

# **SAS/STAT<sup>®</sup> 14.3**

## **User's Guide**

### **The QUANTREG**

### **Procedure**

This document is an individual chapter from *SAS/STAT® 14.3 User's Guide*.

The correct bibliographic citation for this manual is as follows: SAS Institute Inc. 2017. *SAS/STAT® 14.3 User's Guide*. Cary, NC: SAS Institute Inc.

### **SAS/STAT® 14.3 User's Guide**

Copyright © 2017, SAS Institute Inc., Cary, NC, USA

All Rights Reserved. Produced in the United States of America.

**For a hard-copy book:** No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, or otherwise, without the prior written permission of the publisher, SAS Institute Inc.

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

The scanning, uploading, and distribution of this book via the Internet or any other means without the permission of the publisher is illegal and punishable by law. Please purchase only authorized electronic editions and do not participate in or encourage electronic piracy of copyrighted materials. Your support of others' rights is appreciated.

**U.S. Government License Rights; Restricted Rights:** The Software and its documentation is commercial computer software developed at private expense and is provided with RESTRICTED RIGHTS to the United States Government. Use, duplication, or disclosure of the Software by the United States Government is subject to the license terms of this Agreement pursuant to, as applicable, FAR 12.212, DFAR 227.7202-1(a), DFAR 227.7202-3(a), and DFAR 227.7202-4, and, to the extent required under U.S. federal law, the minimum restricted rights as set out in FAR 52.227-19 (DEC 2007). If FAR 52.227-19 is applicable, this provision serves as notice under clause (c) thereof and no other notice is required to be affixed to the Software or documentation. The Government's rights in Software and documentation shall be only those set forth in this Agreement.

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

September 2017

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

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

SAS software may be provided with certain third-party software, including but not limited to open-source software, which is licensed under its applicable third-party software license agreement. For license information about third-party software distributed with SAS software, refer to <http://support.sas.com/thirdpartylicenses>.

# Chapter 98

## The QUANTREG Procedure

### Contents

---

Overview: QUANTREG Procedure . . . . .	<b>7964</b>
Features . . . . .	7966
Quantile Regression . . . . .	7967
Getting Started: QUANTREG Procedure . . . . .	<b>7968</b>
Analysis of Fish-Habitat Relationships . . . . .	7968
Growth Charts for Body Mass Index . . . . .	7974
Syntax: QUANTREG Procedure . . . . .	<b>7978</b>
PROC QUANTREG Statement . . . . .	7978
BY Statement . . . . .	7984
CLASS Statement . . . . .	7984
EFFECT Statement . . . . .	7985
ESTIMATE Statement . . . . .	7986
ID Statement . . . . .	7987
MODEL Statement . . . . .	7988
OUTPUT Statement . . . . .	7992
PERFORMANCE Statement . . . . .	7993
TEST Statement . . . . .	7994
WEIGHT Statement . . . . .	7994
Details: QUANTREG Procedure . . . . .	<b>7995</b>
Quantile Regression as an Optimization Problem . . . . .	7995
Optimization Algorithms . . . . .	7996
Confidence Interval . . . . .	8003
Covariance-Correlation . . . . .	8006
Linear Test . . . . .	8007
Leverage Point and Outlier Detection . . . . .	8009
INEST= Data Set . . . . .	8010
OUTEST= Data Set . . . . .	8010
Computational Resources . . . . .	8011
ODS Table Names . . . . .	8011
ODS Graphics . . . . .	8012
Examples: QUANTREG Procedure . . . . .	<b>8017</b>
Example 98.1: Comparison of Algorithms . . . . .	8017
Example 98.2: Quantile Regression for Econometric Growth Data . . . . .	8021
Example 98.3: Quantile Regression Analysis of Birth-Weight Data . . . . .	8029
Example 98.4: Nonparametric Quantile Regression for Ozone Levels . . . . .	8035
Example 98.5: Quantile Polynomial Regression for Salary Data . . . . .	8037

Example 98.6: Fast Quantile Process Regression . . . . .	8041
References . . . . .	8049

# Overview: QUANTREG Procedure

The QUANTREG procedure uses quantile regression to model the effects of covariates on the conditional quantiles of a response variable.

Quantile regression was introduced by Koenker and Bassett (1978) as an extension of ordinary least squares (OLS) regression, which models the relationship between one or more covariates  $X$  and the *conditional mean* of the response variable  $Y$  given  $X = x$ . Quantile regression extends the OLS regression to model the *conditional quantiles* of the response variable, such as the median or the 90th percentile. Quantile regression is particularly useful when the rate of change in the conditional quantile, expressed by the regression coefficients, depends on the quantile.

Figure 98.1 Trout Density in Streams

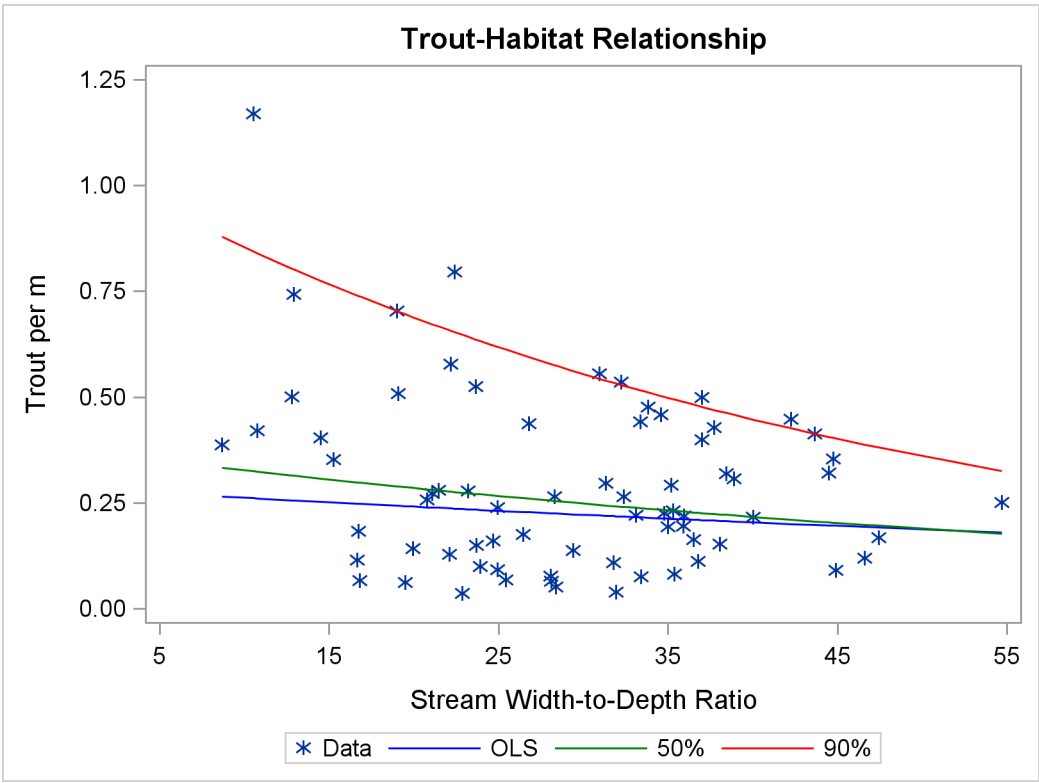


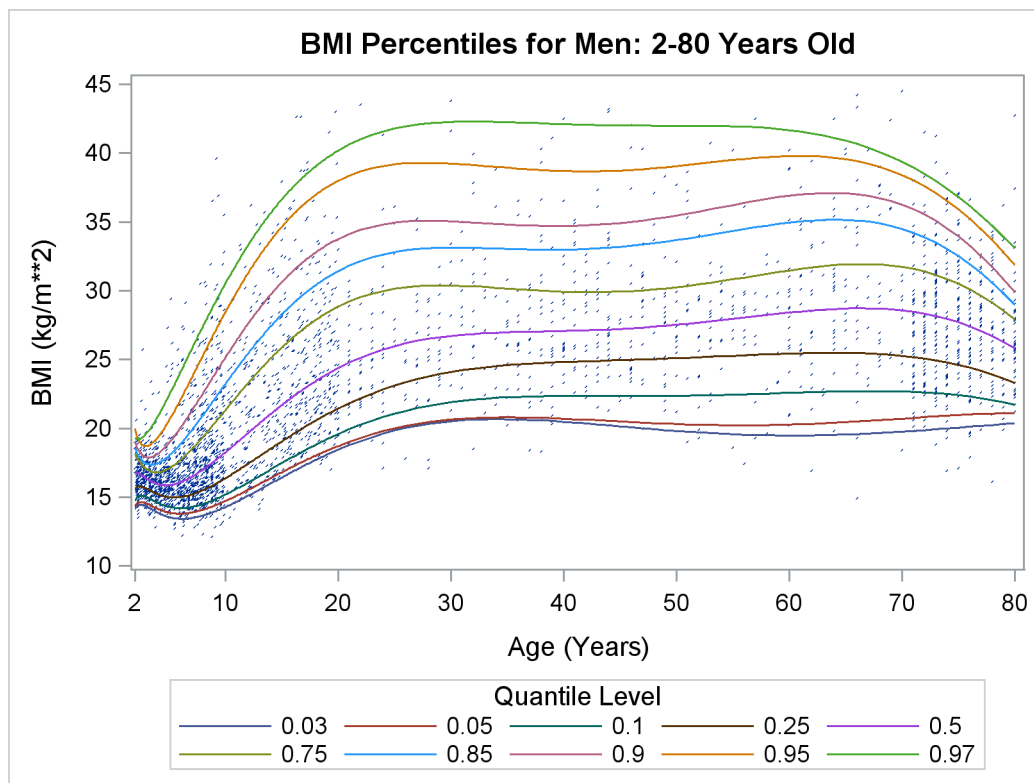
Figure 98.1 illustrates an ecological study in which modeling upper conditional quantiles reveals additional information. The points represent measurements of trout density and stream width-to-depth ratio that were taken at 13 streams over seven years.

As analyzed by Dunham, Cade, and Terrell (2002), both the ratio and the trout density depend on a number of unmeasured limiting factors that are related to the integrity of stream habitat. The interaction of these factors results in unequal variances for the conditional distributions of density given the ratio. When the ratio is the

“active” limiting effect, changes in the upper conditional percentiles of density provide a better estimate of this effect than changes in the conditional mean.

The red and green curves represent the conditional 90th and 50th percentiles of density as determined by the QUANTREG procedure. The analysis was done by using a simple linear regression model for the logarithm of density. (The curves in [Figure 98.1](#) were obtained by transforming the fitted lines back to the original scale. For more information, see the section “[Analysis of Fish-Habitat Relationships](#)” on page 7968.) The slope parameter for the 90th percentile has an estimated value of  $-0.0215$  and is significant with a  $p$ -value less than 0.01. On the other hand, the slope parameter for the 50th percentile is not significantly different from 0. Similarly, the slope parameter for the mean, which is obtained with OLS regression, is not significantly different from 0.

**Figure 98.2** Percentiles for Body Mass Index



Quantile regression is especially useful when the data are heterogeneous in the sense that the tails and the central location of the conditional distributions vary differently with the covariates. An even more pronounced example of heterogeneity is shown in [Figure 98.2](#), which plots the body mass index of 8,250 men versus their age.

Here, both upper (overweight) and lower (underweight) conditional quantiles are important because they provide the basis for developing growth charts and establishing health standards. The curves in [Figure 98.2](#) were determined by using the QUANTREG procedure to perform polynomial quantile regression. For more information, see the section “[Growth Charts for Body Mass Index](#)” on page 7974. Clearly, the rate of change with age (as expressed by the regression coefficients), particularly for ages less than 20, is different for each conditional quantile.

Heterogeneous data occur in many fields, including biomedicine, econometrics, survival analysis, and ecology. Quantile regression, which includes median regression as a special case, provides a complete picture of the

covariate effect when a set of percentiles is modeled. So it can capture important features of the data that might be missed by models that average over the conditional distribution.

Because it makes no distributional assumption about the error term in the model, quantile regression offers considerable model robustness. The assumption of normality, which is often made with OLS regression in order to compute conditional quantiles as offsets from the mean, forces a common set of regression coefficients for all the quantiles. Obviously, quantiles with common slopes would be inappropriate in the preceding examples.

Quantile regression is also flexible because it does not involve a link function that relates the variance and the mean of the response variable. Generalized linear models, which you can fit with the GENMOD procedure, require both a link function and a distributional assumption such as the normal or Poisson distribution. The goal of generalized linear models is inference about the regression parameters in the linear predictor for the mean of the population. In contrast, the goal of quantile regression is inference about regression coefficients for the conditional quantiles of a response variable that is usually assumed to be continuous.

Quantile regression also offers a degree of data robustness. Unlike OLS regression, quantile regression is robust to extreme points in the response direction (outliers). However, it is not robust to extreme points in the covariate space (leverage points). When both types of robustness are of concern, consider using the ROBUSTREG procedure (Chapter 101, “[The ROBUSTREG Procedure](#).”)

Unlike OLS regression, quantile regression is equivariant to monotone transformations of the response variable. For example, as illustrated in the trout example, the logarithm of the 90th conditional percentile of trout density is the 90th conditional percentile of the logarithm of density.

Quantile regression cannot be carried out simply by segmenting the unconditional distribution of the response variable and then obtaining least squares fits for the subsets. This approach leads to disastrous results when, for example, the data include outliers. In contrast, quantile regression uses *all* of the data for fitting quantiles, even the extreme quantiles.

---

## Features

The main features of the QUANTREG procedure are as follows:

- offers simplex, interior point, and smoothing algorithms for estimation
- provides sparsity, rank, and resampling methods for confidence intervals
- provides asymptotic and bootstrap methods for covariance and correlation matrices of the estimated parameters
- provides the Wald, likelihood ratio, and rank tests for the regression parameter estimates and the Wald test for heteroscedasticity
- provides outlier and leverage-point diagnostics
- enables parallel computing when multiple processors are available
- provides rowwise or columnwise output data sets with multiple quantiles
- provides regression quantile spline fits

- produces fit plots, diagnostic plots, and quantile process plots by using ODS Graphics

The next section provides notation and a formal definition for quantile regression.

## Quantile Regression

Quantile regression generalizes the concept of a univariate quantile to a conditional quantile given one or more covariates. Recall that a student's score on a test is at the  $\tau$  quantile if his or her score is better than that of  $100\tau\%$  of the students who took the test. The score is also said to be at the  $100\tau$ th percentile.

For a random variable  $Y$  with probability distribution function

$$F(y) = \text{Prob}(Y \leq y)$$

the  $\tau$  quantile of  $Y$  is defined as the inverse function

$$Q(\tau) = \inf \{y : F(y) \geq \tau\}$$

where the quantile level  $\tau$  ranges between 0 and 1. In particular, the median is  $Q(1/2)$ .

For a random sample  $\{y_1, \dots, y_n\}$  of  $Y$ , it is well known that the sample median minimizes the sum of absolute deviations:

$$\text{median} = \arg \min_{\xi \in \mathbf{R}} \sum_{i=1}^n |y_i - \xi|$$

Likewise, the general  $\tau$  sample quantile  $\xi(\tau)$ , which is the analog of  $Q(\tau)$ , is formulated as the minimizer

$$\xi(\tau) = \arg \min_{\xi \in \mathbf{R}} \sum_{i=1}^n \rho_{\tau}(y_i - \xi)$$

where  $\rho_{\tau}(z) = z(\tau - I(z < 0))$ ,  $0 < \tau < 1$ , and where  $I(\cdot)$  denotes the indicator function. The loss function  $\rho_{\tau}$  assigns a weight of  $\tau$  to positive residuals  $y_i - \xi$  and a weight of  $1 - \tau$  to negative residuals.

Using this loss function, the linear conditional quantile function extends the  $\tau$  sample quantile  $\xi(\tau)$  to the regression setting in the same way that the linear conditional mean function extends the sample mean. Recall that OLS regression estimates the linear conditional mean function  $E(Y|X = x) = \mathbf{x}'\boldsymbol{\beta}$  by solving for

$$\hat{\boldsymbol{\beta}} = \arg \min_{\boldsymbol{\beta} \in \mathbf{R}^p} \sum_{i=1}^n (y_i - \mathbf{x}_i' \boldsymbol{\beta})^2$$

The estimated parameter  $\hat{\beta}$  minimizes the sum of squared residuals in the same way that the sample mean  $\hat{\mu}$  minimizes the sum of squares:

$$\hat{\mu} = \arg \min_{\mu \in \mathbf{R}} \sum_{i=1}^n (y_i - \mu)^2$$

Likewise, quantile regression estimates the linear conditional quantile function,  $Q_Y(\tau|X = x) = \mathbf{x}'\boldsymbol{\beta}(\tau)$ , by solving the following equation for  $\tau \in (0, 1)$ :

$$\hat{\boldsymbol{\beta}}(\tau) = \arg \min_{\boldsymbol{\beta} \in \mathbf{R}^p} \sum_{i=1}^n \rho_{\tau}(y_i - \mathbf{x}_i' \boldsymbol{\beta})$$

The quantity  $\hat{\boldsymbol{\beta}}(\tau)$  is called the  $\tau$  *regression quantile*. The case  $\tau = 0.5$  (which minimizes the sum of absolute residuals) corresponds to median regression (which is also known as  $L_1$  regression).

