SAS/STAT® 12.1 User's Guide
The QUANTSELECT Procedure (Chapter)

SAS® Documentation

This document is an individual chapter from *SAS/STAT® 12.1 User's Guide*.

The correct bibliographic citation for the complete manual is as follows: SAS Institute Inc. 2012. *SAS/STAT® 12.1 User's Guide*. Cary, NC: SAS Institute Inc.

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

Electronic book 1, August 2012

Chapter 78
The QUANTSELECT Procedure (Experimental)

Contents

Overview: QUANTSELECT Procedure **6564**

Features 6564

Getting Started: QUANTSELECT Procedure **6565**

Syntax: QUANTSELECT Procedure **6576**

PROC QUANTSELECT Statement 6576

BY Statement 6583

CLASS Statement 6584

EFFECT Statement 6587

MODEL Statement 6588

OUTPUT Statement 6595

PARTITION Statement 6596

WEIGHT Statement 6596

Details: QUANTSELECT Procedure **6597**

Effect Selection Methods 6597

Full Model Fitted (NONE) 6597

Forward Selection (FORWARD) 6597

Backward Elimination (BACKWARD) 6597

Stepwise Selection (STEPWISE) 6597

LASSO Method (LASSO) 6598

Criteria Used in Model Selection Methods 6598

Macro Variables That Contain Selected Models 6600

Using Validation and Test Data 6602

Displayed Output 6604

ODS Table Names 6606

ODS Graphics 6607

Example: QUANTSELECT Procedure **6608**

Example 78.1: Simulated Data Example 6608

References **6612**

Overview: QUANTSELECT Procedure

Quantile regression, which was introduced by Koenker and Bassett (1978), is a modern method that models the effects of covariates on the conditional quantiles of a response variable. The QUANTSELECT procedure performs effect selection in the framework of quantile regression. A variety of effect selection methods are available, including greedy methods and penalty methods. The QUANTSELECT procedure offers extensive capabilities for customizing the effect selection processes with a variety of candidate selecting, effect-selection stopping, and final-model choosing criteria. PROC QUANTSELECT also provides graphical summaries for the effect selection processes.

The QUANTSELECT procedure compares most closely to the GLMSELECT and QUANTREG procedures. PROC GLMSELECT performs effect selection in the framework of general linear models. PROC QUANTREG supports a variety of estimation and inference methods for quantile regression but does not directly provide effect selection facilities. The QUANTSELECT procedure, as a counterpart of PROC GLMSELECT for quantile regression, fills this gap.

The QUANTSELECT procedure focuses on linear quantile models for univariate responses and offers great flexibility for and insight into the effect selection algorithm. The QUANTSELECT procedure inherits most of its syntax from PROC GLMSELECT and PROC QUANTREG. The QUANTSELECT procedure provides results that are similar to those of PROC GLMSELECT and PROC QUANTREG. These results (displayed tables, output data sets, and macro variables) make it easy to explore the selected models in PROC QUANTREG.

Features

The main features of the QUANTSELECT procedure are as follows:

- supports the following model specifications:
  - interaction (crossed) effects and nested effects
  - constructed effects such as regression splines
  - hierarchy among effects
  - partitioning of data into training, validation, and testing roles
- provides the following selection controls:
  - multiple methods for effect selection
  - selection for quantile process and single quantile levels
  - selection of individual or grouped effects
  - selection based on a variety of selection criteria
  - stopping rules based on a variety of model evaluation criteria
- produces the following display and output:
  - graphical representation of the selection process
  - output data sets that contain predicted values and residuals
  - an output data set that contains the design matrix
  - macro variables that contain selected effects

The QUANTSELECT procedure supports the following effect selection methods. These methods are explained in detail in the section “Effect Selection Methods” on page 6597.

- Forward selection starts with no effects or with forced-in effects in the model and adds more effects.
- Backward elimination starts with all effects in the model and deletes effects.
- Stepwise regression is similar to the forward selection method except that effects already in the model do not necessarily stay there.
- LASSO regression adds and deletes effects based on a constrained version of estimated check risk where the L1-norm of regression coefficients is penalized (Tibshirani 1996; Belloni and Chernozhukov 2011). Adaptive LASSO (Zou 2006; Wu and Liu 2009) is implemented as a special case of LASSO methods where the L1-norm of certain weighted regression coefficients is penalized. See the discussion in the section “LASSO Method (LASSO)” on page 6598 for additional details. The QUANTSELECT procedure uses LASSO methods only to determine the adding and dropping covariate effects at a step; a post-penalized model that is associated with the step is refitted without penalty, and the selection criteria and the parameter estimates are from the post-penalized model.

The QUANTSELECT procedure is intended primarily as an effect selection procedure and does not include regression diagnostics and hypothesis testing. The intention is that you use the QUANTSELECT procedure to select a model or a set of models, where each model contains a set of selected effects, and then you can further investigate these models by using PROC QUANTREG or other analytic tools.

Getting Started: QUANTSELECT Procedure

This example demonstrates how you can use the QUANTSELECT procedure to select covariate effects for quantile regression. The data set under investigation contains salary and performance information for Major League Baseball (MLB) players, excluding pitchers, who played at least one game in both the 1986 and 1987 seasons. The salaries (Time Inc. 1987) are for the 1987 season, and the performance measures are from 1986 (Reichler 1987).

```
data baseball;
   length team $ 12;
   input name $ 1-18 nAtBat nHits nHome nRuns nRBI nBB
         yrMajor crAtBat crHits crHome crRuns crRbi crBB
         League $ Division $ Team $ Position $ nOuts nAssts
         nError Salary;
   label name="Player's Name"
      nAtBat="Times at Bat in 1986"
      nHits="Hits in 1986"
      nHome="Home Runs in 1986"
      nRuns="Runs in 1986"
      nRBI="RBIs in 1986"
      nBB="Walks in 1986"
      yrMajor="Years in the Major Leagues"
      crAtBat="Career times at bat"
      crHits="Career Hits"
      crHome="Career Home Runs"
      crRuns="Career Runs"
      crRbi="Career RBIs"
      crBB="Career Walks"
      League="League at the end of 1986"
      Division="Division at the end of 1986"
      Team="Team at the end of 1986"
      Position="Position(s) in 1986"
      nOuts="Put Outs in 1986"
      nAssts="Assists in 1986"
      nError="Errors in 1986"
      Salary="1987 Salary in $ Thousands";
      if League='American' and Division='East' then Div='AE';
      if League='American' and Division='West' then Div='AW';
      if League='National' and Division='East' then Div='NE';
      if League='National' and Division='West' then Div='NW';
      logSalary = log(Salary);
   datalines;
Allanson, Andy       293    66     1    30    29    14
                       1   293    66     1    30    29    14
                    American East Cleveland C 446 33 20 .
Ashby, Alan          315    81     7    24    38    39

   ... more lines ...

Wilson, Willie       631   170     9    77    44    31
                      11  4908  1457    30   775   357   249
                    American West KansasCity CF 408 4 3 1000
;
```

Suppose you want to investigate how the MLB players' salaries for the 1987 season depend on performance measures for the players' previous season and MLB careers. As a starting point for such a analysis, you can use the following statements to obtain a parsimonious conditional median model at $\tau = 0.5$:

```
proc quantselect data=baseball;
   class Div;
   model Salary = nAtBat nHits nHome nRuns nRBI nBB yrMajor crAtBat
                  crHits crHome crRuns crRbi crBB nAssts nError nOuts
                  Div
         / selection=lasso(adaptive stop=aic choose=sbc sh=7);
run;
```

The SELECTION=LASSO(ADAPTIVE) option in the MODEL statement specifies the adaptive LASSO method (Zou 2006), which controls the effect selection process. The STOP=AIC option specifies that Akaike's information criterion (AIC) be used to determine the stopping condition. The CHOOSE=SBC option specifies that the Schwarz Bayesian information criterion (SBC) be used to determine the final selected model. The SH= option specifies the number of stop horizons, which requests that the selection process be stopped whenever the STOP= criterion values at step $s + 1, \cdots, s + \text{SH}$ are worse than those for step s for some $s \in \{0, 1, \cdots\}$.

Figure 78.1 shows the "Model Information" table, which indicates the effect selection settings. You can see that the default quantile type is single level, so this effect selection is effective only for $\tau = 0.5$.

Figure 78.1 Model Information

```
                    The QUANTSELECT Procedure

                        Model Information

               Data Set                 WORK.BASEBALL
               Dependent Variable              Salary
               Selection Method        Adaptive LASSO
               Quantile Type             Single Level
               Stop Criterion                     AIC
               Choose Criterion                   SBC
```

Figure 78.2 summarizes the effect selection process, which starts with an intercept-only model at step 0. At step 1, the effect that corresponds to the career runs is added to the model that reduced the AIC value from 2691.6511 to 2510.7297. You can see that step 10 has the minimum AIC and that step 7 has the minimum SBC. Common sense also tells you that the SBC favors a smaller model than the AIC.

Figure 78.2 Selection Summary

```
                    The QUANTSELECT Procedure

                          Quantile = 0.5

                        Selection Summary

                                  Number
                 Effect          Effects
         Step    Entered              In          AIC           SBC

            0    Intercept             1    2691.6511     2695.2232
    -----------------------------------------------------------------
            1    crRuns                2    2510.7297     2517.8740
            2    nHits                 3    2470.4807     2481.1971
            3    crHome                4    2463.5953     2477.8839
            4    nBB                   5    2463.7806     2481.6414
            5    nOuts                 6    2455.6212     2477.0541
            6    Div AW                7    2451.4609     2476.4660
            7    nAtBat                8    2445.0446     2473.6218*
            8    crBB                  9    2445.5432     2477.6926
            9    nHome                10    2443.4818     2479.2033
           10    nRuns                11    2442.6036*    2481.8973
           11    crAtBat              12    2442.8387     2485.7046
           12    yrMajor              13    2443.5374     2489.9754
           13    nError               14    2445.2085     2495.2187
           14    Div NE               15    2446.4042     2499.9865
           15    nRBI                 16    2448.2003     2505.3547
           16    nAssts               17    2449.9731     2510.6997

                   * Optimal Value Of Criterion
```

Figure 78.3 shows that the selection process stopped at a local minimum of the STOP= criterion, which is step 10. According to the SH=7 option, the effect selection process is stopped at step 10 because all the AIC values for step 11 through step 17 are no less than the AIC at step 10. Step 17 is ignored in the selection summary table because it is the last step.

Figure 78.3 Stop Reason

```
Selection stopped at a local minimum of the AIC criterion.
```

Figure 78.4 shows how the final selected model is determined. CHOOSE=SBC is specified in this example, so the model at step 7 is chosen as the final selected model.

Figure 78.4 Selection Reason

```
The model at step 7 is selected where SBC is 2473.62.
```

Figure 78.5 shows the final selected effects and Figure 78.6 shows the parameter estimates for the final selected model.

Figure 78.5 Selected Effects

```
                    The QUANTSELECT Procedure

                         Quantile = 0.5

Selected Effects: Intercept nAtBat nHits nBB crHome crRuns nOuts Div AW
```

Figure 78.6 Parameter Estimates

```
                        Parameter Estimates

                                                 Standardized
          Parameter     DF         Estimate          Estimate

          Intercept      1       -18.187539                 0
          nAtBat         1        -1.582714         -0.500417
          nHits          1         7.044354          0.686968
          nBB            1         2.053726          0.097911
          crHome         1         1.429926          0.272726
          crRuns         1         0.425955          0.316167
          nOuts          1         0.282803          0.175489
          Div AW         1       -57.671778         -0.056862
```

Quantile regression can fit a conditional quantile model at any quantile level $\tau \in (0, 1)$, so it can describe the entire distribution of a response variable conditional on covariate effects. To further investigate the effects that might affect the MLB players' salaries, you can also conduct effect selection at $\tau = 0.1$ and $\tau = 0.9$, which correspond to low-end salaries and high-end salaries respectively. The following statements use the same selection settings that are used in the previous program:

```
proc quantselect data=baseball;
   class Div;
   model Salary = nAtBat nHits nHome nRuns nRBI nBB yrMajor crAtBat
                  crHits crHome crRuns crRbi crBB nAssts nError nOuts
                  Div
         / quantiles=0.1 0.9 selection=lasso(adaptive stop=aic choose=sbc sh=7);
run;
```

Figure 78.7 shows the effect selection summary with $\tau = 0.1$.

Figure 78.7 Selection Summary: $\tau = 0.1$

```
                          The QUANTSELECT Procedure

                                Quantile = 0.1

                               Selection Summary

                                          Number
              Effect      Effect         Effects
      Step    Entered     Removed             In         AIC            SBC

         0    Intercept                        1     2008.3489      2011.9211
------------------------------------------------------------------------------
         1    crRuns                           2     1918.7675      1925.9118
         2    nHits                            3     1897.2425      1907.9590*
         3    yrMajor                          4     1897.2476      1911.5362
         4    crBB                             5     1896.1765      1914.0373
         5    nBB                              6     1894.1257      1915.5587
         6    Div AW                           7     1891.0577      1916.0628
         7    crHome                           8     1892.2575      1920.8348
         8                crHome               7     1891.0577      1916.0628
         9    nAtBat                           8     1889.6365*     1918.2137
        10    nHome                            9     1891.2595      1923.4089
        11    crHome                          10     1890.7463      1926.4679
        12    nRBI                            11     1892.7421      1932.0358
        13    Div NW                          12     1892.9745      1935.8404
        14                nRBI                11     1891.3217      1930.6154
        15    crAtBat                         12     1891.8810      1934.7468

                         * Optimal Value Of Criterion
```

Figure 78.8 shows the parameter estimates for the final selected model with $\tau = 0.1$. You can see from Figure 78.8 that low-end salaries for MLB players depend mainly on career runs and hits in 1986.

