SAS/STAT® 14.1 User's Guide
The ROBUSTREG Procedure

This document is an individual chapter from *SAS/STAT® 14.1 User's Guide*.

The correct bibliographic citation for this manual is as follows: SAS Institute Inc. 2015. *SAS/STAT® 14.1 User's Guide*. Cary, NC: SAS Institute Inc.

SAS/STAT® 14.1 User's Guide

SAS Institute Inc., SAS Campus Drive, Cary, NC 27513-2414

July 2015

Chapter 98
The ROBUSTREG Procedure

Contents

Overview: ROBUSTREG Procedure

The main purpose of robust regression is to detect outliers and provide resistant (stable) results in the presence of outliers. In order to achieve this stability, robust regression limits the influence of outliers. Historically, robust regression techniques have addressed three classes of problems:

- problems with outliers in the Y direction (response direction)
- problems with multivariate outliers in the X space (that is, outliers in the covariate space, which are also referred to as leverage points)
- problems with outliers in both the Y direction and the X space

Many methods have been developed in response to these problems. However, in statistical applications of outlier detection and robust regression, the methods that are most commonly used today are Huber M estimation, high breakdown value estimation, and combinations of these two methods. The ROBUSTREG procedure provides four such methods: M estimation, LTS estimation, S estimation, and MM estimation.

- M estimation, introduced by Huber (1973), is the simplest approach both computationally and theoretically. Although it is not robust with respect to leverage points, it is still used extensively in data analysis when contamination can be assumed to be mainly in the response direction.
- Least trimmed squares (LTS) estimation is a high breakdown value method that was introduced by Rousseeuw (1984). The breakdown value is a measure of the proportion of contamination that an estimation method can withstand and still maintain its robustness. The performance of this method was improved by the FAST-LTS algorithm of Rousseeuw and Van Driessen (2000).
- S estimation is a high breakdown value method that was introduced by Rousseeuw and Yohai (1984). Given the same breakdown value, S estimation has a higher statistical efficiency than LTS estimation.
- MM estimation, introduced by Yohai (1987), combines high breakdown value estimation and M estimation. It has the same high breakdown property as S estimation but a higher statistical efficiency.

Features

The ROBUSTREG procedure has the following main features:

- offers four estimation methods: M, LTS, S, and MM
- provides 10 weight functions for M estimation
- provides robust R square and deviance for all estimates
- provides asymptotic covariance and confidence intervals for regression parameters by using the M, S, and MM methods

- provides robust Wald and F tests for regression parameters by using the M and MM methods
- provides Mahalanobis distance and robust Mahalanobis distance by using the generalized minimum covariance determinant (MCD) algorithm
- provides outlier and leverage-point diagnostics
- supports parallel computing for S and LTS estimates
- supports constructed effects, including spline and multimember effects
- produces fit plots and diagnostic plots by using ODS Graphics

Getting Started: ROBUSTREG Procedure

The following examples demonstrate how you can use the ROBUSTREG procedure to fit a linear regression model and obtain outlier and leverage-point diagnostics.

M Estimation

This example shows how you can use the ROBUSTREG procedure to do M estimation, which is a commonly used method for outlier detection and robust regression when contamination is mainly in the response direction.

The following data set, Stack, is the well-known stack loss data set presented by Brownlee (1965). The data describe the operation of a plant for the oxidation of ammonia to nitric acid and consist of 21 four-dimensional observations. The explanatory variables for the response stack loss (y) are the rate of operation (x1), the cooling water inlet temperature (x2), and the acid concentration (x3).

```
data Stack;
   input  x1 x2 x3 y exp $ @@;
   datalines;
80  27   89  42   e1   80  27   88  37   e2
75  25   90  37   e3   62  24   87  28   e4
62  22   87  18   e5   62  23   87  18   e6
62  24   93  19   e7   62  24   93  20   e8
58  23   87  15   e9   58  18   80  14  e10
58  18   89  14  e11   58  17   88  13  e12
58  18   82  11  e13   58  19   93  12  e14
50  18   89   8  e15   50  18   86   7  e16
50  19   72   8  e17   50  19   79   8  e18
50  20   80   9  e19   56  20   82  15  e20
70  20   91  15  e21
;
```

The following PROC ROBUSTREG statements analyze the data:

```
proc robustreg data=stack;
   model y = x1 x2 x3 / diagnostics leverage;
   id    exp;
   test  x3;
run;
```

By default, the ROBUSTREG procedure uses the bisquare weight function to do M estimation, and it uses the median method to estimate the scale parameter. The MODEL statement specifies the covariate effects. The DIAGNOSTICS option requests a table for outlier diagnostics, and the LEVERAGE option adds leverage-point diagnostic results to this table for continuous covariate effects. The ID statement specifies that the variable exp be used to identify each observation (experiment) in this table. If the ID statement is omitted, the observation number is used to identify the observations. The TEST statement requests a test of significance for the covariate effects that are specified. The results of this analysis are displayed in Figures 98.1 through 98.3.

Figure 98.1 displays the model-fitting information and summary statistics for the response variable and the continuous covariates. The Q1, Median, and Q3 columns provide the lower quantile, median, and upper quantile, respectively. The MAD column provides a robust estimate of the univariate scale, which is computed as the standardized median absolute deviation (MAD). For more information about the standardized MAD, see Huber (1981, p. 108). The Mean and Standard Deviation columns provide the usual mean and standard deviation. A large difference between the standard deviation and the MAD for a variable indicates some extreme values for that variable. In the stack loss data, the stack loss (response y) has the biggest difference between the standard deviation and the MAD.

Figure 98.1 Model-Fitting Information and Summary Statistics

The ROBUSTREG Procedure

Model Information	
Data Set	WORK.STACK
Dependent Variable	y
Number of Independent Variables	3
Number of Observations	21
Method	M Estimation

Summary Statistics

Variable	Q1	Median	Q3	Mean	Standard Deviation	MAD
x1	53.0000	58.0000	62.0000	60.4286	9.1683	5.9304
x2	18.0000	20.0000	24.0000	21.0952	3.1608	2.9652
x3	82.0000	87.0000	89.5000	86.2857	5.3586	4.4478
y	10.0000	15.0000	19.5000	17.5238	10.1716	5.9304

Figure 98.2 displays the table of robust parameter estimates, standard errors, and confidence limits. The Scale row provides a point estimate of the scale parameter in the linear regression model, which is obtained by the median method. For more information about scale estimation methods, see the section "M Estimation" on page 8046. For the stack loss data, M estimation yields the fitted linear model:

$$\hat{y} = -42.2845 + 0.9276\text{x1} + 0.6507\text{x2} - 0.1123\text{x3}$$

Figure 98.2 Model Parameter Estimates

Parameter Estimates

Parameter	DF	Estimate	Standard Error	95% Confidence Limits		Chi-Square	Pr > ChiSq
Intercept	1	-42.2854	9.5045	-60.9138	-23.6569	19.79	<.0001
x1	1	0.9276	0.1077	0.7164	1.1387	74.11	<.0001
x2	1	0.6507	0.2940	0.0744	1.2270	4.90	0.0269
x3	1	-0.1123	0.1249	-0.3571	0.1324	0.81	0.3683
Scale	1	2.2819					

Figure 98.3 displays outlier and leverage-point diagnostics. Standardized robust residuals are computed based on the estimated parameters. Both the Mahalanobis distance and the robust MCD distance are displayed. Outliers and leverage points, identified by asterisks, are defined by the standardized robust residuals and robust MCD distances that exceed the corresponding cutoff values displayed in the diagnostics summary. Observations 4 and 21 are outliers because their standardized robust residuals exceed the cutoff value in absolute value. The procedure detects four observations that have high leverage values. Leverage points (points with high leverage values) that have smaller standardized robust residuals than the cutoff value in absolute value are called good leverage points; others are called bad leverage points. Observation 21 is a bad leverage point.

Figure 98.3 Diagnostics

Diagnostics

Obs	exp	Mahalanobis Distance	Robust MCD Distance	Leverage	Standardized Robust Residual	Outlier
1	e1	2.2536	5.5284	*	1.0995	
2	e2	2.3247	5.6374	*	-1.1409	
3	e3	1.5937	4.1972	*	1.5604	
4	e4	1.2719	1.5887		3.0381	*
21	e21	2.1768	3.6573	*	-4.5733	*

Two particularly useful plots for revealing outliers and leverage points are a scatter plot of the standardized robust residuals against the robust distances (RD plot) and a scatter plot of the robust distances against the classical Mahalanobis distances (DD plot).

For the stack loss data, the following statements produce the RD plot in Figure 98.4 and the DD plot in Figure 98.5. The histogram and the normal quantile-quantile plots (shown in Figure 98.6 and Figure 98.7, respectively) for the standardized robust residuals are also created using the HISTOGRAM and QQPLOT suboptions of the PLOTS= option.

```
ods graphics on;

proc robustreg data=stack plots=(rdplot ddplot histogram qqplot);
   model y = x1 x2 x3;
run;

ods graphics off;
```

These plots are helpful in identifying outliers in addition to good and bad high-leverage points.

These plots are requested when ODS Graphics is enabled by specifying the PLOTS= option in the PROC ROBUSTREG statement. For general information about ODS Graphics, see Chapter 21, "Statistical Graphics Using ODS." For specific information about the graphics available in the ROBUSTREG procedure, see the section "ODS Graphics" on page 8073.

Figure 98.4 RD Plot for Stack Loss Data

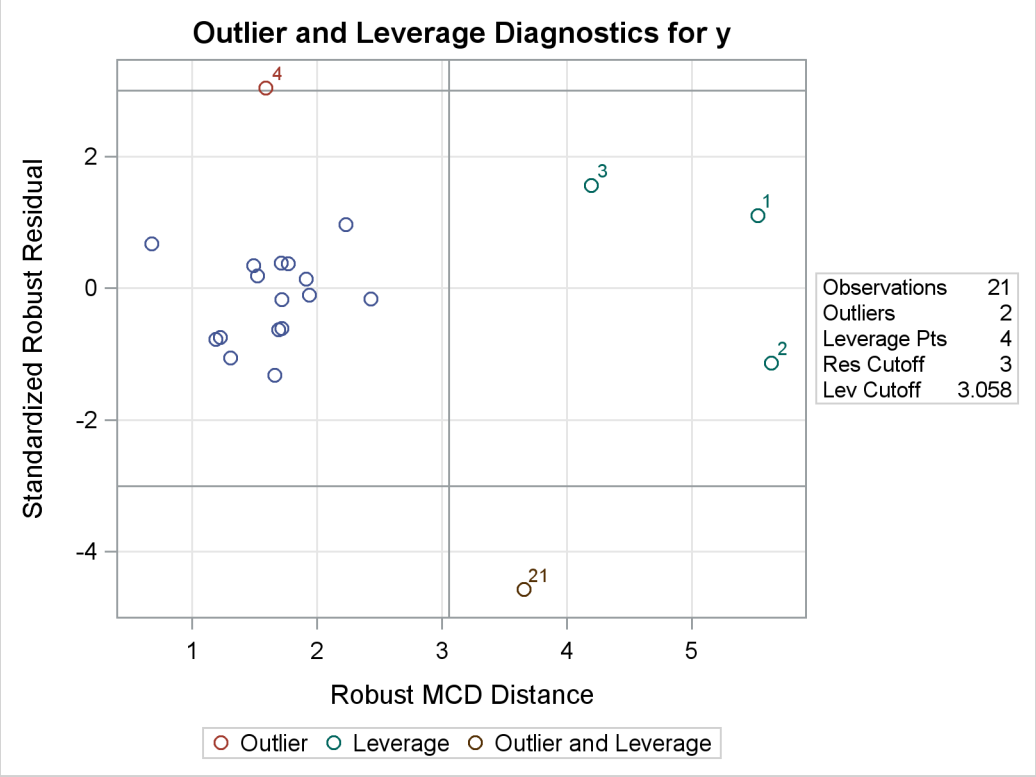


Figure 98.5 DD Plot for Stack Loss Data

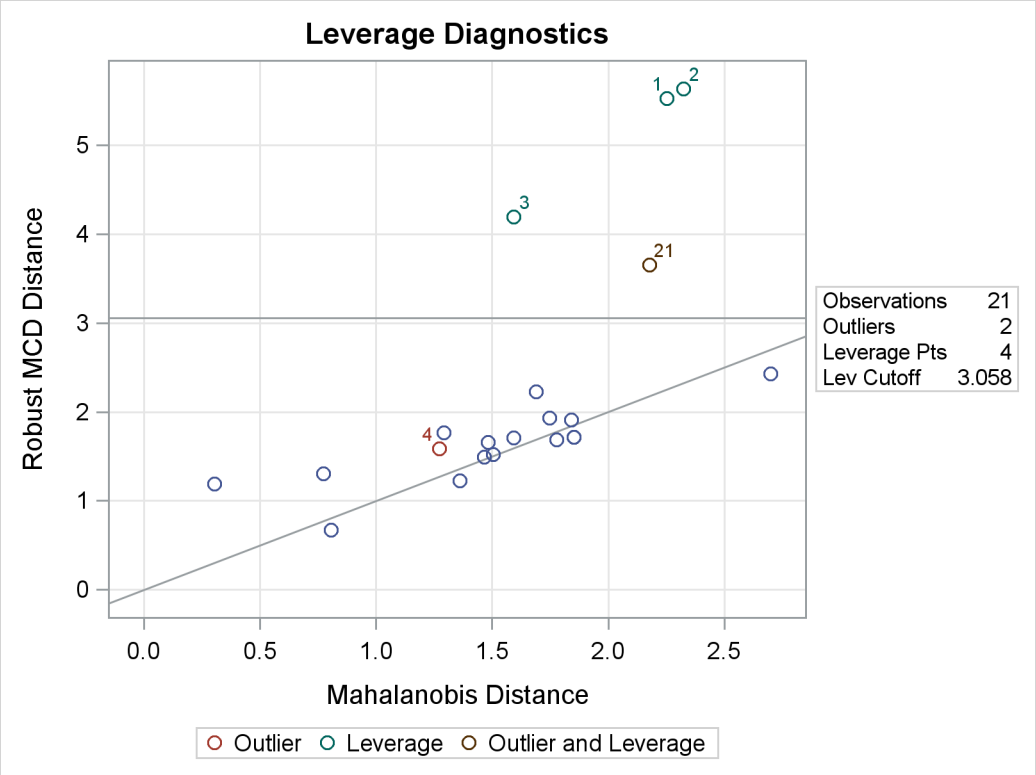


Figure 98.6 Histogram

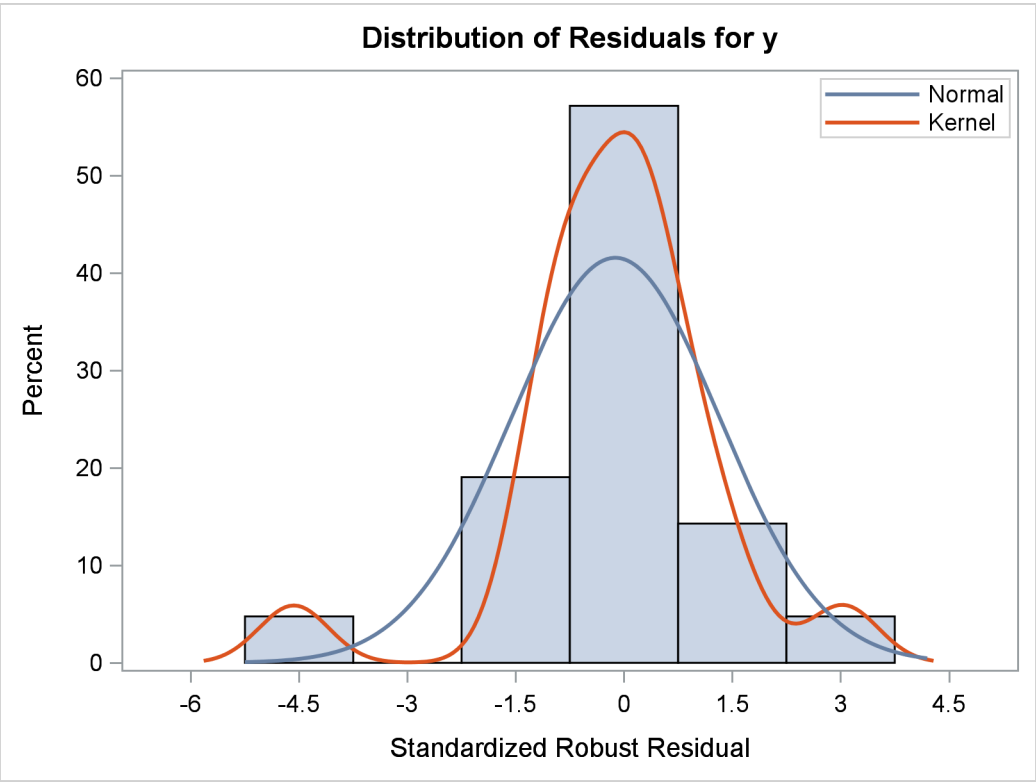


Figure 98.7 Q-Q Plot

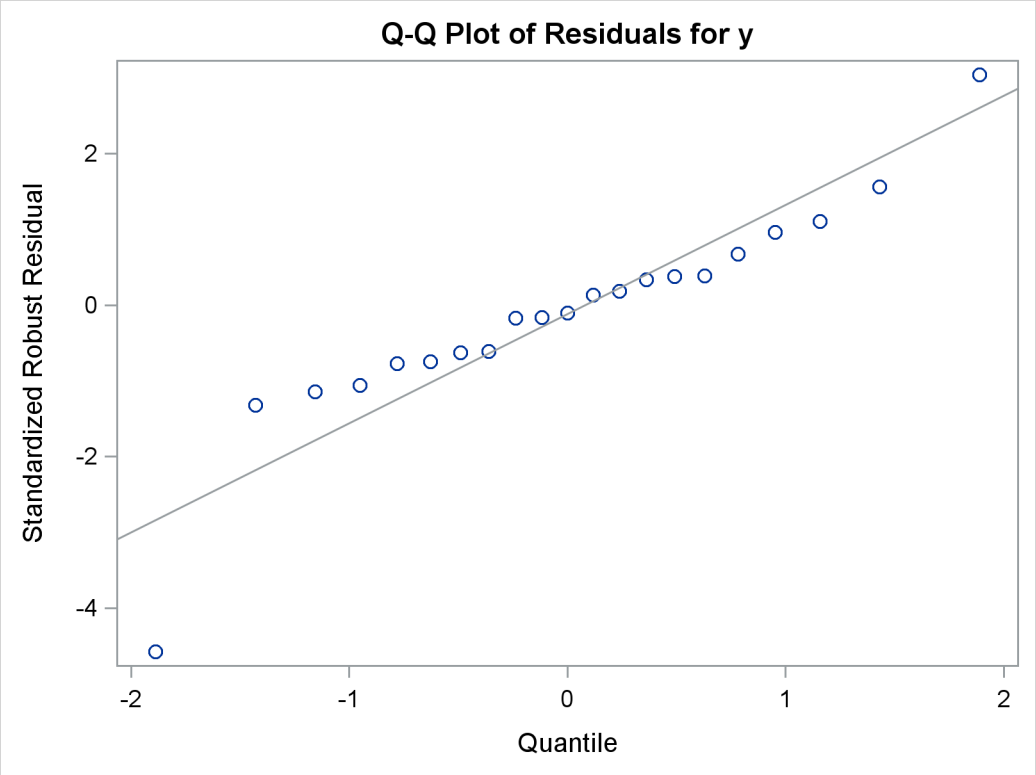


Figure 98.8 displays robust versions of goodness-of-fit statistics for the model. You can use the robust information criteria, AICR and BICR, for model selection and comparison. For both AICR and BICR, the lower the value, the more desirable the model.

Figure 98.8 Goodness-of-Fit Statistics

Goodness-of-Fit	
Statistic	**Value**
R-Square	0.6659
AICR	29.5231
BICR	36.3361
Deviance	125.7905

Figure 98.9 displays the test results that are requested by the TEST statement. The ROBUSTREG procedure conducts two robust linear tests, the ρ test and the R_n^2 test. For information about how the ROBUSTREG procedure computes test statistics and the correction factor lambda, see the section "Linear Tests" on page 8052. Because of the large p-values for both tests, you can conclude that the effect x3 is not significant at the 5% level.

Figure 98.9 Test of Significance

Robust Linear Test					
Test	Test Statistic	Lambda	DF	Chi-Square	Pr > ChiSq
Rho	0.9378	0.7977	1	1.18	0.2782
Rn2	0.8092		1	0.81	0.3683

For the bisquare weight function, the default tuning constant, $c = 4.685$, is chosen to yield a 95% asymptotic efficiency of the M estimates with the Gaussian distribution. For more information, see the section "M Estimation" on page 8046. The smaller the constant c, the lower the asymptotic efficiency but the sharper the M estimate as an outlier detector. For the stack loss data set, you could consider using a sharper outlier detector.

The following PROC ROBUSTREG step uses a smaller constant, $c = 3.5$. This tuning constant corresponds to an efficiency close to 85%. For the relationship between the tuning constant and asymptotic efficiency of M estimates, see Chen and Yin (2002).

```
proc robustreg method=m(wf=bisquare(c=3.5)) data=stack;
   model y = x1 x2 x3 / diagnostics leverage;
   id    exp;
   test  x3;
run;
```

Figure 98.10 displays the table of robust parameter estimates, standard errors, and confidence limits when PROC ROBUSTREG uses the constant $c = 3.5$.

The refitted linear model is

$$\hat{y} = -37.1076 + 0.8191x1 + 0.5173x2 - 0.0728x3$$

Figure 98.10 Model Parameter Estimates

The ROBUSTREG Procedure

Parameter Estimates							
Parameter	DF	Estimate	Standard Error	95% Confidence Limits		Chi-Square	Pr > ChiSq
Intercept	1	-37.1076	5.4731	-47.8346	-26.3805	45.97	<.0001
x1	1	0.8191	0.0620	0.6975	0.9407	174.28	<.0001
x2	1	0.5173	0.1693	0.1855	0.8492	9.33	0.0022
x3	1	-0.0728	0.0719	-0.2138	0.0681	1.03	0.3111
Scale	1	1.4265					

Figure 98.11 displays outlier and leverage-point diagnostics when the constant $c = 3.5$ is used. Besides observations 4 and 21, observations 1 and 3 are also detected as outliers.

Figure 98.11 Diagnostics

Diagnostics						
Obs	exp	Mahalanobis Distance	Robust MCD Distance	Leverage	Standardized Robust Residual	Outlier
1	e1	2.2536	5.5284	*	4.2719	*
2	e2	2.3247	5.6374	*	0.7158	
3	e3	1.5937	4.1972	*	4.4142	*
4	e4	1.2719	1.5887		5.7792	*
21	e21	2.1768	3.6573	*	-6.2727	*

LTS Estimation

If the data are contaminated in the X space, M estimation might yield improper results. It is better to use the high breakdown value method. This example shows how you can use LTS estimation to deal with X-space contaminated data. The following data set, hbk, is an artificial data set that was generated by Hawkins, Bradu, and Kass (1984).

```
data hbk;
   input index $ x1 x2 x3 y @@;
   datalines;
 1  10.1  19.6  28.3   9.7          2   9.5  20.5  28.9  10.1
 3  10.7  20.2  31.0  10.3          4   9.9  21.5  31.7   9.5
 5  10.3  21.1  31.1  10.0          6  10.8  20.4  29.2  10.0
 7  10.5  20.9  29.1  10.8          8   9.9  19.6  28.8  10.3
 9   9.7  20.7  31.0   9.6         10   9.3  19.7  30.3   9.9
11  11.0  24.0  35.0  -0.2         12  12.0  23.0  37.0  -0.4
13  12.0  26.0  34.0   0.7         14  11.0  34.0  34.0   0.1
15   3.4   2.9   2.1  -0.4         16   3.1   2.2   0.3   0.6
17   0.0   1.6   0.2  -0.2         18   2.3   1.6   2.0   0.0
19   0.8   2.9   1.6   0.1         20   3.1   3.4   2.2   0.4
21   2.6   2.2   1.9   0.9         22   0.4   3.2   1.9   0.3
23   2.0   2.3   0.8  -0.8         24   1.3   2.3   0.5   0.7
25   1.0   0.0   0.4  -0.3         26   0.9   3.3   2.5  -0.8
27   3.3   2.5   2.9  -0.7         28   1.8   0.8   2.0   0.3
29   1.2   0.9   0.8   0.3         30   1.2   0.7   3.4  -0.3
31   3.1   1.4   1.0   0.0         32   0.5   2.4   0.3  -0.4
33   1.5   3.1   1.5  -0.6         34   0.4   0.0   0.7  -0.7
35   3.1   2.4   3.0   0.3         36   1.1   2.2   2.7  -1.0
37   0.1   3.0   2.6  -0.6         38   1.5   1.2   0.2   0.9
39   2.1   0.0   1.2  -0.7         40   0.5   2.0   1.2  -0.5
41   3.4   1.6   2.9  -0.1         42   0.3   1.0   2.7  -0.7
43   0.1   3.3   0.9   0.6         44   1.8   0.5   3.2  -0.7
45   1.9   0.1   0.6  -0.5         46   1.8   0.5   3.0  -0.4
47   3.0   0.1   0.8  -0.9         48   3.1   1.6   3.0   0.1
49   3.1   2.5   1.9   0.9         50   2.1   2.8   2.9  -0.4
51   2.3   1.5   0.4   0.7         52   3.3   0.6   1.2  -0.5
```

```
53   0.3   0.4   3.3   0.7        54   1.1   3.0   0.3   0.7
55   0.5   2.4   0.9   0.0        56   1.8   3.2   0.9   0.1
57   1.8   0.7   0.7   0.7        58   2.4   3.4   1.5  -0.1
59   1.6   2.1   3.0  -0.3        60   0.3   1.5   3.3  -0.9
61   0.4   3.4   3.0  -0.3        62   0.9   0.1   0.3   0.6
63   1.1   2.7   0.2  -0.3        64   2.8   3.0   2.9  -0.5
65   2.0   0.7   2.7   0.6        66   0.2   1.8   0.8  -0.9
67   1.6   2.0   1.2  -0.7        68   0.1   0.0   1.1   0.6
69   2.0   0.6   0.3   0.2        70   1.0   2.2   2.9   0.7
71   2.2   2.5   2.3   0.2        72   0.6   2.0   1.5  -0.2
73   0.3   1.7   2.2   0.4        74   0.0   2.2   1.6  -0.9
75   0.3   0.4   2.6   0.2
;
```

Both ordinary least squares (OLS) estimation and M estimation (not shown here) suggest that observations 11 to 14 are outliers. However, these four observations were generated from the underlying model, whereas observations 1 to 10 were contaminated. The reason that OLS estimation and M estimation do not pick up the contaminated observations is that they cannot distinguish good leverage points (observations 11 to 14) from bad leverage points (observations 1 to 10). In such cases, the LTS method identifies the true outliers.

The following statements invoke the ROBUSTREG procedure and use the LTS estimation method:

```
proc robustreg data=hbk fwls method=lts;
   model y = x1 x2 x3 / diagnostics leverage;
   id index;
run;
```

Figure 98.12 displays the model-fitting information and summary statistics for the response variable and independent covariates.

Figure 98.12 Model-Fitting Information and Summary Statistics

The ROBUSTREG Procedure

Model Information	
Data Set	WORK.HBK
Dependent Variable	y
Number of Independent Variables	3
Number of Observations	75
Method	LTS Estimation

Summary Statistics

Variable	Q1	Median	Q3	Mean	Standard Deviation	MAD
x1	0.8000	1.8000	3.1000	3.2067	3.6526	1.9274
x2	1.0000	2.2000	3.3000	5.5973	8.2391	1.6309
x3	0.9000	2.1000	3.0000	7.2307	11.7403	1.7791
y	-0.5000	0.1000	0.7000	1.2787	3.4928	0.8896

Figure 98.13 displays information about the LTS fit, which includes the breakdown value of the LTS estimate. The breakdown value is a measure of the proportion of contamination that an estimation method can withstand and still maintain its robustness. In this example the LTS estimate minimizes the sum of 57 smallest squares of residuals. It can still estimate the true underlying model if the remaining 18 observations are contaminated. This corresponds to a breakdown value around 0.25, which is set as the default.

