3); this leads to a heavy-tailed distribution for which the variance is defined. See the section “Log-Likelihood Functions for Response Distributions” on page 4249 for the density function of the shifted \(t_v\) distribution.

**UNIFORM< (a,b) >** The values \(a\) and \(b\) define the support of the uniform distribution, \(a < b\). By default, \(a = 0\) and \(b = 1\).
**EQUATE=MEAN | SCALE | NONE | EFFECTS(effect-list)**

specifies simple sets of parameter constraints across the components in a MODEL statement; the default is EQUATE=NONE. This option is available only for maximum likelihood estimation. If you specify EQUATE=MEAN, the parameters that determine the mean are reduced to a single set that is applicable to all components in the MODEL statement. If you specify EQUATE=SCALE, a single parameter represents the common scale for all components in the MODEL statement. The EFFECTS option enables you to force the parameters for the chosen model effects to be equal across components; however, the number of parameters is unaffected.

For example, the following statements fit a two-component multiple regression model in which the coefficients for variable logd vary by component and the intercepts and coefficients for variable dose are the same for the two components:

```plaintext
proc hpfmm;
   model num = dose logd / equate=effects(int dose) k=2;
run;
```

To fix all coefficients across the two components, you can write the MODEL statement as

```plaintext
model num = dose logd / equate=effects(int dose logd) k=2;
```

or

```plaintext
model num = dose logd / equate=mean k=2;
```

If you restrict all parameters in a $k$-component MODEL statement to be equal, the HPFMM procedure reduces the model to $k=1$.

**K=n**

**NUMBER=n**

specifies the number of components the MODEL statement contributes to the overall mixture. For the binomial cluster model, this option is not available, since this model is a two-component model by definition.

**KMAX=n**

specifies the maximum number of components the MODEL statement contributes to the overall mixture.

If the maximum number of components in the mixture, as determined by all KMAX= options, is larger than the minimum number of components, the HPFMM procedure fits all possible models and displays summary fit information for the sequence of evaluated models. The “best” model according to the CRITERION= option in the PROC HPFMM statement is then chosen, and the remaining output and analyses performed by PROC HPFMM pertain to this “best” model.

When you use MCMC methods to estimate the parameters of a mixture, you need to ensure that the chain for a given value of $k$ has converged; otherwise, comparisons among models that have varying numbers of components might not be meaningful. You can use the FITDETAILS option to display summary and diagnostic information for the MCMC chains from each model.
If you specify the KMIN= option but not the KMAX= option, then the default value for the KMAX= option is the value of the KMIN= option (unless KMIN=0, in which case the KMAX= option is set to 1).

**KMIN=n**

specifies the minimum number of components that the MODEL statement contributes to the overall mixture. When you use MCMC methods to estimate the parameters of a mixture, you need to ensure that the chain for a given value of \( k \) has converged; otherwise, comparisons among models that have varying numbers of components might not be meaningful.

**KRESTART**

requests that the starting values for each analysis (that is, for each unique number of components as determined by the KMIN= and KMAX= options) be determined separately, in the same way as if no other analyses were performed. If you do not specify the KRESTART option, then the starting values for each analysis are based on results from the previous analysis with one less component.

**LABEL='label'**

specifies an optional label for the model that is used to identify the model in printed output, on graphics, and in data sets created from ODS tables.

**LINK=keyword**

specifies the link function in the model. The *keywords* and expressions for the associated link functions are shown in Table 53.6.

<table>
<thead>
<tr>
<th>LINK=</th>
<th>Alias</th>
<th>Link Function</th>
<th>( g(\mu) = \eta = )</th>
</tr>
</thead>
<tbody>
<tr>
<td>CLOGLOG</td>
<td>CLL</td>
<td>Complementary log-log</td>
<td>( \log(-\log(1 - \mu)) )</td>
</tr>
<tr>
<td>IDENTITY</td>
<td>ID</td>
<td>Identity</td>
<td>( \mu )</td>
</tr>
<tr>
<td>LOG</td>
<td>Log</td>
<td>Log</td>
<td>( \log(\mu) )</td>
</tr>
<tr>
<td>LOGIT</td>
<td>Logit</td>
<td>Log-log</td>
<td>( -\log(-\log(\mu)) )</td>
</tr>
<tr>
<td>LOGLOG</td>
<td>NORMIT</td>
<td>Probit</td>
<td>( \Phi^{-1}(\mu) )</td>
</tr>
<tr>
<td>PROBIT</td>
<td>PROBIT</td>
<td>Probit</td>
<td>( \Phi^{-1}(\mu) )</td>
</tr>
<tr>
<td>POWER((\lambda))</td>
<td>POW((\lambda))</td>
<td>Power with exponent ( \lambda = number )</td>
<td>[ \begin{cases} \mu^\lambda &amp; \text{if } \lambda \neq 0 \ \log(\mu) &amp; \text{if } \lambda = 0 \end{cases} ]</td>
</tr>
<tr>
<td>POWERMINUS2</td>
<td>INVERSE</td>
<td>Power with exponent -2</td>
<td>( 1/\mu^2 )</td>
</tr>
</tbody>
</table>

The default link functions for the various distributions are shown in Table 53.5.

**NOINT**

requests that no intercept be included in the model. An intercept is included by default, unless the distribution is DIST=CONSTANT or DIST=UNIFORM.
OFFSET=variable
 specifies the offset variable function for the linear predictor in the model. An offset variable can be thought of as a regressor variable whose regression coefficient is known to be 1. For example, you can use an offset in a Poisson model when counts have been obtained in time intervals of different lengths. With a log link function, you can model the counts as Poisson variables with the logarithm of the time interval as the offset variable.

PARAMETERS(parameter-specification)
PARMS(parameter-specification)
specifies starting values for the model parameters. If no PARMS option is given, the HPFMM procedure determines starting values by a data-dependent algorithm. To determine initial values for the Markov chain with Bayes estimation, see also the INITIAL= option in the BAYES statement. The specification of the parameters takes the following form: parameters in the mean function precede the scale parameters, and parameters for different components are separated by commas.

The following statements specify starting parameters for a two-component normal model. The initial values for the intercepts are 1 and –3; the initial values for the variances are 0.5 and 4.

```
proc hpfmm;
  model y = / k=2 parms(1 0.5, -3 4);
run;
```

You can specify missing values for parameters whose starting values are to be determined by the default method. Only values for parameters that participate in the optimization are specified. The values for model effects are specified on the linear (linked) scale.

---

**OUTPUT Statement**

```
OUTPUT < OUT=SAS-data-set >
  < keyword< (keyword-options) > <=name> > . . .
  < keyword< (keyword-options) > <=name> > </ options> ;
```

The OUTPUT statement creates a data set that contains observationwise statistics that are computed after fitting the model. The variables in the input data set are not included in the output data set to avoid data duplication for large data sets; however, variables specified in the ID statement are included.

The output statistics are computed based on the parameter estimates of the converged model if the parameters are estimated by maximum likelihood. If a Bayesian analysis is performed, the output statistics are computed based on the arithmetic mean in the posterior sample. You can change to the maximum posterior estimate with the ESTIMATE=MAP option in the BAYES statement.

You can specify the following syntax elements in the OUTPUT statement before the slash (/).

- **OUT=SAS-data-set**
  specifies the name of the output data set. If the OUT= option is omitted, the procedure uses the DATA convention to name the output data set.

- **keyword< (keyword-options) > <=name>**
  specifies a statistic to include in the output data set and optionally assigns the variable the name *name*. If you do not provide a name, the HPFMM procedure assigns a default name based on the type of
statistic requested. If you provide a name for a statistic that leads to multiple output statistics, the name is modified to index the associated component number.

You can use the **keyword-options** to control which type of a particular statistic is computed. The following are valid values for **keyword** and **keyword-options**:

**PREDICTED**< (COMPONENT | OVERALL) >
**PRED**< (COMPONENT | OVERALL) >
**MEAN**< (COMPONENT | OVERALL) >
requests predicted values (predicted means) for the response variable. The predictions in the output data set are mapped onto the data scale in all cases except for a binomial or binary response with **events/trials** syntax and when **PREDTYPE=COUNT** has not been specified. In that case the predictions are predicted success probabilities.

The default is to compute the predicted value for the mixture (OVERALL). You can request predictions for the means of the component distributions by adding the **COMPONENT** suboption in parentheses. The predicted values for some distributions are not identical to the parameter modeled as \( \mu \). For example, in the lognormal distribution the predicted mean is \( \exp(\mu + 0.5\phi) \) where \( \mu \) and \( \phi \) are the parameters of an underlying normal process; see the section “Log-Likelihood Functions for Response Distributions” on page 4249 for details.

**RESIDUAL**< (COMPONENT | OVERALL) >
**RESID**< (COMPONENT | OVERALL) >
requests residuals for the response or residuals in the component distributions. Only “raw” residuals on the data scale are computed (observed minus predicted).

**VARIANCE**< (COMPONENT | OVERALL) >
**VAR**< (COMPONENT | OVERALL) >
requests variances for the mixture or the component distributions.

**LOGLIKE**< (COMPONENT | OVERALL) >
**LOGL**< (COMPONENT | OVERALL) >
requests values of the log-likelihood function for the mixture or the components. For observations used in the analysis, the overall computed value is the observations’ contribution to the log likelihood; if a **FREQ** statement is present, the frequency is accounted for in the computed value. In other words, if all observations in the input data set have been used in the analysis, adding the value of the log-likelihood contributions in the **OUTPUT** data set produces the negative of the final objective function value in the “Iteration History” table. By default, the log-likelihood contribution to the mixture is computed. You can request the individual mixture component contributions with the **COMPONENT** suboption.

**MIXPROBS**< (COMPONENT | MAX) >
**MIXPROB**< (COMPONENT | MAX) >
**PRIOR**< (COMPONENT | MAX) >
**MIXWEIGHTS**< (COMPONENT | MAX) >
requests that the prior weights \( \pi_j (z, \alpha_j) \) be added to the **OUTPUT** data set. By default, the probabilities are output for all components. You can limit the output to a single statistic, the largest mixing probability, with the **MAX** suboption.
NOTE: The keyword “prior” is used here because of long-standing practice to refer to the mixing probabilities as prior weights. This must not be confused with the prior distribution and its parameters in a Bayesian analysis.

**POSTERIOR**\(<\)COMPONENT | MAX\)>  
**POST**\(<\)COMPONENT | MAX\>)  
**PROB**\(<\)COMPONENT | MAX\>)
requests that the posterior weights

\[
\frac{\pi_j(z, \alpha_j) p_j(y; x_j' \beta_j, \phi_j)}{\sum_{j=1}^{k} \pi_j(z, \alpha_j) p_j(y; x_j' \beta_j, \phi_j)}
\]

be added to the OUTPUT data set. By default, the probabilities are output for all components. You can limit the output to a single statistic, the largest posterior probability, with the MAX suboption.

NOTE: The keyword “posterior” is used here because of long-standing practice to refer to these probabilities as posterior probabilities. This must not be confused with the posterior distribution in a Bayesian analysis.

**LINP**  
**XBETA**
requests that the linear predictors for the models be added to the OUTPUT data set.

**CLASS | CATEGORY | GROUP**
adds the estimated component membership to the OUTPUT data set. An observation is associated with the component that has the highest posterior probability.

**MAXPOST | MAXPROB**
adds the highest posterior probability to the OUTPUT data set.

A keyword can appear multiple times. For example, the following OUTPUT statement requests predicted values for the mixture in addition to the predicted means in the individual components:

```
output out=hpfmmout pred=MixtureMean pred(component)=CompMean;
```

In a three-component model, this produces four variables in the hpfmmout data set: MixtureMean, CompMean_1, CompMean_2, and CompMean_3.

You can specify the following options in the OUTPUT statement after a slash (/).

**ALLSTATS**
requests that all statistics are computed. If you do not use a keyword to assign a name, the HPFMM procedure uses the default name.

**PREDTYPE=PROB | COUNT**
specifies the type of predicted values that are produced for a binomial or binary response with events/trials syntax. If PREDTYPE=PROB, the predicted values are success probabilities. If PREDTYPE=COUNT, the predicted values are success counts. The default is PREDTYPE=PROB.
PERFORMANCE Statement

PERFORMANCE < performance-options > ;

The PERFORMANCE statement defines performance parameters for multithreaded and distributed computing, passes variables that describe the distributed computing environment, and requests detailed results about the performance characteristics of the HPFMM procedure.

You can also use the PERFORMANCE statement to control whether the HPFMM procedure executes in single-machine mode or distributed mode.

The PERFORMANCE statement is documented further in the section “PERFORMANCE Statement” (Chapter 2, SAS/STAT User’s Guide: High-Performance Procedures).

PROBMODEL Statement

PROBMODEL < effects > < / probmodel-options > ;

The PROBMODEL statement defines the model effects for the mixing probabilities and their link function and starting values. Model effects (other than the implied intercept) are not supported with Bayesian estimation. By default, the HPFMM procedure models mixing probabilities on the logit scale for two-component models and as generalized logit models in situations with more than two components. The PROBMODEL statement is not required.

The generalized logit model with \( k \) categories has a common vector of regressor or design variables, \( z \), \( k - 1 \) parameter vectors that vary with category, and one linear predictor whose value is constant. The constant linear predictor is assigned by the HPFMM procedure to the last component in the model, and its value is zero \((\alpha_k = 0)\). The probability of observing category \( j \) is then

\[
\pi_j(z, \alpha_j) = \frac{\exp(z'\alpha_j)}{\sum_{i=1}^{k} \exp(z'\alpha_i)}
\]

For \( k=2 \), the generalized logit model reduces to a model with the logit link (a logistic model); hence the attribute generalized logit.

By default, an intercept is included in the model for the mixing probabilities. If you suppress the intercept with the NOINT option, you must specify at least one effect in the statement.

You can specify the following probmodel-options in the PROBMODEL statement after the slash (/):

**ALPHA=number**

requests that confidence intervals that have the confidence level \( 1 - \text{number} \) be constructed for the parameters in the probability model. The value of \( \text{number} \) must be between 0 and 1; the default is 0.05. If the probability model is simple—that is, it does not contain any effects—the confidence intervals are produced for the estimated parameters (on the logit scale) and for the mixing probabilities. This option has no effect when you perform Bayesian estimation. You can modify credible interval settings by specifying the STATISTICS(ALPHA=) option in the BAYES statement.
CL
requests that confidence limits be constructed for each of the parameter estimates. The confidence
level is 0.95 by default; this can be changed with the ALPHA= option.

LINK=keyword
specifies the link function in the model for the mixing probabilities. The default is a logit link for
models with two components. For models with more than two components, only the generalized logit
link is available. The keywords and expressions for the associated link functions for two-component
models are shown in Table 53.7.

<table>
<thead>
<tr>
<th>LINK=</th>
<th>Function</th>
<th>$g(\mu) = \eta =$</th>
</tr>
</thead>
<tbody>
<tr>
<td>CLOGLOG</td>
<td>Complementary log-log</td>
<td>$\log(-\log(1 - \mu))$</td>
</tr>
<tr>
<td>CLL</td>
<td></td>
<td></td>
</tr>
<tr>
<td>LOGIT</td>
<td>Logit</td>
<td>$\log(\mu/(1 - \mu))$</td>
</tr>
<tr>
<td>LOGLOG</td>
<td>Log-log</td>
<td>$-\log(-\log(\mu))$</td>
</tr>
<tr>
<td>PROBIT</td>
<td>Probit</td>
<td>$\Phi^{-1}(\mu)$</td>
</tr>
<tr>
<td>NORMIT</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

NOINT
requests that no intercept be included in the model for the mixing probabilities. An intercept is included
by default. If you suppress the intercept with the NOINT option, you must specify at least one other
effect for the mixing probabilities—since an empty probability model is not meaningful.

PARAMETERS(parameter-specification)
PARMS(parameter-specification)
specifies starting values for the parameters. The specification of the parameters takes the following
form: parameters in the mean function appear in a list, and parameters for different components are
separated by commas. Starting values are given on the linked scale, not in terms of probabilities. Also,
you need to specify starting values for each of the first $k-1$ components in a $k$-component model. The
linear predictor for the last component is always assumed to be zero.

The following statements specify a three-component mixture of multiple regression models. The
PROBMODEL statement does not list any effects, a standard “intercept-only” generalized logit model
is used to model the mixing probabilities.

```plaintext
proc hpfmm;
   model y = x1 x2 / k=3;
   probmodel / parms(2, 1);
run;
```
There are three linear predictors in the model for the mixing probabilities, $\alpha_1$, $\alpha_2$, and $\alpha_3$. With starting values of $\alpha_1 = 2$, $\alpha_2 = 1$, and $\alpha_3 = 0$, this leads to initial mixing probabilities of

\[
\pi_1 = \frac{e^2}{e^2 + e^1 + e^0} = 0.24 \\
\pi_2 = \frac{e^1}{e^2 + e^1 + e^0} = 0.66 \\
\pi_3 = \frac{e^0}{e^2 + e^1 + e^0} = 0.1
\]

You can specify missing values for parameters whose starting values are to be determined by the default method.

**RESTRIC**T Statement

```
RESTRIC**T** < 'label'> constraint-specification <, . . . , constraint-specification> < operator < value > > </ option > ;
```

The RESTRIC**T** statement enables you to specify linear equality or inequality constraints among the parameters of a mixture model. These restrictions are incorporated into the maximum likelihood analysis. The RESTRIC**T** statement is not available for a Bayesian analysis with the HPFMM procedure.

Following are reasons why you might want to place constraints and restrictions on the model parameters:

- to fix a parameter at a particular value
- to equate parameters in different components in a mixture
- to impose order conditions on the parameters in a model
- to specify contrasts among the parameters that the fitted model should honor

A restriction is composed of a left-hand side and a right-hand side, separated by an operator. If the operator and right-hand side are not specified, the restriction is assumed to be an equality constraint against zero. If the right-hand side is not specified, the value is assumed to be zero.

An individual constraint-specification is written in (nearly) the same form as estimable linear functions are specified in the ESTIMATE statement of the GLM, MIXED, or GLIMMIX procedure. The constraint-specification takes the form

\[
\text{model-effect value-list < . . . model-effect value-list > } < \text{(SCALE= value)} >
\]

At least one model-effect must be specified followed by one or more values in the value-list. The values in the list correspond to the multipliers of the corresponding parameter that is associated with the position in the model effect. If you specify more values in the value-list than the model-effect occupies in the model design matrix, the extra coefficients are ignored.

To specify restrictions for effects in specific components in the model, separate the constraint-specification by commas. The following statements provide an example:
proc hpfmm;
  class A;
  model y/n = A x / k = 2;
  restrict A 1 0 -1;
  restrict x 2, x -1 >= 0.5;
run;

The linear predictors for this two-component model can be written as

\[
\begin{align*}
\eta_1 &= \beta_{10} + \alpha_{11} A_1 + \cdots + \alpha_{1A} A_A + x \beta_{11} \\
\eta_2 &= \beta_{20} + \alpha_{21} A_1 + \cdots + \alpha_{2A} A_A + x \beta_{21}
\end{align*}
\]

where \( A_k \) is the binary variable associated with the \( k \)th level of \( A \).