The following set of regression quantiles is referred to as the *quantile process*:

$$\{\boldsymbol{\beta}(\tau) : \tau \in (0, 1)\}$$

The QUANTREG procedure computes the quantile function  $Q_Y(\tau|X = x)$  and conducts statistical inference on the estimated parameters  $\hat{\boldsymbol{\beta}}(\tau)$ .

---

## Getting Started: QUANTREG Procedure

The following examples demonstrate how you can use the QUANTREG procedure to fit linear models for selected quantiles or for the entire quantile process. The first example explains the use of the procedure in a fish-habitat example, and the second example explains the use of the procedure to construct growth charts for body mass index.

---

### Analysis of Fish-Habitat Relationships

Quantile regression is used extensively in ecological studies (Cade and Noon 2003). Recently, Dunham, Cade, and Terrell (2002) applied quantile regression to analyze fish-habitat relationships for Lahontan cutthroat trout in 13 streams of the eastern Lahontan basin, which covers most of northern Nevada and parts of southern Oregon. The density of trout (number of trout per meter) was measured by sampling stream sites from 1993 to 1999. The width-to-depth ratio of the stream site was determined as a measure of stream habitat.

The goal of this study was to explore the relationship between the conditional quantiles of trout density and the width-to-depth ratio. The scatter plot of the data in [Figure 98.1](#) indicates a nonlinear relationship, so it is reasonable to fit regression models for the conditional quantiles of the log of density. Because regression quantiles are equivariant under any monotonic (linear or nonlinear) transformation (Koenker and Hallock 2001), the exponential transformation converts the conditional quantiles to the original density scale.



The data set trout, which follows, includes the average numbers of Lahontan cutthroat trout per meter of stream (Density), the logarithm of Density (LnDensity), and the width-to-depth ratios (WDRatio) for 71 samples:

```
data trout;
  input Density WDRatio LnDensity @@;
  datalines;
0.38732      8.6819      -0.94850      1.16956      10.5102      0.15662
0.42025      10.7636     -0.86690      0.50059      12.7884     -0.69197
0.74235      12.9266     -0.29793      0.40385      14.4884     -0.90672
0.35245      15.2476     -1.04284      0.11499      16.6495     -2.16289
0.18290      16.7188     -1.69881      0.06619      16.7859     -2.71523
0.70330      19.0141     -0.35197      0.50845      19.0548     -0.67639

... more lines ...

0.25125      54.6916     -1.38129
;
```

The following statements use the QUANTREG procedure to fit a simple linear model for the 50th and 90th percentiles of LnDensity:

```
ods graphics on;

proc quantreg data=trout alpha=0.1 ci=resampling;
  model LnDensity = WDRatio / quantile=0.5 0.9
                                CovB seed=1268;
  test WDRatio / wald lr;
run;
```

The MODEL statement specifies a simple linear regression model with LnDensity as the response variable  $Y$  and WDRatio as the covariate  $X$ . The QUANTILE= option requests that the regression quantile function  $Q(\tau|X = x) = \mathbf{x}'\boldsymbol{\beta}(\tau)$  be estimated by solving the following equation, where  $\tau = (0.5, 0.9)$ :

$$\hat{\boldsymbol{\beta}}(\tau) = \arg \min_{\boldsymbol{\beta} \in \mathbb{R}^2} \sum_{i=1}^n \rho_{\tau}(y_i - \mathbf{x}_i' \boldsymbol{\beta})$$

By default, the regression coefficients  $\hat{\boldsymbol{\beta}}(\tau)$  are estimated by using the simplex algorithm, which is explained in the section “[Simplex Algorithm](#)” on page 7996. The ALPHA= option requests 90% confidence limits for the regression parameters, and the option CI=RESAMPLING specifies that the intervals be computed by using the Markov chain marginal bootstrap (MCMB) resampling method of He and Hu (2002). When you specify the CI=RESAMPLING option, the QUANTREG procedure also computes standard errors,  $t$  values, and  $p$ -values of regression parameters by using the MCMB resampling method. The SEED= option specifies a seed for the resampling method. The COVB option requests covariance matrices for the estimated regression coefficients, and the TEST statement requests tests for the hypothesis that the slope parameter (the coefficient of WDRatio) is 0.

Figure 98.3 displays model information and summary statistics for the variables in the model. The summary statistics include the median and the standardized median absolute deviation (MAD), which are robust measures of univariate location and scale, respectively. For more information about the standardized MAD, see Huber (1981, p. 108).

**Figure 98.3** Model Fitting Information and Summary Statistics

The QUANTREG Procedure						
Model Information						
Data Set	WORK.TROUT					
Dependent Variable	LnDensity					
Number of Independent Variables	1					
Number of Observations	71					
Optimization Algorithm	Simplex					
Method for Confidence Limits	Resampling					
Summary Statistics						
Variable	Q1	Median	Q3	Mean	Standard Deviation	MAD
WDRatio	22.0917	29.4083	35.9382	29.1752	9.9859	10.4970
LnDensity	-2.0511	-1.3813	-0.8669	-1.4973	0.7682	0.8214

Figure 98.4 and Figure 98.5 display the parameter estimates, standard errors, 95% confidence limits,  $t$  values, and  $p$ -values that are computed by the resampling method.

**Figure 98.4** Parameter Estimates at QUANTILE=0.5

Parameter Estimates						
Parameter	DF	Estimate	Standard Error	90% Confidence Limits		Pr >  t
Intercept	1	-0.9811	0.3952	-1.6400	-0.3222	-2.48 0.0155
WDRatio	1	-0.0136	0.0123	-0.0341	0.0068	-1.11 0.2705

**Figure 98.5** Parameter Estimates at QUANTILE=0.9

Parameter Estimates						
Parameter	DF	Estimate	Standard Error	90% Confidence Limits		Pr >  t
Intercept	1	0.0576	0.2606	-0.3769	0.4921	0.22 0.8257
WDRatio	1	-0.0215	0.0075	-0.0340	-0.0091	-2.88 0.0053

The 90th percentile of trout density can be predicted from the width-to-depth ratio as follows:

$$\hat{y}_{90} = \exp(0.0576 - 0.0215x)$$

This is the upper dashed curve that is plotted in Figure 98.1. The lower dashed curve for the median can be obtained in a similar fashion.

The covariance matrices for the estimated parameters are shown in Figure 98.6. The resampling method that is used for the confidence intervals is also used to compute these matrices.

**Figure 98.6** Covariance Matrices of the Estimated Parameters

**The QUANTREG Procedure**  
**Quantile Level = 0.5**

Estimated Covariance Matrix  
for Quantile Level = 0.5

	Intercept	WDRatio
Intercept	0.156191	-.004653
WDRatio	-.004653	0.000151

**The QUANTREG Procedure**  
**Quantile Level = 0.9**

Estimated Covariance Matrix  
for Quantile Level = 0.9

	Intercept	WDRatio
Intercept	0.067914	-.001877
WDRatio	-.001877	0.000056

The tests requested by the TEST statement are shown in Figure 98.7. Both the Wald test and the likelihood ratio test indicate that the coefficient of width-to-depth ratio is significantly different from 0 at the 90th percentile, but the difference is not significant at the median.

**Figure 98.7** Tests of Significance

		Test Results			
Quantile Level	Test	Test Statistic	DF	Chi-Square	Pr > ChiSq
0.5	Wald	1.2339	1	1.23	0.2666
0.5	Likelihood Ratio	1.1467	1	1.15	0.2842
0.9	Wald	8.3031	1	8.30	0.0040
0.9	Likelihood Ratio	9.0529	1	9.05	0.0026

In many quantile regression problems it is useful to examine how the estimated regression parameters for each covariate change as a function of  $\tau$  in the interval (0, 1). The following statements use the QUANTREG procedure to request the estimated quantile processes  $\hat{\beta}(\tau)$  for the slope and intercept parameters:

```
proc quantreg data=trout alpha=0.1 ci=resampling;
    model LnDensity = WDRatio / quantile=process seed=1268
                                plot=quantplot;
run;
```

The QUANTILE=PROCESS option requests an estimate of the quantile process for each regression parameter. The options ALPHA=0.1 and CI=RESAMPLING specify that 90% confidence bands for the quantile processes be computed by using the resampling method.

Figure 98.8 displays a portion of the objective function table for the entire quantile process. The objective function is evaluated at 77 values of  $\tau$  in the interval (0, 1). The table also provides predicted values of the conditional quantile function  $Q(\tau)$  at the mean for WDRatio, which can be used to estimate the conditional density function.

**Figure 98.8** Objective Function

Objective Function for Quantile Process			
Label	Quantile Level	Objective Function	Predicted at Mean
t0	0.005634	0.7044	-3.2582
t1	0.020260	2.5331	-3.0331
t2	0.031348	3.7421	-2.9376
t3	0.046131	5.2538	-2.7013
.	.	.	.
.	.	.	.
.	.	.	.
t73	0.945705	4.1433	-0.4361
t74	0.966377	2.5858	-0.4287
t75	0.976060	1.8512	-0.4082
t76	0.994366	0.4356	-0.4082

Figure 98.9 displays a portion of the table of the quantile processes for the estimated parameters and confidence limits.

**Figure 98.9** Objective Function

Parameter Estimates for Quantile Process			
Label	Quantile	Intercept	WDRatio
.	.	.	.
.	.	.	.
.	.	.	.
t57	0.765705	-0.42205	-0.01335
lower90	0.765705	-0.91952	-0.02682
upper90	0.765705	0.07541	0.00012
t58	0.786206	-0.32688	-0.01592
lower90	0.786206	-0.80883	-0.02895
upper90	0.786206	0.15507	-0.00289
.	.	.	.
.	.	.	.
.	.	.	.

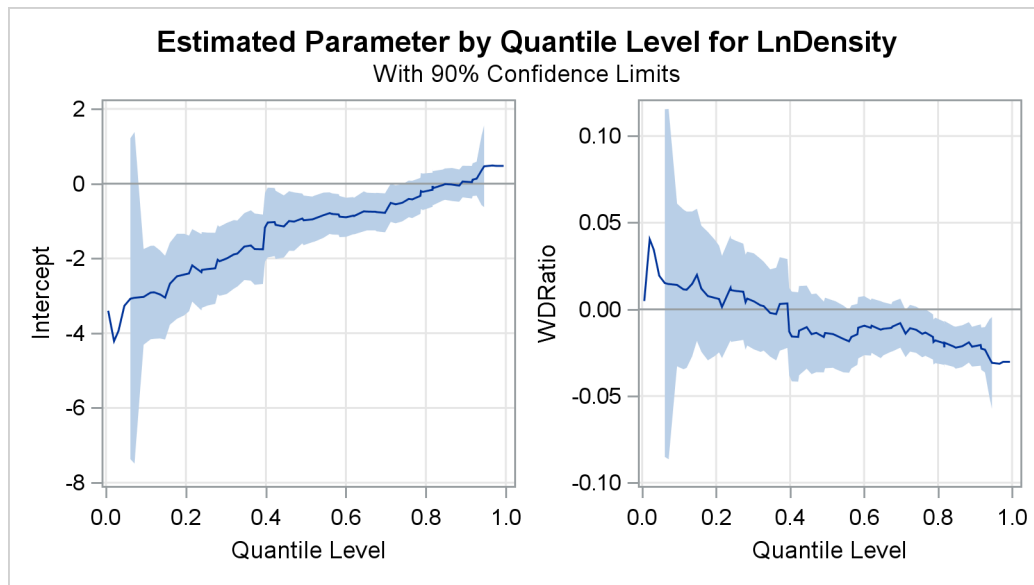
When ODS Graphics is enabled, the PLOT=QUANTPLOT option in the MODEL statement requests a plot of the estimated quantile processes.

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

The left side of [Figure 98.10](#) displays the process for the intercept, and the right side displays the process for the coefficient of WDRatio.

The process plot for WDRatio shows that the slope parameter changes from positive to negative as the quantile increases and that it changes sign with a sharp drop at the 40th percentile. The 90% confidence bands show that the relationship between LnDensity and WDRatio (expressed by the slope) is not significant below the 78th percentile. This situation can also be seen in [Figure 98.9](#), which shows that 0 falls between the lower and upper confidence limits of the slope parameter for quantiles below 0.78. Since the confidence intervals for the extreme quantiles are not stable because of insufficient data, the confidence band is not displayed outside the interval (0.05, 0.95).

**Figure 98.10** Quantile Processes for Intercept and Slope



## Growth Charts for Body Mass Index

Body mass index (BMI) is defined as the ratio of weight (kg) to squared height ( $m^2$ ) and is a widely used measure for categorizing individuals as overweight or underweight. The percentiles of BMI for specified ages are of particular interest. As age increases, these percentiles provide growth patterns of BMI not only for the majority of the population, but also for underweight or overweight extremes of the population. In addition, the percentiles of BMI for a specified age provide a reference for individuals at that age with respect to the population.

Smooth quantile curves have been widely used for reference charts in medical diagnosis to identify unusual subjects, whose measurements lie in the tails of the reference distribution. This example explains how to use the QUANTREG procedure to create growth charts for BMI.

A SAS data set named `bmimen` was created by merging and cleaning the 1999–2000 and 2001–2002 survey results for men that is published by the National Center for Health Statistics. This data set contains the variables Weight (kg), Height (m), BMI ( $kg/m^2$ ), Age (year), and `SeQN` (respondent sequence number) for 8,250 men (Chen 2005).

The data set that is used in this example is a subset of the original data set of Chen (2005). It contains the two variables BMI and Age with 3,264 observations.

```
data bmimen;
  input BMI Age @@;
  SqrtAge = sqrt(Age);
  InveAge = 1/Age;
  LogBMI = log(BMI);
  datalines;
18.6  2.0 17.1  2.0 19.0  2.0 16.8  2.0 19.0  2.1 15.5  2.1
16.7  2.1 16.1  2.1 18.0  2.1 17.8  2.1 18.3  2.1 16.9  2.1
15.9  2.1 20.6  2.1 16.7  2.1 15.4  2.1 15.9  2.1 17.7  2.1

... more lines ...

29.0 80.0 24.1 80.0 26.6 80.0 24.2 80.0 22.7 80.0 28.4 80.0
26.3 80.0 25.6 80.0 24.8 80.0 28.6 80.0 25.7 80.0 25.8 80.0
22.5 80.0 25.1 80.0 27.0 80.0 27.9 80.0 28.5 80.0 21.7 80.0
33.5 80.0 26.1 80.0 28.4 80.0 22.7 80.0 28.0 80.0 42.7 80.0
;
```

The logarithm of BMI is used as the response. (Although this does not improve the quantile regression fit, it helps with statistical inference.) A preliminary median regression is fitted with a parametric model, which involves six powers of Age.

The following statements invoke the QUANTREG procedure:

```
proc quantreg data=bmimen algorithm=interior(tolerance=1e-5) ci=resampling;
  model logbmi = inveage sqrtage age sqrtage*age
              age*age age*age*age
              / diagnostics cutoff=4.5 quantile=.5 seed=1268;
  id age bmi;
  test_age_cubic: test age*age*age / wald lr rankscore(tau);
run;
```

The MODEL statement provides the model, and the option QUANTILE=0.5 requests median regression. The ALGORITHM= option requests that the interior point algorithm be used to compute  $\hat{\beta}(\frac{1}{2})$ . For more information about this algorithm, see the section “[Interior Point Algorithm](#)” on page 7997.

Figure 98.11 displays the estimated parameters, standard errors, 95% confidence intervals,  $t$  values, and  $p$ -values that are computed by the resampling method, which is requested by the CI= option. All of the parameters are considered significant because the  $p$ -values are smaller than 0.001.

**Figure 98.11** Parameter Estimates with Median Regression: Men

Parameter Estimates							
Parameter	DF	Estimate	Standard Error	95% Confidence Limits		t Value	Pr >  t
Intercept	1	7.8909	0.8172	6.2887	9.4932	9.66	<.0001
InveAge	1	-1.8354	0.4355	-2.6892	-0.9816	-4.21	<.0001
SqrtAge	1	-5.1247	0.7137	-6.5241	-3.7253	-7.18	<.0001
Age	1	1.9759	0.2537	1.4784	2.4733	7.79	<.0001
SqrtAge*Age	1	-0.3347	0.0424	-0.4179	-0.2515	-7.89	<.0001
Age*Age	1	0.0227	0.0029	0.0170	0.0284	7.78	<.0001
Age*Age*Age	1	-0.0000	0.0000	-0.0001	-0.0000	-7.40	<.0001

The TEST statement requests Wald, likelihood ratio, and rank tests for the significance of the cubic term in Age. The test results, shown in Figure 98.12, indicate that this term is significant. Higher-order terms are not significant.