Figure 78.8 Parameter Estimates: $\tau = 0.1$

```
                          The QUANTSELECT Procedure

                                Quantile = 0.1

                             Parameter Estimates

                                                  Standardized
               Parameter    DF        Estimate        Estimate

               Intercept     1       -4.397043               0
               nHits         1        0.878564        0.085678
               crRuns        1        0.327350        0.242977
```

Figure 78.9 shows the effect selection summary with $\tau = 0.9$.

Figure 78.9 Selection Summary: $\tau = 0.9$

```
                          The QUANTSELECT Procedure

                                Quantile = 0.9

                              Selection Summary

                                          Number
              Effect      Effect         Effects
      Step    Entered     Removed             In          AIC            SBC

         0    Intercept                        1    2436.7289      2440.3011
-----------------------------------------------------------------------------
         1    crHits                           2    2197.4349      2204.5792
         2    crRbi                            3    2183.6148      2194.3313
         3    nHits                            4    2113.2757      2127.5643
         4                crRbi                3    2127.8632      2138.5797
         5    crRbi                            4    2113.2757      2127.5643
         6                crRbi                3    2127.8632      2138.5797
         7    crRbi                            4    2113.2757      2127.5643
         8    crHome                           5    2099.2203      2117.0811
         9                crRbi                4    2099.3891      2113.6777
        10    crRbi                            5    2099.2203      2117.0811
        11                crRbi                4    2099.3891      2113.6777
        12    nOuts                            5    2067.1926      2085.0533
        13    Div AW                           6    2048.2393      2069.6723
        14    crRuns                           7    2028.8040      2053.8090
        15    nAtBat                           8    2012.8195      2041.3968
        16                crHits               7    2017.0290      2042.0341
        17    crRbi                            8    2009.3551      2037.9324
        18    crAtBat                          9    2011.2415      2043.3908
        19                crRbi                8    2011.4053      2039.9825
        20    crRbi                            9    2011.2415      2043.3908
        21                crAtBat              8    2009.3551      2037.9324
        22    crAtBat                          9    2011.2415      2043.3908
        23    nBB                             10    2004.5033      2040.2249
        24                crAtBat              9    2003.1023      2035.2517*
        25    crAtBat                         10    2004.5033      2040.2249
        26                crAtBat              9    2003.1023      2035.2517
        27    crAtBat                         10    2004.5033      2040.2249
        28    nError                          11    2004.2230      2043.5167
        29    crHits                          12    2003.0544      2045.9203
        30    Div NE                          13    2001.9603      2048.3983
        31    Div AE                          14    2001.8349*     2051.8451
        32    nRuns                           15    2003.5961      2057.1784
        33    nHome                           16    2004.2721      2061.4266
        34    nRBI                            17    2006.0023      2066.7289
        35    yrMajor                         18    2007.9975      2072.2963
        36    crBB                            19    2009.9514      2077.8223

                         * Optimal Value Of Criterion
```

Figure 78.9 *continued*

```
                           The QUANTSELECT Procedure

                                 Quantile = 0.9

                                Selection Summary

                                           Number
               Effect      Effect         Effects
       Step    Entered     Removed             In          AIC            SBC

         37    nAssts                          20    2011.9095      2083.3525

                          * Optimal Value Of Criterion
```

Figure 78.10 shows the parameter estimates for the final selected model with $\tau = 0.9$.

Figure 78.10 Parameter Estimates: $\tau = 0.9$

```
                              Parameter Estimates

                                                   Standardized
                Parameter     DF        Estimate       Estimate

                Intercept      1       92.893875              0
                nAtBat         1       -1.858170      -0.587509
                nHits          1        8.155573       0.795335
                nBB            1        3.392794       0.161751
                crHome         1        3.191472       0.608700
                crRuns         1        1.394317       1.034939
                crRbi          1       -0.913371      -0.664951
                nOuts          1        0.437241       0.271323
                Div AW         1     -167.110005      -0.164764
```

To visually illustrate how the model evolves through the selection process, the QUANTSELECT procedure provides the coefficient plot, the average check loss plot, and several criterion plots in either packed or unpacked forms. You can request these plots by using the PLOTS= option. The following statements request all the plots for the baseball data at $\tau = 0.1$; they also use the STOP=AIC criterion, the CHOOSE=SBC criterion, and the SH=7 option:

```
ods graphics on;
proc quantselect data=baseball plot=all;
   class Div;
   model Salary = nAtBat nHits nHome nRuns nRBI nBB yrMajor crAtBat
                  crHits crHome crRuns crRbi crBB nAssts nError nOuts
                  Div
         / quantiles=0.1 selection=lasso(adaptive stop=aic choose=sbc sh=7);
run;
```

Figure 78.11 shows the progression of the parameter estimates as the selection process proceeds.

Figure 78.11 Coefficient Panel: $\tau = 0.1$

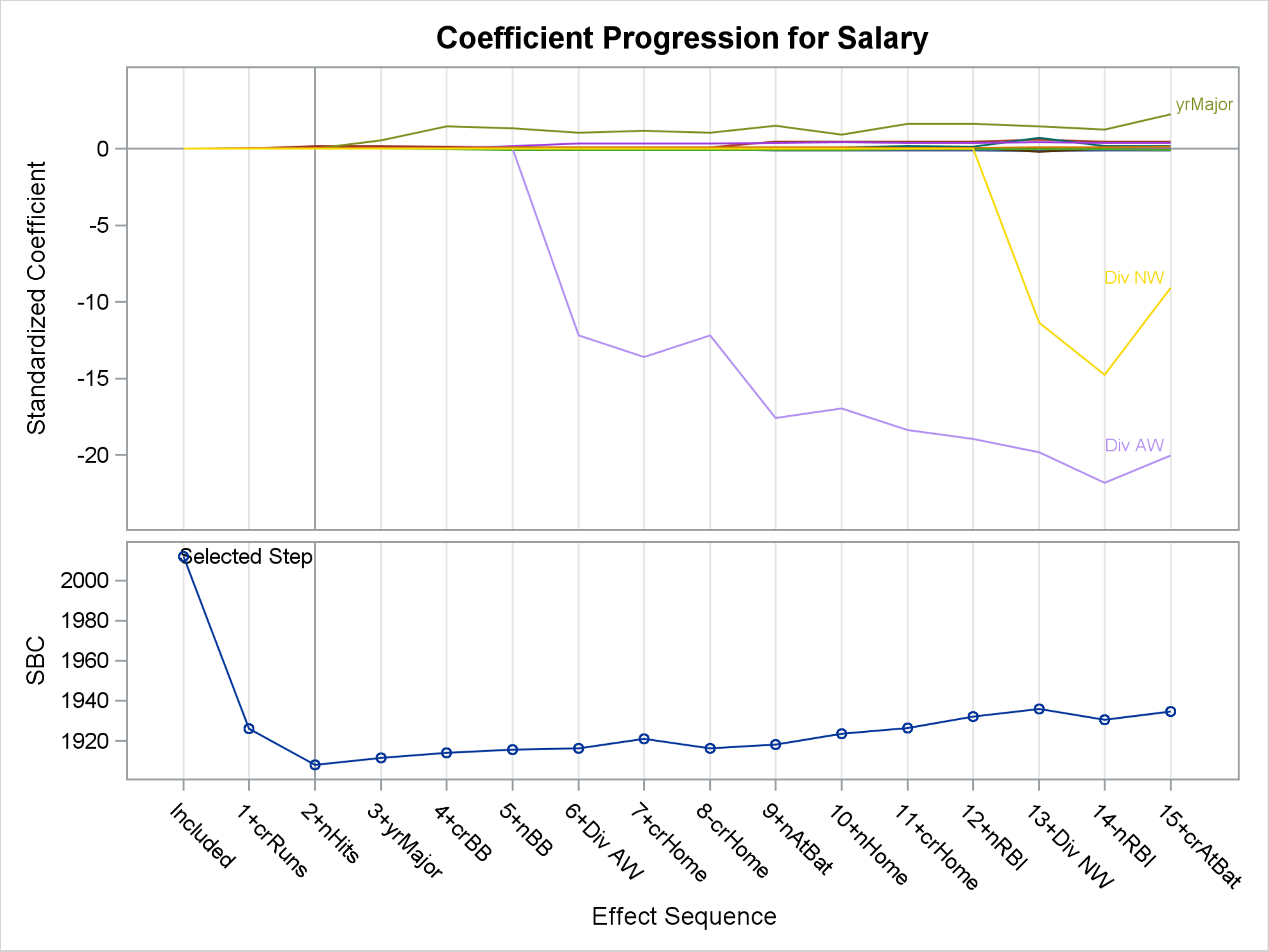


Figure 78.12 shows the progression of the average check losses as the selection process proceeds.

Figure 78.12 Average Check Loss Plot: $\tau = 0.1$

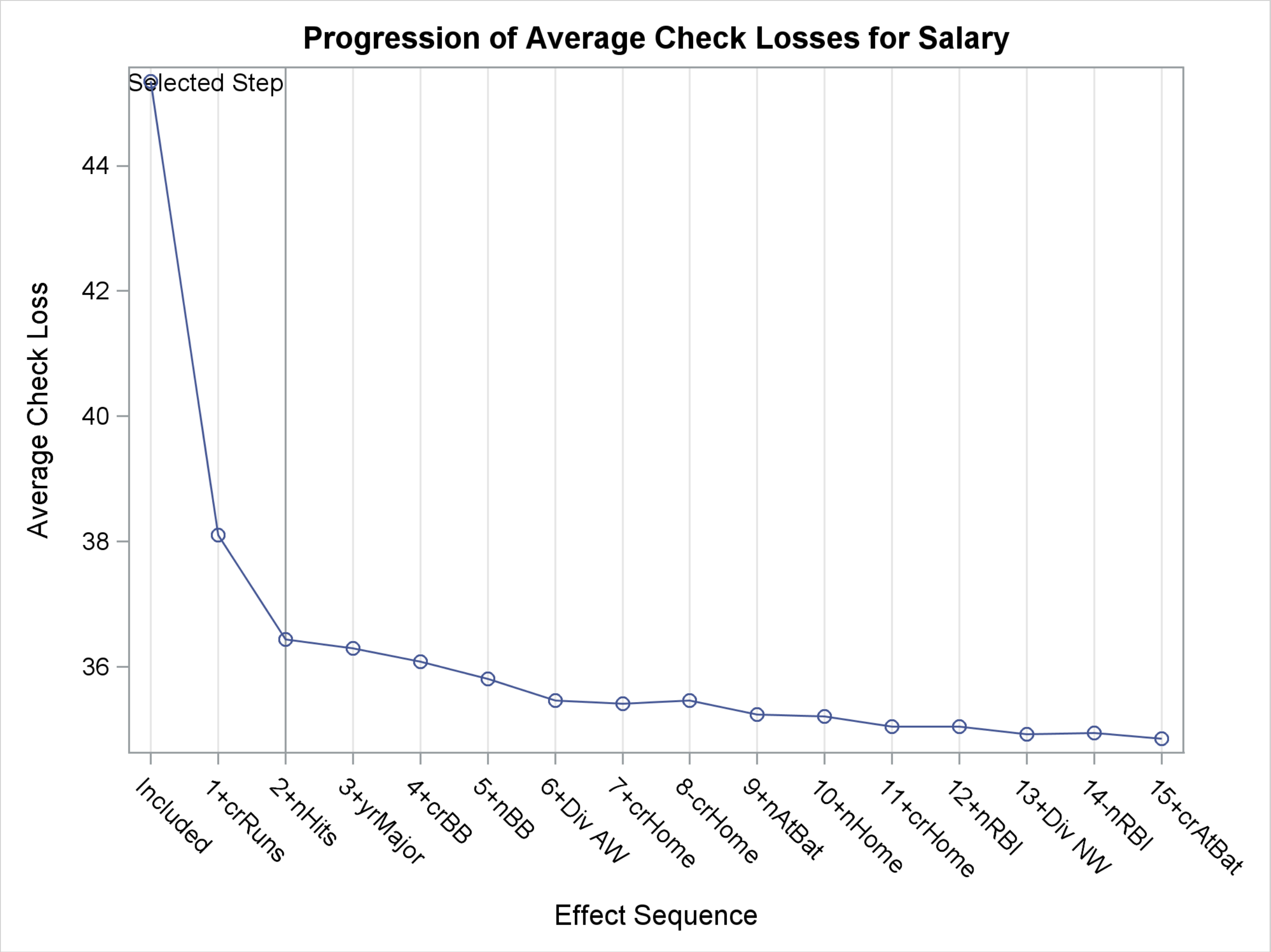
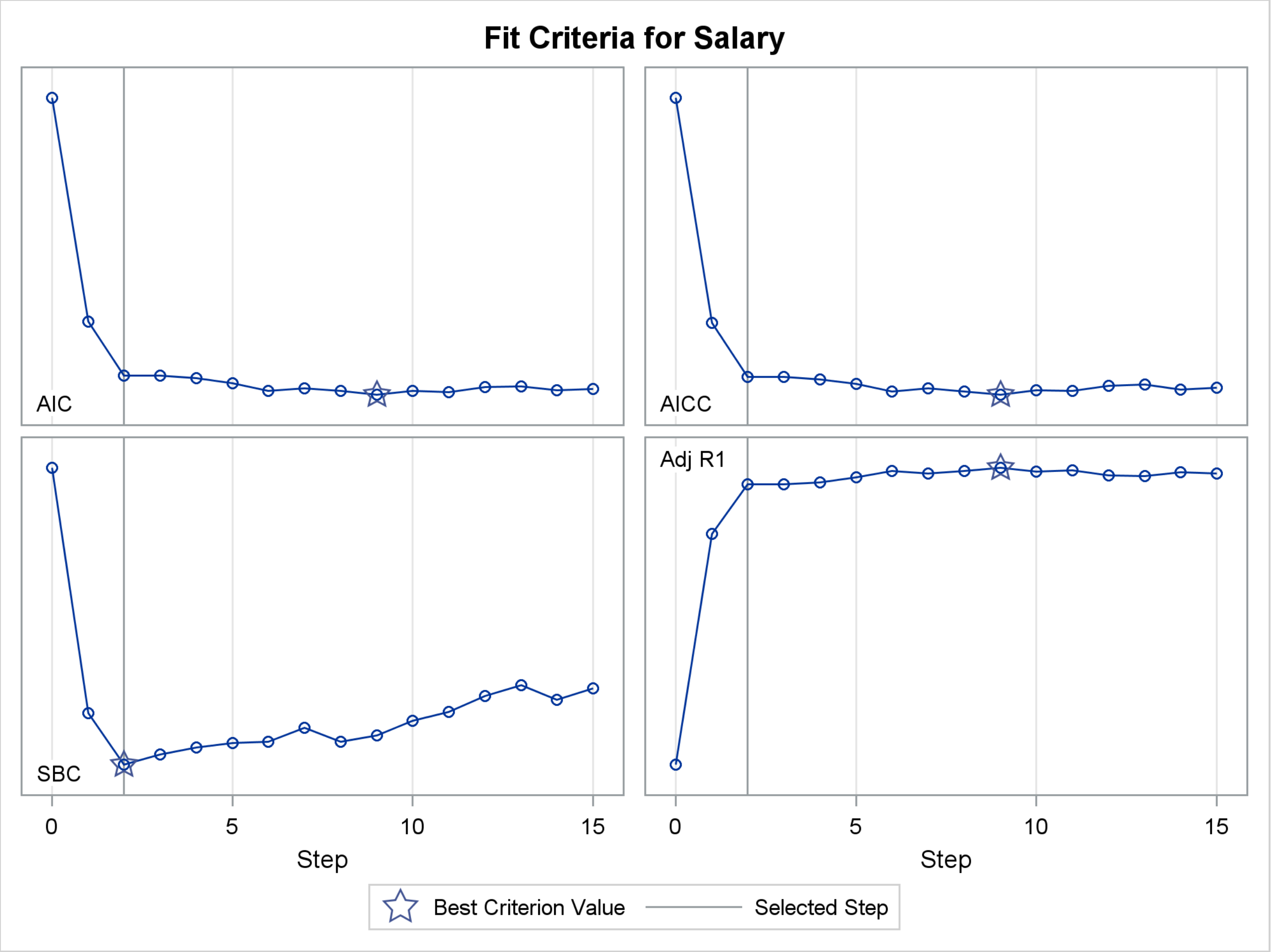