The first RESTRICT statement applies only to the first component and specifies that the parameter estimates that are associated with the first and third level of the \( A \) effect are identical. In terms of the linear predictor, the restriction can be written as

\[\alpha_{11} - \alpha_{13} = 0\]

Now suppose that \( A \) has only two levels. Then the HPFMM procedure ignores the value \(-1\) in the first RESTRICT statement and imposes the restriction

\[\alpha_{11} = 0\]

on the fitted model.

The second RESTRICT statement involves parameters in two different components of the model. In terms of the linear predictors, the restriction can be written as

\[2\beta_{11} - \beta_{21} \geq \frac{1}{2}\]

When restrictions are specified explicitly through the RESTRICT statement or implied through the EQUATE=EFFECTS option in the MODEL statement, the HPFMM procedure lists all restrictions after the model fit in a table of linear constraints and indicates whether a particular constraint is active at the converged solution.

The following operators can be specified to separate the left- and right-hand sides of the restriction: =, >, <, >=, <=.

Some distributions involve scale parameters (the parameter \( \phi \) in the expressions of the log likelihood) and you can also use the constraint-specification to involve a component’s scale parameter in a constraint. To this end, assign a value to the keyword SCALE, separated from the model effects and value lists with parentheses. The following statements fit a two-component normal model and restrict the component variances to be equal:

proc hpfmm;
  model y = / k=2;
  restrict int 0 (scale 1),
            int 0 (scale -1);
run;
The intercept specification is necessary because each constraint-specification requires at least one model effect. The zero coefficient ensures that the intercepts are not involved in the restriction. Instead, the RESTRICT statement leads to $\phi_1 - \phi_2 = 0$.

You can specify the following option in the RESTRICT statement after a slash (/).

\[ \text{DIVISOR}=\text{value} \]

specifies a value by which all coefficients on the right-hand side and left-hand side of the restriction are divided.

---

**WEIGHT Statement**

**WEIGHT variable ;**

The WEIGHT statement is used to perform a weighted analysis. Consult the section “Log-Likelihood Functions for Response Distributions” on page 4249 for expressions on how weight variables are included in the log-likelihood functions. Because the probability structure of a mixture model is different from that of a classical statistical model, the presence of a weight variable in a mixture model cannot be interpreted as altering the variance of an observation.

Observations with nonpositive or missing weights are not included in the PROC HPFMM analysis. If a WEIGHT statement is not included, all observations used in the analysis are assigned a weight of 1.

---

**Details: HPFMM Procedure**

**A Gentle Introduction to Finite Mixture Models**

**The Form of the Finite Mixture Model**

Suppose that you observe realizations of a random variable $Y$, the distribution of which depends on an unobservable random variable $S$ that has a discrete distribution. $S$ can occupy one of $k$ states, the number of which might be unknown but is at least known to be finite. Since $S$ is not observable, it is frequently referred to as a latent variable.

Let $\pi_j$ denote the probability that $S$ takes on state $j$. Conditional on $S = j$, the distribution of the response $Y$ is assumed to be $f_j(y; \alpha_j, \beta_j | S = j)$. In other words, each distinct state $j$ of the random variable $S$ leads to a particular distributional form $f_j$ and set of parameters $\{\alpha_j, \beta_j\}$ for $Y$.

Let $\{\alpha, \beta\}$ denote the collection of $\alpha_j$ and $\beta_j$ parameters across all $j = 1$ to $k$. The marginal distribution of $Y$ is obtained by summing the joint distribution of $Y$ and $S$ over the states in the support of $S$:

\[
f(y; \alpha, \beta) = \sum_{j=1}^{k} \Pr(S = j) f(y; \alpha_j, \beta_j | S = j) \\
= \sum_{j=1}^{k} \pi_j f(y; \alpha_j, \beta_j | S = j)
\]
This is a mixture of distributions, and the $\pi_j$ are called the mixture (or prior) probabilities. Because the number of states $k$ of the latent variable $S$ is finite, the entire model is termed a finite mixture (of distributions) model.

The finite mixture model can be expressed in a more general form by representing $\alpha$ and $\beta$ in terms of regressor variables and parameters with optional additional scale parameters for $\beta$. The section “Notation for the Finite Mixture Model” on page 4183 develops this in detail.

### Mixture Models Contrasted with Mixing and Mixed Models: Untangling the Terminology Web

Statistical terminology can have its limitations. The terms mixture, mixing, and mixed models are sometimes used interchangeably, causing confusion. Even worse, the terms arise in related situations. One application needs to be eliminated from the discussion in this documentation: mixture experiments, where design factors are the proportions with which components contribute to a blend, are not mixture models and do not fall under the purview of the HPFMM procedure. However, the data from a mixture experiment might be analyzed with a mixture model, a mixing model, or a mixed model, besides other types of statistical models.

Suppose that you observe realizations of random variable $Y$ and assume that $Y$ follows some distribution $f(y; \alpha, \beta)$ that depends on parameters $\alpha$ and $\beta$. Furthermore, suppose that the model is found to be deficient in the sense that the variability implied by the fitted model is less than the observed variability in the data, a condition known as overdispersion (see the section “Overdispersion” on page 4249). To tackle the problem the statistical model needs to be modified to allow for more variability. Clearly, one way of doing this is to introduce additional random variables into the process. Mixture, mixing, and mixed models are simply different ways of adding such random variables. The section “The Form of the Finite Mixture Model” on page 4246 explains how mixture models add a discrete state variable $S$. The following two subsections explain how mixing and mixed models instead assume variation for a natural parameter or in the mean function.

#### Mixing Models

Suppose that the model is modified to allow for some random quantity $U$, which might be one of the parameters of the model or a quantity related to the parameters. Now there are two distributions to cope with: the conditional distribution of the response given the random effect $U$,

$$f(y; \alpha, \beta | u)$$

and the marginal distribution of the data. If $U$ is continuous, the marginal distribution is obtained by integration:

$$f(y; \alpha, \beta) = \int f(y; \alpha, \beta | u) \ f(u) \ du$$

Otherwise, it is obtained by summation over the support of $U$:

$$f(y; \alpha, \beta) = \sum_u \Pr(U = u) \ f(y; \alpha, \beta | u)$$

The important entity for statistical estimation is the marginal distribution $f(y; \alpha, \beta)$; the conditional distribution is often important for model description, genesis, and interpretation.

In a mixing model the marginal distribution is known and is typically of a well-known form. For example, if $Y | n$ has a binomial$(n, \mu)$ distribution and $n$ follows a Poisson distribution, then the marginal distribution of $Y$ is Poisson. The preceding operation is called mixing a binomial distribution with a Poisson distribution. Similarly, when mixing a Poisson$(\lambda)$ distribution with a gamma$(a, b)$ distribution for $\lambda$, a negative
binomial distribution results as the marginal distribution. Other important mixing models involve mixing a binomial \((n, \mu)\) random variable with a beta \((a, b)\) distribution for the binomial success probability \(\mu\). This results in a distribution known as the beta-binomial.

The finite mixtures have in common with the mixing models the introduction of random effects into the model to vary some or all of the parameters at random.

**Mixed Models**

The difference between a mixing and a mixed model is that the conditional distribution is not that important in the mixing model. It matters to motivate the overdispersed reference model and to arrive at the marginal distribution. Inferences with respect to the conditional distribution, such as predicting the random variable \(U\), are not performed in mixing models. In a mixed model the random variable \(U\) typically follows a continuous distribution—almost always a normal distribution. The random effects usually do not model the natural parameters of the distribution; instead, they are involved in linear predictors that relate to the conditional mean. For example, a linear mixed model is a model in which the response and the random effects are normally distributed, and the random effects enter the conditional mean function linearly:

\[
\begin{align*}
Y &= X\beta + ZU + \epsilon \\
U &\sim N(0, G) \\
\epsilon &\sim N(0, R) \\
\text{Cov}[U, \epsilon] &= 0
\end{align*}
\]

The conditional and marginal distributions are then

\[
\begin{align*}
Y|U &\sim N(X\beta + ZU + \epsilon, R) \\
Y &\sim N(X\beta, ZGZ' + R)
\end{align*}
\]

For this model, because of the linearity in the mean and the normality of the random effects, you could also refer to mixing the normal vector \(Y\) with the normal vector \(U\), since the marginal distribution is known. The linear mixed model can be fit with the MIXED procedure. When the conditional distribution is not normal and the random effects are normal, the marginal distribution does not have a closed form. In this class of mixed models, called generalized linear mixed models, model approximations and numerical integration methods are commonly used in model fitting; see for example, those models fit by the GLIMMIX and NLMIXED procedures. Chapter 6, “Introduction to Mixed Modeling Procedures,” contains details about the various classes of mixed models and about the relevant SAS/STAT procedures.

The previous expression for the marginal variance in the linear mixed model, \(\text{var}[Y] = ZGZ' + R\), emphasizes again that the variability in the marginal distribution of a model that contains random effects exceeds the variability in a model without the random effects \(R\).

The finite mixtures have in common with the mixed models that the marginal distribution is not necessarily a well-known model, but is expressed through a formal integration over the random-effects distribution. In contrast to the mixed models, in particular those involving nonnormal distributions or nonlinear elements, this integration is rather trivial; it reduces to a weighted and finite sum of densities or mass functions.
Overdispersion

Overdispersion is the condition by which the data are more dispersed than is permissible under a reference model. Overdispersion arises only if the variability a model can capture is limited (for example, because of a functional relationship between mean and variance). For example, a model for normal data can never be overdispersed in this sense, although the reasons that lead to overdispersion also negatively affect a misspecified model for normal data. For example, omitted variables increase the residual variance estimate because variability that should have been modeled through changes in the mean is now “picked up” as error variability.

Overdispersion is important because an overdispersed model can lead to misleading inferences and conclusions. However, diagnosing and remedying overdispersion is complicated. In order to handle it appropriately, the source of overdispersion must be identified. For example, overdispersion can arise from any of the following conditions alone or in combination:

- omitted variables and model effects
- omitted random effects (a source of random variation is not being modeled or is modeled as a fixed effect)
- correlation among the observations
- incorrect distributional assumptions
- incorrectly specified mean-variance relationships
- outliers in the data

As discussed in the previous section, introducing randomness into a system increases its variability. Mixture, mixed, and mixing models have thus been popular in modeling data that appear overdispersed. Finite mixture models are particularly powerful in this regard, because even low-order mixtures of basic, symmetric distributions (such as two- or three-component mixtures of normal or t distributions) enable you to model data with multiple modes, heavy tails, and skewness. In addition, the latent variable $S$ provides a natural way to accommodate omitted, unobservable variables into the model.

One approach to remedy overdispersion is to apply simple modifications of the variance function of the reference model. For example, with binomial-type data this approach replaces the variance of the binomial count variable $Y \sim \text{Binomial}(n, \mu)$, $\text{Var}[Y] = n \times \mu(1 - \mu)$ with a scaled version, $\phi n \times \mu(1 - \mu)$, where $\phi$ is called an overdispersion parameter, $\phi > 0$.

In addressing overdispersion problems, it is important to tackle the problem at its root. A missing scale factor on the variance function is hardly ever the root cause of overdispersion; it is only the easiest remedy.

Log-Likelihood Functions for Response Distributions

The HPFMM procedure calculates the log likelihood that corresponds to a particular response distribution according to the following formulas. The response distribution is the distribution specified (or chosen by default) through the DIST= option in the MODEL statement. The parameterizations used for log-likelihood functions of these distributions were chosen to facilitate expressions in terms of mean parameters that are modeled through an (inverse) link functions and in terms of scale parameters. These are not
necessarily the parameterizations in which parameters of prior distributions are specified in a Bayesian analysis of homogeneous mixtures. See the section “Prior Distributions” on page 4258 for details about the parameterizations of prior distributions.

The HPFMM procedure includes all constant terms in the computation of densities or mass functions. In the expressions that follow, \( l \) denotes the log-likelihood function, \( \phi \) denotes a general scale parameter, \( \mu_i \) is the “mean”, and \( w_i \) is a weight from the use of a WEIGHT statement.

For some distributions (for example, the Weibull distribution) \( \mu_i \) is not the mean of the distribution. The parameter \( \mu_i \) is the quantity that is modeled as \( \mu_i = g^{-1}(x^\prime \beta) \), where \( g^{-1}(\cdot) \) is the inverse link function and the \( x \) vector is constructed based on the effects in the MODEL statement. Situations in which the parameter \( \mu \) does not represent the mean of the distribution are explicitly mentioned in the list that follows.

The parameter \( \phi \) is frequently labeled as a “Scale” parameter in output from the HPFMM procedure. It is not necessarily the scale parameter of the particular distribution.

**Beta(\( \mu, \phi \))**

\[
l(\mu_i, \phi; y_i, w_i) = \log \left( \frac{\Gamma(\phi/w_i)}{\Gamma(\mu_i/\phi) \Gamma((1 - \mu_i)/w_i)} \right)
+ (\mu_i/\phi - 1) \log \{ y_i \}
+ ((1 - \mu_i)/\phi - 1) \log \{1 - y_i\}
\]

This parameterization of the beta distribution is due to Ferrari and Cribari-Neto (2004) and has properties \( E[Y] = \mu \), \( Var[Y] = \mu(1 - \mu)/(1 + \phi) \), \( \phi > 0 \).

**Beta-binomial(n; \( \mu, \phi \))**

\[
\phi = (1 - \rho^2)/\rho^2
\]

\[
l(\mu_i, \rho; y_i) = \log \{ \Gamma(n_i + 1) \} - \log \{ \Gamma(y_i + 1) \}
+ \log \{ \Gamma(\phi) \} - \log \{ \Gamma(n_i + \phi) \} + \log \{ \Gamma(y_i + \phi \mu_i) \}
+ \log \{ \Gamma(n_i - y_i + \phi(1 - \mu_i)) \} - \log \{ \Gamma(\phi \mu_i) \}
- \log \{ \Gamma(\phi(1 - \mu_i)) \}
\]

\[
l(\mu_i, \rho; y_i, w_i) = w_i l(\mu_i, \rho; y_i)
\]

where \( y_i \) and \( n_i \) are the events and trials in the events/trials syntax and \( 0 < \mu < 1 \). This parameterization of the beta-binomial model presents the distribution as a special case of the Dirichlet-Multinomial distribution—see, for example, Neerchal and Morel (1998). In this parameterization, \( E[Y] = n \mu \) and \( Var[Y] = n \mu(1 - \mu) (1 + (n - 1)/(\phi + 1)) \), \( 0 \leq \rho \leq 1 \). The HPFMM procedure models the parameter \( \phi \) and labels it “Scale” on the procedure output. For other parameterizations of the beta-binomial model, see Griffiths (1973) or Williams (1975).

**Binomial(n; \( \mu \))**

\[
l(\mu_i; y_i) = y_i \log \{ \mu_i \} + (n_i - y_i) \log \{1 - \mu_i\}
+ \log \{ \Gamma(n_i + 1) \} - \log \{ \Gamma(y_i + 1) \}
- \log \{ \Gamma(n_i - y_i + 1) \}
\]

\[
l(\mu_i; y_i, w_i) = w_i l(\mu_i; y_i)
\]
where $y_i$ and $n_i$ are the events and trials in the events/trials syntax and $0 < \mu < 1$. In this parameterization $E[Y] = n\mu$, $\text{Var}[Y] = n\mu(1-\mu)$.

**Binomial cluster** $(n; \mu, \pi)$

$$z = \log\{\Gamma(n_i + 1)\} - \log\{\Gamma(y_i + 1)\} - \log\{\Gamma(n_i - y_i + 1)\}$$

$$\mu_i^* = (1 - \mu_i)\pi$$

$$l(\mu_i, \pi; y_i) = \log\{\pi\} + z + y_i \log\{\mu_i^* + \mu_i\}$$

$$+ (n_i - y_i) \log\{1 - \mu_i^* - \mu_i\}$$

$$+ \log\{1 - \pi\} + z + y_i \log\{\mu_i^*\}$$

$$+ (n_i - y_i) \log\{1 - \mu_i^*\}$$

$$l(\mu_i, \pi; y_i, w_i) = w_i l(\mu_i, \pi; y_i)$$

In this parameterization, $E[Y] = n\pi$ and $\text{Var}[Y] = n\pi(1 - \pi)\{1 + \mu^2(n - 1)\}$. The binomial cluster model is a two-component mixture of a binomial$(n, \mu^* + \mu)$ and a binomial$(n, \mu^*)$ random variable. This mixture is unusual in that it fixes the number of components and because the mixing probability $\pi$ appears in the moments of the mixture components. For further details, see Morel and Nagaraj (1993); Morel and Neerchal (1997); Neerchal and Morel (1998) and Example 53.1 in this chapter. The expressions for the mean and variance in the binomial cluster model are identical to those of the beta-binomial model shown previously, with $bc = bb, bc = \rho_{bb}$.

The HPFMM procedure models the parameter $\mu$ through the MODEL statement and the parameter $\pi$ through the PROBMODEL statement.

**Constant($c$)**

$$l(y_i) = \begin{cases} 
0 & y_i = c \\
-1E20 & y_i \neq c 
\end{cases}$$

The extreme value when $y_i \neq c$ is chosen so that $\exp\{l(y_i)\}$ yields a likelihood of zero. You can change this value with the INVALIDLOGL= option in the PROC HPFMM statement. The constant distribution is useful for modeling overdispersion due to zero-inflation (or inflation of the process at support $c$).

The DIST=CONSTANT distribution is useful for modeling an inflated probability of observing a particular value (zero, by default) in data from other discrete distributions, as demonstrated in “Modeling Zero-Inflation: Is It Better to Fish Poorly or Not to Have Fished at All?” on page 4190. While it is syntactically valid to mix a constant distribution with a continuous distribution, such as DIST=LOGNORMAL, such a mixture is not mathematically appropriate, because the constant log-likelihood is the log of a probability, while a continuous log-likelihood is the log of a probability density function. If you want to mix a constant distribution with a continuous distribution, you could model the constant as a very narrow continuous distribution, such as DIST=UNIFORM($c - \delta, c + \delta$) for a small value $\delta$. However, using PROC HPFMM to analyze such mixtures is sensitive to numerical inaccuracy and ultimately unnecessary. Instead, the following approach is mathematically equivalent and more numerically stable:

1. Estimate the mixing probability $P(Y = c)$ as the proportion of observations in the data set such that $|y_i - c| < \epsilon$. 
2. Estimate the parameters of the continuous distribution from the observations for which \(|y_i - c| \geq \epsilon\).

**Exponential(\(\mu\))**

\[
l(\mu_i; y_i, w_i) = \begin{cases} 
-\log\{\mu_i\} - y_i/\mu_i & \text{if } w_i = 1 \\
w_i \log \left\{ w_i y_i/\mu_i \right\} - w_i y_i/\mu_i - \log\{y_i \Gamma(w_i)\} & \text{if } w_i \neq 1
\end{cases}
\]

In this parameterization, \(E[Y] = \mu\) and \(\text{Var}[Y] = \mu^2\).

