PROC DMNEURL: Approximation to PROC NEURAL

Purpose of PROC DMNEURL

In its current form, PROC DMNEURL tries to establish a nonlinear model for the prediction of a binary or interval scaled response variable (called *target* in data mining terminology). The approach will soon be extended to nominal and ordinal scaled response variables.

The algorithm used in DMNEURL was developed to overcome some problems of PROC NEURAL for data mining purposes, especially when the data set contains many highly collinear variables:

1. The nonlinear estimation problem in common Neural Networks is seriously underdetermined yielding to highly rankdeficient Hessian matrices and resulting in extremely slow convergence (close to linear) of nonlinear optimization algorithms.
   $\Longrightarrow$ Full-rank estimation.

2. Each function call in PROC NEURAL corresponds to a single run through the entire (training) data set and normally many function calls are needed for convergent nonlinear optimization with rankdeficient Hessians.
   $\Longrightarrow$ Optimization of discrete problem with all data incore.

3. Because the zero eigenvalues in a Hessian matrix correspond to long and very flat valleys in the shape of the objective function, the traditional Neural Net approach has serious problems to decide when an estimate is close to an appropriate solution and the optimization process can be terminated.
   $\Longrightarrow$ Quadratic convergence.

4. For the same reasons, the common Neural Net algorithms suffer from a high sensibility toward finding local rather than global optimal solutions and the optimization result often is very sensitive w.r.t. the starting point of the optimization.
   $\Longrightarrow$ Good starting point.

With PROC DMNEURL we deal with specified optimization problems (with full rank Hessian matrices) which have not many parameters and for which good starting points can be obtained. The convergence of the nonlinear optimizer is normally very fast, resulting mostly in less than 10 iterations per optimization. The function and derivative calls during the optimization do not need any passes through the data set, however, the search for obtaining good starting points and the final evaluations of the solutions (scoring of all observations) need passes through the data, as well as a number of preliminary tasks. In PROC DMNEURL we fit separately an entire set of about 8 activation functions and select the best result. Since the optimization processes for different activation functions do not depend on each other, the computer time could be reduced greatly by parallel processing.

Except for applications where PROC NEURAL would hit a local solution much worse than the global solution, it is not expected that PROC DMNEURL can beat PROC NEURAL in the precision of the prediction. However, for the applications we have run until now we found the results of PROC DMNEURL very close to those of PROC NEURAL. PROC DMNEURL will be faster than PROC NEURAL only for very large data sets. For small data sets, PROC NEURAL could be much faster than PROC DMNEURL, especially for an interval target. The most efficient application of PROC DMNEURL is the analysis of a binary target variable without FREQ and WEIGHT statement and without COST variables in the input data set.

Application: HMEQ Data Set: Binary Target BAD

To illustrate the use of PROC DMNEURL we choose the HMEQ data set:

```
libname sampsio '/sas/a612/dmine/sampsio';
proc dmdb batch data=sampsio.hmeq out=dmdbout dmdbcat=outcat;
  var LOAN MORTDUE VALUE YOJ DELINQ CLAGE NINQ CLNO DEBTINC;
  class BAD(DESC) REASON(ASC) JOB(ASC) DEROG(ASC);
  target BAD;
run;
```

When selecting the binary target variable BAD a typical run of PROC DMNEURL would be the following:

```
proc dmneurl data=dmdbout dmdbcat=outcat
     outclass=oclass outest=estout out=dsout outfit=ofit
     ptable maxcomp=3 maxstage=5;
 var LOAN MORTDUE VALUE REASON JOB YOJ DEROG DELINQ
     CLAGE NINQ CLNO DEBTINC;
 target BAD;
run;
```

The number of parameters p estimated in each stage of the optimization is $p = 2 * c + 1$, where c is the number of components that is selected at the stage. Since here $c = 3$ is specified with the MAXCOMP= option each optimization process estimates only $p = 7$ parameters.

First some general information is printed and the four moments of the numeric data set variables involved in the analysis:

```
                 The DMNEURL Procedure

         Binary Target                        BAD
         Number Observations                 5960
         NOBS w/o Missing Target             5960
```

Link Function	LOGIST
Selection Criterion	SSE
Optimization Criterion	SSE
Estimation Stages	5
Max. Number Components	3
Minimum R2 Value	0.000050
Number Grid Points	17

Response Profile for Target: BAD

Level	Nobs	Frequency	Weight
1	1189	1189	1189.000000
0	4771	4771	4771.000000

Variable	Mean	Std Dev	Skewness	Kurtosis
LOAN	18608	11207	2.02378	6.93259
MORTDUE	67350	44458	1.81448	6.48187
VALUE	99863	57386	3.05334	24.36280
YOJ	8.15130	7.57398	0.98846	0.37207
DELINQ	0.40570	1.12727	4.02315	23.56545
CLAGE	170.47634	85.81009	1.34341	7.59955
NINQ	1.08456	1.72867	2.62198	9.78651
CLNO	20.50285	10.13893	0.77505	1.15767
DEBTINC	26.59885	8.60175	2.85235	50.50404

For the first stage we select three eigenvectors corresponding to the 4th, 11th, and 2nd largest eigenvalues. Obviously, there is no relationship between

- the R^2 value which measures the prediction of the response (target) variable by each eigenvector
- and the eigenvalue corresponding to each eigenvector which measures the variance explained in the $X^T X$ data matrix.

Therefore, the eigenvalues are not used in the analysis of PROC DMNEURL and are printed only for curiosity.

Component Selection: SS(y) and R2 (SS_total=4771)

Comp	Eigval	R-Square	F Value	p-Value	SSE
4	9397.769045	0.017419	105.640645	<.0001	4687.893424
11	6327.041282	0.006317	38.550835	<.0001	4657.755732
2	13164	0.005931	36.408247	<.0001	4629.461194

The optimization history indicates a maximum of 11 iterations for the activation function LOGIST:

```
------------ Optimization Cycle (Stage=0) --------------
------------ Activation= SQUARE (Stage=0) --------------
NOTE: ABSGCONV convergence criterion satisfied.
SQUARE: Iter=5 Crit=0.06782364: SSE=808.457819 Acc= 81.6443
------------ Activation= TANH (Stage=0) ----------------
NOTE: ABSGCONV convergence criterion satisfied.
TANH: Iter=4 Crit=0.06802595: SSE=810.869323 Acc= 81.6275
------------ Activation= ARCTAN (Stage=0) --------------
NOTE: ABSGCONV convergence criterion satisfied.
ARCTAN: Iter=5 Crit=0.06795346: SSE=810.005204 Acc= 81.6611
------------ Activation= LOGIST (Stage=0) --------------
NOTE: ABSGCONV convergence criterion satisfied.
LOGIST: Iter=11 Crit=0.06802943: SSE= 810.91085 Acc= 81.6107
------------ Activation= GAUSS (Stage=0) ---------------
NOTE: ABSGCONV convergence criterion satisfied.
GAUSS: Iter=10 Crit=0.07727582: SSE=921.127726 Acc= 80.2517
------------ Activation= SIN (Stage=0) -----------------
NOTE: ABSGCONV convergence criterion satisfied.
SIN: Iter=5 Crit=0.06811774: SSE= 811.96345 Acc= 81.6611
------------ Activation= COS (Stage=0) -----------------
NOTE: ABSGCONV convergence criterion satisfied.
COS: Iter=9 Crit=0.07419096: SSE=884.356261 Acc= 81.1913
------------ Activation= EXP (Stage=0) -----------------
NOTE: ABSGCONV convergence criterion satisfied.
EXP: Iter=9 Crit=0.06798656: SSE= 810.39974 Acc= 81.5436
```

The following approximate accuracy rates are based on the discrete values of the predictor (x) variables:

Approximate Goodness-of-Fit Criteria (Stage 0)

Run	Activation	Criterion	SSE	Accuracy
1	SQUARE	0.067824	808.457819	81.644295
3	ARCTAN	0.067953	810.005204	81.661074
8	EXP	0.067987	810.399740	81.543624
2	TANH	0.068026	810.869323	81.627517
4	LOGIST	0.068029	810.910850	81.610738
6	SIN	0.068118	811.963450	81.661074
7	COS	0.074191	884.356261	81.191275
5	GAUSS	0.077276	921.127726	80.251678

After running through the data set we obtain the correct accuracy tables:

Classification Table for CUTOFF = 0.5000

			Predicted	
Activation	Accuracy	Observed	1	0

SQUARE	81.610738	1	229.0	960.0
	0.067548	0	136.0	4635.0
TANH	82.063758	1	254.0	935.0
	0.067682	0	134.0	4637.0
ARCTAN	81.761745	1	242.0	947.0
	0.067722	0	140.0	4631.0
LOGIST	81.845638	1	221.0	968.0
	0.067818	0	114.0	4657.0
SIN	81.275168	1	222.0	967.0
	0.067867	0	149.0	4622.0
EXP	81.543624	1	197.0	992.0
	0.068101	0	108.0	4663.0
COS	81.359060	1	101.0	1088.0
	0.073967	0	23.0000	4748.0
GAUSS	80.167785	1	7.0000	1182.0
	0.079573	0	0	4771.0