Figure 78.13 shows the progression of four effect selection criteria as the selection process proceeds.

Figure 78.13 Criterion Panel: $\tau = 0.1$



Syntax: QUANTSELECT Procedure

The following statements are available in PROC QUANTSELECT:

```
PROC QUANTSELECT < options > ;
    BY variables ;
    CLASS variable < (v-options) > < variable < (v-options . . . ) > > < / v-options > < options > ;
    EFFECT name = effect-type (variables < / options > ) ;
    MODEL variable = < effects > < / options > ;
    OUTPUT < OUT=SAS-data-set > < keyword < =name > > < . . . keyword< =name > > ;
    PARTITION < options > ;
    WEIGHT variable ;
```

The PROC QUANTSELECT statement invokes the procedure. All statements other than the MODEL statement are optional. CLASS and EFFECT statements, if present, must precede the MODEL statement.

PROC QUANTSELECT Statement

PROC QUANTSELECT < *options* > **;**

Table 78.1 lists the *options* available in the PROC QUANTSELECT statement.

Table 78.1 PROC QUANTSELECT Statement Options

option	**Description**
Data Set Options	
DATA=	Names a data set to use for the regression
MAXMACRO=	Sets the maximum number of macro variables to produce
TESTDATA=	Names a data set that contains test data
VALDATA=	Names a data set that contains validation data
ODS Graphics Options	
PLOTS=	Produces ODS Graphics displays
Other Options	
OUTDESIGN=	Names a data set that contains the design matrix
NOPRINT	Suppresses displayed output (including plots)
SEED=	Sets the seed used for pseudorandom number generation

You can specify the following *options* (shown in alphabetical order) in the PROC QUANTSELECT statement.

DATA=*SAS-data-set*

names the SAS data set to be used by PROC QUANTSELECT. If the DATA= option is not specified, PROC QUANTSELECT uses the most recently created SAS data set. If the data set contains a variable named _ROLE_, then this variable is used to assign observations for training, validation, and testing roles. See the section "Using Validation and Test Data" on page 6602 for more information about using the _ROLE_ variable.

MAXMACRO=*n*

specifies the maximum number of macro variables with selected effects to create. By default, MAXMACRO=100.

PROC QUANTSELECT saves the list of selected effects in a macro variable, &_QRSIND. For example, suppose your input effect list consists of x1–x10. Then &_QRSIND would be set to x1 x3 x4 x10 if the first, third, fourth, and tenth effects were selected for the model. This list can be used in the MODEL statement of a subsequent procedure.

If you specify the OUTDESIGN= option in the PROC QUANTSELECT statement, then PROC QUANTSELECT saves the list of columns in the design matrix in a macro variable named &_QRSMOD.

With multiple quantile levels and BY processing, one macro variable is created for each combination of quantile level and BY group, and the macro variables are indexed by the BY-group number and the quantile level index. You can use the MAXMACRO= option to either limit or increase the number of these macro variables when you are processing data sets with many combinations of quantile level and BY group.

With a single quantile level and no BY-group processing, PROC QUANTSELECT creates the macro variables shown in Table 78.2.

Table 78.2 Macro Variables Created for a Single Quantile Level and No BY Processing

Macro Variable Name	Contains
_QRSIND	Selected effects
_QRSIND1	Selected effects
_QRSINDT1	Selected effects
_QRSIND1T1	Selected effects
_QRSMOD	Selected design matrix columns
_QRSMOD1	Selected design matrix columns
_QRSMODT1	Selected design matrix columns
_QRSMOD1T1	Selected design matrix columns

With multiple quantile levels and BY-group processing, PROC QUANTSELECT creates the macro variables shown in Table 78.3.

Table 78.3 Macro Variables Created for a Multiple Quantile Levels and BY-Group Processing

Macro Variable Name	Contains
_QRSIND	Selected effects for quantile 1 and BY group 1
_QRSINDT1	Selected effects for quantile 1 and BY group 1
_QRSINDT2	Selected effects for quantile 2 and BY group 1
.	
.	
.	
_QRSIND1	Selected effects for quantile 1 and BY group 1
_QRSIND1T1	Selected effects for quantile 1 and BY group 1
_QRSIND1T2	Selected effects for quantile 2 and BY group 1
.	
.	
.	
_QRSIND2	Selected effects for quantile 1 and BY group 2
_QRSIND2T1	Selected effects for quantile 1 and BY group 2
_QRSIND2T2	Selected effects for quantile 2 and BY group 2
.	
.	
.	
_QRSIND*m*T*n*	Selected effects for quantile *n* and BY group *m*

If you specify the OUTDESIGN= option, PROC QUANTSELECT also creates the macro variables shown in Table 78.4.

Table 78.4 Macro Variables Created When the OUTDESIGN= Option Is Specified

Macro Variable Name	Contains
_QRSMOD	Selected design matrix columns for BY group 1
_QRSMOD1	Selected design matrix columns for BY group 1
_QRSMOD2	Selected design matrix columns for BY group 2
.	
.	
.	
_QRSMOD*m*T*n*	Selected design matrix columns for quantile *n* and BY group *m*

The macros variables in Table 78.5 show the number of quantiles and BY groups:

Table 78.5 Macro Variables Showing the Number of Quantiles and BY Groups

Macro Variable Name	**Contains**
_QRSNUMBYS	The number of BY groups
_QRSNUMTAUS	The number of quantiles
_QRSBY1NUMTAUS	The number of _QRS1IND*j* macro variables actually made
_QRSBY2NUMTAUS	The number of _QRS2IND*j* macro variables actually made
.	
.	
.	
_QRSNUMBYTAUS	The number of _QRS*i*IND*j* macro variables actually made. This value can be less than _QRSNUMBYS × _QRSNUMTAUS, and it is less than or equal to MAXMACRO=*n*.

See the section "Macro Variables That Contain Selected Models" on page 6600 for more information.

NOPRINT

suppresses all displayed output (including plots).

OUTDESIGN< (*options*)= >< *SAS-data-set* >

creates a data set that contains the design matrix. By default, the QUANTSELECT procedure includes in the OUTDESIGN data set the **X** matrix that corresponds to all the effects in the selected models. Two schemes for naming the columns of the design matrix are available:

- In the first scheme, names of the parameters are constructed from the parameter labels that appear in the parameter estimates table. This naming scheme is the default when you do not request BY processing; it is not available when you do use BY processing.
- In the second scheme, the design matrix column names consist of a prefix followed by an index. The default name prefix is _X. This scheme is used when you specify the PREFIX= option or a BY statement; otherwise the first scheme is used.

You can specify the following *options* in parentheses to control the contents of the OUTDESIGN= data set:

ADDINPUTVARS

includes all the input data set variables in the OUTDESIGN= data set.

ADDVALDATA

includes all the VALDATA= data set variables in the OUTDESIGN= data set. This option is ignored if the VALDATA= data set is not specified.

ADDTESTDATA

includes all the TESTDATA= data set variables in the OUTDESIGN= data set. This option is ignored if TESTDATA= data set is not specified.

FULLMODEL

includes in the OUTDESIGN= data set parameters that correspond to all effects that are specified in the MODEL statement. By default, only parameters that correspond to the selected model are included.

NAMES

produces a table that associates columns in the OUTDESIGN= data set with the labels of the parameters they represent.

PREFIX< =*prefix* >

creates the design matrix column names from a prefix followed by an index. The default *prefix* is _X.

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

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

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

```
plots=all
plots=coefficients(unpack)
plots(unpack)=(coef acl crit)
```

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

```
ods graphics on;
proc quantselect plots=all;
   class temp sex / split;
   model depVar = sex sex*temp;
run;
```

For more information about enabling and disabling ODS Graphics, see the section "Enabling and Disabling ODS Graphics" on page 600 in Chapter 21, "Statistical Graphics Using ODS."

You can specify the following *global-plot-options*, which apply to all plots generated by the QUANTSELECT procedure, unless they are altered by specific plot *options*.

ENDSTEP=*n*

specifies that the step ranges shown on the horizontal axes of plots terminate at the specified step. By default, the step range shown terminates at the final step of the selection process. If you specify the ENDSTEP= option as both a *global-plot-option* and as an *option* for a specific *plot-request*, then PROC QUANTSELECT uses the ENDSTEP=*n option* for the specific *plot-request*.

LOGP | LOGPVALUE

displays the natural logarithm of the entry and removal significance levels when the SELECT=SL option is specified in the MODEL statement.

MAXSTEPLABEL=*n*

specifies the maximum number of characters beyond which labels of effects on plots are truncated. The default is MAXSTEPLABEL=256.

MAXPARMLABEL=*n*

specifies the maximum number of characters beyond which parameter labels on plots are truncated. The default is MAXPARMLABEL=256.

STARTSTEP=*n*

specifies that the step ranges shown on the horizontal axes of plots start at the specified step. By default, the step range shown starts at the initial step of the selection process. If you specify the STATSTEP= option as both a *global-plot-option* and as an *option* for a specific *plot-request*, then PROC QUANTSELECT uses the STARTSTEP=*n option* for the specific *plot-request*. The default is STARTSTEP=0.

STEPAXIS=EFFECT | NORMB | NUMBER

specifies the method for labeling the horizontal plot axis. This axis represents the sequence of entering or departing effects. The default is STEPAXIS=EFFECT.

STEPAXIS=EFFECT

labels each step by a prefix followed by the name of the effect that enters or leaves at that step. The prefix consists of the step number followed by a "+" sign or a "–" sign, depending on whether the effect enters or leaves at that step.

STEPAXIS=NORMB

labels the horizontal axis value at step *i* with the penalty on the parameter estimates at step *i*, normalized by the penalty on the parameter estimates at the final step. This option is valid only with regularization selection methods.

STEPAXIS=NUMBER

labels each step with the step number.

UNPACK

displays each graph separately. (By default, some graphs can appear together in a single panel.) You can also specify UNPACK as a suboption with CRITERIA and COEFFICIENTS options for specific *plot-requests*.

The following list describes the specific *plot-requests* and their *options*.

ALL

displays all appropriate graphs.

ACL | ACLPLOT < (*aclplot-option*) >

plots the progression of the average check losses on the training data, and on the test and validation data when these data are provided with the TESTDATA= or VALDATA= options or are produced by using a PARTITION statement. When the PROC QUANTSELECT procedure is applied on multiple quantile levels, the ACL option and its suboptions apply to the ACL plots for each of the quantile levels.

You can specify the following *aclplot-option*:

STEPAXIS=EFFECT | NORMB | NUMBER

specifies the method for labeling the horizontal plot axis. See the STEPAXIS= option in the *global-plot-options* for more information.

COEF | COEFFICIENTS | COEFFICIENTPANEL < (*coefficient-panel-options*) >

displays a panel of two plots for each quantile level. The upper plot shows the progression of the parameter values as the selection process proceeds. The lower plot shows the progression of the CHOOSE= criterion. If no CHOOSE= criterion is in effect, then the AICC criterion is displayed. You can specify the following *coefficient-panel-options*:

LABELGAP=*percentage*

specifies the percentage of the vertical axis range that forms the minimum gap between successive parameter labels at the final step of the coefficient progression plot. If the values of more than one parameter at the final step are closer than this gap, then the labels on all but one of these parameters are suppressed. The default is LABELGAP=5.

LOGP | LOGPVALUE

displays the natural logarithm of the entry and removal significance levels when the SELECT=SL option is specified in the MODEL statement.

STEPAXIS=EFFECT | NORMB | NUMBER

specifies the horizontal axis to be used. See the STEPAXIS= option in the *global-options* for more information.

UNPACK | UNPACKPANEL

displays the coefficient progression and the CHOOSE= criterion progression in separate plots.

