



THE
POWER
TO KNOW.

**SAS[®] Enterprise Miner[™] and
SAS[®] Text Miner Procedures
Reference for SAS[®] 9.1.3
The DMREG Procedure
(Book Excerpt)**

The correct bibliographic citation for this manual is as follows: SAS Institute Inc., SAS® *Enterprise Miner™* and SAS® *Text Miner Procedures: Reference for SAS® 9.1.3*, Cary, NC: SAS Institute Inc.

SAS® Enterprise Miner™ and SAS® Text Miner Procedures: Reference for SAS 9.1.3

Copyright © 2008 by SAS Institute Inc., Cary, NC, USA.

All rights reserved. Produced in the United States of America.

For a Web download or e-book: Your use of this publication shall be governed by the terms established by the vendor at the time you acquire this publication.

U.S. Government Restricted Rights Notice. Use, duplication, or disclosure of this software and related documentation by the U.S. government is subject to the Agreement with SAS Institute and the restrictions set forth in FAR 52.227-19 Commercial Computer Software-Restricted Rights (June 1987).

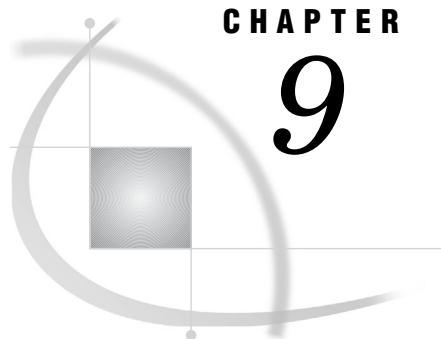
SAS Institute Inc., SAS Campus Drive, Cary, North Carolina 27513.

1st electronic book, October 2008

SAS® Publishing provides a complete selection of books and electronic products to help customers use SAS software to its fullest potential. For more information about our e-books, e-learning products, CDs, and hard-copy books, visit the SAS Publishing Web site at support.sas.com/publishing or call 1-800-727-3228.

SAS® and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

Other brand and product names are registered trademarks or trademarks of their respective companies.



CHAPTER

9

The DMREG Procedure

<i>Overview: DMREG Procedure</i>	161
<i>Syntax: DMREG Procedure</i>	161
<i>PROC DMREG Statement</i>	162
<i>CLASS Statement</i>	164
<i>CODE Statement</i>	164
<i>DECISION Statement</i>	165
<i>FREQ Statement</i>	166
<i>MODEL Statement</i>	167
<i>NLOPTIONS Statement</i>	172
<i>PERFORMANCE Statement</i>	179
<i>REMOTE Statement</i>	180
<i>SCORE Statement</i>	181
<i>Details: DMREG Procedure</i>	184
<i>Input</i>	184
<i>Specification of Effects</i>	184
<i>Optimization Methods</i>	184
<i>Effect-Selection Methods</i>	185
<i>Fit Statistics for OUTEST and OUTFIT Data Sets</i>	186
<i>Examples: DMREG Procedure</i>	188
<i>Example 1: Linear and Quadratic Logistic Regression with an Ordinal Target (Rings Data)</i>	188
<i>Example 2: Performing a Stepwise OLS Regression (DMREG Baseball Data)</i>	201
<i>Example 3: Comparison of the DMREG and LOGISTIC Procedures for Modeling a Categorical Target</i>	210
<i>References</i>	215

Overview: DMREG Procedure

Use the DMREG procedure to fit both linear and logistic regression models. Linear regression attempts to predict the value of a continuous target as a linear function of one or more independent inputs. Logistic regression attempts to predict the probability that a categorical (binary, ordinal, or nominal) target will acquire the event of interest as a function of one or more independent inputs. The procedure supports forward, backward, and stepwise selection methods. It also allows you to score data sets or generate SAS DATA step code to score a data set.

Syntax: DMREG Procedure

```
PROC DMREG < option(s)>;
```

```

CLASS variable(s);
CODE code-option(s);
DECISION DECADATA=<libref.>SAS-data set <DECVARS=decision-variable(s)>
    <option(s)>;
FREQ variable;
MODEL target-variable=input-variable(s) </ model-option(s)>;
NLOPTIONS nonlinear-option(s);
PERFORMANCE performance-option(s);
REMOTE remote-option(s);
SCORE scoring-option(s);

```

PROC DMREG Statement

Invokes the DMREG procedure.

```
PROC DMREG<option(s)>;
```

Required Arguments

DATA= <*libref.*> *SAS-data set*

Identifies the training data set.

DMDBCAT= <*libref.*> *SAS-catalog*

Identifies the training data catalog.

Options

COVOUT

Includes the covariance matrix of the parameter estimates in the OUTEST= data set.

ESTITER= *n*

Includes parameter estimates and fit statistics (for training, test, and validation data) for every *n*th iteration in the OUTEST= data set.

Default: The default value for *n* is *n* = 0. When *n* = 0, only the parameter estimates of the final iteration are output.

INEST= <*libref.*> *SAS-data set*

Specifies the data set that contains initial estimates.

MINIMAL

Minimizes resources by using less memory during logistic regression model fitting. Uses the conjugate gradient technique and standard errors for the regression parameters are not computed. Memory for the Hessian matrix is not needed. Model selection is disabled when the MINIMAL option is specified. The MINIMAL option does not apply to normal error regression models.

NAMELEN= *n* where $20 \leq n \leq 200$

Specifies the length of effect names in the printed output. The effect names are limited to *n* characters, where *n* is an integer value between 20 and 200.

Default: The default value for n is 20 characters.

NOPRINT

Suppresses all printed output.

OUTMAP= <libref.> SAS-data set

Specifies the name of the output data set that contains the variable mappings for each of the parameter names. There is a variable for each model parameter. In addition, the data set contains the character variable `_TYPE_` of length eight, and the character variable `_NAME_` of length eight. Currently the default value of `_TYPE_` is 'MAPPING'. There are two possible values for `_NAME_`. `_NAME_='INPUT'` identifies the input variables that the parameters are associated with. `NAME='CODE'` identifies the equivalent variables that are used in the scoring code.

OUTEST= <libref.> SAS-data set

Specifies the name of an output data set which contains the estimation and fit statistics.

OUTTERM= <libref.> SAS-data set

Specifies the name of the output data set that contains the estimated coefficients, the corresponding t-statistics and the p -values. The data set consists of the following variables:

Term

a numeric variable that represents the parameter number, starting from zero. Observations that have the same term value correspond to the same parameter.

Variable

a character variable that represents the name of the input variables that the parameter is associated with.

ClassLevel

a character variable that represents the categorical level of the corresponding input variable. The ClassLevel value is missing (blank) if the corresponding input is an interval variable.

Coefficient

represents the estimated regression coefficient.

tValue

represents the t-statistic.

pValue

represents the p -value.

Note: OUTTERM= values of Variable and ClassLevel identify a parameter, but it might take more than one observation to do so. In such a case, the Term values for these observations are the same. Δ

SIMPLE

Prints simple descriptive statistics for the input variables.

TESTDATA= <libref.> SAS-data set

Identifies the data set that contains test data.

VALIDATA= <libref.> SAS-data set

Identifies the data set that contains validation data.

CLASS Statement

Specifies one or more categorical variables to be used in the analysis.

CLASS *categorical-variable(s)*;

Required Argument

categorical-variable(s)

Specifies a list of categorical (class) variables to be used in the analysis. You must specify the target variable if the target variable has a categorical (binary, ordinal, or nominal) measurement level.

CODE Statement

If you want to score a data set, you can use a **CODE** statement to write SAS DATA step code to a file or catalog entry. This code can then be included into a DATA step that uses a SET statement to read the data set to be scored. Alternatively, you can use the PMML option in the CODE statement to produce XML scoring code for a database engine. PROC DMREG allows multiple CODE statements.

CODE *<code-option(s)>*;

CODE Options

CATALOG | CAT | C= *library.catalog.entry.type*

Specifies where to write the output SAS DATA step code using the form of library.catalog.entry.type. The compound name can have one to four levels. The default library is determined by the SAS system option USER=, which is usually set to WORK. The default entry is SASCODE, and the default type is SOURCE.

Note: You cannot specify both FILE= and CATALOG= in the same CODE statement. If you specify neither, the code is written to the SAS log unless the PMML option is specified. Δ

DUMMIES | NODUMMIES

Use DUMMIES | NODUMMIES to specify whether to keep dummy variables, standardized variables, or other transformed variables in the data set.

Default: NODUMMIES

ERROR | NOERROR

Specifies whether the error function E_* is to be computed.

FILE=*file-name*

Specifies the file to be used for the output SAS DATA step code.

When enclosed in a quoted string, FILE= provides the path specification to an external file. For example, FILE="c:\mydir\scorecode.sas".

FILE= can also use unquoted SAS filenames of no more than eight characters. If the filename is assigned as a fileref in a FILENAME statement, the file that is

specified in the FILENAME statement is opened. The special filerefs LOG and PRINT are always assigned. If the specified name is not an assigned fileref, then the specified name value is concatenated with a .txt extension before opening. For example, if FOO is not an assigned fileref, FILE=FOO would cause FOO.txt to be opened. If the specified filename has more than eight characters, an error message is printed.

FORMAT= *format*

Use FORMAT= to format weights or other numeric values that don't have a format specified in the input data set.

Default: BEST20.

GROUP= *group identifier*

Use GROUP= to specify the group identifier for group processing. The group identifier should be a valid SAS name of no more than 16 characters, which is used to construct array names and statement labels in the generated code.

LINESIZE | LS= *integer-value*

Use LINESIZE= to specify the line size for generated code. The permissible integer range for LINESIZE = is 64 to 254.

Default: 72

PMML | XML

Produces scoring code in Predictive Modeling Markup Language, an XML-based standard for representing data mining results. You must specify an ODS PMML destination before invoking the procedure, as follows:

```
ods pmml file='foo.xml';
proc dmreg;
  model y=x;
  code pmml;
run;
ods pmml close;
```

For more information, see the PMML Support in Enterprise Miner section in the Enterprise Miner 5.3 Java Help.

RESIDUAL | NORESIDUAL

Use the RESIDUAL option to generate residual values for the variables R_*, F_*, CL_*, CP_*, BL_*, BP_*, and ROI_*. If you request code for residuals and then score a data set that does not contain target values, the residuals will have missing values.

Default: NORESIDUAL

DECISION Statement

Specifies information used for decision processing in the DECIDE, DMREG, NEURAL, and SPLIT procedures. *This documentation applies to all four procedures.*

DECISION DECDATA= *<libref.> SAS-data set* **<DECVARS=**
decision-variable(s)><option(s)>;

DECDATA= *<libref.> SAS-data set*

Specifies the input data set that contains the decision matrix. The DECADATA= data set must contain the target variable.

Note: The DECADATA= data set might also contain decision variables that are specified using the DECVAR= option, and prior probability variable(s) that are specified using the PRIORVAR= option.

The target variable is specified by means of the TARGET statement in the DECIDE, NEURAL, and SPLIT procedures, or by using the MODEL statement in the DMREG procedure. If the target variable in the DATA= data set is categorical, then the target variable of the DECADATA= data set should contain the target category values. The decision variables will display the effects of making those decisions for the corresponding target level. If the target variable is interval, then each decision variable will display the value of the consequence for that decision at a point specified in the target variable. The unspecified regions of the decision function are interpolated by a piecewise linear spline. Δ

Tip: The DECADATA= data set can be of TYPE=LOSS, PROFIT, or REVENUE. If unspecified, TYPE=PROFIT is assumed by default. TYPE= is a data set option that should be specified when the data set is created.

DECVAR= *decision-variable(s)*

Specifies the decision variables in the DECADATA= data set that contain the target-specific consequences for each decision.

COST=*cost-option(s)*

Specifies numeric constants that gives the cost of a decision, or variables in the DATA= data set that contain the case-specific costs, or any combination of constants and variables. The DECVAR= option statement must contain the same number of cost constants and variables as there are decision variables. You cannot use abbreviated variable lists for the COST= option, such as D1-D3, ABC-XYZ, or PQR:.

Note: You can specify the COST= option only when the DECADATA= data set is of TYPE=REVENUE. Δ

Default: Unless otherwise specified, all costs are assumed to be 0.

PRIORVAR= *prior-probability-variable*

Specifies the variable in the DECADATA= data set that contains the prior probabilities to be used for making decisions.

FREQ Statement

Specifies the variable that contains frequencies for training data.

FREQ <variable> ;

variable

Specifies the frequency variable. The frequency variable can contain real values or integer values. That is, noninteger frequency values are not truncated to integers. By default, if the FREQ statement is not specified, the frequency variable in the DMDB is used. If the FREQ statement is specified without a variable, a frequency of 1 is used for all observations.

CAUTION:

If the DMDB contains a frequency variable, it is not advisable to use the FREQ statement to specify a different variable for the FREQ role, since observations that have

invalid DMDB frequency variable values (for example, zero or negative values) in the training data are automatically discarded. This means observations that had valid values for the variable specified in the FREQ statement, but invalid values for the DMDB frequency variable, are improperly excluded from the data mining analysis. △

MODEL Statement

Specifies a single target variable and the explanatory variables.

MODEL *target-variable=input-variable(s) </ model-option(s)>*;

Required Argument

target-variable=input-variable(s)

where the arguments are defined as follows:

target-variable

Specifies the target variable that the model attempts to predict values for.

input-variable(s)

Specifies the input variables or effects that are mined for target variable predictions.

Options

model-options(s)

Specifies options that affect the fit, confidence intervals, variable selection, and specification of the model as follows:

MODEL Options - Display Options

CORRB

Prints the correlation matrix in the output.

COVB

Prints the covariance matrix in the output.

DETAILS

Prints details for each steps of the variable selection process in the output.