**Folded normal(\(\mu, \phi\))**

\[
l(\mu_i, \phi; y_i, w_i) = -\frac{1}{2} \log(2\pi) - \frac{1}{2} \log(\phi/w_i) \\
+ \log \left\{ \exp \left\{ -w_i(y_i - \mu_i)^2 \right\} + \exp \left\{ -w_i(y_i + \mu_i)^2 \right\} \right\}
\]

If \(X\) has a normal distribution with mean \(\mu\) and variance \(\phi\), then \(Y = |X|\) has a folded normal distribution and log-likelihood function \(l(\mu, \phi; y, w)\) for \(y \geq 0\). The folded normal distribution arises, for example, when normally distributed measurements are observed, but their signs are not observed. The mean and variance of the folded normal in terms of the underlying \(N(\mu, \phi)\) distribution are

\[
E[Y] = \frac{1}{\sqrt{2\pi\phi}} \exp \left\{ -\frac{\mu^2}{2\phi} \right\} + \mu \left( 1 - 2\Phi \left( -\mu/\sqrt{\phi} \right) \right)
\]

\[
\text{Var}[Y] = \phi + \mu^2 - E[Y]^2
\]

The HPFMM procedure models the folded normal distribution through the mean \(\mu\) and variance \(\phi\) of the underlying normal distribution. When the HPFMM procedure computes output statistics for the response variable (for example when you use the OUTPUT statement), the mean and variance of the response \(Y\) are reported. Similarly, the fit statistics apply to the distribution of \(Y = |X|\), not the distribution of \(X\). When you model a folded normal variable, the response input variable should be positive; the HPFMM procedure treats negative values of \(Y\) as a support violation.

**Gamma(\(\mu, \phi\))**

\[
l(\mu_i, \phi; y_i, w_i) = w_i \phi \log \left\{ w_i y_i \phi / \mu_i \right\} - w_i y_i \phi / \mu_i - \log\{y_i \} - \log \{ \Gamma(w_i \phi) \}
\]

In this parameterization, \(E[Y] = \mu\) and \(\text{Var}[Y] = \mu^2/\phi, \phi > 0\). This parameterization of the gamma distribution differs from that in the GLIMMIX procedure, which expresses the log-likelihood function in terms of \(1/\phi\) in order to achieve a variance function suitable for mixed model analysis.

**Geometric(\(\mu\))**

\[
l(\mu_i; y_i, w_i) = y_i \log \left\{ \frac{\mu_i}{w_i} \right\} - (y_i + w_i) \log \left\{ 1 + \frac{\mu_i}{w_i} \right\} \\
+ \log \left\{ \frac{\Gamma(y_i + w_i)}{\Gamma(w_i) \Gamma(y_i + 1)} \right\}
\]

In this parameterization, \(E[Y] = \mu\) and \(\text{Var}[Y] = \mu + \mu^2\). The geometric distribution is a special case of the negative binomial distribution with \(\phi = 1\).
Generalized Poisson(μ, φ)

\[ \xi_i = 1 - \exp\{-\phi\}/w_i \]
\[ \mu_i^* = \mu_i - \xi(\mu_i - y_i) \]
\[ l(\mu_i^*, \xi_i; y_i, w_i) = \log\{\mu_i^* - \xi_i y_i\} + (y_i - 1) \log\{\mu_i^*\} \]
\[ -\mu_i^* - \log\{\Gamma(y_i + 1)\} \]

In this parameterization, E[Y] = μ, Var[Y] = \(\mu/(1 - \xi)^2\), and φ ≥ 0. The HPFMM procedure models the mean μ through the effects in the MODEL statement and applies a log link by default. The generalized Poisson distribution provides an overdispersed alternative to the Poisson distribution; φ = ξi = 0 produces the mass function of a regular Poisson random variable. For details about the generalized Poisson distribution and a comparison with the negative binomial distribution, see Joe and Zhu (2005).

Inverse Gaussian(μ, φ)

\[ l(\mu_i, \phi; y_i, w_i) = -1/2 \left[ w_i (y_i - \mu_i)^2 + \log\left( \frac{\phi y_i^3}{w_i} \right) + \log\{2\pi\} \right] \]

The variance is Var[Y] = φμ^3, φ > 0.

Lognormal(μ, φ)

\[ z_i = \log\{y_i\} - \mu_i \]
\[ l(\mu_i, \phi; y_i, w_i) = -1/2 \left( 2 \log\{y_i\} + \log\left( \frac{\phi}{w_i} \right) + \log\{2\pi\} + \frac{w_i z_i^2}{\phi} \right) \]

If X = log{Y} has a normal distribution with mean μ and variance φ, then Y has the log-likelihood function \(l(\mu_i, \phi; y_i, w_i)\). The HPFMM procedure models the lognormal distribution and not the “shortcut” version you can obtain by taking the logarithm of a random variable and modeling that as normally distributed. The two approaches are not equivalent, and the approach taken by PROC HPFMM is the actual lognormal distribution. Although the lognormal model is a member of the exponential family of distributions, it is not in the “natural” exponential family because it cannot be written in canonical form.

In terms of the parameters μ and φ of the underlying normal process for X, the mean and variance of Y are E[Y] = \(\exp\{\mu\}\sqrt{\omega}\) and Var[Y] = \(\exp\{2\mu\}\omega(\omega - 1)\), respectively, where \(\omega = \exp\{\phi\}\). When you request predicted values with the OUTPUT statement, the HPFMM procedure computes E[Y] and not μ.

Negative binomial(μ, φ)

\[ l(\mu_i, \phi; y_i, w_i) = y_i \log\left( \frac{\phi \mu_i}{w_i} \right) - (y_i + w_i/\phi) \log\left( 1 + \frac{\phi \mu_i}{w_i} \right) \]
\[ + \log\left( \frac{\Gamma(y_i + w_i/\phi)}{\Gamma(w_i/\phi) \Gamma(y_i + 1)} \right) \]

The variance is Var[Y] = μ + φμ^2, φ > 0.

For a given φ, the negative binomial distribution is a member of the exponential family. The parameter φ is related to the scale of the data because it is part of the variance function. However, it cannot be factored from the variance, as is the case with the φ parameter in many other distributions.
Normal($\mu, \phi$)

$$l(\mu_i, \phi; y_i, w_i) = -\frac{1}{2} \left[ \frac{w_i(y_i - \mu_i)^2}{\phi} + \log \left\{ \frac{\phi}{w_i} \right\} + \log(2\pi) \right]$$

The mean and variance are $E[Y] = \mu$ and $\text{Var}[Y] = \phi$, respectively, $\phi > 0$

Poisson($\mu$)

$$l(\mu_i; y_i, w_i) = w_i (y_i \log\{\mu_i\} - \mu_i - \log\{\Gamma(y_i + 1)\})$$

The mean and variance are $E[Y] = \mu$ and $\text{Var}[Y] = \mu$.

(Shifted) $T(v; \mu, \phi)$

$$z_i = -0.5 \log\{\phi/w_i\} + \log \{\Gamma(0.5(v + 1))\}$$

$$l(\mu_i, \phi; y_i, w_i) = -\left( \frac{v + 1}{2} \right) \log \left\{ 1 + \frac{w_i (y_i - \mu_i)^2}{\phi} \right\} + z_i$$

$v > 0, \phi > 0$. In this parameterization, $E[Y] = \mu$ when $v > 1$ and $\text{Var}[Y] = \phi v/(v - 2)$ when $v > 2$. Note that this form of the $t$ distribution is not a noncentral distribution, but that of a shifted central $t$ random variable.

Truncated Exponential($\mu; a, b$)

$$l(\mu_i; a, b, y_i, w_i) = w_i \log \left\{ \frac{w_i y_i}{\mu_i} \right\} - \frac{w_i y_i}{\mu_i} - \log\{y_i \Gamma(w_i)\}$$

$$- \log \left[ \frac{\Gamma\left( w_i, \frac{w_i b}{\mu_i} \right)}{\Gamma\left( w_i \right)} - \frac{\Gamma\left( w_i, \frac{w_i a}{\mu_i} \right)}{\Gamma\left( w_i \right)} \right]$$

where

$$\gamma(c_1, c_2) = \int_0^{c_2} t^{c_1-1} \exp(-t)dt$$

is the lower incomplete gamma function. The mean and variance are

$$E[Y] = \frac{(a + \mu_i) \exp(-a/\mu_i) - (b + \mu_i) \exp(-b/\mu_i)}{\exp(-a/\mu_i) - \exp(-b/\mu_i)}$$

$$\text{Var}[Y] = \frac{(a^2 + 2a\mu_i + 2\mu_i^2) \exp(-a/\mu_i) - (b^2 + 2b\mu_i + 2\mu_i^2) \exp(-b/\mu_i)}{\exp(-a/\mu_i) - \exp(-b/\mu_i)} - (E[Y])^2$$

Truncated Lognormal($\mu, \phi; a, b$)

$$z_i = \log\{y_i\} - \mu_i$$

$$l(\mu_i, \phi; a, b, y_i, w_i) = -\frac{1}{2} \left( 2 \log\{y_i\} + \log \left\{ \frac{\phi}{w_i} \right\} + \log(2\pi) + \frac{w_i z_i^2}{\phi} \right)$$

$$- \log \left\{ \Phi \left[ \sqrt{w_i/\phi}(\log b - \mu_i) \right] - \Phi \left[ \sqrt{w_i/\phi}(\log a - \mu_i) \right] \right\}$$
where $\Phi(\cdot)$ is the cumulative distribution function of the standard normal distribution. The mean and variance are

$$E[Y] = \exp(\mu_i + 0.5\phi) \frac{\Phi\left(\sqrt{\phi} - \frac{\log a - \mu_i}{\sqrt{\phi}}\right) - \Phi\left(\sqrt{\phi} - \frac{\log b - \mu_i}{\sqrt{\phi}}\right)}{\Phi\left(\frac{\log b - \mu_i}{\sqrt{\phi}}\right) - \Phi\left(\frac{\log a - \mu_i}{\sqrt{\phi}}\right)}$$

$$\text{Var}[Y] = \exp(2\mu_i + 2\phi) \frac{\Phi\left(2\sqrt{\phi} - \frac{\log a - \mu_i}{\sqrt{\phi}}\right) - \Phi\left(2\sqrt{\phi} - \frac{\log b - \mu_i}{\sqrt{\phi}}\right)}{\Phi\left(\frac{\log b - \mu_i}{\sqrt{\phi}}\right) - \Phi\left(\frac{\log a - \mu_i}{\sqrt{\phi}}\right)} - (E[Y])^2$$

**Truncated Negative binomial($\mu_i, \phi$)**

$$l(\mu_i, \phi; y_i, w_i) = y_i \log \left\{ \frac{\phi \mu_i}{w_i} \right\} - (y_i + w_i / \phi) \log \left\{ 1 + \frac{\phi \mu_i}{w_i} \right\}$$

$$+ \log \left\{ \frac{\Gamma(y_i + w_i / \phi)}{\Gamma(w_i / \phi)\Gamma(y_i + 1)} \right\}$$

$$- \log \left\{ 1 - \left( \frac{\phi \mu_i}{w_i} + 1 \right)^{-w_i / \phi} \right\}$$

The mean and variance are

$$E[Y] = \mu_i \left\{ 1 - (\phi \mu_i + 1)^{-1/\phi} \right\}^{-1}$$

$$\text{Var}[Y] = (1 + \phi \mu_i + \mu_i)E[Y] - (E[Y])^2$$

**Truncated Normal($\mu_i, \phi; a, b$)**

$$l(\mu_i, \phi; a, b, y_i, w_i) = -\frac{1}{2} \left[ \frac{w_i(y_i - \mu_i)^2}{\phi} + \log \left\{ \frac{\phi}{w_i} \right\} + \log(2\pi) \right]$$

$$- \log \left\{ \Phi\left[ \sqrt{w_i/\phi} (b - \mu_i) \right] - \Phi\left[ \sqrt{w_i/\phi} (a - \mu_i) \right] \right\}$$

where $\Phi(\cdot)$ is the cumulative distribution function of the standard normal distribution. The mean and variance are

$$E[Y] = \mu_i + \sqrt{\phi} \frac{\phi(a - \mu_i) - \phi(b - \mu_i)}{\Phi\left(\frac{b - \mu_i}{\sqrt{\phi}}\right) - \Phi\left(\frac{a - \mu_i}{\sqrt{\phi}}\right)}$$

$$\text{Var}[Y] = \phi \left[ 1 + \frac{a - \mu_i}{\sqrt{\phi}} \phi\left(\frac{a - \mu_i}{\sqrt{\phi}}\right) - \frac{b - \mu_i}{\sqrt{\phi}} \phi\left(\frac{b - \mu_i}{\sqrt{\phi}}\right) \right]$$

$$- \left\{ \phi\left(\frac{a - \mu_i}{\sqrt{\phi}}\right) - \phi\left(\frac{b - \mu_i}{\sqrt{\phi}}\right) \right\}^2$$

$$\left\{ \Phi\left(\frac{b - \mu_i}{\sqrt{\phi}}\right) - \Phi\left(\frac{a - \mu_i}{\sqrt{\phi}}\right) \right\}^2$$

where $\phi(\cdot)$ is the probability density function of the standard normal distribution.
**Truncated Poisson**($\mu$)

\[ l(\mu_i; y_i, w_i) = w_i (y_i \log \mu_i - \log \exp(\mu_i) - 1) - \log \Gamma(y_i + 1) \]

The mean and variance are

\[
E[Y] = \frac{\mu}{1 - \exp(-\mu_i)}
\]

\[
\text{Var}[Y] = \mu \left[ 1 - \exp(-\mu_i) - \mu_i \exp(-\mu_i) \right] \left[ 1 - \exp(-\mu_i) \right]^2
\]

**Uniform**($a, b$)

\[ l(\mu_i; y_i, w_i) = -\log (b - a) \]

The mean and variance are \( E[Y] = 0.5(a + b) \) and \( \text{Var}[Y] = \frac{(b - a)^2}{12} \).

**Weibull**($\mu, \phi$)

\[ l(\mu_i, \phi; y_i) = -\frac{\phi - 1}{\phi} \log \left\{ \frac{y_i}{\mu_i} \right\} - \log \mu_i \phi \]

\[ - \exp \left\{ \log \left\{ \frac{y_i}{\mu_i} \right\} / \phi \right\} \]

In this particular parameterization of the two-parameter Weibull distribution, the mean and variance of the random variable \( Y \) are \( E[Y] = \mu \Gamma(1 + \phi) \) and \( \text{Var}[Y] = \mu^2 \left\{ \Gamma(1 + 2\phi) - \Gamma^2(1 + \phi) \right\} \).

---

**Bayesian Analysis**

**Conjugate Sampling**

The HPFMM procedure uses Bayesian analysis via a conjugate Gibbs sampler if the model belongs to a small class of mixture models for which a conjugate sampler is available. See the section “Gibbs Sampler” on page 137 in Chapter 7, “Introduction to Bayesian Analysis Procedures,” for a general discussion of Gibbs sampling. Table 53.8 summarizes the models for which conjugate and Metropolis-Hastings samplers are available.

<table>
<thead>
<tr>
<th>Effects (exclusive of intercept)</th>
<th>Distributions</th>
<th>Available Samplers</th>
</tr>
</thead>
<tbody>
<tr>
<td>No</td>
<td>Normal or T</td>
<td>Conjugate or Metropolis-Hastings</td>
</tr>
<tr>
<td>Yes</td>
<td>Normal or T</td>
<td>Conjugate or Metropolis-Hastings</td>
</tr>
<tr>
<td>No</td>
<td>Binomial, binary, Poisson, exponential</td>
<td>Conjugate or Metropolis-Hastings</td>
</tr>
<tr>
<td>Yes</td>
<td>Binomial, binary, Poisson, exponential</td>
<td>Metropolis-Hastings only</td>
</tr>
</tbody>
</table>
The conjugate sampler enjoys greater efficiency than the Metropolis-Hastings sampler and has the advantage of sampling in terms of the natural parameters of the distribution.

You can always switch to the Metropolis-Hastings sampling algorithm in any model by adding the \texttt{METROPOLIS} option in the \texttt{BAYES} statement.

\section*{Metropolis-Hastings Algorithm}

If Metropolis-Hastings is the only sampler available for the specified model (see Table 53.8) or if the \texttt{METROPOLIS} option is specified in the \texttt{BAYES} statement, \texttt{PROC HPFMM} uses the Metropolis-Hastings approach of Gamerman (1997). See the section “Metropolis and Metropolis-Hastings Algorithms” on page 136 in Chapter 7, “Introduction to Bayesian Analysis Procedures,” for a general discussion of the Metropolis-Hastings algorithm.

The Gamerman (1997) algorithm derives a specific density that is used to generate proposals for the component-specific parameters $\beta_j$. The form of this proposal density is multivariate normal, with mean $m_j$ and covariance matrix $C_j$ derived as follows.

Suppose $\beta_j$ is the vector of model coefficients in the $j$th component and suppose that $\beta_j$ has prior distribution $N(a, R)$. Consider a generalized linear model (GLM) with link function $g(\mu) = \eta = x^T \beta$ and variance function $a(\mu)$. The pseudo-response and weight in the GLM for a weighted least squares step are

$$y^* = \eta + (y - \mu) / \frac{\partial \mu}{\partial \eta}$$

$$w = \frac{\partial \mu}{\partial \eta} / a(\mu)$$

If the model contains offsets or FREQ or WEIGHT statements, or if a \texttt{trials} variable is involved, suitable adjustments are made to these quantities.

In each component, $j = 1, \ldots, k$, form an adjusted cross-product matrix with a “pseudo” border

$$\begin{bmatrix}
X_j^T W_j X_j + R^{-1} & X_j^T W_j y_j^* + R^{-1} a \\
y_j^* W_j X_j + a' R^{-1} c
\end{bmatrix}$$

where $W_j$ is a diagonal matrix formed from the pseudo-weights $w$, $y^*$ is a vector of pseudo-responses, and $c$ is arbitrary. This is basically a system of normal equations with ridging, and the degree of ridging is governed by the precision and mean of the normal prior distribution of the coefficients. Sweeping on the leading partition leads to

$$C_j = (X_j^T W_j X_j + R^{-1})^{-1}$$

$$m_j = C_j \left( X_j^T W_j y_j^* + R^{-1} a \right)$$

where the generalized inverse is a reflexive, $g_2$-inverse (see the section “Linear Model Theory” on page 55 in Chapter 3, “Introduction to Statistical Modeling with SAS/STAT Software,” for details).

\texttt{PROC HPFMM} then generates a proposed parameter vector from the resulting multivariate normal distribution, and then accepts or rejects this proposal according to the appropriate Metropolis-Hastings thresholds.
Latent Variables via Data Augmentation

In order to fit finite Bayesian mixture models, the HPFMM procedure treats the mixture model as a missing data problem and introduces an assignment variable $S$ as in Dempster, Laird, and Rubin (1977). Since $S$ is not observable, it is frequently referred to as a latent variable. The unobservable variable $S$ assigns an observation to a component in the mixture model. The number of states, $k$, might be unknown, but it is known to be finite. Conditioning on the latent variable $S$, the component memberships of each observation is assumed to be known, and Bayesian estimation is straightforward for each component in the finite mixture model. That is, conditional on $S = j$, the distribution of the response is now assumed to be $f(y; \alpha_j, \beta_j | S = j)$. In other words, each distinct state of the random variable $S$ leads to a distinct set of parameters. The parameters in each component individually are then updated using a conjugate Gibbs sampler (where available) or a Metropolis-Hastings sampling algorithm.

The HPFMM procedure assumes that the random variable $S$ has a discrete multinomial distribution with probability $\pi_j$ of belonging to a component $j$; it can occupy one of $k$ states. The distribution for the latent variable $S$ is

$$f(S_i = j | \pi_1, \ldots, \pi_k) = \text{multinomial}(1, \pi_1, \ldots, \pi_k)$$

where $f(\cdot | \cdot)$ denotes a conditional probability density. The parameters in the density $\pi_j$ denote the probability that $S$ takes on state $j$.