CRIT | CRITERIA | CRITERIONPANEL < (*criterion-panel-options*) >

plots a panel of model fit criteria. If multiple quantile levels apply, the CRITERIA option plots a panel of model fit criteria for each quantile level. The criteria that are displayed are AIC, AICC, and SBC, in addition to any other criteria that are named in the CHOOSE=, SELECT=, STOP=, and STATS= options in the MODEL statement. You can specify the following *criterion-panel-options*:

STEPAXIS=EFFECT | NORMB | NUMBER

specifies the horizontal axis to be used. See the STEPAXIS= option in the *global-options* for more information.

UNPACK | UNPACKPANEL

displays each criterion progression on a separate plot.

NONE

suppresses all plots.

SEED=*number*

specifies an integer that is used to start the pseudorandom number generator for random partitioning of data for training, testing, and validation. If you do not specify a seed or if you specify a value less than or equal to 0, the seed is generated by reading the time of day from the computer's clock.

TESTDATA=*SAS-data-set*

names a SAS data set that contains test data. This data set must contain all the effects that are specified in the MODEL statement. Furthermore, when you also specify a BY statement and the TESTDATA= data set contains any of the BY variables, then the TESTDATA= data set must also contain all the

BY variables sorted in the order of the BY variables. In this case, only the test data for a specific BY group are used with the corresponding BY group in the analysis data. If the TESTDATA= data set contains none of the BY variables, then the entire TESTDATA= data set is used with each BY group of the analysis data.

If you specify both a TESTDATA= data set and the PARTITION statement, then the testing observations from the DATA= data set are merged with the TESTDATA= data set for testing purposes.

VALDATA=*SAS-data-set*

names a SAS data set that contains validation data. This data set must contain all the effects that are specified in the MODEL statement. Furthermore, when a BY statement is used and the VALDATA= data set contains any of the BY variables, then the VALDATA= data set must also contain all the BY variables sorted in the order of the BY variables. In this case, only the validation data for a specific BY group are used with the corresponding BY group in the analysis data. If the VALDATA= data set contains none of the BY variables, then the entire VALDATA= data set is used with each BY group of the analysis data.

If you specify both a VALDATA= data set and the PARTITION statement, then the validation observations from the DATA= data set are merged with the VALDATA= data set for validation purposes.

BY Statement

BY *variables* **;**

You can specify a BY statement with PROC QUANTSELECT to obtain separate analyses of observations in groups that are defined by the BY variables. When a BY statement appears, the procedure expects the input data set to be sorted in order of the BY variables. If you specify more than one BY statement, only the last one specified is used.

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

- Sort the data by using the SORT procedure with a similar BY statement.
- Specify the NOTSORTED or DESCENDING option in the BY statement for the QUANTSELECT procedure. The NOTSORTED option does not mean that the data are unsorted but rather that the data are arranged in groups (according to values of the BY variables) and that these groups are not necessarily in alphabetical or increasing numeric order.
- Create an index on the BY variables by using the DATASETS procedure (in Base SAS software).

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

CLASS Statement

CLASS *variable* < **(***v-options***)** > < *variable* < **(***v-options* . . . **)** > > < / *v-options* > < *options* > **;**

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

Table 78.6 summarizes the *options* and *v-options* available in the CLASS statement.

Table 78.6 CLASS Statement Options

option **or** *v-option*	**Description**
DELIMITER=	Specifies the delimiter
DESCENDING	Reverses the sort order
MISSING	Allows for missing values
ORDER=	Specifies the sort order
PARAM=	Specifies the parameterization method
REF=	Specifies the reference level
SHOW	Requests a table for each CLASS variable
SPLIT	Splits CLASS variables into independent effects

You can specify the following *options* after a slash (/):

DELIMITER=*'c'*

specifies the delimiter character, '*c*', to be used between levels of classification variables when parameter names and lists of class level values are built. The default delimiter is a space. This option is useful if the levels of a classification variable contain embedded blanks.

SHOW | SHOWCODING

requests a table that shows the coding used for each classification variable.

You can specify various *v-options* for each variable by enclosing them in parentheses after the variable name; these are called individual *v-options*. You can also specify global *v-options* by placing them after a slash (/) at the end of the CLASS statement. Global *v-options* are applied to all the variables specified in the CLASS statement. If you specify more than one CLASS statement, the global *v-options* specified in any one CLASS statement apply to all CLASS statements. However, individual CLASS variable *v-options* override the global *v-options*.

You can specify the following *v-options*:

DESCENDING

DESC

reverses the sort order of the classification variable.

MISSING

allows missing values, such as `'.'` for a numeric variable or a blank for a character variable, as valid values for the CLASS variable.

ORDER=DATA | FORMATTED | FREQ | INTERNAL

specifies the sort order for the levels of classification variables. If ORDER=FORMATTED for numeric variables for which you have supplied no explicit format, the levels are ordered by their internal values. The following table shows how PROC QUANTSELECT interprets values of the ORDER= option.

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

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

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

PARAM=*keyword*

specifies the parameterization method for the classification variable or variables. Design matrix columns are created from CLASS variables according to the following coding schemes. The default is PARAM=GLM. If PARAM=ORTHPOLY or PARAM=POLY, and the CLASS levels are numeric, then the ORDER= option in the CLASS statement is ignored, and the internal, unformatted values are used. See the section "CLASS Variable Parameterization and the SPLIT Option" on page 3570 in Chapter 45, "The GLMSELECT Procedure," for more information.

EFFECT	specifies effect coding.
GLM	specifies less-than-full-rank coding. This option can be used only as a global *v-option* (after the slash in the CLASS statement).
ORDINAL | THERMOMETER	specifies the cumulative parameterization for an ordinal CLASS variable.
POLYNOMIAL | POLY	specifies polynomial coding.
REFERENCE | REF	specifies reference-cell coding.
ORTHEFFECT	orthogonalizes PARAM=EFFECT.
ORTHORDINAL | ORTHOTHERM	orthogonalizes PARAM=ORDINAL.
ORTHPOLY	orthogonalizes PARAM=POLYNOMIAL.
ORTHREF	orthogonalizes PARAM=REFERENCE.

The EFFECT, POLYNOMIAL, REFERENCE, and ORDINAL coding schemes and their orthogonal parameterizations are full rank. The REF= option in the CLASS statement determines the reference level for the EFFECT and REFERENCE schemes and their orthogonal parameterizations.

REF=*'level'* | FIRST | LAST

specifies the reference level for PARAM=EFFECT, PARAM=REFERENCE, and their orthogonalizations. For an individual (but not a global) REF= *v-option*, you can specify the *level* of the variable to use as the reference level. For a global or individual REF= *v-option*, you can specify REF=FIRST (which designates the first-ordered level as reference) or REF=LAST (which designates the last-ordered level as reference). The default is REF=LAST.

SPLIT

enables the columns of the design matrix that correspond to any effect that contains a split classification variable to be selected to enter or leave a model independently of the other design columns of that effect. For example, suppose a variable named temp has three levels with values `'hot'`, `'warm'`, and `'cold'`, and a variable named sex has two levels with values `'M'` and `'F'`. The following statements include SPLIT as a global *v-option*:

```
proc quantselect;
   class temp sex / split;
   model depVar = sex sex*temp;
run;
```

Because both the classification variables are split, the two effects named in the MODEL statement are split into eight effects. The effect `'sex'` is split into two effects labeled `'sex_M'` and `'sex_F'`. The effect `'sex*temp'` is split into six effects labeled `'sex_M*temp_hot'`, `'sex_F*temp_hot'`, `'sex_M*temp_warm'`, `'sex_F*temp_warm'`, `'sex_M*temp_cold'`, and `'sex_F*temp_cold'`. The previous PROC QUANTSELECT statements are equivalent to the following statements for the split version of the DATA= data set:

```
proc quantselect;
   model depVar =  sex_M sex_F sex_M*temp_hot  sex_F*temp_hot
                               sex_M*temp_warm sex_F*temp_warm
                               sex_M*temp_cold sex_F*temp_cold;
run;
```

You can specify the SPLIT option for individual classification variables. For example, consider the following PROC QUANTSELECT statements:

```
proc quantselect;
   class temp(split) sex;
   model depVar = sex sex*temp;
run;
```

In this case, the effect `'sex'` is not split, and the effect `'sex*temp'` is split into three effects labeled `'sex*temp_hot'`, `'sex*temp_warm'`, and `'sex*temp_cold'`. Furthermore each of these three split effects now has two parameters that correspond to the two levels of `'sex,'` and the previous PROC QUANTSELECT statements are equivalent to the following statements for the split version of the DATA= data set:

```
proc quantselect;
   class sex;
   model depVar = sex sex*temp_hot sex*temp_warm sex*temp_cold;
run;
```

EFFECT Statement

EFFECT *name*=*effect-type* **(***variables* < / *options* >**)** ;

The EFFECT statement enables you to construct special collections of columns for design matrices. These collections are referred to as *constructed effects* to distinguish them from the usual model effects that are formed from continuous or classification variables, as discussed in the section "GLM Parameterization of Classification Variables and Effects" on page 383 in Chapter 19, "Shared Concepts and Topics."

You can specify the following *effect-types*:

COLLECTION
is a collection effect that defines one or more variables as a single effect with multiple degrees of freedom. The variables in a collection are considered as a unit for estimation and inference.

LAG
is a classification effect in which the level that is used for a given period corresponds to the level in the preceding period.

MULTIMEMBER | MM
is a multimember classification effect whose levels are determined by one or more variables that appear in a CLASS statement.

POLYNOMIAL | POLY
is a multivariate polynomial effect in the specified numeric variables.

SPLINE
is a regression spline effect whose columns are univariate spline expansions of one or more variables. A spline expansion replaces the original variable with an expanded or larger set of new variables.

Table 78.7 summarizes the *options* available in the EFFECT statement.

Table 78.7 EFFECT Statement Options

Option	**Description**
Collection Effects Options	
DETAILS	Displays the constituents of the collection effect
Lag Effects Options	
DESIGNROLE=	Names a variable that controls to which lag design an observation is assigned
DETAILS	Displays the lag design of the lag effect
NLAG=	Specifies the number of periods in the lag
PERIOD=	Names the variable that defines the period
WITHIN=	Names the variable or variables that define the group within which each period is defined
Multimember Effects Options	
NOEFFECT	Specifies that observations with all missing levels for the multimember variables should have zero values in the corresponding design matrix columns
WEIGHT=	Specifies the weight variable for the contributions of each of the classification effects

Table 78.7 *continued*

Option	Description
Polynomial Effects Options	
DEGREE=	Specifies the degree of the polynomial
MDEGREE=	Specifies the maximum degree of any variable in a term of the polynomial
STANDARDIZE=	Specifies centering and scaling suboptions for the variables that define the polynomial
Spline Effects Options	
BASIS=	Specifies the type of basis (B-spline basis or truncated power function basis) for the spline expansion
DEGREE=	Specifies the degree of the spline transformation
KNOTMETHOD=	Specifies how to construct the knots for spline effects

For more information about the syntax of these *effect-types* and how columns of constructed effects are computed, see the section "EFFECT Statement" on page 393 in Chapter 19, "Shared Concepts and Topics."

MODEL Statement

MODEL *dependent* **=** < *effects* > / < *options* > **;**

The MODEL statement names the dependent variable and the covariate effects, including covariates, main effects, constructed effects, interactions, and nested effects; see the section "Specification of Effects" on page 3324 in Chapter 42, "The GLM Procedure," for more information. If you omit the explanatory effects, PROC QUANTSELECT fits an intercept-only model.

After the keyword MODEL, specify the dependent (response) variable, followed by an equal sign, followed by the explanatory effects.

Table 78.8 summarizes the *options* available in the MODEL statement.

Table 78.8 MODEL Statement Options

Option	Description
DETAILS=	Specifies the level of effect selection detail to display
HIERARCHY=	Specifies hierarchy of effects to impose
NOINT	Specifies models without an explicit intercept
QUANTILES=	Specifies quantile levels to be applied
SELECTION=	Specifies effect selection method
STATS=	Specifies additional statistics to be displayed
TEST=	Specifies the test type for computing significance levels

The following list provides details about the *options* that you can specify in the MODEL statement after a slash (/):

DETAILS=ALL | SUMMARY

specifies the level of effect selection detail that is displayed. The DETAILS=SUMMARY option produces only the selection summary table. The DETAILS=ALL option produces the following:

- entry and removal statistics for each variable that is selected in the model building process
- average check loss and parameter estimates
- entry and removal statistics for the top five candidates for inclusion or exclusion at each step
- a selection summary table

The default is DETAILS=SUMMARY.

HIERARCHY=*keyword*

HIER=*keyword*

specifies whether and how the model hierarchy requirement is applied. This option also controls whether a single effect or multiple effects are allowed to enter or leave the model in one step. You can specify that only CLASS effects, or both CLASS and continuous effects, be subject to the hierarchy requirement. This option is ignored unless you also specify one of the following options: SELECTION=FORWARD, SELECTION=BACKWARD, or SELECTION=STEPWISE.

Model hierarchy refers to the requirement that for any term to be in the model, all model effects contained in the term must be present in the model. For example, in order for the interaction A*B to enter the model, the main effects A and B must be in the model. Likewise, neither effect A nor effect B can leave the model while the interaction A*B is in the model.

You can specify the following *keywords*:

NONE

specifies that model hierarchy not be maintained. Any single effect can enter or leave the model at any given step of the selection process.

SINGLE

specifies that only one effect enter or leave the model at one time, subject to the model hierarchy requirement. For example, suppose that the model contains the main effects A and B and the interaction A*B. In the first step of the selection process, either A or B can enter the model. In the second step, the other main effect can enter the model. The interaction effect can enter the model only when both main effects have already entered. Also, before A or B can be removed from the model, the A*B interaction must first be removed. All effects (CLASS and interval) are subject to the hierarchy requirement.

SINGLECLASS

is the same as HIERARCHY=SINGLE except that only CLASS effects are subject to the hierarchy requirement.