NODESIGNPRINT | NODP

Suppresses the class variable input coding display in the output.

SHORT

Suppresses printing the results of the intermediate models that the DMREG procedure fits during the variable selection process. When you specify the SHORT model option, the output prints only the variable selection summary and the results of the chosen model. The SHORT option has no effect if the SELECTION= option is not specified.

MODEL Options - Fitting Options

MISCCONV= *number where $0 \leq \text{number} \leq 1$.*

Specifies the critical misclassification rate for the convergence criterion that is based on misclassification rates. If the MISCCONV= value is greater than 0, the optimization stops iterating and declares convergence when the misclassification rate is less than or equal to the MISCCONV= value.

Default: 0

STARTMISC= *integer where $\text{integer} > 0$*

Specifies the number of iterations to be processed before checking the misclassification rate. The default value for the number of iterations depends on the optimization method that is specified in the TECHNIQUE= option of the NLOPTIONS statement:

- When the TECHNIQUE= option specifies NEWRAP (Newton-Raphson Optimization with Line Search), NRRIDG (Newton-Raphson Ridge Optimization) , or TRUREG (Trust Region Optimization), then the default value for STARTMISC= is 3.
- When the TECHNIQUE= option specifies QUANEW (Quasi-Newton Optimization) or DBLDOG (Double Dogleg Optimization) , then the default value for STARTMISC= is 5.
- When the TECHNIQUE= option specifies CONGRA (Conjugate Gradient Optimization) , then the default value for STARTMISC= is 10.

MODEL Options - Confidence Interval Options

ALPHA= *number where $0 < \text{number} < 1$.*

Specifies the significance level for confidence interval regression parameters.

Default: 0.05

CLPARM

Computes and includes confidence intervals for regression parameters in the output.

MODEL Options - Selection Options

CHOOSE= *criterion*

Chooses, from the list of models at the steps of the selection process, the model that yields the best value of the specified criterion. If the optimal value of the specified criterion occurs for models at more than one step, then the model with the smallest number of parameters is chosen. The following criteria are available for choosing the model:

AIC

Chooses the model that has the smallest Akaike Information Criterion (AIC) value.

NONE

Chooses the model at the final step of the selection process.

SBC

Chooses the model that has the smallest Schwarz Bayesian Criterion (BIC) value.

TDECDATA

Chooses the model that has the largest total profit or smallest total loss for the training data.

VDECDATA

Chooses the model that has the largest total profit or smaller total loss for the validation data.

VERROR

Chooses the model that has the smallest error rate for the validation data. The error rate for a least-square regression model is the sum of squared errors. The error rate for a logistic regression model is the negative log-likelihood.

VMISC

Chooses the model that has the smallest misclassification rate for the validation data.

XDECDATA

Chooses the model according to the total profit or loss from cross-validation of the training data. The model with the largest profit or smallest loss is chosen.

XERROR

Chooses the model with the smallest error rate from cross-validation of the training data. The error rate for a least-square regression model is the sum of squared errors. The error rate for a logistic regression model is the negative log-likelihood.

XMISC

Chooses the model with the smallest misclassification rate from cross-validation of the training data.

Default: The default setting is `CHOOSE=VERROR` if you have specified a `VALIDATA=` data set; otherwise, the default is `CHOOSE= TDECDATA` if you use decision processing (that is, the `DECISION` statement is specified) or else the default is `CHOOSE= NONE`.

HIERARCHY= *variable-inclusion-rule*

Specifies the variable inclusion rule to be used to determine variable selection for the model. The following variable inclusion rules are available for the `HIERARCHY=` model selection option:

ALL

All independent variables that meet hierarchical requirements are included in the model.

CLASS

Only class variables that meet hierarchical requirements are included in the model.

Default: `HIERARCHY=ALL`

INCLUDE= *n* where *n* is an integer value ≥ 0 .

Includes the first *n* effects in the model.

Default: 0

MAXSTEP= *integer-value* where *integer-value* ≥ 0 .

Specifies the maximum number of steps permitted when you use the `STEPWISE` variable selection method.

Default: The default setting for the `MAXSTEP=` option is two times the number of effects that are specified in the `MODEL` statement.

RULE= *effects-selection-rule*

Specifies the effects selection rule that you want to use when you specify the SELECTION= option of the MODEL statement to FORWARD, BACKWARD, or STEPWISE. The following effect selection rules are available:

MULTIPLE

One or more effects can be considered for entry or removal at the same time, as long as the hierarchical rule is observed. For example, if main effects A and B are not in the model, and the interaction A*B is not in the model, then the effects that can be considered for entry in a single step using the MULTIPLE effects selection rule are:

- A alone
- B alone
- A, B
- A*B.

SINGLE

A single effect is considered for entry into the model only if its lower order effects are already in the model. A single effect is considered for removal from the model only if its higher order effects are not in the model.

NONE

Effects are included or excluded one at a time, without preservation of any hierarchical order.

Default: The default setting for the RULE= option is NONE.

SELECTION=effect-selection-method

Specifies the method for selecting effects for the selecting process.

The following methods are available and are explained in detail in the section Effect-Selection Methods.

NONE

Specifies no model selection. The complete model in the MODEL statement is used to fit the model.

FORWARD

Specifies forward selection. This method starts with no effects in the model and adds effects.

BACKWARD

Specifies backward elimination. This method starts with all effects in the model and deletes effects.

STEPWISE

Specifies stepwise regression. This is similar to the FORWARD method except that effects already in the model do not necessarily stay there.

Default: The default, if the SELECTION= option is omitted, is SELECTION= NONE.

SEQUENTIAL

Adds or deletes input variables and effects sequentially, as specified in the MODEL statement.

SLENTRY= significance-level where significance-level > 0.

Specifies the significance level to be used as a threshold criterion for adding input variables. Variables that have *p*-values less than or equal to the specified entry threshold are added as inputs.

Default: 0.05

SLSTAY=*significance-level* where *significance-level* > 0.

Specifies the significance level to be used as a threshold criterion for removal of variables. Variables that have p -values greater than the specified stay threshold are removed from inputs.

Default: 0.05

START= *integer-value* where *integer-value* > 0.

Includes the first n effects in the starting model.

- For models that use the FORWARD or STEPWISE selection methods, the default value for the START= option is 0.
- For models that use the BACKWARD selection method, the default value for the START= option is s , the total number of effects in the MODEL statement.

STOP= *number-of-added-or-deleted-effects*

Terminates the effect selection process after the specified number of effects have been either added (using the FORWARD method) or removed (using the BACKWARD method) from the model. The STOP= option has no effect when the model effect selection method is set to SELECTION=NONE or SELECTION=STEPWISE.

The number of added or deleted effects is an integer value that ranges from 0 to s , where s is the total number of effects in the MODEL statement.

Default: The default setting for the number of added effects (FORWARD effects selection method) is s . The default setting for the number of deleted effects (BACKWARD effects selection method) is 0.

MODEL Options - Specification Options

CODING= *class-input-variable-coding-method*

Specifies the coding method that you want to use for class input variables. The following class input variable coding methods are available with the CODING= model option:

DEVIATION

Deviation from mean coding, which is also known as effect coding.

GLM

Non-full rank GLM (Generalized Linear Model) coding, as performed by the GLM procedure.

Default: CODING=DEVIATION

ERROR=MBERNOULLI | NORMAL

Specifies the error distribution.

MBERNOULLI

Multinomial distribution with one trial. This includes the binomial distribution with one trial. MBERNOULLI is not available if the target measurement level is interval.

Alias: BINOMIAL or MULTINOMIAL

NORMAL

Normal distribution. NORMAL is not allowed if the target measurement level is nominal.

Default: ERROR=NORMAL (for LEVEL=INTERVAL), ERROR=MBERNOULLI (otherwise).

LEVEL= INTERVAL | NOMINAL | ORDINAL

Specifies the measurement level of the target variable.

INTERVAL

Interval variable.

NOMINAL

Nominal variable.

ORDINAL

Ordinal variable.

Default: ORDINAL for a categorical target; INTERVAL for a numerical target.

LINK= CLOGLOG | IDENTITY | LOGIT | PROBIT

Specifies the link function that represents the expected values of the target to the linear predictors.

CLOGLOG

Specifies the complementary log-log function, which is the inverse of the extreme value distribution function. The CLOGLOG function is available for ordinal or binary targets.

IDENTITY

Specifies the identity function. The IDENTITY function can be used only for the linear regression analysis (ERROR=NORMAL).

LOGIT

Specifies the logit function, which is the inverse of the logistic distribution function. The LOGIT function is available for nominal, ordinal, or binary targets.

PROBIT

Specifies the probit function, which is the inverse of the standard normal distribution function. The PROBIT function is available for ordinal or binary targets.

Default:

LOGIT (for ERROR=MBERNOULLI), IDENTITY (for ERROR=NORMAL).

Tip: The CLOGLOG, LOGIT, and PROBIT link functions are used for a logistic regression analysis. The IDENTITY link function is used for a linear regression analysis.

NOINT

Suppresses the intercept for the binary target model or the normal error linear regression model.

SINGULAR= n

Specifies the tolerance for testing singularity.

Default: $1e - 6$

NLOPTIONS Statement

Specifies options for nonlinear optimization. These options only apply to fitting the logistic regression model. Let β be the parameter vector of interest. The log-likelihood function $l(\beta)$ is the objective function being maximized. Its gradient vector and Hessian matrix are denoted as $g(\beta)$ and $H(\beta)$, respectively. The gradient with respect to the i th parameter is denoted as $g_i(\beta)$. Superscripts in parentheses denote the iteration count; for example, $l(\beta^{(k)})$ is the value of the log-likelihood at iteration k . Denote

$g^{(k)} = g\left(\beta^{(k)}\right)$ as the gradient vector at iteration k ; denote $H^{(k)} = H\left(\beta^{(k)}\right)$ as the Hessian matrix at iteration k , and denote $H_{i,i}^{(k)}$ as the i th diagonal element of $H^{(k)}$.

NLOPTIONS *nonlinear-option(s)*;

Nonlinear-Options

ABSCONV= number

Specifies an absolute function convergence criterion. Termination requires

$$l\left(\beta^{(k)}\right) \geq \text{number}.$$

Default: The default value is 1e-3 times the log-likelihood of the null model (intercept-only model).

Range: *number* > 0

ABSFCONV= number

Specifies an absolute function convergence criterion. Termination requires a small change of the log-likelihood in successive iterations:

$$\left|l\left(\beta^{(k+1)}\right) - l\left(\beta^{(k)}\right)\right| \leq \text{number}.$$

Default: 0

Range: *number* > 0

ABSGCONV= number

Specifies the absolute gradient convergence criterion. Termination requires the maximum absolute gradient element to be small:

$$\max_j \left|g_j\left(\beta^{(k)}\right)\right| \leq \text{number}.$$

Default: 1E-5

Range: *number* > 0

ABSXCONV= number

Specifies the absolute parameter convergence criterion. Termination requires a small Euclidean distance between successive parameter vectors:

$$\left\|\beta^{(k)} - \beta^{(k-1)}\right\|_2 \leq \text{number}.$$

Default: 1E-8

Range: *number* > 0

DAMPSTEP= number

Specifies that the initial step size value for each line search used by the QUANEW, CONGRA, or NEWRAP techniques cannot be larger than the product of *number* and the step size value used in the previous iteration.

Default: 2

Range: *number* > 0

DIAHES

Forces the optimization algorithm (TRUREG, NEWRAP, or NRRIDG) to take advantage of the diagonality.

FCONV= number

Specifies a function convergence criterion. Termination requires a small relative change of the log-likelihood in successive iterations:

$$\frac{|l(\beta^{(k)}) - l(\beta^{(k-1)})|}{\max(|l(\beta^{(k-1)})|, FSIZE)} \leq number$$

where *FSIZE* is defined in the *FSIZE=* option.

Default: $10^{-FDIGITS}$, where *FDIGITS* is the value of the *FDIGITS=* option.

Range: *number* > 0

FCONV2= number

Specifies another function convergence criterion. Termination requires a small predicted reduction:

$$l(\beta^{(k)}) - l\left(\beta^{(k)} - [H^{(k)}]^{-1} g^{k(0)}\right) \leq number$$

Default: 10^{-4}

Range: *number* > 0

FDIGITS= number

Specifies the number of accurate digits in evaluations of the objective function.

Default: $-\log_{10}(\epsilon)$, where ϵ is the machine precision.

Range: *number* > 0

FSIZE= number

Specifies the parameter of the relative function and relative gradient termination criteria.

Default: 0

Range: *number* \geq 0

GCONV= number

Specifies the relative gradient convergence criterion. Except the CONGRA technique, termination requires the normalized predicted function reduction is small:

$$\frac{[g^{(k)}]' [H^{(k)}]^{-1} g^{(k)}}{\max(|l(\beta^{(k)})|, FSIZE)} \leq number$$

where *FSIZE* is defined in the *FSIZE=* option. For the CONGRA technique, termination requires

$$\frac{\|g^{(k)}\|_2 \|s(\beta^{(k)})\|_2}{\|g^{(k)} - g^{(k-1)}\|_2 \max(|l(\beta^{(k)})|, FSIZE)} \leq number$$

where $s(\beta^{(k)})$ is the step increment at iteration *k*.