**Figure 98.12** Test of Significance for Cubic Term

Test test_age_cubic Results				
Test	Test			
	Statistic	DF	Chi-Square	Pr > ChiSq
Wald	54.8278	1	54.83	<.0001
Likelihood Ratio	56.9473	1	56.95	<.0001
Rank_Tau	42.5730	1	42.57	<.0001

Median regression and, more generally, quantile regression are robust to extremes of the response variable. The **DIAGNOSTICS** option in the **MODEL** statement requests a diagnostic table of outliers, shown in [Figure 98.13](#), which uses a cutoff value that is specified in the **CUTOFF=** option. The variables that are specified in the **ID** statement are included in the table.

With **CUTOFF=4.5**, 14 men are identified as outliers. All of these men have large positive standardized residuals, which indicates that they are overweight for their age. The cutoff value 4.5 is ad hoc. It corresponds to a probability less than  $0.5E-5$  if normality is assumed, but the standardized residuals for median regression usually do not meet this assumption.

In order to construct the chart shown in [Figure 98.2](#), the same model that is used for median regression is used for other quantiles. The **QUANTREG** procedure can compute fitted values for multiple quantiles.

**Figure 98.13** Diagnostics with Median Regression

Diagnostics				
Obs	Age	BMI	Standardized Residual	Outlier
1337	8.900000	36.500000	5.3575	*
1376	9.200000	39.600000	5.8723	*
1428	9.400000	36.900000	5.3036	*
1505	9.900000	35.500000	4.8862	*
1764	14.900000	46.800000	5.6403	*
1838	16.200000	50.400000	5.9138	*
1845	16.300000	42.600000	4.6683	*
1870	16.700000	42.600000	4.5930	*
1957	18.100000	49.900000	5.5053	*
2002	18.700000	52.700000	5.8106	*
2016	18.900000	48.400000	5.1603	*
2264	32.000000	55.600000	5.3085	*
2291	35.000000	60.900000	5.9406	*
2732	66.000000	14.900000	-4.7849	*



The following statements request fitted values for 10 quantile levels that range from 0.03 to 0.97:

```
proc quantreg data=bmimen algorithm=interior(tolerance=1e-5) ci=none;
  model logbmi = inveage sqrtage age sqrtage*age
                age*age age*age*age
                / quantile=0.03,0.05,0.1,0.25,0.5,0.75,
                  0.85,0.90,0.95,0.97;
  output out=outp pred=p/columnwise;
run;

data outbmi;
  set outp;
  pbmi = exp(p);
run;

proc sgplot data=outbmi;
  title 'BMI Percentiles for Men: 2-80 Years Old';
  yaxis label='BMI (kg/m**2)' min=10 max=45 values=(10 15 20 25 30 35 40 45);
  xaxis label='Age (Years)' min=2 max=80 values=(2 10 20 30 40 50 60 70 80);

  scatter x=age y=bmi /markerattrs=(size=1);
  series x=age y=pbmi/group=QUANTILE;
run;
```

The fitted values are stored in the OUTPUT data set outp. The COLUMNWISE option arranges these fitted values for all quantiles in the single variable p by groups of the quantiles. After the exponential transformation, both the fitted BMI values and the original BMI values are plotted against age to create the display shown in [Figure 98.2](#).

The fitted quantile curves reveal important information. During the quick growth period (ages 2 to 20), the dispersion of BMI increases dramatically. It becomes stable during middle age, and then it contracts after age 60. This pattern suggests that effective population weight control should start in childhood.

Compared to the 97th percentile in reference growth charts that were published by the Centers for Disease Control and Prevention (CDC) in 2000 (Kuczmarski, Ogden, and Guo 2002), the 97th percentile for 10-year-old boys in [Figure 98.2](#) is 6.4 BMI units higher (an increase of 27%). This can be interpreted as a warning of overweight or obesity. See Chen (2005) for a detailed analysis.

## Syntax: QUANTREG Procedure

The following statements are available in the QUANTREG procedure:

```

PROC QUANTREG < options > ;
  BY variables ;
  CLASS variables < / option > ;
  EFFECT name = effect-type (variables < / options > ) ;
  ESTIMATE < 'label' > estimate-specification < / options > ;
  ID variables ;
  MODEL response = < effects > < / options > ;
  OUTPUT < OUT= SAS-data-set > < options > ;
  PERFORMANCE < options > ;
  TEST effects < / options > ;
  WEIGHT variable ;

```

The PROC QUANTREG statement invokes the QUANTREG procedure. The CLASS statement specifies which explanatory variables are treated as categorical. The ID statement names variables to identify observations in the outlier diagnostics tables. The MODEL statement is required and specifies the variables used in the regression. Main effects and interaction terms can be specified in the MODEL statement, as in the GLM procedure (Chapter 48, “[The GLM Procedure](#).”) The OUTPUT statement creates an output data set that contains predicted values, residuals, and estimated standard errors. The PERFORMANCE statement tunes the performance of PROC QUANTREG by using single or multiple processors available in the hardware. The TEST statement requests linear tests for the model parameters. The WEIGHT statement identifies a variable in the input data set whose values are used to weight the observations. Multiple OUTPUT and TEST statements are allowed in one invocation of PROC QUANTREG.

The EFFECT and ESTIMATE statements are also available in other procedures. Summary descriptions of functionality and syntax for these statements are provided in this chapter, and you can find full documentation about them in Chapter 19, “[Shared Concepts and Topics](#).”

## PROC QUANTREG Statement

```
PROC QUANTREG < options > ;
```

The PROC QUANTREG statement invokes the QUANTREG procedure. [Table 98.1](#) summarizes the *options* available in the PROC QUANTREG statement.

**Table 98.1** PROC QUANTREG Statement Options

Option	Description
ALGORITHM=	Specifies an algorithm to estimate the regression parameters
ALPHA=	Specifies the level of significance
CI=	Specifies a method to compute confidence intervals
DATA=	Specifies the input SAS data set
INEST=	Specifies an input SAS data set that contains initial estimates
NAMELEN=	Specifies the length of effect names

**Table 98.1** *continued*

Option	Description
ORDER=	Specifies the order in which to sort classification variables
OUTEST=	Specifies an output SAS data set containing the parameter estimates
PLOT	Specifies options that control details of the plots

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

**ALGORITHM=***algorithm* < ( *suboptions* ) >

specifies an algorithm for estimating the regression parameters.

You can specify one of the following four *algorithms*:

**SIMPLEX** < *suboption* >

uses the simplex algorithm to estimate the regression parameters. You can specify the following *suboption*:

**MAXSTATIONARY=***m* requests that the algorithm terminate if the objective function has not improved for *m* consecutive iterations. By default, *m* = 1000.

**INTERIOR** < ( *suboptions* ) >

uses the interior point algorithm to estimate the regression parameters. You can specify the following *suboptions*:

**KAPPA=***value* specifies the step-length parameter for the interior point algorithm. The *value* should be between 0 and 1. The larger the *value*, the faster the algorithm. However, numeric instability can occur as the *value* approaches 1. By default, KAPPA=0.99995. For more information, see the section “[Interior Point Algorithm](#)” on page 7997.

**MAXIT=***m* sets the maximum number of iterations for the interior point algorithm. By default, MAXIT=1000.

**TOLERANCE=***value* specifies the tolerance for the convergence criterion of the interior point algorithm. By default, TOLERANCE=1E–8. The QUANTREG procedure uses the duality gap as the convergence criterion. For more information, see the section “[Interior Point Algorithm](#)” on page 7997.

You can also use the PERFORMANCE statement to enable parallel computing when multiple processors are available in the hardware.

**IPM** < ( *suboptions* ) >

uses the efficient interior point algorithm to estimate the regression parameters. You can specify the following *suboptions*:

**MAXIT=***m* sets the maximum number of iterations for the efficient interior point algorithm. By default, MAXIT=1000.

**TOLERANCE=***value* specifies the tolerance for the convergence criterion of the efficient interior point algorithm. The QUANTREG procedure uses the complementarity value as the convergence criterion. By default, TOLERANCE=1E-8.

**SMOOTH** <*suboption*> uses the smoothing algorithm to estimate the regression parameters. You can specify the following *suboption*:

**RRATIO=***value* specifies the reduction ratio for the smoothing algorithm. This ratio is used to reduce the threshold of the smoothing algorithm. The *value* should be between 0 and 1. In theory, the smaller the *value*, the faster the smoothing algorithm. However, in practice, the optimal ratio is quite dependent on the data. For more information, see the section “Smoothing Algorithm” on page 8000.

The default algorithm depends on the number of observations (*n*) and the number of covariates (*p*) in the model estimation. See Table 98.2 for the relevant defaults.

**Table 98.2** The Default Estimation Algorithm

	$p \leq 100$	$p > 100$
$n \leq 5000$	SIMPLEX	SMOOTH
$n > 5000$	INTERIOR	SMOOTH

**ALPHA=***value* specifies the level of significance  $\alpha$  for  $100(1 - \alpha)\%$  confidence intervals for regression parameters. The *value* must be between 0 and 1. The default is ALPHA=0.05, which corresponds to a 0.95 confidence interval.

**CI=NONE | RANK | SPARSITY**<(BF | HS)></IID> | **RESAMPLING**<(NREP=*n*)> specifies a method for computing confidence intervals for regression parameters. When you specify CI=SPARSITY or CI=RESAMPLING, the QUANTREG procedure also computes standard errors, *t* values, and *p*-values for regression parameters.

Table 98.3 summarizes these methods.

**Table 98.3** Options for Confidence Intervals

Value of CI=	Method	Additional Options
NONE	No confidence intervals computed	
RANK	By inverting rank-score tests	
RESAMPLING	By resampling	NREP
SPARSITY	By estimating sparsity function	HS, BF, and IID

By default, when there are fewer than 5,000 observations, fewer than 20 variables in the data set, and the algorithm is simplex, the QUANTREG procedure computes confidence intervals by using the inverted rank-score test method. Otherwise, the resampling method is used.

By default, confidence intervals are not computed for the quantile process, which is estimated when you specify the **QUANTILE=PROCESS** option in the **MODEL** statement. Confidence intervals for the quantile process are computed by using the sparsity or resampling methods when you specify **CI=SPARSITY** or **CI=RESAMPLING**, respectively. The rank method for confidence intervals is not available for quantile processes because it is computationally prohibitive.

When you specify the **SPARSITY** option, you have two suboptions for estimating the sparsity function. If you specify the **IID** suboption, the sparsity function is estimated by assuming that the errors in the linear model are independent and identically distributed (**iid**). By default, the sparsity function is estimated by assuming that the conditional quantile function is locally linear. For more information, see the section “[Sparsity](#)” on page 8004. For both methods, two bandwidth selection methods are available: You can specify the **BF** suboption for the Bofinger method or the **HS** suboption for the Hall-Sheather method. By default, the Hall-Sheather method is used.

When you specify the **RESAMPLING** option, you can specify the **NREP=*n*** suboption for the number of repetitions. By default, **NREP=200**. The value of *n* must be greater than 50.

**DATA=SAS-data-set**

specifies the input SAS data set to be used by the QUANTREG procedure. By default, the most recently created SAS data set is used.

**INEST=SAS-data-set**

specifies an input SAS data set that contains initial estimates for all the parameters in the model. The interior point algorithm and the smoothing algorithm use these estimates as a start. For a detailed description of the contents of the **INEST=** data set, see the section “[INEST= Data Set](#)” on page 8010.

**NAMELEN=*n***

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

**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 <b>ORDER=</b>	Levels Sorted By
<b>DATA</b>	Order of appearance in the input data set
<b>FORMATTED</b>	External formatted value, except for numeric variables with no explicit format, which are sorted by their unformatted (internal) value
<b>FREQ</b>	Descending frequency count; levels with the most observations come first in the order
<b>INTERNAL</b>	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 *SAS Visual Data Management and Utility Procedures Guide* and the discussion of BY-group processing in *SAS Language Reference: Concepts*.

#### **OUTEST=SAS-data-set**

specifies an output SAS data set to contain the parameter estimates for all quantiles. See the section “[OUTEST= Data Set](#)” on page 8010 for a detailed description of the contents of the OUTEST= data set.

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

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

specifies options that control details of the plots. These plots fall into two categories: diagnostic plots and fit plots. You can also use the [PLOT=](#) option in the MODEL statement to request the quantile process plot for any effects that are specified in the model. If you do not specify the PLOTS= option, PROC QUANTREG produces the quantile fit plot by default when a single continuous variable is specified in the model.

When you specify only one *plot-request*, you can omit the parentheses around the plot request.

Here are some examples:

```
plots=ddplot
plots=(ddplot rdplot)
```

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

```
ods graphics on;

proc quantreg plots=fitplot;
  model y=x1;
run;
```

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

You can specify the following *global-plot-options*, which apply to all plots that PROC QUANTREG generates:

#### **MAXPOINTS=NONE | number**

suppresses plots that have elements that require processing more than *number* points. The default is MAXPOINTS=5000. This cutoff is ignored if you specify MAXPOINTS=NONE.

#### **ONLY**

suppresses the default quantile fit plot. Only plots specifically requested are displayed.

You can specify the following *plot-requests*:

**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 8009. The LABEL= option specifies how the points on this plot are to be labeled, as summarized by [Table 98.4](#).

**Table 98.4** 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 QUANTREG 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 | SHOWLIMITS | NODATA)>**

creates a plot of fitted conditional quantiles against the single continuous variable that is specified in the model. This plot is produced only when the response is modeled as a function of a single continuous variable. Multiple lines or curves are drawn on this plot if you specify several quantiles with the QUANTILE= option in the MODEL statement. By default, confidence limits are added to the plot when a single quantile is requested, and the confidence limits are not shown on the plot when multiple quantiles are requested. The NOLIMITS option suppresses the display of the confidence limits. The SHOWLIMITS option adds the confidence limits when multiple quantiles are requested. The NODATA option suppresses the display of the observed data, which are superimposed on the plot by default.

**HISTOGRAM**

creates a histogram (based on the quantile regression estimates) for the standardized 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 (based on the quantile regression estimates) for the standardized residuals.

**RDPlot<(LABEL=ALL | LEVERAGE | NONE | OUTLIER)>**

creates the plot of standardized residual against robust distance. For more information about robust distance, see the section “[Leverage Point and Outlier Detection](#)” on page 8009.

The LABEL= option specifies a label method for points on this plot. These label methods are described in [Table 98.4](#).

By default, the QUANTREG 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.

## PP

requests preprocessing to speed up the interior point algorithm or the smoothing algorithm. The preprocessing uses a subsampling algorithm (which assumes that the data set is evenly distributed) to iteratively reduce the original problem to a smaller one. Preprocessing should be used only for very large data sets, such as data sets with more than 100,000 observations. For more information, see Portnoy and Koenker (1997).

---

## BY Statement

**BY** *variables* ;

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

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

---

## CLASS Statement

**CLASS** *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 *SAS Visual Data Management and Utility 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 QUANTREG 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*:

<b>COLLECTION</b>	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.
<b>LAG</b>	specifies a classification effect in which the level that is used for a particular period corresponds to the level in the preceding period.
<b>MULTIMEMBER   MM</b>	specifies a multimember classification effect whose levels are determined by one or more variables that appear in a CLASS statement.
<b>POLYNOMIAL   POLY</b>	specifies a multivariate polynomial effect in the specified numeric variables.
<b>SPLINE</b>	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 98.5 summarizes the *options* available in the EFFECT statement.

**Table 98.5** EFFECT Statement Options

Option	Description
<b>Collection Effects Options</b>	
DETAILS	Displays the constituents of the collection effect

Table 98.5 *continued*

Option	Description
<b>Lag Effects Options</b>	
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.
<b>Multimember Effects Options</b>	
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
<b>Polynomial Effects Options</b>	
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
<b>Spline Effects Options</b>	
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](#).”

## ESTIMATE Statement

```
ESTIMATE <'label'> estimate-specification <(divisor=n)>
    <,> ... <'label'> estimate-specification <(divisor=n)> >
    </options>;
```

The ESTIMATE statement provides a mechanism for obtaining custom hypothesis tests. Estimates are formed as linear estimable functions of the form  $\mathbf{L}\boldsymbol{\beta}$ . You can perform hypothesis tests for the estimable functions, construct confidence limits, and obtain specific nonlinear transformations.

Table 98.6 summarizes the *options* available in the ESTIMATE statement.