Figure 98.13 LTS Profile

LTS Profile	
Total Number of Observations	75
Number of Squares Minimized	57
Number of Coefficients	4
Highest Possible Breakdown Value	0.2533

Figure 98.14 displays parameter estimates for covariates and scale. Two robust estimates of the scale parameter are displayed. For information about computing these estimates, see the section "Final Weighted Scale Estimator" on page 8055. The weighted scale estimator (Wscale) is a more efficient estimator of the scale parameter.

Figure 98.14 LTS Parameter Estimates

LTS Parameter Estimates		
Parameter	**DF**	**Estimate**
Intercept	1	-0.3431
x1	1	0.0901
x2	1	0.0703
x3	1	-0.0731
Scale (sLTS)	0	0.7451
Scale (Wscale)	0	0.5749

Figure 98.15 displays outlier and leverage-point diagnostics. The ID variable index is used to identify the observations. If you do not specify this ID variable, the observation number is used to identify the observations. However, the observation number depends on how the data are read. The first 10 observations are identified as outliers, and observations 11 to 14 are identified as good leverage points.

Figure 98.15 Diagnostics

Diagnostics

Obs	index	Mahalanobis Distance	Robust MCD Distance	Leverage	Standardized Robust Residual	Outlier
1	1	1.9168	29.4424	*	17.0868	*
2	2	1.8558	30.2054	*	17.8428	*
3	3	2.3137	31.8909	*	18.3063	*
4	4	2.2297	32.8621	*	16.9702	*
5	5	2.1001	32.2778	*	17.7498	*
6	6	2.1462	30.5892	*	17.5155	*
7	7	2.0105	30.6807	*	18.8801	*
8	8	1.9193	29.7994	*	18.2253	*
9	9	2.2212	31.9537	*	17.1843	*
10	10	2.3335	30.9429	*	17.8021	*
11	11	2.4465	36.6384	*	0.0406	
12	12	3.1083	37.9552	*	-0.0874	
13	13	2.6624	36.9175	*	1.0776	
14	14	6.3816	41.0914	*	-0.7875	

Figure 98.16 displays the final weighted least squares estimates. These estimates are least squares estimates that are computed after the detected outliers are deleted.

Figure 98.16 Final Weighted LS Estimates

Parameter Estimates for Final Weighted Least Squares Fit

Parameter	DF	Estimate	Standard Error	95% Confidence Limits		Chi-Square	Pr > ChiSq
Intercept	1	-0.1805	0.1044	-0.3852	0.0242	2.99	0.0840
x1	1	0.0814	0.0667	-0.0493	0.2120	1.49	0.2222
x2	1	0.0399	0.0405	-0.0394	0.1192	0.97	0.3242
x3	1	-0.0517	0.0354	-0.1210	0.0177	2.13	0.1441
Scale	0	0.5572					

Syntax: ROBUSTREG Procedure

The following statements are available in the ROBUSTREG procedure:

PROC ROBUSTREG < *options* > **;**
BY *variables* **;**
CLASS *variables* **;**
EFFECT *name*=*effect-type*(*variables* < / *options* >) **;**
ID *variables* **;**
MODEL *response* = < *effects* > < / *options* > **;**
OUTPUT < **OUT**=*SAS-data-set* > *keyword*=*name* < . . . *keyword*=*name* > **;**
PERFORMANCE < *options* > **;**
TEST *effects* **;**
WEIGHT *variable* **;**

The PROC ROBUSTREG statement invokes the procedure. The METHOD= option in the PROC ROBUSTREG statement selects one of the four estimation methods, M, LTS, S, and MM. By default, Huber M estimation is used. The MODEL statement is required and specifies the variables to be used in the regression. You can specify main effects and interaction terms in the MODEL statement, as in the GLM procedure. (See Chapter 46, "The GLM Procedure.") The CLASS statement specifies which explanatory variables to treat as categorical. The ID statement names variables to identify observations in the outlier diagnostics tables. The WEIGHT statement identifies a variable in the input data set whose values are used to weight the observations. The OUTPUT statement creates an output data set that contains final weights, predicted values, and residuals. The TEST statement requests robust linear tests for the model parameters. The PERFORMANCE statement tunes the performance of the procedure by using single or multiple processors available on the hardware. Multiple OUTPUT and TEST statements are permitted.

PROC ROBUSTREG Statement

PROC ROBUSTREG < *options* > **;**

The PROC ROBUSTREG statement invokes the ROBUSTREG procedure. Table 98.1 summarizes the *options* available in the PROC ROBUSTREG statement.

Table 98.1 PROC ROBUSTREG Statement Options

Option	Description
COVOUT	Saves the estimated covariance matrix
DATA=	Specifies the input SAS data set
FWLS	Computes the final weighted least squares estimates
INEST=	Specifies an input SAS data set that contains initial estimates
ITPRINT	Displays the iteration history of the iteratively reweighted least squares algorithm
METHOD=	Specifies the estimation method
NAMELEN=	Specifies the length of effect names
ORDER=	Specifies the order in which to sort classification variables
OUTEST=	Specifies an output SAS data set that contains the parameter estimates

Table 98.1 *continued*

Option	Description
PLOT	Specifies options that control details of the plots
SEED=	Specifies the seed for the random number generator

You can specify the following *options* in the PROC ROBUSTREG statement.

COVOUT
saves the estimated covariance matrix in the OUTEST= data set. This option is not supported for LTS estimation.

DATA=*SAS-data-set*
specifies the input SAS data set to be used by PROC ROBUSTREG. By default, the most recently created SAS data set is used.

FWLS
computes the final weighted least squares estimates. These estimates are equivalent to the least squares estimates after the detected outliers are deleted.

INEST=*SAS-data-set*
specifies an input SAS data set that contains initial estimates for all the parameters in the model. For a detailed description of the contents of the INEST= data set, see the section "INEST= Data Set" on page 8070.

ITPRINT
displays the iteration history of the iteratively reweighted least squares algorithm that is used in M and MM estimation. You can also use this option in the MODEL statement.

METHOD=*method-type* **<(***options***)>**
specifies the estimation method and some additional *options* for the estimation method. PROC ROBUSTREG provides four estimation methods: M estimation, LTS estimation, S estimation, and MM estimation. The default method is M estimation.

NOTE: Because the LTS and S methods use subsampling algorithms, these methods are not suitable in an analysis that uses variables that have only a few unequal values or a few unequal values within one BY group. For example, indicator variables that correspond to a classification variable often fall into this category. The same issue also applies to the initial LTS and S estimates in the MM method. For a model that includes classification independent variables or continuous independent variables with a few unequal values, the M method is recommended.

NAMELEN=*n*
specifies the length of effect names in tables and output data sets to be *n* characters, where *n* is a value between 20 and 200. The default length is 20 characters.

ORDER=DATA | FORMATTED | FREQ | INTERNAL
specifies the sort order for the levels of the classification variables (which are specified in the CLASS statement).

This option applies to the levels for all classification variables, except when you use the (default) ORDER=FORMATTED option with numeric classification variables that have no explicit format. In that case, the levels of such variables are ordered by their internal value.

The ORDER= option can take the following values:

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

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

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

OUTEST=*SAS-data-set*
specifies an output SAS data set that contains the parameter estimates and, if the COVOUT option is specified, the estimated covariance matrix. For a detailed description of the contents of the OUTEST= data set, see the section "OUTEST= Data Set" on page 8070.

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

PLOT | PLOTS <(*global-plot-options***)> <=(***plot-request* **< ...** *plot-request* **>)>**
specifies options that control details of the plots. If ODS Graphics is enabled but you do not specify the PLOTS= option, then PROC ROBUSTREG produces the robust fit plot by default when the model includes a single continuous independent variable.

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

```
ods graphics on;
proc robustreg data=stack plots=all;
   model y = x1 x2 x3;
run;
ods graphics off;
```

For more information about enabling and disabling ODS Graphics, see the section "Enabling and Disabling ODS Graphics" on page 609 in Chapter 21, "Statistical Graphics Using ODS."

The *global-plot-options* apply to all plots that are generated by the ROBUSTREG procedure. The following *global-plot-option* is available:

ONLY
suppresses the default robust fit plot. Only plots that are specifically requested are displayed.

You can specify more than one *plot-request* within the parentheses after PLOTS=. For a single *plot-request*, you can omit the parentheses. The following *plot-requests* are available:

ALL
creates all appropriate plots.

DDPLOT < (LABEL=ALL | LEVERAGE | NONE | OUTLIER) >
creates a plot of robust distance against Mahalanobis distance. For more information about robust distance, see the section "Leverage-Point and Outlier Detection" on page 8068. The LABEL= option specifies how the points in this plot are to be labeled, as summarized in Table 98.2.

Table 98.2 Options for Label

Value of LABEL=	Label Method
ALL	Label all points
LEVERAGE	Label leverage points
NONE	No labels
OUTLIERS	Label outliers

By default, the ROBUSTREG procedure labels both outliers and leverage points.

If you specify ID variables in the ID statement, the values of the first ID variable are used as labels; otherwise, observation numbers are used as labels.

FITPLOT < (NOLIMITS) >
creates a plot of robust fit against the single independent continuous variable that is specified in the model. You can request this plot when only a single independent continuous variable is specified in the model. Confidence limits are added to the plot by default. The NOLIMITS option suppresses these limits.

HISTOGRAM
creates a histogram for the standardized robust residuals. The histogram is superimposed with a normal density curve and a kernel density curve.

NONE
suppresses all plots.

QQPLOT
creates the normal quantile-quantile plot for the standardized robust residuals.

RDPLOT < (LABEL=ALL | LEVERAGE | NONE | OUTLIER) >
creates the plot of standardized robust residual against robust distance. For more information about robust distance, see the section "Leverage-Point and Outlier Detection" on page 8068. The LABEL= option specifies a label method for points in this plot. These label methods are described in Table 98.2.

If you specify ID variables in the ID statement, the values of the first ID variable are used as labels; otherwise, observation numbers are used as labels.

SEED=*number*
specifies the seed for the random number generator used to randomly select the subgroups and subsets for LTS and S estimation. By default, or if you specify 0, the ROBUSTREG procedure generates a random seed.

Options with METHOD=M

When you specify METHOD=M <(*options*)>, you can specify the following *options*:

ASYMPCOV=H1 | H2 | H3
specifies the type of asymptotic covariance that is computed for the M estimate. The three types are described in the section "Asymptotic Covariance and Confidence Intervals" on page 8051. By default, ASYMPCOV=H1.

CONVERGENCE=*criterion* **<(EPS=***value***)>**
specifies a convergence criterion for the M estimate. Table 98.3 lists the three criteria that are available.

Table 98.3 Options to Specify Convergence Criteria

Type	Option
Coefficient	**CONVERGENCE=COEF**
Residual	**CONVERGENCE=RESID**
Weight	**CONVERGENCE=WEIGHT**

By default, CONVERGENCE=COEF. You can specify the precision of the convergence criterion by using the EPS= option; by default, EPS=1E–8.

MAXITER=*n*
sets the maximum number of iterations during the parameter estimation. By default, MAXITER=1000.

SCALE=*scale-type* **|** *value*
specifies the scale parameter or a method of estimating the scale parameter. These methods and options are summarized in Table 98.4.

Table 98.4 Options to Specify Scale

Scale	Option	Default *d*
Fixed constant	**SCALE=***value*	
Huber estimate	**SCALE=HUBER** <(D=*d*)>	2.5
Median estimate	**SCALE=MED**	
Tukey estimate	**SCALE=TUKEY** <(D=*d*)>	2.5

By default, SCALE=MED.

WEIGHTFUNCTION=*function-type*
WF=*function-type*
specifies the weight function that is used for the M estimate. The ROBUSTREG procedure provides 10 weight functions, which are listed in Table 98.5. You can specify the parameters in these functions by using the A=, B=, and C= options. These functions are described in the section "M Estimation" on page 8046. The default weight function is bisquare.

Table 98.5 Options to Specify Weight Functions

Weight Function	Option	Default *a*, *b*, *c*
Andrews	**WF=ANDREWS** <(C=*c*)>	1.339
Bisquare	**WF=BISQUARE** <(C=*c*)>	4.685
Cauchy	**WF=CAUCHY** <(C=*c*)>	2.385
Fair	**WF=FAIR** <(C=*c*)>	1.4
Hampel	**WF=HAMPEL** <(<A=*a*><B=*b*><C=*c*>)>	2, 4, 8
Huber	**WF=HUBER** <(C=*c*)>	1.345
Logistic	**WF=LOGISTIC** <(C=*c*)>	1.205
Median	**WF=MEDIAN** <(C=*c*)>	0.01
Talworth	**WF=TALWORTH** <(C=*c*)>	2.795
Welsch	**WF=WELSCH** <(C=*c*)>	2.985

Options with METHOD=LTS

When you specify METHOD=LTS <(*options*)>, you can specify the following *options*:

CSTEP=*n*
specifies the number of concentration steps (C-steps) for the LTS estimate. For information about how the default value is determined, see the section "LTS Estimate" on page 8053.

H=*n*
specifies the quantile for the LTS estimate. For information about how the default value is determined, see the section "LTS Estimate" on page 8053

IADJUST=ALL | NONE
requests (IADJUST=ALL) or suppresses (IADJUST=NONE) the intercept adjustment for all estimates in the LTS algorithm. By default, the intercept adjustment is used for data sets that contain fewer than 10,000 observations. For more information, see the section "Algorithm" on page 8053.

NBEST=*n*
specifies the number of best solutions that are kept for each subgroup during the computation of the LTS estimate. The default number is 10, which is the maximum number allowed.

NREP=*n*
specifies the number of times to repeat least squares fit in subgroups during the computation of the LTS estimate. For information about how the default number is determined, see the section "LTS Estimate" on page 8053.

SUBANALYSIS
requests a display of the subgrouping information and parameter estimates within subgroups. This option generates the ODS tables that are listed in Table 98.6.

Table 98.6 ODS Tables Available with SUBANALYSIS Option

ODS Table Name	Description
BestEstimates	Best final estimates for LTS
BestSubEstimates	Best estimates for each subgroup
CStep	C-step information for LTS
Groups	Grouping information for LTS

SUBGROUPSIZE=*n*
specifies the data set size of the subgroups in the computation of the LTS estimate. The default number is 300.

Options with METHOD=S

When you specify METHOD=S <(*options*)>, you can specify the following *options*:

ASYMPCOV=H1 | H2 | H3 | H4
specifies the type of asymptotic covariance that is computed for the S estimate. The four types are described in the section "Asymptotic Covariance and Confidence Intervals" on page 8058. By default, ASYMPCOV=H4.

CHIF= TUKEY | YOHAI
specifies the χ function for the S estimate. PROC ROBUSTREG provides two χ functions, Tukey's bisquare function and Yohai's optimal function, which you can request by specifying CHIF=TUKEY and CHIF=YOHAI, respectively. The default is Tukey's bisquare function.

EFF=*value*
specifies the efficiency (as a fraction) of the S estimate. The parameter k_0 in the χ function is determined by this efficiency. The default efficiency is determined such that the consistent S estimate has a breakdown value of 25%. This option is overwritten by the K0= option if both options are used.

K0=*value*
specifies the k_0 parameter in the χ function of the S estimate. If you specify CHIF=TUKEY, the default is 1.548. If you specify CHIF=YOHAI, the default is 0.66. These default values correspond to a 50% breakdown value of the consistent S estimate.

MAXITER=*n*
sets the maximum number of iterations for computing the scale parameter of the S estimate. By default, MAXITER=1000.

NREP=*n*
specifies the number of repeats of subsampling in the computation of the S estimate. For information about how the default number of repeats is determined, see the section "Algorithm" on page 8056.

NOREFINE
suppresses the refinement of the S estimate. For more information, see the section "Algorithm" on page 8056.

SUBSETSIZE=*n*
specifies the size of the subset for the S estimate. For information about how the default value is determined, see the section "Algorithm" on page 8056.

TOLERANCE=*value*
specifies the tolerance for the S estimate of the scale. The default value is 0.001.

Options with METHOD=MM

When you specify METHOD=MM <(*options*)>, you can specify the following *options*:

ASYMPCOV=H1 | H2 | H3 | H4
specifies the type of asymptotic covariance that is computed for the MM estimate. The four types are described in the section "Details: ROBUSTREG Procedure" on page 8046. By default, ASYMPCOV=H4.

BIASTEST <(ALPHA=*number***)>**
requests the bias test for the final MM estimate. For more information about this test, see the section "Bias Test" on page 8060.

CHIF=TUKEY | YOHAI
selects the χ function for the MM estimate. PROC ROBUSTREG provides two χ functions, Tukey's bisquare function and Yohai's optimal function, which you can request by specifying CHIF=TUKEY and CHIF=YOHAI, respectively. The default is Tukey's bisquare function. This χ function is also used by the initial S estimate if you specify the INITEST=S option.

CONVERGENCE=*criterion* **<(EPS=***number***)>**
specifies a convergence criterion for the MM estimate. Table 98.7 lists the three criteria that are available.

Table 98.7 Options to Specify Convergence Criteria

Type	Option
Coefficient	**CONVERGENCE=COEF**
Residual	**CONVERGENCE=RESID**
Weight	**CONVERGENCE=WEIGHT**

By default, CONVERGENCE=COEF. You can specify the precision of the convergence criterion by using the EPS= option; by default, EPS=1E–8.

EFF=*value*
specifies the efficiency (as a fraction) of the MM estimate. The parameter k_1 in the χ function is determined by this efficiency. The default efficiency is set to 0.85, which corresponds to $k_1 = 3.440$ if you specify CHIF=TUKEY or $k_1 = 0.868$ if you specify CHIF=YOHAI.

INITEST=LTS | S
specifies the initial estimator for the MM estimator. By default, the LTS estimator with its default settings is used as the initial estimator for the MM estimator.

INITH=*h*
specifies the integer *h* for the initial LTS estimate that is used by the MM estimator. For information about how to specify *h* and how the default is determined, see the section "Algorithm" on page 8059.

K0=*number*
specifies the parameter k_0 in the χ function for the MM estimate. If you specify CHIF=TUKEY, the default is $k_0 = 2.9366$. If you specify CHIF=YOHAI, the default is $k_0 = 0.7405$. These default values correspond to the 25% breakdown value of the MM estimator.

MAXITER=*n*
sets the maximum number of iterations during the parameter estimation. By default, MAXITER=1000.

BY Statement

BY *variables* **;**

You can specify a BY statement with PROC ROBUSTREG 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 ROBUSTREG 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 *variables* < / **TRUNCATE** > **;**

The CLASS statement names the classification variables to be used in the model. Typical classification variables are Treatment, Sex, Race, Group, and Replication. If you use the CLASS statement, it must appear before the MODEL statement.

Classification variables can be either character or numeric. By default, class levels are determined from the entire set of formatted values of the CLASS variables.

NOTE: Prior to SAS 9, class levels were determined by using no more than the first 16 characters of the formatted values. To revert to this previous behavior, you can use the TRUNCATE option in the CLASS statement.

In any case, you can use formats to group values into levels. See the discussion of the FORMAT procedure in the *Base SAS Procedures Guide* and the discussions of the FORMAT statement and SAS formats in *SAS Formats and Informats: Reference*. You can adjust the order of CLASS variable levels with the ORDER= option in the PROC ROBUSTREG statement.

You can specify the following *option* in the CLASS statement after a slash (/):

TRUNCATE
specifies that class levels should be determined by using only up to the first 16 characters of the formatted values of CLASS variables. When formatted values are longer than 16 characters, you can use this option to revert to the levels as determined in releases prior to SAS 9.

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 391 in Chapter 19, "Shared Concepts and Topics."

You can specify the following *effect-types*:

COLLECTION specifies 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 specifies a classification effect in which the level that is used for a particular period corresponds to the level in the preceding period.

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

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

SPLINE specifies 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 98.8 summarizes the *options* available in the EFFECT statement.

Table 98.8 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. This option is required.
WITHIN=	Names the variable or variables that define the group within which each period is defined. This option is required.
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
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 effect
DEGREE=	Specifies the degree of the spline effect
KNOTMETHOD=	Specifies how to construct the knots for the spline effect

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 401 in Chapter 19, "Shared Concepts and Topics."

ID Statement

ID *variables* **;**

When you request the diagnostics table by specifying the DIAGNOSTICS option in the MODEL statement, the variables that are listed in the ID statement are displayed in addition to the observation number. You can

use these variables to identify each observation. If the ID statement is omitted, the observation number is used to identify the observations.

MODEL Statement

< *label* **:>** **MODEL** *response* **=** **<** *effects* **>** **<** */* *options* **>** **;**

You can specify main effects and interaction terms in the MODEL statement, as in the GLM procedure (see Chapter 46, "The GLM Procedure").

The optional *label*, which must be a valid SAS name, is used to label the model in the OUTEST data set.

Table 98.9 summarizes the *options* available in the MODEL statement.

Table 98.9 MODEL Statement Options

Option	Description
ALPHA=	Specifies the significance level
CORRB	Produces the estimated correlation matrix
COVB	Produces the estimated covariance matrix
CUTOFF=	Specifies the multiplier of the cutoff value for outlier detection
DIAGNOSTICS	Requests the outlier diagnostics
FAILRATIO=	Specifies the failure-ratio threshold
ITPRINT	Displays the iteration history
LEVERAGE	Requests an analysis of leverage points
NOGOODFIT	Suppresses the computation of goodness-of-fit statistics
NOINT	Specifies no-intercept regression
SINGULAR=	Specifies the tolerance for testing singularity

You can specify the following *options* for the model fit.

ALPHA=*value*
specifies the significance level for the confidence intervals for regression parameters. The *value* must be between 0 and 1. By default, ALPHA=0.05.

CORRB
produces the estimated correlation matrix of the parameter estimates.

COVB
produces the estimated covariance matrix of the parameter estimates.

CUTOFF=*value*
specifies the multiplier of the cutoff value for outlier detection. By default, CUTOFF=3.

DIAGNOSTICS <(ALL)>
requests the outlier diagnostics. By default, only observations that are identified as outliers or leverage points are displayed. Specify the ALL option to display all observations.

FAILRATIO=*value*
specifies the failure-ratio threshold for the subsampling algorithm of an LTS or S estimate. It also applies to the initial LTS or S step in an MM estimate. The threshold must be between 0 and 1. Its default value is 0.99. For more information, see the section "LTS Estimate" on page 8053 or "S Estimate" on page 8055.

ITPRINT
displays the iteration history of the iteratively reweighted least squares algorithm that is used by M and MM estimation. You can also use this option in the PROC ROBUSTREG statement.

LEVERAGE < (*leverage-options***) >**
requests an analysis of leverage points for the covariates. The results are added to the diagnostics table, which you can request by specifying the DIAGNOSTICS option in the MODEL statement.

You can use the following *leverage-options*:

CUTOFF=*value*
specifies the leverage cutoff value for leverage-point detection. For more information, see the section "Leverage-Point and Outlier Detection" on page 8068. You can also specify the cutoff value by using the CUTOFFALPHA= option.

CUTOFFALPHA=*alpha-value*
specifies the leverage cutoff α value for leverage-point detection. The respective leverage cutoff value equals $\sqrt{\chi^2_{p;1-\alpha}}$ (or $\sqrt{\chi^2_{q;1-\alpha}}$ if projection is applied in the generalized MCD algorithm). By default, $\alpha = 0.025$.

H=*n*

QUANTILE=*n*
specifies the quantile to be minimized for the MCD algorithm that is used for the leverage-point analysis. By default, H=$[(3n + p + 1)/4]$, where n is the number of observations and p is the number of independent variables, excluding the intercept.

MCDALPHA=*alpha-value*
specifies the MCD cutoff α value for the final MCD reweighting step. The respective MCD cutoff value equals $\sqrt{\chi^2_{p;1-\alpha}}$ (or $\sqrt{\chi^2_{q;1-\alpha}}$ if projection is applied in the generalized MCD algorithm). By default, $\alpha = 0.025$.

MCDCUTOFF=*value*

MCDCUT=*value*
specifies the MCD cutoff value for the final MCD reweighting step. For more information, see the section "Mahalanobis Distance versus Robust Distance" on page 8062 and Rousseeuw and Van Driessen (1999). You can also specify the cutoff value by using the MCDALPHA= option.

MCDINFO
requests that detailed information about the MCD covariance estimate be displayed, including the low-dimensional structure, the breakdown value, the MCD center, and the MCD covariance itself. The option outputs the ODS tables of the MCD profile, MCD center, MCD covariance, and MCD correlation.

OPC | OFFPLANECOEF
requests the ODS table of the coefficients for MCD-dropped components, when projection is applied in the generalized MCD algorithm. The OFFPLANECOEF option is ignored for the regular MCD algorithm.

PROJECTIONALPHA=*alpha-value*

PALPHA=*alpha-value*
specifies the projection cutoff α value to be used to judge whether an observation is on or off the low-dimensional hyperplane that is identified by the generalized MCD algorithm. The respective projection cutoff value equals $\sqrt{\chi^2_{1;1-\alpha}}$. By default, $\alpha = 0.001$.

PROJECTIONCUTOFF=*value*

PCUTOFF=*value*
specifies the projection cutoff value to be used to judge whether an observation is on or off the low-dimensional hyperplane identified by the projected MCD algorithm. For more information, see the section "Mahalanobis Distance versus Robust Distance" on page 8062 and Rousseeuw and Van Driessen (1999). You can also specify the projection cutoff value by using the PALPHA= option.

PROJECTIONTOLERANCE=*value*

PTOL=*value*
specifies the projection tolerance value for the low-dimensional structure detection. For more information, see the section "Leverage-Point and Outlier Detection" on page 8068.

NOGOODFIT
suppresses the computation of goodness-of-fit statistics.

NOINT
specifies no-intercept regression.

SINGULAR=*value*
specifies the tolerance for testing singularity of the information matrix and the crossproducts matrix for the initial least squares estimates. By default, SINGULAR=1E–12.

OUTPUT Statement

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

The OUTPUT statement creates an output SAS data set that contains statistics calculated after the model is fitted. At least one specification of the form *keyword*=*name* is required.

All variables in the original data set are included in the new data set, along with the variables that are created by using *keyword*= options in the OUTPUT statement. These new variables contain fitted values and estimated quantiles. To create a SAS data set in a permanent library, you must specify a two-level name. For more information about permanent libraries and SAS data sets, see *SAS Language Reference: Concepts*.

You can use the following specifications in the OUTPUT statement:

OUT=*SAS-data-set* specifies the new data set. By default, the procedure uses the DATA*n* convention to name the new data set.

keyword=*name* specifies the statistics to include in the output data set and gives names to the new variables. Specify a *keyword* for each desired statistic (see the following list), an equal sign, and the variable to contain the statistic.

The *keywords* that are allowed and the statistics that they represent are as follows:

LEVERAGE specifies a variable to indicate leverage points. To include this variable in the OUT= data set, you must specify the LEVERAGE option in the MODEL statement. For information about how to define the LEVERAGE keyword, see the section "Leverage-Point and Outlier Detection" on page 8068.

MD specifies a variable to contain the Mahalanobis distances. For the definition of Mahalanobis distance, see the section "Robust Distance" on page 8062.

OUTLIER specifies a variable to indicate outliers. For information about how to define the OUTLIER keyword, see the section "Leverage-Point and Outlier Detection" on page 8068.

PMD specifies a variable to contain the projected Mahalanobis distances. For the definition of projected Mahalanobis distance, see the section "Robust Distance" on page 8062.