Default: 10^{-6}

Range: *number* > 0

GCONV2= number

Specifies another relative gradient convergence criterion. Termination requires

$$\max_i \left| \frac{g_i(\beta^{(k)})}{\sqrt{l(\beta^{(k)}) H_{i,i}^{(k)}}} \right| \leq number.$$

Default: 0**Range:** *number* > 0**HESCAL=** 0 | 1 | 2 | 3

Specifies the scaling version of the Hessian or cross-product Jacobian matrix used in NRRIDG, TRUREG, LEVMAR, NEWRAP, or DBLDOG optimization. Specifies the scaling version of the Hessian (or crossproduct Jacobian) matrix used in NR-RIDG, TRUREG, LEVMAR, NEWRAP, or DBLDOG optimization. If HESCAL is not equal to 0, the first iteration and each restart iteration set the diagonal scaling matrix $D^{(0)} = \text{diag} \left(d_i^{(0)} \right)$: $d_i^{(0)} = \sqrt{\max \left(\left| H_{i,i}^{(0)} \right|, \epsilon \right)}$ where $H_{i,i}^{(0)}$ are the diagonal elements of the Hessian (or crossproduct Jacobian). In every other iteration, the diagonal scaling matrix $D^{(0)} = \text{diag} \left(d_i^{(0)} \right)$ is updated depending on the HESCAL option:

HESCAL Value	Result
0	specifies that no scaling be done.
1	specifies the More' (1978) scaling update: $d_i^{(k+1)} = \max \left[d_i^{(k)}, \sqrt{\max \left(\left H_{i,i}^{(k)} \right , \epsilon \right)} \right]$
2	specifies the Dennis, Gay, and Welsch (1981) scaling update: $d_i^{(k+1)} = \max \left[0.6 * d_i^{(k)}, \sqrt{\max \left(\left H_{i,i}^{(k)} \right , \epsilon \right)} \right]$
3	specifies that d_i be reset in each iteration: $d_i^{(k+1)} = \sqrt{\max \left(\left H_{i,i}^{(k)} \right , \epsilon \right)}$

In each scaling update, ϵ is the relative machine precision. The default value is HESCAL=0. Scaling of the Hessian can be time-consuming in the case where general linear constraints are active.

Default:

1 - for LEVMAR minimization technique

0 - for all others

INHESIAN= *number*

Specifies how the initial estimate of the approximate Hessian is defined for the quasi-Newton techniques QUANEW and DBLDOG. The initial estimate of the approximate Hessian is set *number* times the matrix. If the specified *number* is 0, the initial estimate of the approximate Hessian is computed from the magnitude of the initial gradient.

Range: $number \geq 0$

Default: the initial estimate of the approximate Hessian is set to $H^{(0)}$, the Hessian at the initial estimate $\beta^{(0)}$.

INSTEP= *number*

Specifies a larger or smaller radius of the trust region used in the TRUREG, DBLDOG, and LEVMAR algorithms.

Default: 1

Range: $number > 0$

LINESEARCH= *number*

Specifies the line-search method for the CONGRA, QUANEW, and NEWRAP optimization techniques.

Default: 2

Range: $1 \leq number \leq 8$

LSPRECISION= *number*

Specifies the degree of accuracy that should be obtained by the second and third line-search algorithms.

Default:

Table 9.1 Table of Line-Search Precision Values

Technique=	Update=	LSPrecision Value
QUANEW	DBFGS, BFGS	0.4
QUANEW	DDFP, DFP	0.06
CONGRA	all	0.1
NEWRAP	no update	0.9

Range: $number > 0$

MAXFUNC= *number*

Specifies the maximum number of function calls in the optimization process. The objective function that is minimized is the negative log-likelihood.

Default:

125 for TRUREG, NRRIDG, and NEWRAP.

500 for QUANEW and DBLDOG.

1000 for CONGRA.

Range: $number > 0$

MAXITER= *number*

Specifies the maximum number of iterations in the optimization process.

Default:

50 for TRUREG, NRRIDG and NEWRAP

200 for QUANEW and DBLDOG

400 for CONGRA

Range: *number* > 0

MAXSTEP= *number*

Specifies the upper bound for the step length of the line-search algorithms.

Default: The largest double precision value

Range: *number* > 0

MAXTIME= *number*

Specifies the upper limit of CPU time for the optimization process. It is measured in seconds.

Default: 7 days, that is, MAXTIME=604800 seconds

Range: *number* > 0

NOPRINT

Suppresses all output printed and only ERRORS, WARNINGS, and NOTES are printed on the log file.

PALL

Prints all optional output except the output generated by the PSTDERR , LIST, or LISTCODE options.

PHISTORY

Prints the optimization history. If PSUMMARY or NOPRINT are not specified, then the PHISTORY option is set automatically. The iteration history is printed by default.

PSUMMARY

Restricts the amount of default printed output to a short form of iteration history and NOTES, WARNINGS, and ERRORS.

RESTART= *number*

Specifies that the QUANEW or CONGRA algorithm is restarted with a steepest descent/ascent search direction after the *number* of iterations has been completed.

Default:

For TECHNIQUE=CONGRA, and UPDATE= PB, restart is done automatically, so *number* is not used;

For TECHNIQUE=CONGRA, and UPDATE not = PB, *number* is the number of parameters.

For TECHNIQUE=QUANEW, *number* is the largest integer available.

Range: *number* > 1

SINGULAR= *number*

Specifies an absolute singularity criterion for the computation of the inertia of Hessian and cross-product Jacobian and their projected forms.

Default: 1E-8

Range: *number* > 0

TECHNIQUE= *method*

where *method* is one of the following:

NONE

Specifies no method; no optimization is performed.

TRUREG

Specifies the Trust-Region optimization technique.

NEWRAP

Specifies the Newton-Raphson with Line Search optimization technique.

NRRIDG

Specifies the Newton-Raphson with Ridging optimization technique. This is the default when the number of parameters to be estimated is $n \leq 40$.

DBLDOG

Specifies the Double-Dogleg optimization technique.

QUANEW

Specifies the quasi-Newton optimization technique. This is the default when the number of convergence parameters to be estimated is in the range: $40 < n \leq 400$.

CONGRA

Specifies the Conjugate Gradient optimization technique. This is the default when the number of convergence parameters to be estimated is $n \geq 400$.

Default: The default technique is either NRRIDG, QUANEW, or CONGRA, depending on the value of the number of convergence parameters to be estimated.

See for more information.

UPDATE=update-type

where *update-type* is one of the following:

BFGS

For TECHNIQUE=QUANEW, performs the BFGS (Broyden-Fletcher-Goldfarb-Shanno) update of the Cholesky factor of the Hessian matrix.

CD

For TECHNIQUE=CONGRA, performs a conjugate descent update of Fletcher.

DBFGS

For TECHNIQUE=DBLDOG or QUANEW, performs the dual BFGS (Broyden-Fletcher-Goldfarb-Shanno) update of the Cholesky factor of the Hessian matrix. This is the default for TECHNIQUE=QUANEW and DBLDOG.

DDFP

For TECHNIQUE=DBLDOG or QUANEW, performs the dual DFP (Davidson-Fletcher-Powell) update of the Cholesky factor of the Hessian matrix.

DFP

For TECHNIQUE=QUANEW, performs the original DFP (Davidson-Fletcher-Powell) update of the inverse Hessian matrix.

FR

For TECHNIQUE=CONGRA, performs the Fletcher-Reeves update.

PB

For TECHNIQUE=CONGRA, performs the automatic restart update method of Powell and Beale. This is the default for TECHNIQUE= CONGRA.

PR

For TECHNIQUE=CONGRA, performs the Polak-Ribiere update.

VERSION= 1 | 2 | 3

Specifies the version of the hybrid quasi-Newton optimization technique or the version of the quasi-Newton optimization technique with nonlinear constraints.

Default: 2

XCONV= number

Specifies the relative parameter convergence criterion. Termination requires a small relative parameter change in successive iterations:

$$\frac{\max(\beta_i^{(k)} - \beta_i^{(k-1)})}{\max(\beta_i^{(k)}, \beta_i^{(k-1)}, XSIZE)} \leq \text{number}$$

where *SXIZE* is defined in the *XSIZE=* option.

Default: 1E-8

Range: *number* > 0

XSIZE= *number*

Specifies the number of successive iterations for which the criterion must be satisfied before the optimization process can be terminated.

Default: 0

Range: *number* \geq 0

PERFORMANCE Statement

Overview

The PERFORMANCE statement is used to change default options that affect the performance of the DMREG procedure and to request tables that show the performance options in effect and timing details. Threaded code can be used only for binary target models.

PERFORMANCE <*option(s)*> ;

Options

The following options are available for the PERFORMANCE statement:

CPUCOUNT= ACTUAL | *integer* where $1 \leq \textit{integer} \leq 1024$

Specifies the number of processors that you want the DMREG procedure to use for multithreaded processing.

ACTUAL

CPUCOUNT=ACTUAL sets CPUCOUNT to be the number of physical processors available. Note that this can be less than the physical number of CPUs if the SAS process has been restricted by system administration tools.

integer where $1 \leq \textit{integer} \leq 1024$

Manually specifies between 1 and 1024 processors for the DMREG procedure to use during multithreaded processing. Setting CPUCOUNT= to a number greater than the actual number of available CPUs might result in reduced performance.

Note: The CPUCOUNT= option for the PERFORMANCE statement overrides the SAS system option CPUCOUNT=. If CPUCOUNT=1, NOTTHREADS is in effect. (Multithreading is not normally performed on a single processor.) Δ

DETAILS

Includes additional detail in the PERFORMANCE statement output, such as the performance settings table that displays the settings that you configure for the PERFORMANCE statement, and a timing table that displays a broad timing breakdown of the steps in the DMREG procedure.

NOTTHREADS

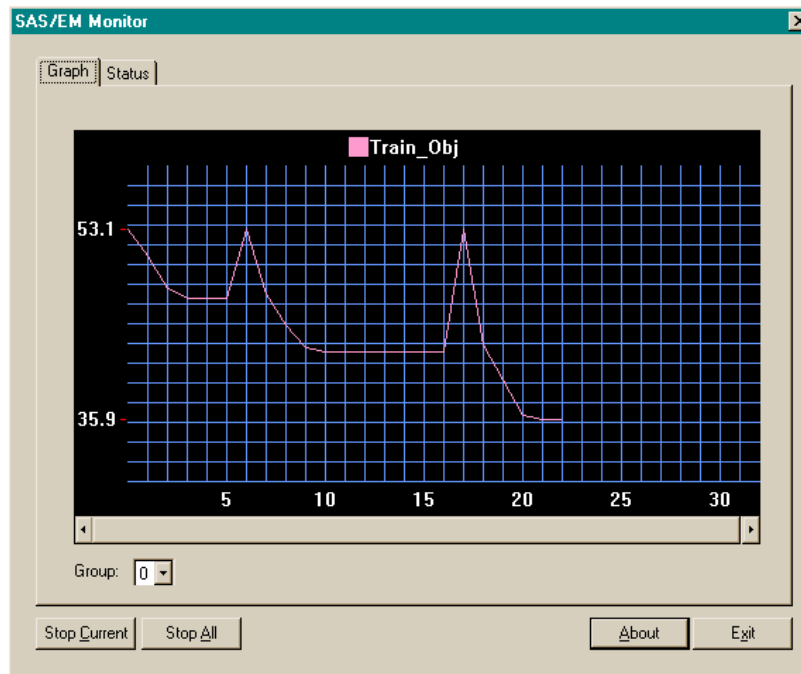
Disables multithreaded computation for fitting a binary target model. The NOTTHREADS option for the PERFORMANCE statement overrides the SAS system THREADS | NOTTHREADS option.

THREADS

Enables multithreaded computation for fitting a binary target model. The THREADS option for the PERFORMANCE statement overrides the SAS system THREADS | NOTTHREADS option.

REMOTE Statement**Overview**

The REMOTE statement is implemented in the NEURAL, DMREG, and DMVQ Enterprise Miner procedures. You can use it to communicate with an MFC monitor (an external process on a Windows client) to observe the progress of the iterative algorithm or to interrupt the iterative process. The monitor has a Graph tab and a Status tab as shown below:



The Graph tab displays the iteration history: objective function versus iteration number and maximum absolute gradient versus iteration number. Click **Stop Current** or **Stop All** to stop the current or all optimization process. The Status tab displays the objective function and the maximum absolute element of the gradient vectors for each iteration.

REMOTE *remote-option(s)*;

Options

remote-options can be the following:

PLOTFILE=fileref | **'external-file'**

Specifies the external file that contains the iterative history (for example, the iteration number, the objective function, and the maximum absolute gradient). You

can specify the path of the external file in quotes or you can use the FILENAME statement to specify a file reference. This option is obsolete if you can take advantage of the SOCKET= option.

SOCKET=socket-reference

establishes a TCP/IP socket connection to an MFC monitor on the Windows client to receive the report of the ongoing optimization. The socket reference contains the IP address and the port number and can be defined by using the following FILENAME statement:

```
FILENAME <socket-reference> SOCKET '<ip_address:portnum>';
```

where ip_address is the IP address of the Window client and portnum is the socket port number. The socket port number is any number that you use to invoke the MFC monitor.

STOPFILE

Specifies an external file such that the iterative process will be terminated if this file exists. This is useful when you run a project with a large data set. To stop the process, you must create the external file. The DMREG procedure stops the iterative process when it detects this file. The file does not have to have any content. You can specify the path of an external file in quotes or use the LIBNAME statement to specify the file reference. This option is obsolete if you can take advantage of the SOCKET= option.

Example:

```
FILENAME abc SOCKET 'd6026.us.sas.com:1234';
PROC DMREG DATA=SAMPPIO.DMDCENS DMDBCAT=SAMPPIO.DMDCENS;
  REMOTE SOCKET=abc;
  CLASS CLASS WORKCLAS MARTAL OCCUPATN RELATION RACE SEX COUNTRY;
  MODEL CLASS=AGE FNLWGT EDUC_NUM CAP_GAIN CAP_LOSS HOURWEEK
        WORKCLAS MARITAL OCCUPATN RELATION RACE SEX COUNTRY
        / SELECTION=F CHOOSE=AIC;
RUN;
```

You can invoke the monitor any time by using the port number (1234) that you choose. After the socket connect is made you can see the display of the iteration history of the ongoing optimization.