The HPFMM procedure assumes a conjugate Dirichlet prior distribution on the mixture proportions $\pi_j$ written as:

$$p(\pi) = \text{Dirichlet}(a_1, \ldots, a_k)$$

where $p(\cdot)$ indicates a prior distribution.

Using Bayes’ theorem, the likelihood function and prior distributions determine a conditionally conjugate posterior distribution of $S$ and $\pi$ from the multinomial distribution and Dirichlet distribution, respectively.

Prior Distributions

The following list displays the parameterization of prior distributions for situations in which the HPFMM procedure uses a conjugate sampler in mixture models without model effects and certain basic distributions (binary, binomial, exponential, Poisson, normal, and $t$). You specify the parameters $a$ and $b$ in the formulas below in the MUPRIORPARMS= and PHIPRIORPARMS= options in the BAYES statement in these models.

**Beta**$(a, b)$

$$f(y) = \frac{\Gamma(a + b)}{\Gamma(a)\Gamma(b)} y^{a-1} (1 - y)^{b-1}$$

where $a > 0$, $b > 0$. In this parameterization, the mean and variance of the distribution are $\mu = a/(a + b)$ and $\mu (1 - \mu)/(a + b + 1)$, respectively. The beta distribution is the prior distribution for the success probability in binary and binomial distributions when conjugate sampling is used.

**Dirichlet**$(a_1, \ldots, a_k)$

$$f(y) = \frac{\Gamma\left(\sum_{i=1}^k a_i\right)}{\prod_{i=1}^k \Gamma(a_i)} y_1^{a_1-1} \cdots y_k^{a_k-1}$$
where $\sum_{i=1}^{k} y_i = 1$ and the parameters $a_i > 0$. If any $a_i$ were zero, an improper density would result. The Dirichlet density is the prior distribution for the mixture proportions. You can affect the choice of the $a_i$ through the MIXPRIORPARMS option in the BAYES statement. If $k=2$, the Dirichlet is the same as the beta($a, b$) distribution.

**Gamma($a, b$)**

$$f(y) = \frac{b^a}{\Gamma(a)} y^{a-1} \exp\{-by\}$$

where $a > 0$, $b > 0$. In this parameterization, the mean and variance of the distribution are $\mu = a/b$ and $\mu^2/b$, respectively. The gamma distribution is the prior distribution for the mean parameter of the Poisson distribution when conjugate sampling is used.

**Inverse gamma($a, b$)**

$$f(y) = \frac{b^a}{\Gamma(a)} y^{-a-1} \exp\{-b/y\}$$

where $a > 0$, $b > 0$. In this parameterization, the mean and variance of the distribution are $\mu = b/(a-1)$ if $a > 1$ and $\mu^2/(a-2)$ if $a > 2$, respectively. The inverse gamma distribution is the prior distribution for the mean parameter of the exponential distribution when conjugate sampling is used. It is also the prior distribution for the scale parameter $\phi$ in all models.

**Multinomial(1, $\pi_1$, $\cdots$, $\pi_k$)**

$$f(y) = \frac{1}{y_1! \cdots y_k!} \pi_1^{y_1} \cdots \pi_k^{y_k}$$

where $\sum_{j=1}^{k} y_j = n$, $y_j \geq 0$, $\sum_{j=1}^{k} \pi_j = 1$, and $n$ is the number of observations included in the analysis. The multinomial density is the prior distribution for the mixture proportions. The mean and variance of $Y_j$ are $\mu_j = \pi_j$ and $\mu_j(1-\mu_j)$, respectively.

**Normal($a, b$)**

$$f(y) = \frac{a}{\sqrt{2\pi b}} \exp\left\{-\frac{1}{2} \frac{(y-a)^2}{b}\right\}$$

where $b > 0$. The mean and variance of the distribution are $\mu = a$ and $b$, respectively. The normal distribution is the prior distribution for the mean parameter of the normal and $t$ distribution when conjugate sampling is used.

When a MODEL statement contains effects or if you specify the METROPOLIS option, the prior distribution for the regression parameters is multivariate normal, and you can specify the means and variances of the parameters in the BETAPRIORPARMS= option in the BAYES statement.

**Parameterization of Model Effects**

PROC HPFMM constructs a finite mixture model according to the specifications in the CLASS, MODEL, and PROBMODEL statements. Each effect in the MODEL statement generates one or more columns in the matrix $X$ for that model. The same $X$ matrix applies to all components that are associated with the MODEL
statement. Each effect in the PROBMODEL statement generates one or more columns in the matrix $Z$ from which the linear predictors in the model for the mixture probability models is formed. The same $Z$ matrix applies to all components.

The formation of effects from continuous and classification variables in the HPFMM procedure follows the same general rules and techniques as for other linear modeling procedures. For information about constructing the model effects, see the section “Specification and Parameterization of Model Effects” (Chapter 3, SAS/STAT User’s Guide: High-Performance Procedures).

---

### Computational Method

#### Multithreading

Threading refers to the organization of computational work into multiple tasks (processing units that can be scheduled by the operating system). A task is associated with a thread. Multithreading refers to the concurrent execution of threads. When multithreading is possible, substantial performance gains can be realized compared to sequential (single-threaded) execution.

The number of threads spawned by the HPFMM procedure is determined by the number of CPUs on a machine and can be controlled in the following ways:

- You can specify the CPU count with the CPUCOUNT= SAS system option. For example, if you specify the following statements, the HPFMM procedure schedules threads as if it were executing on a system that had four CPUs, regardless of the actual CPU count:

  ```sas
  options cpucount=4;
  ```

- You can specify the NTHREADS= option in the PERFORMANCE statement to determine the number of threads. This specification overrides the system option. Specify NTHREADS=1 to force single-threaded execution.

The number of threads per machine is displayed in the “Performance Information” table, which is part of the default output. The HPFMM procedure allocates one thread per CPU.

The tasks that are multithreaded by the HPFMM procedure are primarily defined by dividing the data processed on a single machine among the threads—that is, the HPFMM procedure implements multithreading through a data-parallel model. For example, if the input data set has 1,000 observations and you are running with four threads, then 250 observations are associated with each thread. All operations that require access to the data are then multithreaded. This operations include the following:

- variable levelization
- effect levelization
- formation of the crossproducts matrix
- objective function, gradient, and Hessian evaluations
- scoring of observations
In addition, operations on matrices such as sweeps might be multithreaded if the matrices are of sufficient size to realize performance benefits from managing multiple threads for the particular matrix operation.

---

**Choosing an Optimization Algorithm**

**First- or Second-Order Algorithms**

The factors that affect how you choose a particular optimization technique for a particular problem are complex. Occasionally, you might benefit from trying several algorithms.

For many optimization problems, computing the gradient takes more computer time than computing the function value. Computing the Hessian sometimes takes much more computer time and memory than computing the gradient, especially when there are many decision variables. Unfortunately, optimization techniques that do not use some kind of Hessian approximation usually require many more iterations than techniques that do use a Hessian matrix; as a result, the total run time of these techniques is often longer. Techniques that do not use the Hessian also tend to be less reliable. For example, they can terminate more easily at stationary points than at global optima.

Table 53.9 shows which derivatives are required for each optimization technique.

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>First-Order</th>
<th>Second-Order</th>
</tr>
</thead>
<tbody>
<tr>
<td>TRUREG</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>NEWRAP</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>NRRIDG</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>QUANEW</td>
<td>x</td>
<td>-</td>
</tr>
<tr>
<td>DBLDOG</td>
<td>x</td>
<td>-</td>
</tr>
<tr>
<td>CONGRA</td>
<td>x</td>
<td>-</td>
</tr>
<tr>
<td>LEVMAR</td>
<td>x</td>
<td>-</td>
</tr>
<tr>
<td>NMSIMP</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

The second-derivative methods (TRUREG, NEWRAP, and NRRIDG) are best for small problems for which the Hessian matrix is not expensive to compute. Sometimes the NRRIDG algorithm can be faster than the TRUREG algorithm, but TRUREG can be more stable. The NRRIDG algorithm requires only one matrix with \( p(p + 1)/2 \) double words; TRUREG and NEWRAP require two such matrices. Here, \( p \) denotes the number of parameters in the optimization.

The first-derivative methods QUANEW and DBLDOG are best for medium-sized problems for which the objective function and the gradient are much faster to evaluate than the Hessian. In general, the QUANEW and DBLDOG algorithms require more iterations than TRUREG, NRRIDG, and NEWRAP, but each iteration can be much faster. The QUANEW and DBLDOG algorithms require only the gradient to update an approximate Hessian, and they require slightly less memory than TRUREG or NEWRAP.

The first-derivative method CONGRA is best for large problems for which the objective function and the gradient can be computed much faster than the Hessian and for which too much memory is required to store the (approximate) Hessian. In general, the CONGRA algorithm requires more iterations than QUANEW or
DBLDOG, but each iteration can be much faster. Because CONGRA requires only a factor of \( p \) double-word memory, many large applications can be solved only by CONGRA.

The no-derivative method NMSIMP is best for small problems for which derivatives are not continuous or are very difficult to compute.

The LEVMAR method is appropriate only for least squares optimization problems.

Each optimization method uses one or more convergence criteria that determine when it has converged. An algorithm is considered to have converged when any one of the convergence criteria is satisfied. For example, under the default settings, the QUANEW algorithm converges if \( \text{ABSGCONV} < 1 \times 10^{-5} \), \( \text{FCONV} < 2 \times \epsilon \), or \( \text{GCONV} < 1 \times 10^{-8} \).

By default, the HPFMM procedure applies the NRRIDG algorithm because it can take advantage of multi-threading in Hessian computations and inversions. If the number of parameters becomes large, specifying \text{TECHNIQUE=QUANEW} (which is a first-order method with good overall properties) is recommended.

**Algorithm Descriptions**

The following subsections provide details about each optimization technique and follow the same order as Table 53.9.

**Trust Region Optimization (TRUREG)**

The trust region method uses the gradient \( g(\psi^{(k)}) \) and the Hessian matrix \( H(\psi^{(k)}) \); thus, it requires that the objective function \( f(\psi) \) have continuous first- and second-order derivatives inside the feasible region.

The trust region method iteratively optimizes a quadratic approximation to the nonlinear objective function within a hyperelliptic trust region that has radius \( \Delta \). The radius constrains the step size that corresponds to the quality of the quadratic approximation. The trust region method is implemented based on Dennis, Gay, and Welsch (1981); Gay (1983); Moré and Sorensen (1983).

The trust region method performs well for small- to medium-sized problems, and it does not need many function, gradient, and Hessian calls. However, if the computation of the Hessian matrix is computationally expensive, one of the quasi-Newton or conjugate gradient algorithms might be more efficient.

**Newton-Raphson Optimization with Line Search (NEWRAP)**

The NEWRAP technique uses the gradient \( g(\psi^{(k)}) \) and the Hessian matrix \( H(\psi^{(k)}) \); thus, it requires that the objective function have continuous first- and second-order derivatives inside the feasible region.

If second-order derivatives are computed efficiently and precisely, the NEWRAP method can perform well for medium-sized to large problems, and it does not need many function, gradient, and Hessian calls.

This algorithm uses a pure Newton step when the Hessian is positive-definite and when the Newton step reduces the value of the objective function successfully. Otherwise, a combination of ridging and line search is performed to compute successful steps. If the Hessian is not positive-definite, a multiple of the identity matrix is added to the Hessian matrix to make it positive-definite (Eskow and Schnabel 1991).

In each iteration, a line search is performed along the search direction to find an approximate optimum of the objective function. The default line-search method uses quadratic interpolation and cubic extrapolation (LIS=2).
Newton-Raphson Ridge Optimization (NRRIDG)
The NRRIDG technique uses the gradient $g(\psi^{(k)})$ and the Hessian matrix $H(\psi^{(k)})$; thus, it requires that the objective function have continuous first- and second-order derivatives inside the feasible region.

This algorithm uses a pure Newton step when the Hessian is positive-definite and when the Newton step reduces the value of the objective function successfully. If at least one of these two conditions is not satisfied, a multiple of the identity matrix is added to the Hessian matrix.

The NRRIDG method performs well for small- to medium-sized problems, and it does not require many function, gradient, and Hessian calls. However, if the computation of the Hessian matrix is computationally expensive, one of the quasi-Newton or conjugate gradient algorithms might be more efficient.

Because the NRRIDG technique uses an orthogonal decomposition of the approximate Hessian, each iteration of NRRIDG can be slower than an iteration of the NEWRAP technique, which works with a Cholesky decomposition. However, NRRIDG usually requires fewer iterations than NEWRAP.

Quasi-Newton Optimization (QUANEW)
The (dual) quasi-Newton method uses the gradient $g(\psi^{(k)})$, and it does not need to compute second-order derivatives because they are approximated. It works well for medium-sized to moderately large optimization problems, where the objective function and the gradient can be computed much faster than the Hessian. However, in general it requires more iterations than the TRUREG, NEWRAP, and NRRIDG techniques, which compute second-order derivatives. QUANEW is the default optimization algorithm because it provides an appropriate balance between the speed and stability that are required for most nonlinear mixed model applications.

The QUANEW technique that is implemented by the HPFMM procedure is the dual quasi-Newton algorithm, which updates the Cholesky factor of an approximate Hessian.

In each iteration, a line search is performed along the search direction to find an approximate optimum. The line-search method uses quadratic interpolation and cubic extrapolation to obtain a step size $\alpha$ that satisfies the Goldstein conditions (Fletcher 1987). One of the Goldstein conditions can be violated if the feasible region defines an upper limit of the step size. Violating the left-side Goldstein condition can affect the positive-definiteness of the quasi-Newton update. In that case, either the update is skipped or the iterations are restarted by using an identity matrix, resulting in the steepest descent or ascent search direction.

The QUANEW algorithm uses its own line-search technique.

Double-Dogleg Optimization (DBLDOG)
The double-dogleg optimization method combines the ideas of the quasi-Newton and trust region methods. In each iteration, the double-dogleg algorithm computes the step $s^{(k)}$ as the linear combination of the steepest descent or ascent search direction $s_{1}^{(k)}$ and a quasi-Newton search direction $s_{2}^{(k)}$:

$$s^{(k)} = \alpha_{1}s_{1}^{(k)} + \alpha_{2}s_{2}^{(k)}$$

The step is requested to remain within a prespecified trust region radius (Fletcher 1987, p. 107). Thus, the DBLDOG subroutine uses the dual quasi-Newton update but does not perform a line search.

The double-dogleg optimization technique works well for medium-sized to moderately large optimization problems, where the objective function and the gradient are much faster to compute than the Hessian. The implementation is based on Dennis and Mei (1979); Gay (1983), but it is extended for dealing with boundary and linear constraints. The DBLDOG technique generally requires more iterations than the TRUREG,
NEWRAP, and NRRIDG techniques, which require second-order derivatives; however, each of the DBLDOG iterations is computationally cheap. Furthermore, the DBLDOG technique requires only gradient calls for the update of the Cholesky factor of an approximate Hessian.

**Conjugate Gradient Optimization (CONGRA)**

Second-order derivatives are not required by the CONGRA algorithm and are not even approximated. The CONGRA algorithm can be expensive in function and gradient calls, but it requires only $O(p)$ memory for unconstrained optimization. In general, the algorithm must perform many iterations to obtain a precise solution, but each of the CONGRA iterations is computationally cheap.

The CONGRA algorithm should be used for optimization problems that have large $p$. For the unconstrained or boundary-constrained case, the CONGRA algorithm requires only $O(p)$ bytes of working memory, whereas all other optimization methods require order $O(p^2)$ bytes of working memory. During $p$ successive iterations, uninterrupted by restarts or changes in the working set, the CONGRA algorithm computes a cycle of $p$ conjugate search directions. In each iteration, a line search is performed along the search direction to find an approximate optimum of the objective function. The default line-search method uses quadratic interpolation and cubic extrapolation to obtain a step size $\alpha$ that satisfies the Goldstein conditions. One of the Goldstein conditions can be violated if the feasible region defines an upper limit for the step size. Other line-search algorithms can be specified with the LIS= option.

**Levenberg-Marquardt Optimization (LEVMAR)**

The LEVMAR algorithm performs a highly stable optimization; however, for large problems, it consumes more memory and takes longer than the other techniques. The Levenberg-Marquardt optimization technique is a slightly improved variant of the Moré (1978) implementation.

**Nelder-Mead Simplex Optimization (NMSIMP)**

The Nelder-Mead simplex method does not use any derivatives and does not assume that the objective function has continuous derivatives. The objective function itself needs to be continuous. This technique is quite expensive in the number of function calls, and it might be unable to generate precise results for $p \gg 40$.

The original Nelder-Mead simplex algorithm is implemented and extended to boundary constraints. This algorithm does not compute the objective for infeasible points, but it changes the shape of the simplex by adapting to the nonlinearities of the objective function. This adaptation contributes to an increased speed of convergence. NMSIMP uses a special termination criterion.

---

**Output Data Set**

Many procedures in SAS software add the variables from the input data set when an observationwise output data set is created. The assumption of high-performance analytical procedures is that the input data sets can be large and contain many variables. For performance reasons, the output data set contains the following:

- those variables explicitly created by the statement
- variables listed in the ID statement

This enables you to add output data set information that is necessary for subsequent SQL joins without copying the entire input data set to the output data set. For more information about output data sets that are
produced when PROC HPFMM is run in distributed mode, see the section “Output Data Sets” (Chapter 2, *SAS/STAT User’s Guide: High-Performance Procedures*).

### Default Output

The following sections describe the output that PROC HPFMM produces by default. The output is organized into various tables, which are discussed in the order of appearance for maximum likelihood and Bayes estimation, respectively.

#### Performance Information

The “Performance Information” table is produced by default. It displays information about the execution mode. For single-machine mode, the table displays the number of threads used. For distributed mode, the table displays the grid mode (symmetric or asymmetric), the number of compute nodes, and the number of threads per node.

#### Model Information

The “Model Information” table displays basic information about the model, such as the response variable, frequency variable, link function, and the model category that the HPFMM procedure determined based on your input and options. The “Model Information” table is one of a few tables that are produced irrespective of estimation technique. Most other tables are specific to Bayes or maximum likelihood estimation.

If the analysis depends on generated random numbers, the “Model Information” table also displays the random number seed used to initialize the random number generators. If you repeat the analysis and pass this seed value in the SEED= option in the PROC HPFMM statement, an identical stream of random numbers results.

#### Class Level Information

The “Class Level Information” table lists the levels of every variable specified in the CLASS statement. You should check this information to make sure that the data are correct. You can adjust the order of the CLASS variable levels with the ORDER= option in the CLASS statement. You can suppress the “Class Level Information” table completely or partially with the NOCLPRINT= option in the PROC HPFMM statement.

#### Number of Observations

The “Number of Observations” table displays the number of observations read from the input data set and the number of observations used in the analysis. If you specify a FREQ statement, the table also displays the sum of frequencies read and used. If the *events/trials* syntax is used for the response, the table also displays the number of events and trials used in the analysis.