POD specifies a variable to contain the projected off-plane distances. For the definition of off-plane distance, see the section "Robust Distance" on page 8062.

PRD specifies a variable to contain the projected robust MCD Mahalanobis distances. For the definition of projected robust distance, see the section "Robust Distance" on page 8062.

PREDICTED | P specifies a variable to contain the estimated responses

$$\hat{y}_i = \mathbf{x}_i' \hat{\boldsymbol{\theta}}$$

RD specifies a variable to contain the robust MCD Mahalanobis distances. For the definition of robust distance, see the section "Robust Distance" on page 8062.

RESIDUAL | R specifies a variable to contain the unstandardized residuals

$$y_i - \hat{y}_i \text{ or } y_i - \mathbf{x}_i' \hat{\boldsymbol{\theta}}$$

SRESIDUAL | SR specifies a variable to contain the standardized residuals

$$\frac{y_i - \hat{y}_i}{\hat{\sigma}} \text{ or } \frac{y_i - \mathbf{x}_i' \hat{\boldsymbol{\theta}}}{\hat{\sigma}}.$$

By default, the LTS method uses Wscale as $\hat{\sigma}$ for computing the standardized residuals.

STDP specifies a variable to contain the estimates of the standard errors of the estimated mean responses

$$\sqrt{\mathbf{x}_i' \boldsymbol{\Sigma} \mathbf{x}_i}$$

where $\boldsymbol{\Sigma}$ denotes the covariance matrix of the parameter estimates. You can request the ODS table of this covariance matrix by specifying the COVB option in the MODEL statement. The STDP= option is applied to M, S, and MM estimation, but not to LTS estimation.

STDI specifies a variable to contain the estimates of the standard errors of the individual predicted values

$$\sqrt{\mathbf{x}_i' \boldsymbol{\Sigma} \mathbf{x}_i + \hat{\sigma}^2}.$$

The STDI= option is applied to M, S, and MM estimation, but not to LTS estimation.

WEIGHT specifies a variable to contain the computed final weights.

PERFORMANCE Statement

The PERFORMANCE statement is used to specify options that affect the performance of PROC ROBUSTREG and to request tables that show the performance options in effect and timing details. See Chen (2002) for some empirical results.

PERFORMANCE < *options* > **;**

You can specify the following *options*:

CPUCOUNT=*n* | **ACTUAL**
specifies the number of processors to use in forming crossproduct matrices. You can specify any integer in the range 1–1024 for *n*. CPUCOUNT=ACTUAL sets CPUCOUNT to the number of physical processors available. This can be less than the number of physical CPUs if the SAS process has been restricted by system administration tools. Setting CPUCOUNT= to a number greater than the actual number of available CPUs might result in reduced performance. This option overrides the SAS system option CPUCOUNT=. If CPUCOUNT=1, then NOTHREADS is in effect, and PROC ROBUSTREG uses singly threaded code.

DETAILS
requests the PerfSettings table that shows the performance settings in effect and the "Timing" table that provides a broad timing breakdown of the PROC ROBUSTREG step.

NOTHREADS
disables multithreaded computation. This option overrides the SAS system option THREADS.

THREADS
enables multithreaded computation. This option overrides the SAS system option NOTHREADS.

TEST Statement

< *label* : > **TEST** *effects* **;**

When you use M estimation and MM estimation, the TEST statement provides a means of obtaining a test for the canonical linear hypothesis about the parameters of the tested effects

$$\theta_j = 0, \quad j = i_1, \ldots, i_q$$

where q is the total number of parameters of the tested effects.

PROC ROBUSTREG provides two kinds of robust tests: the ρ test and the R_n^2 test. They are described in the section "Details: ROBUSTREG Procedure" on page 8046. No test is available for LTS and S estimation.

The optional *label*, which must be a valid SAS name, is used to label output from the corresponding TEST statement.

WEIGHT Statement

WEIGHT *variable* **;**

The WEIGHT statement specifies a weight variable in the input data set.

If you want to use fixed weights for each observation in the input data set, place the weights in a variable in the data set and specify the name in a WEIGHT statement. The values of the WEIGHT variable can be nonintegral and are not truncated. Observations that have nonpositive or missing values for the weight variable do not contribute to the fit of the model.

Details: ROBUSTREG Procedure

This section describes the statistical and computational aspects of the ROBUSTREG procedure. The following notation is used throughout the section.

Let $\mathbf{X} = (x_{ij})$ denote an $n \times p$ matrix, let $\mathbf{y} = (y_1, \ldots, y_n)'$ denote a given n-vector of responses, and let $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_p)'$ denote an unknown p-vector of parameters or coefficients whose components are to be estimated. The matrix $\mathbf{X}$ is called the design matrix. Consider the usual linear model,

$$\mathbf{y} = \mathbf{X}\boldsymbol{\theta} + \mathbf{e}$$

where $\mathbf{e} = (e_1, \ldots, e_n)'$ is an n-vector of unknown errors. It is assumed that (for a given $\mathbf{X}$) the components e_i of $\mathbf{e}$ are independent and identically distributed according to a distribution $L(\cdot/\sigma)$, where σ is a scale parameter (usually unknown). Often $L(\cdot) \approx \Phi(\cdot)$, the standard normal distribution function. The vector of residuals for a given value of $\hat{\boldsymbol{\theta}}$ is denoted by $\mathbf{r} = (r_1, \ldots, r_n)'$, and the ith row of the matrix $\mathbf{X}$ is denoted by $\mathbf{x}_i'$.

M Estimation

M estimation in the context of regression was first introduced by Huber (1973) as a result of making the least squares approach robust. Although M estimators are not robust with respect to leverage points, they are popular in applications where leverage points are not an issue.

Instead of minimizing a sum of squares of the residuals, a Huber-type M estimator $\hat{\boldsymbol{\theta}}_M$ of $\boldsymbol{\theta}$ minimizes a sum of less rapidly increasing functions of the residuals:

$$Q(\boldsymbol{\theta}) = \sum_{i=1}^{n} \rho\left(\frac{r_i}{\sigma}\right)$$

where $\mathbf{r} = \mathbf{y} - \mathbf{X}\boldsymbol{\theta}$. For the ordinary least squares estimation, ρ is the square function, $\rho(z) = z^2$.

If σ is known, then when derivatives are taken with respect to $\boldsymbol{\theta}$, $\hat{\boldsymbol{\theta}}_M$ is also a solution of the system of p equations:

$$\sum_{i=1}^{n} \psi\left(\frac{r_i}{\sigma}\right) x_{ij} = 0,\ j = 1, \ldots, p$$

where $\psi = \frac{\partial \rho}{\partial z}$. If ρ is convex, $\hat{\boldsymbol{\theta}}_M$ is the unique solution.

The ROBUSTREG procedure solves this system by using iteratively reweighted least squares (IRLS). The weight function $w(x)$ is defined as

$$w(z) = \frac{\psi(z)}{z}$$

The ROBUSTREG procedure provides 10 kinds of weight functions through the WEIGHTFUNCTION= option in the MODEL statement. Each weight function corresponds to a ρ function. For a complete discussion, see the section "Weight Functions" on page 8048. You can specify the scale parameter σ by using the SCALE= option in the PROC ROBUSTREG statement.

If σ is unknown, both $\boldsymbol{\theta}$ and σ are estimated by minimizing the function

$$Q(\boldsymbol{\theta}, \sigma) = \sum_{i=1}^{n} \left[\rho\left(\frac{r_i}{\sigma}\right) + a\right]\sigma,\ a > 0$$

The algorithm proceeds by alternately improving $\hat{\boldsymbol{\theta}}$ in a location step and $\hat{\sigma}$ in a scale step.

For the scale step, the following three methods are available to estimate σ, which you can select by specifying the SCALE= option:

1. (SCALE=HUBER <(D=*d*)>) Compute $\hat{\sigma}$ by the iteration

$$\left(\hat{\sigma}^{(m+1)}\right)^2 = \frac{1}{nh} \sum_{i=1}^{n} \chi_d\left(\frac{r_i}{\hat{\sigma}^{(m)}}\right) \left(\hat{\sigma}^{(m)}\right)^2$$

where

$$\chi_d(x) = \begin{cases} x^2/2 & \text{if } |x| < d \\ d^2/2 & \text{otherwise} \end{cases}$$

is the Huber function and $h = \frac{n-p}{n}\left(d^2 + (1-d^2)\Phi(d) - 0.5 - d\sqrt{2\pi}e^{-\frac{1}{2}d^2}\right)$ is the Huber constant (Huber 1981, p. 179). You can use the D=*d* option to specify *d*. By default, D=2.5.

2. (SCALE=TUKEY <(D=*d*)>) Compute $\hat{\sigma}$ by solving the supplementary equation

$$\frac{1}{n-p}\sum_{i=1}^{n}\chi_d\left(\frac{r_i}{\sigma}\right)=\beta$$

where

$$\chi_d(x)=\begin{cases}\frac{3x^2}{d^2}-\frac{3x^4}{d^4}+\frac{x^6}{d^6} & \text{if } |x|<d\\ 1 & \text{otherwise}\end{cases}$$

Here $\psi=\frac{1}{6}\chi_1'$ is Tukey's bisquare function, and $\beta=\int\chi_d(s)d\Phi(s)$ is the constant such that the solution $\hat{\sigma}$ is asymptotically consistent when $L(\cdot/\sigma)=\Phi(\cdot)$ (Hampel et al. 1986, p. 149). You can use the D=*d* option to specify d. By default, D=2.5.

3. (SCALE=MED) Compute $\hat{\sigma}$ by the iteration

$$\hat{\sigma}^{(m+1)}=\text{median}\left\{|y_i-\mathbf{x}_i'\hat{\boldsymbol{\theta}}^{(m)}|/\beta_0, i=1,\ldots,n\right\}$$

where $\beta_0=\Phi^{-1}(0.75)$ is the constant such that the solution $\hat{\sigma}$ is asymptotically consistent when $L(\cdot/\sigma)=\Phi(\cdot)$ (Hampel et al. 1986, p. 312).

SCALE=MED is the default.

Algorithm

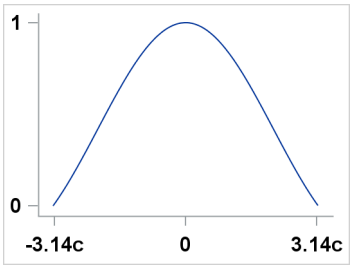
The basic algorithm for computing M estimates for regression is iteratively reweighted least squares (IRLS). As the name suggests, a weighted least squares fit is carried out inside an iteration loop. For each iteration, a set of weights for the observations is used in the least squares fit. The weights are constructed by applying a weight function to the current residuals. Initial weights are based on residuals from an initial fit. The ROBUSTREG procedure uses the unweighted least squares fit as a default initial fit. The iteration terminates when a convergence criterion is satisfied. The maximum number of iterations is set to 1,000. You can specify both the weight function and the convergence criteria.

Weight Functions

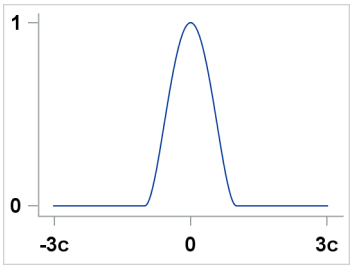
You can specify the weight function for M estimation by using the WEIGHTFUNCTION= option. The ROBUSTREG procedure provides 10 weight functions. By default, the procedure uses the bisquare weight function. In most cases, M estimates are more sensitive to the parameters of these weight functions than to the type of weight function. The median weight function is not stable and is seldom recommended in data analysis; it is included in the ROBUSTREG procedure for completeness. You can specify the parameters for these weight functions. Except for the Hampel and median weight functions, default values for these parameters are defined such that the corresponding M estimates have 95% asymptotic efficiency in the location model with the Gaussian distribution (Holland and Welsch 1977).

The following list shows the weight functions available. See Table 98.5 for the default values of the constants in these weight functions.

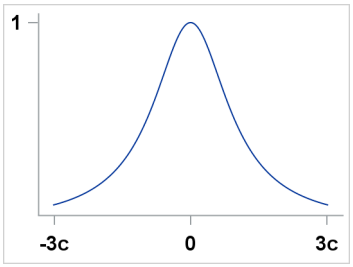
Andrews $$W(x,c) = \begin{cases} \frac{\sin(x/c)}{x/c} & \text{if } |x| \le \pi c \\ 0 & \text{otherwise} \end{cases}$$



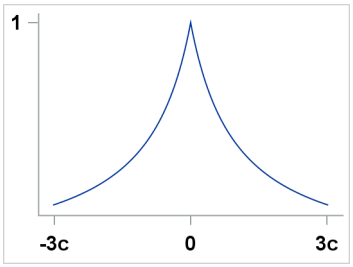
Bisquare $$W(x,c) = \begin{cases} \left(1-(x/c)^2\right)^2 & \text{if } |x| < c \\ 0 & \text{otherwise} \end{cases}$$



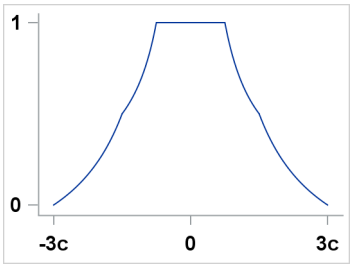
Cauchy $$W(x,c) = \frac{1}{1+(|x|/c)^2}$$



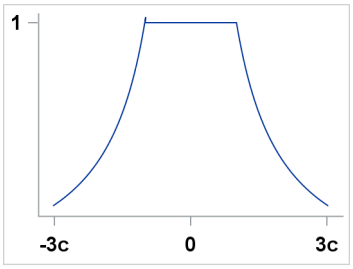
Fair $$W(x,c) = \frac{1}{(1+|x|/c)}$$



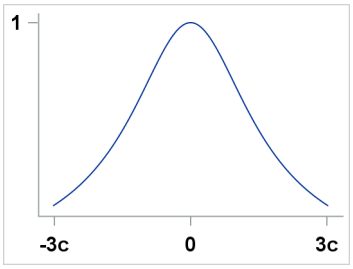
Hampel $$W(x,a,b,c) = \begin{cases} 1 & |x| < a \\ \frac{a}{|x|} & a < |x| \le b \\ \frac{a}{|x|}\frac{c-|x|}{c-b} & b < |x| \le c \\ 0 & \text{otherwise} \end{cases}$$



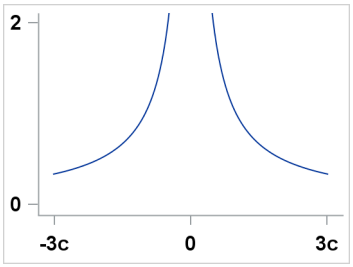
Huber $W(x,c) = \begin{cases} 1 & \text{if } |x| < c \\ \frac{c}{|x|} & \text{otherwise} \end{cases}$



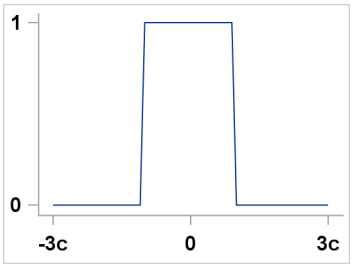
Logistic $W(x,c) = \frac{\tanh(x/c)}{x/c}$



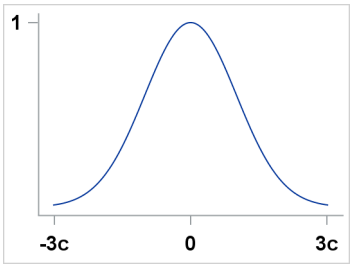
Median $W(x,c) = \begin{cases} \frac{1}{c} & \text{if } x = 0 \\ \frac{1}{|x|} & \text{otherwise} \end{cases}$



Talworth $W(x,c) = \begin{cases} 1 & \text{if } |x| < c \\ 0 & \text{otherwise} \end{cases}$



Welsch $W(x,c) = \exp\left(-\frac{(x/c)^2}{2}\right)$



Convergence Criteria

The following convergence criteria are available in PROC ROBUSTREG:

- relative change in the coefficients (CONVERGENCE=COEF)

- relative change in the scaled residuals (CONVERGENCE=RESID)
- relative change in weights (CONVERGENCE=WEIGHT)

You can specify the criteria by using the CONVERGENCE= option in the PROC ROBUSTREG statement. The default is CONVERGENCE=COEF.

You can specify the precision of the convergence criterion by using the EPS= suboption. The default is EPS=1E–8.

In addition to these convergence criteria, a convergence criterion that is based on a scale-independent measure of the gradient is always checked. For more information, see Coleman et al. (1980). A warning is issued if this additional criterion is not satisfied.

Asymptotic Covariance and Confidence Intervals

The following three estimators of the asymptotic covariance of the robust estimator are available in PROC ROBUSTREG:

$$\text{H1: } K^2 \frac{[1/(n-p)]\sum(\psi(r_i))^2}{[(1/n)\sum(\psi'(r_i))]^2}(\mathbf{X}'\mathbf{X})^{-1}$$

$$\text{H2: } K \frac{[1/(n-p)]\sum(\psi(r_i))^2}{[(1/n)\sum(\psi'(r_i))]}\mathbf{W}^{-1}$$

$$\text{H3: } K^{-1} \frac{1}{(n-p)} \sum(\psi(r_i))^2 \mathbf{W}^{-1}(\mathbf{X}'\mathbf{X})\mathbf{W}^{-1}$$

where $K = 1 + \frac{p}{n}\frac{\text{Var}(\psi')}{(E\psi')^2}$ is a correction factor and $W_{jk} = \sum \psi'(r_i)x_{ij}x_{ik}$. For more information, see Huber (1981, p. 173).

You can specify the asymptotic covariance estimate by using the ASYMPCOV= option. The ROBUSTREG procedure uses H1 as the default because of its simplicity and stability. Confidence intervals are computed from the diagonal elements of the estimated asymptotic covariance matrix.

R Square and Deviance

The robust version of R square is defined as

$$R^2 = \frac{\sum \rho\left(\frac{y_i - \hat{\mu}}{\hat{s}}\right) - \sum \rho\left(\frac{y_i - \mathbf{x}_i'\hat{\boldsymbol{\theta}}}{\hat{s}}\right)}{\sum \rho\left(\frac{y_i - \hat{\mu}}{\hat{s}}\right)}$$

The robust deviance is defined as the optimal value of the objective function on the σ^2 scale,

$$D = 2\hat{s}^2 \sum \rho\left(\frac{y_i - \mathbf{x}_i'\hat{\boldsymbol{\theta}}}{\hat{s}}\right)$$

where $\rho' = \psi$, $\hat{\boldsymbol{\theta}}$ is the M estimator of $\boldsymbol{\theta}$, $\hat{\mu}$ is the M estimator of location, and $\hat{s}$ is the M estimator of the scale parameter in the full model.

Linear Tests

Two tests are available in PROC ROBUSTREG for the canonical linear hypothesis

$$H_0: \quad \theta_j = 0, \quad j = i_1, \ldots, i_q$$

where q is the total number of parameters of the tested effects. The first test is a robust version of the F test, which is referred to as the ρ test. Denote the M estimators in the full and reduced models as $\hat{\boldsymbol{\theta}}(0) \in \Omega_0$ and $\hat{\boldsymbol{\theta}}(1) \in \Omega_1$, respectively. Let

$$\begin{aligned} Q_0 &= Q(\hat{\boldsymbol{\theta}}(0)) = \min\{Q(\boldsymbol{\theta}) | \boldsymbol{\theta} \in \Omega_0\} \\ Q_1 &= Q(\hat{\boldsymbol{\theta}}(1)) = \min\{Q(\boldsymbol{\theta}) | \boldsymbol{\theta} \in \Omega_1\} \end{aligned}$$

with

$$Q = \sum_{i=1}^{n} \rho\left(\frac{r_i}{\sigma}\right)$$

The robust F test is based on the test statistic

$$S_n^2 = \frac{2}{q}[Q_1 - Q_0]$$

Asymptotically $S_n^2 \sim \lambda \chi_q^2$ under H_0, where the standardization factor is $\lambda = \int \psi^2(s) d\Phi(s) / \int \psi'(s)\, d\Phi(s)$ and Φ is the cumulative distribution function of the standard normal distribution. Large values of S_n^2 are significant. This test is a special case of the general τ test of Hampel et al. (1986, Section 7.2).

The second test is a robust version of the Wald test, which is referred to as the R_n^2 test. This test uses a test statistic

$$R_n^2 = n(\hat{\theta}_{i_1}, \ldots, \hat{\theta}_{i_q}) \mathbf{H}_{22}^{-1} (\hat{\theta}_{i_1}, \ldots, \hat{\theta}_{i_q})'$$

where $\frac{1}{n}\mathbf{H}_{22}$ is the $q \times q$ block (corresponding to $\theta_{i_1}, \ldots, \theta_{i_q}$) of the asymptotic covariance matrix of the M estimate $\hat{\boldsymbol{\theta}}_M$ of $\boldsymbol{\theta}$ in a p-parameter linear model.

Under H_0, the statistic R_n^2 has an asymptotic χ^2 distribution with q degrees of freedom. Large values of R_n^2 are significant. For more information, see Hampel et al. (1986, Chapter 7).

Model Selection

When M estimation is used, two criteria are available in PROC ROBUSTREG for model selection. The first criterion is a counterpart of Akaike's (1974) information criterion for robust regression (AICR); it is defined as

$$\text{AICR} = 2 \sum_{i=1}^{n} \rho(r_{i:p}) + \alpha p$$

where $r_{i:p} = (y_i - \mathbf{x}_i' \hat{\boldsymbol{\theta}})/\hat{\sigma}$, $\hat{\sigma}$ is a robust estimate of σ and $\hat{\boldsymbol{\theta}}$ is the M estimator with the p-dimensional design matrix.

As with AIC, α is the weight of the penalty for dimensions. The ROBUSTREG procedure uses $\alpha = 2E\psi^2 / E\psi'$ (Ronchetti 1985) and estimates it by using the final robust residuals.

The second criterion is a robust version of the Schwarz information criteria (BICR); it is defined as

$$\text{BICR} = 2\sum_{i=1}^{n}\rho(r_{i:p}) + p\log(n)$$

High Breakdown Value Estimation

The *breakdown value* of an estimator is the smallest contamination fraction of the data that can cause the estimates on the entire data to be arbitrarily far from the estimates on only the uncontaminated data. The breakdown value of an estimator can be used to measure the robustness of the estimator. Rousseeuw and Leroy (1987) and others introduced the following high breakdown value estimators for linear regression.

LTS Estimate

The least trimmed squares (LTS) estimate that was proposed by Rousseeuw (1984) is defined as the p-vector

$$\hat{\boldsymbol{\theta}}_{LTS} = \arg\min_{\boldsymbol{\theta}} Q_{LTS}(\boldsymbol{\theta}) \text{ with } Q_{LTS}(\boldsymbol{\theta}) = \sum_{i=1}^{h} r^2_{(i)}$$

where $r^2_{(1)} \le r^2_{(2)} \le \ldots \le r^2_{(n)}$ are the ordered squared residuals $r_i^2 = (y_i - \mathbf{x}_i'\boldsymbol{\theta})^2$, $i = 1, \ldots, n$, and h is defined in the range $\frac{n}{2} + 1 \le h \le \frac{3n+p+1}{4}$.

You can specify the parameter h by using the H= option in the PROC ROBUSTREG statement. By default, $h = [\frac{3n+p+1}{4}]$. The breakdown value is $\frac{n-h}{n}$ for the LTS estimate.

The ROBUSTREG procedure computes LTS estimates by using the FAST-LTS algorithm of Rousseeuw and Van Driessen (2000). The estimates are often used to detect outliers in the data, which are then downweighted in the resulting weighted LS regression.

Algorithm

Least trimmed squares (LTS) regression is based on the subset of h observations (out of a total of n observations) whose least squares fit possesses the smallest sum of squared residuals. The coverage h can be set between $\frac{n}{2}$ and n. The LTS method was proposed by Rousseeuw (1984, p. 876) as a highly robust regression estimator with breakdown value $\frac{n-h}{n}$. The ROBUSTREG procedure uses the FAST-LTS algorithm that was proposed by Rousseeuw and Van Driessen (2000). The intercept adjustment technique is also used in this implementation. However, because this adjustment is expensive to compute, it is optional. You can use the IADJUST= option in the PROC ROBUSTREG statement to request or suppress the intercept adjustment. By default, PROC ROBUSTREG does intercept adjustment for data sets that contain fewer than 10,000 observations. The steps of the algorithm are described briefly as follows. For more information, see Rousseeuw and Van Driessen (2000).

1. The default h is $[\frac{3n+p+1}{4}]$, where p is the number of independent variables. You can specify any integer h with $[\frac{n}{2}] + 1 \le h \le [\frac{3n+p+1}{4}]$ by using the H= option in the MODEL statement. The breakdown value for LTS, $\frac{n-h}{n}$, is reported. The default h is a good compromise between breakdown value and statistical efficiency.

2. If $p = 1$ (single regressor), the procedure uses the exact algorithm of Rousseeuw and Leroy (1987, p. 172).

3. If $p \geq 2$, PROC ROBUSTREG uses the following algorithm. If $n < 2$ *ssubs*, where *ssubs* is the size of the subgroups (you can specify *ssubs* by using the SUBGROUPSIZE= option in the PROC ROBUSTREG statement; by default, *ssubs* = 300), PROC ROBUSTREG draws a random p-subset and computes the regression coefficients by using these p points (if the regression is degenerate, another p-subset is drawn). The absolute residuals for all observations in the data set are computed, and the first h points that have the smallest absolute residuals are selected. From this selected h-subset, PROC ROBUSTREG carries out *nsteps* C-steps (concentration steps; for more information, see Rousseeuw and Van Driessen 2000). You can specify *nsteps* by using the CSTEP= option in the PROC ROBUSTREG statement; by default, *nsteps* = 2. PROC ROBUSTREG redraws p-subsets and repeats the preceding computation *nrep* times, and then finds the *nbsol* (at most) solutions that have the lowest sums of h squared residuals. You can specify *nrep* by using the NREP= option in the PROC ROBUSTREG statement; by default, NREP=$\min\{500, \binom{n}{p}\}$. For small n and p, all $\binom{n}{p}$ subsets are used and the NREP= option is ignored (Rousseeuw and Hubert 1996). You can specify *nbsol* by using the NBEST= option in the PROC ROBUSTREG statement; by default, NBEST=10. For each of these *nbsol* best solutions, C-steps are taken until convergence and the best final solution is found.

4. If $n \geq 5ssubs$, construct five disjoint random subgroups with size *ssubs*. If $2ssubs < n < 5ssubs$, the data are split into at most four subgroups with *ssubs* or more observations in each subgroup, so that each observation belongs to a subgroup and the subgroups have roughly the same size. Let *nsubs* denote the number of subgroups. Inside each subgroup, PROC ROBUSTREG repeats the step 3 algorithm *nrep* / *nsubs* times, keeps the *nbsol* best solutions, and pools the subgroups, yielding the merged set of size n_{merged}. In the merged set, for each of the $nsubs \times nbsol$ best solutions, *nsteps* C-steps are carried out by using n_{merged} and $h_{merged} = [n_{merged}\frac{h}{n}]$ and the *nbsol* best solutions are kept. In the full data set, for each of these *nbsol* best solutions, C-steps are taken by using n and h until convergence and the best final solution is found.

NOTE: At step 3 in the algorithm, a randomly selected p-subset might be degenerate (that is, its design matrix might be singular). If the total number of p-subsets from any subgroup is greater than 4,000 and the ratio of degenerate p-subsets is higher than the threshold that is specified in the FAILRATIO= option, the algorithm terminates with an error message.