The default is HIERARCHY=NONE.

NOINT

suppresses the intercept term that is otherwise included in the model.

QUANTILES=< *number-list* **| PROCESS < (***option***) > >**

QUANTILE=< *number-list* **| PROCESS < (***option***) > >**

specifies the quantile levels for the quantile regression. You can specify any number of quantile levels in $(0, 1)$. If you do not specify this option, the QUANTSELECT procedure performs median regression effect selection that corresponds to QUANTILES=0.5.

If you specify the QUANTILES=PROCESS option, the QUANTSELECT procedure performs effect selection for quantile process regression. The QUANTILES=PROCESS option cannot be used with LASSO selection methods. You can specify the following *option* in parentheses after QUANTILES=PROCESS.

NTAU=*n* **| ALL**

specifies how many quantile levels that you expect to cover for the quantile process. If you specify NTAU=ALL, the QUANTSELECT procedure performs effect selection for accurate quantile process regression. If you specify NTAU=*n*, the QUANTSELECT procedure performs effect selection for approximate quantile process regression. The approximate quantile process is computed at *n* equally spaced quantile levels: $\{\frac{1}{n+1}, \cdots, \frac{n}{n+1}\}$.

SELECTION=*method* **< (***method-options***) >**

specifies the *method* used to select the model, optionally followed by parentheses that enclose *method-options* that apply to the specified method. The default is SELECTION=STEPWISE.

You can specify the following *methods*, which are explained in detail in the section "Effect Selection Methods" on page 6597.

NONE specifies full model fitting without effect selection.

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.

LASSO specifies a method that adds and deletes parameters based on a version of estimated check risk where the weighted L1-norm of certain weighted regression coefficients is penalized. For more information, see the section "LASSO Method (LASSO)" on page 6598. If the model contains CLASS variables or constructed effects, these CLASS variables or constructed effects are split into separate covariates.

Table 78.9 lists the applicable *method-options* for each *method.*

Table 78.9 Applicable *method-options* for Each *method*

method-option	**FORWARD**	**BACKWARD**	**STEPWISE**	**LASSO**
ADAPTIVE				x
CHOOSE=	x	x	x	x
INCLUDE=	x	x	x	x
MAXSTEP=	x	x	x	x
SELECT=	x	x	x	
SLENTRY=	x		x	
SLSTAY=		x	x	
STOP=	x	x	x	x
STOPHORIZON=	x	x	x	x

You can specify the following *method-option* in parentheses after the *method*. As described in Table 78.9, not all *method-options* apply to every SELECTION= method.

ADAPTIVE

ADAPT

specifies the adaptive LASSO selection method. The ADAPTIVE option can be used only with the SELECTION=LASSO option.

CHOOSE=*criterion*

chooses from the list of models (with one model at each step of the selection process) the model that yields the best value of the specified *criterion* as the final selected model. If the optimal value of the specified *criterion* occurs for more than one model, then the model with the smallest number of parameters is chosen. If you do not specify the CHOOSE= option, then the model selected is the model at the final step in the selection process for the SELECT=SL criterion, or the STOP= option is applied as the CHOOSE= option for all the other cases.

You can specify the following values for *criterion* in the CHOOSE= option. See the section "Criteria Used in Model Selection Methods" on page 6598 for more information about these criteria.

ADJR1	chooses the model with the largest adjusted quantile regression R statistic.
AIC	chooses the model with the smallest Akaike's information criterion.
AICC	chooses the model with the smallest corrected Akaike's information criterion.
SBC	chooses the model with the smallest Schwarz Bayesian information criterion.
VALIDATE	chooses the model with the smallest average check loss for the validation data. You can specify CHOOSE=VALIDATE only if you have specified a VALDATA= data set in the PROC QUANTSELECT statement or if you have reserved part of the input data for validation by using either a PARTITION statement or a _ROLE_ variable in the input data.

INCLUDE=*n*

forces the first *n* effects listed in the MODEL statement to be included in all models. The selection methods are performed on the other effects in the MODEL statement. The INCLUDE= option is available only with SELECTION=FORWARD, SELECTION=STEPWISE, and SELECTION=BACKWARD.

MAXSTEP=*n*

specifies the maximum number of selection steps. The default value of *n* is the number of effects in the MODEL statement when SELECTION=FORWARD or SELECTION=BACKWARD and is three times the number of effects when SELECTION=STEPWISE or SELECTION=LASSO.

SELECT=*criterion*

specifies the criterion that PROC QUANTSELECT uses to determine the order in which effects enter or leave at each step of the specified selection method. This option is not valid when SELECTION=LASSO. You can specify the following values for *criterion*: ADJR1, AIC, AICC, SBC, SL, and VALIDATE. See the section "Criteria Used in Model Selection Methods" on page 6598 for more information about these criteria.

When SELECT=SL, the effect selection depends on the selection method and is described in the relevant subsection of the section "Effect Selection Methods" on page 6597. Otherwise, the effect that is selected to enter or leave at a step of the selection process is the effect whose addition to or removal from the current model produces the maximum improvement in the specified criterion.

If validation data exist, the default is SELECT=VALIDATE; otherwise, the default is SELECT=SBC.

SLENTRY=*value*

SLE=*value*

specifies the significance level for entry, used when the STOP=SL or SELECT=SL option is in effect. The defaults are 0.50 when SELECTION=FORWARD and 0.15 when SELECTION=STEPWISE.

SLSTAY=*value*

SLS=*value*

specifies the significance level for staying in the model, used when the STOP=SL or SELECT=SL option is in effect. The defaults are 0.10 when SELECTION=BACKWARD and 0.15 when SELECTION=STEPWISE.

STOP=*criterion*

specifies the *criterion* for stopping the selection process. If the maximum number of steps is specified in the MAXSTEP= option and the *criterion* does not stop the selection process before the maximum number of steps for the selection method, then the selection process terminates at the maximum number of steps.

You can specify the following values for *criterion*. See the section "Criteria Used in Model Selection Methods" on page 6598 for more detailed descriptions of these criteria.

NONE enables the model selection process to go through all possible steps.

ADJR1 stops selection at the step where the next SH= steps (or all remaining steps) would yield models with smaller values of the adjusted quantile regression R (ADJR1) statistic.

AIC stops selection at the step where the next SH= steps (or all remaining steps) would yield models with larger values of Akaike's information criterion.

AICC stops selection at the step where the next SH= steps (or all remaining steps) would yield models with larger values of the corrected Akaike's information criterion.

SBC stops selection at the step where the next SH= steps (or all remaining steps) would yield models with larger values of the Schwarz Bayesian information criterion.

VALIDATE stops selection at the step where the next SH= steps (or all remaining steps) would yield models with larger values of the average check loss for the validation data. You can specify STOP=VALIDATE only if you have specified a VALDATA= data set in the PROC QUANTSELECT statement or if you have reserved part of the input data for validation by using either a PARTITION statement or a _ROLE_ variable in the input data.

The default value of this option depends on other options as follows:

- If you specify SELECT=SL and you also specify the SLE= *method-option* or the SLS= *method-option*, the STOP option is ignored.
- If do not specify both SELECT=SL and one or both of the SLE= or SLS= *method-options* and validation data exist, the default is STOP=VALIDATE.
- If do not specify both SELECT=SL and one or both of the SLE= or SLS= *method-options* and validation data do not exist, the default is one of the following:
  - When you specify SELECTION=LASSO, the default is STOP=SBC.
  - When you specify SELECTION=NONE, FORWARD, BACKWARD, or STEPWISE, the default is the criterion specified in the SELECT= *method-option*.

STOPHORIZON=*n*

SH=*n*

looks ahead for the specified number of steps to decide whether an extremum of the stop criterion is achieved. This option applies only to the STOP= criterion. The default is STOPHORIZON=1.

For example, suppose that the stop criterion values at steps 1 through 5 are 4, 3, 5, 6, and 2, respectively. If you specify STOPHORIZON=1, then the selection process terminates after looking at the model at step 3, and the final selected model is the model at step 2. If you specify STOPHORIZON=2, the selection process stops after looking at the model at step 4, and the final selected model is the model at step 2. However, if you specify STOPHORIZON=3 or higher, then the local minimum in the stop value sequence at step 2 cannot stop the selection process because a lower value is achieved at step 5, which is within 3 steps beyond this local minimum step.

STAT=*name* | (*names*)

STATS=*name* | (*names*)

specifies which model fit statistics to display in the selection summary table. To specify multiple model fit statistics, specify a list of *names* in parentheses. If you omit this option, the default set of statistics that are displayed in these tables includes all the criteria that are specified in any of the CHOOSE=, SELECT=, and STOP= *method-options*.

You can specify the following values for *name*:

ADJR1 displays the adjusted quantile regression R statistic.

AIC displays the Akaike's information criterion.

AICC displays the corrected Akaike's information criterion.

ACL displays the average check losses for the training, test, and validation data. The ACL statistics for the test and validation data are reported only if you have specified the TESTDATA= option or the VALDATA= option in the PROC QUANTSELECT statement or if you have reserved part of the input data for testing or validation by using either a PARTITION statement or a _ROLE_ variable in the input data.

R1 displays the quantile regression R statistic.

SBC displays the Schwarz Bayesian information criterion.

The statistics ADJR1, AIC, AICC, and SBC can be computed with little computation cost. However, computing ACL for test and validation data when these are not used in any of the CHOOSE=, SELECT=, and STOP= *method-options* can hurt performance.

TEST=*name*

specifies the test type for computing significance levels.

You can specify the following values for *name*:

LR1 specifies the likelihood ratio test Type I. The LR1 test score is defined as

$$\frac{2(D_1(\tau) - D_2(\tau))}{\tau(1-\tau)\hat{s}}$$

where $D_1(\tau) = \sum \rho_\tau \left(y_i - \mathbf{x}_i \hat{\boldsymbol{\beta}}_1(\tau) \right)$ is the sum of check losses for the reduced model, $D_2(\tau) = \sum \rho_\tau \left(y_i - \mathbf{x}_i \hat{\boldsymbol{\beta}}(\tau) \right)$ is the sum of check losses for the extended model, and $\hat{s}$ is the estimated sparsity function. For more information about the sparsity function, see the section "Sparsity" on page 6523 of Chapter 77, "The QUANTREG Procedure."

LR2 specifies the likelihood ratio test Type II. The LR2 test score is defined as

$$\frac{2D_2(\tau)\left(\log(D_1(\tau)) - \log(D_2(\tau))\right)}{\tau(1-\tau)\hat{s}}$$

where $\hat{s}$ is computed on the reduced model for testing the entry effect and on the extended model for testing the removal effect.

OUTPUT Statement

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

The OUTPUT statement creates a new SAS data set that saves diagnostic measures that are calculated for the selected model. If you do not specify a *keyword*, then the only diagnostic included is the predicted response.

All the variables in the original data set are included in the new data set, along with variables that are created by the *keyword* options in the OUTPUT statement. These new variables contain the values of a variety of statistics and diagnostic measures that are calculated for each observation in the data set.

The OUTPUT data set is created in row-wise form, and the variable _QUANTILE_ is optional. For each appropriate keyword specified in the OUTPUT statement, one variable for each specified quantile level is generated. These variables appear in the sorted order of the specified quantile levels.

If you specify a BY statement, then a variable _BY_ that indexes the BY groups is included. For each observation, the value of _BY_ is the index of the BY group to which this observation belongs. This variable is useful for matching BY groups with macro variables that PROC QUANTSELECT creates. See the section "Macro Variables That Contain Selected Models" on page 6600 for more information.

If you have partitioned the input data with a PARTITION statement, then a character variable _ROLE_ is included in the output data set. The following table shows the value of _ROLE_ for each observation:

ROLE Value	**Observation Role**
TEST	Testing
TRAIN	Training
VALIDATE	Validation

If you want to create a permanent SAS data set, you must specify a two-level name. For more information about permanent SAS data sets, see see the discussion in *SAS Language Reference: Concepts*.

You can specify the following arguments in the OUTPUT statement:

keyword < **=***name*>

specifies the statistics to include in the output data set and optionally names the new variables that contain the statistics. Specify one of the following *keywords* for each desired statistic, followed optionally by an equal sign, and the *name* of a variable to contain the statistic. If you specify *keyword=name*, the new variable that contains the requested statistic has the specified name. If you omit the optional *=name* after a *keyword*, then the new variable name is formed by using a prefix of one or more characters that identify the statistic, followed by an underscore (_), followed by the dependent variable name.

PREDICTED | PRED | P includes predicted values in the output data set. The prefix for the default name is p.

RESIDUAL | RESID | R includes residuals, calculated as ACTUAL – PREDICTED, in the output data set. The prefix for the default name is r.

OUT=*SAS-data-set*

names the output data set. By default, PROC QUANTSELECT uses the DATA*n* convention to name the new data set.

PARTITION Statement

PARTITION < *option* > **;**

The PARTITION statement specifies how observations in the input data set are logically partitioned into disjoint subsets for model training, validation, and testing. Either you can designate a variable in the input data set and a set of formatted values of that variable to determine the role of each observation, or you can specify proportions to use for random assignment of observations for each role.

An alternative to using a PARTITION statement is to provide a variable named _ROLE_ in the input data set to define roles of observations in the input data. If you specify a PARTITION statement, then any _ROLE_ variable in the input data set is ignored. If you do not use a PARTITION statement and the input data do not contain a variable named _ROLE_, then all observations in the input data set are assigned to model training.

You can specify either (but not both) of the following *options*:

ROLEVAR=*variable* **(< TEST=***value* **> < TRAIN=***value* **> < VALIDATE=***value* **>)**

ROLE=*variable* **(< TEST=***value* **> < TRAIN=***value* **> < VALIDATE=***value* **>)**

names the variable in the input data set whose values are used to assign roles to each observation. The TEST=, TRAIN=, and VALIDATE= suboptions specify the formatted values of this variable that are used to assign observations roles. If you do not specify the TRAIN= suboption, then all observations whose role is not determined by the TEST= or VALIDATE= suboptions are assigned to training. If you specify a TESTDATA= data set in the PROC QUANTSELECT statement, then you cannot also specify the TEST= suboption in the PARTITION statement. If you specify a VALDATA= data set in the PROC QUANTSELECT statement, then you cannot also specify the VALIDATE= suboption in the PARTITION statement.