**Table 98.6** ESTIMATE Statement Options

Option	Description
<b>Construction and Computation of Estimable Functions</b>	
DIVISOR=	Specifies a list of values to divide the coefficients
NOFILL	Suppresses the automatic fill-in of coefficients for higher-order effects
SINGULAR=	Tunes the estimability checking difference
<b>Degrees of Freedom and <math>p</math>-values</b>	
ADJUST=	Determines the method for multiple comparison adjustment of estimates
ALPHA= $\alpha$	Determines the confidence level $(1 - \alpha)$
LOWER	Performs one-sided, lower-tailed inference
STEPPDOWN	Adjusts multiplicity-corrected $p$ -values further in a step-down fashion
TESTVALUE=	Specifies values under the null hypothesis for tests
UPPER	Performs one-sided, upper-tailed inference
<b>Statistical Output</b>	
CL	Constructs confidence limits
CORR	Displays the correlation matrix of estimates
COV	Displays the covariance matrix of estimates
E	Prints the <b>L</b> matrix
JOINT	Produces a joint $F$ or chi-square test for the estimable functions
PLOTS=	Requests ODS statistical graphics if the analysis is sampling-based
SEED=	Specifies the seed for computations that depend on random numbers
<b>Generalized Linear Modeling</b>	
CATEGORY=	Specifies how to construct estimable functions with multinomial data
EXP	Exponentiates and displays estimates
ILINK	Computes and displays estimates and standard errors on the inverse linked scale

For details about the syntax of the ESTIMATE statement, see the section “ESTIMATE Statement” on page 448 in Chapter 19, “Shared Concepts and Topics.”

## ID Statement

ID *variables* ;

When the diagnostics table is requested by the **DIAGNOSTICS** option in the **MODEL** statement, the variables listed in the **ID** statement are displayed in addition to the observation number. These values are useful for identifying observations. If the **ID** statement is omitted, only the observation number is displayed.

## MODEL Statement

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

You can specify main effects and interaction terms in the **MODEL** statement, as you can in the GLM procedure (Chapter 48, “[The GLM Procedure](#).”) Classification variables in the **MODEL** statement must also be specified in the **CLASS** statement.

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

## Options

Table 98.7 summarizes the *options* available in the **MODEL** statement.

**Table 98.7** MODEL Statement Options

Option	Description
<b>CORRB</b>	Produces the estimated correlation matrix
<b>COVB</b>	Produces the estimated covariance matrix
<b>CUTOFF=</b>	Specifies the multiplier of the cutoff value for outlier detection
<b>DIAGNOSTICS</b>	Requests the outlier diagnostics
<b>ITPRINT</b>	Displays the iteration history
<b>LEVERAGE</b>	Requests an analysis of leverage points
<b>NODIAG</b>	Suppresses the computation for outlier diagnostics
<b>NOINT</b>	Specifies no-intercept regression
<b>NOSUMMARY</b>	Suppresses the computation for summary statistics
<b>PLOT=</b>	Requests plots
<b>QUANTILE=</b>	Specifies the quantile levels
<b>SCALE=</b>	Specifies the scale value used to compute the standardized residuals
<b>SEED=</b>	Specifies the seed for the random number generator
<b>SINGULAR=</b>	Specifies the tolerance for testing singularity

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

### **CORRB**

produces the estimated correlation matrix of the parameter estimates. When the resampling method is used to compute the confidence intervals, the QUANTREG procedure computes the bootstrap correlation. When the sparsity method is used to compute the confidence intervals, PROC QUANTREG bases its computation of the asymptotic correlation on an estimator of the sparsity function. The rank method for confidence intervals does not provide a correlation estimate.

**COVB**

produces the estimated covariance matrix of the parameter estimates. When the resampling method is used to compute the confidence intervals, the QUANTREG procedure computes the bootstrap covariance. When the sparsity method is used to compute the confidence intervals, PROC QUANTREG bases its computation of the asymptotic covariance on an estimator of the sparsity function. The rank method for confidence intervals does not provide a covariance estimate.

**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. To request that all observations be displayed, specify the ALL option.

**ITPRINT**

displays the iteration history of the interior point algorithm or the smoothing algorithm.

**LEVERAGE**<(CUTOFF= *value* | CUTOFFALPHA= *value* | H= *n*)>

requests an analysis of leverage points for the continuous covariates. The results are added to the diagnostics table, which you can request with the DIAGNOSTICS option in the MODEL statement. You can specify the cutoff value for leverage-point detection with the CUTOFF= option. The default cutoff value is  $\sqrt{\chi^2_{p;1-\alpha}}$ , where  $\alpha$  can be specified with the CUTOFFALPHA= option. By default,  $\alpha = 0.025$ . You can use the H= option to specify the number of points to be minimized for the MCD algorithm 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. The LEVERAGE option is ignored if the model includes classification variables as covariates.

**NODIAG**

suppresses the computation for outlier diagnostics. If you specify the NODIAG option, the diagnostics summary table is not provided.

**NOINT**

specifies no intercept regression.

**NOSUMMARY**

suppresses the computation of summary statistics. If you specify the NOSUMMARY option, the summary statistics table is not provided.

**PLOT=***plot-option***PLOTS=**(*plot-option*)

You can use the PLOTS= option in the MODEL statement together with ODS Graphics to request the quantile process plot in addition to all that plots that you request in the **PLOT=** option in the PROC QUANTREG statement.

You can specify the following *plot-option* in the MODEL statement:

**QUANTPLOT**<(EFFECTS) </ <NOLIMITS> <EXTENDCI> <UNPACK> <OLS> > >

plots the regression quantile process. The estimated coefficient of each specified covariate effect is plotted as a function of the quantile. If you do not specify a covariate effect, quantile processes are plotted for all covariate effects in the MODEL statement. You can use the NOLIMITS option to suppress confidence bands for the quantile processes. By default, confidence bands are plotted,

and process plots are displayed in panels, each of which can hold up to four plots. By default, the confidence limits are plotted for quantiles in the range between 0.05 and 0.95. You can use the `EXTENDCI` option to plot the confidence limits even for quantiles outside this range. You can use the `UNPACK` option to create individual process plots. For an individual process plot, you can superimpose the ordinary least squares estimate by specifying the `OLS` option.

ODS Graphics must be enabled before you request plots.

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

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

**QUANTLEV=***number-list* | **PROCESS** | **FQPR**<(suboption)>

specifies the quantile levels for the quantile regression. A valid quantile level must be a number in the range of (0,1). You can specify the following values for the **QUANTILE=** option:

*number-list*

computes quantile regression for quantile levels that are specified in the *number-list*.

#### **PROCESS**

computes the entire quantile process regression. If you specify **QUANTILE=PROCESS**, the value of the **ALGORITHM=** option in the **PROC QUANTREG** statement must be **SIMPLEX** either by default or by specifying it.

The **QUANTILE=PROCESS** option produces the quantile process estimates table and the quantile process objective function table. The size of these two tables are proportional to the number of the training observations and can be large for a large training data set. You can suppress displaying these two tables by specifying the following **ODS EXCLUDE** statement:

```
ods exclude ProcessEst ProcessObj;
```

In addition, you can output these tables to data sets for further processing by specifying the following **ODS OUTPUT** statement:

```
ods output ProcessEst=PE ProcessObj=PO;
```

**FQPR**<(suboption)>

uses a fast quantile process regression method to approximate quantile process regression on a grid of  $n$  equally spaced quantile levels. If you specify **QUANTILE=FQPR**, the value of the **ALGORITHM=** option in the **PROC QUANTREG** statement must be **IPM**. For more details about the fast quantile process regression method, see the section “[Fast Quantile Process Regression](#)” on page 8003. You can specify the following suboptions for the **FQPR** option:

**N**= $n$

specifies the number  $n$  of equally spaced quantile levels at which to fit the quantile process regression.

**OBSRATIO=***value*

**OR=***value*

specifies the number of equally spaced quantile levels as its ratio to the total number of training observations. For example, if the number of training observations is 1,000 and you specify the OR=0.2 suboption, a quantile process regression model is fit for  $n = 0.2 \times 1,000 = 200$  equally spaced quantile levels. The FQPR option ignores the OR= suboption if a valid N= suboption is specified.

**L=***value*

specifies the starting quantile level of the quantile-level grid. By default,  $L=1/2n$  if **U** is not specified, otherwise  $L=U/(2n-1)$ .

**U=***value*

specifies the ending quantile level of the quantile-level grid.  $U=(2n-1)/2n$  if **L** is not specified, otherwise  $U=(L+2n-2)/(2n-1)$ .

If you specify neither the N=*n* nor the OR=*value* suboption, the FQPR option determines the number of quantile levels as the lesser of 100 and half the number of the training observations.

Unlike the QUANTILE=PROCESS option, the QUANTILE=FQPR option does not display the quantile process estimates table and the quantile process objective function table. Instead the QUANTILE=FQPR option produces the average parameter estimates table and the average objective function table for the specified quantile-level grid. However, you can output the two quantile process tables to data sets by specifying the following ODS OUTPUT statement:

```
ods output ProcessEst=PE ProcessObj=PO;
```

By default, QUANTILE=0.5, which fits a median regression.

**SCALE=***number*

specifies the scale value to use to compute the standardized residuals. By default, the scale is computed as the corrected median of absolute residuals. See the section “[Leverage Point and Outlier Detection](#)” on page 8009 for details.

**SEED=***number*

specifies the seed for the random number generator used to compute the MCMB confidence intervals. This seed is also used to randomly select the subgroups for preprocessing when you specify the PP option in the PROC QUANTREG statement. If you do not specify a seed, or if you specify a value less than or equal to 0, the seed is generated from reading the time of day from the computer clock.

By default or if you specify SEED=0, the QUANTREG procedure generates a seed between one and one billion.

**SINGULAR=***value*

sets the tolerance for testing singularity of the information matrix and the crossproducts matrix for the initial least squares estimates. Approximately, the test requires that a pivot be at least this value times the original diagonal value. By default, SINGULAR=1E-12.

## OUTPUT Statement

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

The OUTPUT statement creates a SAS data set to contain statistics that are calculated after PROC QUANTREG fits models for all specified quantiles that are specified in the QUANTILE= option in the MODEL statement. 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 from options in the OUTPUT statement. These new variables contain fitted values and estimated quantiles. If you want 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*.

If you specify multiple quantiles in the MODEL statement, the COLUMNWISE option arranges the created OUTPUT data set in columnwise form. This arrangement repeats the input data for each quantile. By default, the OUTPUT data set is created in rowwise form. For each appropriate keyword specified in the OUTPUT statement, one variable for each specified quantile is generated. These variables appear in the sorted order of the specified quantiles.

The following specifications can appear in the OUTPUT statement:

**OUT**=*SAS-data-set* specifies the new data set. By default, PROC QUANTREG 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. For each desired statistic, specify a *keyword* from the following list of *keywords*, an equal sign, and the name of a variable to contain the statistic.

You can specify the following *keywords*:

<b>LEVERAGE</b>	specifies a variable to indicate leverage points. To include this variable in the OUTPUT data set, you must specify the LEVERAGE option in the MODEL statement. See the section “ <a href="#">Leverage Point and Outlier Detection</a> ” on page 8009 for how to define LEVERAGE.
<b>MAHADIST</b>   <b>MD</b>	names a variable to contain the Mahalanobis distance. To include this variable in the OUTPUT data set, you must specify the LEVERAGE option in the MODEL statement.
<b>OUTLIER</b>	specifies a variable to indicate outliers. See the section “ <a href="#">Leverage Point and Outlier Detection</a> ” on page 8009 for how to define OUTLIER.
<b>PREDICTED</b>   <b>P</b>	names a variable to contain the estimated response.
<b>QUANTILE</b>   <b>Q</b>	names a variable to contain the quantile for which the quantile regression is fitted. If you specify the COLUMNWISE option, this variable is created by default. If multiple quantiles are specified in the MODEL statement and the COLUMNWISE option is not specified, this variable is not created.
<b>RESIDUAL</b>   <b>RES</b>	names a variable to contain the residuals (unstandardized):

$$y_i - \mathbf{x}_i' \hat{\boldsymbol{\beta}}$$



<b>ROBDIST   RD</b>	names a variable to contain the robust MCD distance. To include this variable in the OUTPUT data set, you must specify the LEVERAGE option in the MODEL statement.
<b>SPLINE   SP</b>	names a variable to contain the estimated spline effect, which includes all spline effects in the model and their interactions.
<b>SRESIDUAL   SR</b>	names a variable to contain the standardized residuals: $\frac{y_i - \mathbf{x}_i' \hat{\boldsymbol{\beta}}}{\hat{\sigma}}$ See the section “ <a href="#">Leverage Point and Outlier Detection</a> ” on page 8009 for how to compute $\sigma$ .
<b>STDP</b>	names a variable to contain the estimates of the standard errors of the estimated response.

---

## PERFORMANCE Statement

You can use the PERFORMANCE statement to change default options that affect the performance of PROC QUANTREG and to request tables that show the performance options in effect and the timing details.

**PERFORMANCE** <options> ;

You can specify the following *options*:

### **CPUCOUNT=number | ACTUAL**

specifies the number of processors to use in the computation of the interior point algorithm. CPUCOUNT=ACTUAL sets CPUCOUNT to be the number of physical processors available, which this can be less than the physical number of CPUs if the SAS process has been restricted by system administration tools. You can specify any integer from 1 to 1024 for *number*. Setting CPUCOUNT= to a *number* greater than the actual number of available CPUs might result in reduced performance. If CPUCOUNT=1, then [NOTHREADS](#) is in effect, and PROC QUANTREG uses singly threaded code. This option overrides the SAS system option CPUCOUNT=.

### **DETAILS**

creates the PerfSettings table that shows the performance settings in effect and the “Timing” table that provides a broad timing breakdown of the PROC QUANTREG step.

### **THREADS**

enables multithreaded computation for the interior point algorithm. If you do not specify the ALGORITHM=INTERIOR option in the PROC QUANTREG statement, then PROC QUANTREG ignores this option and uses singly threaded code. This option overrides the SAS system option THREADS | NOTHREADS.

### **NOTHREADS**

disables multithreaded computation for the interior point algorithm. This option overrides the SAS system option THREADS | NOTHREADS.

---

## TEST Statement

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

In quantile regression analysis, you might be interested in testing whether a covariate effect is statistically significant for a given quantile. In other situations, you might be interested in testing whether the coefficients of a covariate are the same across a set of quantiles. You can use the TEST Statement to perform these tests.

You can submit multiple TEST statements, provided that they appear after the MODEL statement. The optional *label*, which must be a valid SAS name, is used to identify output from the corresponding TEST statement. For more information about these tests, see the section “[Linear Test](#)” on page 8007.

### Testing Effects of Covariates

You can use TEST statement to obtain a test for the canonical linear hypothesis concerning the parameters of the tested effects,

$$\beta_j = 0, \quad j = i_1, \dots, i_q$$

where  $q$  is the total number of parameters of the tested effects. The tested *effects* can be any set of effects in the MODEL statement. You can specify three types of tests (Wald, likelihood ratio, and rank methods) for testing effects of covariates by using the following *options* in the TEST statement after a slash (/):

#### WALD

requests Wald tests.

#### LR

requests likelihood ratio tests.

#### RANKSCORE < (NORMAL | WILCOXON | SIGN | TAU) >

requests rank tests. The NORMAL, WILCOXON, and SIGN functions are implemented and suitable for iid error models, and the TAU score function is implemented and appropriate for non-iid error models. By default, the TAU score function is used. See Koenker (2005) for more information about the score functions.

### Testing for Heteroscedasticity

You can test whether there is any difference among the estimated coefficients across quantiles if several quantiles are specified in the MODEL statement. The test for such heteroscedasticity can be requested by the option QINTERACT after a slash (/) in the TEST statement. See [Example 98.5](#).

---

## WEIGHT Statement

**WEIGHT variable ;**

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

To request weighted quantile regression, place the weights in a *variable*. The values of the WEIGHT *variable* can be nonintegral and are not truncated. Observations with nonpositive or missing values for the weight variable do not contribute to the fit of the model. For more information about weighted quantile regression, see the section “[Details: QUANTREG Procedure](#)” on page 7995.

## Details: QUANTREG Procedure

### Quantile Regression as an Optimization Problem

The model for linear quantile regression is

$$y = \mathbf{A}'\boldsymbol{\beta} + \boldsymbol{\epsilon}$$

where  $\mathbf{y} = (y_1, \dots, y_n)'$  is the  $(n \times 1)$  vector of responses,  $\mathbf{A}' = (\mathbf{x}_1, \dots, \mathbf{x}_n)'$  is the  $(n \times p)$  regressor matrix,  $\boldsymbol{\beta} = (\beta_1, \dots, \beta_p)'$  is the  $(p \times 1)$  vector of unknown parameters, and  $\boldsymbol{\epsilon} = (\epsilon_1, \dots, \epsilon_n)'$  is the  $(n \times 1)$  vector of unknown errors.