Note that the number of observations “used” in the analysis is not unambiguous in a mixture model. An observation that is “unusable” for one component distribution (because the response value is outside of the support of the distribution) might still be usable in the mixture model when the response value is in the support of another component distribution. You can affect the way in which PROC HPFMM handles exclusion of observations due to support violations with the EXCLUSION= option in the PROC HPFMM statement.
Response Profile

For binary data, the “Response Profile” table displays the ordered value from which the HPFMM procedure determines the probability being modeled as an event for binary data. For each response category level, the frequency used in the analysis is reported.

Default Output for Maximum Likelihood

Optimization Information

The “Optimization Information” table displays basic information about the optimization setup to determine the maximum likelihood estimates, such as the optimization technique, the parameters that participate in the optimization, and the number of threads used for the calculations. This table is not produced during model selection—that is, if the KMAX= option is specified in the MODEL statement.

Iteration History

The “Iteration History” table displays for each iteration of the optimization the number of function evaluations (including gradient and Hessian evaluations), the value of the objective function, the change in the objective function from the previous iteration, and the absolute value of the largest (projected) gradient element. The objective function used in the optimization in the HPFMM procedure is the negative of the mixture log likelihood; consequently, PROC HPFMM performs a minimization. This table is not produced if you specify the KMAX= option in the MODEL statement. If you wish to see the “Iteration History” table in this setting, you must also specify the FITDETAILS option in the PROC HPFMM statement.

Convergence Status

The convergence status table is a small ODS table that follows the “Iteration History” table in the default output. In the listing, it appears as a message that identifies whether the optimization succeeded and which convergence criterion was met. If the optimization fails, the message indicates the reason for the failure. If you save the “Convergence Status” table to an output data set, a numeric Status variable is added that allows you to assess convergence programmatically. The values of the Status variable encode the following:

0  Convergence was achieved or an optimization was not performed (because of TECHNIQUE=NONE).
1  The objective function could not be improved.
2  Convergence was not achieved because of a user interrupt or because a limit was exceeded, such as the maximum number of iterations or the maximum number of function evaluations. To modify these limits, see the MAXITER=, MAXFUNC=, and MAXTIME= options in the PROC HPFMM statement.
3  Optimization failed to converge because function or derivative evaluations failed at the starting values or during the iterations or because a feasible point that satisfies the parameter constraints could not be found in the parameter space.

Fit Statistics

The “Fit Statistics” table displays a variety of fit measures based on the mixture log likelihood in addition to the Pearson statistic. All statistics are presented in “smaller is better” form. If you are fitting a single-component normal, gamma, or inverse gaussian model, the table also contains the unscaled Pearson statistic. If you are fitting a mixture model or the model has been fitted under restrictions, the table also contains the number of effective components and the number of effective parameters.
The calculation of the information criteria uses the following formulas, where \( p \) denotes the number of effective parameters, \( n \) denotes the number of observations used (or the sum of the frequencies used if a FREQ statement is present), and \( l \) is the log likelihood of the mixture evaluated at the converged estimates:

\[
\text{AIC} = -2l + 2p \\
\text{AICC} = \begin{cases} 
-2l + 2pn/(n - p - 1) & n > p + 2 \\
-2l + 2p(p + 2) & \text{otherwise}
\end{cases} \\
\text{BIC} = -2l + p \log(n)
\]

The Pearson statistic is computed simply as

\[
\text{Pearson statistic} = \sum_{i=1}^{n} f_i \frac{(y_i - \hat{\mu}_i)^2}{\text{Var}[Y_i]}
\]

where \( n \) denotes the number of observations used in the analysis, \( f_i \) is the frequency associated with the \( i \)th observation (or 1 if no frequency is specified), \( \mu_i \) is the mean of the mixture, and the denominator is the variance of the \( i \)th observation in the mixture. Note that the mean and variance in this expression are not those of the component distributions, but the mean and variance of the mixture:

\[
\mu_i = \mathbb{E}[Y_i] = \sum_{j=1}^{k} \pi_{ij} \mu_{ij} \\
\text{Var}[Y_i] = -\mu_i^2 + \sum_{j=1}^{k} \pi_{ij} \left( \sigma_{ij}^2 + \mu_{ij}^2 \right)
\]

where \( \mu_{ij} \) and \( \sigma_{ij}^2 \) are the mean and variance, respectively, for observation \( i \) in the \( j \)th component distribution and \( \pi_{ij} \) is the mixing probability for observation \( i \) in component \( j \).

The unscaled Pearson statistic is computed with the same expression as the Pearson statistic with \( n, f_i, \) and \( \mu_i \) as previously defined, but the scale parameter \( \phi \) is set to 1 in the \( \text{Var}[Y_i] \) expression.

The number of effective components and the number of effective parameters are determined by examining the converged solution for the parameters that are associated with model effects and the mixing probabilities. For example, if a component has an estimated mixing probability of zero, the values of its parameter estimates are immaterial. You might argue that all parameters should be counted towards the penalty in the information criteria. But a component with zero mixing probability in a \( k \)-component model effectively reduces the model to a \((k - 1)\)-component model. A situation of an overfit model, for which a parameter penalty needs to be taken when calculating the information criteria, is a different situation; here the mixing probability might be small, possibly close to zero.

**Parameter Estimates**

The parameter estimates, their estimated (asymptotic) standard errors, and \( p \)-values for the hypothesis that the parameter is zero are presented in the “Parameter Estimates” table. A separate table is produced for each MODEL statement, and the components that are associated with a MODEL statement are identified with an overall component count variable that counts across MODEL statements. If you assign a label to a model with the LABEL= option in the MODEL statement, the label appears in the title of the “Parameter Estimates” table. Otherwise, the internal label generated by the HPFMM procedure is used.
If the MODEL statement does not contain effects and the link function is not the identity, the inversely linked estimate is also displayed in the table. For many distributions, the inverse linked estimate is the estimated mean on the data scale. For example, in a binomial or binary model, it represents the estimated probability of an event. For some distributions (for example, the Weibull distribution), the inverse linked estimate is not the component distribution mean.

If you request confidence intervals with the CL or ALPHA= option in the MODEL statement, confidence limits are produced for the estimate on the linear scale. If the inverse linked estimate is displayed, confidence intervals for that estimate are also produced by inversely linking the confidence bounds on the linear scale.

**Mixing Probabilities**

If you fit a model with more than one component, the table of mixing probabilities is produced. If there are no effects in the PROBMODEL statement or if there is no PROBMODEL statement, the parameters are reported on the linear scale and as mixing probabilities. If model effects are present, only the linear parameters (on the scale of the logit, generalized logit, probit, and so on) are displayed.

**Default Output for Bayes Estimation**

**Bayes Information**

This table provides basic information about the sampling algorithm. The HPFMM procedure uses either a conjugate sampler or a Metropolis-Hastings sampling algorithm based on Gamerman (1997). The table reveals, for example, how many model parameters are sampled, how many parameters associated with mixing probabilities are sampled, and how many threads are used to perform multithreaded analysis.

**Prior Distributions**

The “Prior Distributions” table lists for each sampled parameter the prior distribution and its parameters. The mean and variance (if they exist) for those values of the parameters are also displayed, along with the initial value for the parameter in the Markov chain. The Component column in this table identifies the mixture component to which a particular parameter belongs. You can control how the HPFMM procedure determines initial values with the INITIAL= option in the BAYES statement.

**Bayesian Fit Statistics**

The “Bayesian Fit Statistics” table shows three measures based on the posterior sample. The “Average -2 Log Likelihood” is derived from the average mixture log-likelihood for the data, where the average is taken over the posterior sample. The deviance information criterion (DIC) is a Bayesian measure of model fit and the effective number of parameters \( p_D \) is a penalization term used in the computation of the DIC. See the section “Summary Statistics” on page 156 in Chapter 7, “Introduction to Bayesian Analysis Procedures,” for a detailed discussion of the DIC and \( p_D \).

**Posterior Summaries**

The arithmetic mean, standard deviation, and percentiles of the posterior distribution of the parameter estimates are displayed in the “Posterior Summaries” table. By default, the HPFMM procedure computes the 25th, 50th (median), and 75th percentiles of the sampling distribution. You can modify the percentiles through suboptions of the STATISTICS option in the BAYES statement. If a parameter corresponds to a singularity in the design and was removed from sampling for that purpose, it is also displayed in the table of posterior summaries (and in other tables that relate to output from the BAYES statement). The posterior sample size for such a parameter is shown as \( N = 0 \).
Posterior Intervals
The table of “Posterior Intervals” displays equal-tail intervals and intervals of highest posterior density for each parameter. By default, intervals are computed for an $\alpha$-level of 0.05, which corresponds to 95% intervals. You can modify this confidence level by providing one or more $\alpha$ values in the ALPHA= suboption of the STATISTICS option in the BAYES statement. The computation of these intervals is detailed in section “Summary Statistics” on page 156 in Chapter 7, “Introduction to Bayesian Analysis Procedures.”

Posterior Autocorrelations
Autocorrelations for the posterior estimates are computed by default for autocorrelation lags 1, 5, 10, and 50, provided that a sufficient number of posterior samples is available. See the section “Assessing Markov Chain Convergence” on page 142 in Chapter 7, “Introduction to Bayesian Analysis Procedures,” for the computation of posterior autocorrelations and their utility in diagnosing convergence of Markov chains. You can modify the list of lags for which posterior autocorrelations are calculated with the AUTOCORR suboption of the DIAGNOSTICS= option in the BAYES statement.

ODS Table Names
Each table created by PROC HPFMM has a name associated with it, and you must use this name to reference the table when you use ODS statements. These names are listed in Table 53.10.

<table>
<thead>
<tr>
<th>Table Name</th>
<th>Description</th>
<th>Required Statement / Option</th>
</tr>
</thead>
<tbody>
<tr>
<td>Autocorr</td>
<td>Autocorrelation among posterior estimates</td>
<td>BAYES</td>
</tr>
<tr>
<td>BayesInfo</td>
<td>Basic information about Bayesian estimation</td>
<td>BAYES</td>
</tr>
<tr>
<td>ClassLevels</td>
<td>Level information from the CLASS statement</td>
<td>CLASS</td>
</tr>
<tr>
<td>CompDescription</td>
<td>Component description in models with varying number of components</td>
<td>KMAX= in MODEL with ML estimation</td>
</tr>
<tr>
<td>CompEvaluation</td>
<td>Comparison of mixture models with varying number of components</td>
<td>KMAX= in MODEL with ML estimation</td>
</tr>
<tr>
<td>CompInfo</td>
<td>Component information</td>
<td>COMPONENTINFO option in PROC HPFMM statement</td>
</tr>
<tr>
<td>ConvergenceStatus</td>
<td>Status of optimization at conclusion of optimization</td>
<td>Default output</td>
</tr>
<tr>
<td>Constraints</td>
<td>Linear equality and inequality constraints</td>
<td>RESTRICT statement or EQUIATE=EFFECTS option in MODEL statement</td>
</tr>
<tr>
<td>Corr</td>
<td>Asymptotic correlation matrix of parameter estimates (ML) or empirical correlation matrix of the Bayesian posterior estimates</td>
<td>CORR option in PROC HPFMM statement</td>
</tr>
<tr>
<td>Table Name</td>
<td>Description</td>
<td>Required Statement / Option</td>
</tr>
<tr>
<td>-------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>-----------------------------------------------------------------</td>
</tr>
<tr>
<td>Cov</td>
<td>Asymptotic covariance matrix of parameter estimates (ML) or empirical covariance matrix of the Bayesian posterior estimates</td>
<td>COV option in PROC HPFMM statement</td>
</tr>
<tr>
<td>CovI</td>
<td>Inverse of the covariance matrix of the parameter estimates</td>
<td>COVI option in PROC HPFMM statement</td>
</tr>
<tr>
<td>ESS</td>
<td>Effective sample size</td>
<td>DIAG=ESS option in BAYES statement</td>
</tr>
<tr>
<td>FitStatistics</td>
<td>Fit statistics</td>
<td>Default output</td>
</tr>
<tr>
<td>Geweke</td>
<td>Geweke diagnostics (Geweke 1992) for Markov chain</td>
<td>DIAG=GEWEKE option in BAYES statement</td>
</tr>
<tr>
<td>Hessian</td>
<td>Hessian matrix from the maximum likelihood optimization, evaluated at the converged estimates</td>
<td>HESSIAN</td>
</tr>
<tr>
<td>IterHistory</td>
<td>Iteration history</td>
<td>Default output for ML estimation</td>
</tr>
<tr>
<td>MCSE</td>
<td>Monte Carlo standard errors</td>
<td>DIAG=MCERROR in BAYES statement</td>
</tr>
<tr>
<td>MixingProbs</td>
<td>Solutions for the parameter estimates associated with effects in PROBMODEL statements</td>
<td>Default output for ML estimation if number of components is greater than 1</td>
</tr>
<tr>
<td>ModelInfo</td>
<td>Model information</td>
<td>Default output</td>
</tr>
<tr>
<td>NObs</td>
<td>Number of observations read and used, number of trials and events</td>
<td>Default output</td>
</tr>
<tr>
<td>OptInfo</td>
<td>Optimization information</td>
<td>Default output for ML estimation</td>
</tr>
<tr>
<td>ParameterEstimates</td>
<td>Solutions for the parameter estimates associated with effects in MODEL statements</td>
<td>Default output for ML estimation</td>
</tr>
<tr>
<td>ParameterMap</td>
<td>Mapping of parameter names to OUTPOST= data set</td>
<td>OUTPOST= option in BAYES statement</td>
</tr>
<tr>
<td>PriorInfo</td>
<td>Prior distributions and initial value of Markov chain</td>
<td>BAYES</td>
</tr>
<tr>
<td>PostSummaries</td>
<td>Summary statistics for posterior estimates</td>
<td>BAYES</td>
</tr>
<tr>
<td>PostIntervals</td>
<td>Equal-tail and highest posterior density intervals for posterior estimates</td>
<td>BAYES</td>
</tr>
<tr>
<td>ResponseProfile</td>
<td>Response categories and category modeled</td>
<td>Default output in models with binary response</td>
</tr>
</tbody>
</table>
ODS Graphics

You can reference every graph produced through ODS Graphics with a name. The names of the graphs that PROC HPFMM generates are listed in Table 53.11, along with the required statements and options.

<table>
<thead>
<tr>
<th>ODS Graph Name</th>
<th>Plot Description</th>
<th>Option</th>
</tr>
</thead>
<tbody>
<tr>
<td>TADPanel</td>
<td>Panel of diagnostic graphs to assess convergence of Markov chains</td>
<td>BAYES</td>
</tr>
<tr>
<td>DensityPlot</td>
<td>Histogram and density with component distributions</td>
<td>Default plot for homogeneous mixtures</td>
</tr>
<tr>
<td>CriterionPanel</td>
<td>Panel of plots showing progression of model fit criteria for mixtures with different numbers of components</td>
<td>KMIN= and KMAX= options in MODEL statement</td>
</tr>
</tbody>
</table>

Examples: HPFMM Procedure

Example 53.1: Modeling Mixing Probabilities: All Mice Are Equal, but Some Mice Are More Equal Than Others

This example demonstrates how you can model the means and mixture proportions separately in a binomial cluster model. It also compares the binomial cluster model to the beta-binomial model.

In a typical teratological experiment, the offspring of animals that were exposed to a toxin during pregnancy are studied for malformation. If you count the number of malformed offspring in a litter of size $n$, then this count is typically not binomially distributed. The responses of the offspring from the same litter are not independent; hence their sum does not constitute a binomial random variable. Relative to a binomial model, data from teratological experiments exhibit overdispersion because ignoring positive correlation among the responses tends to overstate the precision of the parameter estimates. Overdispersion mechanisms are briefly discussed in the section “Overdispersion” on page 4249.
In this application, the focus is on mixtures and models that involve a mixing mechanism. The mixing approach (Williams 1975; Haseman and Kupper 1979) supposes that the binomial success probability is a random variable that follows a beta($\alpha, \beta$) distribution:

\[
Y | \mu \sim \text{Binomial}(n, \mu) \\
\mu \sim \text{Beta}(\alpha, \beta) \\
Y \sim \text{Beta-binomial}(n, \mu, \phi) \\
E[Y] = n\pi \\
\text{Var}[Y] = n\pi(1 - \pi) \{1 + \mu^2(n - 1)\}
\]

If $\mu = 0$, then the beta-binomial distribution reduces to a standard binomial model with success probability $\pi$. The parameterization of the beta-binomial distribution used by the HPFMM procedure is based on Neerchal and Morel (1998), see the section “Log-Likelihood Functions for Response Distributions” on page 4249 for details.

Morel and Nagaraj (1993); Morel and Neerchal (1997); Neerchal and Morel (1998) propose a different model to capture dependency within binomial clusters. Their model is a two-component mixture that gives rise to the same mean and variance function as the beta-binomial model. The genesis is different, however. In the binomial cluster model of Morel and Neerchal, suppose there is a cluster of $n$ Bernoulli outcomes with success probability $\mu$. The number of responses in the cluster decomposes into $N$ outcomes that all respond with either “success” or “failure”; the important aspect is that they all respond identically. The remaining $n - N$ Bernoulli outcomes respond independently, so the sum of successes in this group is a binomial$(n - N, \pi)$ random variable. Denote the probability with which cluster members fall into the group of identical respondents as $\pi$. Then $1 - \mu$ is the probability that a response belongs to the group of independent Bernoulli outcomes.

It is easy to see how this process of dividing the individual Bernoulli outcomes creates clustering. The binomial cluster model can be written as the two-component mixture

\[
\Pr(Y = y) = \pi \Pr(U = y) + (1 - \pi) \Pr(V = y)
\]

where $U \sim \text{Binomial}(n, \mu^* + \mu)$, $V \sim \text{Binomial}(n, \mu^*)$, and $\mu^* = (1 - \mu)\pi$. This mixture model is somewhat unusual because the mixing probability $\pi$ appears as a parameter in the component distributions. The two probabilities involved, $\pi$ and $\mu$, have the following interpretation: $\pi$ is the unconditional probability of success for any observation, and $\mu$ is the probability with which the Bernoulli observations respond identically. The complement of this probability, $1 - \mu$, is the probability with which the Bernoulli outcomes respond independently. If $\mu = 0$, then the two-component mixture reduces to a standard Binomial model with success probability $\pi$. Since both $\pi$ and $\mu$ are involved in the success probabilities of the two Binomial variables in the mixture, you can affect these binomial means by specifying effects in the PROBMODEL statement (for the $\pi$s) or the MODEL statement (for the $\mu$s). In a “straight” two-component Binomial mixture,

\[
\pi \text{Binomial}(n, \mu_1) + (1 - \pi)\text{Binomial}(n, \mu_2)
\]

you would vary the success probabilities $\mu_1$ and $\mu_2$ through the MODEL statement.

With the HPFMM procedure, you can fit the beta-binomial model by specifying DIST=BETABIN and the binomial cluster model by specifying DIST=BINOMCLUS in the MODEL statement.

Morel and Neerchal (1997) report data from a completely randomized design that studies the teratogenicity of phenytoin in 81 pregnant mice. The treatment structure of the experiment is an augmented factorial. In
addition to an untreated control, mice received 60 mg/kg of phenytoin (PHT), 100 mg/kg of trichloropropene oxide (TCP0), and their combination. The design was augmented with a control group that was treated with water. As in Morel and Neerchal (1997), the two control groups are combined here into a single group.