SCORE Statement

Specifies options for scoring data.

SCORE *scoring-option(s)*;

Options

Scoring-options can be the following:

ADDITIONALRESIDUALS

Specifies that the OUT= data set contains additional residuals such as: RS_ for logistic regressions and RS_, RT_, RD_, RDS_, RDT_ for normal error regression.

Alias: ADDRESS | AR

ALPHA=number

Specifies the significance level p for the construction of $100(1-p)\%$ confidence interval for the posterior probabilities. This number must be between 0 and 1.

Default: 0.05

CLP

Specifies that the OUT= data set contains the confidence limits for the posterior probabilities. The significance level is controlled by the ALPHA= option.

DATA=<libref.> SAS-data set

Specifies the input data set that contains inputs and optional targets.

Default: The default is the same as the DATA= data set in the PROC statement.

OUT=<libref.> SAS-data set

Specifies the output data set for scoring the DATA=data set.

Default: DATA n

Names for computed variables are normally taken from the data dictionary. If necessary, names for these variables can be generated by concatenating a prefix to the name of the corresponding target variable according to the rules in the following tables:

Table 9.2 Statistics Generated in the OUT=SAS-data set for Normal Error Regression

Name	Label
P_targetname	Predicted: targetname
E_targetname	Error Function: targetname
R_targetname	Residuals: targetname
RD_targetname	Deviance Residuals: targetname
F_targetname	From: targetname
I_targetname	Into: targetname
RS_targetname	Standardized Residuals: targetname
RT_targetname	Studentized Residuals: targetname
RDS_targetname	Standardized Deviance Residuals: targetname
RDT_targetname	Studentized Deviance Residuals: targetname

Note: In the table above, targetname is the name of the target variable. For example, if PURCHASE is the targetname, the predicted value statistic is named P_PURCHASE and the studentized deviance residual is named RDT_PURCHASE. [If the constructed names are longer than the maximum of 32 characters allowed for SAS variable names, they are truncated to 32 characters.] Δ

Table 9.3 Statistics Generated in the OUT= SAS-data set for Binomial or Multinomial Regression

Name	Label
P_targetname&value	Predicted: targetname=targetvalue
F_targetname	From: targetname
I_targetname	Into: targetname
E_targetname	Error Function:
R_targetname&value	Residual: targetname=targetvalue

If the ADDITIONALRESIDUALS option is specified, the OUT=SAS-data set includes the RS_targetname&value with label Standardized Residual: targetname=targetvalue.

Note: In the table above, targetname&value is a combination of the target name (targetname) and target value (targetvalue). For example, if PURCHASE is the targetname and “YES” and “NO” are the two values possible for targetvalue, the predicted value statistics are named P_PURCHASEYES and P_PURCHASENO. Δ

OUTFIT= <libref.>SAS-data set

Specifies the output data set that contains the fit statistics.

OUTSTEP

Scores the data for each variable selection step.

ROLE=role-value

Specifies the role of the DATA= data set. The ROLE= option primarily affects which fit statistics are computed and what their names and labels are.

Role-value can be:

TRAIN

This value is the default when the same data set name is used in the DATA= option in both the PROC and SCORE statements. Specifying TRAIN with any data set other than the actual training set is an error.

VALID | VALIDATION

This value is the default when the DATA= data set name in the SCORE statement is the same as the data set in the VALIDATA= in the PROC statement.

TEST

This value is the default when the DATA= data set name in the SCORE statement is the same as the data set name in the TESTDATA= option of the PROC statement.

SCORE

Predicted values are produced but residuals, error functions, and other fit statistics are not produced.

Details: DMREG Procedure

Input

The input to the DMREG procedure can be assigned one of these roles:

Training

The DATA= data set is used for training models.

Validation

The VALIDATA= data set is used to compute assessment statistics and to fine-tune the model during stepwise selection.

Test

The TESTDATA= data set is an additional "hold out" data set that you can use to compute assessment statistics.

Score

The DATA= data set in the SCORE statement is used for predicting target values for a new data set that might not contain the target.

Specification of Effects

Different types of effects can be used in the DMREG procedure. In the following list, assume that A, B, and C are class variables and that X1, X2, and Y are continuous variables:

- 1 Regressor effects are specified by writing continuous variables individually:
X1 X2
- 2 Polynomial effects are specified by joining two or more continuous variables with asterisks:
X1*X1 X1*X2
- 3 Main effects are specified by writing class variables individually:
A C
- 4 Crossed effects (interactions) are specified by joining class variables with asterisks:
A*B B*C A*B*C
- 5 Continuous-by-class effects are written by joining continuous variables and class variables with asterisks:
X1*A.

Optimization Methods

The following table provides a list of the general nonlinear optimization methods and the default maximum number of iterations and function calls for each method.

Table 9.4 Optimization Methods for the Regression node

Optimization Method	Maximum Iterations	Maximum Function Calls
Conjugate Gradient	400	1000
Double Dogleg	200	500
Newton-Raphson with Line Search	50	125
Newton-Raphson with Ridging	50	125
Quasi-Newton	200	500
Trust-Region	50	125

You should set the optimization method based on the size of the data mining problem, as follows:

- 1 Small-to-medium problems – The Trust-Region, Newton-Raphson with Ridging, and Newton-Raphson with Line Search methods are appropriate for small and medium sized optimization problems (number of model parameters up to 40) where the Hessian matrix is easy and cheap to compute. Sometimes, Newton-Raphson with Ridging can be faster than Trust-Region, but Trust-Region is numerically more stable. If the Hessian matrix is not singular at the optimum, then the Newton-Raphson with Line Search can be a very competitive method.
- 2 Medium Problems – The quasi-Newton and Double Dogleg methods are appropriate for medium optimization problems (number of model parameters up to 400) where the objective function and the gradient are must faster to compute than the Hessian. Quasi-Newton and Double Dogleg require more iterations than does the Trust-Region or the Newton-Raphson methods, but each iteration is much faster.
- 3 Large Problems – The Conjugate Gradient method is appropriate for large data mining problems (number of model parameters greater than 400) where the objective function and the gradient are much faster to compute than the Hessian matrix, and where they need too much memory to store the approximate Hessian matrix.

Note: To learn about these optimization methods, see The NLP Procedure in the *SAS/OR User's Guide: Mathematical Programming* Δ

The underlying “Default” optimization entry method depends on the number of parameters in the model. If the number of parameters is less than or equal to 40, then the default method is set to Newton-Raphson with Ridging. If the number of parameters is greater than 40 and less than 400, then the default method is set to quasi-Newton. If the number of parameters is greater than 400, then Conjugate Gradient is the default method.

Effect-Selection Methods

Four effect-selection methods are available by specifying the SELECTION= option in the MODEL statement. The simplest method (and the default) is SELECTION=NONE, for which PROC DMREG fits the complete model as specified in the MODEL statement. The other three methods are FORWARD for forward selection, BACKWARD for backward elimination, and STEPWISE for stepwise selection. Intercept parameters are forced to stay in the model unless the NOINT option is specified.

When SELECTION=FORWARD, PROC DMREG first estimates parameters for effects forced into the model. These effects are the intercepts and the first n explanatory effects in the MODEL statement, where n is the number specified by the START= or INCLUDE= option in the MODEL statement (n is zero by default). Next, the procedure computes the score chi-square statistic for each effect not in the model and examines the largest of these statistics. If it is significant at the SLENTY= level, the corresponding effect is added to the model. Once an effect is entered in the model, it is never removed from the model. The process is repeated until none of the remaining effects meet the specified level for entry or until the STOP= value is reached.

When SELECTION=BACKWARD, parameters for the complete model as specified in the MODEL statement are estimated unless the START= option is specified. In that case, only the parameters for the intercepts and the first n explanatory effects in the MODEL statement are estimated, where n is the number specified by the START= option. Results of the Wald test for individual parameters are examined. The least significant effect that does not meet the SLSTAY= level for staying in the model is removed. Once an effect is removed from the model, it remains excluded. The process is repeated until no other effect in the model meets the specified level for removal or until the STOP= value is reached. Backward selection is often less successful than forward or stepwise selection because the full model fit in the first step is the model most likely to result in a monotone likelihood.

The SELECTION=STEPWISE option is similar to the SELECTION=FORWARD option except that effects already in the model do not necessarily remain. Effects are entered into and removed from the model in such a way that each forward selection step can be followed by one or more backward elimination steps. The stepwise selection process terminates if no further effect can be added to the model or if the current model is identical to a previously visited model.

Fit Statistics for OUTEST and UTFIT Data Sets

The OUTEST= data set in the PROC DMREG statement contains fit statistics for the training, test, and/or validation data. Depending on the ROLE= option in the SCORE statement, the UTFIT= data set contains fit statistics for either the training, test, or validation data.

Table 9.5 Fit Statistics for the Training Data

Fit Statistic	Training Data
AIC	Train: Akaike's Information Criterion
ASE	Train: Average Squared Error
AVERR	Train: Average Error Function
DFE	Train: Degrees of Freedom for Error
DFM	Train: Model Degrees of Freedom
DFT	Train: Total Degrees of Freedom
DIV	Train: Divisor for ASE
ERR	Train: Error Function
FPE	Train: Final Prediction Error
MAX	Train: Maximum Absolute Error

Fit Statistic	Training Data
MSE	Train: Mean Square Error
NOBS	Train: Sum of Frequencies
NW	Train: Number of Estimate Weights
RASE	Train: Root Average Sum of Squares
RFPE	Train: Root Final Prediction Error
RMSE	Train: Root Mean Squared Error
SBC	Train: Schwarz's Bayesian Criterion
SSE	Train: Sum of Squared Errors
SUMW	Train: Sum of Case Weights Times Frequency
MISC	Train: Misclassification Rate

Table 9.6 Fit Statistics for the Test Data

Fit Statistic	Test Data
TASE	Test: Average Squared Error
TASEL	Test: Lower 95% Confidence Limit for _TASE_
TASEU	Test: Upper 95% Confidence Limit for _TASE_
TAVERR	Test: Average Error Function
TDIV	Test: Divisor for _TASE_
TERR	Test: Error Function
TMAX	Test: Maximum Absolute Error
TMSE	Test: Mean Square Error
TNOBS	Test: Sum of Frequencies
TRASE	Test: Root Average Squared Error
TRMSE_	Test: Root Mean Square Error
TSSE	Test: Sum of Square Errors
TSUMW	Test: Sum of Case Weights Times Frequency
TMISC	Test: Misclassification Rate
TMISL	Test: Lower 95% Confidence Limit for _TMISC_
TMISU	Test: Upper 95% Confidence Limit for _TMISC_

Table 9.7 Fit Statistics for the Validation Data

Fit Statistic	Validation Data
VASE	Valid: Average Squared Error
VAVERR	Valid: Average Error Function
VDIV	Valid: Divisor for VASE
VERR	Valid: Error Function
VMAX	Valid: Maximum Absolute Error
VMSE	Valid: Mean Square Error
VNOBS	Valid: Sum of Frequencies
VRASE	Valid: Root Average Squared Error
VRMSE	Valid: Root Mean Square Error
VSSE	Valid: Sum of Square Errors
VSUMW	Valid: Sum of Case Weights Times Frequency
VMISC	Valid: Misclassification Rate

Examples: DMREG Procedure

The following examples were executed using the Windows Professional operating system and the SAS software release 9.1.3 Service Pack 4.

Example 1: Linear and Quadratic Logistic Regression with an Ordinal Target (Rings Data)

Features

- Scoring a Test Data Set
- Outputting Fit Statistics
- Creating a Classification Table
- Plotting the Posterior Probabilities

This example demonstrates how to perform both a linear and a quadratic logistic regression with an ordinal target. The example training data set SAMPSIO.DMLRING contains an ordinal target with three levels (C= 0, 1, or 2) and two continuous inputs (X and Y). There are 180 observations in the data set. The SAMPSIO.DMSRING data set is scored using the scoring formula from the trained models. Both data sets are stored in the sample library.

Linear-Logistic Program

PROC GPLOT creates a scatter plot of the Rings training data.

```
proc gplot data=sampsio.dmlring;
```

```

plot y*x=c /haxis=axis1 vaxis=axis2;
symbol c=black i=none v=dot;
symbol2 c=red i=none v=square;
symbol3 c=green i=none v=triangle;
axis1 c=black width=2.5 order=(0 to 30 by 5);
axis2 c=black width=2.5 minor=none order=(0 to 20 by 2);
title 'Plot of the Rings Training Data';
run;

```

The PROC DMREG statement invokes the procedure. The DATA= option identifies the training data set that is used to fit the model. The DMDBCAT=option identifies the DMDB training data catalog. You can create DMDB encoded data sets and catalogs with the DMDB procedure.

```
proc dmreg data=sampsio.dmlring dmdbcat=sampsio.dmdring;
```

The CLASS statement identifies the target C as a categorical variable.

```
class c;
```

The MODEL statement specifies the linear-logistic model.

```
model c = x y;
```

The SCORE statement scores the training data set and outputs fit statistics to the OUTFIT= data set. A note is printed in the log that indicates the training data set is scored when the DATA= option is omitted.

```
score out=out outfit=fit;
```

The second SCORE statement scores the SAMPSIO.DMSRING data set.

```
score data=sampsio.dmsring out=gridout;
      title 'Linear-Logistic Regression with Ordinal Target';
run;
```

PROC PRINT report of selected fit statistics for the training data.

```
proc print data=fit noobs label;
  var _aic_ _max_ _rfpe_ _misc_ ;
  title2 'Fit Statistics for the Training Data Set';
run;
```

PROC FREQ report of the misclassification rate for the training data set. The F_C variable is the actual target value for each case and the I_C variable is the target value into which the case is classified.

```
proc freq data=out;
  tables f_c*i_c;
  title2 'Misclassification Table: Training Data';
run;
```

PROC GPLOT produces a plot of the classification results for the training data.