FRACTION(< TEST=*fraction* **> < VALIDATE=***fraction* **>)**

requests that specified proportions of the observations in the input data set be randomly assigned training and validation roles. You specify the proportions for testing and validation by using the TEST= and VALIDATE= suboptions. If you specify both the TEST= and the VALIDATE= suboptions, then the sum of the specified fractions must be less than 1 and the remaining fraction of the observations are assigned to the training role. If you specify a TESTDATA= data set in the PROC QUANTSELECT statement, then you cannot also specify the TEST= suboption in the PARTITION statement. If you specify a VALDATA= data set in the PROC QUANTSELECT statement, then you cannot also specify the VALIDATE= suboption in the PARTITION statement.

WEIGHT Statement

WEIGHT *variable* **;**

A WEIGHT statement names a variable in the input data set with values that are relative weights for a weighted quantile regression fit.

Values of the weight variable must be nonnegative. If an observation's weight is 0, the observation is deleted from the analysis. If a weight is negative or missing, it is set to 0, and the observation is excluded from the analysis.

Details: QUANTSELECT Procedure

Effect Selection Methods

The effect selection methods implemented in PROC QUANTSELECT are specified with the SELECTION= option in the MODEL statement.

Full Model Fitted (NONE)

The complete model specified in the MODEL statement is used to fit the model, and no effect selection is done. You request this method by specifying SELECTION=NONE in the MODEL statement.

Forward Selection (FORWARD)

The forward selection technique begins with just the forced-in covariates and then sequentially adds the effect that most improves the fit. The process terminates when no significant improvement can be obtained by adding any effect. You request this method by specifying SELECTION=FORWARD in the MODEL statement.

If you specify the SELECT=SL *method-option*, you can use the TEST= *method-option* to specify a test statistic for gauging improvement in fit. For example, if TEST=LR1, at each step the effect that yields the most significant likelihood ratio statistic is added and the process continues until all effects that are not in the model have LR1 statistics that are not significant at the entry significance level (which is specified in the SLE= option). Because effects can contribute different degrees of freedom to the model, it is necessary to compare the p-values that correspond to these statistics.

Backward Elimination (BACKWARD)

The backward elimination technique starts from the full model, which includes all independent effects. Then effects are deleted one by one until a stopping condition is satisfied. At each step, the effect that shows the smallest contribution to the model is deleted. You request this method by specifying SELECTION=BACKWARD in the MODEL statement.

Suppose you specify the SELECT=SL *method-option* and the TEST=LR1 *method-option* to gauge improvement in quantile regression fit. At any step, the predictor that produces the least significant LR1 statistic is dropped and the process continues until all effects that remain in the model have LR1 statistics that are significant at the stay significance level (which is specified in the SLS= option).

Stepwise Selection (STEPWISE)

The stepwise method is a modification of the forward selection technique in which effects already in the model do not necessarily stay there. You request this method by specifying SELECTION=STEPWISE in the MODEL statement.

In the implementation of the stepwise selection method, the same entry and removal approaches for the forward selection and backward elimination methods are used to assess contributions of effects as they are added to or removed from a model. Suppose you specify SELECT=SL. If, at a step of the stepwise method, any effect in the model is not significant at the level specified by the SLSTAY= *method-option*, then the least significant of these effects is removed from the model and the algorithm proceeds to the next step. This ensures that no effect can be added to a model while some effect currently in the model is not deemed significant. Only after all necessary deletions have been accomplished can another effect be added to the model. In this case, the effect whose addition yields the most significant statistic value is added to the model and the algorithm proceeds to the next step. The stepwise process ends when none of the effects outside the model is significant at the level specified by the SLENTRY= *method-option* and every effect in the model is significant at the level specified by the SLSTAY= *method-option*. In some cases, neither of these two conditions for stopping is met and the sequence of models cycles. In this case, the stepwise method terminates at the end of the second cycle.

Just as with forward selection and backward elimination, you can use the SELECT= *method-option* to change the criterion used to assess effect contributions. You can also use the STOP= *method-option* to specify a stopping criterion and use the CHOOSE= *method-option* to specify a criterion used to select among the sequence of models produced.

LASSO Method (LASSO)

The standard LASSO method uses a standardized design matrix that orthogonalizes selectable covariates against forced-in covariates, and then scales the orthogonalized selectable covariates so that they all have the same sum of squares. See the information about the standard parameter estimate in the section "Parameter Estimates" on page 6606 for more information about design matrix orthogonalization. The LASSO method initializes all the selectable coefficients into 0 at step 0. The predictor that reduces the average check loss the fastest relative to the L1-norm of the selectable coefficient increment is determined, and a step is taken in the direction of this predictor.

The difference between adaptive LASSO and standard LASSO methods is in the prescaling of the selectable coefficients. After orthogonalization against forced-in covariates, the adaptive LASSO method first fits a full model without penalty, and then scales the orthogonalized selectable covariates with the corresponding coefficients from the full model. This adaptive scaling can be equivalently substituted by using a weighted L1-norm penalty, where the weights are the reciprocals of the corresponding coefficients from the full model.

The length of this step determines the coefficient of this predictor and is chosen when some residual changes its sign or some predictor that is not used in the model can reduce the average check loss more efficiently. This process continues until all predictors are in the model.

As with other selection methods, the issue of when to stop the selection process is crucial. You can use the CHOOSE= *method-option* to specify a criterion for choosing among the models at each step. You can also use the STOP= *method-option* to specify a stopping criterion. See the section "Criteria Used in Model Selection Methods" on page 6598 for more information and Table 78.10 for the formulas for evaluating these criteria.

Criteria Used in Model Selection Methods

PROC QUANTSELECT supports a variety of fit statistics that you can specify as criteria for the CHOOSE=, SELECT=, and STOP= *method-options* in the MODEL statement.

Single Quantile Effect Section

The following fit statistics are available for single quantile effect selection:

AIC applies the Akaike's information criterion (Akaike 1981; Darlington 1968; Judge et al. 1985).

AICC applies the corrected Akaike's information criterion (Hurvich and Tsai 1989).

SBC applies the Schwarz Bayesian information criterion (Schwarz 1978; Judge et al. 1985).

SL<(LR1 | LR2)> specifies the significance level of a statistic used to assess an effect's contribution to the fit when it is added to or removed from a model. LR1 specifies likelihood ratio Type I, and LR2 specifies the likelihood ratio Type II. By default, the LR1 statistic is applied.

ADJR1 applies the adjusted quantile regression R statistic.

VALIDATE applies the average check loss for the validation data.

Table 78.10 provides formulas and definitions for these fit statistics.

Table 78.10 Formulas and Definitions for Model Fit Summary Statistics for Single Quantile Effect Section

Statistic	Definition or Formula
n	Number of observations
p	Number of parameters including the intercept
$r_i(\tau)$	Residual for the ith observation; $r_i(\tau) = y_i - \mathbf{x}_i \boldsymbol{\beta}(\tau)$
$D(\tau)$	Total sum of check losses; $D(\tau) = \sum_{i=1}^{n} \rho_\tau(r_i)$
$D_0(\tau)$	Total sum of check losses for intercept-only model if intercept is a forced-in effect, otherwise for empty-model.
$\mathrm{ACL}(\tau)$	Average check loss; $\mathrm{ACL}(\tau) = \dfrac{D(\tau)}{n}$
$\mathrm{R1}(\tau)$	Counterpart of linear regression R-square for quantile regression; $1 - \dfrac{D(\tau)}{D_0(\tau)}$
$\mathrm{ADJR1}(\tau)$	Adjusted R1; $(\tau) = 1 - \dfrac{(n-1)D(\tau)}{(n-p)D_0(\tau)}$
$\mathrm{AIC}(\tau)$	$2n \ln(\mathrm{ACL}(\tau)) + 2p$
$\mathrm{AICC}(\tau)$	$2n \ln(\mathrm{ACL}(\tau)) + \dfrac{2pn}{n-p-1}$
$\mathrm{SBC}(\tau)$	$2n \ln(\mathrm{ACL}(\tau)) + p \ln(n)$

Quantile Process Effect Section

The following statistics are available for quantile process effect selection:

AIC specifies Akaike's information criterion (Akaike 1981; Darlington 1968; Judge et al. 1985).

AICC specifies the corrected Akaike's information criterion (Hurvich and Tsai 1989).

SBC specifies Schwarz Bayesian information criterion (Schwarz 1978; Judge et al. 1985).

ADJR1 specifies the adjusted quantile regression R statistic.

VALIDATE specifies average check loss for the validation data.

Table 78.11 provides formulas and definitions for the fit statistics.

Table 78.11 Formulas and Definitions for Model Fit Summary Statistics for Quantile Process Effect Section

Statistic	Definition or Formula
D	Integral of total sum of check losses; $D = \int_0^1 D(\tau) d\tau$
D_0	Integral of total sum of check losses for intercept-only model or empty-model if the NOINT option is used; $D_0 = \int_0^1 D_0(\tau) d\tau$
ACL	Integral of average check loss; $\text{ACL} = \frac{D}{n}$
R1	$1 - \frac{D}{D_0}$
ADJR1	Adjusted R1; $1 - \frac{(n-1)D}{(n-p)D_0}$
AIC	$\int_0^1 \text{AIC}(\tau) d\tau$
AICC	$\int_0^1 \text{AICC}(\tau) d\tau$
SBC	$\int_0^1 \text{SBC}(\tau) d\tau$

Macro Variables That Contain Selected Models

PROC QUANTSELECT saves the list of selected effects in a macro variable so that you can use other SAS procedures to perform post-selection analyses. This list does not explicitly include the intercept so that you can use it in the MODEL statement of other SAS/STAT regression procedures.

Table 78.12 describes the macro variables that PROC QUANTSELECT creates. When multiple quantile levels or BY processing are used, one macro variable, indexed by the quantile level order and the BY group number, is created for each quantile level and BY group combination.

Table 78.12 Macro Variables Created for Subsequent Processing

Macro Variable	Description
Single Quantile Level and No BY processing	
_QRSIND	Selected model
Multiple Quantile Levels and No BY Processing	
_QRSNUMTAUS	Number of quantile levels
_QRSINDT1	Selected model for the first quantile level
_QRSINDT2	Selected model for the second quantile level
. . .	
Single Quantile Level and BY Processing	
_QRSNUMBYS	Number of BY groups
_QRSIND1	Selected model for BY group 1
_QRSIND2	Selected model for BY group 2
. . .	
Multiple Quantile Levels and BY Processing	
_QRSNUMTAUS	Number of quantile levels
_QRSNUMBYS	Number of BY groups
_QRSIND1T1	Selected model for the first quantile level and BY group 1
_QRSIND1T2	Selected model for the second quantile level and BY group 1
. . .	
_QRSIND2T1	Selected model for the first quantile level and BY group 2
_QRSIND2T2	Selected model for the second quantile level and BY group 2
. . .	

The macro variables _QRSIND, _QRSINDT1, _QRSIND1, and _QRSIND1T1 are all synonyms. If you do not specify multiple quantile levels or BY processing, the macro variables _QRSNUMTAUS and _QRSNUMBYS are both set to 1.

PROC QUANTSELECT creates two output data set variables, _BY_ and _QUANTILE_, to aid in associating macro variables with output data set observations when multiple quantile levels or BY processing are used. The values of these two variables are integers that match the *i*,*j* components of the macro variable names _QRSIND*i*T*j*.

Using Validation and Test Data

When you have sufficient data, you can subdivide your data into three parts: training, validation, and test data. During the selection process, models are fit on the training data, and the prediction error for the models so obtained is found by using the validation data. This prediction error on the validation data can be used to decide when to terminate the selection process or to decide which effects to include as the selection process proceeds. Finally, after a selected model has been obtained, the test set can be used to assess how the selected model generalizes on data that played no role in selecting the model.

In some cases you might want to use only training and test data. For example, you might decide to use an information criterion to decide which effects to include and when to terminate the selection process. In this case no validation data are required, but test data can still be useful in assessing the predictive performance of the selected model. In other cases you might decide to use validation data during the selection process but forgo assessing the selected model on test data. Hastie, Tibshirani, and Friedman (2001) note that it is difficult to give a general rule for how many observations you should assign to each role. They note that a typical split might be 50% for training and 25% each for validation and testing.

PROC QUANTSELECT provides several methods for partitioning data into training, validation, and test data. You can provide data for each role in separate data sets that you specify with the DATA=, TESTDATA=, and VALDATA= options in the PROC QUANTSELECT procedure. An alternative method is to use a PARTITION statement to logically subdivide the DATA= data set into separate roles. You can name the fractions of the data that you want to reserve as test data and validation data. The following statements randomly subdivide the inData data set to use 25% of the data for validation and 25% for testing, leaving 50% of the data for training:

```
proc quantselect data=inData;
   partition fraction(test=0.25 validate=0.25);
   ...
run;
```

If you need to exercise more control over the partitioning of the input data set, you can name a variable in the input data set and a formatted value of that variable to correspond to each role. The following statements assign roles to observations in the inData data set based on the value of the variable named group in that data set:

```
proc quantselect data=inData;
   partition roleVar=group(test='group 1' train='group 2')
   ...
run;
```

Observations whose value of the variable group is `'group 1'` are assigned for testing, and those whose value is `'group 2'` are assigned to training. All other observations are ignored.

You can also combine the use of the PARTITION statement with named data sets for specifying data roles. For example, the following statements reserve 40% of the inData data set for validation, leaving the remaining 60% for training:

```
proc quantselect data=inData testData=inTest;
   partition fraction(validate=0.4);
   ...
run;
```

Data for testing are supplied in the inTest data set. Because a TESTDATA= data set is specified, additional observations for testing cannot be reserved by specifying a PARTITION statement.