The following DATA step creates the data for this analysis as displayed in Table 1 of Morel and Neerchal (1997). The second DATA step creates continuous variables $x_1$–$x_3$ to match the parameterization of these authors.

```plaintext
data ossi;
  length tx $8;
  input tx$ n @@;
  do i=1 to n;
    input y m @@;
    output;
  end;
  drop i;
  datalines;
Control 18 8 8 9 9 7 9 0 5 3 3 5 8 9 10 5 8 5 8 1 6 0 5
8 8 9 10 5 5 4 7 9 10 6 6 3 5
Control 17 8 9 7 10 10 10 1 6 6 6 1 9 8 9 6 7 5 5 7 9
2 5 5 6 2 8 1 8 0 2 7 8 5 7
PHT 19 1 9 4 9 3 7 4 7 0 7 0 4 1 8 1 7 2 7 2 8 1 7
0 2 3 10 3 7 2 7 0 8 0 8 1 0 1 1
TCPO 16 0 5 7 10 4 4 8 11 6 10 6 9 3 4 2 8 0 6 0 9
3 6 2 9 7 9 1 1 0 8 8 6 9
PHT+TCPO 11 2 2 0 7 1 8 7 8 0 10 0 4 0 6 0 7 6 6 1 6 1 7
;

data ossi;
  set ossi;
  array xx{3} x1-x3;
  do i=1 to 3; xx(i)=0; end;
  pht = 0;
  tcpo = 0;
  if (tx='TCPO') then do;
    xx{1} = 1;
    tcpo = 100;
  end; else if (tx='PHT') then do;
    xx{2} = 1;
    pht = 60;
  end; else if (tx='PHT+TCPO') then do;
    pht = 60;
    tcpo = 100;
    xx{1} = 1; xx{2} = 1; xx{3}=1;
  end;
  run;
```

The HPFMM procedure models the mean parameters $\mu$ through the MODEL statement and the mixing proportions $\pi$ through the PROBMODEL statement. In the binomial cluster model, you can place a regression structure on either set of probabilities, and the regression structure does not need to be the same. In the following statements, the unconditional probability of ossification is modeled as a two-way factorial, whereas the intralitter effect—the propensity to group within a cluster—is assumed to be constant:
The **CLASS** statement declares the PHT and TCPO variables as classification variables. They affect the analysis through their levels, not through their numeric values. The **MODEL** statement declares the distribution of the data to follow a binomial cluster model. The HPFMM procedure then automatically assumes that the model is a two-component mixture. An intercept is included by default. The **PROBMODEL** statement declares the effect structure for the mixing probabilities. The unconditional probability of ossification of a fetus depends on the main effects and the interaction in the factorial.

The “Model Information” table displays important details about the model fit with the HPFMM procedure (Output 53.1.1). Although no K= option was specified in the **MODEL** statement, the HPFMM procedure recognizes the model as a two-component model. The “Class Level Information” table displays the levels and values of the PHT and TCPO variables. Eighty-one observations are read from the data and are used in the analysis. These observations comprise 287 events and 585 total outcomes.

**Output 53.1.1** Model Information in Binomial Cluster Model with Constant Clustering Probability

<table>
<thead>
<tr>
<th>The HPFMM Procedure</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Model Information</strong></td>
</tr>
<tr>
<td>Data Set</td>
</tr>
<tr>
<td>Response Variable (Events)</td>
</tr>
<tr>
<td>Response Variable (Trials)</td>
</tr>
<tr>
<td>Type of Model</td>
</tr>
<tr>
<td>Distribution</td>
</tr>
<tr>
<td>Components</td>
</tr>
<tr>
<td>Link Function</td>
</tr>
<tr>
<td>Estimation Method</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>Class Level Information</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Class Levels Values</strong></td>
</tr>
<tr>
<td>pht</td>
</tr>
<tr>
<td>tcpo</td>
</tr>
</tbody>
</table>

| Number of Observations Read | 81                     |
| Number of Observations Used | 81                     |
| Number of Events            | 287                    |
| Number of Trials            | 585                    |

The “Optimization Information” table in **Output 53.1.2** gives details about the maximum likelihood optimization. By default, the HPFMM procedure uses a quasi-Newton algorithm. The model contains five parameters, four of which are part of the model for the mixing probabilities. The fifth parameter is the intercept in the model for $\mu$. 
Example 53.1: Modeling Mixing Probabilities: All Mice Are Equal, but Some Mice Are More Equal Than Others

Output 53.1.2 Optimization in Binomial Cluster Model with Constant Clustering Probability

<table>
<thead>
<tr>
<th>Optimization Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Optimization Technique</td>
</tr>
<tr>
<td>Parameters in Optimization</td>
</tr>
<tr>
<td>Mean Function Parameters</td>
</tr>
<tr>
<td>Scale Parameters</td>
</tr>
<tr>
<td>Mixing Prob Parameters</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Iteration History</th>
</tr>
</thead>
<tbody>
<tr>
<td>Iteration</td>
</tr>
<tr>
<td>-----------</td>
</tr>
<tr>
<td>0</td>
</tr>
<tr>
<td>1</td>
</tr>
<tr>
<td>2</td>
</tr>
<tr>
<td>3</td>
</tr>
<tr>
<td>4</td>
</tr>
<tr>
<td>5</td>
</tr>
<tr>
<td>6</td>
</tr>
<tr>
<td>7</td>
</tr>
<tr>
<td>8</td>
</tr>
<tr>
<td>9</td>
</tr>
</tbody>
</table>

Convergence criterion (GCONV=1E-8) satisfied.

<table>
<thead>
<tr>
<th>Fit Statistics</th>
</tr>
</thead>
<tbody>
<tr>
<td>-2 Log Likelihood</td>
</tr>
<tr>
<td>AIC (Smaller is Better)</td>
</tr>
<tr>
<td>AICC (Smaller is Better)</td>
</tr>
<tr>
<td>BIC (Smaller is Better)</td>
</tr>
<tr>
<td>Pearson Statistic</td>
</tr>
<tr>
<td>Effective Parameters</td>
</tr>
<tr>
<td>Effective Components</td>
</tr>
</tbody>
</table>

After nine iterations, the iterative optimization converges. The $-2$ log likelihood at the converged solution is 305.1, and the Pearson statistic is 89.2077. The HPFMM procedure computes the Pearson statistic as a general goodness-of-fit measure that expresses the closeness of the fitted model to the data.

The estimates of the parameters in the conditional probability $\mu$ and in the unconditional probability $\pi$ are given in Output 53.1.3. The intercept estimate in the model for $\mu$ is 0.3356. Since the default link in the binomial cluster model is the logit link, the estimate of the conditional probability is

$$\hat{\mu} = \frac{1}{1 + \exp\{-0.3356\}} = 0.5831$$

This value is displayed in the “Inverse Linked Estimate” column. There is greater than a 50% chance that the individual fetuses in a litter provide the same response. The clustering tendency is substantial.
Output 53.1.3 Parameter Estimates in Binomial Cluster Model with Constant Clustering Probability

<table>
<thead>
<tr>
<th>Parameter Estimates for Binomial Cluster Model</th>
</tr>
</thead>
<tbody>
<tr>
<td>Component Effect</td>
</tr>
<tr>
<td>1 Intercept</td>
</tr>
</tbody>
</table>

Parameter Estimates for Mixing Probabilities

| Component Effect | pht | tcpo | Estimate | Standard Error | z Value | Pr > |z| |
|------------------|-----|------|----------|----------------|---------|-------|---|
| 1 Intercept      | -1.2194 | 0.4690 | -2.60 | 0.0093 |
| 1 pht            | 0 | 0.9129 | 0.5608 | 1.63 | 0.1036 |
| 1 tcpo           | 0 | 0.3295 | 0.5534 | 0.60 | 0.5516 |
| 1 tcpo           | 100 | 0 | . | . |
| 1 pht*tcpo       | 0 | 0 | 0.6162 | 0.6678 | 0.92 | 0.3561 |
| 1 pht*tcpo       | 100 | 0 | . | . |
| 1 pht*tcpo       | 60 | 0 | . | . |
| 1 pht*tcpo       | 60 | 100 | 0 | . | . |

The “Mixing Probabilities” table displays the estimates of the parameters in the model for \( \pi \) on the logit scale (Output 53.1.3). Table 53.12 constructs the estimates of the unconditional probabilities of ossification.

Table 53.12 Estimates of Ossification Probabilities

<table>
<thead>
<tr>
<th>PHT</th>
<th>TCPO</th>
<th>( \hat{\eta} )</th>
<th>( \hat{\pi} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>-1.2194 + 0.9129 + 0.3295 + 0.6162 = 0.6392</td>
<td>0.6546</td>
</tr>
<tr>
<td>60</td>
<td>0</td>
<td>-1.2194 + 0.3295 = -0.8899</td>
<td>0.2911</td>
</tr>
<tr>
<td>0</td>
<td>100</td>
<td>-1.2194 + 0.9129 = -0.3065</td>
<td>0.4240</td>
</tr>
<tr>
<td>60</td>
<td>100</td>
<td>-1.2194</td>
<td>0.2280</td>
</tr>
</tbody>
</table>

Morel and Neerchal (1997) considered a model in which the intralitter effects also depend on the treatments. This model is fit with the HPFMM procedure with the following statements:

```sas
proc hpfmm data=ossi;
  class pht tcpo;
  model y/m = pht tcpo pht*tcpo / dist=binomcluster;
  probmodel pht tcpo pht*tcpo;
run;
```

The –2 log likelihood of this model is much reduced compared to the previous model with constant conditional probability (compare 287.8 in Output 53.1.4 with 305.1 in Output 53.1.2). The likelihood-ratio statistic of 17.3 is significant, \( Pr(\chi^2_3 > 17.3 = 0.0006) \). Varying the conditional probabilities by treatment improved the model fit significantly.
Example 53.1: Modeling Mixing Probabilities: All Mice Are Equal, but Some Mice Are More Equal Than Others

Output 53.1.4  Fit Statistics and Parameter Estimates in Binomial Cluster Model

The HPFMM Procedure

<table>
<thead>
<tr>
<th>Fit Statistics</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>-2 Log Likelihood</td>
<td>287.8</td>
</tr>
<tr>
<td>AIC (Smaller is Better)</td>
<td>303.8</td>
</tr>
<tr>
<td>AICC (Smaller is Better)</td>
<td>305.8</td>
</tr>
<tr>
<td>BIC (Smaller is Better)</td>
<td>323.0</td>
</tr>
<tr>
<td>Pearson Statistic</td>
<td>85.5998</td>
</tr>
<tr>
<td>Effective Parameters</td>
<td>8</td>
</tr>
<tr>
<td>Effective Components</td>
<td>2</td>
</tr>
</tbody>
</table>

Table 53.13 computes the conditional probabilities in the four treatment groups. Recall that the previous model estimated a constant clustering probability of 0.5831.

Table 53.13  Estimates of Clustering Probabilities

<table>
<thead>
<tr>
<th>PHT</th>
<th>TCPO</th>
<th>(\hat{\eta})</th>
<th>(\hat{\mu})</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>1.8213 – 1.4962 – 3.1828 + 3.3736 = 0.5159</td>
<td>0.6262</td>
</tr>
<tr>
<td>60</td>
<td>0</td>
<td>1.8213 – 3.1828 = -1.3615</td>
<td>0.2040</td>
</tr>
<tr>
<td>0</td>
<td>100</td>
<td>1.8213 – 1.4962 = 0.3251</td>
<td>0.5806</td>
</tr>
<tr>
<td>60</td>
<td>100</td>
<td>1.8213</td>
<td>0.8607</td>
</tr>
</tbody>
</table>
The presence of phenytoin alone reduces the probability of response clustering within the litter. The presence of trichloropropene oxide alone does not have a strong effect on the clustering. The simultaneous presence of both agents substantially increases the probability of clustering.

The following statements fit the binomial cluster model in the parameterization of Morel and Neerchal (1997).

```
proc hpfmm data=ossi;
   model y/m = x1-x3 / dist=binomcluster;
   probmodel  x1-x3;
run;
```

The model fit is the same as in the previous model (compare the “Fit Statistics” tables in Output 53.1.5 and Output 53.1.4). The parameter estimates change due to the reparameterization of the treatment effects and match the results in Table III of Morel and Neerchal (1997).

**Output 53.1.5** Fit Statistics and Estimates (Morel and Neerchal Parameterization)

<table>
<thead>
<tr>
<th>Fit Statistics</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>-2 Log Likelihood</td>
<td>287.8</td>
</tr>
<tr>
<td>AIC (Smaller is Better)</td>
<td>303.8</td>
</tr>
<tr>
<td>AICC (Smaller is Better)</td>
<td>305.8</td>
</tr>
<tr>
<td>BIC (Smaller is Better)</td>
<td>323.0</td>
</tr>
<tr>
<td>Pearson Statistic</td>
<td>85.5999</td>
</tr>
<tr>
<td>Effective Parameters</td>
<td>8</td>
</tr>
<tr>
<td>Effective Components</td>
<td>2</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Parameter Estimates for Binomial Cluster Model</th>
</tr>
</thead>
<tbody>
<tr>
<td>Component Effect</td>
</tr>
<tr>
<td>1 Intercept</td>
</tr>
<tr>
<td>1 x1</td>
</tr>
<tr>
<td>1 x2</td>
</tr>
<tr>
<td>1 x3</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Parameter Estimates for Mixing Probabilities</th>
</tr>
</thead>
<tbody>
<tr>
<td>Component Effect</td>
</tr>
<tr>
<td>1 Intercept</td>
</tr>
<tr>
<td>1 x1</td>
</tr>
<tr>
<td>1 x2</td>
</tr>
<tr>
<td>1 x3</td>
</tr>
</tbody>
</table>

The following sets of statements fit the binomial and beta-binomial models, respectively, as single-component mixtures in the parameterization akin to the first binomial cluster model. Note that the model effects that affect the underlying Bernoulli success probabilities are specified in the **MODEL** statement, in contrast to the binomial cluster model.

```
proc hpfmm data=ossi;
   model y/m = x1-x3 / dist=binomial;
run;
```
Example 53.2: The Usefulness of Custom Starting Values: When Do Cows Eat?

This example with a mixture of normal and Weibull distributions illustrates the benefits of specifying starting values for some of the components.

The data for this example were generously provided by Dr. Luciano A. Gonzalez of the Lethbridge Research Center of Agriculture and Agri-Food Canada and his collaborator, Dr. Bert Tolkamp, from the Scottish Agricultural College.
The outcome variable of interest is the logarithm of a time interval between consecutive visits by cattle to feeders. The intervals fall into three categories:

- short breaks within meals—such as when an animal stops eating for a moment and resumes shortly thereafter
- somewhat longer breaks when eating is interrupted to go have a drink of water
- long breaks between meals

Modeling such time interval data is important to understand the feeding behavior and biology of the animals and to derive other biological parameters such as the probability of an animal to stop eating after it has consumed a certain amount of a given food. Because there are three distinct biological categories, data of this nature are frequently modeled as three-component mixtures. The point at which the second and third components cross over is used to separate feeding events into meals.

The original data set comprises 141,414 observations of log feeding intervals. For the purpose of presentation in this document, where space is limited, the data have been rounded to precision 0.05 and grouped by frequency. The following DATA step displays the modified data used in this example. A comparison with the raw data and the results obtained in a full analysis of the original data show that the grouping does not alter the presentation or conclusions in a way that matters for the purpose of this example.

```
data cattle;
  input LogInt Count @@;
datalines;
0.70 195 1.10 233 1.40 355 1.60 563
1.80 822 1.95 926 2.10 1018 2.20 1712
2.30 3190 2.40 2212 2.50 1692 2.55 1558
2.65 1622 2.70 1637 2.75 1568 2.85 1599
2.90 1575 2.95 1526 3.00 1537 3.05 1561
3.10 1555 3.15 1427 3.20 2852 3.25 1396
3.30 1343 3.35 2473 3.40 1310 3.45 2453
3.50 1168 3.55 2300 3.60 2174 3.65 2050
3.70 1926 3.75 1849 3.80 1687 3.85 2416
3.90 1449 3.95 2095 4.00 1278 4.05 1864
4.10 1672 4.15 2104 4.20 1443 4.25 1341
4.30 1685 4.35 1445 4.40 1369 4.45 1284
4.50 1523 4.55 1367 4.60 1027 4.65 1491
4.70 1057 4.75 1155 4.80 1095 4.85 1019
4.90 1158 4.95 1088 5.00 1075 5.05 912
5.10 1073 5.15 803 5.20 924 5.25 916
5.30 784 5.35 751 5.40 766 5.45 833
5.50 748 5.55 725 5.60 674 5.65 690
5.70 659 5.75 695 5.80 529 5.85 639
5.90 580 5.95 557 6.00 524 6.05 473
6.10 538 6.15 444 6.20 456 6.25 453
6.30 374 6.35 406 6.40 409 6.45 371
6.50 320 6.55 334 6.60 353 6.65 305
6.70 302 6.75 301 6.80 263 6.85 218
6.90 255 6.95 240 7.00 219 7.05 202
7.10 192 7.15 180 7.20 162 7.25 126
7.30 148 7.35 173 7.40 142 7.45 163
```
Example 53.2: The Usefulness of Custom Starting Values: When Do Cows Eat?

If you scan the columns for the Count variable in the DATA step, the prevalence of values between 2 and 5 units of LogInt is apparent, as is a long right tail. To explore these data graphically, the following statements produce a histogram of the data and a kernel density estimate of the density of the LogInt variable.

```plaintext
ods graphics on;
proc kde data=cattle;
   univar LogInt / bwm=4;
   freq count;
run;
```
Two modes are clearly visible in Output 53.2.1. Given the biological background, one would expect that three components contribute to the mixture. The histogram would suggest either a two-component mixture with modes near 4 and 9, or a three-component mixture with modes near 3, 5, and 9.

Following Dr. Gonzalez’ suggestion, the process is modeled as a three-component mixture of two normal distributions and a Weibull distribution. The Weibull distribution is chosen because it can have long left and right tails and it is popular in modeling data that relate to time intervals.

```
proc hpfmm data=cattle gconv=0;
   model LogInt = / dist=normal k=2 parms(3 1, 5 1);
   model + / dist=weibull;
   freq count;
run;
```

The GCONV= convergence criterion is turned off in this PROC HPFMM run to avoid the early stoppage of the iterations when the relative gradient changes little between iterations. Turning the criterion off usually ensures that convergence is achieved with a small absolute gradient of the objective function. The PARMS option in the first MODEL statement provides starting values for the means and variances for the parameters of the normal distributions. The means for the two components are started at $\mu = 3$ and $\mu = 5$, respectively. Specifying starting values is generally not necessary. However, the choice of starting values can play an
important role in modeling finite mixture models; the importance of the choice of starting values in this example is discussed further below.

The “Model Information” table shows that the model is a three-component mixture and that the HPFMM procedure considers the estimation of a density to be the purpose of modeling. The procedure draws this conclusion from the absence of effects in the MODEL statements. There are 187 observations in the data set, but these actually represent 141,414 measurements (Output 53.2.2).