R Square

For models with the intercept term, the robust version of R square for the LTS estimate is defined as

$$R^2_{LTS} = 1 - \frac{s^2_{LTS}(\mathbf{X}, \mathbf{y})}{s^2_{LTS}(\mathbf{1}, \mathbf{y})}$$

For models without the intercept term, it is defined as

$$R^2_{LTS} = 1 - \frac{s^2_{LTS}(\mathbf{X}, \mathbf{y})}{s^2_{LTS}(\mathbf{0}, \mathbf{y})}$$

For both models,

$$s_{LTS}(\mathbf{X}, \mathbf{y}) = d_{h,n}\sqrt{\frac{1}{h}\sum_{i=1}^{h} r^2_{(i)}}$$

Note that s_{LTS} is a preliminary estimate of the parameter σ in the distribution function $L(\cdot/\sigma)$.

Here $d_{h,n}$ is chosen to make s_{LTS} consistent, assuming a Gaussian model. Specifically,

$$\begin{aligned} d_{h,n} &= 1/\sqrt{1-\frac{2n}{hc_{h,n}}\phi(1/c_{h,n})} \\ c_{h,n} &= 1/\Phi^{-1}\left(\frac{h+n}{2n}\right) \end{aligned}$$

where Φ and ϕ are the distribution function and the density function of the standard normal distribution, respectively.

Final Weighted Scale Estimator

The ROBUSTREG procedure displays two scale estimators, s_{LTS} and Wscale. The estimator Wscale is a more efficient scale estimator based on the preliminary estimate s_{LTS}; it is defined as

$$\text{Wscale} = \sqrt{\frac{\sum_i w_i {r_i}^2}{\sum_i w_i - p}}$$

where

$$w_i = \begin{cases} 0 & \text{if } |r_i|/s_{LTS} > k \\ 1 & \text{otherwise} \end{cases}$$

You can specify k by using the CUTOFF= option in the MODEL statement. By default, $k = 3$.

S Estimate

The S estimate that was proposed by Rousseeuw and Yohai (1984) is defined as the p-vector

$$\hat{\boldsymbol{\theta}}_S = \arg\min_{\theta} S(\boldsymbol{\theta})$$

where the dispersion $S(\boldsymbol{\theta})$ is the solution of

$$\frac{1}{n-p}\sum_{i=1}^{n}\chi\left(\frac{y_i - \mathbf{x}_i'\boldsymbol{\theta}}{S}\right) = \beta$$

Here β is set to $\int \chi(s)d\Phi(s)$ such that $\hat{\boldsymbol{\theta}}_S$ and $S(\hat{\boldsymbol{\theta}}_S)$ are asymptotically consistent estimates of $\boldsymbol{\theta}$ and σ for the Gaussian regression model. The breakdown value of the S estimate is

$$\frac{\beta}{\max_s \chi(s)}$$

The ROBUSTREG procedure provides two choices for χ: Tukey's bisquare function and Yohai's optimal function.

Tukey's bisquare function, which you can specify by using the option CHIF=TUKEY, is

$$\chi_{k_0}(s) = \begin{cases} 3(\frac{s}{k_0})^2 - 3(\frac{s}{k_0})^4 + (\frac{s}{k_0})^6, & \text{if } |s| \le k_0 \\ 1 & \text{otherwise} \end{cases}$$

The constant k_0 controls the breakdown value and efficiency of the S estimate. If you use the EFF= option to specify the efficiency, you can determine the corresponding k_0. The default k_0 is 2.9366, such that the breakdown value of the S estimate is 0.25, with a corresponding asymptotic efficiency for the Gaussian model of 75.9%.

The Yohai function, which you can specify by using the option CHIF=YOHAI, is

$$\chi_{k_0}(s) = \begin{cases} \frac{s^2}{2} & \text{if } |s| \le 2k_0 \\ k_0^2[b_0 + b_1(\frac{s}{k_0})^2 + b_2(\frac{s}{k_0})^4 & \\ +b_3(\frac{s}{k_0})^6 + b_4(\frac{s}{k_0})^8] & \text{if } 2k_0 < |s| \le 3k_0 \\ 3.25k_0^2 & \text{if } |s| > 3k_0 \end{cases}$$

where $b_0 = 1.792$, $b_1 = -0.972$, $b_2 = 0.432$, $b_3 = -0.052$, and $b_4 = 0.002$. If you use the EFF= option to specify the efficiency, you can determine the corresponding k_0. By default, k_0 is set to 0.7405, such that the breakdown value of the S estimate is 0.25, with a corresponding asymptotic efficiency for the Gaussian model of 72.7%.

Algorithm

The ROBUSTREG procedure implements the algorithm that was proposed by Marazzi (1993) for the S estimate, which is a refined version of the algorithm that was proposed by Ruppert (1992). The refined algorithm is briefly described as follows.

Initialize *iter* = 1.

1. Draw a random q-subset of the total n observations, and compute the regression coefficients by using these q observations (if the regression is degenerate, draw another q-subset), where $q \ge p$ can be specified by using the SUBSIZE= option. By default, $q = p$.

2. Compute the residuals: $r_i = y_i - \sum_{j=1}^{p} x_{ij}\theta_j$ for $i = 1, \ldots, n$. For the first iteration, where *iter* = 1, take the following substeps:

   a) If all $|r_i| = 0$ for $i = 1, \ldots, n$, which means y_i exactly equals $\sum_{j=1}^{p} x_{ij}\theta_j$ for all $i = 1, \ldots, n$, this algorithm terminates with a message for exact fit.

   b) Otherwise, set $s^* = 2\text{median}\{|r_i|, i = 1, \ldots, n\}$.

   c) If $s^* = 0$, update $s^* = \min\{|r_i| > 0, i = 1, \ldots, n\}$.

   d) If $\sum_{i=1}^{n} \chi(r_i/s^*) > (n-p)\beta$, set $s^* = 1.5s^*$; go to step 3.

   If *iter* > 1 and $\sum_{i=1}^{n} \chi(r_i/s^*) <= (n-p)\beta$, go to step 3; otherwise, go to step 5.

3. Solve the following equation for s by using an iterative algorithm:

$$\frac{1}{n-p}\sum_{i=1}^{n} \chi(r_i/s) = \beta$$

4. If *iter* > 1 and $s > s^*$, go to step 5. Otherwise, set $s^* = s$ and $\boldsymbol{\theta}^* = \boldsymbol{\theta}$. If $s^* <$ TOLS, return s^* and $\boldsymbol{\theta}^*$; otherwise, go to step 5.

5. If *iter* < NREP, set $iter = iter + 1$ and return to step 1; otherwise, return s^* and $\boldsymbol{\theta}^*$.

The ROBUSTREG procedure performs the following refinement step by default. You can request that this refinement not be performed by specifying the NOREFINE option in the PROC ROBUSTREG statement.

6. Let $\psi = \chi'$. Using the values s^* and $\boldsymbol{\theta}^*$ from the previous steps, compute the M estimates $\boldsymbol{\theta}_M$ and σ_M of $\boldsymbol{\theta}$ and σ with the setup for M estimation that is described in the section "M Estimation" on page 8046. If $\sigma_M > s^*$, give a warning and return s^* and $\boldsymbol{\theta}^*$; otherwise, return σ_M and $\boldsymbol{\theta}_M$.

You can specify TOLS by using the TOLERANCE= option; by default, TOLERANCE=0.001. Alternately, you can specify NREP by using the NREP= option. You can also use the option NREP=NREP0 or NREP=NREP1 to determine NREP according to Table 98.11. NREP=NREP0 is set as the default.

Table 98.11 Default NREP

P	NREP0	NREP1
1	150	500
2	300	1000
3	400	1500
4	500	2000
5	600	2500
6	700	3000
7	850	3000
8	1250	3000
9	1500	3000
>9	1500	3000

NOTE: At step 1 in the algorithm, a randomly selected q-subset might be degenerate. If the total number of q-subsets from any subgroup is greater than 4,000 and the ratio of degenerate q-subsets is higher than the threshold specified in the FAILRATIO= option, the algorithm terminates with an error message.

R Square and Deviance

For the model with the intercept term, the robust version of R square for the S estimate is defined as

$$R_S^2 = 1 - \frac{(n-p)S_p^2}{(n-1)S_\mu^2}$$

For the model without the intercept term, it is defined as

$$R_S^2 = 1 - \frac{(n-p)S_p^2}{nS_0^2}$$

In both cases, S_p is the S estimate of the scale in the full model, S_μ is the S estimate of the scale in the regression model with only the intercept term, and S_0 is the S estimate of the scale without any regressor. The deviance D is defined as the optimal value of the objective function on the σ^2 scale:

$$D = S_p^2$$

Asymptotic Covariance and Confidence Intervals

Because the S estimate satisfies the first-order necessary conditions as the M estimate, it has the same asymptotic covariance as the M estimate. All three estimators of the asymptotic covariance for the M estimate in the section "Asymptotic Covariance and Confidence Intervals" on page 8051 can be used for the S estimate. Besides, the weighted covariance estimator H4 that is described in the section "Asymptotic Covariance and Confidence Intervals" on page 8061 is also available and is set as the default. Confidence intervals for estimated parameters are computed from the diagonal elements of the estimated asymptotic covariance matrix.

MM Estimation

MM estimation is a combination of high breakdown value estimation and efficient estimation that was introduced by Yohai (1987). It has the following three steps:

1. Compute an initial (consistent) high breakdown value estimate $\hat{\boldsymbol{\theta}}'$. The ROBUSTREG procedure provides two kinds of estimates as the initial estimate: the LTS estimate and the S estimate. By default, the LTS estimate is used because of its speed and high breakdown value. The breakdown value of the final MM estimate is decided by the breakdown value of the initial LTS estimate and the constant k_0 in the χ function. To use the S estimate as the initial estimate, specify the INITEST=S option in the PROC ROBUSTREG statement. In this case, the breakdown value of the final MM estimate is decided only by the constant k_0. Instead of computing the LTS estimate or the S estimate as the initial estimate, you can also specify the initial estimate explicitly by using the INEST= option in the PROC ROBUSTREG statement. For more information, see the section "INEST= Data Set" on page 8070.

2. Find $\hat{\sigma}'$ such that

$$\frac{1}{n-p}\sum_{i=1}^{n}\chi\left(\frac{y_i-\mathbf{x}_i'\hat{\boldsymbol{\theta}}'}{\hat{\sigma}'}\right)=\beta$$

   where $\beta = \int \chi(s)d\Phi(s)$.

   The ROBUSTREG procedure provides two choices for χ: Tukey's bisquare function and Yohai's optimal function.

   Tukey's bisquare function, which you can specify by using the option CHIF=TUKEY, is

$$\chi_{k_0}(s)=\begin{cases}3(\frac{s}{k_0})^2-3(\frac{s}{k_0})^4+(\frac{s}{k_0})^6 & \text{if } |s|\le k_0\\ 1 & \text{otherwise}\end{cases}$$

   where k_0 can be specified by using the K0= option. The default k_0 is 2.9366, such that the asymptotically consistent scale estimate $\hat{\sigma}'$ has a breakdown value of 25%.

   Yohai's optimal function, which you can specify by using the option CHIF=YOHAI, is

$$\chi_{k_0}(s)=\begin{cases}\frac{s^2}{2} & \text{if } |s|\le 2k_0\\ k_0^2[b_0+b_1(\frac{s}{k_0})^2+b_2(\frac{s}{k_0})^4 & \\ \quad +b_3(\frac{s}{k_0})^6+b_4(\frac{s}{k_0})^8] & \text{if } 2k_0<|s|\le 3k_0\\ 3.25k_0^2 & \text{if } |s|>3k_0\end{cases}$$

where $b_0 = 1.792$, $b_1 = -0.972$, $b_2 = 0.432$, $b_3 = -0.052$, and $b_4 = 0.002$. You can use the K0= option to specify k_0. The default k_0 is 0.7405, such that the asymptotically consistent scale estimate $\hat{\sigma}'$ has a breakdown value of 25%.

3. Find a local minimum $\hat{\boldsymbol{\theta}}_{MM}$ of

$$Q_{MM} = \sum_{i=1}^{n} \rho\left(\frac{y_i - \mathbf{x}_i'\boldsymbol{\theta}}{\hat{\sigma}'}\right)$$

such that $Q_{MM}(\hat{\boldsymbol{\theta}}_{MM}) \leq Q_{MM}(\hat{\boldsymbol{\theta}}')$. The algorithm for M estimation is used here.

The ROBUSTREG procedure provides two choices for ρ: Tukey's bisquare function and Yohai's optimal function.

Tukey's bisquare function, which you can specify by using the option CHIF=TUKEY, is

$$\rho(s) = \chi_{k_1}(s) = \begin{cases} 3(\frac{s}{k_1})^2 - 3(\frac{s}{k_1})^4 + (\frac{s}{k_1})^6 & \text{if } |s| \leq k_1 \\ 1 & \text{otherwise} \end{cases}$$

where k_1 can be specified by using the K1= option. The default k_1 is 3.440 such that the MM estimate has 85% asymptotic efficiency with the Gaussian distribution.

Yohai's optimal function, which you can specify by using the option CHIF=YOHAI, is

$$\rho(s) = \chi_{k_1}(s) = \begin{cases} \frac{s^2}{2} & \text{if } |s| \leq 2k_1 \\ k_1^2[b_0 + b_1(\frac{s}{k_1})^2 + b_2(\frac{s}{k_1})^4 & \\ \quad +b_3(\frac{s}{k_1})^6 + b_4(\frac{s}{k_1})^8] & \text{if } 2k_1 < |s| \leq 3k_1 \\ 3.25k_1^2 & \text{if } |s| > 3k_1 \end{cases}$$

where k_1 can be specified by using the K1= option. The default k_1 is 0.868 such that the MM estimate has 85% asymptotic efficiency with the Gaussian distribution.

Algorithm

The initial LTS estimate is computed using the algorithm described in the section "LTS Estimate" on page 8053. You can control the quantile of the LTS estimate by specifying the option INITH=h, where h is an integer between $[\frac{n}{2}] + 1$ and $[\frac{3n+p+1}{4}]$. By default, $h = [\frac{3n+p+1}{4}]$, which corresponds to a breakdown value of around 25%.

The initial S estimate is computed using the algorithm described in the section "S Estimate" on page 8055. You can control the breakdown value and efficiency of this initial S estimate by the constant k_0, which you can specify by using the K0= option.

The scale parameter σ is solved by an iterative algorithm

$$(\sigma^{(m+1)})^2 = \frac{1}{(n-p)\beta} \sum_{i=1}^{n} \chi_{k_0}\left(\frac{r_i}{\sigma^{(m)}}\right)\left(\sigma^{(m)}\right)^2$$

where $\beta = \int \chi_{k_0}(s) d\Phi(s)$.

After the scale parameter is computed, the iteratively reweighted least squares (IRLS) algorithm with fixed scale parameter is used to compute the final MM estimate.

Convergence Criteria

In the iterative algorithm for the scale parameter, the relative change of the scale parameter controls the convergence.

In the iteratively reweighted least squares algorithm, the same convergence criteria for the M estimate that are used before are used here.

Bias Test

Although the final MM estimate inherits the high breakdown value property, its bias from the distortion of the outliers can be high. Yohai, Stahel, and Zamar (1991) introduced a bias test. The ROBUSTREG procedure implements this test when you specify the BIASTEST= option in the PROC ROBUSTREG statement. This test is based on the initial scale estimate $\hat{\sigma}'$ and the final scale estimate $\hat{\sigma}'_1$, which is the solution of

$$\frac{1}{n-p}\sum_{i=1}^{n}\chi\left(\frac{y_i-\mathbf{x}_i'\hat{\boldsymbol{\theta}}_{MM}}{\hat{\sigma}'_1}\right)=\beta$$

Let $\psi_{k_0}(z)=\frac{\partial\chi_{k_0}(z)}{\partial z}$ and $\psi_{k_1}(z)=\frac{\partial\chi_{k_1}(z)}{\partial z}$. Compute

$$\begin{aligned}\tilde{r}_i &= (y_i-\mathbf{x}_i'\hat{\boldsymbol{\theta}}')/\hat{\sigma}' \quad \text{for } i=1,\ldots,n\\ v_0 &= \frac{(1/n)\sum\psi'_{k_0}(\tilde{r}_i)}{(\hat{\sigma}'_1/n)\sum\psi_{k_0}(\tilde{r}_i)\tilde{r}_i}\end{aligned}$$

$$\begin{aligned}p_i^{(0)} &= \frac{\psi_{k_0}(\tilde{r}_i)}{(1/n)\sum\psi'_{k_0}(\tilde{r}_i)} \quad \text{for } i=1,\ldots,n\\ p_i^{(1)} &= \frac{\psi_{k_1}(\tilde{r}_i)}{(1/n)\sum\psi'_{k_1}(\tilde{r}_i)} \quad \text{for } i=1,\ldots,n\\ d^2 &= \frac{1}{n}\sum(p_i^{(1)}-p_i^{(0)})^2\end{aligned}$$

Let

$$T=\frac{2n(\hat{\sigma}'_1-\hat{\sigma}')}{v_0 d^2(\hat{\sigma}')^2}$$

Standard asymptotic theory shows that T approximately follows a χ^2 distribution with p degrees of freedom. If T exceeds the α quantile χ^2_α of the χ^2 distribution with p degrees of freedom, then the ROBUSTREG procedure gives a warning and recommends that you use other methods. Otherwise, the final MM estimate and the initial scale estimate are reported. You can specify α by using the ALPHA= option after the BIASTEST= option. By default, ALPHA=0.99.

Asymptotic Covariance and Confidence Intervals

Because the MM estimate is computed as an M estimate with a known scale in the last step, the asymptotic covariance for the M estimate can be used here for the asymptotic covariance of the MM estimate. Besides the three estimators H1, H2, and H3 as described in the section "Asymptotic Covariance and Confidence Intervals" on page 8051, a weighted covariance estimator H4 is available. H4 is calculated as

$$K^2 \frac{[1/(n-p)]\sum(\psi(r_i))^2}{[(1/n)\sum(\psi'(r_i))]^2}\mathbf{W}^{-1}$$

where $K = 1 + \frac{p}{n}\frac{\mathrm{Var}(\psi')}{(E\psi')^2}$ is the correction factor and $W_{jk} = \frac{1}{\bar{w}}\sum w_i x_{ij} x_{ik}$, $\bar{w} = \frac{1}{n}\sum w_i$.

You can specify these estimators by using the option ASYMPCOV=[H1 | H2 | H3 | H4]. The ROBUSTREG procedure uses H4 as the default. Confidence intervals for estimated parameters are computed from the diagonal elements of the estimated asymptotic covariance matrix.

R Square and Deviance

The robust version of R square for the MM estimate is defined as

$$R^2 = \frac{\sum\rho\left(\frac{y_i-\hat{\mu}}{\hat{s}}\right) - \sum\rho\left(\frac{y_i-\mathbf{x}_i'\hat{\boldsymbol{\theta}}}{\hat{s}}\right)}{\sum\rho\left(\frac{y_i-\hat{\mu}}{\hat{s}}\right)}$$

and the robust deviance is defined as the optimal value of the objective function on the σ^2 scale,

$$D = 2\hat{s}^2\sum\rho\left(\frac{y_i - \mathbf{x}_i'\hat{\boldsymbol{\theta}}}{\hat{s}}\right)$$

where $\rho' = \psi$, $\hat{\boldsymbol{\theta}}$ is the MM estimator of $\boldsymbol{\theta}$, $\hat{\mu}$ is the MM estimator of location, and $\hat{s}$ is the MM estimator of the scale parameter in the full model.

Linear Tests

For MM estimation, you can use the same ρ test and R_n^2 test that used for M estimation. For more information, see the section "Linear Tests" on page 8052.

Model Selection

For MM estimation, you can use the same two model selection methods that are used for M estimation. For more information, see the section "Model Selection" on page 8052.

Robust Distance

The ROBUSTREG procedure uses the robust multivariate location and scatter estimates for leverage-point detection. The procedure computes a robust version of the Mahalanobis distance by using a generalized minimum covariance determinant (MCD) method. The original MCD method was proposed by Rousseeuw (1984).

Algorithm

PROC ROBUSTREG implements a generalized MCD algorithm that is based on the fast-MCD algorithm formulated by Rousseeuw and Van Driessen (1999), which is similar to the algorithm for least trimmed squares (LTS).

Mahalanobis Distance versus Robust Distance

The canonical Mahalanobis distance is defined as

$$\mathrm{MD}(\mathbf{x}_i) = [(\mathbf{x}_i - \bar{\mathbf{x}})'\bar{\mathbf{C}}(\mathbf{X})^{-1}(\mathbf{x}_i - \bar{\mathbf{x}})]^{1/2}$$

where $\bar{\mathbf{x}} = \frac{1}{n}\sum_{i=1}^{n}\mathbf{x}_i$ and $\bar{\mathbf{C}}(\mathbf{X}) = \frac{1}{n-1}\sum_{i=1}^{n}(\mathbf{x}_i - \bar{\mathbf{x}})'(\mathbf{x}_i - \bar{\mathbf{x}})$ are the empirical multivariate location and scatter, respectively. Here $\mathbf{x}_i = (x_{i1}, \ldots, x_{ip})'$ excludes the intercept. The relationship between the Mahalanobis distance $\mathrm{MD}(\mathbf{x}_i)$ and the hat matrix $\mathbf{H} = (h_{ij}) = \mathbf{X}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'$ is

$$h_{ii} = \frac{1}{n-1}\mathrm{MD}_i^2 + \frac{1}{n}$$

The canonical robust distance is defined as

$$\mathrm{RD}(\mathbf{x}_i) = [(\mathbf{x}_i - \mathbf{T}(\mathbf{X}))'\mathbf{C}(\mathbf{X})^{-1}(\mathbf{x}_i - \mathbf{T}(\mathbf{X}))]^{1/2}$$

where $\mathbf{T}(\mathbf{X})$ and $\mathbf{C}(\mathbf{X})$ are the robust multivariate location and scatter, respectively, that are obtained by the MCD method.

To achieve robustness, the MCD algorithm estimates the covariance of a multivariate data set mainly through an MCD h-point subset of the data set. This subset has the smallest sample-covariance determinant among all possible h-subsets. Accordingly, the breakdown value for the MCD algorithm equals $\frac{(n-h)}{n}$. This means the MCD estimate is reliable, even if up to $\frac{100(n-h)}{n}\%$ observations in the data set are contaminated.

Low-Dimensional Structure

It is possible that the original data are in p-dimensional space but the h-point subset that yields the minimum covariance determinant lies in a lower-dimensional hyperplane. Applying the canonical MCD algorithm to such a data set would result in a singular covariance problem (called exact fit in Rousseeuw and Van Driessen 1999), such that the relevant robust distances cannot be computed. To deal with the singularity problem and provide further leverage-point analysis, PROC ROBUSTREG implements a generalized MCD algorithm. (For more information, see the section "Generalized MCD Algorithm" on page 8066.) The algorithm distinguishes in-(hyper)plane points from off-(hyper)plane points, and performs MCD leverage-point analysis in the dimension-reduced space by projecting all points onto the hyperplane.

Low-dimensional structure is often induced by classification covariates. Suppose that, in a study that has 25 female subjects and 5 male subjects, *gender* is the only classification effect. If the breakdown setting is greater than $\frac{5}{(25+5)}$, the canonical MCD algorithm fails, and so does the relevant leverage-point analysis. In this case, the MCD *h*-subset would contain only female observations, and the constant gender in the *h*-subset would cause the relevant MCD estimate to be singular. The generalized MCD algorithm solves that problem by identifying all male observations as off-plane leverage points, and then carries out the leverage-point analysis, with all the other covariates being centered separately for female and male groups against their group means.

In general, low-dimensional structure is not necessarily due to classification covariates. Imagine that 80 children are supposed to play on a straight trail (denoted by $y = x$) but that some adventurous children go off the trail. The following statements generate the Children data set and the relevant scatter plot:

```
data Children;
   do i=1 to 80;
      off_trail=ranuni(321)>.9;
      x=rannor(111)*ranuni(321);
      trail_x=(i-40)/80*3;
      trail_y=trail_x;
      if off_trail=1 then y=x-1+rannor(321);
      else y=x;
      output;
   end;
run;

proc sgplot data=Children;
   series   x=trail_x y=trail_y/lineattrs=(color="red" pattern=4);
   scatter  x=x y=y/group=off_trail;
   ellipse  x=x y=y/alpha=.05 lineattrs=(color="green" pattern=34);
run;
```

Figure 98.17 shows the positions of all 80 children, the trail (as a red dashed line), and a contour curve of regular Mahalanobis distance that is centered at the mean position (as a green dotted ellipse). In terms of regular Mahalanobis distance, the associated covariance estimate is not singular, but its relevant leverage-point analysis completely ignores the trail (which is the entity of the low-dimensional structure). The children outside the ellipse are defined as leverage points, but the children off the trail would not be viewed as leverage points unless they had large Mahalanobis distances. As mentioned in Rousseeuw and Van Driessen (1999), the canonical MCD method can find the low-dimensional structure, but it does not provide further robust covariance estimation because the MCD covariance estimate is singular. As an improved version of the canonical MCD method, the generalized MCD method can find the trail, identify the children off the trail as off-plane leverage points, and further execute in-plane leverage analysis. The following statements apply the generalized MCD algorithm to the Children data set:

```
ods graphics on;
proc robustreg data=children plots=ddplot(label=none);
   model i = x y/leverage(mcdinfo opc);
run;
ods graphics off;
```

Figure 98.17 Scatter Plot for Children Data

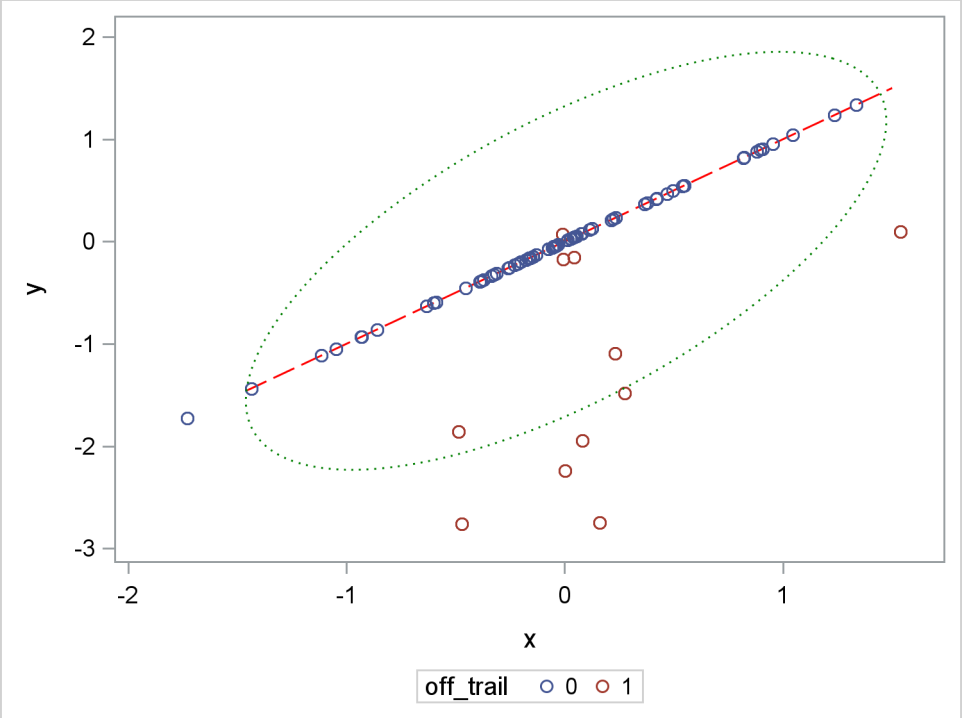


Figure 98.18 exactly identifies the equation that underlies the trail. The analysis projects off-plane points onto the trail and computes their projected robust distances and projected Mahalanobis distances the same way as it does for the in-plane points.