When you use a PARTITION statement, the output data set that is created by an OUTPUT statement contains a character variable _ROLE_ whose values `'TRAIN'`, `'TEST'`, and `'VALIDATE'` indicate the role of each observation. _ROLE_ is blank for observations that were not assigned to any of these three roles. When the input data set specified in the DATA= option in the PROC QUANTSELECT statement contains an _ROLE_ variable, no PARTITION statement is used, and the TESTDATA= and VALDATA= options are not specified, then the _ROLE_ variable is used to define the roles of each observation. This is useful when you want to rerun PROC QUANTSELECT but use the same data partitioning as you used in a previous PROC QUANTSELECT step. For example, the following statements use the same data for testing and training in both PROC QUANTSELECT steps:

```
proc quantselect data=inData;
   partition fraction(test=0.5);
   model y=x1-x10 / selection=forward;
   output out=outDataForward;
run;

proc quantselect data=outDataForward;
   model y=x1-x10 / selection=backward;
run;
```

When you have reserved observations for training, validation, and testing, a model that is fit on the training data is scored on the validation and test data, and the average check loss, denoted by ACL, is computed separately for each of these subsets. The ACL for each data role is the sum of check losses for observations in that role divided by the number of observations in that role.

Using the Validation ACL as the STOP= Criterion

If you have provided observations for validation, then you can use the STOP=VALIDATE *method-option* to specify the validation ACL as the STOP= criterion in the SELECTION= option in the MODEL statement. At step *k* of the selection process, the best candidate effect to enter or leave the current model is determined. The “best candidate” means the effect that gives the best value of the SELECT= criterion that does not need to be based on the validation data. The validation ACL for the model with this candidate effect added is computed. If this validation ACL is greater than the validation ACL for the model at step *k*, then the selection process terminates at step *k*.

Using the Validation ACL as the CHOOSE= Criterion

When you specify the CHOOSE=VALIDATE *method-option* in the SELECTION= option in the MODEL statement, the validation ACL is computed for the models at each step of the selection process. The model that yields the smallest validation ACL and contains the fewest effects is selected.

Using the Validation ACL as the SELECT= Criterion

You request the validation ACL as the selection criterion by specifying the SELECT=VALIDATE *method-option* in the SELECTION= option in the MODEL statement. At step *k* of the selection process, the validation ACL is computed for each model where a candidate for entry is added or candidate for removal is dropped. The selected candidate for entry or removal is the one that yields a model with the minimal validation ACL.

Displayed Output

The following sections describe the output that is displayed by PROC QUANTSELECT. The output is organized into various tables, which are discussed in the order of appearance. The contents of a table might change depending on the options you specify.

Model Information

The "Model Information" table displays basic information about the data sets and the settings used to control effect selection. These settings include the following:

- the selection method
- the criteria used to select effects, stop the selection, and choose the selected model
- the effect hierarchy enforced

The ODS name of the "Model Information" table is ModelInfo.

Number of Observations

The "Number of Observations" table displays the number of observations read from the input data set and the number of observations used in the analysis. If you use a PARTITION statement, the table also displays the number of observations used for each data role. If you specify TESTDATA= or VALDATA= data sets in the PROC QUANTSELECT statement, then "Number of Observations" tables are also produced for these data sets. The ODS name of the "Number of Observations" table is NObs.

Class Level Information

The "Class Level Information" table lists the levels of every variable specified in the CLASS statement. The ODS name of the "Class Level Information" table is ClassLevelInfo.

Class Level Coding

The "Class Level Coding" table shows the coding used for every variable specified in the CLASS statement. The ODS name of the "Class Level Coding" table is ClassLevelCoding.

Candidates

The "Candidates" table displays the effect name and value of the criterion used to select entering or departing effects at each step of the selection process. The effects are displayed in sorted order from best to worst of the selection criterion. You request this table with the DETAILS= option in the MODEL statement. The ODS name of the "Candidates" table is either EntryCandidates for addition candidates or RemovalCandidates for removal candidates.

Selection Summary

The "Selection Summary" table displays details about the sequence of steps of the selection process. For each step, the effect that entered or dropped out is displayed along with the statistics used to select the effect, stop the selection, and choose the selected model. You can request that additional statistics be displayed with the STATS= option in the MODEL statement. For all criteria that you can use for effect selection, the steps at which the optimal values of these criteria occur are also indicated. The ODS name of the "Selection Summary" table is SelectionSummary.

Stop Reason

The "Stop Reason" table displays the reason why the selection stopped. Table 78.13 shows the possible stop reasons.

Table 78.13 Reasons for Stopping

Stop Reason	Description
1	The selected model is a perfect fit.
2	The specified maximum number of steps has been reached.
3	The specified maximum number of effects are in the model.
4	The specified minimum number of effects are in the model.
5	The stopping criterion found a local optimum.
6	No suitable add or drop candidate is available.
7	All effects are in the model.
8	All effects have been dropped.
9	The sequence of effect additions and removals is cycling.
10	Adding or dropping any effect does not improve the SELECT= criterion.
11	No effect is significant at the specified significance level for entry or significance level for staying levels.
12	All remaining effects are required.

The ODS name of the "Stop Reason" table is StopReason.

Selection Reason

The "Selection Reason" table displays how the final selected model is determined. Table 78.14 shows the possible selection reasons:

Table 78.14 Selection Reasons

Selection Reason	Description
1	The last valid model that occurs in the selection process is the final model.
2	The first model with the minimum CHOOSE= criterion value in the selection process is the final model.

The ODS name of the "Selection Reason" table is SelectionReason.

Selected Effects

The “Selected Effects” table displays a string that contains the list of effects in the selected model. The ODS name of the “Selected Effects” table is SelectedEffects.

Parameter Estimates

The “Parameter Estimates” table displays the parameters in the selected model and their estimates. The following information is displayed for each parameter in the selected model:

- the parameter label that includes the effect name and level information for effects that contain classification variables
- the degrees of freedom (DF) for the parameter. There is one degree of freedom unless the model is not full rank.
- the parameter estimate
- the standard parameter estimate, which is computed on a standardized design matrix. Let $\mathbf{X} = (\mathbf{X}_1, \mathbf{X}_2)$ denote the original design matrix, where $\mathbf{X}_1$ is the submatrix for all the forced-in effects, and $\mathbf{X}_2$ is the submatrix for the rest of the effects that are subject to selection. Let

  $$\mathbf{X}_2^* = \left[\mathbf{I} - \mathbf{X}_1(\mathbf{X}_1{}'\mathbf{X}_1)^{-1}\mathbf{X}_1{}'\right]\mathbf{X}_2 \text{ and } \mathbf{X}_2^{**} = \left(s_Y\sqrt{n-p_1}\right)^{-1}\left[\mathrm{diag}(\mathbf{X}_2^{*}{}'\mathbf{X}_2^*)\right]^{-\frac{1}{2}}\mathbf{X}_2^*$$

  where p_1 is the rank of $\mathbf{X}_1$ and $s_Y = \sqrt{\frac{\mathbf{Y}^{*}{}'\mathbf{Y}^*}{n-p_1}}$ with $\mathbf{Y}^* = \left[\mathbf{I} - \mathbf{X}_1(\mathbf{X}_1{}'\mathbf{X}_1)^{-1}\mathbf{X}_1{}'\right]\mathbf{Y}$.
  Then standard parameter estimates are defined as $(\mathbf{0}, \boldsymbol{\beta}_2^{**})$, where $(\boldsymbol{\beta}_1, \boldsymbol{\beta}_2^{**})$ are the parameter estimates computed on the standardized design matrix $(\mathbf{X}_1, \mathbf{X}_2^{**})$.

You can also use the DETAILS= option in the MODEL statement to request “Parameter Estimates” tables for the models at each step of the selection process. The ODS name of the “Parameter Estimates” table is “ParameterEstimates.”

ODS Table Names

PROC QUANTSELECT assigns a name to each table it creates. You can use these names to refer to the table when you use the Output Delivery System (ODS) to select tables and create output data sets. These names are listed in Table 78.15.

For more information about ODS, see Chapter 20, “Using the Output Delivery System.”

Table 78.15 ODS Tables Produced by PROC QUANTSELECT

ODS Table Name	Description	Statement	Option
BSplineDetails	B-spline basis details	EFFECT	DETAILS
EntryCandidates	Entry effect ranking	MODEL	DETAILS=
RemovalCandidates	Removal effect ranking	MODEL	DETAILS=

Table 78.15 *continued*

ODS Table Name	Description	Statement	Option
ClassLevelCoding	Classification variable coding	CLASS	SHOWCODING
ClassLevelInfo	Classification variable levels	CLASS	Default
CollectionLevelInfo	Levels of collection effects	EFFECT	DETAILS
MMLevelInfo	Levels of multimember effects	EFFECT	DETAILS
ModelInfo	Model information	MODEL	Default
NObs	Number of observations	MODEL	Default
ParameterNames	Labels for column names in the design matrix	PROC	OUTDESIGN(names)
ParameterEstimates	Selected model parameter estimates	MODEL	Default
PolynomialDetails	Polynomial details	EFFECT	DETAILS
PolynomialScaling	Polynomial scaling	EFFECT	DETAILS
SelectedEffects	List of selected effects	MODEL	Default
SelectionSummary	Selection summary	MODEL	Default
StopDetails	Stopping criterion details	MODEL	Default
StopReason	Reason why selection stopped	MODEL	Default
TPFSplineDetails	Thin-plate spline basis details	EFFECT	DETAILS

ODS Graphics

Statistical procedures use ODS Graphics to create graphs as part of their output. ODS Graphics is described in detail in Chapter 21, “Statistical Graphics Using ODS.”

Before you create graphs, ODS Graphics must be enabled (for example, by specifying the ODS GRAPHICS ON statement). For more information about enabling and disabling ODS Graphics, see the section “Enabling and Disabling ODS Graphics” on page 600 in Chapter 21, “Statistical Graphics Using ODS.”

The overall appearance of graphs is controlled by ODS styles. Styles and other aspects of using ODS Graphics are discussed in the section “A Primer on ODS Statistical Graphics” on page 599 in Chapter 21, “Statistical Graphics Using ODS.”

PROC QUANTSELECT assigns a name to each graph it creates using ODS. You can use these names to refer to the graphs when using ODS. The names are listed in Table 78.16.

Table 78.16 ODS Graphics Produced by PROC QUANTSELECT

ODS Graph Name	Plot Description	PLOTS Option
ACLPlot	Average check loss by step	ACL
AICCPlot	Corrected Akaike’s information criterion by step	CRITERIA(UNPACK)
AICPlot	Akaike’s information criterion by step	CRITERIA(UNPACK)
AdjR1Plot	Adjusted quantile regression R by step	CRITERIA(UNPACK)
ChooseCriterionPlot	CHOOSE= criterion by step	COEFFICIENTS(UNPACK)

Table 78.16 *continued*

ODS Graph Name	Plot Description	PLOTS= Option
CoefficientPanel	Coefficients and CHOOSE= criterion by step	COEFFICIENTS
CoefficientPlot	Coefficients by step	COEFFICIENTS(UNPACK)
CriterionPanel	Fit criteria by step	CRITERIA
SBCPlot	Schwarz Bayesian information criterion by step	CRITERIA(UNPACK)
ValidateACLPlot	Average square error on validation data by step	CRITERIA(UNPACK)

Example: QUANTSELECT Procedure

Example 78.1: Simulated Data Example

This simulation study exemplifies the unity of motive and effect for the PROC QUANTSELECT procedure. The following statements generate a data set that is based on a naive instrumental model (Chernozhukov and Hansen 2008):

```
%let seed=321;
%let p=20;
%let n=3000;

data analysisData;
   array x{&p} x1-x&p;
   do i=1 to &n;
      U  = ranuni(&seed);
      x1 = ranuni(&seed);
      x2 = ranexp(&seed);
      x3 = abs(rannor(&seed));
      y  = x1*(U-0.1) + x2*(U*U-0.25) + x3*(exp(U)-exp(0.9));
      do j=4 to &p;
         x{j} = ranuni(&seed);
      end;
      output;
   end;
run;
```

Variable U of the data set indicates the true quantile level of the response y conditional on $\mathbf{x} = (x_1, \cdots, x_p)$.

Let $Q_y(\tau|\mathbf{x}) = \mathbf{x}\boldsymbol{\beta}(\tau)$ denote the underlying quantile regression model, where $\boldsymbol{\beta}(\tau) = (\beta_1(\tau), \cdots, \beta_p(\tau))'$. Then, the true parameter functions are

$$\begin{aligned}
\beta_1(\tau) &= \tau - 0.1 \\
\beta_2(\tau) &= \tau^2 - 0.25 \\
\beta_3(\tau) &= \exp(\tau) - \exp(0.9) \\
\beta_4(\tau) &= ... = \beta_p(\tau) = 0
\end{aligned}$$

It is easy to see that, at $\tau = 0.1$, only $\beta_2(0.1) = -0.24$ and $\beta_3(0.1) = \exp(0.1) - \exp(0.9) \approx -1.354432$ are nonzero parameters. Therefore, an effective effect selection method should select x_2 and x_3 and drop all the other effects in this data set at $\tau = 0.1$. By the same rationale, x_1 and x_3 should be selected at $\tau = 0.5$ with $\beta_1(0.5) = 0.4$ and $\beta_3(0.5) \approx -0.810882$, and x_1 and x_2 should be selected at $\tau = 0.9$ with $\beta_1(0.9) = 0.8$ and $\beta_2(0.9) = 0.56$.

The following statements use PROC QUANTSELECT with the adaptive LASSO method:

```
proc quantselect data=analysisData;
   model y= x1-x&p / quantile=0.1 0.5 0.9
         selection=lasso(adaptive);
run;
```

The selected effects and the relevant estimates are shown in Output 78.1.1 for $\tau = 0.1$, Output 78.1.2 for $\tau = 0.5$, and Output 78.1.3 for $\tau = 0.9$. You can see that the adaptive LASSO method correctly selects active effects for all three quantile levels.