$L_1$  regression, also known as median regression, is a natural extension of the sample median when the response is conditioned on the covariates. In  $L_1$  regression, the least absolute residuals estimate  $\hat{\boldsymbol{\beta}}_{LAR}$ , referred to as the  $L_1$ -norm estimate, is obtained as the solution of the following minimization problem:

$$\min_{\boldsymbol{\beta} \in \mathbf{R}^p} \sum_{i=1}^n |y_i - \mathbf{x}_i' \boldsymbol{\beta}|$$

More generally, for quantile regression Koenker and Bassett (1978) defined the  $\tau$  regression quantile,  $0 < \tau < 1$ , as any solution to the following minimization problem:

$$\min_{\boldsymbol{\beta} \in \mathbf{R}^p} \left[ \sum_{i \in \{i: y_i \geq \mathbf{x}_i' \boldsymbol{\beta}\}} \tau |y_i - \mathbf{x}_i' \boldsymbol{\beta}| + \sum_{i \in \{i: y_i < \mathbf{x}_i' \boldsymbol{\beta}\}} (1 - \tau) |y_i - \mathbf{x}_i' \boldsymbol{\beta}| \right]$$

The solution is denoted as  $\hat{\boldsymbol{\beta}}(\tau)$ , and the  $L_1$ -norm estimate corresponds to  $\hat{\boldsymbol{\beta}}(1/2)$ . The  $\tau$  regression quantile is an extension of the  $\tau$  sample quantile  $\hat{\xi}(\tau)$ , which can be formulated as the solution of

$$\min_{\xi \in \mathbf{R}} \left[ \sum_{i \in \{i: y_i \geq \xi\}} \tau |y_i - \xi| + \sum_{i \in \{i: y_i < \xi\}} (1 - \tau) |y_i - \xi| \right]$$

If you specify weights  $w_i, i = 1, \dots, n$ , with the WEIGHT statement, weighted quantile regression is carried out by solving

$$\min_{\boldsymbol{\beta}_w \in \mathbf{R}^p} \left[ \sum_{i \in \{i: y_i \geq \mathbf{x}_i' \boldsymbol{\beta}_w\}} w_i \tau |y_i - \mathbf{x}_i' \boldsymbol{\beta}_w| + \sum_{i \in \{i: y_i < \mathbf{x}_i' \boldsymbol{\beta}_w\}} w_i (1 - \tau) |y_i - \mathbf{x}_i' \boldsymbol{\beta}_w| \right]$$

Weighted regression quantiles  $\boldsymbol{\beta}_w$  can be used for L-estimation (Koenker and Zhao 1994).

## Optimization Algorithms

The optimization problem for median regression has been formulated and solved as a linear programming (LP) problem since the 1950s. Variations of the simplex algorithm, especially the method of Barrodale and Roberts (1973), have been widely used to solve this problem. The simplex algorithm is computationally demanding in large statistical applications, and in theory the number of iterations can increase exponentially with the sample size. This algorithm is often useful with data that contain no more than tens of thousands of observations.

Several alternatives have been developed to handle  $L_1$  regression for larger data sets. The interior point approach of Karmarkar (1984) solves a sequence of quadratic problems in which the relevant interior of the constraint set is approximated by an ellipsoid. The worst-case performance of the interior point algorithm has been proved to be better than the worst-case performance of the simplex algorithm. More important, experience has shown that the interior point algorithm is advantageous for larger problems.

Like  $L_1$  regression, general quantile regression fits nicely into the standard primal-dual formulations of linear programming.

In addition to the interior point method, various heuristic approaches are available for computing  $L_1$ -type solutions. Among these, the finite smoothing algorithm of Madsen and Nielsen (1993) is the most useful. It approximates the  $L_1$ -type objective function with a smoothing function, so that the Newton-Raphson algorithm can be used iteratively to obtain a solution after a finite number of iterations. The smoothing algorithm extends naturally to general quantile regression.

The QUANTREG procedure implements the simplex, interior point, and smoothing algorithms. The remainder of this section describes these algorithms in more detail.

### Simplex Algorithm

Let  $\mu = [y - A'\beta]_+$ ,  $\nu = [A'\beta - y]_+$ ,  $\phi = [\beta]_+$ , and  $\varphi = [-\beta]_+$ , where  $[z]_+$  is the nonnegative part of  $z$ .

Let  $D_{LAR}(\beta) = \sum_{i=1}^n |y_i - x_i'\beta|$ . For the  $L_1$  problem, the simplex approach solves  $\min_{\beta} D_{LAR}(\beta)$  by reformulating it as the constrained minimization problem

$$\min_{\beta} \{e'\mu + e'\nu \mid y = A'\beta + \mu - \nu, \{\mu, \nu\} \in \mathbf{R}_+^n\}$$

where  $e$  denotes an  $(n \times 1)$  vector of ones.

Let  $\mathbf{B} = [A' - A' \mathbf{I} - \mathbf{I}]$ ,  $\theta = (\phi' \varphi' \mu' \nu')'$ , and  $\mathbf{d} = (0' 0' e' e')'$ , where  $0' = (0 \ 0 \ \dots \ 0)_p$ . The reformulation presents a standard LP problem:

$$(P) \quad \min_{\theta} \mathbf{d}'\theta; \text{ subject to } \mathbf{B}\theta = \mathbf{y}, \theta \geq 0$$

This problem has the following dual formulation:

$$(D) \quad \max_z \mathbf{y}'z; \text{ subject to } \mathbf{B}'z \leq \mathbf{d}$$

This formulation can be simplified as

$$\max_z \mathbf{y}'z; \text{ subject to } \mathbf{A}z = 0, z \in [-1, 1]^n$$

By setting  $\eta = \frac{1}{2} + \frac{1}{2}\mathbf{e}$ ,  $\mathbf{b} = \frac{1}{2}\mathbf{A}\mathbf{e}$ , the problem becomes

$$\max_{\eta} \mathbf{y}'\eta; \text{ subject to } \mathbf{A}\eta = \mathbf{b}, \eta \in [0, 1]^n$$

For quantile regression, the minimization problem is  $\min_{\beta} \sum \rho_{\tau}(y_i - \mathbf{x}'_i\beta)$ , and a similar set of steps leads to the dual formulation

$$\max_{\mathbf{z}} \mathbf{y}'\mathbf{z}; \text{ subject to } \mathbf{A}\mathbf{z} = (1 - \tau)\mathbf{A}\mathbf{e}, \mathbf{z} \in [0, 1]^n$$

The QUANTREG procedure solves this LP problem by using the simplex algorithm of Barrodale and Roberts (1973). This algorithm exploits the special structure of the coefficient matrix  $\mathbf{B}$  by solving the primary LP problem ( $P$ ) in two stages: The first stage chooses the columns in  $\mathbf{A}'$  or  $-\mathbf{A}'$  as pivotal columns. The second stage interchanges the columns in  $\mathbf{I}$  or  $-\mathbf{I}$  as basis or nonbasis columns, respectively. The algorithm obtains an optimal solution by executing these two stages interactively. Moreover, because of the special structure of  $\mathbf{B}$ , only the main data matrix  $\mathbf{A}$  is stored in the current memory.

Although this special version of the simplex algorithm was introduced for median regression, it extends naturally to quantile regression for any given quantile and even to the entire quantile process (Koenker and d'Orey 1994). It greatly reduces the computing time that is required by the general simplex algorithm, and it is suitable for data sets with fewer than 5,000 observations and 50 variables.

### Interior Point Algorithm

The ALGORITHM=INTERIOR option implements an interior point algorithm. This algorithm uses the primal-dual predictor-corrector method that is proposed by Lustig, Marsten, and Shanno (1992). Roos, Terlaky, and Vial (1997) provide more information about this particular algorithm. The following brief introduction of this algorithm uses the notation in the first reference.

To be consistent with the conventional linear programming setting, let  $\mathbf{c} = -\mathbf{y}$ , let  $\mathbf{b} = (1 - \tau)\mathbf{A}\mathbf{e}$ , and let  $\mathbf{u}$  be the general upper bound. The dual form of quantile regression solves the following linear programming primal problem:

$$\min\{\mathbf{c}'\mathbf{z}\}; \text{ subject to } \mathbf{A}\mathbf{z} = \mathbf{b}, \mathbf{0} \leq \mathbf{z} \leq \mathbf{u}$$

This primal problem has  $n$  variables. The index  $i$  denotes a variable number, and  $k$  denotes an iteration number. If  $k$  is used as a subscript or superscript, it denotes “of iteration  $k$ .”

Let  $\mathbf{v}$  be the primal slack so that  $\mathbf{z} + \mathbf{v} = \mathbf{u}$ . Associate dual variables  $\mathbf{w}$  with these constraints. The interior point algorithm solves the system of equations to satisfy the Karush-Kuhn-Tucker (KKT) conditions for optimality:

$$\begin{aligned} \mathbf{b} &= \mathbf{A}\mathbf{z} \\ \mathbf{u} &= \mathbf{z} + \mathbf{v} \\ \mathbf{c} &= \mathbf{A}'\mathbf{t} + \mathbf{s} - \mathbf{w} \\ \mathbf{0} &= \mathbf{Z}\mathbf{S}\mathbf{e} \\ \mathbf{0} &= \mathbf{V}\mathbf{W}\mathbf{e} \\ \mathbf{z}, \mathbf{s}, \mathbf{v}, \mathbf{w} &\geq \mathbf{0} \end{aligned}$$

where  $\mathbf{W} = \text{diag}(\mathbf{w})$  (that is,  $W_{i,j} = \begin{cases} w_i & \text{for } i = j \\ 0 & \text{otherwise} \end{cases}$ ),  $\mathbf{V} = \text{diag}(\mathbf{v})$ ,  $\mathbf{Z} = \text{diag}(\mathbf{z})$ ,  $\mathbf{S} = \text{diag}(\mathbf{s})$ .

These are the conditions for feasibility, with the addition of *complementarity conditions*  $\mathbf{ZSe} = \mathbf{0}$  and  $\mathbf{VWe} = \mathbf{0}$ . The equality  $\mathbf{c}'\mathbf{z} = \mathbf{b}'\mathbf{t} - \mathbf{u}'\mathbf{w}$  must occur at the optimum. Complementarity forces the optimal objectives of the primal and dual to be equal,  $\mathbf{c}'\mathbf{z}_{opt} = \mathbf{b}'\mathbf{t}_{opt} - \mathbf{u}'\mathbf{w}_{opt}$ , because

$$\begin{aligned} 0 &= \mathbf{v}'_{opt}\mathbf{w}_{opt} = (\mathbf{u} - \mathbf{z}_{opt})'\mathbf{w}_{opt} = \mathbf{u}'\mathbf{w}_{opt} - \mathbf{z}'_{opt}\mathbf{w}_{opt} \\ 0 &= \mathbf{z}'_{opt}\mathbf{s}_{opt} = \mathbf{s}'_{opt}\mathbf{z}_{opt} = (\mathbf{c} - \mathbf{A}'\mathbf{t}_{opt} + \mathbf{w}_{opt})'\mathbf{z}_{opt} \\ &= \mathbf{c}'\mathbf{z}_{opt} - \mathbf{t}'_{opt}(\mathbf{Az}_{opt}) + \mathbf{w}'_{opt}\mathbf{z}_{opt} = \mathbf{c}'\mathbf{z}_{opt} - \mathbf{b}'\mathbf{t}_{opt} + \mathbf{u}'\mathbf{w}_{opt} \end{aligned}$$

Therefore

$$0 = \mathbf{c}'\mathbf{z}_{opt} - \mathbf{b}'\mathbf{t}_{opt} + \mathbf{u}'\mathbf{w}_{opt}$$

The *duality gap*,  $\mathbf{c}'\mathbf{z} - \mathbf{b}'\mathbf{t} + \mathbf{u}'\mathbf{w}$ , measures the convergence of the algorithm. You can specify a tolerance for this convergence criterion in the TOLERANCE= option in the PROC QUANTREG statement.

Before the optimum is reached, it is possible for a solution  $(\mathbf{z}, \mathbf{t}, \mathbf{s}, \mathbf{v}, \mathbf{w})$  to violate the KKT conditions in one of several ways:

- Primal bound constraints can be broken:  $\delta_b = \mathbf{u} - \mathbf{z} - \mathbf{v} \neq \mathbf{0}$ .
- Primal constraints can be broken:  $\delta_c = \mathbf{b} - \mathbf{Az} \neq \mathbf{0}$ .
- Dual constraints can be broken:  $\delta_d = \mathbf{c} - \mathbf{A}'\mathbf{t} - \mathbf{s} + \mathbf{w} \neq \mathbf{0}$ .
- Complementarity conditions are unsatisfied:  $\mathbf{z}'\mathbf{s} \neq 0$  and  $\mathbf{v}'\mathbf{w} \neq 0$ .

The interior point algorithm works by using Newton's method to find a direction  $(\Delta \mathbf{z}^k, \Delta \mathbf{t}^k, \Delta \mathbf{s}^k, \Delta \mathbf{v}^k, \Delta \mathbf{w}^k)$  to move from the current solution  $(\mathbf{z}^k, \mathbf{t}^k, \mathbf{s}^k, \mathbf{v}^k, \mathbf{w}^k)$  toward a better solution:

$$(\mathbf{z}^{k+1}, \mathbf{t}^{k+1}, \mathbf{s}^{k+1}, \mathbf{v}^{k+1}, \mathbf{w}^{k+1}) = (\mathbf{z}^k, \mathbf{t}^k, \mathbf{s}^k, \mathbf{v}^k, \mathbf{w}^k) + \kappa(\Delta \mathbf{z}^k, \Delta \mathbf{t}^k, \Delta \mathbf{s}^k, \Delta \mathbf{v}^k, \Delta \mathbf{w}^k)$$

$\kappa$  is the *step length* and is assigned a value as large as possible, but not so large that a  $\mathbf{z}_i^{k+1}$  or  $\mathbf{s}_i^{k+1}$  is “too close” to 0. You can control the step length in the KAPPA= option in the PROC QUANTREG statement.

The QUANTREG procedure implements a predictor-corrector variant of the primal-dual interior point algorithm. First, Newton's method is used to find a direction  $(\Delta \mathbf{z}_{aff}^k, \Delta \mathbf{t}_{aff}^k, \Delta \mathbf{s}_{aff}^k, \Delta \mathbf{v}_{aff}^k, \Delta \mathbf{w}_{aff}^k)$  in which to move. This is known as the *affine* step.

In iteration  $k$ , the affine step system that must be solved is

$$\begin{aligned} \delta_b &= \Delta \mathbf{z}_{aff} + \Delta \mathbf{v}_{aff} \\ \delta_c &= \mathbf{A}\Delta \mathbf{z}_{aff} \\ \delta_d &= \mathbf{A}'\Delta \mathbf{t}_{aff} + \Delta \mathbf{s}_{aff} - \Delta \mathbf{w}_{aff} = \delta_d \\ -\mathbf{ZSe} &= \mathbf{S}\Delta \mathbf{z}_{aff} + \mathbf{Z}\Delta \mathbf{s}_{aff} \\ -\mathbf{VWe} &= \mathbf{V}\Delta \mathbf{w}_{aff} + \mathbf{W}\Delta \mathbf{z}_{aff} \end{aligned}$$

Therefore, the following computations are involved in solving the affine step, where  $\kappa$  is the *step length* as before:

$$\begin{aligned}
\Theta &= \mathbf{SZ}^{-1} + \mathbf{WV}^{-1} \\
\rho &= \Theta^{-1}(\delta_d + (\mathbf{S} - \mathbf{W})\mathbf{e} - \mathbf{V}^{-1}\mathbf{W}\delta_b) \\
\Delta t_{aff} &= (\mathbf{A}\Theta^{-1}\mathbf{A}')^{-1}(\delta_c + \mathbf{A}\rho) \\
\Delta z_{aff} &= \Theta^{-1}\mathbf{A}'\Delta t_{aff} - \rho \\
\Delta v_{aff} &= \delta_b - \mathbf{A}\Delta z_{aff} \\
\Delta w_{aff} &= -\mathbf{W}\mathbf{e} - \mathbf{V}^{-1}\mathbf{W}\Delta z_{aff} \\
\Delta s_{aff} &= -\mathbf{S}\mathbf{e} - \mathbf{Z}^{-1}\mathbf{S}\Delta z_{aff} \\
(\mathbf{z}_{aff}, \mathbf{t}_{aff}, \mathbf{s}_{aff}, \mathbf{v}_{aff}, \mathbf{w}_{aff}) &= (\mathbf{z}, \mathbf{t}, \mathbf{s}, \mathbf{v}, \mathbf{w}) + \kappa(\Delta z_{aff}, \Delta t_{aff}, \Delta s_{aff}, \Delta v_{aff}, \Delta w_{aff})
\end{aligned}$$

The success of the affine step is gauged by calculating the complementarity of  $\mathbf{z}'\mathbf{s}$  and  $\mathbf{v}'\mathbf{w}$  at  $(\mathbf{z}_{aff}^k, \mathbf{t}_{aff}^k, \mathbf{s}_{aff}^k, \mathbf{v}_{aff}^k, \mathbf{w}_{aff}^k)$  and comparing it with the complementarity at the starting point  $(\mathbf{z}^k, \mathbf{t}^k, \mathbf{s}^k, \mathbf{v}^k, \mathbf{w}^k)$ . If the affine step was successful in reducing the complementarity by a substantial amount, the need for centering is not great. Therefore, a value close to 0 is assigned to  $\sigma$  in the following second linear system, which is used to determine a centering vector.