**Output 53.2.2** Model Information and Number of Observations

<table>
<thead>
<tr>
<th>The HPFMM Procedure</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Model Information</strong></td>
</tr>
<tr>
<td>Data Set</td>
</tr>
<tr>
<td>Response Variable</td>
</tr>
<tr>
<td>Frequency Variable</td>
</tr>
<tr>
<td>Type of Model</td>
</tr>
<tr>
<td>Components</td>
</tr>
<tr>
<td>Estimation Method</td>
</tr>
<tr>
<td>Number of Observations Read</td>
</tr>
<tr>
<td>Number of Observations Used</td>
</tr>
<tr>
<td>Sum of Frequencies Read</td>
</tr>
<tr>
<td>Sum of Frequencies Used</td>
</tr>
</tbody>
</table>

There are eight parameters in the optimization: the means and variances of the two normal distributions, the μ and φ parameter of the Weibull distribution, and the two mixing probabilities (Output 53.2.3). At the converged solution, the −2 log likelihood is 563,153 and all parameters and components are effective—that is, the model is not overspecified in the sense that components have collapsed during the model fitting. The Pearson statistic is close to the number of observations in the data set, indicating a good fit.

**Output 53.2.3** Optimization Information and Fit Statistics

<table>
<thead>
<tr>
<th>Optimization Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Optimization Technique</td>
</tr>
<tr>
<td>Parameters in Optimization</td>
</tr>
<tr>
<td>Mean Function Parameters</td>
</tr>
<tr>
<td>Scale Parameters</td>
</tr>
<tr>
<td>Mixing Prob Parameters</td>
</tr>
<tr>
<td>Lower Boundaries</td>
</tr>
<tr>
<td>Upper Boundaries</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Fit Statistics</th>
</tr>
</thead>
<tbody>
<tr>
<td>-2 Log Likelihood</td>
</tr>
<tr>
<td>AIC (Smaller is Better)</td>
</tr>
<tr>
<td>AICC (Smaller is Better)</td>
</tr>
<tr>
<td>BIC (Smaller is Better)</td>
</tr>
<tr>
<td>Pearson Statistic</td>
</tr>
<tr>
<td>Effective Parameters</td>
</tr>
<tr>
<td>Effective Components</td>
</tr>
</tbody>
</table>
Output 53.2.4 displays the parameter estimates for the three models and for the mixing probabilities. The order in which the “Parameter Estimates” tables appear in the output corresponds to the order in which the MODEL statements were specified.

### Output 53.2.4 Optimization Information and Fit Statistics

| Component | Parameter | Estimate | Standard Error | z Value | Pr > |z| |
|-----------|-----------|----------|----------------|---------|------|---|
| 1         | Intercept | 3.3415   | 0.01260        | 265.16  | <.0001 |
| 2         | Intercept | 4.8940   | 0.05447        | 89.84   | <.0001 |
| 1         | Variance  | 0.6718   | 0.01287        |         |       |   |
| 2         | Variance  | 1.4497   | 0.05247        |         |       |   |

| Component | Parameter | Estimate | Standard Error | z Value | Pr > |z| | Inverse Linked Estimate |
|-----------|-----------|----------|----------------|---------|------|---|------------------------|
| 3         | Intercept | 2.2531   | 0.000506       | 4452.11 | <.0001 | 9.5174 |
| 3         | Scale     | 0.06848  | 0.000427       |         |       |   |

| Component | Mixing Probability | GLogit(Prob) | Standard Error | z Value | Pr > |z| |
|-----------|---------------------|--------------|----------------|---------|------|---|
| 1         | 0.4545              | 0.8106       | 0.03409        | 23.78   | <.0001 |
| 2         | 0.3435              | 0.5305       | 0.04640        | 11.43   | <.0001 |
| 3         | 0.2021              | 0            |               |         |       |   |

The estimated means of the two normal components are 3.3415 and 4.8940, respectively. Note that the means are displayed here as Intercept. The inverse linked estimate is not produced because the default link for the normal distribution is the identity link; hence the Estimate column represents the means of the component distributions. The parameter estimates in the Weibull model are \( \hat{\beta}_0 = 2.2531, \hat{\phi} = 0.06848, \) and \( \hat{\mu} = \exp(\hat{\beta}_0) = 9.5174. \) In the Weibull distribution, the \( \mu \) parameter does not estimate the mean of the distribution, the maximum likelihood estimate of the distribution’s mean is \( \mu = \Gamma(\hat{\phi} + 1) = 9.1828. \)

The estimated mixing probabilities are \( \hat{\pi}_1 = 0.4545, \hat{\pi}_2 = 0.3435, \) and \( \hat{\pi}_3 = 0.2021. \) In other words, the estimated distribution of log feeding intervals is a 45:35:20 mixture of an \( N(3.3415, 0.6718) \), a \( N(4.8940, 1.4497) \), and a Weibull(9.5174, 0.06848) distribution.

You can obtain a graphical display of the observed and estimated distribution of these data by enabling ODS Graphics. The PLOTS option in the PROC HPFMM statement modifies the default density plot by adding the densities of the mixture components:

```plaintext
ods select DensityPlot;
proc hpfmm data=cattle gconv=0;
   model LogInt = / dist=normal k=2 parms(3 1, 5 1);
   model _scale / dist=weibull;
   freq count;
run;
```
Example 53.2: The Usefulness of Custom Starting Values: When Do Cows Eat?

The estimated mixture density matches the histogram of the observed data closely (Output 53.2.5). The component densities are displayed in such a way that, at each point in the support of the LogInt variable, their sum combines to the overall mixture density. The three components in the mixtures are well separated.

The excellent quality of the fit is even more evident when the distributions are displayed cumulatively by adding the CUMULATIVE option in the DENSITY option (Output 53.2.6):

```
ods select DensityPlot;
proc hpfmm data=cattle plot=density(cumulative) gconv=0;
   model LogInt = / dist=normal k=2 parms(3 1, 5 1);
   model + / dist=weibull;
   freq count;
run;
```

The component cumulative distribution functions are again scaled so that their sum produces the overall mixture cumulative distribution function. Because of this scaling, the percentage reached at the maximum value of LogInt corresponds to the mixing probabilities in Output 53.2.4.
The importance of starting values for the parameter estimates was mentioned previously. Suppose that different starting values are selected for the three components (for example, the default starting values).

```sas
proc hpfmm data=cattle gconv=0;
   model LogInt = / dist=normal k=2;
   model LogInt = / dist=weibull;
   freq count;
run;
ods graphics off;
```

The fit statistics and parameter estimates from this run are displayed in Output 53.2.7, and the density plot is shown in Output 53.2.8.
All components are active; no collapsing of components occurred. However, a closer look at the “Parameter Estimates” tables in Output 53.2.7 shows an important difference from the tables in Output 53.2.4. The means of the two normal distributions are now 4.9106 and 9.2883. Previously, the means were 3.3415 and 4.8940. The “position” of the Weibull distribution has moved from right to left, and the third component is now modeled by a symmetric normal distribution (Output 53.2.8). The mixture probabilities have also changed—in particular, for the first and third component.
Such switching is not uncommon in mixture modeling. As judged by the information criteria, the model in which the Weibull distribution is the component with the smallest mean does not fit the data as well as the first model in which the specification of the starting values guided the optimization towards placing the normal distributions first. The converged solution found in the last HPFMM run represents a local minimum of the log-likelihood surface. There are other local minima—for example, when components are removed from the model, which is tantamount to estimating the associated mixture probabilities as zero.

**Example 53.3: Enforcing Homogeneity Constraints: Count and Dispersion—It Is All Over!**

The following example demonstrates how you can use either the EQUATE= option in the MODEL statement or the RESTRICT statement to impose homogeneity constraints on chosen model effects.

The data for this example were presented by Margolin, Kaplan, and Zeiger (1981) and analyzed by various authors applying a number of techniques. The following DATA step shows the number of revertant salmonella colonies (variable num) at six levels of quinoline dosing (variable dose). There are three replicate plates at each dose of quinoline.
Example 53.3: Enforcing Homogeneity Constraints: Count and Dispersion—It Is All Over!

The basic notion is that the data are overdispersed relative to a Poisson distribution in which the logarithm of the mean count is modeled as a linear regression in dose (in \( \mu g/\text{plate} \)) and in the derived variable \( \log(dose + 10) \) (Lawless 1987). The log of the expected count of revertants is thus

\[
\beta_0 + \beta_1 \text{dose} + \beta_2 \log(dose + 10)
\]

The following statements fit a standard Poisson regression model to these data:

```sas
proc hpfmm data=assay;
    model num = dose logd / dist=Poisson;
run;
```

The Pearson statistic for this model is rather large compared to the number of degrees of freedom (18 - 3 = 15). The ratio 46.2707/15 = 3.08 indicates an overdispersion problem in the Poisson model (Output 53.3.1).

**Output 53.3.1** Result of Fitting Poisson Regression Models

<table>
<thead>
<tr>
<th>The HPFMM Procedure</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of Observations Read</td>
</tr>
<tr>
<td>Number of Observations Used</td>
</tr>
<tr>
<td>Fit Statistics</td>
</tr>
<tr>
<td>-2 Log Likelihood</td>
</tr>
<tr>
<td>AIC (Smaller is Better)</td>
</tr>
<tr>
<td>AICC (Smaller is Better)</td>
</tr>
<tr>
<td>BIC (Smaller is Better)</td>
</tr>
<tr>
<td>Pearson Statistic</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Parameter Estimates for Poisson Model</th>
</tr>
</thead>
<tbody>
<tr>
<td>Effect</td>
</tr>
<tr>
<td>--------</td>
</tr>
<tr>
<td>Intercept</td>
</tr>
<tr>
<td>dose</td>
</tr>
<tr>
<td>logd</td>
</tr>
</tbody>
</table>
Breslow (1984) accounts for overdispersion by including a random effect in the predictor for the log rate and applying a quasi-likelihood technique to estimate the parameters. Wang et al. (1996) examine these data using mixtures of Poisson regression models. They fit several two- and three-component Poisson regression mixtures. Examining the log likelihoods, AIC, and BIC criteria, they eventually settle on a two-component model in which the intercepts vary by category and the regression coefficients are the same. This mixture model can be written as

\[
\begin{align*}
  f(y) &= \pi \frac{1}{y!} \lambda_1^y \exp\{-\lambda_1\} + (1 - \pi) \frac{1}{y!} \lambda_2^y \exp\{-\lambda_2\} \\
  \lambda_1 &= \exp\{\beta_0 + \beta_1 \text{dose} + \beta_2 \log[\text{dose} + 10]\} \\
  \lambda_2 &= \exp\{\beta_0 + \beta_1 \text{dose} + \beta_2 \log[\text{dose} + 10]\}
\end{align*}
\]

This model is fit with the HPFMM procedure with the following statements:

```plaintext
proc hpfmm data=assay;
  model num = dose logd / dist=Poisson k=2
              equate=effects(dose logd);
run;
```

The `EQUATE=` option in the `MODEL` statement places constraints on the optimization and makes the coefficients for `dose` and `logd` homogeneous across components in the model. Output 53.3.2 displays the “Fit Statistics” and parameter estimates in the mixture. The Pearson statistic is drastically reduced compared to the Poisson regression model in Output 53.3.1. With \(18 - 5 = 13\) degrees of freedom, the ratio of the Pearson and the degrees of freedom is now \(16.1573/13 = 1.2429\). Note that the effective number of parameters was used to compute the degrees of freedom, not the total number of parameters, because of the equality constraints.

**Output 53.3.2** Result for Two-Component Poisson Regression Mixture

<table>
<thead>
<tr>
<th>Fit Statistics</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>-2 Log Likelihood</td>
<td>121.8</td>
</tr>
<tr>
<td>AIC (Smaller is Better)</td>
<td>131.8</td>
</tr>
<tr>
<td>AICC (Smaller is Better)</td>
<td>136.8</td>
</tr>
<tr>
<td>BIC (Smaller is Better)</td>
<td>136.3</td>
</tr>
<tr>
<td>Pearson Statistic</td>
<td>16.1573</td>
</tr>
<tr>
<td>Effective Parameters</td>
<td>5</td>
</tr>
<tr>
<td>Effective Components</td>
<td>2</td>
</tr>
</tbody>
</table>

| Parameter Estimates for Poisson Model | Standard Error | z Value | Pr > |z| |
|--------------------------------------|----------------|---------|------|-----|
| Component Effect | Estimate     |         |      |     |
| 1 Intercept      | 1.9097        | 0.2654  | 7.20 | <.0001 |
| 1 dose           | -0.00126      | 0.000273| -4.62| <.0001 |
| 1 logd           | 0.3639        | 0.06602 | 5.51 | <.0001 |
| 2 Intercept      | 2.4770        | 0.2731  | 9.07 | <.0001 |
| 2 dose           | -0.00126      | 0.000273| -4.62| <.0001 |
| 2 logd           | 0.3639        | 0.06602 | 5.51 | <.0001 |
You could also have used RESTRICT statements to impose the homogeneity constraints on the model fit, as shown in the following statements:

```plaintext
proc hpfmm data=assay;
  model num = dose logd / dist=Poisson k=2;
  restrict 'common dose' dose 1, dose -1;
  restrict 'common logd' logd 1, logd -1;
run;
```

The first RESTRICT statement equates the coefficients for the dose variable in the two components, and the second RESTRICT statement accomplishes the same for the coefficients of the logd variable. If the right-hand side of a restriction is not specified, PROC HPFMM defaults to equating the left-hand side of the restriction to zero. The “Linear Constraints” table in Output 53.3.3 shows that both linear equality constraints are active. The parameter estimates match the previous HPFMM run.

### Output 53.3.3 Result for Two-Component Mixture with RESTRICT Statements

#### The HPFMM Procedure

<table>
<thead>
<tr>
<th>Linear Constraints at Solution</th>
<th>Constraint Active</th>
</tr>
</thead>
<tbody>
<tr>
<td>Label</td>
<td>k = 1  k = 2</td>
</tr>
<tr>
<td>common dose</td>
<td>dose - dose = 0</td>
</tr>
<tr>
<td>common logd</td>
<td>logd - logd = 0</td>
</tr>
</tbody>
</table>

#### Parameter Estimates for Poisson Model

| Component | Effect | Estimate | Standard Error | z Value | Pr > |z| |
|-----------|--------|----------|----------------|---------|------|---|
| 1 Intercept | 1.9097 | 0.2654   | 7.20           | <.0001  |
| 1 dose     | -0.00126 | 0.000273 | -4.62          | <.0001  |
| 1 logd     | 0.3639  | 0.06602  | 5.51           | <.0001  |
| 2 Intercept | 2.4770 | 0.2731   | 9.07           | <.0001  |
| 2 dose     | -0.00126 | 0.000273 | -4.62          | <.0001  |
| 2 logd     | 0.3639  | 0.06602  | 5.51           | <.0001  |

#### Parameter Estimates for Mixing Probabilities

| Mixing Component Probability | Logit(Prob) | Standard Error | z Value | Pr > |z| |
|-------------------------------|-------------|----------------|---------|------|---|
| 1                             | 0.8173      | 1.4984         | 0.6875  | 2.18 | 0.0293 |
| 2                             | 0.1827      | -1.4984        |         |      |      |
Wang et al. (1996) note that observation 12 with a revertant colony count of 60 is comparably high. The following statements remove the observation from the analysis and fit their selected model:

```sas
proc hpfmm data=assay(where=(num ne 60));
    model num = dose logd / dist=Poisson k=2
equate=effects(dose logd);
run;
```

**Output 53.3.4** Result for Two-Component Model without Outlier

**The HPFMM Procedure**

<table>
<thead>
<tr>
<th>Fit Statistics</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>-2 Log Likelihood</td>
<td>111.5</td>
</tr>
<tr>
<td>AIC (Smaller is Better)</td>
<td>121.5</td>
</tr>
<tr>
<td>AICCC (Smaller is Better)</td>
<td>126.9</td>
</tr>
<tr>
<td>BIC (Smaller is Better)</td>
<td>125.6</td>
</tr>
<tr>
<td>Pearson Statistic</td>
<td>16.5987</td>
</tr>
<tr>
<td>Effective Parameters</td>
<td>5</td>
</tr>
<tr>
<td>Effective Components</td>
<td>2</td>
</tr>
</tbody>
</table>

**Parameter Estimates for Poisson Model**

<table>
<thead>
<tr>
<th>Component</th>
<th>Effect</th>
<th>Estimate</th>
<th>Standard Error</th>
<th>z Value</th>
<th>Pr &gt;</th>
<th>z</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Intercept</td>
<td>2.2272</td>
<td>0.3022</td>
<td>7.37</td>
<td>&lt;0.001</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>dose</td>
<td>-0.00065</td>
<td>0.000445</td>
<td>-1.46</td>
<td>0.1440</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>logd</td>
<td>0.2432</td>
<td>0.1045</td>
<td>2.33</td>
<td>0.0199</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>Intercept</td>
<td>2.5477</td>
<td>0.3331</td>
<td>7.65</td>
<td>&lt;0.001</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>dose</td>
<td>-0.00065</td>
<td>0.000445</td>
<td>-1.46</td>
<td>0.1440</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>logd</td>
<td>0.2432</td>
<td>0.1045</td>
<td>2.33</td>
<td>0.0199</td>
<td></td>
</tr>
</tbody>
</table>

**Parameter Estimates for Mixing Probabilities**

<table>
<thead>
<tr>
<th>Component</th>
<th>Mixing Probability</th>
<th>Linked Scale</th>
<th>Linked Scale</th>
<th>Linked Scale</th>
<th>Linked Scale</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.5777</td>
<td>0.3134</td>
<td>1.7261</td>
<td>0.18</td>
<td>0.8559</td>
</tr>
<tr>
<td>2</td>
<td>0.4223</td>
<td>-0.3134</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

The ratio of Pearson Statistic over degrees of freedom (12) is only slightly worse than in the previous model; the loss of 5% of the observations carries a price (**Output 53.3.4**). The parameter estimates for the two intercepts are now fairly close. If the intercepts were identical, then the two-component model would collapse to the Poisson regression model:

```sas
proc hpfmm data=assay(where=(num ne 60));
    model num = dose logd / dist=Poisson;
run;
```
Output 53.3.5 Result of Fitting Poisson Regression Model without Outlier

The HPFMM Procedure

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of Observations Read</td>
<td>17</td>
</tr>
<tr>
<td>Number of Observations Used</td>
<td>17</td>
</tr>
</tbody>
</table>

Fit Statistics

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>-2 Log Likelihood</td>
<td>114.1</td>
</tr>
<tr>
<td>AIC (Smaller is Better)</td>
<td>120.1</td>
</tr>
<tr>
<td>AICC (Smaller is Better)</td>
<td>121.9</td>
</tr>
<tr>
<td>BIC (Smaller is Better)</td>
<td>122.5</td>
</tr>
<tr>
<td>Pearson Statistic</td>
<td>27.8008</td>
</tr>
</tbody>
</table>

Parameter Estimates for Poisson Model

<table>
<thead>
<tr>
<th>Effect</th>
<th>Estimate</th>
<th>Standard Error</th>
<th>z Value</th>
<th>Pr &gt;</th>
<th>z</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Intercept</td>
<td>2.3164</td>
<td>0.2244</td>
<td>10.32</td>
<td>&lt;.0001</td>
<td></td>
<td></td>
</tr>
<tr>
<td>dose</td>
<td>-0.00072</td>
<td>0.000258</td>
<td>-2.78</td>
<td>0.0055</td>
<td></td>
<td></td>
</tr>
<tr>
<td>logd</td>
<td>0.2603</td>
<td>0.05996</td>
<td>4.34</td>
<td>&lt;.0001</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Compared to the same model applied to the full data, the Pearson statistic is much reduced (compare 46.2707 in Output 53.3.1 to 27.8008 in Output 53.3.5). The outlier—or overcount, if you will—induces at least some of the overdispersion.