Output 78.1.1 Parameter Estimates at $\tau = 0.1$

```
                         The QUANTSELECT Procedure

                              Quantile = 0.1

                    Selected Effects: Intercept x2 x3


                           Parameter Estimates

                                                  Standardized
              Parameter     DF        Estimate        Estimate

              Intercept      1        0.011793               0
              x2             1       -0.228709       -0.218287
              x3             1       -1.379907       -0.784520
```

Output 78.1.2 Parameter Estimates at $\tau = 0.5$

```
                         The QUANTSELECT Procedure

                              Quantile = 0.5

                    Selected Effects: Intercept x1 x3


                           Parameter Estimates

                                                  Standardized
              Parameter     DF        Estimate        Estimate

              Intercept      1        0.011778               0
              x1             1        0.425843        0.118792
              x3             1       -0.863316       -0.490822
```

Output 78.1.3 Parameter Estimates at $\tau = 0.9$

```
                  The QUANTSELECT Procedure

                       Quantile = 0.9

              Selected Effects: Intercept x1 x2


                     Parameter Estimates

                                                Standardized
         Parameter     DF         Estimate          Estimate

         Intercept      1        -0.007738                 0
         x1             1         0.782942          0.218407
         x2             1         0.576445          0.550177
```

The QUANTSELECT procedure can perform effect selection not only at a single quantile level but also for the entire quantile process. You can specify the QUANTILE=PROCESS option to do effect selection for the entire quantile process. With the QUANTILE=PROCESS option specified, the ParameterEstimates table produced by the QUANTSELECT procedure actually shows the mean prediction model of y conditional on $\mathbf{x}$. In this simulation study, the true mean model is

$$\mathrm{E}(y|\mathbf{x}) = \mathbf{x}\boldsymbol{\beta}$$

where

$$\begin{aligned}
\beta_1 &= \mathrm{E}(U) - 0.1 = 0.4 \\
\beta_2 &= \mathrm{E}(U^2) - 0.25 \approx 0.083333 \\
\beta_3 &= \mathrm{E}(\exp(U)) - \exp(0.9) \approx -0.741321 \\
\beta_4 &= \ldots = \beta_p = 0
\end{aligned}$$

The following statements perform effect selection for the quantile process with the forward selection method.

```
proc quantselect data=analysisData;
   model y= x1-x&p / quantile=process(ntau=all)
         selection=forward;
run;
```

Output 78.1.4 shows that, by default, the SELECT= and STOP= options are both set to SBC. The selected effects and the relevant estimates for the conditional mean model are shown in Output 78.1.5.

Output 78.1.4 Model Information

```
                    The QUANTSELECT Procedure

                        Model Information

           Data Set                 WORK.ANALYSISDATA
           Dependent Variable                       y
           Selection Method                   Forward
           Quantile Type                      Process
           Select Criterion                       SBC
           Stop Criterion                         SBC
           Choose Criterion                       SBC
```

Output 78.1.5 Parameter Estimates

```
                       Parameter Estimates

                                                Standardized
       Parameter    DF         Estimate             Estimate

       Intercept     1         0.007833                    0
       x1            1         0.418825             0.116834
       x2            1         0.094791             0.090472
       x3            1        -0.785686            -0.446687
```

Linear regression is the most popular method for estimating conditional means. The following statements show how to select effects with the GLMSELECT procedure, and Output 78.1.6 shows the resulting selected effects and their estimates. You can see that the mean estimates from the QUANTSELECT procedure are similar to those from the GLMSELECT procedure. However, quantile regression can provide detailed distribution information, which is not available from linear regression.

```
proc glmselect data=analysisData;
   model y= x1-x3 / selection=forward(select=sbc stop=sbc choose=sbc);
run;
```

Output 78.1.6 Parameter Estimates

```
                     The GLMSELECT Procedure
                          Selected Model

                       Parameter Estimates

                                              Standard
  Parameter     DF        Estimate               Error    t Value

  Intercept      1       -0.010143            0.043129      -0.24
  x1             1        0.434553            0.057385       7.57
  x2             1        0.114183            0.016771       6.81
  x3             1       -0.797194            0.028156     -28.31
```

References

Akaike, H. (1981), "Likelihood of a Model and Information Criteria," *Journal of Econometrics*, 16, 3–14.

Belloni, A. and Chernozhukov, V. (2011), "L1-Penalized Quantile Regression in High-Dimensional Sparse Models," *Annals of Statistics*, 39, 82–130.

Chernozhukov, V. and Hansen, C. (2008), "Instrumental Variable Quantile Regression: A Robust Inference Approach," *Journal of Econometrics*, 142(1), 379–398.

Darlington, R. B. (1968), "Multiple Regression in Psychological Research and Practice," *Psychological Bulletin*, 69, 161–182.

Hastie, T., Tibshirani, R., and Friedman, J. (2001), *The Elements of Statistical Learning*, New York: Springer-Verlag.

Hurvich, C. M. and Tsai, C.-L. (1989), "Regression and Time Series Model Selection in Small Samples," *Biometrika*, 76, 297–307.

Judge, G. G., Griffiths, W. E., Hill, R. C., Lutkepohl, H., and Lee, T. C. (1985), *The Theory and Practice of Econometrics*, Second Edition, New York: John Wiley & Sons.

Koenker, R. and Bassett, G. W. (1978), "Regression Quantiles," *Econometrica*, 46, 33–50.

Reichler, J. L., ed. (1987), *The 1987 Baseball Encyclopedia Update*, New York: Macmillan.

Schwarz, G. (1978), "Estimating the Dimension of a Model," *Annals of Statistics*, 6, 461–464.

Tibshirani, R. (1996), "Regression Shrinkage and Selection via the Lasso," *Journal of the Royal Statistical Society, Series B*, 58, 267–288.

Time Inc. (1987), "What They Make," *Sports Illustrated*, 54–81.

Wu, Y. and Liu, Y. (2009), "Variable Selection in Quantile Regression," *Statistica Sinica*, 19, 801–817.

Zou, H. (2006), "The Adaptive Lasso and Its Oracle Properties," *Journal of the American Statistical Association*, 101, 1418–1429.

Subject Index

backward elimination
  QUANTSELECT procedure, 6597

candidates for addition or removal
  QUANTSELECT procedure, 6604
class level coding
  QUANTSELECT procedure, 6604
class level information
  QUANTSELECT procedure, 6604

displayed output
  QUANTSELECT procedure, 6604

effect selection
  QUANTSELECT procedure, 6597

fit statistics
  QUANTSELECT procedure, 6598
forward selection
  QUANTSELECT procedure, 6597

GLMSELECT procedure
  ODS Graphics, 6580

hierarchy
  QUANTSELECT procedure, 6589

macro variables
  QUANTSELECT procedure, 6600
model
  hierarchy (QUANTSELECT), 6589
model information
  QUANTSELECT procedure, 6604

number of observations
  QUANTSELECT procedure, 6604

ODS graph names
  QUANTSELECT procedure, 6607
ODS Graphics
  GLMSELECT procedure, 6580
options summary
  EFFECT statement, 6587

parameter estimates
  QUANTSELECT procedure, 6606

QUANTSELECT procedure, 6598
  backward elimination, 6597
  candidates for addition or removal, 6604
  class level coding, 6604
  class level information, 6604
  displayed output, 6604
  effect selection, 6597
  fit statistics, 6598
  forward selection, 6597
  hierarchy, 6589
  macro variables, 6600
  model hierarchy, 6589
  model information, 6604
  number of observations, 6604
  ODS graph names, 6607
  output table names, 6606
  parameter estimates, 6606
  selected effects, 6606
  selection reason, 6605
  selection summary, 6605
  stepwise selection, 6597
  stop reason, 6605
  test data, 6602
  validation data, 6602

selected effects
  QUANTSELECT procedure, 6606
selection reason
  QUANTSELECT procedure, 6605
selection summary
  QUANTSELECT procedure, 6605
stepwise selection
  QUANTSELECT procedure, 6597
stop reason
  QUANTSELECT procedure, 6605

test data
  QUANTSELECT procedure, 6602

validation data
  QUANTSELECT procedure, 6602

Syntax Index

ACL
  STATS= option (QUANTSELECT), 6594
ADAPTIVE option
  MODEL statement (QUANTSELECT), 6591
ADJR1
  STATS= option (QUANTSELECT), 6594
AIC
  STATS= option (QUANTSELECT), 6594
AICC
  STATS= option (QUANTSELECT), 6594

BY statement
  QUANTSELECT procedure, 6583

CHOOSE= option
  MODEL statement (QUANTSELECT), 6591
CLASS statement
  QUANTSELECT procedure, 6584

DATA= option
  PROC QUANTSELECT statement, 6577
DELIMITER option
  CLASS statement (QUANTSELECT), 6584
DESCENDING option
  CLASS statement (QUANTSELECT), 6584
DETAILS option
  MODEL statement (QUANTSELECT), 6589

EFFECT statement
  QUANTSELECT procedure, 6587

HIERARCHY= option
  MODEL statement (QUANTSELECT), 6589

INCLUDE= option
  MODEL statement (QUANTSELECT), 6591

keyword option
  OUTPUT statement (QUANTSELECT), 6595

LR1
  TEST= option (QUANTSELECT), 6594
LR2
  TEST= option (QUANTSELECT), 6594

MAXMACRO= option
  PROC QUANTSELECT statement, 6577
MAXSTEP option
  MODEL statement (QUANTSELECT), 6592
MISSING option
  CLASS statement (QUANTSELECT), 6584
MODEL statement
  QUANTSELECT procedure, 6588

NOINT option
  MODEL statement (QUANTSELECT), 6589
NOPRINT option
  PROC QUANTSELECT statement, 6579

ORDER= option
  CLASS statement (QUANTSELECT), 6585
OUT= option
  OUTPUT statement (QUANTSELECT), 6596
OUTDESIGN= option
  PROC QUANTSELECT statement, 6579
OUTPUT statement
  QUANTSELECT procedure, 6595

PARAM= option
  CLASS statement (QUANTSELECT), 6585
PARTITION statement
  QUANTSELECT procedure, 6596
PLOT option
  PROC QUANTSELECT statement, 6580
PLOTS option
  PROC QUANTSELECT statement, 6580
PREDICTED keyword
  OUTPUT statement (QUANTSELECT), 6595
PROC QUANTSELECT statement, *see*
    QUANTSELECT procedure

QUANTILES option
  MODEL statement (QUANTSELECT), 6590
QUANTSELECT procedure, 6576
  syntax, 6576
QUANTSELECT procedure, BY statement, 6583
QUANTSELECT procedure, CLASS statement, 6584
  DELIMITER option, 6584
  DESCENDING option, 6584
  MISSING option, 6584
  ORDER= option, 6585
  PARAM= option, 6585
  REF= v-option, 6586
  SHOWCODING option, 6584
  SPLIT option, 6586
QUANTSELECT procedure, MODEL statement,
    6588
  ADAPTIVE option, 6591
  CHOOSE= option, 6591

DETAILS option, 6589
HIERARCHY= option, 6589
INCLUDE= option, 6591
MAXSTEP option, 6592
NOINT option, 6589
NTAU= option, 6590
QUANTILES= option, 6590
SELECT= option, 6592
SELECTION= option, 6590
SLENTRY= option, 6592
SLSTAY= option, 6592
STATS option, 6593
STOP= option, 6592
STOPHORIZON= option, 6593
TEST option, 6594
QUANTSELECT procedure, OUTPUT statement, 6595
keyword option, 6595
OUT= option, 6596
PREDICTED keyword, 6595
RESIDUAL keyword, 6595
QUANTSELECT procedure, PARTITION statement, 6596
FRACTION option, 6596
ROLEVAR= option, 6596
QUANTSELECT procedure, PROC QUANTSELECT statement, 6576
DATA= option, 6577
MAXMACRO= option, 6577
NOPRINT option, 6579
OUTDESIGN= option, 6579
PLOT option, 6580
PLOTS option, 6580
SEED= option, 6582
TESTDATA= option, 6582
VALDATA= option, 6583
QUANTSELECT procedure, STATS= option
ACL, 6594
ADJR1, 6594
AIC, 6594
AICC, 6594
R1, 6594
SBC, 6594
QUANTSELECT procedure, TEST= option
LR1, 6594
LR2, 6594
QUANTSELECT procedure, WEIGHT statement, 6596
QUANTSELECT procedure, EFFECT statement, 6587

R1
STATS= option (QUANTSELECT), 6594
RANDOM option
QUANTSELECT procedure, PARTITION statement, 6596
REF= v-option
CLASS statement (QUANTSELECT), 6586
RESIDUAL keyword
OUTPUT statement (QUANTSELECT), 6595
ROLEVAR= option
QUANTSELECT procedure, PARTITION statement, 6596

SBC
STATS= option (QUANTSELECT), 6594
SEED= option
PROC QUANTSELECT statement, 6582
SELECT= option
MODEL statement (QUANTSELECT), 6592
SELECTION= option
MODEL statement (QUANTSELECT), 6590
SHOWCODING option
CLASS statement (QUANTSELECT), 6584
SLENTRY= option
MODEL statement (QUANTSELECT), 6592
SLSTAY= option
MODEL statement (QUANTSELECT), 6592
SPLIT option
CLASS statement (QUANTSELECT), 6586
STATS option
MODEL statement (QUANTSELECT), 6593
STOP= option
MODEL statement (QUANTSELECT), 6592
STOPHORIZON= option
MODEL statement (QUANTSELECT), 6593

TEST option
MODEL statement (QUANTSELECT), 6594
TESTDATA= option
PROC QUANTSELECT statement, 6582

VALDATA= option
PROC QUANTSELECT statement, 6583

WEIGHT statement
QUANTSELECT procedure, 6596

Your Turn

We welcome your feedback.

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