The following linear system is solved to determine a centering vector  $(\Delta z_c, \Delta t_c, \Delta s_c, \Delta v_c, \Delta w_c)$  from  $(\mathbf{z}_{aff}, \mathbf{t}_{aff}, \mathbf{s}_{aff}, \mathbf{v}_{aff}, \mathbf{w}_{aff})$ :

$$\begin{aligned}
\Delta z_c + \Delta v_c &= 0 \\
\mathbf{A}\Delta z_c &= 0 \\
\mathbf{A}'\Delta t_c + \Delta s_c - \Delta w_c &= 0 \\
\mathbf{S}\Delta z_c + \mathbf{Z}\Delta s_c &= -\mathbf{Z}_{aff}\mathbf{S}_{aff}\mathbf{e} + \sigma\mu\mathbf{e} \\
\mathbf{V}\Delta w_c + \mathbf{W}\Delta v_c &= -\mathbf{V}_{aff}\mathbf{W}_{aff}\mathbf{e} + \sigma\mu\mathbf{e}
\end{aligned}$$

where  $\zeta_{start} = \mathbf{z}'\mathbf{s} + \mathbf{v}'\mathbf{w}$ , complementarity at the start of the iteration  
 $\zeta_{aff} = \mathbf{z}'_{aff}\mathbf{s}_{aff} + \mathbf{v}'_{aff}\mathbf{w}_{aff}$ , the affine complementarity  
 $\mu = \zeta_{aff}/2n$ , the average complementarity  
 $\sigma = (\zeta_{aff}/\zeta_{start})^3$

However, if the affine step was unsuccessful, then centering is deemed beneficial, and a value close to 1.0 is assigned to  $\sigma$ . In other words, the value of  $\sigma$  is adaptively altered depending on the progress made toward the optimum.

Therefore, the following computations are involved in solving the centering step:

$$\begin{aligned}\rho &= \Theta^{-1}(\sigma\mu(\mathbf{Z}^{-1} - \mathbf{V}^{-1})\mathbf{e} - \mathbf{Z}^{-1}\mathbf{Z}_{\text{aff}}\mathbf{S}_{\text{aff}}\mathbf{e} + \mathbf{V}^{-1}\mathbf{V}_{\text{aff}}\mathbf{W}_{\text{aff}}\mathbf{e}) \\ \Delta t_c &= (\mathbf{A}\Theta^{-1}\mathbf{A}')^{-1}\mathbf{A}\rho \\ \Delta z_c &= \Theta^{-1}\mathbf{A}'\Delta t_c - \rho \\ \Delta v_c &= -\Delta z_c \\ \Delta w_c &= \sigma\mu\mathbf{V}^{-1}\mathbf{e} - \mathbf{V}^{-1}\mathbf{V}_{\text{aff}}\mathbf{W}_{\text{aff}}\mathbf{e} - \mathbf{V}^{-1}\mathbf{W}_{\text{aff}}\Delta v_c \\ \Delta s_c &= \sigma\mu\mathbf{Z}^{-1}\mathbf{e} - \mathbf{Z}^{-1}\mathbf{Z}_{\text{aff}}\mathbf{S}_{\text{aff}}\mathbf{e} - \mathbf{Z}^{-1}\mathbf{S}_{\text{aff}}\Delta z_c\end{aligned}$$

Then

$$\begin{aligned}(\Delta z, \Delta t, \Delta s, \Delta v, \Delta w) &= (\Delta z_{\text{aff}}, \Delta t_{\text{aff}}, \Delta s_{\text{aff}}, \Delta v_{\text{aff}}, \Delta w_{\text{aff}}) + (\Delta z_c, \Delta t_c, \Delta s_c, \Delta v_c, \Delta w_c) \\ (\mathbf{z}^{k+1}, \mathbf{t}^{k+1}, \mathbf{s}^{k+1}, \mathbf{v}^{k+1}, \mathbf{w}^{k+1}) &= (\mathbf{z}^k, \mathbf{t}^k, \mathbf{s}^k, \mathbf{v}^k, \mathbf{w}^k) + \kappa(\Delta z, \Delta t, \Delta s, \Delta v, \Delta w)\end{aligned}$$

where, as before,  $\kappa$  is the step length, which is assigned a value as large as possible but not so large that a  $\mathbf{z}_i^{k+1}$ ,  $\mathbf{s}_i^{k+1}$ ,  $\mathbf{v}_i^{k+1}$ , or  $\mathbf{w}_i^{k+1}$  is “too close” to 0.

Although the predictor-corrector variant entails solving two linear systems instead of one, fewer iterations are usually required to reach the optimum. The additional overhead of the second linear system is small because the matrix  $(\mathbf{A}\Theta^{-1}\mathbf{A}')$  has already been factorized in order to solve the first linear system.

You can specify the starting point in the INEST= option in the PROC QUANTREG statement. By default, the starting point is set to be the least squares estimate.

### Efficient Interior Point Algorithm

The ALGORITHM=IPM option implements a more efficient interior point algorithm than the one that is used when ALGORITHM=INTERIOR. The computing strategy of the ALGORITHM=IPM option is the same as the strategy for the ALGORITHM=INTERIOR option, but the ALGORITHM=IPM option implements the algorithm by using more efficient matrix functions. The ALGORITHM=IPM option uses the complementarity value to measure the convergence of the algorithm, which is different from the dual gap value that is used when ALGORITHM=INTERIOR. The complementarity value is defined as  $(\mathbf{z}'\mathbf{s} + \mathbf{v}'\mathbf{w})$ . You can specify a tolerance for this complementarity convergence criterion by using the TOLERANCE= option in the PROC QUANTREG statement. Unlike the ALGORITHM=INTERIOR option, the ALGORITHM=IPM option does not support the KAPPA= option.

### Smoothing Algorithm

To minimize the sum of the absolute residuals  $D_{LAR}(\boldsymbol{\beta})$ , the smoothing algorithm approximates the nondifferentiable function  $D_{LAR}$  by the following smooth function (which is referred to as the Huber function),

$$D_\gamma(\boldsymbol{\beta}) = \sum_{i=1}^n H_\gamma(r_i(\boldsymbol{\beta}))$$

where

$$H_\gamma(t) = \begin{cases} t^2/(2\gamma) & \text{if } |t| \leq \gamma \\ |t| - \gamma/2 & \text{if } |t| > \gamma \end{cases}$$



Here  $r_i(\boldsymbol{\beta}) = y_i - \mathbf{x}_i' \boldsymbol{\beta}$ , and the threshold  $\gamma$  is a positive real number. The function  $D_\gamma$  is continuously differentiable, and a minimizer  $\boldsymbol{\beta}_\gamma$  of  $D_\gamma$  is close to a minimizer  $\hat{\boldsymbol{\beta}}_{LAR}$  of  $D_{LAR}(\boldsymbol{\beta})$  when  $\gamma$  is close to 0.

The advantage of the smoothing algorithm as described in Madsen and Nielsen (1993) is that the  $L_1$  solution  $\hat{\boldsymbol{\beta}}_{LAR}$  can be detected when  $\gamma > 0$  is small. In other words, it is not necessary to let  $\gamma$  converge to 0 in order to find a minimizer of  $D_{LAR}(\boldsymbol{\beta})$ . The algorithm terminates before going through the entire sequence of values of  $\gamma$  that are generated by the algorithm. Convergence is indicated by no change of the status of residuals  $r_i(\boldsymbol{\beta})$  as  $\gamma$  goes through this sequence.

The smoothing algorithm extends naturally from  $L_1$  regression to general quantile regression (Chen 2007). The function

$$D_{\rho_\tau}(\boldsymbol{\beta}) = \sum_{i=1}^n \rho_\tau(y_i - \mathbf{x}_i' \boldsymbol{\beta})$$

can be approximated by the smooth function

$$D_{\gamma,\tau}(\boldsymbol{\beta}) = \sum_{i=1}^n H_{\gamma,\tau}(r_i(\boldsymbol{\beta}))$$

where

$$H_{\gamma,\tau}(t) = \begin{cases} t(\tau - 1) - \frac{1}{2}(\tau - 1)^2\gamma & \text{if } t \leq (\tau - 1)\gamma \\ \frac{t^2}{2\gamma} & \text{if } (\tau - 1)\gamma \leq t \leq \tau\gamma \\ t\tau - \frac{1}{2}\tau^2\gamma & \text{if } t \geq \tau\gamma \end{cases}$$

The function  $H_{\gamma,\tau}$  is determined by whether  $r_i(\boldsymbol{\beta}) \leq (\tau - 1)\gamma$ ,  $r_i(\boldsymbol{\beta}) \geq \tau\gamma$ , or  $(\tau - 1)\gamma \leq r_i(\boldsymbol{\beta}) \leq \tau\gamma$ . These inequalities divide  $\mathbf{R}^p$  into subregions that are separated by the parallel hyperplanes  $r_i(\boldsymbol{\beta}) = (\tau - 1)\gamma$  and  $r_i(\boldsymbol{\beta}) = \tau\gamma$ . The set of all such hyperplanes is denoted by  $B_{\gamma,\tau}$ :

$$B_{\gamma,\tau} = \{\boldsymbol{\beta} \in \mathbf{R}^p \mid \exists i : r_i(\boldsymbol{\beta}) = (\tau - 1)\gamma \text{ or } r_i(\boldsymbol{\beta}) = \tau\gamma\}$$

Define the sign vector  $\mathbf{s}_\gamma(\boldsymbol{\beta}) = (s_1(\boldsymbol{\beta}), \dots, s_n(\boldsymbol{\beta}))'$  as

$$s_i = s_i(\boldsymbol{\beta}) = \begin{cases} -1 & \text{if } r_i(\boldsymbol{\beta}) \leq (\tau - 1)\gamma \\ 0 & \text{if } (\tau - 1)\gamma \leq r_i(\boldsymbol{\beta}) \leq \tau\gamma \\ 1 & \text{if } r_i(\boldsymbol{\beta}) \geq \tau\gamma \end{cases}$$

and introduce

$$w_i = w_i(\boldsymbol{\beta}) = 1 - s_i^2(\boldsymbol{\beta})$$

Therefore,

$$\begin{aligned} H_{\gamma,\tau}(r_i(\boldsymbol{\beta})) &= \frac{1}{2\gamma} w_i r_i^2(\boldsymbol{\beta}) \\ &+ s_i \left[ \frac{1}{2} r_i(\boldsymbol{\beta}) + \frac{1}{4} (1 - 2\tau)\gamma + s_i(r_i(\boldsymbol{\beta})(\tau - \frac{1}{2}) - \frac{1}{4} (1 - 2\tau + 2\tau^2)\gamma) \right] \end{aligned}$$

This equation yields

$$D_{\gamma,\tau}(\boldsymbol{\beta}) = \frac{1}{2\gamma} \mathbf{r}' \mathbf{W}_{\gamma,\tau} \mathbf{r} + \mathbf{v}'(s) \mathbf{r} + c(s)$$

where  $\mathbf{W}_{\gamma,\tau}$  is the diagonal  $n \times n$  matrix with diagonal elements  $w_i(\boldsymbol{\beta})$ ,  $\mathbf{v}'(s) = (s_1((2\tau - 1)s_1 + 1)/2, \dots, s_n((2\tau - 1)s_n + 1)/2)$ ,  $c(s) = \sum [\frac{1}{4}(1 - 2\tau)\gamma s_i - \frac{1}{4}s_i^2(1 - 2\tau + 2\tau^2)\gamma]$ , and  $\mathbf{r}(\boldsymbol{\beta}) = (r_1(\boldsymbol{\beta}), \dots, r_n(\boldsymbol{\beta}))'$ .

The gradient of  $D_{\gamma,\tau}$  is given by

$$D_{\gamma,\tau}^{(1)}(\boldsymbol{\beta}) = -\mathbf{A}[\frac{1}{\gamma}\mathbf{W}_{\gamma,\tau}(\boldsymbol{\beta})\mathbf{r}(\boldsymbol{\beta}) + \mathbf{v}(s)]$$

For  $\boldsymbol{\beta} \in \mathbf{R}^p \setminus B_{\gamma,\tau}$  the Hessian exists and is given by

$$D_{\gamma,\tau}^{(2)}(\boldsymbol{\beta}) = \frac{1}{\gamma}\mathbf{A}\mathbf{W}_{\gamma,\tau}(\boldsymbol{\beta})\mathbf{A}'$$

The gradient is a continuous function in  $\mathbf{R}^p$ , whereas the Hessian is piecewise constant.

Following Madsen and Nielsen (1993), the vector  $\mathbf{s}$  is referred to as a  $\gamma$ -feasible sign vector if there exists  $\boldsymbol{\beta} \in \mathbf{R}^p \setminus B_{\gamma,\tau}$  with  $\mathbf{s}_\gamma(\boldsymbol{\beta}) = \mathbf{s}$ . If  $\mathbf{s}$  is  $\gamma$ -feasible, then  $Q_s$  is defined as the quadratic function  $Q_s(\boldsymbol{\alpha})$  that is derived from  $D_{\gamma,\tau}(\boldsymbol{\beta})$  by substituting  $\mathbf{s}$  for  $\mathbf{s}_\gamma$ . Thus, for any  $\boldsymbol{\beta}$  with  $\mathbf{s}_\gamma = \mathbf{s}$ ,

$$Q_s(\boldsymbol{\alpha}) = \frac{1}{2}(\boldsymbol{\alpha} - \boldsymbol{\beta})' D_{\gamma,\tau}^{(2)}(\boldsymbol{\beta})(\boldsymbol{\alpha} - \boldsymbol{\beta}) + D_{\gamma,\tau}^{(1)}(\boldsymbol{\beta})(\boldsymbol{\alpha} - \boldsymbol{\beta}) + D_{\gamma,\tau}(\boldsymbol{\beta})$$

In the domain  $C_s = \{\boldsymbol{\alpha} | \mathbf{s}_\gamma(\boldsymbol{\alpha}) = \mathbf{s}\}$ ,

$$D_{\gamma,\tau}(\boldsymbol{\alpha}) = Q_s(\boldsymbol{\alpha})$$

For each  $\gamma > 0$  and  $\boldsymbol{\theta} \in \mathbf{R}^p$ , there can be one or several corresponding quadratics,  $Q_s$ . If  $\boldsymbol{\theta} \notin B_{\gamma,\tau}$ , then  $Q_s$  is characterized by  $\boldsymbol{\theta}$  and  $\gamma$ . However, for  $\boldsymbol{\theta} \in B_{\gamma,\tau}$ , the quadratic is not unique. Therefore, the following reference determines the quadratic:

$$(\gamma, \boldsymbol{\theta}, \mathbf{s})$$

Again following Madsen and Nielsen (1993), let  $(\gamma, \boldsymbol{\theta}, \mathbf{s})$  be a *feasible reference* if  $\mathbf{s}$  is a  $\gamma$ -feasible sign vector, where  $\boldsymbol{\theta} \in C_s$ , and let  $(\gamma, \boldsymbol{\theta}, \mathbf{s})$  be a *solution reference* if  $\mathbf{s}$  is feasible and  $\boldsymbol{\theta}$  minimizes  $D_{\gamma,\tau}$ .

The smoothing algorithm for minimizing  $D_{\rho_\tau}$  is based on minimizing  $D_{\gamma,\tau}$  for a set of decreasing  $\gamma$ . For each new value of  $\gamma$ , information from the previous solution is used. Finally, when  $\gamma$  is small enough, a solution can be found by the following modified Newton-Raphson algorithm as stated by Madsen and Nielsen (1993):

1. Find an initial solution reference  $(\gamma, \boldsymbol{\beta}_\gamma, \mathbf{s})$ .
2. Repeat the following substeps until  $\gamma = 0$ .
  - a) Decrease  $\gamma$ .
  - b) Find a solution reference  $(\gamma, \boldsymbol{\beta}_\gamma, \mathbf{s})$ .

$\boldsymbol{\beta}_0$  is the solution.

By default, the initial solution reference is found by letting  $\boldsymbol{\beta}_\gamma$  be the least squares solution. Alternatively, you can specify the initial solution reference with the INEST= option in the PROC QUANTREG statement. Then  $\gamma$  and  $\mathbf{s}$  are chosen according to these initial values.