References


### Subject Index

<table>
<thead>
<tr>
<th>Term</th>
<th>Page Numbers</th>
</tr>
</thead>
<tbody>
<tr>
<td>alpha level</td>
<td>4234, 4242</td>
</tr>
<tr>
<td>HPFMM procedure</td>
<td></td>
</tr>
<tr>
<td>Bayes information</td>
<td>4268</td>
</tr>
<tr>
<td>Bayesian analysis</td>
<td>4220</td>
</tr>
<tr>
<td>Bayesian fit statistics</td>
<td>4268</td>
</tr>
<tr>
<td>Bernoulli distribution</td>
<td>4235</td>
</tr>
<tr>
<td>HPFMM procedure</td>
<td></td>
</tr>
<tr>
<td>beta distribution</td>
<td>4235</td>
</tr>
<tr>
<td>HPFMM procedure</td>
<td></td>
</tr>
<tr>
<td>beta-binomial distribution</td>
<td>4235</td>
</tr>
<tr>
<td>HPFMM procedure</td>
<td></td>
</tr>
<tr>
<td>binary distribution</td>
<td>4235</td>
</tr>
<tr>
<td>HPFMM procedure</td>
<td></td>
</tr>
<tr>
<td>binomial distribution</td>
<td>4235</td>
</tr>
<tr>
<td>HPFMM procedure</td>
<td></td>
</tr>
<tr>
<td>class level</td>
<td>4215, 4265</td>
</tr>
<tr>
<td>HPFMM procedure</td>
<td></td>
</tr>
<tr>
<td>computational method</td>
<td>4260</td>
</tr>
<tr>
<td>HPFMM procedure</td>
<td></td>
</tr>
<tr>
<td>confidence limits</td>
<td></td>
</tr>
<tr>
<td>model parameters (HPFMM), 4234, 4243</td>
<td></td>
</tr>
<tr>
<td>constrained analysis</td>
<td>4244</td>
</tr>
<tr>
<td>HPFMM procedure</td>
<td></td>
</tr>
<tr>
<td>convergence criterion</td>
<td>4210, 4213</td>
</tr>
<tr>
<td>HPFMM procedure</td>
<td></td>
</tr>
<tr>
<td>convergence status</td>
<td>4266</td>
</tr>
<tr>
<td>HPFMM procedure</td>
<td></td>
</tr>
<tr>
<td>default output</td>
<td>4265</td>
</tr>
<tr>
<td>HPFMM procedure</td>
<td></td>
</tr>
<tr>
<td>effect</td>
<td></td>
</tr>
<tr>
<td>name length (HPFMM), 4214</td>
<td></td>
</tr>
<tr>
<td>examples, HPFMM</td>
<td></td>
</tr>
<tr>
<td>binomial data, 4273</td>
<td></td>
</tr>
<tr>
<td>cattle feeding data, 4280</td>
<td></td>
</tr>
<tr>
<td>logistic model, binomial cluster, 4273</td>
<td></td>
</tr>
<tr>
<td>ossification data, 4273</td>
<td></td>
</tr>
<tr>
<td>PROBMODEL specification, 4273</td>
<td></td>
</tr>
<tr>
<td>salmonella assay, 4288</td>
<td></td>
</tr>
<tr>
<td>three-component mixture, 4282</td>
<td></td>
</tr>
<tr>
<td>Weibull distribution, 4282</td>
<td></td>
</tr>
<tr>
<td>exponential distribution</td>
<td>4235</td>
</tr>
<tr>
<td>HPFMM procedure</td>
<td></td>
</tr>
<tr>
<td>finite mixture models</td>
<td>4180</td>
</tr>
<tr>
<td>See HPFMM procedure, 4236, 4249</td>
<td></td>
</tr>
<tr>
<td>exponential distribution</td>
<td>4235</td>
</tr>
<tr>
<td>HPFMM procedure</td>
<td></td>
</tr>
<tr>
<td>fit statistics</td>
<td>4266</td>
</tr>
<tr>
<td>HPFMM procedure</td>
<td></td>
</tr>
<tr>
<td>folded normal distribution</td>
<td>4235</td>
</tr>
<tr>
<td>HPFMM procedure</td>
<td></td>
</tr>
<tr>
<td>frequency variable</td>
<td>4231</td>
</tr>
<tr>
<td>HPFMM procedure</td>
<td></td>
</tr>
<tr>
<td>gamma distribution</td>
<td>4235</td>
</tr>
<tr>
<td>HPFMM procedure</td>
<td></td>
</tr>
<tr>
<td>Gaussian distribution</td>
<td>4235</td>
</tr>
<tr>
<td>HPFMM procedure</td>
<td></td>
</tr>
<tr>
<td>generalized Poisson distribution</td>
<td>4235</td>
</tr>
<tr>
<td>HPFMM procedure</td>
<td></td>
</tr>
<tr>
<td>geometric distribution</td>
<td>4235</td>
</tr>
<tr>
<td>HPFMM procedure</td>
<td></td>
</tr>
<tr>
<td>heavy-tailed density</td>
<td></td>
</tr>
<tr>
<td>finite mixture models (HPFMM), 4236, 4249</td>
<td></td>
</tr>
<tr>
<td>HPFMM procedure, 4180</td>
<td></td>
</tr>
<tr>
<td>alpha level, 4234, 4242</td>
<td></td>
</tr>
<tr>
<td>Bayes information, 4268</td>
<td></td>
</tr>
<tr>
<td>Bayesian analysis, 4220</td>
<td></td>
</tr>
<tr>
<td>Bayesian fit statistics, 4268</td>
<td></td>
</tr>
<tr>
<td>Bernoulli distribution, 4235</td>
<td></td>
</tr>
<tr>
<td>beta distribution, 4235</td>
<td></td>
</tr>
<tr>
<td>HPFMM procedure, 4235</td>
<td></td>
</tr>
<tr>
<td>beta-binomial distribution, 4235</td>
<td></td>
</tr>
<tr>
<td>HPFMM procedure, 4235</td>
<td></td>
</tr>
<tr>
<td>binary distribution, 4235</td>
<td></td>
</tr>
<tr>
<td>HPFMM procedure, 4235</td>
<td></td>
</tr>
<tr>
<td>binomial distribution, 4235</td>
<td></td>
</tr>
<tr>
<td>HPFMM procedure, 4266</td>
<td></td>
</tr>
<tr>
<td>centering and scaling, 4214</td>
<td></td>
</tr>
<tr>
<td>class level, 4215, 4265</td>
<td></td>
</tr>
<tr>
<td>computational method, 4260</td>
<td></td>
</tr>
<tr>
<td>confidence limits, 4234, 4243</td>
<td></td>
</tr>
<tr>
<td>constrained analysis, 4244</td>
<td></td>
</tr>
<tr>
<td>HPFMM procedure, 4210, 4213</td>
<td></td>
</tr>
<tr>
<td>convergence criterion, 4210, 4213</td>
<td></td>
</tr>
<tr>
<td>convergence status, 4266</td>
<td></td>
</tr>
<tr>
<td>default output, 4265</td>
<td></td>
</tr>
<tr>
<td>effect name length, 4214</td>
<td></td>
</tr>
<tr>
<td>exponential distribution, 4235</td>
<td></td>
</tr>
<tr>
<td>fit statistics, 4266</td>
<td></td>
</tr>
<tr>
<td>folded normal distribution, 4235</td>
<td></td>
</tr>
<tr>
<td>function-based convergence criteria, 4210, 4212</td>
<td></td>
</tr>
<tr>
<td>gamma distribution, 4235</td>
<td></td>
</tr>
<tr>
<td>Gaussian distribution, 4235</td>
<td></td>
</tr>
<tr>
<td>generalized Poisson distribution, 4235</td>
<td></td>
</tr>
<tr>
<td>geometric distribution, 4235</td>
<td></td>
</tr>
<tr>
<td>gradient-based convergence criteria, 4210, 4213</td>
<td></td>
</tr>
<tr>
<td>heavy-tailed density, 4236, 4249</td>
<td></td>
</tr>
</tbody>
</table>
HPFMM procedure, 4269
  posterior summaries
  HPFMM procedure, 4268
  prior distributions
  HPFMM procedure, 4268
  probability distributions
  HPFMM procedure, 4235
random number seed
  HPFMM procedure, 4219
response level ordering
  HPFMM procedure, 4233
response profile
  HPFMM procedure, 4266
response variable options
  HPFMM procedure, 4233
restricted analysis
  HPFMM procedure, 4244
reverse response level ordering
  HPFMM procedure, 4233
statistical graphics
  HPFMM procedure, 4271
t distribution
  HPFMM procedure, 4235
Weibull distribution
  HPFMM procedure, 4235
weighting
  HPFMM procedure, 4246
zero-inflated model
  finite mixture models (HPFMM), 4181, 4184,
  4190, 4191, 4236, 4251
Syntax Index

ABSCONV option
   PROC HPFMM statement, 4210
ABSFCONV option
   PROC HPFMM statement, 4210
ABSGCONV option
   PROC HPFMM statement, 4210
ABSGTOL option
   PROC HPFMM statement, 4210
ABSTOL option
   PROC HPFMM statement, 4210
ALPHA= option
   MODEL statement (HPFMM), 4234
PROBMODEL statement (HPFMM), 4242
BAYES statement
   HPFMM procedure, 4220
BETAPRIORPARMS option
   BAYES statement (HPFMM), 4221
BY statement
   HPFMM procedure, 4230
CINFO option
   PROC HPFMM statement, 4210
CL option
   MODEL statement (HPFMM), 4234
PROBMODEL statement (HPFMM), 4243
CLASS statement
   HPFMM procedure, 4230
COMPINFO option
   PROC HPFMM statement, 4210
COMPONENTINFO option
   PROC HPFMM statement, 4210
CORR option
   PROC HPFMM statement, 4211
COV option
   PROC HPFMM statement, 4211
COVI option
   PROC HPFMM statement, 4211
CRIT= option
   PROC HPFMM statement, 4211
CRITERION= option
   PROC HPFMM statement, 4211
DATA= option
   PROC HPFMM statement, 4211
DESCENDING option
   MODEL statement, 4233
DIAGNOSTICS option
   BAYES statement (HPFMM), 4222
DIST= option
   MODEL statement (HPFMM), 4235
DISTRIBUTION= option
   MODEL statement (HPFMM), 4235
DIVISOR= option
   RESTRICT statement (HPFMM), 4246
EQUATE= option
   MODEL statement (HPFMM), 4237
ESTIMATE= option
   BAYES statement (HPFMM), 4224
EVENT= option
   MODEL statement, 4233
EXCLUDE= option
   PROC HPFMM statement, 4211
EXCLUSION= option
   PROC HPFMM statement, 4211
FCONV option
   PROC HPFMM statement, 4212
FCONV2 option
   PROC HPFMM statement, 4212
FITDETAILS option
   PROC HPFMM statement, 4213
FREQ statement
   HPFMM procedure, 4231
FTOL option
   PROC HPFMM statement, 4212
FTOL2 option
   PROC HPFMM statement, 4212
GCONV option
   PROC HPFMM statement, 4213
GTOL option
   PROC HPFMM statement, 4213
HESSIAN option
   PROC HPFMM statement, 4213
HPFMM procedure, 4208
   BAYES statement, 4220
   FREQ statement, 4231
   ID statement, 4231
   MODEL statement, 4231
   OUTPUT statement, 4239
   PERFORMANCE statement, 4242
   PROBMODEL statement, 4242
   PROC HPFMM statement, 4208
RESTRICT statement, 4244
syntax, 4208
WEIGHT statement, 4246
HPFMM procedure, BAYES statement, 4220
   BETAPRIORPARMS option, 4221
   DIAGNOSTICS option, 4222
   ESTIMATE= option, 4224
   INITIAL= option, 4224
   METROPOLIS option, 4225
   MIXPRIORPARMS option, 4224
   MUPRIORPARMS option, 4225
   NBI= option, 4226
   NMC= option, 4226
   OUTPOST= option, 4226
   PHIPRIORPARMS option, 4226
   PRIOROPTIONS option, 4227
   PRIOROPTS option, 4227
   STATISTICS option, 4228
   SUMMARIES option, 4228
   THIN= option, 4229
   THINNING= option, 4229
   TIMEINC= option, 4229
HPFMM procedure, BY statement, 4230
HPFMM procedure, CLASS statement, 4230
   UPCASE option, 4230
HPFMM procedure, FREQ statement, 4231
HPFMM procedure, ID statement, 4231
HPFMM procedure, MODEL statement, 4231
   ALPHA= option, 4234
   CL option, 4234
   DESCENDING option, 4233
   DIST= option, 4235
   DISTRIBUTION= option, 4235
   EQUATE= option, 4237
   EVENT= option, 4233
   K= option, 4237
   KMAX= option, 4237
   KMIN= option, 4238
   KRESTART option, 4238
   LABEL= option, 4238
   LINK= option, 4238
   NOINT option, 4238
   NUMBER= option, 4237
   OFFSET= option, 4239
   ORDER= option, 4233
   PARAMETERS option, 4239
   PARMSTYLE= option, 4239
   REFERENCE= option, 4234
   ALLSTATS option, 4241
   keyword= option, 4239
   OUT= option, 4239
   PREDTYPE option, 4241
HPFMM procedure, PERFORMANCE statement, 4242
HPFMM procedure, PROBMODEL statement, 4242
   ALPHA= option, 4242
   CL option, 4243
   LINK= option, 4243
   NOINT option, 4243
   PARAMETERS option, 4243
   PARMSTYLE= option, 4243
HPFMM procedure, PROC HPFMM statement, 4208
   ABSCONV option, 4210
   ABSFCONV option, 4210
   ABSFTOL option, 4210
   ABSGCONV option, 4210
   ABSGTOl option, 4210
   ABSTOL option, 4210
   CINFO option, 4210
   COMPINFO option, 4210
   COMPONENTINFO option, 4210
   CORR option, 4211
   COV option, 4211
   COV option, 4211
   CRIT= option, 4211
   CRITERION= option, 4211
   DATA= option, 4211
   EXCLUDE= option, 4211
   EXCLUSION= option, 4211
   FCONV option, 4212
   FCNV2 option, 4212
   FITDETAILS option, 4213
   FTOL option, 4212
   FTOL2 option, 4212
   GCONV option, 4213
   GTOL option, 4213
   HESSIAN option, 4213
   INVALIDDLOGL= option, 4213
   ITDETAILS option, 4213
   MAXFUNC= option, 4214
   MAXITER= option, 4214
   MAXTIME= option, 4214
   MEMBERSHIP= option, 4215
   NAMELEN= option, 4214
   NOCENTER option, 4214
   NOCLPRINT option, 4215
   NOITPRINT option, 4215
   NOPRINT option, 4215
   PARMASTYLE= option, 4215
   PARTIAL= option, 4215
   PLOTS option, 4216
   SEED= option, 4219
   SINGCHOL= option, 4219
   SINGULAR= option, 4219
   TECHNIQUE= option, 4219
   ZEROPROB= option, 4220
HPFMM procedure, RESTRICT statement, 4244
DIVISOR= option, 4246
HPFMM procedure, WEIGHT statement, 4246
ID statement
  HPFMM procedure, 4231
  INITIAL= option
    BAYES statement (HPFMM), 4224
  INVALIDLOGL= option
    PROC HPFMM statement, 4213
  ITDETAILS option
    PROC HPFMM statement, 4213
K= option
  MODEL statement (HPFMM), 4237
  keyword= option
    OUTPUT statement (HPFMM), 4239
KMAX= option
  MODEL statement (HPFMM), 4237
KMIN= option
  MODEL statement (HPFMM), 4238
KRESTART option
  MODEL statement (HPFMM), 4238
LABEL= option
  MODEL statement (HPFMM), 4238
LINK= option
  MODEL statement (HPFMM), 4238
  PROBMODEL statement (HPFMM), 4243
MAXFUNC= option
  PROC HPFMM statement, 4214
MAXITER= option
  PROC HPFMM statement, 4214
MAXTIME= option
  PROC HPFMM statement, 4214
MEMBERSHIP= option
  PROC HPFMM statement, 4214
METROPOLIS option
  BAYES statement (HPFMM), 4225
MIXPRIORPARMS option
  BAYES statement (HPFMM), 4224
MODEL statement
  HPFMM procedure, 4231
MUPRIORPARMS option
  BAYES statement (HPFMM), 4225
NAMELEN= option
  PROC HPFMM statement, 4214
NBI= option
  BAYES statement (HPFMM), 4226
NMC= option
  BAYES statement (HPFMM), 4226
NOCENTER option
  PROC HPFMM statement, 4214
  NOCLPRINT option
    PROC HPFMM statement, 4215
NOINT option
  MODEL statement (HPFMM), 4238
  PROBMODEL statement (HPFMM), 4243
NOITPRINT option
  PROC HPFMM statement, 4215
NOPRINT option
  PROC HPFMM statement, 4215
NUMBER= option
  MODEL statement (HPFMM), 4237
OFFSET= option
  MODEL statement (HPFMM), 4239
ORDER= option
  MODEL statement, 4233
OUT= option
  OUTPUT statement (HPFMM), 4239
OUTPOST= option
  BAYES statement (HPFMM), 4226
OUTPUT statement
  HPFMM procedure, 4239
PARAMETERS option
  MODEL statement (HPFMM), 4239
  PROBMODEL statement (HPFMM), 4243
PARMS option
  MODEL statement (HPFMM), 4239
  PROBMODEL statement (HPFMM), 4243
PARMSTYLE= option
  PROC HPFMM statement, 4215
PARTIAL= option
  PROC HPFMM statement, 4215
PERFORMANCE statement
  HPFMM procedure, 4242
PHIPRIORPARMS option
  BAYES statement (HPFMM), 4226
PLOTS option
  PROC HPFMM statement, 4216
PRETYPE option
  OUTPUT statement (HPFMM), 4241
PRIOROPTIONS option
  BAYES statement (HPFMM), 4227
PRIOROPTS option
  BAYES statement (HPFMM), 4227
PROBMODEL statement
  HPFMM procedure, 4242
PROC HPFMM procedure, PROC HPFMM statement
  SINGRES= option, 4219
PROC HPFMM statement
  HPFMM procedure, 4208
REFERENCE= option
  MODEL statement (HPFMM), 4234
RESTRICT statement
HPFMM procedure, 4244

SEED= option
   PROC HPFMM statement, 4219

SINGCHOL= option
   PROC HPFMM statement, 4219

SINGRES= option
   PROC HPFMM statement, 4219

SINGULAR= option
   PROC HPFMM statement, 4219

STATISTICS option
   BAYES statement (HPFMM), 4228

SUMMARIES option
   BAYES statement (HPFMM), 4228

syntax
   HPFMM procedure, 4208

TECHNIQUE= option
   PROC HPFMM statement, 4219

THIN= option
   BAYES statement (HPFMM), 4229

THINNING= option
   BAYES statement (HPFMM), 4229

TIMEINC= option
   BAYES statement (HPFMM), 4229

UPCASE option
   CLASS statement (HPFMM), 4230

WEIGHT statement
   HPFMM procedure, 4246

ZEROPROB= option
   PROC HPFMM statement, 4220