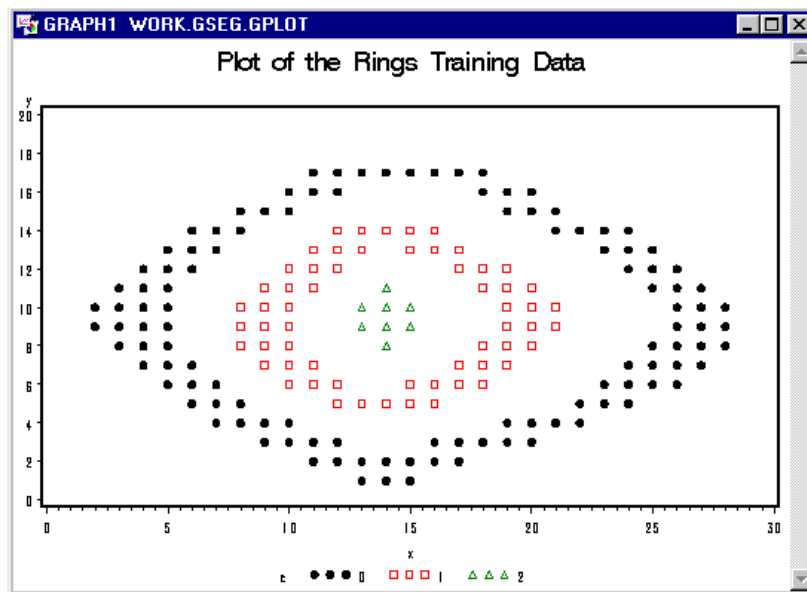
```
proc gplot data=out;
  plot y*x=i_c / haxis=axis1 vaxis=axis2;
  symbol c=black i=none v=dot;
  symbol2 c=red i=none v=square;
  symbol3 c=green i=none v=triangle;
  axis1 c=black width=2.5 order=(0 to 30 by 5);
  axis2 c=black width=2.5 minor=none order=(0 to 20 by 2);
  title2 'Classification Results';
run;
```

PROC GCONTOUR produces plots of the posterior probabilities.

```
proc gcontour data=gridout;
  plot y*x=p_c1 / pattern ctext=black coutline=gray;
  plot y*x=p_c2 / pattern ctext=black coutline=gray;
  plot y*x=p_c3 / pattern ctext=black coutline=gray;
  title2 'Posterior Probabilities';
  pattern v=msolid;
  legend frame;
run;
```

Linear-Logistic Output

PROC GPLOT Plot of the Rings Training Data



DMREG Summary Profile Information

PROC DMREG first lists background information about the fitting of the linear-logistic model. Included are the name of the input data set, the response variable, the number of response levels, the number of observations used, the error distribution, and the link function.

Linear-Logistic Regression with Ordinal Target

The DMREG Procedure

Model Information

```

Training Data Set          SAMPSIO.DMDRING.DATA
DMDB Catalog              SAMPSIO.DMDRING
Target Variable           c
Target Measurement Level  Ordinal
Number of Target Categories 3
Error                    MBernoulli
Link Function             Logit
Number of Model Parameters 4
Number of Observations    180
    
```

DMREG Target Profile

The Target Profile table lists the target categories, their ordered values, and their total frequencies for the given data.

Target Profile

Ordered Value	c	Total Frequency
1	1	8
2	2	59
3	3	113

DMREG Optimization Table

The Optimization table provides a summary of the Newton-Raphson Ridge optimization results.

The DMREG Procedure

Newton-Raphson Ridge Optimization

Without Parameter Scaling

Parameter Estimates 4

Optimization Start

```

Active Constraints          0    Objective Function    143.32716812
Max Abs Gradient Element  2.5948717949
    
```

Iter	Rest arts	Func Calls	Act Con	Objective Function	Obj Fun Change	Max Abs Gradient Element	Ridge	Actual Over Pred Change
1	0	2	0	142.85145	0.4757	0.1180	0	1.001
2	0	3	0	142.85125	0.000198	0.000051	0	1.000
3	0	4	0	142.85125	6.42E-11	1.26E-11	0	1.001

Optimization Results

Iterations	3	Function Calls	6
Hessian Calls	4	Active Constraints	0
Objective Function	142.8512536	Max Abs Gradient Element	1.260375E-11
Ridge	0	Actual Over Pred Change	1.0005790385

Convergence criterion (GCONV=1E-6) satisfied.

DMREG Model Fitting Information and Testing Global Null Hypothesis Beta=0

The Model Fitting Information and Testing Global Null Hypothesis Beta=0 table contains the negative of twice the log likelihood (-2 LOG L) for the fitted model. Results of the likelihood ratio test for testing the joint significance of the explanatory inputs are also printed in the table.

Likelihood Ratio Test for Global Null Hypothesis: BETA=0

-2 Log Likelihood		Likelihood		
Intercept Only	Intercept & Covariates	Ratio Chi-Square	DF	Pr > ChiSq
286.654	285.703	0.9518	2	0.6213

DMREG Analysis of Maximum Likelihood Estimates

The Analysis of Maximum Likelihood Estimates table lists the parameter estimates, their standard errors, and the results of the Wald test for the individual parameters. A standardized estimate for each slope parameter and the odds ratio for each estimate is also printed. An odds ratio is obtained by exponentiating the corresponding parameter estimate.

Analysis of Maximum Likelihood Estimates

Parameter	DF	Estimate	Standard Error	Wald Chi-Square	Pr > ChiSq	Standardized Estimate	Exp(Est)
Intercept 1	1	-3.0449	0.5918	26.47	<.0001		0.048
Intercept 2	1	-0.4945	0.4931	1.01	0.3159		0.610
x	1	-0.0166	0.0222	0.56	0.4561	-0.0633	0.984
y	1	0.0231	0.0367	0.39	0.5298	0.0530	1.023

DMREG Odds Ratio Estimates

The Odd Ratio Estimates table lists the odd ratios for the explanatory inputs. The odd ratio estimates provide the change in odds for a unit increase in each input.

Linear-Logistic Regression with Ordinal Target

The DMREG Procedure

Odds Ratio Estimates

Effect	Point Estimate
x	0.984
y	1.023

PROC PRINT Report of Selected Fit Statistics for the Training Data Set

The misclassification rate for the training data set is 37.22%.

Linear-Logistic Regression with Ordinal Target
Fit Statistics for the Training Data Set

Train: Akaike's Information Criterion	Train: Maximum Absolute Error	Train: Root Final Prediction Error	Train: Misclassification Rate
293.703	0.95656	0.41066	0.37222

PROC FREQ Misclassification Table for the Training Data

All observations in the training data are classified into the C=3 level. The linear model is not adequate.

Linear-Logistic Regression with Ordinal Target
Misclassification Table: Training Data

The FREQ Procedure

Table of F_c by I_c

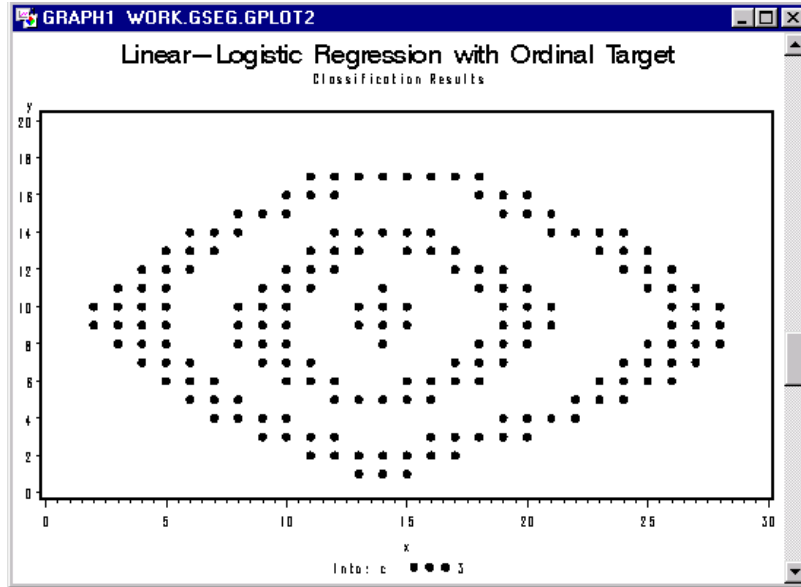
```

F_c(From: c)
      I_c(Into: c)
Frequency|
Percent  |
Row Pct  |
Col Pct  |3          | Total
|||
1         |      8    |      8
          |     4.44  |     4.44
          |    100.00 |
          |     4.44  |
|||
2         |     59    |     59
          |    32.78  |    32.78
          |    100.00 |
          |    32.78  |
|||
3         |    113    |    113
          |    62.78  |    62.78
          |    100.00 |
          |    62.78  |
|||
    
```

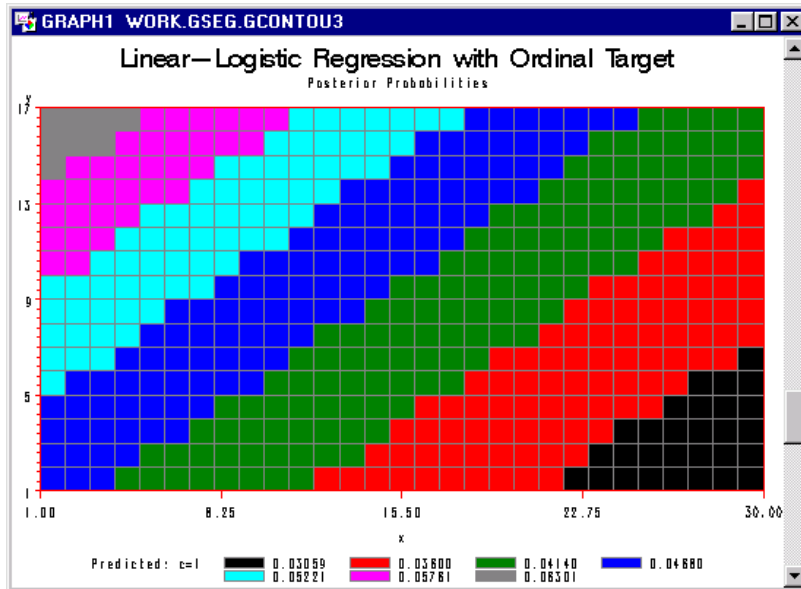
Total	180	180
	100.00	100.00

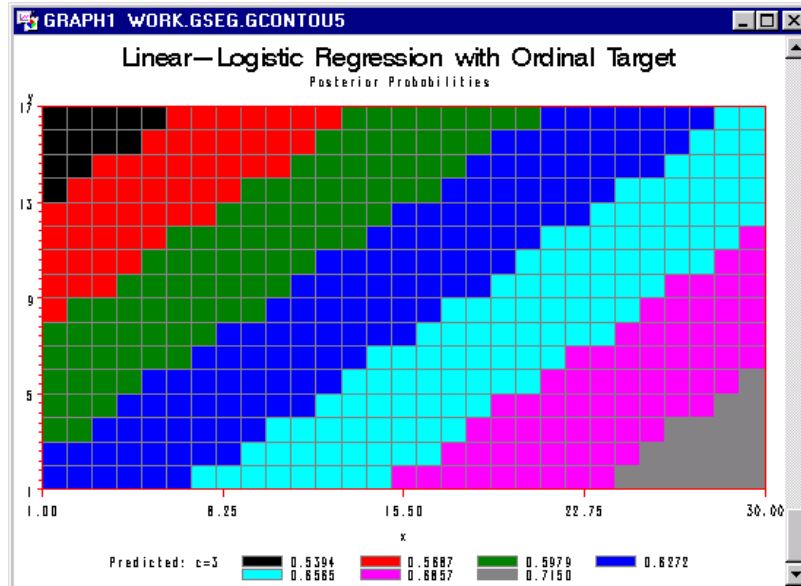
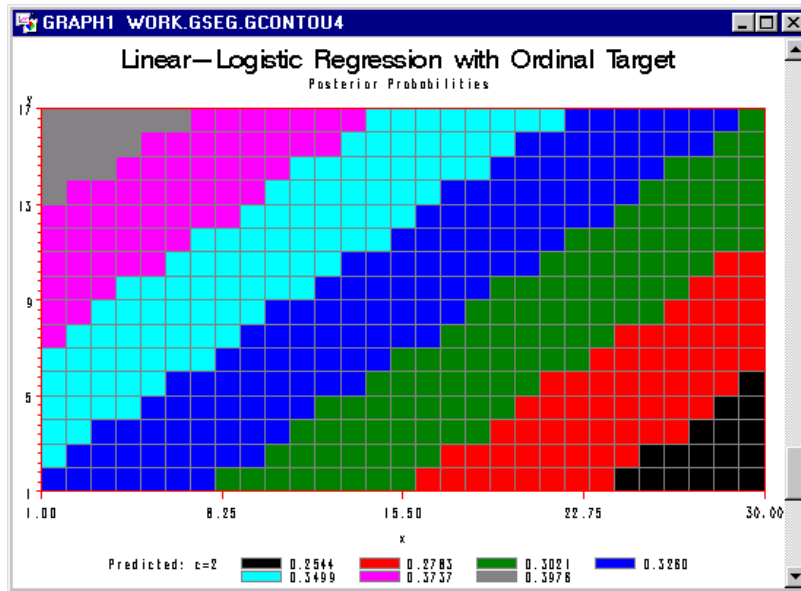
PROC GPLOT Plot of the Classification Results

The target classes are not linearly separable.