The activation function SQUARE seems to be most appropriate for the first stage (stage=0) of estimation. However, TANH yields an even higher accuracy rate:

Goodness-of-Fit Criteria (Ordered by SSE, Stage 0)

Run	Activation	SSE	RMSE	Accuracy
1	SQUARE	805.19026	0.367558	81.610738
3	ARCTAN	805.89106	0.367718	81.778523
8	EXP	806.66533	0.367895	81.593960
4	LOGIST	807.30313	0.368040	81.778523
2	TANH	807.72088	0.368135	81.778523
6	SIN	809.31533	0.368499	81.291946
7	COS	881.68579	0.384622	81.359060
5	GAUSS	949.21059	0.399078	80.167785

The following is the start of the second stage of estimation (stage=1). It starts with selecting three eigenvectors which may predict the residuals best:

Component Selection: SS(y) and R2 (Stage=1)

Comp	Eigval	R-Square	F Value	p-Value
23	4763.193233	0.023292	142.109442	<.0001
21	5192.070258	0.018366	114.178467	<.0001
24	4514.317020	0.017493	110.756118	<.0001

When fitting the first order residuals the average value of the objective function dropped from 0.068 to 0.063. For time reasons the approximate accuracy rates are not computed after the first stage:

```
------------ Optimization Cycle (Stage=1) --------------
------------ Activation= SQUARE (Stage=1) --------------
NOTE: ABSGCONV convergence criterion satisfied.
SQUARE: Iter=1 Crit=0.06280042: SSE=741.517369 Acc= 83.1376
------------ Activation= TANH (Stage=1) ----------------
NOTE: ABSGCONV convergence criterion satisfied.
TANH: Iter=4 Crit=0.06299539: SSE=748.120566 Acc= 83.0705
------------ Activation= ARCTAN (Stage=1) --------------
NOTE: ABSGCONV convergence criterion satisfied.
ARCTAN: Iter=5 Crit=0.06300879: SSE=748.255575 Acc= 83.0705
------------ Activation= LOGIST (Stage=1) --------------
NOTE: ABSGCONV convergence criterion satisfied.
LOGIST: Iter=7 Crit=0.06302038: SSE=749.059872 Acc= 83.1208
------------ Activation= GAUSS (Stage=1) ---------------
NOTE: ABSGCONV convergence criterion satisfied.
GAUSS: Iter=37 Crit=0.06583974: SSE=783.332834 Acc= 82.2987
------------ Activation= SIN (Stage=1) -----------------
NOTE: ABSGCONV convergence criterion satisfied.
SIN: Iter=5 Crit=0.06294376: SSE=747.863259 Acc= 83.0201
------------ Activation= COS (Stage=1) -----------------
NOTE: ABSGCONV convergence criterion satisfied.
COS: Iter=6 Crit=0.06575231: SSE=782.280952 Acc= 82.2819
------------ Activation= EXP (Stage=1) -----------------
NOTE: ABSGCONV convergence criterion satisfied.
EXP: Iter=5 Crit=0.06318155: SSE=746.822973 Acc= 83.1208
```

The best accuracy went up from 81.61 to 83.28, the (1,1) entry from 229 to 319 counts (which is still less than that of TANH in the first stage):

Classification Table for CUTOFF = 0.5000

Activation	Accuracy	Observed	Predicted 1	Predicted 0
SQUARE	83.288591	1	319.0	870.0
	741.977606	0	126.0	4645.0
EXP	83.187919	1	304.0	885.0
	747.249371	0	117.0	4654.0
SIN	83.120805	1	306.0	883.0
	748.284252	0	123.0	4648.0
TANH	83.221477	1	309.0	880.0
	748.436926	0	120.0	4651.0
ARCTAN	83.255034	1	310.0	879.0
	748.614897	0	119.0	4652.0
LOGIST	83.171141	1	305.0	884.0
	749.309756	0	119.0	4652.0
COS	82.583893	1	309.0	880.0
	781.342717	0	158.0	4613.0
GAUSS	82.567114	1	309.0	880.0

```
              782.484989       0        159.0      4612.0

        Goodness-of-Fit Criteria (Ordered by SSE, Stage 1)

 Run    Activation        SSE            RMSE         Accuracy

   1    SQUARE      741.97761      0.352835       83.288591
   8    EXP         747.24937      0.354087       83.187919
   6    SIN         748.28425      0.354332       83.120805
   2    TANH        748.43693      0.354368       83.221477
   3    ARCTAN      748.61490      0.354410       83.255034
   4    LOGIST      749.30976      0.354575       83.171141
   7    COS         781.34272      0.362074       82.583893
   5    GAUSS       782.48499      0.362339       82.567114
```

Here starts the third stage (stage=2):

```
      Component Selection: SS(y) and R2 (Stage=2)

Comp        Eigval     R-Square     F Value    p-Value

   1         15337     0.006514   39.068994     <.0001
   5   8117.555354     0.005566   33.564983     <.0001
   7   7371.205837     0.005429   32.918782     <.0001
```

When fitting the second order residuals the average value of the objective function dropped from 0.063 to 0.061.

```
------------ Optimization Cycle (Stage=2) --------------
------------ Activation= SQUARE (Stage=2) --------------
NOTE: ABSGCONV convergence criterion satisfied.
SQUARE: Iter=1 Crit=0.06128736: SSE=720.996529 Acc= 83.4228
------------ Activation= TANH (Stage=2) ----------------
NOTE: ABSGCONV convergence criterion satisfied.
TANH: Iter=5 Crit=0.06164516: SSE= 726.12791 Acc= 83.6242
------------ Activation= ARCTAN (Stage=2) --------------
NOTE: ABSGCONV convergence criterion satisfied.
ARCTAN: Iter=5 Crit=0.06167716: SSE=726.468943 Acc= 83.6074
------------ Activation= LOGIST (Stage=2) --------------
NOTE: ABSGCONV convergence criterion satisfied.
LOGIST: Iter=8 Crit=0.06157175: SSE=725.393203 Acc= 83.5403
------------ Activation= GAUSS (Stage=2) ---------------
NOTE: ABSGCONV convergence criterion satisfied.
GAUSS: Iter=2 Crit=0.06272854: SSE=740.745518 Acc= 83.3054
------------ Activation= SIN (Stage=2) -----------------
NOTE: ABSGCONV convergence criterion satisfied.
SIN: Iter=6 Crit=0.06164909: SSE=726.185402 Acc= 83.6074
------------ Activation= COS (Stage=2) -----------------
NOTE: ABSGCONV convergence criterion satisfied.
```

```
COS: Iter=3 Crit=0.06277476: SSE=741.349891 Acc= 83.2886
------------ Activation= EXP (Stage=2) -----------------
NOTE: ABSGCONV convergence criterion satisfied.
EXP: Iter=4 Crit=0.06131052: SSE=721.679089 Acc= 83.4396
```

The best accuracy increased from 83.29 to 83.47 and the (1,1) entry from 319 to 343 counts:

Classification Table for CUTOFF = 0.5000

Activation	Accuracy	Observed	Predicted 1	0
SQUARE	83.473154	1	343.0	846.0
	721.118524	0	139.0	4632.0
EXP	83.422819	1	337.0	852.0
	721.301910	0	136.0	4635.0
LOGIST	83.607383	1	337.0	852.0
	724.687746	0	125.0	4646.0
TANH	83.607383	1	340.0	849.0
	725.553808	0	128.0	4643.0
ARCTAN	83.607383	1	341.0	848.0
	725.724668	0	129.0	4642.0
SIN	83.607383	1	340.0	849.0
	725.889780	0	128.0	4643.0
GAUSS	83.372483	1	317.0	872.0
	741.471407	0	119.0	4652.0
COS	83.322148	1	316.0	873.0
	741.668156	0	121.0	4650.0

Even though SQUARE shows the best SSE, the accuracy rates for some other functions (e.g. LOGIST) are slightly better:

Goodness-of-Fit Criteria (Ordered by SSE, Stage 2)

Run	Activation	SSE	RMSE	Accuracy
1	SQUARE	721.11852	0.347841	83.473154
8	EXP	721.30191	0.347885	83.422819
4	LOGIST	724.68775	0.348700	83.607383
2	TANH	725.55381	0.348909	83.607383
3	ARCTAN	725.72467	0.348950	83.607383
6	SIN	725.88978	0.348989	83.607383
5	GAUSS	741.47141	0.352715	83.372483
7	COS	741.66816	0.352762	83.322148

Component selection w.r.t. the residuals of the stage 2 starts the estimation of stage 3. Note, that the R^2 values become smaller and smaller.

```
          Component Selection: SS(y) and R2 (Stage=3)

Comp         Eigval     R-Square      F Value   p-Value


   8    6938.083228     0.005571    33.383374     <.0001
  20    5345.603436     0.004223    25.409312     <.0001
  12    6136.575271     0.004059    24.517995     <.0001
```