Figure 98.18 Robust Dependence Equations

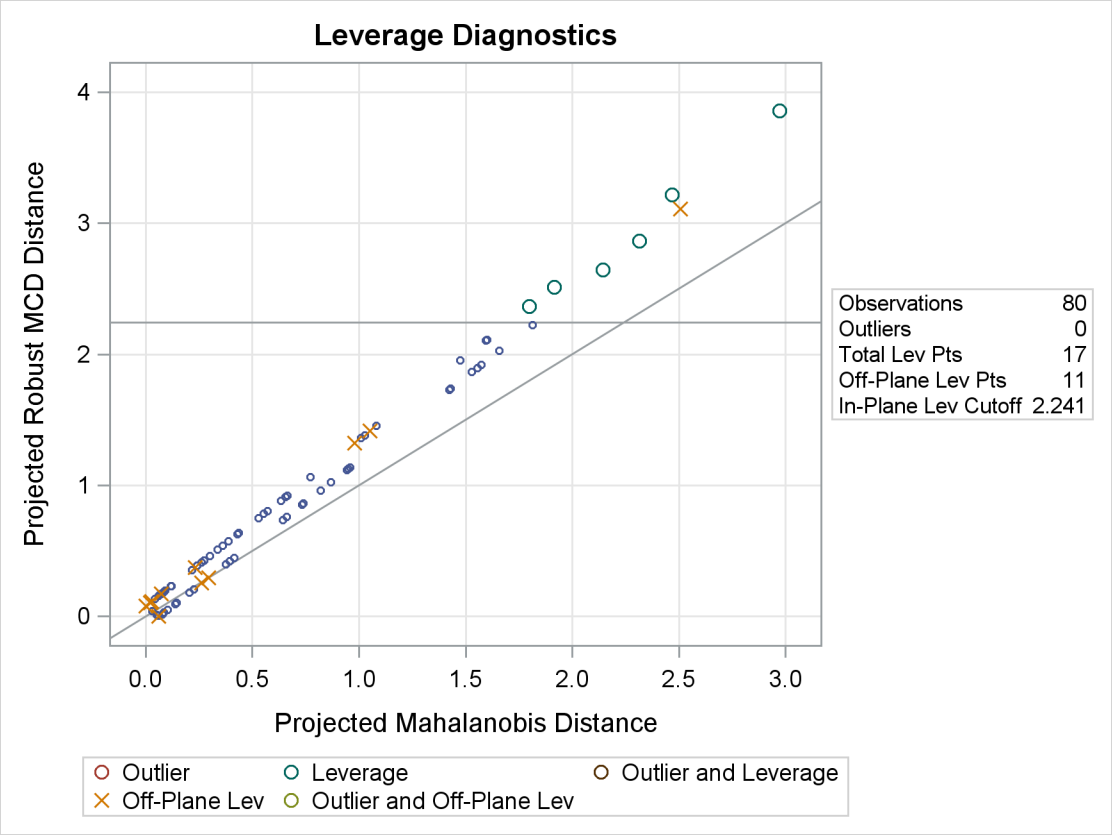
The ROBUSTREG Procedure

Note: The following robust dependence equations simultaneously hold for 86.25% of the observations in the data set. The breakdown setting for the MCD algorithm is 25.00 %.

y = x

Figure 98.19 shows the relevant distance-distance plot. Robust distance is typically greater than Mahalanobis distance because the sample covariance can be strongly influenced by unusual points that cause the sample covariance to be larger than the MCD covariance.

Figure 98.19 DD Plot for Children Data



NOTE: The PROC ROBUSTREG step in this example is used to obtain the leverage diagnostics; the response is not relevant for this analysis.

Through the off-plane and in-plane symbols and the horizontal cutoff line in Figure 98.19, you can separate the children into four groups:

- on the trail and close to the MCD center
- on the trail but far away from the MCD center
- off the trail but close to the MCD center
- off the trail and far away from the MCD center

The children in the latter three groups are defined as leverage points in PROC ROBUSTREG.

Generalized MCD Algorithm

The generalized MCD algorithm follows the same resampling strategy as the canonical MCD algorithm by Rousseeuw and Van Driessen (1999) but with the following modifications:

1. Data are orthonormalized before further processing. The orthonormalized covariates, $\mathbf{x}_i^*$, are defined by $\mathbf{x}_i^* = (\mathbf{x}_i - \bar{\mathbf{x}})\mathbf{P}\mathbf{\Lambda}^{-1/2}$, where $\mathbf{P}$ and $\mathbf{\Lambda}$ are the eigenvector and eigenvalue matrices, respectively, of $\bar{\mathbf{C}}(\mathbf{X})$ (that is, $\bar{\mathbf{C}}(\mathbf{X}) = \mathbf{P}\mathbf{\Lambda}\mathbf{P}'$).

2. Let

$$S_h(\mathbf{X}^*) = \frac{1}{h-1}\sum_{j=1}^{h}(\mathbf{x}_{i_j}^* - \bar{\mathbf{x}}^*)'(\mathbf{x}_{i_j}^* - \bar{\mathbf{x}}^*) = \sum_{j=1}^{p-1}\lambda_j \mathbf{p}_j \mathbf{p}_j{}'$$

denote the covariance and eigendecomposition of a low-dimensional h-subset $\{\mathbf{x}_{i_1}^*, \ldots, \mathbf{x}_{i_h}^*\}$, where $\bar{\mathbf{x}}^* = \frac{1}{h}\sum_{j=1}^{h}\mathbf{x}_{i_j}^*$ and the eigenvalues satisfy

$$\lambda_1 \geq \cdots \geq \lambda_q > 0 = \lambda_{q+1} = \cdots = \lambda_p$$

Then, the rank of $S_h(\mathbf{X}^*)$ equals q, and the pseudo-determinant of $S_h(\mathbf{X}^*)$ is defined as $\prod_{j=1}^{q}\lambda_j$. In finite-precision arithmetic, q is defined as the number of λ's where $\frac{\lambda_i}{\lambda_1}$ is greater than a certain tolerance value. You can specify this tolerance by using the PTOL= suboption of the LEVERAGE option.

3. If $S_h(\mathbf{X}^*)$ and $\bar{\mathbf{x}}^*$ are the covariance and center estimates, respectively, then the projected Mahalanobis distance for $\mathbf{x}_i$ is defined as

$$\left[\sum_{j=1}^{q}\frac{\left((\mathbf{x}_i^* - \bar{\mathbf{x}}^*)\mathbf{p}_j\right)^2}{\lambda_j}\right]^{1/2}$$

The generalized algorithm also computes off-plane distance for each $\mathbf{x}_i$ as

$$\left[\sum_{j=q+1}^{p}\left((\mathbf{x}_i^* - \bar{\mathbf{x}}^*)\mathbf{p}_j\right)^2\right]^{1/2}$$

In finite-precision arithmetic, $\left((\mathbf{x}_i^* - \bar{\mathbf{x}}^*)\mathbf{p}_j\right)^2$ in the previous off-plane formula are truncated to zero if they satisfy

$$\frac{\left((\mathbf{x}_i^* - \bar{\mathbf{x}}^*)\mathbf{p}_j\right)^2}{\lambda_j} \leq \text{cutoff}$$

You can tune this cutoff by using either the PCUTOFF= or PALPHA= suboption of the LEVERAGE option. The points with zero off-plane distances are called in-plane points; otherwise, they are called off-plane points. Analogous to ordering all points in terms of their canonical Mahalanobis distances, for the generalized MCD algorithm the points are first sorted by their off-plane distances, and the points with the same off-plane distance values are further sorted by their projected Mahalanobis distances.

4. Instead of comparing the determinants of h-subset covariance matrices, the generalized algorithm compares both the ranks and pseudo-determinants of the h-subset covariance matrices. If the ranks of two matrices are different, the matrix that has the smaller rank is treated as if its determinant were smaller. If two matrices are of the same rank, they are compared in terms of their pseudo-determinants.

5. Suppose that the $S_h(\mathbf{X}^*)$ of the minimum determinant is singular. Then the relevant low-dimensional structure or hyperplane can be identified by using the eigendecomposition of $S_h(\mathbf{X}^*)$. The eigenvectors that correspond to the nonzero eigenvalues form a basis for the low-dimensional hyperplane. The projected off-plane distance (POD) for $\mathbf{x}_i$ is defined as the off-plane distance that is associated with the $S_h(\mathbf{X}^*)$. To provide further leverage analysis on the low-dimensional hyperplane, every $\mathbf{x}_i^*$ is transformed into $(\mathbf{x}_i^*\mathbf{p}_1, \ldots, \mathbf{x}_i^*\mathbf{p}_q)$, where $\mathbf{p}_j$ are the eigenvectors of the $S_h(\mathbf{X}^*)$. The projected robust distance (PRD) is then computed as the reweighted Mahalanobis distance on all the transformed in-plane points. The off-plane points are assigned zero weights at the reweighting stage, because they are leverage points by definition. The in-plane points are classified into two groups, the normal group and the in-plane leverage group. This classification is made by comparing their projected robust distances with a leverage cutoff value. (For more information, see the section "Leverage-Point and Outlier Detection" on page 8068.) This reweighting process mirrors the one that was proposed by Rousseeuw and Van Driessen (1999). However, the degrees of freedom p for the reweighting critical χ^2 value are replaced by q. You can control the χ^2 critical value by specifying the MCDCUTOFF= or MCDALPHA= option.

If the data set under investigation has a low-dimensional structure, you can use two ODS objects, DependenceEquations and MCDDependenceEquations, to identify the regressors that are linear combinations of other regressors plus certain constants. The equations in DependenceEquations hold for the entire data set, whereas the equations in MCDDependenceEquations apply only to the majority of the observations.

By using the OPC suboption of the LEVERAGE option, you can request an ODS table called DroppedComponents. Figure 98.20 shows the DroppedComponents table for the Children data set example. This table contains a set of coefficient vectors for regressors, which form a basis of the complementary space for the relevant low-dimensional structure.

Figure 98.20 MCD-Dropped Components

Coefficients for MCD-Dropped Components

Parameter	RobustDrop1
x	-1.000
y	1.0000

By using the MCDINFO suboption of the LEVERAGE option, you can request that detailed information about the MCD covariance estimate be displayed in four ODS tables: MCDProfile, MCDCenter, MCDCov, and MCDCorr. Figure 98.21 shows an example of the MCD information tables for the Children data set. The number of dimensions in the table MCDProfile equals the number of nonintercept regressors minus the number of design-dropped components. The specified value of H is the same as h for the h-subset that you can specify by using the QUANTILE= suboption of the LEVERAGE option in the MODEL statement. The reweighted H is the number of observations that are actually used to compute the MCD center and MCD covariance after the reweighting step of the MCD algorithm.

Figure 98.21 MCD Information

MCD Profile	
Number of Dimensions	2
Number of Robust Dropped Components	1
Number of Observations	80
Number of Off-Plane Observations	11
Specified Value of H	60
Reweighted Value of H	63
Breakdown Value	0.2500

MCD Center		
Parameter Name	Parameter	Center
x	x	0.0307
y	y	0.0307

MCD Covariance		
	x	y
x	0.207713	0.207713
y	0.207713	0.207713

MCD Correlation		
	x	y
x	1	1
y	1	1

Leverage-Point and Outlier Detection

The regular variable LEVERAGE is defined as

$$\text{LEVERAGE} = \begin{cases} 0 & \text{if } \mathrm{RD}(\mathbf{x}_i) \le C(p) \\ 1 & \text{otherwise} \end{cases}$$

where $C(p) = \sqrt{\chi^2_{p;1-\alpha}}$ is the cutoff value. $C(p)$ can be set by using the leverage CUTOFF= option, and α can be set by using the leverage CUTOFFALPHA= option.

If projected robust distances are computed for a data set that has a low-dimensional structure, the default cutoff value is $C(q) = \sqrt{\chi^2_{q;1-\alpha}}$, where q is the dimensionality of the low-dimensional space. LEVERAGE is then defined as

$$\text{LEVERAGE} = \begin{cases} 0 & \text{if } \mathrm{POD}(\mathbf{x}_i) = 0 \text{ and } \mathrm{PRD}(\mathbf{x}_i) \le C(q) \\ 1 & \text{if } \mathrm{POD}(\mathbf{x}_i) = 0 \text{ and } \mathrm{PRD}(\mathbf{x}_i) > C(q) \text{ (called in-plane leverage)} \\ 1 & \text{if } \mathrm{POD}(\mathbf{x}_i) > 0 \text{ (called off-plane leverage)} \end{cases}$$

where POD is the projected off-plane distance and PRD denotes the projected robust distance. You can specify a cutoff value by using the CUTOFF= or CUTOFFALPHA= suboption of the LEVERAGE option in the MODEL statement.

Residuals $r_i, i = 1, \ldots, n$, based on robust regression estimates are used to detect vertical outliers. The variable OUTLIER is defined as

$$\text{OUTLIER} = \begin{cases} 0 & \text{if } |r_i| \leq k\hat{\sigma} \\ 1 & \text{otherwise} \end{cases}$$

where $\hat{\sigma}$ is the estimated scale in the model and the multiplier k of the cutoff value is specified by the CUTOFF= option in the MODEL statement. By default, $k = 3$.

An ODS table called Diagnostics contains the LEVERAGE and OUTLIER variables.

Implementation of the WEIGHT Statement

You can use the WEIGHT statement to specify a weight variable in the input data set. (For more information, see the section "WEIGHT Statement" on page 8046.) This section describes how PROC ROBUSTREG implements the WEIGHT statement for each of the estimation methods and for leverage detection.

M Estimation

If you use M estimation with a known scale, then instead of minimizing $Q(\boldsymbol{\theta}) = \sum_{i=1}^{n} \rho(\frac{r_i}{\sigma})$, the weighted M estimation minimizes the weighted Huber-type objective function

$$Q(\boldsymbol{\theta}) = \sum_{i=1}^{n} v_i \rho\left(\frac{r_i}{\sigma}\right)$$

where v_i is the weight variable that is specified by the WEIGHT statement. If you use M estimation with an unknown scale, the weight variable is used in the location steps but not in the scale steps. (For more information, see the section "M Estimation" on page 8046 and the SCALE= option.) For estimating the covariance of the weighted M estimation, $\psi(r_i)$ and $\psi'(r_i)$ are obtained from the final iteration of the weighted M estimation, and $\mathbf{X}'\mathbf{X}$ and $\mathbf{W}$ are replaced, respectively, by $\mathbf{X}'\mathbf{V}\mathbf{X}$ and $W_{jk} = \sum v_i \psi'(r_i) x_{ij} x_{ik}$, where $\mathbf{V}$ is a diagonal matrix whose diagonal elements are v_i. (For more information, see the section "Asymptotic Covariance and Confidence Intervals" on page 8051.) The weight variable does not affect the model degrees of freedom p and the error degrees of freedom $n - p$.

LTS Estimation

LTS estimation ignores the weight variable.

S Estimation

S estimation applies the weight variable only in its M-refinement step. Except for the initial estimates, the M-refinement step of S estimation is the same as the weighted M estimation with unknown scale. If you use the NOREFINE suboption, S estimation ignores the weight variable along with the M-refinement step.

MM Estimation

By default, the initial step of MM estimation is the initial LTS estimation. Unlike the regular LTS estimation, the initial LTS estimation is applied to the weighted data $(y_i{}^*, \mathbf{x}_i{}^*)$'s, where $y_i{}^* = \sqrt{v_i}\, y_i$ and $\mathbf{x}_i{}^* = \sqrt{v_i}\, \mathbf{x}_i$. After the initial LTS estimation, the weight variable is ignored for the subsequent scale adjustment.

You can use INITEST=S to specify the initial S estimation as the initial step of the MM estimation. As with the regular S estimation, the weight variable is used only in the M-refinement step of the initial S estimation. There is no subsequent scale adjustment step if the initial S estimation is applied.

Except for the initial estimates, the final M estimation of the MM estimation is the same as the weighted M estimation with known scale.

Final Weighted Least Squares Estimation

Final weighted least squares estimation is always applied to the weighted data $(y_i^*, \mathbf{x}_i^*)$, no matter how the weight variable is applied in the preceding estimation. For example, if the option METHOD=LTS is specified along with the FWLS option, although the outliers that are identified by LTS estimation do not depend on the weight variable, final weighted least squares estimation applies the weight variable to all the points that are not outliers.

Robust Distances and Leverage Detection

Robust distance computation ignores the weight variable. Because leverage detection depends on robust distance, it also ignores the weight variable.

INEST= Data Set

When you use M or MM estimation, you can use the INEST= data set to specify initial estimates for all the parameters in the model. The INEST= option is ignored if you specify LTS or S estimation by using the METHOD=LTS or METHOD=S option or if you specify the INITEST= option after the METHOD=MM option in the PROC ROBUSTREG statement. The INEST= data set must contain the intercept variable (named Intercept) and all independent variables in the MODEL statement.

If BY processing is used, the INEST= data set should also include the BY variables, and there must be at least one observation in each BY group. If there is more than one observation in a BY group, the first one that is read is used for that BY group.

If the INEST= data set also contains the _TYPE_ variable, only observations that have the _TYPE_ value "PARMS" are used as starting values.

You can specify starting values for the iteratively reweighted least squares algorithm in the INEST= data set. The INEST= data set has the same structure as the OUTEST= data set but is not required to contain all the variables or observations that appear in the OUTEST= data set. One simple use of the INEST= option is to pass the previous OUTEST= data set directly to the next model as an INEST= data set, assuming that the two models have the same parameterization.

OUTEST= Data Set

The OUTEST= data set contains parameter estimates for the model. You can specify a label in the MODEL statement to distinguish between the estimates for different models that the ROBUSTREG procedure uses. If the COVOUT option is specified, the OUTEST= data set also contains the estimated covariance matrix of the parameter estimates. If the ROBUSTREG procedure does not converge, the parameter estimates are set to missing in the OUTEST data set.

The OUTEST= data set contains all variables that are specified in the MODEL statement and the BY statement. For each BY group, the OUTEST= data set has the following:

- one observation that consists of the model's parameter estimates, where the value of the dependent variable is set to –1
- if the COVOUT option is specified, p observations that contain the rows of the estimated covariance matrix. For these observations, the dependent variable contains the parameter estimates for the corresponding row variables.

The following variables are also added to the data set in their display order:

MODEL	is a character variable that contains the label of the MODEL statement, if present. Otherwise, the variable's value is blank.
NAME	is a character variable that contains the name of the dependent variable for the parameter estimates or the name of the row for the covariance matrix estimates.
TYPE	is a character variable that contains the type of the observation: either PARMS for parameter estimates or COV for covariance estimates.
METHOD	is a character variable that contains the type of estimation method: either M estimation, LTS estimation, S estimation, or MM estimation.
STATUS	is a character variable that contains the status of model fitting: either Converged, Warning, or Failed.
INTERCEPT	is a numeric variable that contains the intercept parameter estimates and covariances.
SCALE	is a numeric variable that contains the scale parameter estimates.

Any BY variables that are specified are also added to the OUTEST= data set.

Computational Resources

The algorithms for the various estimation methods need a different amount of memory for working space. Let p be the number of parameters that are estimated, and let n be the number of observations that are used in the model estimation.

For M estimation, the minimum required working space (in bytes) is

$$3n + 2p^2 + 30p$$

If sufficient space is available, the input data set is also kept in memory; otherwise, the input data set is read again to compute the iteratively reweighted least squares estimates, and the execution time of the procedure increases substantially. For each of the reweighted least squares, $O(np^2 + p^3)$ multiplications and additions are required for computing the crossproduct matrix and its inverse. The $O(v)$ notation means that, for large values of the argument, v, $O(v)$ is approximately a constant times v.

Because the iteratively reweighted least squares algorithm converges very quickly (usually within fewer than 20 iterations), the computation of M estimates is fast.

LTS estimation is more expensive in computation. The minimum required working space (in bytes) is

$$np + 12n + 4p^2 + 60p$$

The memory is mainly used to store the current data that the LTS algorithm uses for modeling. The LTS algorithm uses subsampling and spends much of its computing time on resampling and computing estimates for subsamples. Because it resamples if singularity is detected, the LTS algorithm might take more time if the data set has serious singularities.

The MCD algorithm for leverage-point diagnostics is similar to the LTS algorithm.

ODS Table Names

The ROBUSTREG procedure assigns a name to each table that it creates. You can specify these names when using the Output Delivery System (ODS) to select tables and create output data sets. These names are listed in Table 98.12.

Table 98.12 ODS Tables Produced by PROC ROBUSTREG

ODS Table Name	Description	Statement	Option
BestEstimates	Best final estimates for LTS	PROC	SUBANALYSIS
BestSubEstimates	Best estimates for each subgroup	PROC	SUBANALYSIS
BiasTest	Bias test for MM estimation	PROC	BIASTEST
ClassLevels	Classification variable levels	CLASS	Default
CorrB	Parameter estimate correlation matrix	MODEL	CORRB
CovB	Parameter estimate covariance matrix	MODEL	COVB
CStep	C-step for LTS fitting	PROC	SUBANALYSIS
DependenceEquations	Design dependence equations	MODEL	LEVERAGE
Diagnostics	Outlier diagnostics	MODEL	DIAGNOSTICS
DiagSummary	Summary of the outlier diagnostics	MODEL	Default
DroppedComponents	Coefficients for MCD-dropped components	MODEL	LEVERAGE (OPC)
GoodFit	R square, deviance, AIC, and BIC	PROC	METHOD=
InitLTSProfile	Profile for initial LTS estimate	PROC	METHOD=
InitSProfile	Profile for initial S estimate	PROC	METHOD=
IterHistory	Iteration history	PROC	ITPRINT
LTSEstimates	LTS parameter estimates	PROC	METHOD=
LTSLocationScale	Location and scale for LTS	PROC	METHOD=
LTSProfile	Profile for LTS estimator	PROC	METHOD=
LTSRsquare	R square for LTS estimate	PROC	METHOD=
MCDDependenceEquations	Robust dependence equations	MODEL	LEVERAGE
MCDProfile	MCD profile	MODEL	LEVERAGE (MCDINFO)
MCDCenter	MCD center estimate	MODEL	LEVERAGE (MCDINFO)

Table 98.12 (continued)

ODS Table Name	Description	Statement	Option
MCDCov	MCD covariance estimate	MODEL	LEVERAGE (MCDINFO)
MCDCorr	MCD correlation estimate	MODEL	LEVERAGE (MCDINFO)
MMProfile	Profile for MM estimator	PROC	METHOD=
ModelInfo	Model information	MODEL	Default
NObs	Observations summary	PROC	Default
ParameterEstimates	Parameter estimates	MODEL	Default
ParameterEstimatesF	Final weighted LS estimates	PROC	FWLS
ParameterEstimatesR	Reduced parameter estimates	TEST	Default
ParmInfo	Parameter indices	MODEL	Default
SProfile	Profile for S estimator	PROC	METHOD=
Groups	Groups for LTS fitting	PROC	SUBANALYSIS
SummaryStatistics	Summary statistics for model variables	MODEL	Default
Tests	Results for tests	TEST	Default

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 609 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 608 in Chapter 21, "Statistical Graphics Using ODS."

If the model includes a single continuous independent variable, a plot of robust fit against this variable (fit plot) is provided by default. Two plots are particularly useful in revealing outliers and leverage points. The first is a scatter plot of the standardized robust residuals against the robust distances (RD plot). The second is a scatter plot of the robust distances against the classical Mahalanobis distances (DD plot). In addition to these two plots, a histogram and a quantile-quantile plot of the standardized robust residuals are also helpful.

PROC ROBUSTREG assigns a name to each graph that it creates using ODS Graphics. You can use these names to refer to the graphs when using ODS. The graph names and corresponding PLOTS= options are listed in Table 98.13.

Table 98.13 Graphs Produced by PROC ROBUSTREG

ODS Graph Name	Plot Description	Statement	PLOTS= Option
DDPlot	Robust distance versus Mahalanobis distance (or projected robust distance versus projected Mahalanobis distance)	PROC	DDPLOT
FitPlot	Robust fit versus independent variable	PROC	FITPLOT
Histogram	Histogram of standardized robust residuals	PROC	HISTOGRAM
QQPlot	Quantile-quantile plot of standardized robust residuals	PROC	QQPLOT
RDPlot	Standardized robust residual versus robust distance (or projected robust distance)	PROC	RDPLOT

Fit Plot

When the model has a single independent continuous variable (with or without the intercept), the ROBUSTREG procedure automatically creates a plot of robust fit against this independent variable.

The following simple example shows the fit plot. The data, from Rousseeuw and Leroy (1987, Table 3), include the logarithm of surface temperature and the logarithm of light intensity for 47 stars in the direction of the constellation Cygnus.

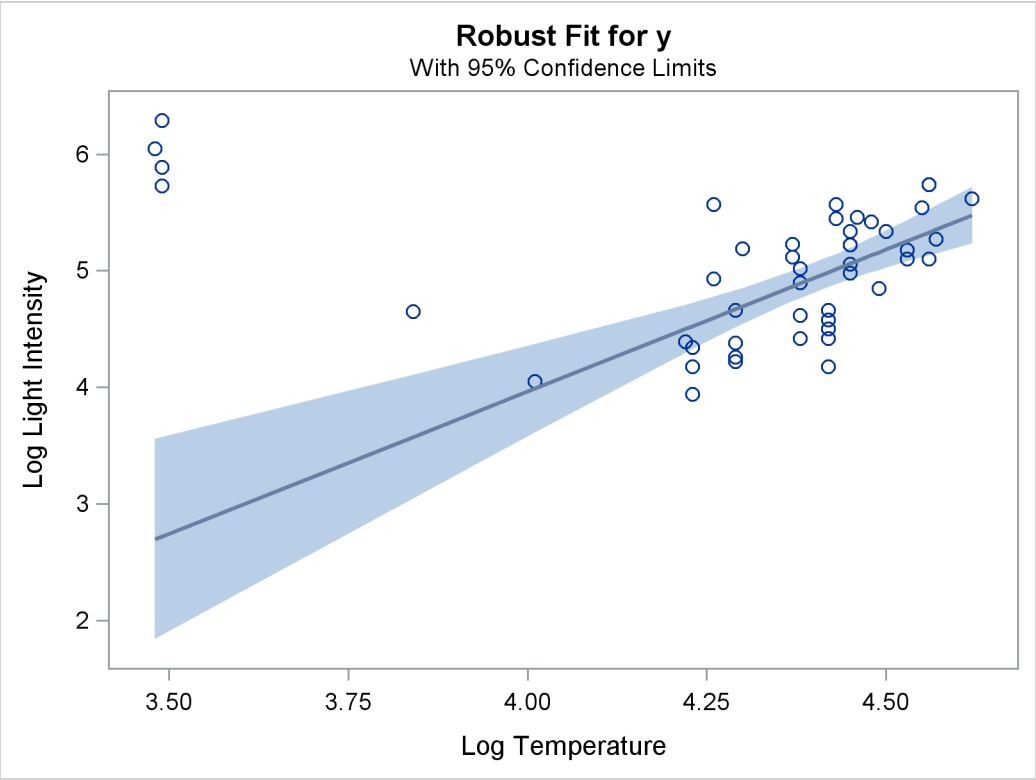
```
data star;
   input index x y @@;
   label x = 'Log Temperature'
         y = 'Log Light Intensity';
   datalines;
 1  4.37  5.23    2  4.56  5.74    3  4.26  4.93    4  4.56  5.74
 5  4.30  5.19    6  4.46  5.46    7  3.84  4.65    8  4.57  5.27
 9  4.26  5.57   10  4.37  5.12   11  3.49  5.73   12  4.43  5.45
13  4.48  5.42   14  4.01  4.05   15  4.29  4.26   16  4.42  4.58
17  4.23  3.94   18  4.42  4.18   19  4.23  4.18   20  3.49  5.89
21  4.29  4.38   22  4.29  4.22   23  4.42  4.42   24  4.49  4.85
25  4.38  5.02   26  4.42  4.66   27  4.29  4.66   28  4.38  4.90
29  4.22  4.39   30  3.48  6.05   31  4.38  4.42   32  4.56  5.10
33  4.45  5.22   34  3.49  6.29   35  4.23  4.34   36  4.62  5.62
37  4.53  5.10   38  4.45  5.22   39  4.53  5.18   40  4.43  5.57
41  4.38  4.62   42  4.45  5.06   43  4.50  5.34   44  4.45  5.34
45  4.55  5.54   46  4.45  4.98   47  4.42  4.50
;
```

The following statements use the MM method to plot the robust fit of the logarithm of light intensity against the logarithm of the surface temperature:

```
ods graphics on;
proc robustreg data=star method=mm;
   model y = x;
run;
```

Figure 98.22 shows the fit plot. Confidence limits are added to the plot by default.