PROC GCONTOUR Plots of the Posterior Probabilities





Quadratic-Logistic Program

The model statement specifies the quadratic-logistic model. The vertical bars indicate that interactions of the specified inputs should be generated. “@2” indicates that only interactions up to the second order should be used. The option MISCONV= 0.1 stops the optimization at the iteration where the misclassification rate is less than or equal to one.

```
proc dmreg data=sampsio.dmdring dmdbcats=sampsio.dmdring;
  class c;
  model c=x|x|y|y @2/misconv=0.1;
  score out=qout outfit=qfit;
  score data=sampsio.dmsring nodmdb out=qgridout;
  title1 'Quadratic-Logistic Regression with Ordinal Target';
run;
```

PROC PRINT produces a report of selected fit statistics for the training data.

```
proc print data=qfit noobs label;
  var _aic_ _max_ _rfpe_ _misc_;
  title2 'Fit Statistics for the Training Data Set';
run;
```

PROC FREQ creates a report of the misclassification matrix for the training data set.

```
proc freq data=qout;
  tables f_c*i_c;
  title2 'Misclassification Table: Training Data';
run;
```

PROC GPLOT plots the classification results for the training data set.

```
proc gplot data=qout;
  plot y*x=i_c / haxis=axis1 vaxis=axis2;
  symbol c=black i=none v=dot;
  symbol2 c=red i=none v=square;
  symbol3 c=green i=none v=triangle;
  axis1 c=black width=2.5 order=(0 to 30 by 5);
  axis2 c=black width=2.5 minor=none order=(0 to 20 by 2);
  title2 'Classification Results';
run;
```

PROC GCONTOUR plots the posterior probabilities.

```
proc gcontour data=qgridout;
  plot y*x=p_c1 / pattern ctext=black outline=gray;
  plot y*x=p_c2 / pattern ctext=black outline=gray;;
  plot y*x=p_c3 / pattern ctext=black outline=gray;;
  title2 'Posterior Probabilities';
  pattern v=msolid;
  legend frame;
run;
```

Quadratic-Logistic Output

DMREG Output

The DMREG Procedure

Model Information

Training Data Set	SAMPSIO.DMDRING.DATA
DMDB Catalog	SAMPSIO.DMDRING
Target Variable	c
Target Measurement Level	Ordinal
Number of Target Categories	3
Error	MBernoulli
Link Function	Logit
Number of Model Parameters	7
Number of Observations	180

Target Profile

Ordered Value	c	Total Frequency
1	1	8
2	2	59
3	3	113

Newton-Raphson Ridge Optimization

Without Parameter Scaling

Parameter Estimates

7

Optimization Start

Active Constraints 0 Objective Function 143.32716812
 Max Abs Gradient Element 8.4022792023

Iter	Rest arts	Func Calls	Act Con	Objective Function	Obj Fun Change	Max Abs Gradient Element	Ridge	Actual Over Pred Change
1	0	2	0	50.31687	93.0103	8.4596	0	1.095
2	0	3	0	26.02018	24.2967	2.9573	0	1.292
3	0	4	0	13.98860	12.0316	1.1136	0	1.306
4	0	5	0	7.43545	6.5532	0.5352	0	1.312
5	0	6	0	3.75004	3.6854	0.3031	0	1.310
6	0	7	0	1.75436	1.9957	0.1649	0	1.301
7	0	8	0	0.75995	0.9944	0.0776	0	1.290
8	0	9	0	0.30934	0.4506	0.0323	0	1.280
9	0	10	0	0.12182	0.1875	0.0125	0	1.274

Optimization Results

Iterations 9 Function Calls 12
 Hessian Calls 10 Active Constraints 0
 Objective Function 0.121823985 Max Abs Gradient Element 0.0125231148
 Ridge 0 Actual Over Pred Change 1.2737233683

Convergence criterion (ABSCONV=0.1433271681) satisfied.

NOTE: At least one element of the gradient is greater than 1e-3.

Likelihood Ratio Test for Global Null Hypothesis: BETA=0

-2 Log Likelihood Intercept Only	Likelihood Intercept & Covariates	Likelihood Ratio Chi-Square	DF	Pr > ChiSq
----------------------------------	-----------------------------------	-----------------------------	----	------------

286.654 0.244 286.4107 5 <.0001

Quadratic-Logistic Regression with Ordinal Target

The DMREG Procedure

Analysis of Maximum Likelihood Estimates

Parameter	DF	Estimate	Standard Error	Wald Chi-Square	Pr > ChiSq	Standardized Estimate	Exp(Est)
Intercept 1	1	-232.2	236.4	0.96	0.3260		0.000
Intercept 2	1	-197.4	188.0	1.10	0.2939		0.000
x	1	18.4897	16.3812	1.27	0.2590	70.7018	999.000
x*x	1	-0.6584	0.4552	2.09	0.1481		0.518
y	1	22.5413	31.5097	0.51	0.4744	51.8034	999.000
x*y	1	0.0469	0.7657	0.00	0.9512		1.048
y*y	1	-1.2320	1.3963	0.78	0.3776		0.292

PROC PRINT Report of Selected Fit Statistics for the Training Data

Note that the training misclassification rate is 0. All cases are correctly classified by the quadratic-logistic model.

Quadratic-Logistic Regression with Ordinal Target
Fit Statistics for the Training Data Set

Train: Akaike's Information Criterion	Train: Maximum Absolute Error	Train: Root Final Prediction Error	Train: Misclassification Rate
14.2436	0.023582	.002403151	0

PROC FREQ Misclassification Table for the Training Data

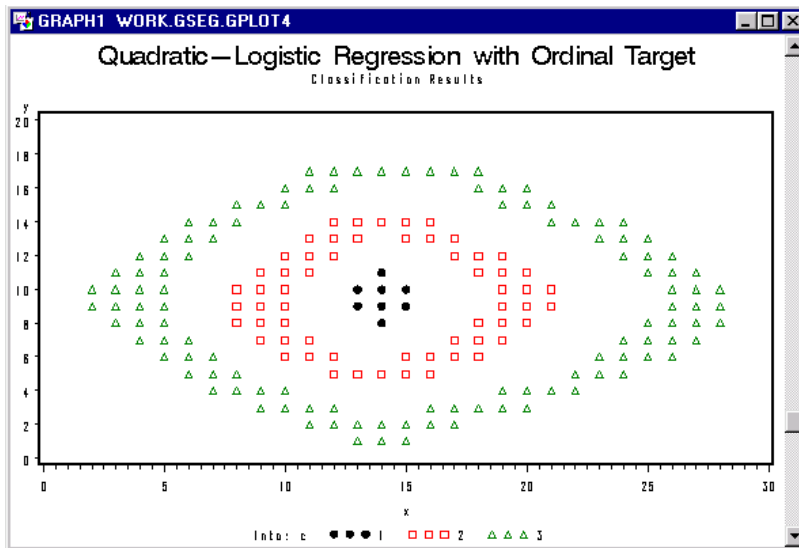
The FREQ Procedure

Table of F_c by I_c

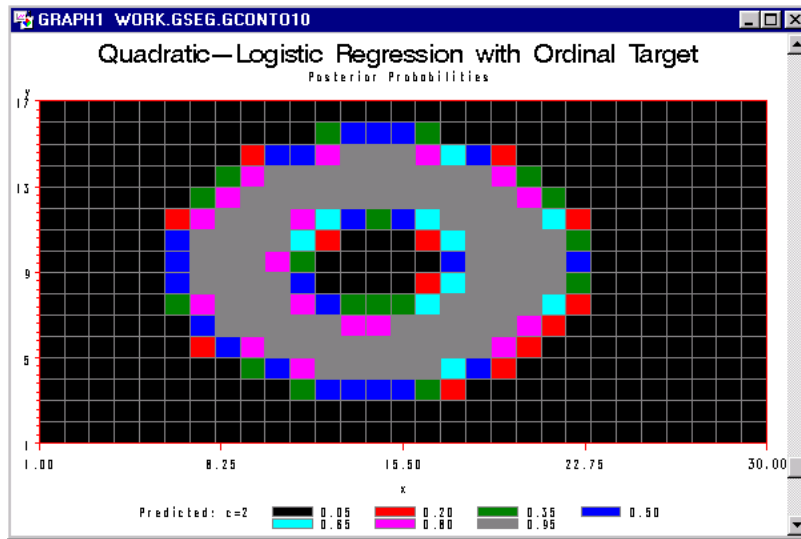
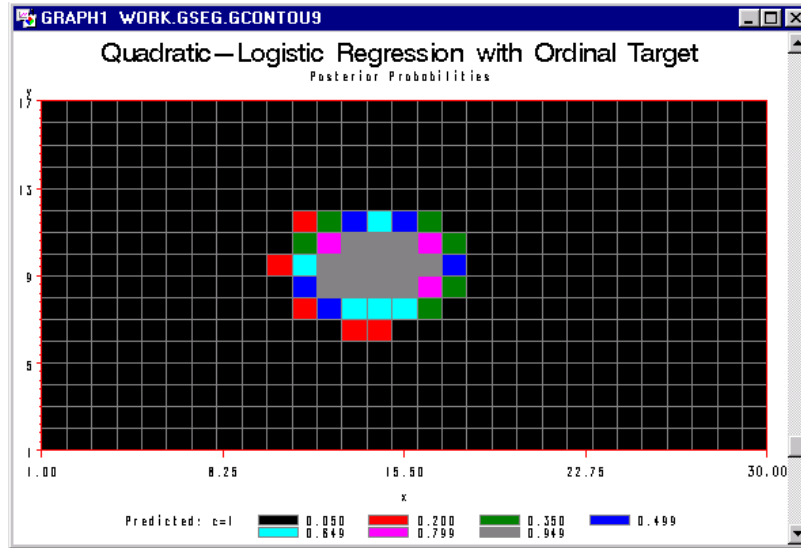
Freq	Percent	Row Pct	Col Pct	1	2	3	Total
1				8	0	0	8

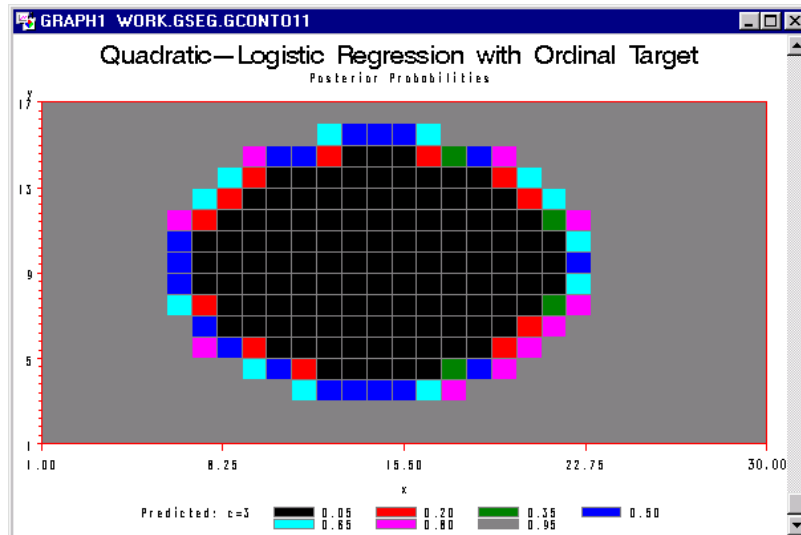
	4.44	0.00	0.00	4.44
	100.00	0.00	0.00	
	100.00	0.00	0.00	
2	0	59	0	59
	0.00	32.78	0.00	32.78
	0.00	100.00	0.00	
	0.00	100.00	0.00	
3	0	0	113	113
	0.00	0.00	62.78	62.78
	0.00	0.00	100.00	
	0.00	0.00	100.00	
Total	8	59	113	180
	4.44	32.78	62.78	100.00

PROC Gplot Plot of the Classification Results



PROC GCONTOUR Plots of the Posterior Probabilities





Example 2: Performing a Stepwise OLS Regression (DMREG Baseball Data)

Features

- Stepwise Regression using the SBC selection criterion
- Scoring a Test Data Set with the Score statement
- Outputting Fit Statistics
- Creating Diagnostic Plots

This example demonstrates how to perform a stepwise OLS regression using the DMREG procedure. The example training data set SAMPSIO.DMLBASE (baseball data set) contains performance measures and salary levels for regular hitters and leading substitute hitters in major league baseball for the year 1986 (Collier 1987). There is one observation per hitter. The continuous response variable is the log of the players salary (logsalar). The SAMPSIO.DMTBASE data set is a test data set which is scored using the scoring formula from the trained model. The SAMPSIO.DMLBASE and SAMPSIO.DMTBASE data sets and the SAMPSIO.DMDBASE data mining catalog are stored in the sample library.