Also the size of the objective function at the optimization results decreases:

```
------------ Optimization Cycle (Stage=3) --------------
------------ Activation= SQUARE (Stage=3) --------------
NOTE: ABSGCONV convergence criterion satisfied.
SQUARE: Iter=1 Crit=0.06049339: SSE=710.516275 Acc= 83.7081
------------ Activation= TANH (Stage=3) ----------------
NOTE: ABSGCONV convergence criterion satisfied.
TANH: Iter=4 Crit=0.06052425: SSE=710.396136 Acc= 83.7752
------------ Activation= ARCTAN (Stage=3) --------------
NOTE: ABSGCONV convergence criterion satisfied.
ARCTAN: Iter=3 Crit=0.06052607: SSE=710.489715 Acc= 83.7081
------------ Activation= LOGIST (Stage=3) --------------
NOTE: ABSGCONV convergence criterion satisfied.
LOGIST: Iter=6 Crit=0.06055936: SSE=711.054572 Acc= 83.6577
------------ Activation= GAUSS (Stage=3) ---------------
NOTE: ABSGCONV convergence criterion satisfied.
GAUSS: Iter=6 Crit=0.06111674: SSE= 719.41694 Acc= 83.3725
------------ Activation= SIN (Stage=3) -----------------
NOTE: ABSGCONV convergence criterion satisfied.
SIN: Iter=3 Crit=0.06051959: SSE=710.308709 Acc= 83.8087
------------ Activation= COS (Stage=3) -----------------
NOTE: ABSGCONV convergence criterion satisfied.
COS: Iter=6 Crit=0.06117044: SSE=719.262211 Acc= 83.3725
------------ Activation= EXP (Stage=3) -----------------
NOTE: ABSGCONV convergence criterion satisfied.
EXP: Iter=2 Crit=0.06051088: SSE=710.810558 Acc= 83.7081
```

The accuracy of the best fit impoves slightly from 83.47 to 83.79 and the size of the (1,1) entry inceases from 343 to 364.

```
         Classification Table for CUTOFF = 0.5000

                                              Predicted
 Activation       Accuracy   Observed       1           0

     SIN       83.791946       1        364.0       825.0
              709.778632       0        141.0      4630.0
     TANH      83.758389       1        363.0       826.0
```

```
            709.889256       0        142.0      4629.0
ARCTAN       83.708054       1        361.0       828.0
            710.036390       0        143.0      4628.0
SQUARE       83.724832       1        355.0       834.0
            710.075198       0        136.0      4635.0
EXP          83.741611       1        356.0       833.0
            710.212159       0        136.0      4635.0
LOGIST       83.691275       1        357.0       832.0
            710.822647       0        140.0      4631.0
COS          83.355705       1        340.0       849.0
            718.944913       0        143.0      4628.0
GAUSS        83.288591       1        328.0       861.0
            719.269965       0        135.0      4636.0
```

```
   Goodness-of-Fit Criteria (Ordered by SSE, Stage 3)

Run    Activation         SSE          RMSE       Accuracy


  6    SIN          709.77863      0.345095      83.791946
  2    TANH         709.88926      0.345122      83.758389
  3    ARCTAN       710.03639      0.345157      83.708054
  1    SQUARE       710.07520      0.345167      83.724832
  8    EXP          710.21216      0.345200      83.741611
  4    LOGIST       710.82265      0.345348      83.691275
  7    COS          718.94491      0.347316      83.355705
  5    GAUSS        719.26997      0.347394      83.288591
```

Now the residuals are computed and components are selected for the last estimation stage:

```
       Component Selection: SS(y) and R2 (Stage=4)

Comp        Eigval    R-Square      F Value    p-Value


  28  1195.710958    0.003997    23.916548     <.0001
  27  3456.490592    0.001822    10.919693     0.0010
  25  3935.018952    0.001803    10.824185     0.0010
```

There are no problems with the optimization processes:

```
------------ Optimization Cycle (Stage=4) --------------
------------ Activation= SQUARE (Stage=4) --------------
NOTE: ABSGCONV convergence criterion satisfied.
SQUARE: Iter=1 Crit=0.05983921: SSE=703.669268 Acc= 83.6913
------------ Activation= TANH (Stage=4) ----------------
NOTE: ABSGCONV convergence criterion satisfied.
```

```
TANH: Iter=5 Crit=0.06015823: SSE=706.476969 Acc= 83.6074
------------ Activation= ARCTAN (Stage=4) --------------
NOTE: ABSGCONV convergence criterion satisfied.
ARCTAN: Iter=3 Crit=0.06013359: SSE=706.212332 Acc= 83.7081
------------ Activation= LOGIST (Stage=4) --------------
NOTE: ABSGCONV convergence criterion satisfied.
LOGIST: Iter=3 Crit=0.06017552: SSE=706.851414 Acc= 83.7919
------------ Activation= GAUSS (Stage=4) ---------------
NOTE: ABSGCONV convergence criterion satisfied.
GAUSS: Iter=4 Crit=0.06032127: SSE=708.571854 Acc= 83.8255
------------ Activation= SIN (Stage=4) -----------------
NOTE: ABSGCONV convergence criterion satisfied.
SIN: Iter=3 Crit=0.06014411: SSE=706.402904 Acc= 83.6745
------------ Activation= COS (Stage=4) -----------------
NOTE: ABSGCONV convergence criterion satisfied.
COS: Iter=4 Crit=0.06007575: SSE=707.016805 Acc= 83.8087
------------ Activation= EXP (Stage=4) -----------------
NOTE: ABSGCONV convergence criterion satisfied.
EXP: Iter=3 Crit=0.05983526: SSE=703.933766 Acc= 83.6074
```

The accuracy of the result is no longer improved and drops from 83.79 to 83.72, and also the (1,1) entry was decreased from 365 to 363. This can happen only when the discretization error becomes too large in relation to the goodness of fit of the nonlinear model. Perhaps the specification of larger values for MAXCOMP= and NPOINT= could improve the solution. However, in most applications we would see this behavior as a sign that no further improvement of the model fit is possible.

Classification Table for CUTOFF = 0.5000

Activation	Accuracy	Observed	Predicted 1	Predicted 0
SQUARE	83.724832	1	363.0	826.0
	702.899794	0	144.0	4627.0
EXP	83.691275	1	361.0	828.0
	703.295564	0	144.0	4627.0
ARCTAN	83.775168	1	364.0	825.0
	705.243085	0	142.0	4629.0
SIN	83.691275	1	363.0	826.0
	705.508160	0	146.0	4625.0
TANH	83.708054	1	362.0	827.0
	705.634506	0	144.0	4627.0
LOGIST	83.708054	1	360.0	829.0
	705.732595	0	142.0	4629.0
COS	83.842282	1	364.0	825.0
	707.292433	0	138.0	4633.0
GAUSS	83.791946	1	362.0	827.0
	708.659944	0	139.0	4632.0

Even though accuracy did not improve, the SSE value still dropped from 710 to 703 during the last stage.

```
     Goodness-of-Fit Criteria (Ordered by SSE, Stage 4)

 Run    Activation        SSE          RMSE      Accuracy

   1    SQUARE      702.89979     0.343418     83.724832
   8    EXP         703.29556     0.343515     83.691275
   3    ARCTAN      705.24309     0.343990     83.775168
   6    SIN         705.50816     0.344055     83.691275
   2    TANH        705.63451     0.344086     83.708054
   4    LOGIST      705.73259     0.344110     83.708054
   7    COS         707.29243     0.344490     83.842282
   5    GAUSS       708.65994     0.344823     83.791946
```

The following summary table shows the improvements in SSE and Accuracy rates across the 5 stages:

```
                Summary Table Across Stages

Stage Activation Link      SSE    RMSE   Accuracy    AIC

    0 SQUARE     LOGIST 805.1902 0.36756 81.61074 -11730
    1 SQUARE     IDENT  741.9776 0.35284 83.28859 -12018
    2 SQUARE     IDENT  721.1185 0.34784 83.47315 -11988
    3 SIN        IDENT  709.7786 0.34510 83.79195 -11882
    4 SQUARE     IDENT  702.8998 0.34342 83.72483 -11740
```

All 40 optimizations were very efficient with about 5 iterations per optimization and less than 10 function calls per optimization:

```
*** Total Number of Runs through Data :       27
*** Total Number of NL Optimizations  :       40
*** Total Number of Iterations in NLP :      219
*** Total Number Function Calls in NLP:      392
```

In this application those solutions were selected which had the smallest Sum-of-Squares Error. By specifying the *selcrit=acc* option we can instead select the solutions with the largest accuracy rate:

```
proc dmneurl data=dmdbout dmdbcat=outcat
  outclass=oclass outest=estout out=dsout outfit=ofit
  ptable maxcomp=3 maxstage=5 selcrit=acc;
  var LOAN MORTDUE VALUE REASON JOB YOJ DEROG DELINQ
      CLAGE NINQ CLNO DEBTINC;
  target BAD;
run;
```