There are several approaches for determining a decreasing sequence of values of  $\gamma$ . The QUANTREG procedure uses a strategy by Madsen and Nielsen (1993). The computation that it uses is not significant compared to the Newton-Raphson step. You can control the ratio of consecutive decreasing values of  $\gamma$  by specifying the RRATIO= suboption in the ALGORITHM= option in the PROC QUANTREG statement. By default,

$$\text{RRATIO} = \begin{cases} 0.1 & \text{if } n \geq 10,000 \text{ and } p \leq 20 \\ 0.9 & \text{if } \frac{p}{n} \geq 0.1 \text{ or } \{n \leq 5,000 \text{ and } p \geq 300\} \\ 0.5 & \text{otherwise} \end{cases}$$

For the  $L_1$  and quantile regression, it turns out that the smoothing algorithm is very efficient and competitive, especially for a *fat* data set—namely, when  $\frac{p}{n} > 0.05$  and  $\mathbf{A}\mathbf{A}'$  is dense. See Chen (2007) for a complete smoothing algorithm and details.

### Fast Quantile Process Regression

The QUANTILE=FQPR option in the MODEL statement implements a fast quantile process regression (FQPR) method. This method can efficiently fit multiple quantile regression models by using the divide-and-conquer strategy proposed by Yao (2017).

The FQPR method begins by fitting a quantile regression model for a selected quantile level in a specified quantile-level grid of  $q$ -nodes. The quantile level is selected as the closest to 0.5 among all the quantile levels in the grid. Using this fit, FQPR defines two subsets of the data based on whether observed values  $y$  are above or below their linear predictors  $x\beta$  in this regression fit. Then FQPR proceeds to recursively perform separate quantile process regressions on the two subsets.

The successive quantile regression steps of FQPR are thus fit to smaller and smaller data sets. It is this sequence of reductions in problem size that provides the very significant reduction in computational cost that FQPR can achieve. In particular, FQPR can fit a quantile process regression model for  $q$  equally spaced quantiles in the time that it would approximately take to fit just  $\log(q)$  quantile regression models to all the data.

The QUANTILE=FQPR option uses the efficient interior point algorithm to fit single-level quantile regression models as described in the section “[Efficient Interior Point Algorithm](#)” on page 8000.

---

## Confidence Interval

The QUANTREG procedure provides three methods to compute confidence intervals for the regression quantile parameter  $\beta(\tau)$ : sparsity, rank, and resampling. The sparsity method is the most direct and the fastest, but it involves estimation of the sparsity function, which is not robust for data that are not independently and identically distributed. To deal with this problem, the QUANTREG procedure uses a local estimate of the sparsity function to compute a Huber sandwich estimate. The rank method, which computes confidence intervals by inverting the rank score test, does not suffer from this problem. However, the rank method uses the simplex algorithm and is computationally expensive with large data sets. The resampling method, which uses the bootstrap approach, addresses these problems, but at a computation cost.

Based on these properties, the QUANTREG uses a combination of the resampling and rank methods as the default. For data sets that have more than either 5,000 observations or more than 20 variables, the QUANTREG procedure uses the MCMB resampling method; otherwise it uses the rank method. You can request a particular method by using the CI= option in the PROC QUANTREG statement.

## Sparsity

Consider the linear model

$$y_i = \mathbf{x}_i' \boldsymbol{\beta} + \epsilon_i$$

Assume that  $\{\epsilon_i\}$ ,  $i = 1, \dots, n$ , are iid with a distribution  $F$  and a density  $f = F'$ , where  $f(F^{-1}(\tau)) > 0$  in a neighborhood of  $\tau$ . Under some mild conditions,

$$\sqrt{n}(\hat{\boldsymbol{\beta}}(\tau) - \boldsymbol{\beta}(\tau)) \rightarrow N(0, \omega^2(\tau, F)\boldsymbol{\Omega}^{-1})$$

where  $\omega^2(\tau, F) = \tau(1 - \tau)/f^2(F^{-1}(\tau))$  and  $\boldsymbol{\Omega} = \lim_{n \rightarrow \infty} n^{-1} \sum \mathbf{x}_i \mathbf{x}_i'$  (Koenker and Bassett 1982b).

This asymptotic distribution for the regression quantile  $\hat{\boldsymbol{\beta}}(\tau)$  can be used to construct confidence intervals. However, the reciprocal of the density function,

$$s(\tau) = [f(F^{-1}(\tau))]^{-1}$$

which is called the *sparsity function*, must first be estimated.

Because

$$s(t) = \frac{d}{dt} F^{-1}(t)$$

$s(t)$  can be estimated by the difference quotient of the empirical quantile function—that is,

$$\hat{s}_n(t) = [\hat{F}_n^{-1}(t + h_n) - \hat{F}_n^{-1}(t - h_n)]/2h_n$$

where  $\hat{F}_n$  is an estimate of  $F^{-1}$  and  $h_n$  is a bandwidth that tends to 0 as  $n \rightarrow \infty$ .

The QUANTREG procedure provides two bandwidth methods. The Bofinger bandwidth

$$h_n = n^{-1/5} \left( \frac{4.5s^2(t)}{(s^{(2)}(t))^2} \right)^{1/5}$$

is an optimizer of mean squared error for standard density estimation. The Hall-Sheather bandwidth

$$h_n = n^{-1/3} z_\alpha^{2/3} \left( \frac{1.5s(t)}{s^{(2)}(t)} \right)^{1/3}$$

is based on Edgeworth expansions for studentized quantiles, where  $s^{(2)}(t)$  is the second derivative of  $s(t)$  and  $z_\alpha$  satisfies  $\Phi(z_\alpha) = 1 - \alpha/2$  for the construction of  $1 - \alpha$  confidence intervals. The following quantity is not sensitive to  $f$  and can be estimated by assuming  $f$  is Gaussian:

$$\frac{s(t)}{s^{(2)}(t)} = \frac{f^2}{2(f^{(1)}/f)^2 + [(f^{(1)}/f)^2 - f^{(2)}/f]}$$

$F^{-1}$  can be estimated in either of the following ways:

- by the empirical quantile function of the residuals from the quantile regression fit,

$$\hat{F}^{-1}(t) = r_{(i)}, \text{ for } t \in [(i-1)/n, i/n),$$

- by the empirical quantile function of regression proposed by Bassett and Koenker (1982),

$$\hat{F}^{-1}(t) = \bar{\mathbf{x}}' \hat{\boldsymbol{\beta}}(t)$$

The QUANTREG procedure interpolates the first empirical quantile function and produces the piecewise linear version:

$$\hat{F}^{-1}(t) = \begin{cases} r_{(1)} & \text{if } t \in [0, 1/2n) \\ \lambda r_{(i+1)} + (1 - \lambda)r_{(i)} & \text{if } t \in [(2i - 1)/2n, (2i + 1)/2n) \\ r_{(n)} & \text{if } t \in [(2n - 1), 1] \end{cases}$$

$\hat{F}^{-1}$  is set to a constant if  $t \pm h_n$  falls outside  $[0, 1]$ .

This estimator of the sparsity function is sensitive to the iid assumption. Alternately, Koenker and Machado (1999) consider the non-iid case. By assuming local linearity of the conditional quantile function  $Q(\tau|x)$  in  $x$ , they propose a local estimator of the density function by using the difference quotient. A Huber sandwich estimate of the covariance and standard error is computed and used to construct the confidence intervals. One difficulty with this method is the selection of the bandwidth when using the difference quotient. With a small sample size, either the Bofinger or the Hall-Sheather bandwidth tends to be too large to assure local linearity of the conditional quantile function. The QUANTREG procedure uses a heuristic bandwidth selection in these cases.

By default, the QUANTREG procedure computes non-iid confidence intervals. You can request iid confidence intervals by specifying the IID option in the PROC QUANTREG statement.

## Inversion of Rank Tests

The classical theory of rank tests can be extended to test the hypothesis  $H_0: \beta_2 = \eta$  in the linear regression model  $\mathbf{y} = \mathbf{X}_1\beta_1 + \mathbf{X}_2\beta_2 + \boldsymbol{\epsilon}$ . Here,  $(\mathbf{X}_1, \mathbf{X}_2) = \mathbf{A}'$ . See Gutenbrunner and Jureckova (1992) for more details. By inverting this test, confidence intervals can be computed for the regression quantiles that correspond to  $\beta_2$ .

The rank score function  $\hat{a}_n(t) = (\hat{a}_{n1}(t), \dots, \hat{a}_{nn}(t))$  can be obtained by solving the dual problem:

$$\max_a \{(\mathbf{y} - \mathbf{X}_2\boldsymbol{\eta})' \mathbf{a} | \mathbf{X}_1' \mathbf{a} = (1 - t)\mathbf{X}_1' \mathbf{e}, \mathbf{a} \in [0, 1]^n\}$$

For a fixed quantile  $\tau$ , integrating  $\hat{a}_{ni}(t)$  with respect to the  $\tau$ -quantile score function

$$\varphi_\tau(t) = \tau - I(t < \tau)$$

yields the  $\tau$ -quantile scores

$$\hat{b}_{ni} = - \int_0^1 \varphi_\tau(t) d\hat{a}_{ni}(t) = \hat{a}_{ni}(\tau) - (1 - \tau)$$

Under the null hypothesis  $H_0: \beta_2 = \eta$ ,

$$S_n(\eta) = n^{-1/2} \mathbf{X}_2' \hat{\mathbf{b}}_n(\eta) \rightarrow N(0, \tau(1 - \tau) \boldsymbol{\Omega}_n)$$

for large  $n$ , where  $\boldsymbol{\Omega}_n = n^{-1} \mathbf{X}_2' (\mathbf{I} - \mathbf{X}_1 (\mathbf{X}_1' \mathbf{X}_1)^{-1} \mathbf{X}_1') \mathbf{X}_2$ .

Let

$$T_n(\eta) = \frac{1}{\sqrt{\tau(1-\tau)}} S_n(\eta) \mathbf{\Omega}_n^{-1/2}$$

Then  $T_n(\hat{\beta}_2(\tau)) = 0$  from the constraint  $\mathbf{A}\hat{\mathbf{a}} = (1-\tau)\mathbf{A}\mathbf{e}$  in the full model. In order to obtain confidence intervals for  $\beta_2$ , a critical value can be specified for  $T_n$ . The dual vector  $\hat{\mathbf{a}}_n(\eta)$  is a piecewise constant in  $\eta$ , and  $\eta$  can be altered without compromising the optimality of  $\hat{\mathbf{a}}_n(\eta)$  as long as the signs of the residuals in the primal quantile regression problem do not change. When  $\eta$  gets to such a boundary, the solution does change. But it can be restored by taking one simplex pivot. The process can continue in this way until  $T_n(\eta)$  exceeds the specified critical value. Because  $T_n(\eta)$  is piecewise constant, interpolation can be used to obtain the desired level of confidence interval (Koenker and d'Orey 1994).

## Resampling

The bootstrap can be implemented to compute confidence intervals for regression quantile estimates. As in other regression applications, both the residual bootstrap and the  $xy$ -pair bootstrap can be used. The former assumes iid random errors and resamples from the residuals, whereas the latter resamples  $xy$  pairs and accommodates some forms of heteroscedasticity. Koenker (1994) considered a more interesting resampling mechanism, resampling directly from the full regression quantile process, which he called the Heqf bootstrap.

In contrast with these bootstrap methods, Parzen, Wei, and Ying (1994) observed that the following estimating equation for the  $\tau$  regression quantile is a pivotal quantity for the  $\tau$  quantile regression parameter  $\beta_\tau$ :

$$S(\boldsymbol{\beta}) = n^{-1/2} \sum_{i=1}^n \mathbf{x}_i (\tau - I(y_i \leq \mathbf{x}_i' \boldsymbol{\beta}))$$

In other words, the distribution of  $\mathbf{S}(\boldsymbol{\beta})$  can be generated exactly by a random vector  $\mathbf{U}$ , which is a weighted sum of independent, re-centered Bernoulli variables. They further showed that for large  $n$ , the distribution of  $\hat{\boldsymbol{\beta}}(\tau) - \boldsymbol{\beta}_\tau$  can be approximated by the conditional distribution of  $\hat{\boldsymbol{\beta}}_U - \hat{\boldsymbol{\beta}}_n(\tau)$ , where  $\hat{\boldsymbol{\beta}}_U$  solves an augmented quantile regression problem by using  $n+1$  observations that have  $\mathbf{x}_{n+1} = -n^{-1/2}\mathbf{u}/\tau$  and  $y_{n+1}$  sufficiently large for a given realization of  $\mathbf{u}$ . By exploiting the asymptotically pivotal role of the quantile regression “gradient condition,” this approach also achieves some robustness to certain heteroscedasticity.

Although the bootstrap method by Parzen, Wei, and Ying (1994) is much simpler, it is too time-consuming for relatively large data sets, especially for high-dimensional data sets. The QUANTREG procedure implements a new, general resampling method developed by He and Hu (2002), which is called the Markov chain marginal bootstrap (MCMB). For quantile regression, the MCMB method has the advantage that it solves  $p$  one-dimensional equations instead of solving  $p$ -dimensional equations, as the previous bootstrap methods do. This greatly improves the feasibility of the resampling method in computing confidence intervals for regression quantiles.

---

## Covariance-Correlation

You can specify the COVB and CORRB options in the MODEL statement to request covariance and correlation matrices for the estimated parameters.

The QUANTREG procedure provides two methods for computing the covariance and correlation matrices of the estimated parameters: an asymptotic method and a bootstrap method. Bootstrap covariance and correlation

matrices are computed when resampling confidence intervals are computed. Asymptotic covariance and correlation matrices are computed when asymptotic confidence intervals are computed. The rank method for confidence intervals does not provide a covariance-correlation estimate.

### Asymptotic Covariance-Correlation

This method corresponds to the sparsity method for the confidence intervals. For the sparsity function in the computation of the asymptotic covariance and correlation, the QUANTREG procedure provides both iid and non-iid estimates. By default, the QUANTREG procedure computes non-iid estimates.

### Bootstrap Covariance-Correlation

This method corresponds to the resampling method for the confidence intervals. The Markov chain marginal bootstrap (MCMB) method is used.

---

## Linear Test

Consider the linear model

$$y_i = \mathbf{x}'_{1i}\boldsymbol{\beta}_1 + \mathbf{x}'_{2i}\boldsymbol{\beta}_2 + \epsilon_i$$

where  $\boldsymbol{\beta}_1$  and  $\boldsymbol{\beta}_2$  are  $p$ - and  $q$ -dimensional unknown parameters and  $\{\epsilon_i\}$ ,  $i = 1, \dots, n$ , are errors with unknown density function  $f_i$ . Let  $\mathbf{x}'_i = (\mathbf{x}'_{1i}, \mathbf{x}'_{2i})$ , and let  $\hat{\boldsymbol{\beta}}_1(\tau)$  and  $\hat{\boldsymbol{\beta}}_2(\tau)$  be the parameter estimates for  $\boldsymbol{\beta}_1$  and  $\boldsymbol{\beta}_2$ , respectively at the  $\tau$  quantile. The covariance matrix  $\boldsymbol{\Omega}$  for the parameter estimates is partitioned correspondingly as  $\boldsymbol{\Omega}_{ij}$  with  $i = 1, 2$ ;  $j = 1, 2$ ; and  $\boldsymbol{\Omega}^{22} = (\boldsymbol{\Omega}_{22} - \boldsymbol{\Omega}_{21}\boldsymbol{\Omega}_{11}^{-1}\boldsymbol{\Omega}_{12})^{-1}$ .

### Testing Effects of Covariates

Three tests are available in the QUANTREG procedure for the linear null hypothesis  $H_0 : \boldsymbol{\beta}_2 = 0$  at the  $\tau$  quantile:

- The Wald test statistic, which is based on the estimated coefficients for the unrestricted model, is given by

$$T_W(\tau) = \hat{\boldsymbol{\beta}}'_2(\tau) \hat{\boldsymbol{\Sigma}}(\tau)^{-1} \hat{\boldsymbol{\beta}}_2(\tau)$$

where  $\hat{\boldsymbol{\Sigma}}(\tau)$  is an estimator of the covariance of  $\hat{\boldsymbol{\beta}}_2(\tau)$ . The QUANTREG procedure provides two estimators for the covariance, as described in the previous section. The estimator that is based on the asymptotic covariance is

$$\hat{\boldsymbol{\Sigma}}(\tau) = \frac{1}{n} \hat{\omega}(\tau)^2 \boldsymbol{\Omega}^{22}$$

where  $\hat{\omega}(\tau) = \sqrt{\tau(1-\tau)}\hat{s}(\tau)$  and  $\hat{s}(\tau)$  is the estimated sparsity function. The estimator that is based on the bootstrap covariance is the empirical covariance of the MCMB samples.