Program

The PROC DMREG statement invokes the procedure. The DATA= option identifies the training data set that is used to fit the model. The DMDBCAT= option identifies the training data catalog.

```
proc dmreg data=sampsio.dmlbase dmdbcat=sampsio.dmdbase
```

The TESTDATA= option identifies the test data set. The OUTEST= option creates the output data set containing estimates and fit statistics.

```
testdata=sampsio.dmtbase outest=regest;
```

The CLASS statement specifies the categorical variables to be used in the regression analysis.

```
class league division position;
```

The MODEL statement specifies the linear model. The ERROR=normal model option specifies to use the normal error distribution. The CHOOSE=SBC model option specifies to choose the model subset with the smallest Schwarz Bayesian criterion.

```
model logsalar = no_atbat no_hits no_home no_runs no_rbi no_bb
                yr_major cr_atbat cr_hits cr_home cr_runs
                cr_rbi cr_bb league division position no_outs
                no_assts no_error

                / error=normal
                choose=sbc
```

The MODEL option SELECTION=STEPWISE specifies to use the stepwise variable selection method. Stepwise selection systematically adds and deletes inputs from the model based on the SLENTRY= and SLSTAY= significance levels. The subset models are created based on the SLENTRY and SLSTAY significance levels, but the model that is chosen is based solely on the subset model that has the smallest SBC criterion.

```
selection=stepwise
slentry=0.25 slstay=0.25;
```

The SCORE statement specifies the data set that you want to score in conjunction with training. The DATA= option identifies the score data set (for this example, the test data set).

```
score data=sampsio.dmtbase
```

The OUT=option identifies the output data set that contains estimates and fit statistics for the scored data set. The RENAME=option enables you to rename variables in the OUT= data set.

```
out=regout(rename=(p_logsalar=predict r_logsalar=residual));
title 'Output from the DMREG Procedure';
run;
```

PROC PRINT produces a report of selected variables from the OUTEST= data set.

```
proc print data=regest noobs label;
  var _step_ _chosen_ _sbc_ _mse_ _averr_ _tmse_ _taverr_;
  where _type_ = 'PARMS';
  title 'Partial Listing of the OUTEST= Data Set';
run;
```

PROC GPLOT produces diagnostic plots of the scored test data. The first PLOT statement plots the response versus the predicted values.

```
proc gplot data=regout;
```

```

plot logsalar*predict / haxis=axis1 vaxis=axis2 frame;
symbol c=black i=none v=dot h=3 pct;
axis1 c=black width=2.5;
axis2 c=black width=2.5;
title 'Diagnostic Plots for the Scored Baseball Data';

```

The second PLOT statement plots the residuals versus the predicted values.

```

plot residual*predict / haxis=axis1 vaxis=axis2;
run;
quit;

```

Output

Summary Profile Information

The first section of the output lists the two-level data set name, the response variable, the number of observations, the error distribution, and the link function.

Output from the DMREG Procedure

Model Information

Training Data Set	SAMPSIO.DMDBASE.DATA
DMDB Catalog	SAMPSIO.DMDBASE
Target Variable	logsalar (Log Salary)
Target Measurement Level	Interval
Error	Normal
Link Function	Identity
Number of Model Parameters	38
Number of Observations	163

Design Matrix For Classification Effects

The DMREG procedure uses a deviation from the means method to generate the design matrix for the classification inputs. Each row of the design matrix is generated by a unique combination of the nominal input values. Each column of the design matrix corresponds to a model parameter.

If a nominal variable SWING has k levels (3), then its main effect has $k-1$ (2) degrees of freedom, and the design matrix has $k-1$ (2) columns that correspond to the first $k-1$ levels. The i th column contains a 1 in the i th row, a -1 in the last row, and 0 everywhere else. If α_i denotes the parameter that corresponds to the i th level of variable SWING, then $k-1$ columns yield estimates of the independent parameter $\alpha_1, \alpha_2, \dots, \alpha_{k-1}$. The last parameter is not needed because DMREG constrains the k parameters to sum to 0. Crossed effects, such as SWING*LEAGUE, are formed by the horizontal direct product of main effects.

Table 9.8 Design Matrix Classification Table

Data Levels for SWING	Design Columns		
Left	1	0	
Right	0	1	
Switch	-1	-1	

The printing of the design matrix can be suppressed by using the MODEL statement option NODESIGNPRINT.

Class Level Information

Class	Value	Design Variables																
league	American	1																
	National	-1																
division	East	1																
	West	-1																
position	1B	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	23	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	2B	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	2S	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
	3B	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
	3O	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
	3S	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
	C	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
	CD	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
	CF	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
	DH	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
	DO	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
	LF	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
	O1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
	OF	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
	OS	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	RF	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	S3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	SS	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	UT		-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1

Class Level Information

Class	Value	Design Variables			
league	American				
	National				
division	East				
	West				
position	1B	0	0	0	0
	23	0	0	0	0

Class Level Information

Class	Value	Design Variables			
	2B	0	0	0	0
	2S	0	0	0	0

3B	0	0	0	0
3O	0	0	0	0
3S	0	0	0	0
C	0	0	0	0
CD	0	0	0	0
CF	0	0	0	0
DH	0	0	0	0
DO	0	0	0	0
LF	0	0	0	0
O1	0	0	0	0
OF	0	0	0	0
OS	1	0	0	0
RF	0	1	0	0
S3	0	0	1	0
SS	0	0	0	1
UT	-1	-1	-1	-1

Model Fitting Information for Each Subset Model of the Stepwise Selection Process

For brevity, only steps number 5 and 8 from the stepwise selection process are listed in the following output. Step number 5 contains the model that has the smallest SBC statistic. This model is used to score the test data set. Because no other inputs met the condition for removal from the model and no other variables met the criterion for addition to the model, the stepwise algorithm terminates after step number 8.

For each model subset of the stepwise modeling process, DMREG provides:

- 1 An analysis of variance table which lists degrees of freedom, sums of squares, mean squares, the Model F, and its associated p -value.
- 2 Model fitting information which contains the following statistics that enable you to assess the fit of each stepwise model:
 - *R-square* – which is calculated as $1 - \frac{SSE}{SST}$, where SSE is the error sums of squares and SST is the total sums of squares. The R^2 statistic ranges from 0 to 1. Models that have large values of R^2 are preferred. For step number 8, the regression equation explains 60.17% of the variability in the target.
 - *Adj R-sq* – the Adj- R^2 is an alternative criterion to the R^2 statistic that is adjusted for the number of parameters in the model. This statistic is calculated as $1 - \left((n - i) (1 - R^2) / (n - p) \right)$, where n is the number cases, and i is an indicator variable that is 1 if the model includes an intercept and 0, otherwise. Large differences between the R^2 and the Adj- R^2 values for a given model can indicate that you have used too many inputs in the model.
 - *AIC* – Akaike’s Information Criterion, which is a goodness-of-fit statistic that you can use to compare one model to another. Lower values indicate a more desirable model. It is calculated as $(n) \ln \left(\frac{SSE}{n} \right) + 2p$, where n is the number of cases, SSE is the error sums of squares, and p is the number of model parameters.
 - *BIC* – Bayesian Information Criterion is another goodness-of-fit statistic that is calculated as $(n) \ln (SSE/n) + 2(p + 2)q - 2q^2$, where $q = MSE / (SSE/n)$ (MSE is obtained from the full model). Smaller BIC values are preferred.
 - *SBC* – Schwarz’s Bayesian Criterion is another goodness-of-fit statistic that is calculated as $(n) \ln \left(\frac{SSE}{n} \right) + (p) \ln (n)$. Models that have small SBC values are preferred. Because the CHOOSE=SBC option was specified, DMREG selects the model that has the smallest SBC value.

- $C(p)$ – Mallow’s C_p Statistic enables you to determine whether your model is under or overspecified. This statistic is calculated as $\left(\frac{SSE(p)}{MSE}\right) - (n - 2p)$, where $SSE(p)$ is the error sums of squares for the subset model with p parameters including the intercept if any, MSE is the error mean square for the full model, and n is the number of cases. For any subset model $C(p) > p$, there is evidence of bias due to an incompletely specified model (your model might not contain enough inputs). However, if there are values of $C(p) < p$, the full model is said to be overspecified. When the right model is chosen, the parameter estimates are unbiased, and this is reflected in $C_p < p$ or at least near p .

3 Analysis of effects and parameter estimates that contains the effect, degrees of freedom, parameter estimate, standard error, type 3 sums of squares, F -value and the corresponding p -value.

Step 5: Effect no_bb entered.

Analysis of Variance

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	5	72.833938	14.566788	43.41	<.0001
Error	157	52.678742	0.335533		
Corrected Total	162	125.512680			

Model Fit Statistics

R-Square	0.5803	Adj R-Sq	0.5669
AIC	-172.1147	BIC	-169.6060
SBC	-153.5522	C(p)	5.3257

Type 3 Analysis of Effects

Effect	DF	Sum of Squares	F Value	Pr > F
no_hits	1	5.6348	16.79	<.0001
no_bb	1	1.7104	5.10	0.0253
cr_hits	1	27.5518	82.11	<.0001
no_outs	1	2.5542	7.61	0.0065
no_error	1	2.0405	6.08	0.0147

Analysis of Maximum Likelihood Estimates

Parameter	DF	Estimate	Standard Error	t Value	Pr > t
Intercept	1	4.5835	0.1290	35.52	<.0001
no_hits	1	0.00562	0.00137	4.10	<.0001
no_bb	1	0.00602	0.00267	2.26	0.0253

cr_hits	1	0.000701	0.000077	9.06	<.0001
no_outs	1	0.000453	0.000164	2.76	0.0065
no_error	1	-0.0180	0.00729	-2.47	0.0147

Step 8: Effect no_rbi entered.

Output from the DMREG Procedure 70
10:26 Tuesday, May 6, 2008

The DMREG Procedure

Analysis of Variance

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	8	75.525299	9.440662	29.08	<.0001
Error	154	49.987381	0.324593		
Corrected Total	162	125.512680			

Model Fit Statistics

R-Square	0.6017	Adj R-Sq	0.5810
AIC	-174.6627	BIC	-170.9032
SBC	-146.8189	C(p)	3.3390

Type 3 Analysis of Effects

Effect	DF	Sum of Squares	F Value	Pr > F
no_hits	1	1.3988	4.31	0.0396
no_rbi	1	0.4606	1.42	0.2354
no_bb	1	2.1274	6.55	0.0114
yr_major	1	1.6517	5.09	0.0255
cr_hits	1	1.6766	5.17	0.0244
cr_bb	1	0.7491	2.31	0.1308
no_outs	1	2.3525	7.25	0.0079
no_error	1	1.3915	4.29	0.0401

Analysis of Maximum Likelihood Estimates

Parameter	DF	Estimate	Standard Error	t Value	Pr > t
Intercept	1	4.2938	0.1834	23.41	<.0001
no_hits	1	0.00424	0.00204	2.08	0.0396
no_rbi	1	0.00362	0.00304	1.19	0.2354

no_bb	1	0.00880	0.00344	2.56	0.0114
yr_major	1	0.0580	0.0257	2.26	0.0255
cr_hits	1	0.000571	0.000251	2.27	0.0244
cr_bb	1	-0.00084	0.000551	-1.52	0.1308
no_outs	1	0.000439	0.000163	2.69	0.0079
no_error	1	-0.0152	0.00733	-2.07	0.0401

NOTE: No (additional) effects met the 0.25 significance level for entry into the model.

Summary of the Stepwise Selection Process

The Summary of Stepwise Selection section provides the step number, the explanatory input or inputs entered or removed at each step, the F statistic, and the corresponding p -value on which the entry or removal of the input is based. For this example, 8 of the 19 original inputs met the 0.25 entry and stay probability values.

Summary of Stepwise Selection

Step	Effect Entered	DF	Number		F Value	Pr > F
			In			
1	cr_hits	1	1		97.16	<.0001
2	no_hits	1	2		49.78	<.0001
3	no_outs	1	3		8.71	0.0036
4	no_error	1	4		6.42	0.0123
5	no_bb	1	5		5.10	0.0253
6	yr_major	1	6		4.38	0.0379
7	cr_bb	1	7		2.43	0.1210
8	no_rbi	1	8		1.42	0.2354

List Report of Selected Variables in the OUTEST= data set

The example PROC PRINT report of the OUTEST= data set lists selected fit statistics for the training and test data sets. The default OUTEST= data set contains three observations for each step number. These observations are distinguished by value of the _TYPE_ variable:

- _TYPE_ = "PARMS" - contains parameter estimates and the fit statistics.
- _TYPE_ = "T" - contains the t -values for the parameter estimates.
- _TYPE_ = "P" - contains the p -values for the parameter estimates.

Because a WHERE statement was used to select only values of TYPE = "PARMS", this report contains one observation per step number. An additional observation is displayed that identifies the model chosen based on the SBC criterion (CHOOSE="SBC").