The following output only shows the summary table. For this example, the total accuracy was slightly increased in all stages except the second. However, this behavior must not always be true for other examples.

```
                     Summary Table Across Stages
Stage Activation Link       SSE     RMSE  Accuracy    AIC

    0 ARCTAN     LOGIST 805.8911 0.36772 81.77852 -11725
    1 SQUARE     IDENT  746.6223 0.35394 83.27181 -11981
    2 LOGIST     IDENT  730.7282 0.35015 83.85906 -11909
    3 EXP        IDENT  714.0782 0.34614 84.02685 -11846
    4 COS        IDENT  711.5807 0.34553 83.97651 -11667
```

Application: HMEQ Data Set: Interval Target LOAN

Now we show the specification and results of PROC DMNEURL for the interval target LOAN. First we have to obtain the DMDB data set and catalog from the raw data set:

```
libname sampsio '/sas/a612/dmine/sampsio';
proc dmdb batch data=sampsio.hmeq out=dmdbout dmdbcat=outcat;
  var LOAN MORTDUE VALUE YOJ DELINQ CLAGE NINQ CLNO DEBTINC;
  class BAD(ASC) REASON(ASC) JOB(ASC) DEROG(ASC);
  target LOAN;
run;
```

The PROC DMNEURL call is very similar, but here 5 stages with each 3 components ($p = 7$) are specified:

```
proc dmneurl data=dmdbout dmdbcat=outcat
  outclass=oclass outest=estout out=dsout outfit=ofit
  ptable maxcomp=3 maxstage=6;
  var BAD MORTDUE VALUE REASON JOB YOJ DEROG DELINQ
      CLAGE NINQ CLNO DEBTINC;
  target LOAN;
run;
```

The link function for interval target is by default specified as the identity:

```
                 The DMNEURL Procedure

         Interval Target                         LOAN
         Number Observations                     5960
         NOBS w/o Missing Target                 5960
         Target Range                  [ 1100, 89900]
         Link Function                          IDENT
         Selection Criterion                      SSE
         Optimization Criterion                   SSE
```

Estimation Stages	6
Max. Number Components	3
Minimum R2 Value	0.000050
Number Grid Points	17

Variable	Mean	Std Dev	Skewness	Kurtosis
LOAN	18608	11207	2.02378	6.93259
MORTDUE	67350	44458	1.81448	6.48187
VALUE	99863	57386	3.05334	24.36280
YOJ	8.15130	7.57398	0.98846	0.37207
DELINQ	0.40570	1.12727	4.02315	23.56545
CLAGE	170.47634	85.81009	1.34341	7.59955
NINQ	1.08456	1.72867	2.62198	9.78651
CLNO	20.50285	10.13893	0.77505	1.15767
DEBTINC	26.59885	8.60175	2.85235	50.50404

For an interval target the percentiles of the response (target) variable are computed as an aside of the preliminary runs through the data. (Note, that the values of the response y are not all stored in RAM.)

Percentiles of Target LOAN in [1100 : 89900]

	Nobs	Y Value	Label
1	596	7600.000000	0.073198198
2	1192	10000	0.100225225
3	1788	12100	0.123873874
4	2384	14400	0.149774775
5	2980	16300	0.171171171
6	3576	18800	0.199324324
7	4172	21700	0.231981982
8	4768	25000	0.269144144
9	5364	30500	0.331081081
10	5960	89900	1

The first estimation stage starts with the selection of the best predictor components (eigenvectors):

Component Selection: SS(y) and R2 (SS_total=326.60303927)

Comp	Eigval	R-Square	F Value	p-Value	SSE
2	14414	0.015964	96.672480	<.0001	321.389163
28	1232.230727	0.005739	34.949673	<.0001	319.514886
11	6335.576701	0.005490	33.620923	<.0001	317.721686

A maximum of 8 iterations is needed for convergence:

```
------------ Optimization Cycle (Stage=0) --------------
------------ Activation= SQUARE (Stage=0) --------------
NOTE: ABSGCONV convergence criterion satisfied.
SQUARE: Iter=1 Crit=0.00719589: SSE=6.76374E11 Acc= 32.7484
------------ Activation= TANH (Stage=0) ----------------
NOTE: ABSGCONV convergence criterion satisfied.
TANH: Iter=7 Crit=0.00729363: SSE=6.85561E11 Acc= 29.2423
------------ Activation= ARCTAN (Stage=0) --------------
NOTE: ABSGCONV convergence criterion satisfied.
ARCTAN: Iter=4 Crit=0.00730523: SSE=6.86651E11 Acc= 29.2427
------------ Activation= LOGIST (Stage=0) --------------
NOTE: ABSGCONV convergence criterion satisfied.
LOGIST: Iter=5 Crit=  0.007296: SSE=6.85784E11 Acc= 28.7727
------------ Activation= GAUSS (Stage=0) ---------------
NOTE: ABSGCONV convergence criterion satisfied.
GAUSS: Iter=8 Crit=0.00753243: SSE=7.08006E11 Acc= 15.4180
------------ Activation= SIN (Stage=0) -----------------
NOTE: ABSGCONV convergence criterion satisfied.
SIN: Iter=2 Crit=0.00732518: SSE=6.88526E11 Acc= 29.0399
------------ Activation= COS (Stage=0) -----------------
NOTE: ABSGCONV convergence criterion satisfied.
COS: Iter=5 Crit=0.00753876: SSE=7.08602E11 Acc= 22.5534
------------ Activation= EXP (Stage=0) -----------------
NOTE: ABSGCONV convergence criterion satisfied.
EXP: Iter=2 Crit=0.00724534: SSE=6.81022E11 Acc= 29.9011
```

For interval target y the accuracy is computed as the Goodman-Kruskal γ coefficient for a observed-predicted frequency table using the percentiles of y for row and column definitions. (Note, that the Goodman-Kruskal γ can have negative values for extrem bad fit.)

Approximate Goodness-of-Fit Criteria (Stage 0)

Run	Activation	Criterion	SSE	Accuracy
1	SQUARE	0.007196	676373814905	32.748384
8	EXP	0.007245	681021814295	29.901149
2	TANH	0.007294	685560807525	29.242251
4	LOGIST	0.007296	685783817673	28.772685
3	ARCTAN	0.007305	686651267193	29.242724
6	SIN	0.007325	688526431720	29.039929
5	GAUSS	0.007532	708006119992	15.418007
7	COS	0.007539	708601735521	22.553358

The Root-Mean-Squared-Estimate RMSE for the first stage is 10589:

Goodness-of-Fit Criteria (Ordered by SSE, Stage 0)

```
 Run   Activation         SSE      RMSE     Accuracy

   1   SQUARE      6.68237E11     10589    33.925841
   2   TANH        6.74156E11     10635    30.776431
   4   LOGIST      6.74885E11     10641    30.492000
   8   EXP         6.77111E11     10659    30.503925
   3   ARCTAN      6.78503E11     10670    29.832932
   6   SIN         6.81758E11     10695    29.402115
   5   GAUSS       7.02918E11     10860    21.174016
   7   COS         7.04934E11     10876    20.022492
```

The second stage (stage=1) starts with selecting the best principal components for predicting the residual:

```
   Component Selection: SS(y) and R2 (Stage=1)

Comp       Eigval   R-Square      F Value  p-Value

   1        16197   0.023135   141.126005   <.0001
   3        12170   0.017130   106.340108   <.0001
  20  5623.081574   0.012121    76.193667   <.0001
```

Now a maximum of 5 iterations is needed for convergence:

```
------------ Optimization Cycle (Stage=1) --------------
------------ Activation= SQUARE (Stage=1) --------------
NOTE: ABSGCONV convergence criterion satisfied.
SQUARE: Iter=1 Crit=0.00675824: SSE=6.35237E11 Acc= 39.9782
------------ Activation= TANH (Stage=1) ----------------
NOTE: ABSGCONV convergence criterion satisfied.
TANH: Iter=4 Crit=0.00677155: SSE=6.36489E11 Acc= 40.1296
------------ Activation= ARCTAN (Stage=1) --------------
NOTE: ABSGCONV convergence criterion satisfied.
ARCTAN: Iter=4 Crit=0.00675928: SSE=6.35335E11 Acc= 41.0832
------------ Activation= LOGIST (Stage=1) --------------
NOTE: ABSGCONV convergence criterion satisfied.
LOGIST: Iter=5 Crit=0.00676491: SSE=6.35864E11 Acc= 41.1768
------------ Activation= GAUSS (Stage=1) ---------------
NOTE: ABSGCONV convergence criterion satisfied.
GAUSS: Iter=5 Crit=0.00701114: SSE=6.59009E11 Acc= 37.5715
------------ Activation= SIN (Stage=1) -----------------
NOTE: ABSGCONV convergence criterion satisfied.
SIN: Iter=4 Crit=0.00676857: SSE=6.36208E11 Acc= 40.1257
------------ Activation= COS (Stage=1) -----------------
NOTE: ABSGCONV convergence criterion satisfied.
COS: Iter=5 Crit=0.00699351: SSE=6.57351E11 Acc= 36.1925
------------ Activation= EXP (Stage=1) -----------------
NOTE: ABSGCONV convergence criterion satisfied.
EXP: Iter=1 Crit=0.00676817: SSE= 6.3617E11 Acc= 40.3710
```