Figure 98.22 Robust Fit



You can suppress the confidence limits by specifying the NOLIMITS option, as shown in the following statements:

```
proc robustreg data=star method=mm plot=fitplot(nolimits);
   model y = x;
run;
```

Distance-Distance Plot

The distance-distance (DD) plot is mainly used for leverage-point diagnostics. It is a scatter plot of the robust distances (or projected robust distances) against the classical Mahalanobis distances (or projected classical Mahalanobis distances) for the independent variables. For more information about the robust distance, see the section "Leverage-Point and Outlier Detection" on page 8068.

You can use the PLOT=DDPLOT option to request this plot. The following statements use the Stack data set in the section "M Estimation" on page 8019 to create the single plot shown in Figure 98.5:

```
proc robustreg data=Stack plot=ddplot;
   model y = x1 x2 x3;
run;
```

The reference lines represent the cutoff values. The diagonal line is also drawn to show the distribution of the distances. By default, all outliers and leverage points are labeled with observation numbers. To change the default, you can use the LABEL= option as described in Table 98.2.

If you specify ID variables in the ID statement, the values of the first ID variable instead of observation numbers are used as labels.

Residual-Distance Plot

The residual-distance (RD) plot is used for both outlier and leverage-point diagnostics. It is a scatter plot of the standardized robust residuals against the robust distances. For more information about the robust distance, see the section "Leverage-Point and Outlier Detection" on page 8068.

You can use the PLOT=RDPLOT option to request the RD plot. The following statements use the Stack data set in the section "M Estimation" on page 8019 to create the plot shown in Figure 98.4:

```
proc robustreg data=Stack plot=rdplot;
   model y = x1 x2 x3;
run;
```

The reference lines represent the cutoff values. By default, all outliers and leverage points are labeled with observation numbers. To change the default, you can use the LABEL= option as described in Table 98.2.

If you specify ID variables in the ID statement, the values of the first ID variable instead of observation numbers are used as labels.

Histogram and Q-Q Plot

PROC ROBUSTREG produces a histogram and a Q-Q plot for the standardized robust residuals. The histogram is superimposed with a normal density curve and a kernel density curve. The following statements use the Stack data set from the section "M Estimation" on page 8019 to create the plots in Figure 98.6 and Figure 98.7:

```
proc robustreg data=Stack plots=(histogram qqplot);
   model y = x1 x2 x3;
run;
```

Examples: ROBUSTREG Procedure

Example 98.1: Comparison of Robust Estimates

This example contrasts several of the robust methods available in the ROBUSTREG procedure.

The following statements generate 1,000 random observations. The first 900 observations are from a linear model, and the last 100 observations are significantly biased in the Y direction. In other words, 10% of the observations are contaminated with outliers.

```
data a (drop=i);
   do i=1 to 1000;
      x1=rannor(1234);
      x2=rannor(1234);
      e=rannor(1234);
      if i > 900 then y=100 + e;
      else y=10 + 5*x1 + 3*x2 + .5 * e;
      output;
   end;
run;
```

The following statements invoke PROC REG and PROC ROBUSTREG with the data set a:

```
proc reg data=a;
   model y = x1 x2;
run;

proc robustreg data=a method=m;
   model y = x1 x2;
run;

proc robustreg data=a method=mm seed=100;
   model y = x1 x2;
run;

proc robustreg data=a method=s seed=100;
   model y = x1 x2;
run;

proc robustreg data=a method=lts seed=100;
   model y = x1 x2;
run;
```

The tables of parameter estimates that are generated by using M estimation, MM estimation, S estimation, and LTS estimation in the ROBUSTREG procedure are shown in Output 98.1.2, Output 98.1.3, Output 98.1.4, and Output 98.1.5, respectively. For comparison, the ordinary least squares (OLS) estimates that are produced by the REG procedure (see Chapter 97, "The REG Procedure") are shown in Output 98.1.1. The four robust methods, M, MM, S, and LTS, correctly estimate the regression coefficients for the underlying model (10, 5, and 3), but the OLS estimate does not.

Output 98.1.1 OLS Estimates for Data with 10% Contamination

The REG Procedure
Model: MODEL1
Dependent Variable: y

Parameter Estimates

Variable	DF	Parameter Estimate	Standard Error	t Value	Pr > |t|
Intercept	1	19.06712	0.86322	22.09	<.0001
x1	1	3.55485	0.86892	4.09	<.0001
x2	1	2.12341	0.83039	2.56	0.0107

Output 98.1.2 M Estimates for Data with 10% Contamination

The ROBUSTREG Procedure

Model Information	
Data Set	WORK.A
Dependent Variable	y
Number of Independent Variables	2
Number of Observations	1000
Method	M Estimation

Parameter Estimates

Parameter	DF	Estimate	Standard Error	95% Confidence Limits		Chi-Square	Pr > ChiSq
Intercept	1	10.0024	0.0174	9.9683	10.0364	331908	<.0001
x1	1	5.0077	0.0175	4.9735	5.0420	82106.9	<.0001
x2	1	3.0161	0.0167	2.9834	3.0488	32612.5	<.0001
Scale	1	0.5780					

Output 98.1.3 MM Estimates for Data with 10% Contamination

The ROBUSTREG Procedure

Model Information	
Data Set	WORK.A
Dependent Variable	y
Number of Independent Variables	2
Number of Observations	1000
Method	MM Estimation

Parameter Estimates

Parameter	DF	Estimate	Standard Error	95% Confidence Limits		Chi-Square	Pr > ChiSq
Intercept	1	10.0035	0.0176	9.9690	10.0379	323947	<.0001
x1	1	5.0085	0.0178	4.9737	5.0433	79600.6	<.0001
x2	1	3.0181	0.0168	2.9851	3.0511	32165.0	<.0001
Scale	0	0.6733					

Output 98.1.4 S Estimates for Data with 10% Contamination

The ROBUSTREG Procedure

Model Information	
Data Set	WORK.A
Dependent Variable	y
Number of Independent Variables	2
Number of Observations	1000
Method	S Estimation

Output 98.1.4 *continued*

Parameter Estimates							
Parameter	DF	Estimate	Standard Error	95% Confidence Limits		Chi-Square	Pr > ChiSq
Intercept	1	10.0055	0.0180	9.9703	10.0408	309917	<.0001
x1	1	5.0096	0.0182	4.9740	5.0452	76045.2	<.0001
x2	1	3.0210	0.0172	2.9873	3.0547	30841.3	<.0001
Scale	0	0.6721					

Output 98.1.5 LTS Estimates for Data with 10% Contamination

The ROBUSTREG Procedure

Model Information	
Data Set	WORK.A
Dependent Variable	y
Number of Independent Variables	2
Number of Observations	1000
Method	LTS Estimation

LTS Parameter Estimates		
Parameter	DF	Estimate
Intercept	1	10.0083
x1	1	5.0316
x2	1	3.0396
Scale (sLTS)	0	0.5880
Scale (Wscale)	0	0.5113

The next statements demonstrate that if the percentage of contamination is increased to 40%, the M method and the MM method with default options fail to estimate the underlying model. Output 98.1.6 and Output 98.1.7 display these estimates. However, by tuning the constant c for the M method and the constants INITH and K0 for the MM method, you can increase the breakdown values of the estimates and capture the right model. Output 98.1.8 and Output 98.1.9 display these estimates. Similarly, you can tune the constant EFF for the S method and the constant H for the LTS method and correctly estimate the underlying model by using these methods. Results are not presented.

```
data b (drop=i);
   do i=1 to 1000;
      x1=rannor(1234);
      x2=rannor(1234);
      e=rannor(1234);
      if i > 600 then y=100 + e;
      else y=10 + 5*x1 +  3*x2 + .5 * e;
      output;
   end;
run;
```

```
proc robustreg data=b method=m;
   model y = x1 x2;
run;

proc robustreg data=b method=mm;
   model y = x1 x2;
run;

proc robustreg data=b method=m(wf=bisquare(c=2));
   model y = x1 x2;
run;

proc robustreg data=b method=mm(inith=502 k0=1.8);
   model y = x1 x2;
run;
```

Output 98.1.6 M Estimates (Default Setting) for Data with 40% Contamination

The ROBUSTREG Procedure

Model Information	
Data Set	WORK.B
Dependent Variable	y
Number of Independent Variables	2
Number of Observations	1000
Method	M Estimation

Parameter Estimates

Parameter	DF	Estimate	Standard Error	95% Confidence Limits		Chi-Square	Pr > ChiSq
Intercept	1	44.8991	1.5609	41.8399	47.9584	827.46	<.0001
x1	1	2.4309	1.5712	-0.6485	5.5104	2.39	0.1218
x2	1	1.3742	1.5015	-1.5687	4.3171	0.84	0.3601
Scale	1	56.6342					

Output 98.1.7 MM Estimates (Default Setting) for Data with 40% Contamination

The ROBUSTREG Procedure

Model Information	
Data Set	WORK.B
Dependent Variable	y
Number of Independent Variables	2
Number of Observations	1000
Method	MM Estimation

Output 98.1.7 *continued*

Parameter Estimates

Parameter	DF	Estimate	Standard Error	95% Confidence Limits		Chi-Square	Pr > ChiSq
Intercept	1	43.0607	1.7978	39.5370	46.5844	573.67	<.0001
x1	1	2.7369	1.8140	-0.8185	6.2924	2.28	0.1314
x2	1	1.5211	1.7265	-1.8628	4.9049	0.78	0.3783
Scale	0	52.8496					

Output 98.1.8 M Estimates (Tuned) for Data with 40% Contamination

The ROBUSTREG Procedure

Model Information	
Data Set	WORK.B
Dependent Variable	y
Number of Independent Variables	2
Number of Observations	1000
Method	M Estimation

Parameter Estimates

Parameter	DF	Estimate	Standard Error	95% Confidence Limits		Chi-Square	Pr > ChiSq
Intercept	1	10.0137	0.0219	9.9708	10.0565	209688	<.0001
x1	1	4.9905	0.0220	4.9473	5.0336	51399.1	<.0001
x2	1	3.0399	0.0210	2.9987	3.0811	20882.4	<.0001
Scale	1	1.0531					

Output 98.1.9 MM Estimates (Tuned) for Data with 40% Contamination

The ROBUSTREG Procedure

Model Information	
Data Set	WORK.B
Dependent Variable	y
Number of Independent Variables	2
Number of Observations	1000
Method	MM Estimation

Parameter Estimates

Parameter	DF	Estimate	Standard Error	95% Confidence Limits		Chi-Square	Pr > ChiSq
Intercept	1	10.0103	0.0213	9.9686	10.0520	221639	<.0001
x1	1	4.9890	0.0218	4.9463	5.0316	52535.9	<.0001
x2	1	3.0363	0.0201	2.9970	3.0756	22895.5	<.0001
Scale	0	1.8992					

When there are bad leverage points, the M method fails to estimate the underlying model no matter what constant c you use. In this case, other methods (LTS, S, and MM) in PROC ROBUSTREG, which are robust to bad leverage points, correctly estimate the underlying model.

The following statements generate and analyze 1,000 observations, 1% of which are bad high-leverage points.

```
data c (drop=i);
   do i=1 to 1000;
      x1=rannor(1234);
      x2=rannor(1234);
      e=rannor(1234);
      if i > 600 then y=100 + e;
      else y=10 + 5*x1 + 3*x2 + .5 * e;
      if i < 11 then x1=200 * rannor(1234);
      if i < 11 then x2=200 * rannor(1234);
      if i < 11 then y= 100*e;
      output;
   end;
run;

proc robustreg data=c method=mm(inith=502 k0=1.8) seed=100;
   model y = x1 x2;
run;

proc robustreg data=c method=s(k0=1.8) seed=100;
   model y = x1 x2;
run;

proc robustreg data=c method=lts(h=502) seed=100;
   model y = x1 x2;
run;
```

Output 98.1.10 displays the MM estimates with initial LTS estimates, Output 98.1.11 displays the S estimates, and Output 98.1.12 displays the LTS estimates.

Output 98.1.10 MM Estimates for Data with 1% Leverage Points

The ROBUSTREG Procedure

Model Information	
Data Set	WORK.C
Dependent Variable	y
Number of Independent Variables	2
Number of Observations	1000
Method	MM Estimation

Parameter Estimates

Parameter	DF	Estimate	Standard Error	95% Confidence Limits		Chi-Square	Pr > ChiSq
Intercept	1	9.9820	0.0215	9.9398	10.0241	215369	<.0001
x1	1	5.0303	0.0206	4.9898	5.0707	59469.1	<.0001
x2	1	3.0222	0.0221	2.9789	3.0655	18744.9	<.0001
Scale	0	2.2134					

Output 98.1.11 S Estimates for Data with 1% Leverage Points

The ROBUSTREG Procedure

Model Information	
Data Set	WORK.C
Dependent Variable	y
Number of Independent Variables	2
Number of Observations	1000
Method	S Estimation

Parameter Estimates

Parameter	DF	Estimate	Standard Error	95% Confidence Limits		Chi-Square	Pr > ChiSq
Intercept	1	9.9808	0.0216	9.9383	10.0232	212532	<.0001
x1	1	5.0303	0.0208	4.9896	5.0710	58656.3	<.0001
x2	1	3.0217	0.0222	2.9782	3.0652	18555.7	<.0001
Scale	0	2.2094					

Output 98.1.12 LTS Estimates for Data with 1% Leverage Points

The ROBUSTREG Procedure

Model Information	
Data Set	WORK.C
Dependent Variable	y
Number of Independent Variables	2
Number of Observations	1000
Method	LTS Estimation

LTS Parameter Estimates

Parameter	DF	Estimate
Intercept	1	9.9742
x1	1	5.0010
x2	1	3.0219
Scale (sLTS)	0	0.9952
Scale (Wscale)	0	0.5216

Example 98.2: Robust ANOVA

The classical analysis of variance (ANOVA) technique that is based on least squares assumes that the underlying experimental errors are normally distributed. However, data often contain outliers as a result of recording or other errors. In other cases, extreme responses occur when control variables in the experiments are set to extremes. It is important to distinguish among these extreme points and determine whether they are outliers or important extreme cases. You can use the ROBUSTREG procedure for robust analysis of variance based on M estimation. Usually there are no high-leverage points in a well-designed experiment, so M estimation is appropriate.

This example shows how to use the ROBUSTREG procedure for robust ANOVA.

An experiment studied the effects of two successive treatments (T1, T2) on the recovery time of mice that had certain diseases. Sixteen mice were randomly assigned to four groups for the four different combinations of the treatments. The recovery times (time) were recorded (in hours) as shown in the following data set:

```
data recover;
   input  T1 $ T2 $ time @@;
   datalines;
0 0 20.2  0 0 23.9  0 0 21.9  0 0 42.4
1 0 27.2  1 0 34.0  1 0 27.4  1 0 28.5
0 1 25.9  0 1 34.5  0 1 25.1  0 1 34.2
1 1 35.0  1 1 33.9  1 1 38.3  1 1 39.9
;
```

The following statements invoke the GLM procedure (see Chapter 46, "The GLM Procedure") for a standard ANOVA:

```
proc glm data=recover;
   class T1 T2;
   model time = T1 T2 T1*T2;
run;
```

Output 98.2.1 indicates that the overall model effect is not significant at the 10% level, and Output 98.2.2 indicates that neither treatment is significant at the 10% level.

Output 98.2.1 Overall ANOVA

The GLM Procedure

Dependent Variable: time

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	3	209.9118750	69.9706250	1.86	0.1905
Error	12	451.9225000	37.6602083		
Corrected Total	15	661.8343750			

R-Square	Coeff Var	Root MSE	time Mean
0.317167	19.94488	6.136791	30.76875

Output 98.2.2 Model ANOVA

Source	DF	Type I SS	Mean Square	F Value	Pr > F
T1	1	81.4506250	81.4506250	2.16	0.1671
T2	1	106.6056250	106.6056250	2.83	0.1183
T1*T2	1	21.8556250	21.8556250	0.58	0.4609

The following statements invoke the ROBUSTREG procedure and use the same model:

```
proc robustreg data=recover;
   class T1 T2;
   model time = T1 T2 T1*T2 / diagnostics;
   T1_T2: test T1*T2;
   output out=robout r=resid sr=stdres;
run;
```

Output 98.2.3 shows some basic information about the model and the response variable time.

Output 98.2.3 Model-Fitting Information and Summary Statistics

The ROBUSTREG Procedure

Model Information	
Data Set	WORK.RECOVER
Dependent Variable	time
Number of Independent Variables	2
Number of Continuous Independent Variables	0
Number of CLASS Independent Variables	2
Number of Observations	16
Method	M Estimation

Summary Statistics

Variable	Q1	Median	Q3	Mean	Standard Deviation	MAD
time	25.5000	31.2000	34.7500	30.7688	6.6425	6.8941

The "Parameter Estimates" table in Output 98.2.4 indicates that the main effects of both treatments are significant at the 5% level.

Output 98.2.4 Model Parameter Estimates

Parameter Estimates

Parameter			DF	Estimate	Standard Error	95% Confidence Limits		Chi-Square	Pr > ChiSq
Intercept			1	36.7655	2.0489	32.7497	40.7814	321.98	<.0001
T1	**0**		1	-6.8307	2.8976	-12.5100	-1.1514	5.56	0.0184
T1	**1**		0	0.0000	.	.	.	.	.
T2	**0**		1	-7.6755	2.8976	-13.3548	-1.9962	7.02	0.0081
T2	**1**		0	0.0000	.	.	.	.	.
T1*T2	**0**	**0**	1	-0.2619	4.0979	-8.2936	7.7698	0.00	0.9490
T1*T2	**0**	**1**	0	0.0000	.	.	.	.	.
T1*T2	**1**	**0**	0	0.0000	.	.	.	.	.
T1*T2	**1**	**1**	0	0.0000	.	.	.	.	.
Scale			1	3.5346					

The reason for the difference between the traditional ANOVA and the robust ANOVA is explained by Output 98.2.5, which shows that the fourth observation is an outlier. Further investigation shows that the original value of 24.4 for the fourth observation was recorded incorrectly.

Output 98.2.5 Diagnostics

Diagnostics

Obs	Standardized Robust Residual	Outlier
4	5.7722	*

Output 98.2.6 displays the robust test results. The interaction between the two treatments is not significant.

Output 98.2.6 Test of Significance

Robust Linear Test T1_T2					
Test	Test Statistic	Lambda	DF	Chi-Square	Pr > ChiSq
Rho	0.0041	0.7977	1	0.01	0.9431
Rn2	0.0041		1	0.00	0.9490

Output 98.2.7 displays the robust residuals and standardized robust residuals.

Output 98.2.7 PROC ROBUSTREG Output

Obs	T1	T2	time	resid	stdres
1	0	0	20.2	-1.7974	-0.50851
2	0	0	23.9	1.9026	0.53827
3	0	0	21.9	-0.0974	-0.02756
4	0	0	42.4	20.4026	5.77222
5	1	0	27.2	-1.8900	-0.53472
6	1	0	34.0	4.9100	1.38911
7	1	0	27.4	-1.6900	-0.47813
8	1	0	28.5	-0.5900	-0.16693
9	0	1	25.9	-4.0348	-1.14152
10	0	1	34.5	4.5652	1.29156
11	0	1	25.1	-4.8348	-1.36785
12	0	1	34.2	4.2652	1.20668
13	1	1	35.0	-1.7655	-0.49950
14	1	1	33.9	-2.8655	-0.81070
15	1	1	38.3	1.5345	0.43413
16	1	1	39.9	3.1345	0.88679

Example 98.3: Growth Study of De Long and Summers

Robust regression and outlier detection techniques have considerable applications to econometrics. This example, from Zaman, Rousseeuw, and Orhan (2001), shows how these techniques substantially improve the ordinary least squares (OLS) results for the growth study of De Long and Summers.

De Long and Summers (1991) studied the national growth of 61 countries from 1960 to 1985 by applying OLS to the following data set:

```
data growth;
   input country $ GDP LFG EQP NEQ GAP @@;
   datalines;
Argentin  0.0089 0.0118 0.0214 0.2286 0.6079
Austria   0.0332 0.0014 0.0991 0.1349 0.5809
Belgium   0.0256 0.0061 0.0684 0.1653 0.4109
Bolivia   0.0124 0.0209 0.0167 0.1133 0.8634
Botswana  0.0676 0.0239 0.1310 0.1490 0.9474

   ... more lines ...

Venezuel  0.0120 0.0378 0.0340 0.0760 0.4974
Zambia   -0.0110 0.0275 0.0702 0.2012 0.8695
Zimbabwe  0.0110 0.0309 0.0843 0.1257 0.8875
;
```

The regression equation that they used is

$$\mathrm{GDP} = \beta_0 + \beta_1 \mathrm{LFG} + \beta_2 \mathrm{GAP} + \beta_3 \mathrm{EQP} + \beta_4 \mathrm{NEQ} + \epsilon$$

where the response variable is the growth in gross domestic product per worker (GDP) and the regressors are labor force growth (LFG), relative GDP gap (GAP), equipment investment (EQP), and nonequipment investment (NEQ).

The following statements invoke the REG procedure (see Chapter 97, "The REG Procedure") for the OLS analysis:

```
proc reg data=growth;
   model GDP  = LFG GAP EQP NEQ;
run;
```

The OLS analysis that is shown in Output 98.3.1 indicates that GAP and EQP have a significant influence on GDP at the 5% level.

Output 98.3.1 OLS Estimates

The REG Procedure
Model: MODEL1
Dependent Variable: GDP

Parameter Estimates

Variable	DF	Parameter Estimate	Standard Error	t Value	Pr > |t|
Intercept	1	-0.01430	0.01028	-1.39	0.1697
LFG	1	-0.02981	0.19838	-0.15	0.8811
GAP	1	0.02026	0.00917	2.21	0.0313
EQP	1	0.26538	0.06529	4.06	0.0002
NEQ	1	0.06236	0.03482	1.79	0.0787

The following statements invoke the ROBUSTREG procedure and use the default M estimation:

```
ods graphics on;

proc robustreg data=growth plots=all;
   model GDP  = LFG GAP EQP NEQ / diagnostics leverage;
   id country;
run;

ods graphics off;
```

Output 98.3.2 displays model information and summary statistics for variables in the model.

Output 98.3.2 Model-Fitting Information and Summary Statistics

The ROBUSTREG Procedure

Model Information	
Data Set	WORK.GROWTH
Dependent Variable	GDP
Number of Independent Variables	4
Number of Observations	61
Method	M Estimation

Summary Statistics

Variable	Q1	Median	Q3	Mean	Standard Deviation	MAD
LFG	0.0118	0.0239	0.0281	0.0211	0.00979	0.00949
GAP	0.5796	0.8015	0.8863	0.7258	0.2181	0.1778
EQP	0.0265	0.0433	0.0720	0.0523	0.0296	0.0325
NEQ	0.0956	0.1356	0.1812	0.1399	0.0570	0.0624
GDP	0.0121	0.0231	0.0310	0.0224	0.0155	0.0150

Output 98.3.3 displays the M estimates. Besides GAP and EQP, the robust analysis also indicates that NEQ is significant. This new finding is explained by Output 98.3.4, which shows that Zambia, the 60th country in the data, is an outlier. Output 98.3.4 also identifies leverage points that are based on the robust MCD distances; however, there are no serious high-leverage points in this data set.

Output 98.3.3 M Estimates

Parameter Estimates

Parameter	DF	Estimate	Standard Error	95% Confidence Limits		Chi-Square	Pr > ChiSq
Intercept	1	-0.0247	0.0097	-0.0437	-0.0058	6.53	0.0106
LFG	1	0.1040	0.1867	-0.2619	0.4699	0.31	0.5775
GAP	1	0.0250	0.0086	0.0080	0.0419	8.36	0.0038
EQP	1	0.2968	0.0614	0.1764	0.4172	23.33	<.0001
NEQ	1	0.0885	0.0328	0.0242	0.1527	7.29	0.0069
Scale	1	0.0099					

Output 98.3.4 Diagnostics

Diagnostics						
Obs	country	Mahalanobis Distance	Robust MCD Distance	Leverage	Standardized Robust Residual	Outlier
1	Argentin	2.6083	4.0639	*	-0.9424	
5	Botswana	3.4351	6.7391	*	1.4200	
8	Canada	3.1876	4.6843	*	-0.1972	
9	Chile	3.6752	5.0599	*	-1.8784	
17	Finland	2.6024	3.8186	*	-1.7971	
23	HongKong	2.1225	3.8238	*	1.7161	
27	Israel	2.6461	5.0336	*	0.0909	
31	Japan	2.9179	4.7140	*	0.0216	
53	Tanzania	2.2600	4.3193	*	-1.8082	
57	U.S.	3.8701	5.4874	*	0.1448	
58	Uruguay	2.5953	3.9671	*	-0.0978	
59	Venezuel	2.9239	4.1663	*	0.3573	
60	Zambia	1.8562	2.7135		-4.9798	*
61	Zimbabwe	1.9634	3.9128	*	-2.5959	

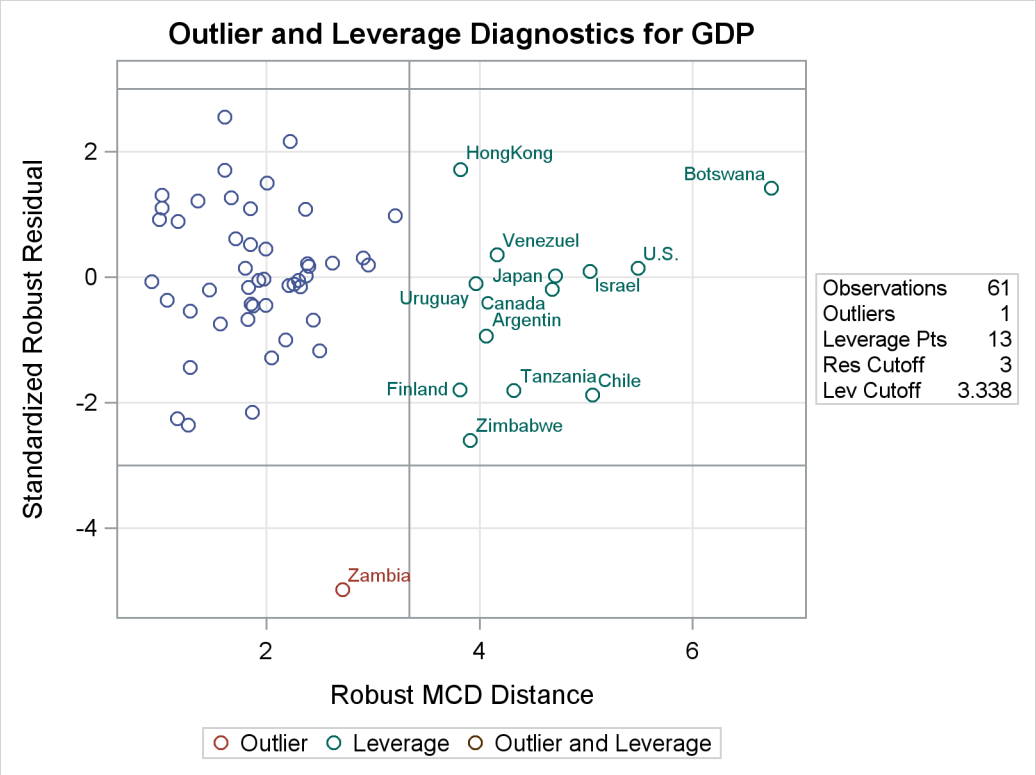
Output 98.3.5 displays robust versions of goodness-of-fit statistics for the model.