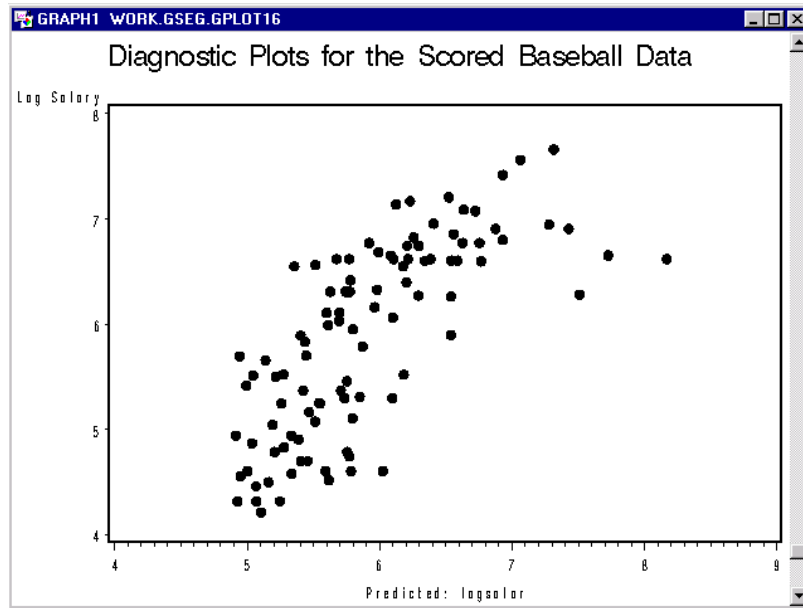
Partial Listing of the OUTEST= Data Set

Model Selection Step Number	Model Chosen Criterion	Train: Schwarz's Bayesian Criterion	Train: Mean Square Error	Train: Average Error Function	Test: Mean Square Error	Test: Average Error Function
0		-37.505	0.77477	0.77002	0.81858	0.81858

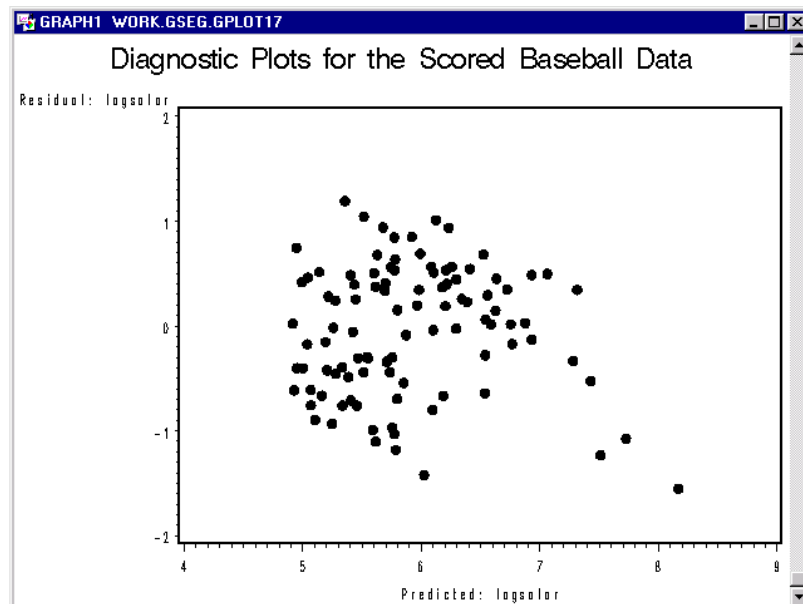
1		-109.377	0.48618	0.48021	0.42598	0.42598
2		-148.441	0.37312	0.36625	0.34632	0.34632
3		-152.041	0.35597	0.34723	0.37438	0.37438
4		-153.438	0.34424	0.33368	0.38553	0.38553
5		-153.552	0.33553	0.32318	0.37563	0.37563
6		-152.974	0.32846	0.31435	0.34713	0.34713
7		-150.418	0.32547	0.30950	0.35231	0.35231
8		-146.819	0.32459	0.30667	0.35978	0.35978
5	SBC	-153.552	0.33553	0.32318	0.37563	0.3756

GPLOT Diagnostic Plots for the Scored Baseball Test Data

Plot of the log of salary versus the predicted log of salary.



Plot of the residual values versus the predicted log of salary.



Example 3: Comparison of the DMREG and LOGISTIC Procedures for Modeling a Categorical Target

DMREG and LOGISTIC procedures fit the same models for a categorical target. Both procedures have the CLASS statement for specifying categorical input variables, and both use the deviation from the mean coding as the default parameterization for a CLASS input variable.

There are many differences between the two procedures, both in syntax and in features. For example, to specify the GLM parameterization of CLASS variables, you specify the MODEL statement option CODING= GLM in DMREG, but in LOGISTIC, you specify the CLASS statement option PARAM= GLM. You are required to specify a DMDB catalog of input data in DMREG, but no such requirement is needed in LOGISTIC. DMREG produces DATA step scoring code, but LOGISTIC does not.

In terms of training a model, you might expect the estimates from both procedures to be identical. Often the estimates between the two procedures are very close but not necessarily identical for a number of reasons. DMREG and LOGISTIC procedures do not use the same routines to carry out the optimization, and the convergence criterion and optimization technique used might not be the same. However, discrepancies of the parameter estimates between the two procedures would not make any difference in prediction. Consider the data set SAMPSIO.HMEQ, which contains fictitious mortgage data where each case represents an applicant for a home equity loan. The binary target BAD represents whether an applicant eventually defaults the loan (BAD=1 for delinquent and BAD=0 otherwise). For illustration, one interval input variable (DEBTINC) and one categorical input variable (JOB) are used to predict the target BAD.

To model the probability of BAD= 1 in the LOGISTIC procedure, you can use the DESCENDING option in the PROC LOGISTIC statement (to model the last level of the binary target instead of the first level).

```
proc logistic data=sampsio.hmeq descending;
  class bad job;
  model bad=job debtinc;
  score data=sampsio.hmeq(where=(job^=' ' and debtinc^=.)) out=logitout;
  title 'LOGISTIC Analysis of Home Equity Data';
run;
```

To train the same model using DMREG, you need the DMDB catalog which is created in the DMDB procedure:

```
proc dmdb batch data=sampsio.hmeq dmdbcat=dm_cat;
  var debtinc;
  class bad(desc) job;
  target bad;
run;
```

Because the order of target BAD was set to descending in the DMDB catalog, DMREG also models the probability BAD=1.

```
proc dmreg data=sampsio.hmeq dmdbcat=dm_cat;
  class bad job;
  model bad=job debtinc;
  score data=sampsio.hmeq(where=(job^=' ' and debtinc^=.)) out=dmregout;
```

```

title 'DMREG Analysis of Home Equity Data';
run;

```

PROC LOGISTIC does not score observations with missing inputs, but PROC DMREG does. The SCORE statement is used in each procedure to score the subset of the training data that do not have missing values in the input variables. The COMPARE procedure is used to compare the predicted probabilities computed by the two procedures.

```

proc compare data=dmregout compare=logitout criterion=1e-4;
  var P_BAD1 P_BAD0;
run;

```

Output

The printed output of the LOGISTIC and DMREG procedures is as follows. Both procedures use the default deviation from the mean coding for the CLASS input variable as shown in the “Class Level Information” table. Parameter estimates and standard errors from both procedures are almost identical. The discrepancies in the estimates are in the fourth decimal places and should give nearly identical scoring results. PROC COMPARE found no differences in the predicted probabilities from the scoring results between the two procedures for the training data.

The LOGISTIC Procedure

Model Information

Data Set	SAMPSIO.HMEQ
Response Variable	BAD
Number of Response Levels	2
Model	binary logit
Optimization Technique	Fisher's scoring

Number of Observations Read	5960
Number of Observations Used	4459

Response Profile

Ordered Value	BAD	Total Frequency
1	1	397
2	0	4062

Probability modeled is BAD=1.

Class Level Information

Class	Value	Design Variables				
JOB	Mgr	1	0	0	0	0
	Office	0	1	0	0	0
	Other	0	0	1	0	0

ProfExe	0	0	0	1	0
Sales	0	0	0	0	1
Self	-1	-1	-1	-1	-1

Model Convergence Status

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

Model Fit Statistics

Criterion	Intercept Only	Intercept and Covariates
AIC	2680.038	2502.150
SC	2686.440	2546.969
-2 Log L	2678.038	2488.150

The LOGISTIC Procedure

Testing Global Null Hypothesis: BETA=0

Test	Chi-Square	DF	Pr > ChiSq
Likelihood Ratio	189.8875	6	<.0001
Score	212.7650	6	<.0001
Wald	144.4931	6	<.0001

Type 3 Analysis of Effects

Effect	DF	Wald Chi-Square	Pr > ChiSq
JOB	5	21.7341	0.0006
DEBTINC	1	119.8652	<.0001

Analysis of Maximum Likelihood Estimates

Parameter	DF	Estimate	Standard Error	Wald Chi-Square	Pr > ChiSq
Intercept	1	-5.3310	0.3074	300.7599	<.0001
JOB Mgr	1	-0.0802	0.1408	0.3244	0.5690
JOB Office	1	-0.5386	0.1465	13.5127	0.0002
JOB Other	1	0.0354	0.1027	0.1189	0.7303
JOB ProfExe	1	-0.2757	0.1314	4.4034	0.0359
JOB Sales	1	0.7201	0.2541	8.0299	0.0046
DEBTINC	1	0.0868	0.00792	119.8652	<.0001

Odds Ratio Estimates

Effect	Point Estimate	95% Wald Confidence Limits
JOB Mgr vs Self	0.803	0.436 1.480
JOB Office vs Self	0.508	0.273 0.943
JOB Other vs Self	0.902	0.512 1.589
JOB ProfExe vs Self	0.661	0.363 1.203
JOB Sales vs Self	1.788	0.807 3.959
DEBTINC	1.091	1.074 1.108

Association of Predicted Probabilities and Observed Responses

Percent Concordant	65.6	Somers' D	0.323
Percent Discordant	33.4	Gamma	0.326
Percent Tied	1.0	Tau-a	0.052
Pairs	1612614	c	0.661

The DMREG Procedure

Model Information

Training Data Set	SAMPSIO.HMEQ.DATA
DMDB Catalog	WORK.DM_CAT
Target Variable	BAD
Target Measurement Level	Ordinal
Number of Target Categories	2
Error	MBernoulli
Link Function	Logit
Number of Model Parameters	7
Number of Observations	4459

Target Profile

Ordered Value	BAD	Total Frequency
1	1	397
2	0	4062

Class Level Information

Class	Value	Design Variables				
JOB	Mgr	1	0	0	0	0
	Office	0	1	0	0	0
	Other	0	0	1	0	0
	ProfExe	0	0	0	1	0
	Sales	0	0	0	0	1
	Self	-1	-1	-1	-1	-1

The DMREG Procedure

Newton-Raphson Ridge Optimization

Without Parameter Scaling

Parameter Estimates

7

Optimization Start

Active Constraints 0 Objective Function 1339.0188708
 Max Abs Gradient Element 28.60461987

Iter	Rest arts	Func Calls	Act Con	Objective Function	Obj Fun Change	Max Abs Gradient Element	Ridge	Actual Over Pred Change
1	0	2	0	1251	88.5028	64.8620	0	0.832
2	0	3	0	1244	6.4002	4.2148	0	1.048
3	0	4	0	1244	0.0407	0.0308	0	1.006
4	0	5	0	1244	4.097E-6	2.947E-6	0	1.000

Optimization Results

Iterations 4 Function Calls 7
 Hessian Calls 5 Active Constraints 0
 Objective Function 1244.0751361 Max Abs Gradient Element 2.9467434E-6
 Ridge 0 Actual Over Pred Change 1.0000861277

Convergence criterion (GCONV=1E-6) satisfied.

Likelihood Ratio Test for Global Null Hypothesis: BETA=0

-2 Log Likelihood Intercept Only	Intercept & Covariates	Likelihood Ratio Chi-Square	DF	Pr > ChiSq
2678.038	2488.150	189.8875	6	<.0001

Type 3 Analysis of Effects

Effect	DF	Wald Chi-Square	Pr > ChiSq
JOB	5	21.7259	0.0006
DEBTINC	1	119.8741	<.0001

The DMREG Procedure

Analysis of Maximum Likelihood Estimates

Parameter	DF	Estimate	Standard Error	Wald Chi-Square	Pr > ChiSq	Standardized Estimate
Intercept	1	-5.3315	0.3074	300.78	<.0001	
JOB Mgr	1	-0.0800	0.1408	0.32	0.5699	
JOB Office	1	-0.5386	0.1465	13.51	0.0002	
JOB Other	1	0.0356	0.1027	0.12	0.7290	
JOB ProfExe	1	-0.2756	0.1314	4.40	0.0359	
JOB Sales	1	0.7198	0.2542	8.02	0.0046	
DEBTINC	1	0.0868	0.00793	119.87	<.0001	0.4024

Analysis of Maximum Likelihood Estimates

Parameter	Exp(Est)
Intercept	0.005
JOB Mgr	0.923
JOB Office	0.584
JOB Other	1.036
JOB ProfExe	0.759
JOB Sales	2.054
DEBTINC	1.091

Odds Ratio Estimates

Effect	Point Estimate
JOB Mgr vs Self	0.804
JOB Office vs Self	0.508
JOB Other vs Self	0.902
JOB ProfExe vs Self	0.661
JOB Sales vs Self	1.788
DEBTINC	1.091

References

- Berry, M. J. A. and Linoff, G. (1997), *Data Mining Techniques for Marketing, Sales, and Customer Support*, New York: John Wiley and Sons, Inc.
- Cox, D. R. and Snell, E. J. (1989), *The Analysis of Binary Data*, 2nd Edition, London: Chapman and Hall.
- Dennis, J. E. and Mei, H.H.W. (1979), "Two New Unconstrained Optimization Algorithms that use Function and Gradient Values", *J. Optim. Theory Appl.*, 28, 453–482.
- Draper, N. and Smith, H. (1981), *Applied Regression Analysis*, 2nd Edition, New York: John Wiley and Sons, Inc.
- Little, R. J. A. and Rubin, D. B. (1987), *Statistical Analysis with Missing Data*, New York: John Wiley and Sons, Inc.
- Little, R. J. A. (1992), "Regression with Missing X's: A review," *Journal of the American Statistical Association*, 87, 1227–1237.

- McCullagh, P. and Nelder, J. A. (1989), *Generalized Linear Models*, 2nd Edition, New York: Chapman and Hall.
- Moré, J.J. (1978), “The Levenberg-Marquardt Algorithm: Implementation and Theory” in: G.A. Watson (ed.) *Lecture Notes in Mathematics 630*, Berlin-Heidelberg-NewYork: Springer Verlag.
- Rawlings, J. O. (1988), *Applied Regression Analysis: A Research Tool*, Pacific Grove, California: Wadsworth and Brooks/Cole Advanced Books and Software.
- SAS Institute Inc. (1995), *Logistic Regression Examples using the SAS System*, Version 6, 1st Edition, Cary, NC: SAS Institute Inc.
- SAS Institute Inc. (2002), *SAS/OR User’s Guide: Mathematical Programming*, in SAS OnlineDoc 9.1
- SAS Institute Inc. (2002), *SAS/STAT User’s Guide*, in SAS OnlineDoc 9.1