The RMSE dropped from 10589 to 10294:

```
 Goodness-of-Fit Criteria (Ordered by SSE, Stage 1)

 Run    Activation        SSE    RMSE     Accuracy


   1    SQUARE     6.31521E11   10294    41.275893
   3    ARCTAN     6.32669E11   10303    40.218692
   8    EXP        6.32795E11   10304    40.947613
   6    SIN        6.32908E11   10305    40.499603
   2    TANH        6.3331E11   10308    40.464591
   4    LOGIST     6.33355E11   10309    40.566497
   7    COS        6.56699E11   10497    36.381117
   5    GAUSS      6.58666E11   10513    37.785522
```

The third stage starts with selecting the best eigenvectors for prediction of the residuals of the last stage:

```
    Component Selection: SS(y) and R2 (Stage=2)

Comp        Eigval    R-Square     F Value  p-Value

  23   4811.081772    0.011805   71.186100   <.0001
  28   1232.230727    0.007479   45.434565   <.0001
  18   5865.674624    0.006724   41.123895   <.0001
```

Now, the maximum of iterations is four!

```
------------ Optimization Cycle (Stage=2) --------------
------------ Activation= SQUARE (Stage=2) --------------
NOTE: ABSGCONV convergence criterion satisfied.
SQUARE: Iter=1 Crit=0.00652258: SSE=6.13086E11 Acc= 42.4729
------------ Activation= TANH (Stage=2) ----------------
NOTE: ABSGCONV convergence criterion satisfied.
TANH: Iter=3 Crit=0.00650972: SSE=6.11878E11 Acc= 41.5654
------------ Activation= ARCTAN (Stage=2) --------------
NOTE: ABSGCONV convergence criterion satisfied.
ARCTAN: Iter=3 Crit=0.00650403: SSE=6.11342E11 Acc= 42.2974
------------ Activation= LOGIST (Stage=2) --------------
NOTE: ABSGCONV convergence criterion satisfied.
LOGIST: Iter=4 Crit=0.00641908: SSE=6.03358E11 Acc= 43.3779
------------ Activation= GAUSS (Stage=2) ---------------
NOTE: ABSGCONV convergence criterion satisfied.
GAUSS: Iter=2 Crit=0.00671802: SSE=6.31456E11 Acc= 41.2660
------------ Activation= SIN (Stage=2) -----------------
NOTE: ABSGCONV convergence criterion satisfied.
SIN: Iter=4 Crit=0.00651654: SSE=6.12519E11 Acc= 41.9809
------------ Activation= COS (Stage=2) -----------------
NOTE: ABSGCONV convergence criterion satisfied.
```

```
COS: Iter=4 Crit=0.00671738: SSE=6.31396E11 Acc= 41.2783
------------ Activation= EXP (Stage=2) -----------------
NOTE: ABSGCONV convergence criterion satisfied.
EXP: Iter=0 Crit=0.00656615: SSE=6.17182E11 Acc= 41.6251
```

The RMSE dropped from 10294 to 10035:

```
Goodness-of-Fit Criteria (Ordered by SSE, Stage 2)

 Run   Activation        SSE     RMSE    Accuracy

   4   LOGIST     6.00171E11    10035   44.076324
   3   ARCTAN     6.11611E11    10130   42.722853
   2   TANH       6.12574E11    10138   42.454925
   6   SIN        6.12902E11    10141   42.618536
   1   SQUARE     6.14545E11    10154   43.452922
   8   EXP        6.17964E11    10183   42.588823
   7   COS        6.31415E11    10293   41.153349
   5   GAUSS      6.31533E11    10294   41.028769
```

In stage 3 components are selected w.r.t. the residuals from stage 2:

```
     Component Selection: SS(y) and R2 (Stage=3)

Comp        Eigval    R-Square      F Value p-Value

   5   8108.233368    0.008115    48.751302   <.0001
   7   7678.598513    0.004569    27.574638   <.0001
  27   3496.302840    0.003929    23.802006   <.0001

------------ Optimization Cycle (Stage=3) --------------
------------ Activation= SQUARE (Stage=3) --------------
NOTE: ABSGCONV convergence criterion satisfied.
SQUARE: Iter=1 Crit=0.00627437: SSE=5.89756E11 Acc= 46.5664
------------ Activation= TANH (Stage=3) ----------------
NOTE: ABSGCONV convergence criterion satisfied.
TANH: Iter=2 Crit=0.00628152: SSE=5.90428E11 Acc= 46.2812
------------ Activation= ARCTAN (Stage=3) --------------
NOTE: ABSGCONV convergence criterion satisfied.
ARCTAN: Iter=2 Crit=0.00628195: SSE=5.90469E11 Acc= 46.2125
------------ Activation= LOGIST (Stage=3) --------------
NOTE: ABSGCONV convergence criterion satisfied.
LOGIST: Iter=5 Crit=0.00628127: SSE=5.90404E11 Acc= 45.8040
------------ Activation= GAUSS (Stage=3) ---------------
NOTE: ABSGCONV convergence criterion satisfied.
GAUSS: Iter=2 Crit=0.00637032: SSE=5.98775E11 Acc= 45.3250
------------ Activation= SIN (Stage=3) -----------------
NOTE: ABSGCONV convergence criterion satisfied.
SIN: Iter=6 Crit=0.00627884: SSE=5.90176E11 Acc= 46.0972
```

```
------------ Activation= COS (Stage=3) -----------------
NOTE: ABSGCONV convergence criterion satisfied.
COS: Iter=5 Crit=0.00637738: SSE=5.99438E11 Acc= 44.6439
------------ Activation= EXP (Stage=3) -----------------
NOTE: ABSGCONV convergence criterion satisfied.
EXP: Iter=1 Crit=0.00628498: SSE=5.90753E11 Acc= 46.4650
```

The RMSE dropped from 10035 to 9939:

```
  Goodness-of-Fit Criteria (Ordered by SSE, Stage 3)

 Run   Activation        SSE         RMSE   Accuracy

   1   SQUARE     5.88794E11  9939.361833  46.874168
   6   SIN        5.89308E11  9943.696786  46.045231
   4   LOGIST     5.89532E11  9945.590341  45.936035
   2   TANH       5.89658E11  9946.655850  46.521683
   3   ARCTAN     5.89714E11  9947.122017  46.462470
   8   EXP        5.89994E11  9949.489480  46.590155
   5   GAUSS      5.98777E11        10023  45.338954
   7   COS        5.99432E11        10029  44.563112
```

Again, the new stage 4 starts with component selection w.r.t. the residuals of the last stage3:

```
   Component Selection: SS(y) and R2 (Stage=4)

Comp        Eigval  R-Square    F Value  p-Value


  14   5977.581155  0.004044  24.196310   <.0001
  24   4589.938565  0.002803  16.817296   <.0001
   8   7098.575517  0.002425  14.583721   0.0001
```

A maximum of 7 iterations is needed for convergence:

```
------------ Optimization Cycle (Stage=4) --------------
------------ Activation= SQUARE (Stage=4) --------------
NOTE: ABSGCONV convergence criterion satisfied.
SQUARE: Iter=1 Crit=0.00618628: SSE=5.81476E11 Acc= 46.9112
------------ Activation= TANH (Stage=4) ----------------
NOTE: ABSGCONV convergence criterion satisfied.
TANH: Iter=5 Crit= 0.0061812: SSE=5.80998E11 Acc= 45.5545
------------ Activation= ARCTAN (Stage=4) --------------
NOTE: ABSGCONV convergence criterion satisfied.
ARCTAN: Iter=3 Crit=0.00618984: SSE= 5.8181E11 Acc= 45.7876
------------ Activation= LOGIST (Stage=4) --------------
NOTE: ABSGCONV convergence criterion satisfied.
LOGIST: Iter=2 Crit= 0.0062313: SSE=5.85708E11 Acc= 47.2978
```

```
------------ Activation= GAUSS (Stage=4) ---------------
NOTE: ABSGCONV convergence criterion satisfied.
GAUSS: Iter=3 Crit=0.00617962: SSE= 5.8085E11 Acc= 46.3454
------------ Activation= SIN (Stage=4) -----------------
NOTE: ABSGCONV convergence criterion satisfied.
SIN: Iter=7 Crit=0.00614634: SSE=5.77722E11 Acc= 45.3470
------------ Activation= COS (Stage=4) -----------------
NOTE: ABSGCONV convergence criterion satisfied.
COS: Iter=2 Crit=0.00619158: SSE=5.81974E11 Acc= 47.5278
------------ Activation= EXP (Stage=4) -----------------
NOTE: ABSGCONV convergence criterion satisfied.
EXP: Iter=2 Crit=0.00620074: SSE=5.82835E11 Acc= 47.1760
```