Output 98.3.5 Goodness-of-Fit Statistics

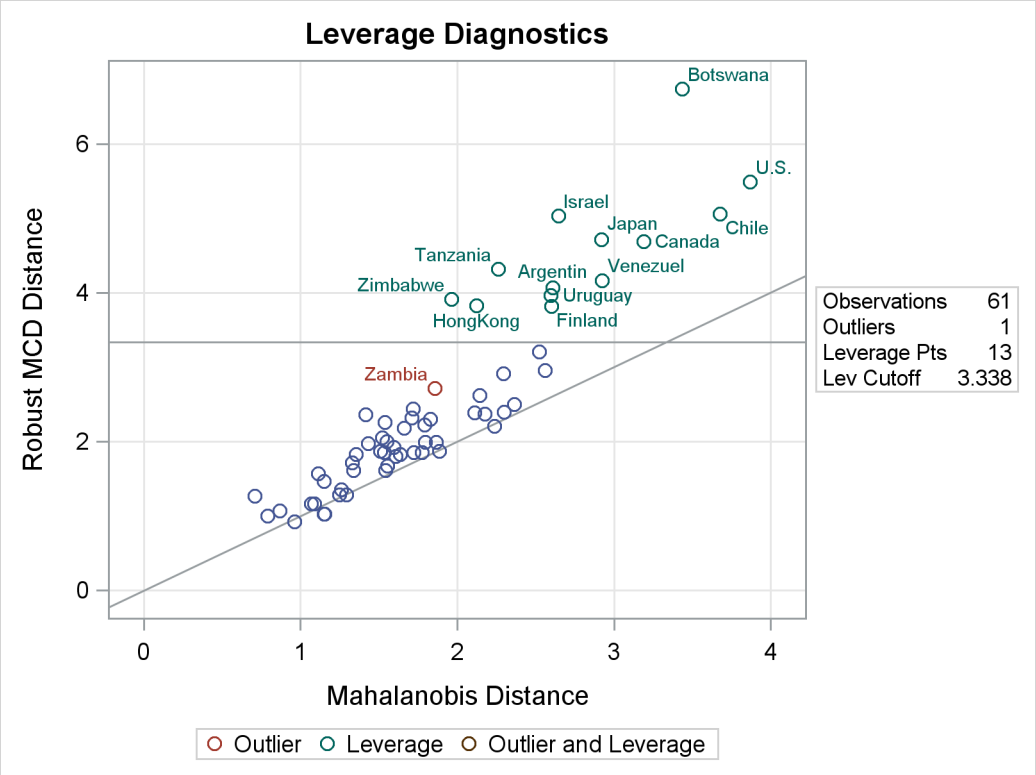
Goodness-of-Fit	
Statistic	Value
R-Square	0.3178
AICR	80.2134
BICR	91.5095
Deviance	0.0070

The PLOTS=ALL option generates four diagnostic plots. Output 98.3.6 and Output 98.3.7 are for outlier and leverage-point diagnostics. Output 98.3.8 and Output 98.3.9 are a histogram and a Q-Q plot, respectively, of the standardized robust residuals.

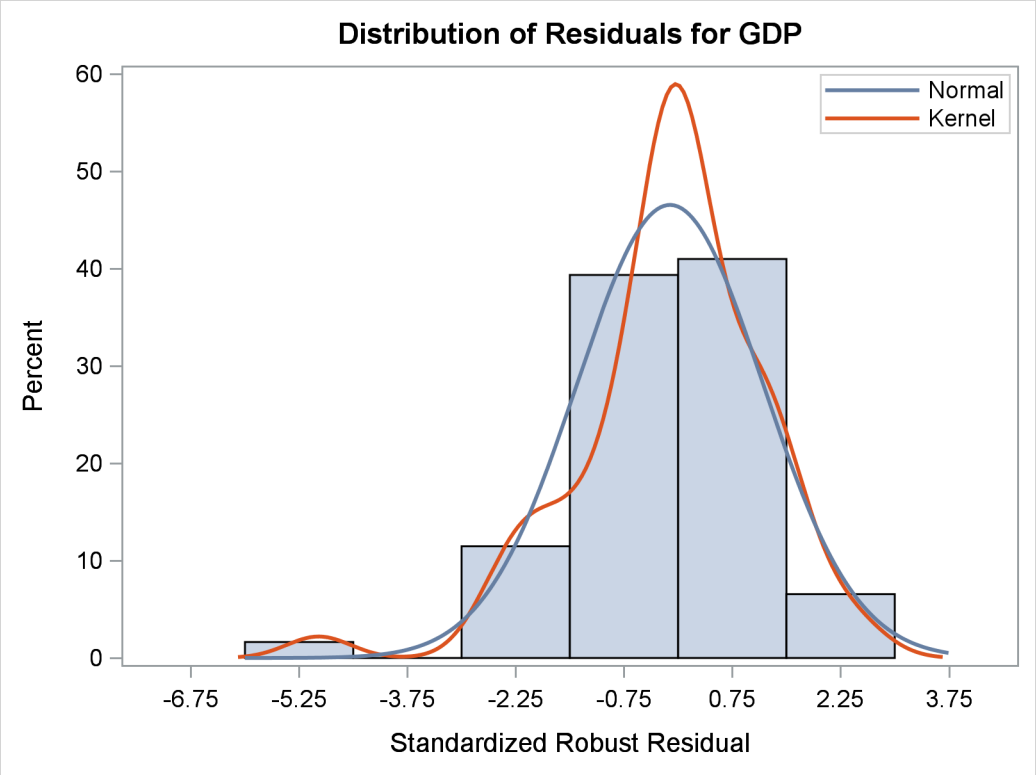
Output 98.3.6 RD Plot for Growth Data



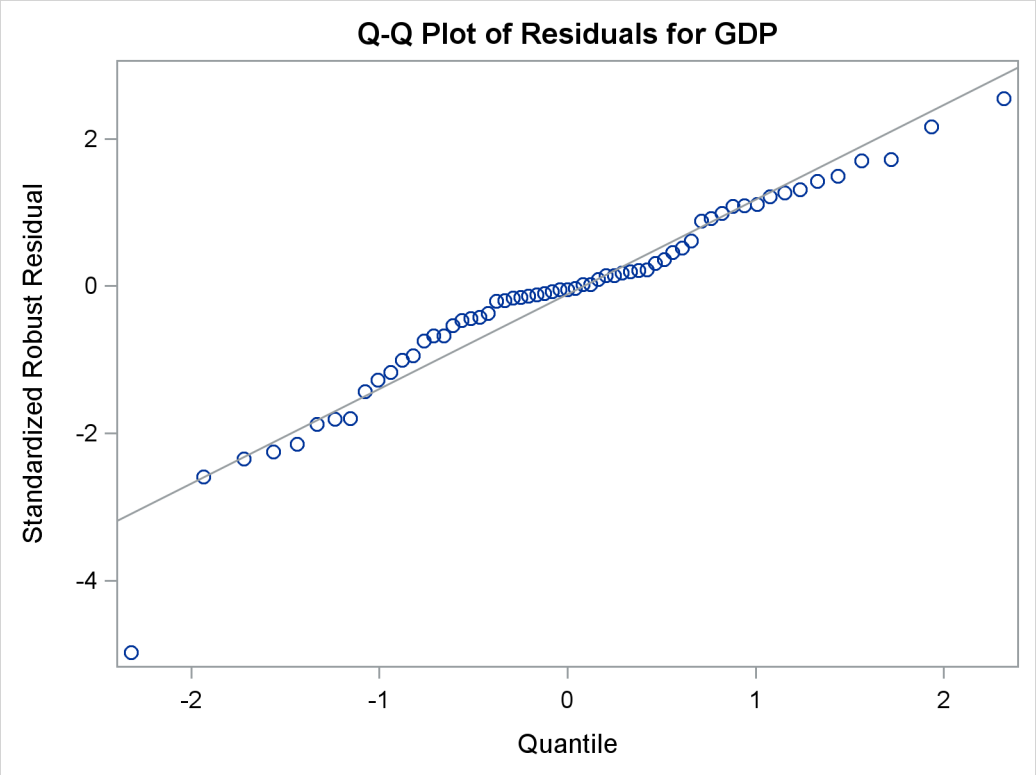
Output 98.3.7 DD Plot for Growth Data



Output 98.3.8 Histogram



Output 98.3.9 Q-Q Plot



The following statements invoke the ROBUSTREG procedure and use LTS estimation, which was used by Zaman, Rousseeuw, and Orhan (2001). The results are consistent with those of M estimation.

```
proc robustreg method=lts(h=33) fwls data=growth seed=100;
   model GDP  = LFG GAP EQP NEQ / diagnostics leverage;
   id country;
run;
```

Output 98.3.10 displays the LTS estimates and the LTS R square.

Output 98.3.10 LTS Estimates and LTS R Square

The ROBUSTREG Procedure

LTS Parameter Estimates

Parameter	DF	Estimate
Intercept	1	-0.0249
LFG	1	0.1123
GAP	1	0.0214
EQP	1	0.2669
NEQ	1	0.1110
Scale (sLTS)	0	0.0076
Scale (Wscale)	0	0.0109

R-Square for LTS Estimation	
R-Square	0.7418

Output 98.3.11 displays outlier and leverage-point diagnostics that are based on the LTS estimates and the robust MCD distances.

Output 98.3.11 Diagnostics

Diagnostics

Obs	country	Mahalanobis Distance	Robust MCD Distance	Leverage	Standardized Robust Residual	Outlier
1	Argentin	2.6083	4.0639	*	-1.0715	
5	Botswana	3.4351	6.7391	*	1.6574	
8	Canada	3.1876	4.6843	*	-0.2324	
9	Chile	3.6752	5.0599	*	-2.0896	
17	Finland	2.6024	3.8186	*	-1.6367	
23	HongKong	2.1225	3.8238	*	1.7570	
27	Israel	2.6461	5.0336	*	0.2334	
31	Japan	2.9179	4.7140	*	0.0971	
53	Tanzania	2.2600	4.3193	*	-1.2978	
57	U.S.	3.8701	5.4874	*	0.0605	
58	Uruguay	2.5953	3.9671	*	-0.0857	
59	Venezuel	2.9239	4.1663	*	0.4113	
60	Zambia	1.8562	2.7135		-4.4984	*
61	Zimbabwe	1.9634	3.9128	*	-2.1201	

Output 98.3.12 displays the final weighted least squares estimates, which are identical to those that are reported in Zaman, Rousseeuw, and Orhan (2001).

Output 98.3.12 Final Weighted LS Estimates

Parameter Estimates for Final Weighted Least Squares Fit

Parameter	DF	Estimate	Standard Error	95% Confidence Limits		Chi-Square	Pr > ChiSq
Intercept	1	-0.0222	0.0093	-0.0405	-0.0039	5.65	0.0175
LFG	1	0.0446	0.1771	-0.3026	0.3917	0.06	0.8013
GAP	1	0.0245	0.0082	0.0084	0.0406	8.89	0.0029
EQP	1	0.2824	0.0581	0.1685	0.3964	23.60	<.0001
NEQ	1	0.0849	0.0314	0.0233	0.1465	7.30	0.0069
Scale	0	0.0116					

Example 98.4: Constructed Effects

The algorithms of PROC ROBUSTREG assume that a response variable is linearly dependent on the regressors. However, in practice, a response often depends on some factors in a nonlinear manner. This example demonstrates how a nonlinear response-factor relationship can be modeled by using constructed effects. (For more information, see the section "EFFECT Statement" on page 401 in Chapter 19, "Shared Concepts and Topics.")

The following data set contains 526 female observations and 474 male observations that are sampled from the 2003 National Health and Nutrition Examination Survey (NHANES). Each observation is composed of three values, which correspond to the variables BMI (body mass index), Age, and Gender, measured for subjects between ages 20 and 60.

```
data BMI;
   input BMI Age Gender $ @@;
   datalines;
46.16 30.33 F 20.67 31.83 F 30.98 51.33 F 30.71 31.42 F
29.81 30.50 M 19.94 25.08 F 29.97 41.67 F 24.48 26.92 F
34.34 51.25 F 20.24 53.67 F 27.72 60.25 F 32.85 41.67 M
22.75 47.50 F 32.78 22.42 F 43.07 29.50 F 38.34 58.50 F
40.03 39.92 F 21.78 56.42 M 28.77 39.83 F 28.77 28.75 F
29.73 54.25 M 33.75 35.67 M 28.48 35.83 M 22.12 29.58 F

   ... more lines ...

26.98 42.50 F 29.44 39.75 M 25.60 52.67 F 19.30 22.00 F
26.53 27.92 F 23.77 29.00 F 29.86 60.58 M 25.41 44.08 M
26.53 24.83 M 33.33 42.08 F 30.52 32.50 F 31.89 38.17 F
32.20 35.92 F 21.73 26.67 M 32.10 39.33 M 25.13 51.75 M
;
```

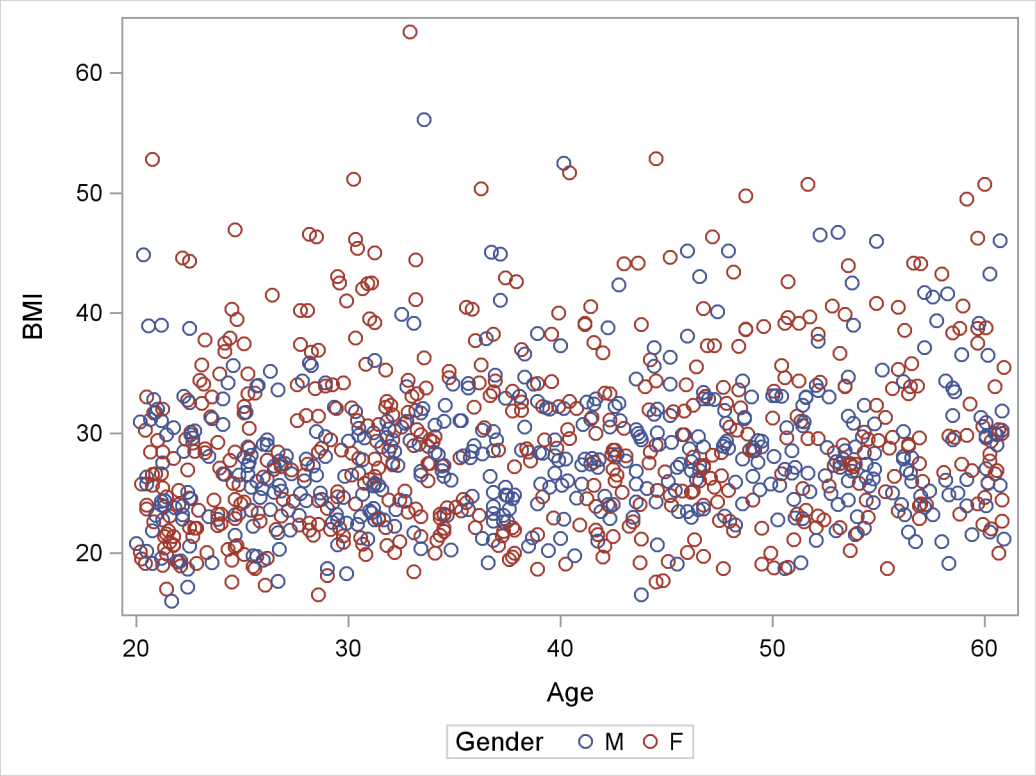
The goal of this analysis is to evaluate whether the BMI by Age curves for women and men are different at a 5% significance level. In order to provide sufficient flexibility to model the effect of Age on BMI, you can use regression splines that you define by using an EFFECT statement. In this example, a regression spline of degree 2 that has three knots is used for the variable Age. The knots are placed at the 25th, 50th, and 75th percentiles of Age. This analysis assume that there is no interaction between Gender and Age, so that the

BMI by Age curves for women and men are the same up to a constant. The following statements produce the BMI by Age scatter plot shown in Output 98.4.1:

```
proc sort data=bmi;
   by age;
run;

proc sgplot data=bmi;
   scatter x=age y=bmi/group=gender;
run;
```

Output 98.4.1 Scatter Plot for BMI Data



The observations that have large BMI values (for example, BMI > 40) are outliers that can substantially influence an ordinary least squares (OLS) analysis. Output 98.4.1 shows that the distributions of BMI conditional on Age are skewed toward the side of large BMI, and that there are more observations that have large BMI values (outliers) in the female group. Hence you can expect a significant Gender difference in the BMI by Age OLS regression analysis. This expectation is confirmed by the OLS Gender p-value 0.0059 in Output 98.4.2, which is produced by the following statements:

```
proc glmselect data=bmi;
   class gender;
   effect Age_Sp=spl(age/degree=2 knotmethod=percentiles(3));
   model bmi = gender age_sp /selection=none showpvalues;
   output out=out_ols P=pred R=res;
run;
```

Output 98.4.2 OLS Estimates

The GLMSELECT Procedure
Least Squares Model (No Selection)

Parameter Estimates

Parameter		DF	Estimate	Standard Error	t Value	Pr > |t|
Intercept		1	29.890089	1.022825	29.22	<.0001
Gender	F	1	1.167332	0.422565	2.76	0.0058
Gender	M	0	0	.	.	.
Age_Sp	1	1	-4.404487	1.473761	-2.99	0.0029
Age_Sp	2	1	-3.329537	1.374096	-2.42	0.0156
Age_Sp	3	1	-0.966875	1.314964	-0.74	0.4623
Age_Sp	4	1	-1.611621	1.123854	-1.43	0.1519
Age_Sp	5	1	-0.484787	1.701281	-0.28	0.7757
Age_Sp	6	0	0	.	.	.

A robust regression method can reduce the outlier influence by automatically assigning smaller or even zero weights to outliers. For the BMI data, a robust regression method is likely to assign less weight to observations that have large BMI, so more female observations than male observations would receive smaller weights. The following statements invoke PROC ROBUSTREG and the BMI data set:

```
proc robustreg data=bmi method=s seed=100;
   class gender;
   effect Age_Sp=spl(age/degree=2 knotmethod=percentiles(3));
   model bmi = gender age_sp;
   output out=out_s P=pred R=res;
run;
```

Output 98.4.3 shows the parameter estimates and the diagnostics summary that are produced by PROC ROBUSTREG and the S method. In contrast to OLS, the robust p-value 0.5581 of the Gender coefficient indicates that the Gender effect is not significant. The outlier diagnostics that are based on the S estimates find 19 outliers that are assigned lower weights by the S method than by the OLS method.

Output 98.4.3 S Estimates and S Diagnostics Summary

The ROBUSTREG Procedure

Parameter Estimates

Parameter		DF	Estimate	Standard Error	95% Confidence Limits		Chi-Square	Pr > ChiSq
Intercept		1	28.2858	1.0081	26.3100	30.2616	787.33	<.0001
Gender	F	1	0.2409	0.4114	-0.5654	1.0473	0.34	0.5581
Gender	M	0	0.0000	.	.	.	.	.
Age_Sp	1	1	-3.8956	1.4376	-6.7133	-1.0779	7.34	0.0067
Age_Sp	2	1	-1.8692	1.3430	-4.5014	0.7630	1.94	0.1640
Age_Sp	3	1	-0.8336	1.2877	-3.3574	1.6903	0.42	0.5174
Age_Sp	4	1	-0.2329	1.1055	-2.3997	1.9338	0.04	0.8331
Age_Sp	5	1	0.0055	1.6632	-3.2543	3.2652	0.00	0.9974
Age_Sp	6	0	0.0000	.	.	.	.	.
Scale		0	6.1715					

Diagnostics Summary

Observation Type	Proportion	Cutoff
Outlier	0.0190	3.0000

To further compare the OLS and S outputs, the following statements plot the BMI predictions in the variable Age for both methods in the same graph, which is shown in Output 98.4.4:

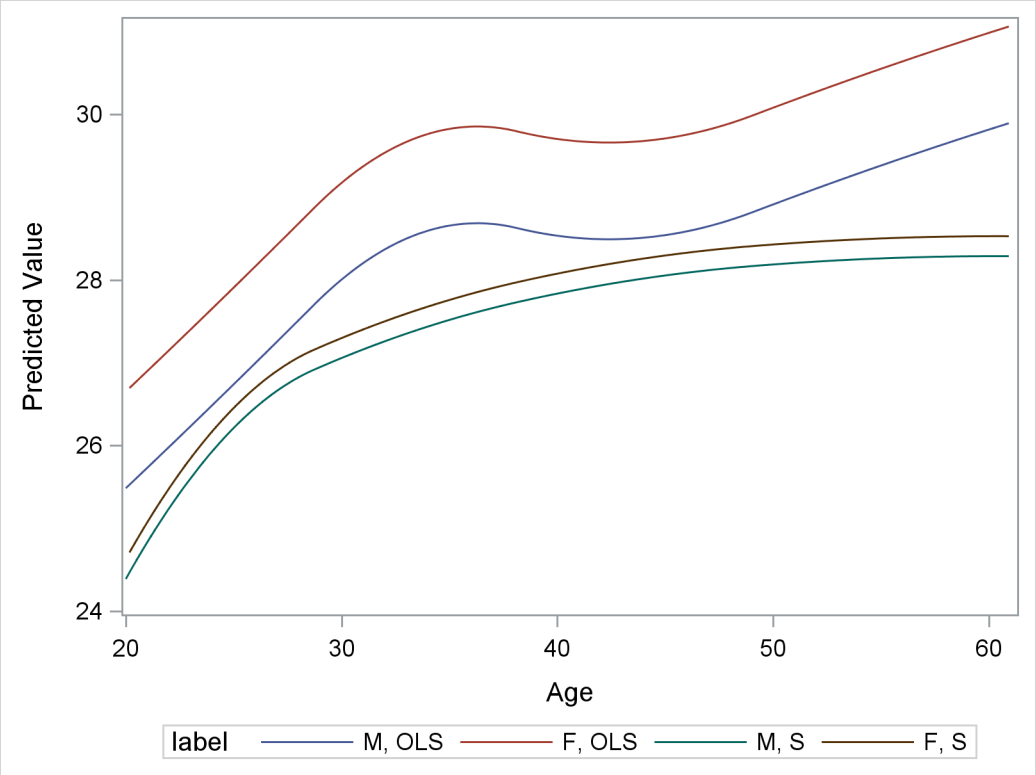
```
data out2_s;
   set out_s;
   if gender="F" then label="F, S  ";
   if gender="M" then label="M, S  ";
run;

data out2_ols;
   merge bmi out_ols;
   if gender='F' then label='F, OLS';
   if gender='M' then label='M, OLS';
   keep pred bmi gender age label;
run;

data out2;
   set out2_ols out2_s;
run;

proc sgplot data=out2;
   series x=age y=pred/group=label;
run;
```

Output 98.4.4 OLS and S Predictions



You can observe the following differences between the OLS and S predictions:

- The OLS prediction is larger.
- The OLS curves have a local maximum near Age = 35.

One question remains: is the significance of the Gender effect for the OLS regression due solely to the outlying observations? To tentatively answer this question, the following statements omit the observations that have the top 10% of BMI values from the original data set and reapply OLS and S methods to the reduced data set:

```
data three;
   set bmi;
   where bmi<38.315;
run;

proc robustreg data=three method=s seed=100;
   class gender;
   effect Age_Sp=spl(age/degree=2 knotmethod=percentiles(3));
   model bmi = gender age_sp;
   output out=out_s P=pred R=res;
run;
```

```
data out2_s;
   set out_s;
   if gender="F" then label="F, S  ";
   if gender="M" then label="M, S  ";
run;

proc glmselect data=three outdesign=four;
   class gender;
   effect Age_Sp=spl(age/degree=2 knotmethod=percentiles(3));
   model bmi = gender age_sp /selection=none showpvalues;
   output out=out_ols P=pred R=res;
run;

data out2_ols;
   merge three out_ols;
   if gender='F' then label='F, OLS';
   if gender='M' then label='M, OLS';
   keep pred bmi gender age label;
run;

data out2;
   set out2_ols out2_s;
run;

proc sgplot data=out2;
   series x=age y=pred/group=label;
run;
ods graphics off;
```

In the reduced data set, 71 female observations and 29 male observations are dropped. Output 98.4.5 and Output 98.4.6 show the refitted S and OLS parameter estimates, respectively.

Output 98.4.5 S Estimates

The ROBUSTREG Procedure

Parameter Estimates

Parameter		DF	Estimate	Standard Error	95% Confidence Limits		Chi-Square	Pr > ChiSq
Intercept		1	27.6427	0.9741	25.7334	29.5520	805.23	<.0001
Gender	F	1	-0.2650	0.4023	-1.0535	0.5234	0.43	0.5100
Gender	M	0	0.0000	.	.	.	.	.
Age_Sp	1	1	-3.1859	1.4032	-5.9361	-0.4356	5.15	0.0232
Age_Sp	2	1	-1.5354	1.3051	-4.0934	1.0226	1.38	0.2394
Age_Sp	3	1	-0.3776	1.2499	-2.8273	2.0721	0.09	0.7626
Age_Sp	4	1	0.3299	1.0668	-1.7610	2.4208	0.10	0.7572
Age_Sp	5	1	0.0949	1.6221	-3.0845	3.2742	0.00	0.9534
Age_Sp	6	0	0.0000	.	.	.	.	.
Scale		0	4.9440					

Output 98.4.6 OLS Estimates

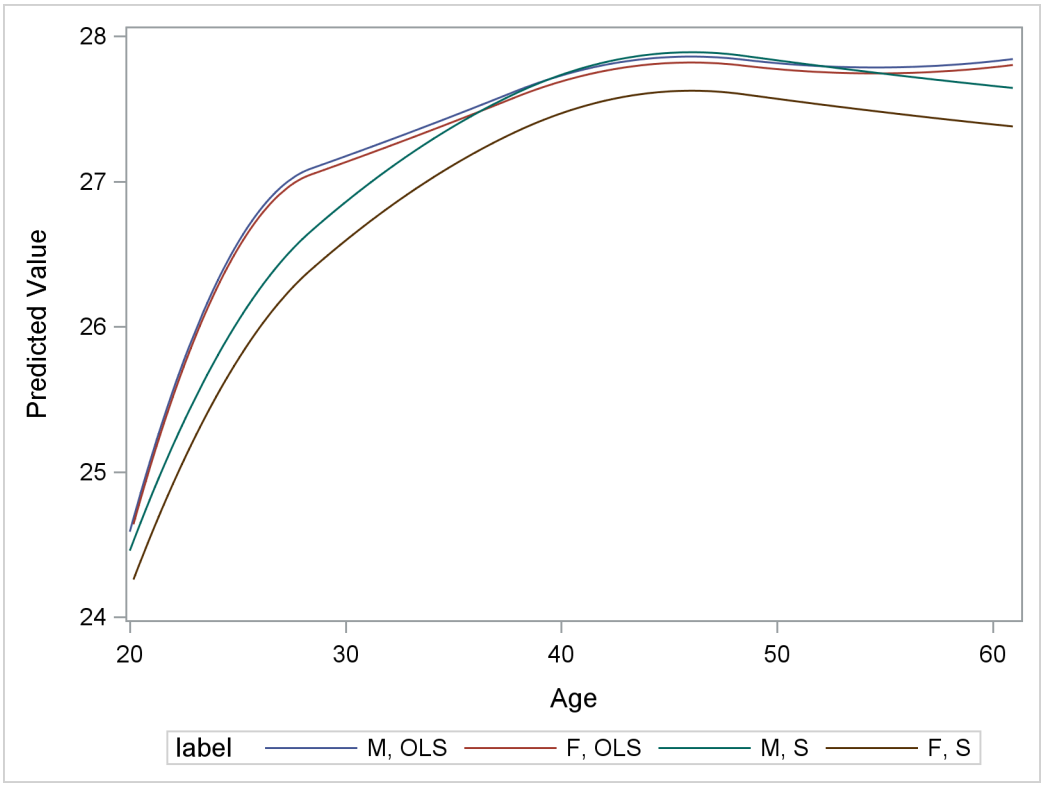
The GLMSELECT Procedure
Least Squares Model (No Selection)

Parameter Estimates

Parameter		DF	Estimate	Standard Error	t Value	Pr > |t|
Intercept		1	27.841568	0.780817	35.66	<.0001
Gender	F	1	-0.040924	0.317749	-0.13	0.8976
Gender	M	0	0	.	.	.
Age_Sp	1	1	-3.253964	1.121292	-2.90	0.0038
Age_Sp	2	1	-0.975273	1.034172	-0.94	0.3459
Age_Sp	3	1	-0.508979	0.999609	-0.51	0.6108
Age_Sp	4	1	0.089393	0.852774	0.10	0.9165
Age_Sp	5	1	-0.113706	1.298157	-0.09	0.9302
Age_Sp	6	0	0	.	.	.