The RMSE dropped from 9939 to 9889:

```
 Goodness-of-Fit Criteria (Ordered by SSE, Stage 4)

 Run   Activation        SSE           RMSE     Accuracy

   5   GAUSS      5.82844E11   9889.013804    46.529424
   1   SQUARE     5.82906E11   9889.541858    47.202054
   7   COS        5.83336E11   9893.184747    47.340247
   8   EXP        5.83553E11   9895.031798    47.907199
   2   TANH        5.8489E11   9906.353245    46.238301
   4   LOGIST     5.86142E11   9916.953006    47.605286
   3   ARCTAN     5.88716E11   9938.707816    46.013181
   6   SIN        6.12035E11         10134    45.185468
```

For space reasons we are skipping the results of stage 5 except the following table which shows that the RMSE dropped again.

```
   Goodness-of-Fit Criteria (Ordered by SSE, Stage 5)

 Run   Activation        SSE           RMSE    Accuracy

   1   SQUARE     5.78114E11   9848.803178   47.337721
   8   EXP        5.78394E11   9851.192061   47.342324
   4   LOGIST     5.78507E11   9852.150383   47.269327
   3   ARCTAN     5.79057E11   9856.832196   46.609720
   2   TANH        5.8133E11   9876.166691   46.144529
   5   GAUSS      5.82144E11   9883.077740   46.792103
   7   COS        5.82405E11   9885.287904   46.540469
   6   SIN        6.10243E11         10119   45.311873
```

This is a summary table for the first six estimation stages:

```
                        Summary Table Across Stages

Stage Activation Link         SSE       RMSE Accuracy    AIC
```

```
0 SQUARE     IDENT 6.68237E11      10589 33.92584 110675
1 SQUARE     IDENT 6.31521E11      10294 41.27589 110544
2 LOGIST     IDENT 6.00171E11      10035 44.07632 110447
3 SQUARE     IDENT 5.88794E11 9939.36183 46.87417 110539
4 GAUSS      IDENT 5.82844E11 9889.01380 46.52942 110684
5 SQUARE     IDENT 5.78114E11 9848.80318 47.33772 110842
```

The six stages took 48 optimizations (each with 7 parameters) and 33 runs through the data. In average less than 4 iterations and about 7 function calls are needed for each optimization:

```
*** Total Number of Runs through Data :       33
*** Total Number of NL Optimizations  :       48
*** Total Number of Iterations in NLP :      159
*** Total Number Function Calls in NLP:      348
```

Missing Values

Observations with missing values in the target variable (response or dependend variable) are not included in the analysis. Those observations are, however, scored, i.e. predicted values are computed.
Observations with missing values in the predictor variables (independend variables) are processed depending on the scale type of the variable:

- For numeric variables, missing values are replaced by the (weighted) mean of the variable.
- For class variables, missing values are treated as an additional category.

Syntax of PROC DMNEURL

PROC DMNEURL *options*;	required statement
VAR *variables*;	optional statements
TARGET *variables*;	
FUNCTION *names*;	
LINK *name*;	
FREQ *variables*;	
WEIGHT *variables*;	
DECISION *options*;	

Overview of PROC DMNEURL Options

PROC DMNEURL *options* **;**

This statement invokes the DMNEURL procedure. The options available with the PROC DMNEURL statement are:

DATA=*SASdataset* :
specifies an input data set generated by PROC DMDB which is associated with a valid catalog specified by the DMDBCAT= option. This option must be specified, no default is permitted. The DATA= data set must contain interval scaled variables and CLASS variables in a specific form written by PROC DMDB.

DMDBCAT=*SAScatalog* :
specifies an input catalog of meta information generated by PROC DMDB which is assiciated with a valid data set specified by the DATA= option. The catalog contains important information (e.g. range of variables, number of missing values of each variable, moments of variables) which is used by many other procedures which require a DMDB data set. That means, that both, the DMDBCAT= catalog and the DATA= data set must be **InSync** to obtain proper results! This option must be specified, no default is permitted.

TESTDATA=*SASdataset* :
specifies a second input data set which is by default NOT generated by PROC DMDB, which however must contain all variables of the DATA= input data set which are used in the model. The variables not used in the model may be different. The order of variables is not relevant. If TESTDATA= is specified, you can specify a TESTOUT= output data set (containing predicted values and residuals) which relates to the TESTDATA= input data set the same as the OUT= data set relates to the DATA= input training data set. When specifying the TESTDMDB option you may use a data set generated by PROC DMDB as the TESTDATA= input data set.

OUTCLASS=*SASdataset* :
specifies an output data set generated by PROC DMNEURL which contains the mapping inbetween compound variable names and the names of variables and categories of CLASS variables used in the model. The compound variable names are used to denote dummy variables which are created for each category of a CLASS variable with more than two categories. Since the compound names of dummy variables are used for variable names in other data sets the user must know to which category each compound name corresponds. The OUTCLASS= data set has only three character variables

NAME contains compound name used as variable names in other output data sets

VAR contains variable name used in DATA= input data set

LEVEL contains level name of variable as used in DATA= input data set.

Note, if the DATA= input data set does not contain any CLASS variables the OUTCLASS= data set is not written.

OUTEST=*SASdataset* :
specifies an output data set generated by PROC DMNEURL which contains all the model information necessary for scoring additional cases or data sets.

Variables of the output data set:

TARGET (character) name of the target

TYPE (character) type of observation

NAME (character) name of observation

STAGE number of stage

MEAN contains different numeric information

STDEV contains different numeric information

$varname_i$ variables in the model variables; the first variables correspond to CLASS (categorical) the remaining variables are continuously (interval or ratio) scaled. Note, that for nonbinary CLASS (nominal or ordinal categorical) variables a set of binary dummy variables is created. In those cases the prefix of variable names $varname_i$ used for a group of variables in the data set may be the same for a successive group of variables which differs only by a numeric suffix.

This data set contains all the model information necessary to compute the predicted model values (scores).

1. The _TYPE_=_V_MAP_ and _TYPE_=_C_MAP_ observations contain the mapping indices between the variables used in the model and the number of the variable in the data set.
   - The _MEAN_ variable contains the number of index mappings.
   - The _STDEV_ variable contains the index of the target (response) variable in the data set for the _TYPE_=_V_MAP_ observation. For _TYPE_=_C_MAP_ it contains the level (category) number of a categorical target variable that corresponds to missing values.
2. The _TYPE_=_EIGVAL_ observation contains the sorted eigenvalues of the $X'X$ matrix. Here, the _MEAN_ variable contains the number of model variables (rows/columns of the model $X'X$ matrix) and the _STDEV_ variable contains the number c of model components.
3. For each stage of the estimation process two groups of observations are written to the OUTEST= data set:

   (a) The _TYPE_=_EIGVEC_ observations contain a set of c principal components which are used as predictor variables for the estimation of the original traget value y (in stage 0) or for the prediction of the stage i residual. Here, the _MEAN_ variable contains the value for the criterion used to include the component into the model which is normally the R^2 value. The _STDEV_ variable contains the eigenvalue number to which the eigenvector corresponds.

1	**SQUARE**	$(a + b * x) * x$
2	**TANH**	$a * tanh(b * x)$
3	**ARCTAN**	$a * atan(b * x)$
4	**LOGIST**	$exp(a * x)/(1. + exp(b * x)$
5	**GAUSS**	$a * exp(-(b * x)^2)$
6	**SIN**	$a * sin(b * x)$
7	**COS**	$a * cos(b * x)$
8	**EXP**	$a * exp(b * x)$

   The _NAME_ variable reports the corresponding name of the best activation function found.

(b) The _TYPE_=_PARMS_ observations contain for each activation function the $p = 2c + 1$ parameter estimates. Here, the _MEAN_ variable contains the value for the optimization criterion and the _STDEV_ variable contains the accuracy value of the prediction.

OUT=*SASdataset* : specifies an output data set generated by PROC DMNEURL which contains the predicted values (*posteriors*) and residuals for all observations in the DATA= input data set.

Variables of the output data set:

$idvarnam_i$ values of all ID variables

TARGET (character) name of the target

STAGE number of stage

_P_ predicted value ($\hat{y}$)

_R_ residual ($y - \hat{y}$)

The following variables are added if a DECISION statement is used:

BSTDEC

CONSEQ

EVALUE expected profit or cost value

$decvar_i$ expected values for all decision variables

The number of observations in the OUT= data set agrees with that of the DATA= input data set.

TESTOUT=*SASdataset* :
specifies an output data set which is in structur identical to the OUT= output data set but relates to the information given in the TESTDATA= input data set rather than that of the DATA= input data set used in the OUT= output data set. The number of observations in the TESTOUT= data set agrees with that of the TESTDATA= input data set.

OUTFIT=*SASdataset* :
specifies an output data set generated by PROC DMNEURL which contains a number of fit indices for each stage and for the final model estimates. For a binary target (response variable) it also contains the frequencies of the 2×2 accuracy table of the best fit at the final stage. The same information is additionally provided if a TESTDATA= input data set is specified.

Variables of the output data set:

TARGET (character) name of the target

DATA (character) specifies the data set to which the fit criteria correspond: =TRAINING: fit criteria belong to DATA= input data set =TESTDATA: fir criterai belong to TESTDATA= input data set

TYPE (character) describes type of observation

TYPE=_FITIND_ for fit indices;

TYPE=_ACCTAB_ for frequencies of accuracy table (only for binary target)

STAGE number of stages in the estimation process

SSE sum-of-squared error of solution

RMSE root mean squared error of solution

ACCU percentage of accuracy of prediction (only for categorical target)

AIC Akaike information criterion

SBC Schwarz' information criterion

The following variables are added if a DECISION statement is used:

PROF

APROF

LOSS

ALOSS

_IC_

ROI

OUTSTAT=*SASdataset* :
specifies an output data set generated by PROC DMNEURL which contains all eigenvalues and eigenvectors of the $X'X$ matrix. When this option is specified, no other computations are performed and the procedure terminates after writing this data set.

Variables of the OUTSTAT= output data set:

TYPE (character) type of observation

EIGVAL contains different numeric information

$varname_i$ variables in the model; the first variables correspond to CLASS (categorical) the remaining variables are continuously (interval or ratio) scaled. Note, that for nonbinary CLASS (nominal or ordinal categorical) variables a set of binary dummy variables is created. In those cases the prefix of variable names $varname_i$ used for a group of variables in the data set may be the same for a successive group of variables which differs only by a numeric suffix.

Observations of the OUTSTAT= output data set:

1. The first three observations, _TYPE_=_V_MAP_ and _TYPE_=_C_MAP_, contain the mapping indices between the variables used in the model and the number of the variables in the data set. The _EIGVAL_ variable contains the number of index mappings. This is the same information as in the first observation of the OUTEST= data set, except that here the _TYPE_=_EIGVAL_ variables replaces the _TYPE_=_MEAN_ variable in the OUTEST= data set.
2. The _TYPE_=_EIGVAL_ observation contains the sorted eigenvalues of the $X'X$ matrix.
3. The _TYPE_=_EIGVEC_ observations contain a set of n eigenvectors of the $X'X$ matrix. Here, the _EIGVAL_ variable contains the eigenvalue to which the eigenvector corresponds.

ABSGCONV, ABSGTOL : $r \geq 0$
specifies an absolute gradient convergence criterion for the default (OPTCRIT=SSE) optimization process. See the document of PROC NLP in SAS/OR for more details. Default is ABSGCONV=5e-4 in general and ABSCONV=1e-3 for FUNCTION=EXP.

CORRDF : specifies that the correct number of degrees of freedom is used for the values of RMSE, AIC, and SBC. Without specifying CORRDF the error degrees of freedom are computed as $W - p$, where W is the sum of weights (if the WEIGHT statement is not used, each observation has a weight of 1 assigned, and W is the total number of observations) and p is the number of parameters. When CORRDF is spefified the value p is replaced by the rank of the joint Jacobian.

COV, CORR : specifies that a covariance or correlation matrix is used for computing eigenvalues and eigenvectors compatible with the PRINCOMP procedure. The COV and CORR options are valid only if an OUTSTAT= data set is specified. If neither COV nor CORR are specified, the eigenvalues and eigenvectors of the cross product matrix $X^T X$ are computed and written to the OUTSTAT= data set.

CRITWGT=r : $r > 0$
specifies a positive weight for a weighted least squares fit. Currently this option is valid only for binary target. Values of $r > 1.$ will enforce a better fit of the (1,1) entry in the accuracy table which may be useful for fitting rare events. Values of $0 < r < 1.$ will enforce a better fit of the (0,0) entry in the accuracy table. Note, that values for r which are far away from $r = 1$ will reduce the fit quality of the remaining entries in the frequency table. At this time values of either $1 < r < 2$ or $.5 < r < 1$ are preferred.

CUTOFF=r : $0 < r < 1$
specifies a cutoff threshold for deciding when a predicted value of a binary response is classified as 0 or 1. The default is $cutoff = .5$. If the value of the posterior, $\hat(y_i)$, for observation i is smaller the specified cutoff value, the observation is counted in the first column of the accuracy table (i.e. as 0), otherwise it is counted in the second column (i.e. as 1). For nonbinary target the cutoff= value is not used.

GCONV, GTOL : $r \geq 0$
specifies a relative gradient convergence criterion for the optimization process. See the document of PROC NLP in SAS/OR for more details. Default is GCONV=1e-8.

FCRIT specifies that the probability of the F test is being used for the selction of principal components rather than the default R^2 criterium.

MAXCOMP=i : $2 \leq i \leq 8$
specifies an upper bound for the number of components selected for predicting the target in each stage. Good values for MAXCOMP are inbetween 3 and 5. Note, that the computer time and core memory will increase superlinear for

larger values than 5. There is one memory allocation which takes n^m long integer values, where n is the value specified with the NPOINT= option and m is the value specified by the MAXCOMP= option. The following table lists values of $4n^m/1000000$ for specific combinations of (n, m). This is the actual memory requirement in Megabytes assuming that a long integer takes 4 bytes storage.

n	m=3	m=4	m=5	m=6	m=7	m=8
5	0	0	0	0	0	2
7	0	0	0	0	3	23*
9	0	0	0	2	19*	172
11	0	0	1	7*	78	857
13	0	0	2*	19	250	3263
15	0	0*	3	46	683	10252
17	0*	0	6	97	1641	27903
19	0	1	10	188	3575	67934

The trailing asterisk indicates the default number of points for a given number of components. Therefore, values larger than 8 for i in `MAXCOMP=i` are reduced to this upper range. It seems to be better to increase the value i of the MAXSTAGE=i option when higher precision is requested.

MAXFUNC=i : $i \geq 0$
specifies an upper bound for the number of function calls in each optimization. The default is MAXFUNC=500. Normally the default number of function calls will be sufficient to reach convergence. Larger values should be used if the iteration history indicates that the optimization process was close to a promising solution but would have needed more than the specified number of function calls. Smaller values should be specified when a faster but suboptimal solution may be sufficient.

MAXITER=i : $i \geq 0$
specifies an upper bound for the number of iterations in each optimization. The default is MAXITER=200. Normally the default number of iterations will be sufficient to reach convergence. Larger values should be used if the iteration history indicates that the optimization process was close to a promising solution but would have needed more than the specified number of iterations. Smaller values should be specified when a faster but suboptimal solution may be sufficient.

MAXROWS=i : $i \geq 1$
specifies an upper bound for the number of independent variables selected for the model. More specific, this is an upper bound for the rows and columns of the X'X matrix of the regression problem. The default is $maxrows = 3000$. Note, that the $X^T X$ matrix used for the stepwise regression takes $n_{rows}(n_{rows} + 1)/2$ double precision values storage in RAM. For the default maximum size of $n_{rows} = 3000$ you will need more than $3000 * 1500 * 8$ bytes RAM, which is slightly more than 36 megabytes.

MAXSTAGE=i : $i \geq 1$
specifies an upper bound for the number of stages of estimation. If

MAXSTAGE is not specified, the default is MAXSTAGE=5. When a missing value is specified, the multistage estimation process is terminated

- if the sum-of-squares residual in the component selection process changes by less than 1%
- or when an upper range of 100 stages are processed.

That means, not specifying MAXSTAGE= or specifying a missing value are treated differently. Large values for MAXSTAGE= may result in numerical problems: the discretization error may be too large and the fit criterion does no longer improve and can actually become worse. In such a case the stagewise process is terminated with the last good stage.

MAXSTPT=i : $i \geq 1$
specifies the number of values of the objective function inspected for the start of the optimization process. Larger values than the default value may improve the result of the optimization especially when more than three components are used. The default is MAXSTPT=250.

MAXVECT=i : $i \geq 2$
specifies an upper bound for the number of eigenvectors made available for selection. The default is MAXVECT=400. Smaller values should be used only if there are memory problems for storing the eigenvectors when too many variables are included in the analysis. The specified value for MAXVECT= cannot be smaller than that for MINCOMP=. If the specified value of MAXVECT= is larger than the value for MAXROWS= it is reduced to the value of MAXROWS=.

MEMSIZ=i : $i \geq 1$
For interval targets and in a multiple stage process some memory consuming operations are being performed. For very large data sets the computations may significantly depend on the size of the available RAM memory for those computations. By default MEMSIZ=8 specifies the availability of 8 mb of RAM for such operations. Since other operations need additional memory not more than 25 percent of the total amount of memory should be specified here. If you are running out of memory during the DMNEURL run, you may actually specify a smaller amout than the default 8 mb.

MINCOMP=i : $2 \leq i \leq 8$
specifies a lower bound for the number of components selected for predicting the target in each stage. The default is MINCOMP=2. The specified value for MINCOMP= cannot be larger than that for MAXCOMP=. The MINCOMP= specification may permit the selection of components which otherwise would be rejected by the STOPR2= option. PROC DMNEURL may override the specified value when the rank of the $X'X$ matrix is less than the specified value.