Output 98.4.7 displays the fitted curves on the reduced data set. You can see that Gender is no longer significant for the OLS model, and the OLS turning pattern has also disappeared, but the new S curves do not change much from the previous ones. The OLS BMI by Age curves in Output 98.4.7 are closer to the S curves than to the OLS curves in Output 98.4.4. This suggests that the difference between the OLS and S estimate results is indeed due solely to the influence of the outlying observations.

Output 98.4.7 OLS and S Predictions for the Reduced Data Set



Example 98.5: Robust Diagnostics

This example models the selling price of a house as a function of several covariates. One of these covariates is a classification variable that indicates whether a house is located on a corner lot (called a corner house in this example). Because corner houses are relatively rare, the inclusion of this classification effect in the model introduces a low-dimensional structure (that is, the majority of the observations are located in a lower-dimensional hyperplane that is defined as containing non-corner houses) into the design matrix. As discussed in the section "Robust Distance" on page 8062, the presence of this low-dimensional structure causes difficulties in the traditional computation of robust distances. This example illustrates how you can use the projected robust distance to address those difficulties and to obtain meaningful leverage diagnostics. It also shows how you can use the RDPLOT= and DDPLOT= options to illustrate the outlier-leverage relationship.

The following house price data set contains 66 home resale records on seven variables from February 15 to April 30, 1993 (Data and Story Library 2005). The records are randomly selected from a database that is maintained by the Albuquerque Board of Realtors.

```
data house;
   input price sqft age feats ne cor tax @@;
   label price = "Selling price"
         sqft  = "Square feet of living space"
         age   = "Age of home in year"
         feats = "Number out of 11 features (dishwasher, refrigerator,
                  microwave, disposer, washer, intercom, skylight(s),
                  compactor, dryer, handicap fit, cable TV access)"
         ne    = "Located in northeast sector of city (1) or not (0)"
         cor   = "Corner location (1) or not (0)"
         tax   = "Annual taxes";
   sum = sqft+age+feats+ne+cor+tax;
   id  = _N_;
   datalines;
2050 2650 13 7 1 0 1639
2150 2664  6 5 1 0 1193
2150 2921  3 6 1 0 1635
1999 2580  4 4 1 0 1732

   ... more lines ...

 870 1273  4 4 0 0  638
 869 1165  7 4 0 0  694
 766 1200  7 4 0 1  634
 739  970  4 4 0 1  541
;
```

To illustrate the dependence detection ability of the generalized MCD algorithm, an extra variable called sum is created such that all the observations satisfy sum = sqft + age + feats + ne + cor + tax. Adding the variable sum does not change the rank of the original design matrix; sum is expected to be ignored in the model and also in the diagnostics. The following statements apply the MM method and the generalized MCD algorithm to the house price data:

```
ods graphics on;
proc robustreg data=house method=MM plots=all;
   model price = sqft age feats ne cor tax sum /
         leverage(opc mcdinfo) diagnostics;
run;
```

As shown in Output 98.5.1 and Output 98.5.2, PROC ROBUSTREG finds the design dependence equation and forces the parameter estimate of variable sum to be 0.

Output 98.5.1 MM Estimates

The ROBUSTREG Procedure

Parameter Estimates

Parameter	DF	Estimate	Standard Error	95% Confidence Limits		Chi-Square	Pr > ChiSq
Intercept	1	46.4062	79.1714	-108.767	201.5792	0.34	0.5578
sqft	1	0.3809	0.0756	0.2327	0.5291	25.37	<.0001
age	1	-2.6067	1.7610	-6.0582	0.8449	2.19	0.1388
feats	1	8.3627	14.7107	-20.4697	37.1951	0.32	0.5697
ne	1	65.0081	40.1329	-13.6508	143.6671	2.62	0.1053
cor	1	-19.2997	38.1907	-94.1520	55.5526	0.26	0.6133
tax	1	0.4699	0.1260	0.2229	0.7170	13.90	0.0002
sum	0	0.0000	.	.	.	.	.
Scale	0	157.5593					

Output 98.5.2 Design Dependence Equations

Note: The following variables have been ignored in the MCD computation because of linear dependence.

sum = sqft + age + feats + ne + cor + tax

Moreover, PROC ROBUSTREG also identifies a robust dependence equation on cor in Output 98.5.3, which holds for 77.27% of the observations but not for the entire data set.

Output 98.5.3 Robust Dependence Equations

Note: The following robust dependence equations simultaneously hold for 77.27% of the observations in the data set. The breakdown setting for the MCD algorithm is 22.73 %.

cor = 0

Another way to represent the low-dimensional structure is to specify the coefficients of the MCD-dropped components on the data (see Output 98.5.4), which form a basis of the complementary space to the relevant low-dimensional hyperplane.

Output 98.5.4 Coefficients of MCD-Dropped Components

Coefficients for MCD-Dropped Components		
Parameter	**DesignDrop0**	**RobustDrop1**
sqft	0	0
age	0	0
feats	0	0
ne	0	0
cor	0	1.0000
tax	0	0
sum	1.0000	0

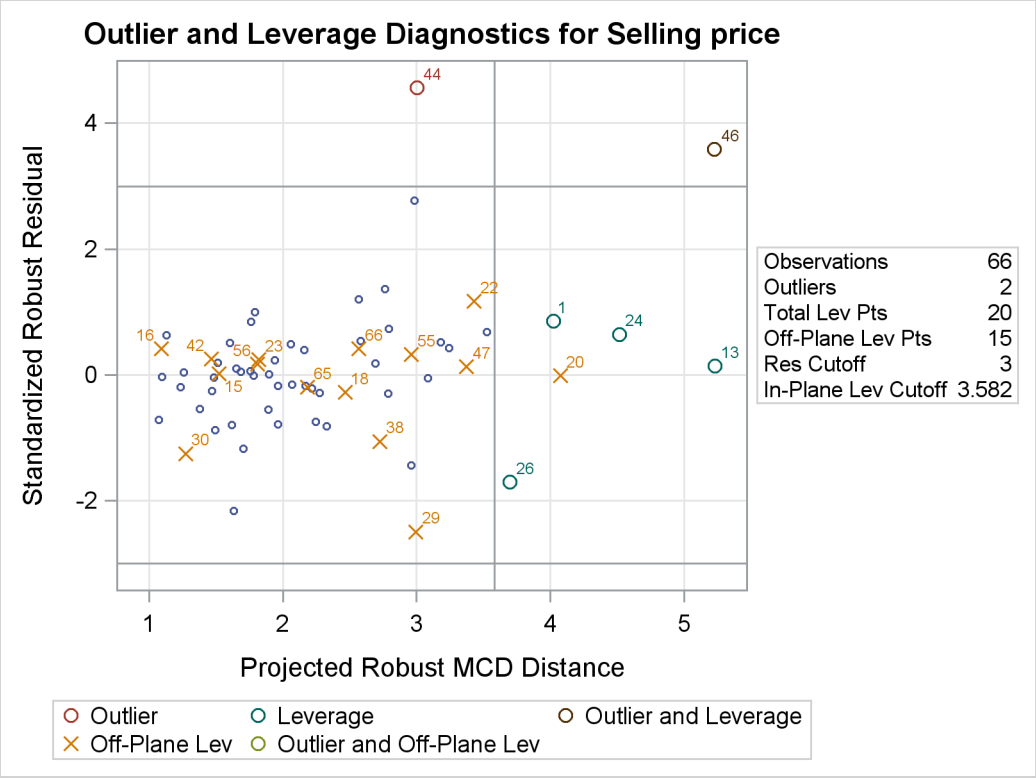
By the definitions of projected robust distance and leverage point, an observation is called an off-plane leverage point if at least one of the robust or design dependence equations does not apply to the observation. In this example, the observations in which cor = 1 are all off-plane leverage points. Output 98.5.5 lists the leverage points and outliers along with the relevant distance measurements and standardized residuals.

Output 98.5.5 Diagnostics

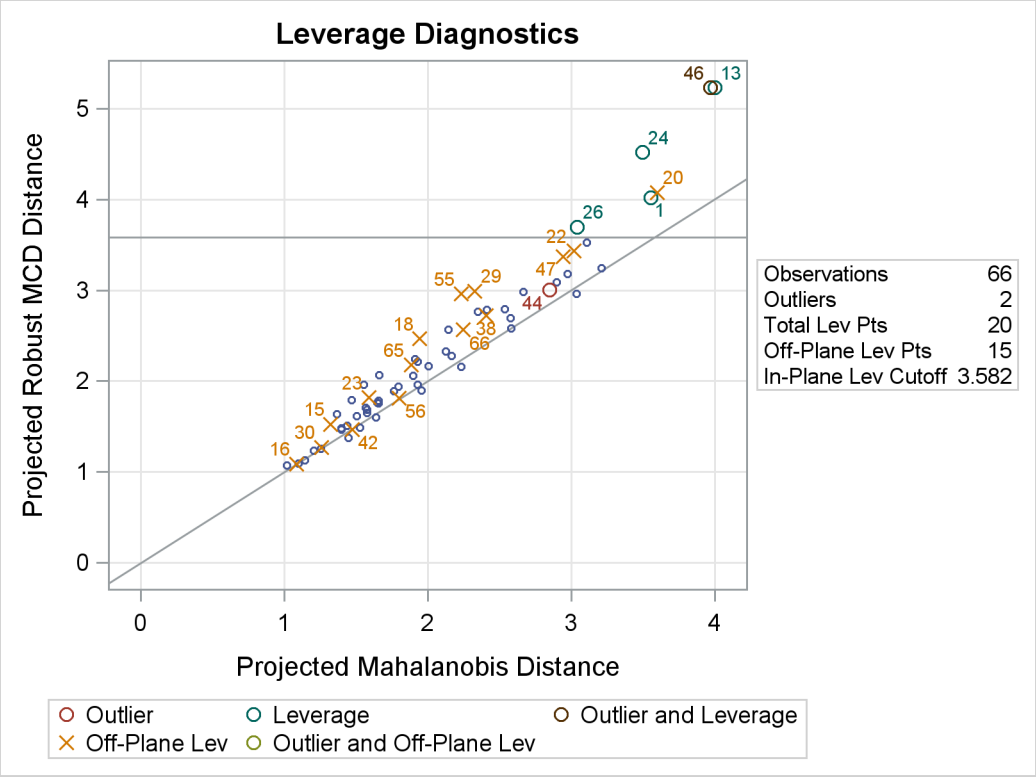
Diagnostics						
	Projected Distance					
Obs	**Mahalanobis**	**Robust**	**Off-Plane**	**Leverage**	**Standardized Robust Residual**	**Outlier**
1	3.5567	4.0211	0.0000	*	0.8522	
13	4.0034	5.2310	0.0000	*	0.1411	
15	1.3221	1.5219	2.3681	*	0.0226	
16	1.0839	1.0905	2.3681	*	0.4148	
18	1.9452	2.4655	2.3681	*	-0.2789	
20	3.6006	4.0771	2.3681	*	-0.0150	
22	3.0210	3.4307	2.3681	*	1.1664	
23	1.5920	1.8197	2.3681	*	0.2422	
24	3.4967	4.5154	0.0000	*	0.6464	
26	3.0420	3.6975	0.0000	*	-1.7068	
29	2.3264	2.9925	2.3681	*	-2.4980	
30	1.2587	1.2714	2.3681	*	-1.2558	
38	2.4064	2.7249	2.3681	*	-1.0620	
42	1.4722	1.4645	2.3681	*	0.2584	
44	2.8491	3.0019	0.0000		4.5665	*
46	3.9725	5.2271	0.0000	*	3.5835	*
47	2.9431	3.3728	2.3681	*	0.1365	
55	2.2325	2.9590	2.3681	*	0.3217	
56	1.7999	1.8119	2.3681	*	0.1715	
65	1.8831	2.1822	2.3681	*	-0.1990	
66	2.2483	2.5673	2.3681	*	0.4134	

From Output 98.5.6 and Output 98.5.7, you can see that there is no apparent corner-related difference for the houses in terms of standardized robust residual and projected MD versus projected RD, although all the corner houses are defined as off-plane leverage points.

Output 98.5.6 Projected RD Plot



Output 98.5.7 Projected DD Plot



Output 98.5.8 shows more details of the robust diagnostics. The number of dimensions indicates that six regressors are used in the MCD analysis. Because sum is excluded in model fitting, it is ignored in the MCD analysis. The number of robust dropped components equals 1 because cor = 1. The number of off-plane points implies the 15 corner-house observations. The reweighted value of H is the number of observations that are finally used to estimate the MCD covariance.

Output 98.5.8 MCD Information

MCD Profile	
Number of Dimensions	6
Number of Robust Dropped Components	1
Number of Observations	66
Number of Off-Plane Observations	15
Specified Value of H	51
Reweighted Value of H	47
Breakdown Value	0.2273

Output 98.5.8 *continued*

MCD Center		
Parameter Name	Parameter	Center
sqft	sqft	1752.7
age	age	12.809
feats	feats	4.0426
ne	ne	0.6170
cor	cor	-2E-16
tax	tax	895.40
sum	sum	2665.6

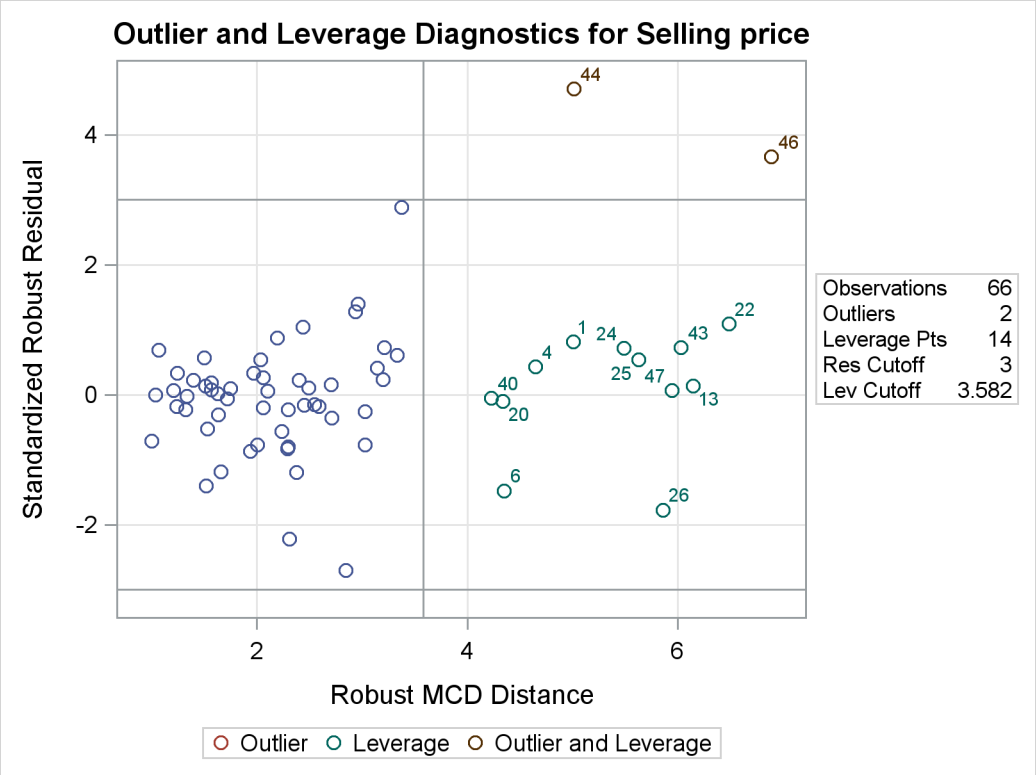
MCD Covariance							
	sqft	age	feats	ne	cor	tax	sum
sqft	248870.3	-853.232	147.0347	88.60083	0	148494.5	396747.3
age	-853.232	126.2886	-1.18733	1.229417	0	-1251.44	-1978.34
feats	147.0347	-1.18733	0.99815	0.234043	0	87.0259	361.5814
ne	88.60083	1.229417	0.234043	0.241443	0	45.76688	134.42
cor	0	0	0	0	0	0	0
tax	148494.5	-1251.44	87.0259	45.76688	0	106652.5	255147
sum	396747.3	-1978.34	361.5814	134.42	0	255147	650413.7

MCD Correlation							
	sqft	age	feats	ne	cor	tax	sum
sqft	1	-0.15219	0.295009	0.361446	0	0.911462	0.986126
age	-0.15219	1	-0.10575	0.222643	0	-0.34099	-0.21829
feats	0.295009	-0.10575	1	0.476749	0	0.266726	0.448759
ne	0.361446	0.222643	0.476749	1	0	0.285206	0.339204
cor	0	0	0	0	0	0	0
tax	0.911462	-0.34099	0.266726	0.285206	0	1	0.968747
sum	0.986126	-0.21829	0.448759	0.339204	0	0.968747	1

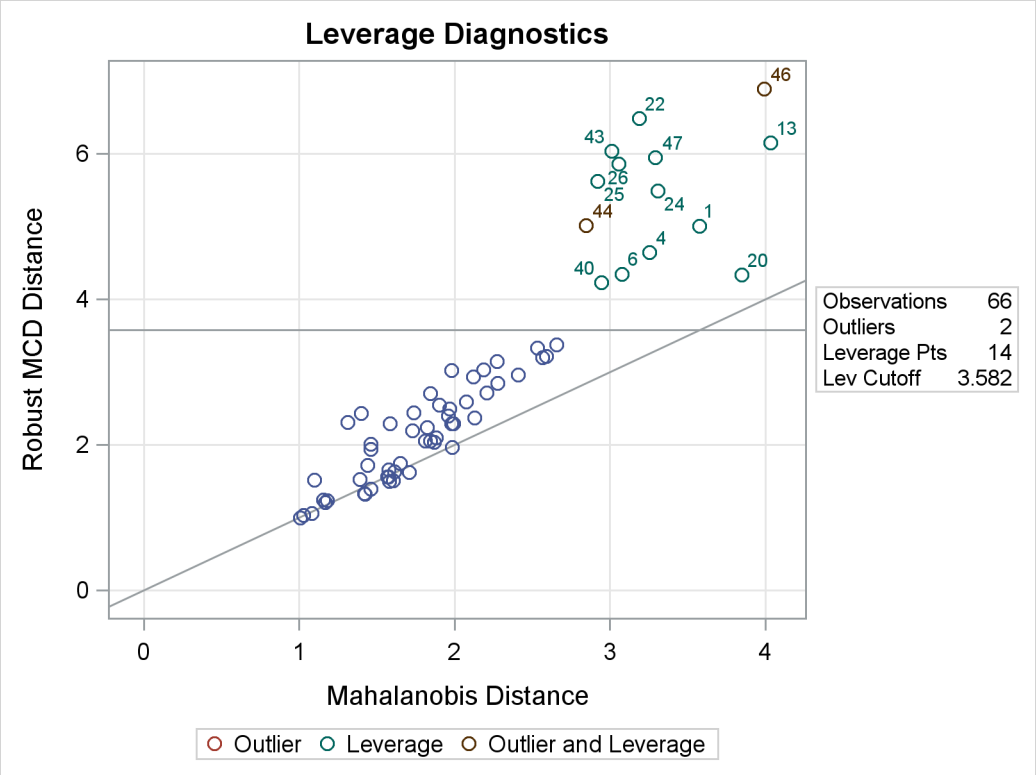
You might speculate that the projected MD and projected RD are equal to the regular MD and RD on the same data set without the variable cor. In fact, this is not true. (See Output 98.5.9 and Output 98.5.10 for the RD plot and DD plot of the data set without cor.) When cor is included in the MODEL statement, it is omitted from the distance calculation, but it is still used for the initial orthonormalization step and the h-subset searching. In this example, inclusion of cor causes all the other covariates to be centered separately for corner houses and non-corner houses. However, without cor, the centering process does not distinguish corner houses from non-corner houses, and therefore the MCD algorithm can still be influenced by cor through the correlation between cor and other covariates. The following statements drop the variable cor and produce the RD plot and DD plot for the reduced model, which are shown in Output 98.5.9 and Output 98.5.10, respectively:

```
proc robustreg data=house method=MM plots=all;
   model price = sqft age feats ne tax/leverage(mcdinfo) diagnostics;
run;
ods graphics off;
```

Output 98.5.9 RD Plot for the Reduced Model



Output 98.5.10 DD Plot for the Reduced Model



Compared with Output 98.5.8, Output 98.5.11 shows the changes of the MCD information by removing cor from the model. You can see that the corner houses are no longer identified as off-plane points and that the reweighted value of H is increased from 47 to 52. The breakdown value is intact because it depends only on the specified value of H and the total number of observations.

Output 98.5.11 MCD Information for the Reduced Model

MCD Profile	
Number of Dimensions	5
Number of Robust Dropped Components	0
Number of Observations	66
Number of Off-Plane Observations	0
Specified Value of H	51
Reweighted Value of H	52
Breakdown Value	0.2273

MCD Center

Parameter Name	Parameter	Center
sqft	**sqft**	1710.9
age	**age**	11.173
feats	**feats**	3.9423
ne	**ne**	0.5962
tax	**tax**	858.10

MCD Covariance

	sqft	age	feats	ne	tax
sqft	216974.7	681.2327	199.2492	103.0388	107503.1
age	681.2327	64.49887	-0.9506	1.855581	-187.135
feats	199.2492	-0.9506	0.878959	0.152715	114.9076
ne	103.0388	1.855581	0.152715	0.245475	49.98077
tax	107503.1	-187.135	114.9076	49.98077	66558.68

MCD Correlation

	sqft	age	feats	ne	tax
sqft	1	0.182102	0.456255	0.44647	0.89457
age	0.182102	1	-0.12625	0.466337	-0.09032
feats	0.456255	-0.12625	1	0.328771	0.475075
ne	0.44647	0.466337	0.328771	1	0.391018
tax	0.89457	-0.09032	0.475075	0.391018	1

References

Akaike, H. (1974). "A New Look at the Statistical Model Identification." *IEEE Transactions on Automatic Control* AC-19:716–723.

Brownlee, K. A. (1965). *Statistical Theory and Methodology in Science and Engineering*. New York: John Wiley & Sons.

Chen, C. (2002). "Robust Regression and Outlier Detection with the ROBUSTREG Procedure." In *Proceedings of the Twenty-Seventh Annual SAS Users Group International Conference*. Cary, NC: SAS Institute Inc. http://www2.sas.com/proceedings/sugi27/p265-27.pdf.

Chen, C., and Yin, G. (2002). "Computing the Efficiency and Tuning Constants for M-Estimation." In *Proceedings of the 2002 Joint Statistical Meetings*, 478–482. Alexandria, VA: American Statistical Association.

Coleman, D. E., Holland, P. W., Kaden, N., Klema, V., and Peters, S. C. (1980). "A System of Subroutines for Iteratively Reweighted Least Squares Computations." *ACM Transactions on Mathematical Software* 6:327–336.

Data and Story Library (2005). "Home Prices." Carnegie Mellon University, Department of Statistics. Accessed July 22, 2011. http://lib.stat.cmu.edu/DASL/Datafiles/homedat.html.

De Long, J. B., and Summers, L. H. (1991). "Equipment Investment and Economic Growth." *Quarterly Journal of Economics* 106:445–501.

Hampel, F. R., Ronchetti, E. M., Rousseeuw, P. J., and Stahel, W. A. (1986). *Robust Statistics: The Approach Based on Influence Functions*. New York: John Wiley & Sons.

Hawkins, D. M., Bradu, D., and Kass, G. V. (1984). "Location of Several Outliers in Multiple Regression Data Using Elemental Sets." *Technometrics* 26:197–208.

Holland, P. W., and Welsch, R. E. (1977). "Robust Regression Using Iteratively Reweighted Least-Squares." *Communications in Statistics—Theory and Methods* 6:813–827.

Huber, P. J. (1973). "Robust Regression: Asymptotics, Conjectures, and Monte Carlo." *Annals of Statistics* 1:799–821.

Huber, P. J. (1981). *Robust Statistics*. New York: John Wiley & Sons.

Marazzi, A. (1993). *Algorithm, Routines, and S Functions for Robust Statistics*. Pacific Grove, CA: Wadsworth & Brooks/Cole.

Ronchetti, E. M. (1985). "Robust Model Selection in Regression." *Statistics and Probability Letters* 3:21–23.

Rousseeuw, P. J. (1984). "Least Median of Squares Regression." *Journal of the American Statistical Association* 79:871–880.

Rousseeuw, P. J., and Hubert, M. (1996). "Recent Development in PROGRESS." *Computational Statistics and Data Analysis* 21:67–85.

Rousseeuw, P. J., and Leroy, A. M. (1987). *Robust Regression and Outlier Detection*. New York: John Wiley & Sons.

Rousseeuw, P. J., and Van Driessen, K. (1999). "A Fast Algorithm for the Minimum Covariance Determinant Estimator." *Technometrics* 41:212–223.

Rousseeuw, P. J., and Van Driessen, K. (2000). "An Algorithm for Positive-Breakdown Regression Based on Concentration Steps." In *Data Analysis: Scientific Modeling and Practical Application*, edited by W. Gaul, O. Opitz, and M. Schader, 335–346. New York: Springer-Verlag.

Rousseeuw, P. J., and Yohai, V. (1984). "Robust Regression by Means of S-Estimators." In *Robust and Nonlinear Time Series Analysis*, edited by J. Franke, W. Härdle, and R. D. Martin, 256–274. Vol. 26 of Lecture Notes in Statistics. Berlin: Springer-Verlag.

Ruppert, D. (1992). "Computing S Estimators for Regression and Multivariate Location/Dispersion." *Journal of Computational and Graphical Statistics* 1:253–270.

Yohai, V. J. (1987). "High Breakdown Point and High Efficiency Robust Estimates for Regression." *Annals of Statistics* 15:642–656.

Yohai, V. J., Stahel, W. A., and Zamar, R. H. (1991). "A Procedure for Robust Estimation and Inference in Linear Regression." In *Directions in Robust Statistics and Diagnostics, Part 2*, edited by W. A. Stahel, and S. W. Weisberg, 365–374. New York: Springer-Verlag.

Yohai, V. J., and Zamar, R. H. (1997). "Optimal Locally Robust M-Estimates of Regression." *Journal of Statistical Planning and Inference* 64:309–323.

Zaman, A., Rousseeuw, P. J., and Orhan, M. (2001). "Econometric Applications of High-Breakdown Robust Regression Techniques." *Econometrics Letters* 71:1–8.

Subject Index

Syntax Index