NOMONITOR :
supresses the output of the status monitor indicating the progress made in the computations.

NOPRINT :
supresses all output printed in the output window.

NPOINT=i : $5 \leq i \leq 19$
number of discretization points (should be even inbetween 5 and 19). By default NPOINT= is selected depending on the number of components selected in the model using the MINCOMP= and MAXCOMP= options.

OPTCRIT=$SSE|ACC|WSSE$:
specifies the criterion for the optimization:

OPTCRIT=SSE the sum-of-squares error is minimzed.

OPTCRIT=ACC a measure of the accuracy rate is maximized. (For interval target the Goodman-Kruskal γ is applied on a frequency table defined by deciles of the actual target value.)

OPTCRIT=WSSE a weighted sum-of-squares criterion is minimzed. When this option is specified the weight must be specified using the CRITWGT= option. Currently this option is valid only for binary target.

PALL :

- If an OUTSTAT= data set is specified, i.e. only principal components are being computed, the following table illustrates the output options:

Output	**PSHORT**	**default**	**PALL**
Simple Stat	x	x	x
Eigenvalues	x	x	x

  If PMATRIX is specified, the $X'X$, the covariance, or the correlation matrix is also printed (depending on COV and CORR option).

- If no OUTSTAT= data set is specified, i.e. a nonlinear model based on activation and link functions is being optimized, the following table illustrates the output options:

Output	**NOPRINT**	**PSHORT**	**default**	**PALL**

PMATRIX :
This option is valid only if an OUTSTAT= data set is specified, i.e. when DMNEURL is used only for computing eigenvalues and eigenvectors of the $X'X$, covariance, or correlation matrix. If PMATRIX is specified, this matrix is being printed. Since this matrix may be very large its printout is not included by that of the PALL option.

POPTHIS :
print the detailed histories of all optimization processes. The PALL option includes only the summarized forms of the history output (header and result).

PSHORT :
see the PALL option for the amount of output being printed.

PTABLE :
specifies the output of accuracy tables. This option is invoked automatically if the PALL option is specified.

SELCRIT=$SSE|ACC|WSSE$:
specifies the criterion for selecting the best result among all of the activation functions:

SELCRIT=SSE select solution with smallest sum-of-squares error.

SELCRIT=ACC select solution with largest accuracy rate. (For interval target the Goodman-Kruskal γ is applied on a frequency table defined by deciles of the actual target value.)

SELCRIT=WSSE select solution with smallest weighted sum-of-squares error. This option is valid only for binary target. When this option is specified the weight must be specified using the CRITWGT= option.

SINGULAR=r :
specifies a criterion for the singularity test. The default is $r = 1.e - 8$ and should not be changed if there are no significant reasons to do so.

STOPR2=r :
specifies a lower value for the incremental model R^2 value at which the variable selection process is stopped. The STOPR2= criterion is used only for the R2 values of the components selected in the range specified by the MINCOMP= and MAXCOMP= values. The default is $r = 5e - 5$.

TESTDMDB :
permits the use of a data set generated by PROC DMDB to be specified as a TESTDATA= input data set. If this option is not specified, the data set specified with TESTDATA= must be a normal SAS data set.

DECISION Statement

For the syntax of the DECISION statement see the document of PROC DECIDE.

FUNCTION and LINK Statement

An activation function f and a link function g may be specified for the mapping inbetween the component scores s_{ij} and the values y_i of the response variable (stage=0) (or the residuals in stage > 0),

$$\hat{y}_i = g(f^{(k)}(s_{ij}, \theta_j)), \quad i = 1, \ldots, N, j = 1, \ldots, p$$

for each activation function $f^{(k)}, k = 1, \ldots, K$. The FUNCTION and LINK statement can be used to specify the functions $f^{(k)}$ and g:

FUNCTION statement One or more of the following activation functions f can be specified

SQUARE	$(a + b * x) * x$
TANH	$a * tanh(b * x)$
ARCTAN	$a * atan(b * x)$
LOGIST	$exp(a * x)/(1. + exp(b * x)$
GAUSS	$a * exp(-(b * x)^2)$
SIN	$a * sin(b * x)$
COS	$a * cos(b * x)$
EXP	$a * exp(b * x)$

If more than one function $f^{(k)}$ is specified, each of the specified functions is evaluated during the estimation process and the best result w.r.t. to the sum-of-squares residual or accuracy (see SELCRIT= option) is selected. By default all available activation functions are used.

LINK statement Currently only one of the following link functions can be used for the outer function g:

IDENT	x
LOGIST	$exp(x)/(1. + exp(x)$
RECIPR	$1/x$

By default, the LOGIST function is used for a binary target and the IDENT(ity) function is used for interval target. In a parallelized version of PROC DMNEURL, multiple functions g could be feasible.

TARGET Statement

TARGET *onevar* **;**

One variable name may be specified identifying the target (response) variable for the two regressions. Note, that one or more target variables may be specified already with the PROC DMDB run. If a target is specified in the PROC DMDB run, it must not be specified in the PROC DMNEURL call.

VAR or VARIABLES Statement

VAR *varlist* **;**

VARIABLES *varlist* **;**

All variables, numeric (interval) and categorical (CLASS) variables which may be used for independent variables are specified with the VAR statement.

FREQ or FREQUENCY Statement

FREQ *onevar* **;**

FREQUENCY *onevar* **;**

One numeric (interval scaled) variable may be specified as a FREQ variable. Note, that a rational value is truncated to the next integer. It is recommended to specify the FREQ variable already in the PROC DMDB run. Then the information is saved in the catalog and that variable is used automatically as a FREQ variable in PROC DMNEURL. This also ensures that the FREQ variable is being used automatically by all other PROCs in the EM project.

WEIGHT or WEIGHTS Statement

WEIGHT *onevar* **;**

WEIGHTS *onevar* **;**

One numeric (interval scaled) variable may be specified as a WEIGHT variable. It is recommended to specify the WEIGHT variable already in the PROC DMDB invocation. Then the information is saved in the catalog and that variable is used automatically as a FREQ variable in PROC DMNEURL.

Scoring the Model Using the OUTEST= Data set

The score value $\hat{y}_i$ is computed for each observation $i = 1, \ldots, N_{obs}$ with nonmissing value of the target (response) variable y of the input data set. All information needed for scoring an observation of the DMDB data set is contained in the output of the OUTEST= data set. First an observation from the input data set is mapped into a vector v of n new values in which

1. CLASS predictor variables with K categories are replaced by $K+1$ or K dummy (binary) variables, depending on the fact whether the variable has missing values or not.

2. Missing values in interval predictor variables are replaced by the mean value of this variable in the DMDB data set. This mean value is taken from the catalog of the DMDB data set.

3. The values of a WEIGHT or FREQ variable are multiplied into the observation.

4. For an interval target variable y its value is transformed into the interval [0,1] by the relationship
$$y_i^{new} = \frac{y_i - y_{min}}{y_{max} - y_{min}}$$

5. All predictor variables are transformed into values with zero mean and unit standard deviation by
$$x_{ij}^{new} = \frac{x_{ij} - Mean(x_j)}{StDev(x_j)}$$
The values for $Mean(x_j)$ and $StDev(x_j)$ are listed in the OUTEST= data set.

This means, that in the presence of CLASS variables the n-vector v has more entries than the observation in the data set.
The scoring is additive across the stages. The following information is available for scoring each stage

- c components (eigenvectors) z_l each of dimension n
- the best activation function f and a specified link function g
- the $p = 2c + 1$ optimal parameter estimates θ_j

For each component z_l we compute the component score u_l,

$$u_l = \sum_{j=1}^{n} z_{lj} v_j$$

similar to principal component analysis. With those values u_l the model can be expressed as

$$\hat{y} = \sum_{istage=1}^{nstage} g(f(u, \theta))$$

where f is the best activation function and g is the specified link function.
In other words, this means, that given the u_l the value w is computed from

$$w = \theta_0 + \sum_{l} f(u_l, a_l, b_l)$$

where a_l and b_l are two of the $p = 2 * c + 1$ optimal parameters θ and f is defined as

SQUARE	$w = (a + b * u) * u$
TANH	$w = a * tanh(b * u)$
ARCTAN	$w = a * atan(b * u)$
LOGIST	$w = exp(a * u)/(1. + exp(b * u))$
GAUSS	$w = a * exp(-(b * u)^2)$
SIN	$w = a * sin(b * u)$
COS	$w = a * cos(b * u)$
EXP	$w = a * exp(b * u)$

For the first component $a_1 = \theta_1$ and $b_1 = \theta_2$, for the second component $a_2 = \theta_3$ and $b_2 = \theta_4$, and for the last component $a_c = \theta_{p-1}$ and $b_c = \theta_p$ are used.
The link function g is applied on w and yields to h

IDENT	$h = w$
LOGIST	$h = exp(w)/(1. + exp(w))$
RECIPR	$h = 1/w$

Across all stages the values of h are added to the predicted value (posterior) $\hat{y}$.