- The likelihood ratio test is based on the difference between the objective function values in the restricted and unrestricted models. Let  $D_0(\tau) = \sum \rho_\tau(y_i - \mathbf{x}_i \hat{\boldsymbol{\beta}}(\tau))$ , and let  $D_1(\tau) = \sum \rho_\tau(y_i - \mathbf{x}_{1i} \hat{\boldsymbol{\beta}}_1(\tau))$ . Set

$$T_{LR}(\tau) = 2(\tau(1 - \tau)\hat{s}(\tau))^{-1}(D_1(\tau) - D_0(\tau))$$

where  $\hat{s}(\tau)$  is the estimated sparsity function.

- The rank test statistic is given by

$$T_R(\tau) = \mathbf{S}_n' \mathbf{M}_n^{-1} \mathbf{S}_n / A^2(\varphi)$$

where

$$\mathbf{S}_n = n^{-1/2}(\mathbf{X}_2 - \hat{\mathbf{X}}_2)' \hat{\mathbf{b}}_n$$

$$\Psi = \text{diag}(f_i(Q_{y_i}(\tau | \mathbf{x}_{1i}, \mathbf{x}_{2i})))$$

$$\hat{\mathbf{X}}_2 = \mathbf{X}_1(\mathbf{X}_1' \Psi \mathbf{X}_1)^{-1} \mathbf{X}_1' \mathbf{X}_2$$

$$\mathbf{M}_n = (\mathbf{X}_2 - \hat{\mathbf{X}}_2)(\mathbf{X}_2 - \hat{\mathbf{X}}_2)' / n$$

$$\hat{\mathbf{b}}_{ni} = \int_0^1 \hat{\mathbf{a}}_{ni}(t) d\varphi(t)$$

$$\hat{\mathbf{a}}(t) = \max_{\mathbf{a}} \{\mathbf{y}' \mathbf{a} | \mathbf{X}_1' \mathbf{a} = (1 - t) \mathbf{X}_1' \mathbf{e}, \mathbf{a} \in [0, 1]^n\}$$

$$A^2(\varphi) = \int_0^1 (\varphi(t) - \bar{\varphi}(t))^2 dt$$

$$\bar{\varphi}(t) = \int_0^1 \varphi(t) dt$$

and  $\varphi(t)$  is one of the following score functions:

- Wilcoxon scores:  $\phi(t) = t - 1/2$
- normal scores:  $\phi(t) = \Phi^{-1}(t)$ , where  $\Phi$  is the normal distribution function
- sign scores:  $\phi(t) = 1/2 \text{sign}(t - 1/2)$
- tau scores:  $\phi_\tau(t) = \tau - I(t < \tau)$ .

The rank test statistic  $T_R(\tau)$ , unlike Wald tests or likelihood ratio tests, requires no estimation of the nuisance parameter  $f_i$  under iid error models (Gutenbrunner et al. 1993).

Koenker and Machado (1999) prove that the three test statistics ( $T_W(\tau)$ ,  $T_{LR}(\tau)$ , and  $T_R(\tau)$ ) are asymptotically equivalent and that their distributions converge to  $\chi_q^2$  under the null hypothesis, where  $q$  is the dimension of  $\beta_2$ .

## Testing for Heteroscedasticity

After you obtain the parameter estimates for several quantiles specified in the MODEL statement, you can test whether there are significant differences for the estimates for the same covariates across the quantiles. For example, if you want to test whether the parameters  $\beta_2$  are the same across quantiles, the null hypothesis  $H_0$  can be written as  $\beta_2(\tau_1) = \dots = \beta_2(\tau_k)$ , where  $\tau_j$ ,  $j = 1, \dots, k$ , are the quantiles specified in the MODEL statement. See Koenker and Bassett (1982a) for details.



## Leverage Point and Outlier Detection

The QUANTREG procedure uses robust multivariate location and scale estimates for leverage-point detection.

Mahalanobis distance is defined as

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

where  $\bar{\mathbf{x}} = \frac{1}{n} \sum_{i=1}^n \mathbf{x}_i$  and  $\bar{\mathbf{C}}(\mathbf{A}) = \frac{1}{n-1} \sum_{i=1}^n (\mathbf{x}_i - \bar{\mathbf{x}})' (\mathbf{x}_i - \bar{\mathbf{x}})$  are the empirical multivariate location and scale, respectively. Here,  $\mathbf{x}_i = (\mathbf{x}_{i1}, \dots, \mathbf{x}_{i(p-1)})'$  does not include the intercept variable. The relationship between the Mahalanobis distance  $MD(\mathbf{x}_i)$  and the matrix  $\mathbf{H} = (h_{ij}) = \mathbf{A}'(\mathbf{A}\mathbf{A}')^{-1}\mathbf{A}$  is

$$h_{ii} = \frac{1}{n-1} MD_i^2 + \frac{1}{n}$$

Robust distance is defined as

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

where  $\mathbf{T}(\mathbf{A})$  and  $\mathbf{C}(\mathbf{A})$  are robust multivariate location and scale estimates that are computed according to the minimum covariance determinant (MCD) method of Rousseeuw and Van Driessen (1999).

These distances are used to detect leverage points. You can use the LEVERAGE and DIAGNOSTICS options in the MODEL statement to request leverage-point and outlier diagnostics, respectively. Two new variables, Leverage and Outlier, respectively, are created and saved in an output data set that is specified in the OUTPUT statement.

Let  $C(p) = \sqrt{\chi_{p;1-\alpha}^2}$  be the cutoff value. The variable LEVERAGE is defined as

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

You can specify a cutoff value in the LEVERAGE option in the MODEL statement.

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

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

You can specify the multiplier  $k$  of the cutoff value in the CUTOFF= option in the MODEL statement. You can specify the scale  $\sigma$  in the SCALE= option in the MODEL statement. By default,  $k = 3$  and the scale  $\sigma$  is computed as the corrected median of the absolute residuals:

$$\sigma = \text{median}\{|r_i|/\beta_0, i = 1, \dots, n\}$$

where  $\beta_0 = \Phi^{-1}(0.75)$  is an adjustment constant for consistency when the normal distribution is used.

An ODS table called DIAGNOSTICS contains the Leverage and Outlier variables.

---

## INEST= Data Set

The INEST= data set specifies initial estimates for all the parameters in the model. 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 for each BY group. If there is more than one observation in one BY group, the first one read is used for that BY group.

If the INEST= data set also contains the `_TYPE_` variable, only observations with the `_TYPE_` value 'PARMS' are used as starting values.

You can specify starting values for the interior point algorithm or the smoothing algorithm in the INEST= data set. The INEST= data set has the same structure as the OUTEST= data set, but it is not required to have all the variables or observations that appear in the OUTEST= data set. One simple use of the INEST= option is passing the previous OUTEST= data set directly to the next model as an INEST= data set, assuming that the two models have the same parameterization. If you specify more than one quantile in the `MODEL` statement, the same initial values are used for all quantiles.

---

## OUTEST= Data Set

The OUTEST= data set contains parameter estimates for the specified model with all quantiles. A set of observations is created for each quantile specified. You can also specify a label in the `MODEL` statement to distinguish between the estimates for different models that are used by the QUANTREG procedure.

If the QUANTREG procedure does not produce valid solutions, the parameter estimates are set to missing in the OUTEST data set.

If this data set is created, it contains all the variables that are specified in the `MODEL` statement and the BY statement. Each observation consists of parameter values for a specified quantile, and the dependent variable has the value `-1`.

The following variables are also added to the data set:

<code>_MODEL_</code>	a character variable of length 8 that contains the label of the <code>MODEL</code> statement, if present. Otherwise, the variable's value is blank.
<code>_ALGORITHM_</code>	a character variable of length 8 that contains the name of the algorithm that is used for computing the parameter estimates, either <code>SIMPLEX</code> , <code>INTERIOR</code> , or <code>SMOOTH</code> .
<code>_TYPE_</code>	a character variable of length 8 that contains the type of the observation. This variable is fixed as <code>PARMS</code> to indicate that the observation includes parameter estimates.
<code>_STATUS_</code>	a character variable of length 12 that contains the status of model fitting (either <code>NORMAL</code> , <code>NOUNIQUE</code> , or <code>NOVALID</code> ).
<code>Intercept</code>	a numeric variable that contains the intercept parameter estimates.
<code>_QUANTILE_</code>	a numeric variable that contains the specified quantile levels.

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

## Computational Resources

The various algorithms need different amounts of memory for working space. Let  $p$  be the number of parameters that are estimated,  $n$  be the number of observations that are used in the model estimation, and  $s$  be the size (in bytes) of the double data type.

For the simplex algorithm, the minimum working space (in bytes) that is needed is

$$(2np + 6n + 10p)s$$

For the interior point algorithm, the minimum working space (in bytes) that is needed is

$$(np + p^2 + 13n + 4p)s$$

For the smoothing algorithm, the minimum working space (in bytes) that is needed is

$$(np + p^2 + 6n + 4p)s$$

For the last two algorithms, if you want to use preprocessing, the following additional amount of working space (in bytes) is needed:

$$(np + 6n + 2p)s$$

If sufficient space is available, the input data set is kept in memory. Otherwise, the input data set is reread as necessary, and the execution time of the procedure increases substantially.

## ODS Table Names

The QUANTREG procedure assigns a name to each table it creates. You can specify these names when you use the Output Delivery System (ODS) to select tables and create output data sets. These names are listed in the [Table 98.8](#).

**Table 98.8** ODS Tables Produced in PROC QUANTREG

ODS Table Name	Description	Statement	Option
AvgParameterEst	Average parameter estimates	MODEL	FQPR
AvgObjFunction	Average objective function	MODEL	FQPR
ClassLevels	Classification variable levels	CLASS	Default
CorrB	Parameter estimate correlation matrix	MODEL	CORRB
CovB	Parameter estimate covariance matrix	MODEL	COVB
Diagnostics	Outlier diagnostics	MODEL	DIAGNOSTICS
DiagSummary	Summary of the outlier diagnostics	MODEL	DIAGNOSTICS
IPIterHistory	Iteration history (interior point)	MODEL	ITPRINT
ModelInfo	Model information	MODEL	Default
NObs	Number of observations	PROC	Default
ObjFunction	Objective function	MODEL	Default
ParameterEstimates	Parameter estimates	MODEL	Default
ParmInfo	Parameter indices	MODEL	Default

**Table 98.8** (continued)

ODS Table Name	Description	Statement	Option
PerfSettings	Performance settings	PERFORMANCE	DETAILS
ProcessEst	Quantile process estimates	MODEL	QUANTILE=
ProcessObj	Objective function for quantile process	MODEL	QUANTILE=
SMIterHistory	Iteration history (smoothing)	MODEL	ITPRINT
SummaryStatistics	Summary statistics for model variables	MODEL	Default
Tests	Results for tests	TEST	Default
ScalableTiming	Timing details	PERFORMANCE	DETAILS

## ODS Graphics

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

Before you create graphs, ODS Graphics must be enabled (for example, by specifying the ODS GRAPHICS ON statement). For more information about enabling and disabling ODS Graphics, see the section “[Enabling and Disabling ODS Graphics](#)” on page 615 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 614 in Chapter 21, “[Statistical Graphics Using ODS](#).”

For a single quantile, two plots are particularly useful in revealing outliers and leverage points: a scatter plot of the standardized residuals for the specified quantile against the robust distances and a scatter plot of the robust distances against the classical Mahalanobis distances. You can request these two plots by using the PLOT=RDPLOT and PLOT=DDPLOT options, respectively.

You can also request a normal quantile-quantile plot and a histogram of the standardized residuals for the specified quantile by using the PLOT=QQPLOT and PLOT=HISTOGRAM options, respectively.

You can request a plot of fitted conditional quantiles by the single continuous variable that is specified in the model by using the PLOT=FITPLOT option.

All these plots can be requested by specifying corresponding plot options in either the PROC QUANTREG statement or the MODEL statement. If you specify same plot options in both statements, options in the PROC QUANTREG statement override options in the MODEL statement.

You can specify the PLOT=QUANTPLOT option only in the MODEL statement to request a quantile process plot with confidence bands.

The plot options in the PROC QUANTREG statement and the MODEL statement are summarized in [Table 98.9](#). See the [PLOT=](#) option in the PROC QUANTREG statement and the [PLOT=](#) option in the MODEL statement for details.

**Table 98.9** Options for Plots

<b>Keyword</b>	<b>Plot</b>
ALL	All appropriate plots
DDPLOT	Robust distance versus Mahalanobis distance
FITPLOT	Conditional quantile fit versus independent variable
HISTOGRAM	Histogram of standardized robust residuals
NONE	No plot
QUANTPLOT	Scatter plot of regression quantile
QQPLOT	Q-Q plot of standardized robust residuals
RDPLOT	Standardized robust residual versus robust distance

The following subsections provide information about these graphs.

## ODS Graph Names

The QUANTREG procedure assigns a name to each graph it creates. You can use these names to refer to the graphs when you use ODS. The names along with the required statements and options are listed in Table 98.10.

**Table 98.10** Graphs Produced by PROC QUANTREG

<b>ODS Graph Name</b>	<b>Plot Description</b>	<b>Statement</b>	<b>Option</b>
DDPlot	Robust distance versus Mahalanobis distance	PROC MODEL	DDPLOT
FitPlot	Quantile fit versus independent variable	PROC MODEL	FITPLOT
Histogram	Histogram of standardized robust residuals	PROC MODEL	HISTOGRAM
QQPlot	Q-Q plot of standardized robust residuals	PROC MODEL	QQPLOT
QuantPanel	Panel of quantile plots with confidence limits	MODEL	QUANTPLOT
QuantPlot	Scatter plot for regression quantiles with confidence limits	MODEL	QUANTPLOT UNPACK
RDPlot	Standardized robust residual versus robust distance	PROC MODEL	RDPLOT

## Fit Plot

When the model has a single independent continuous variable (with or without the intercept), the QUANTREG procedure automatically creates a plot of fitted conditional quantiles against this independent variable for one or more quantiles that are specified in the MODEL statement.

The following example reuses the trout data set in the section “[Analysis of Fish-Habitat Relationships](#)” on page 7968 to show the fit plot for one or several quantiles:

```
ods graphics on;

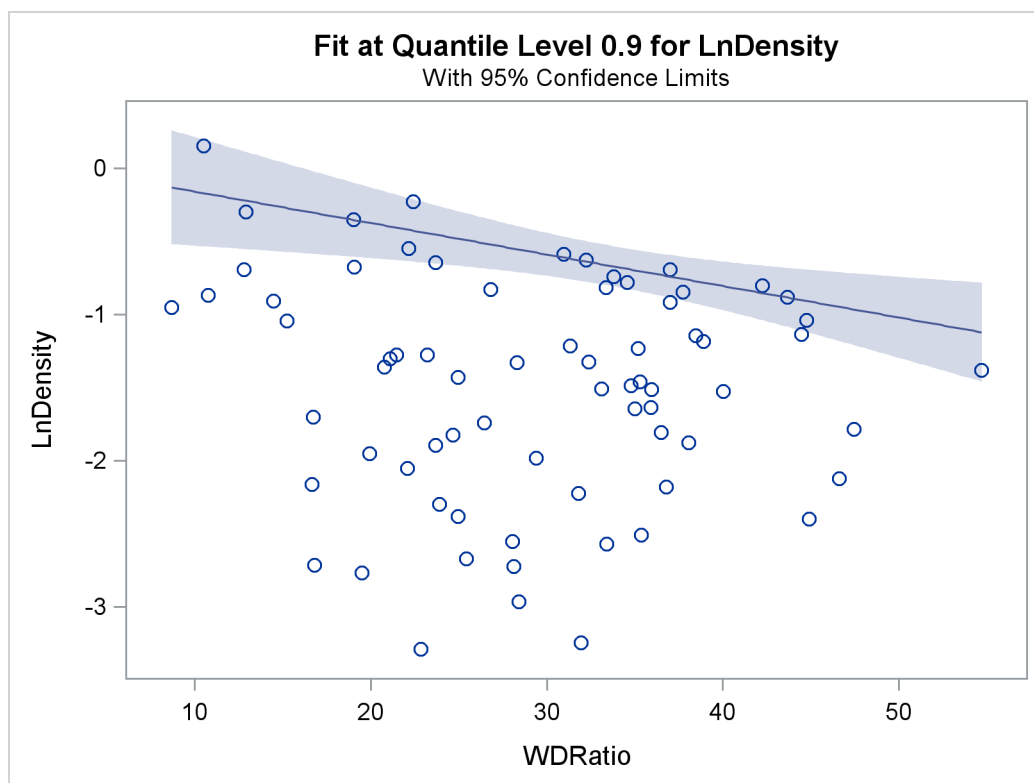
proc quantreg data=trout ci=resampling;
  model LnDensity = WDRatio / quantile=0.9 seed=1268;
run;

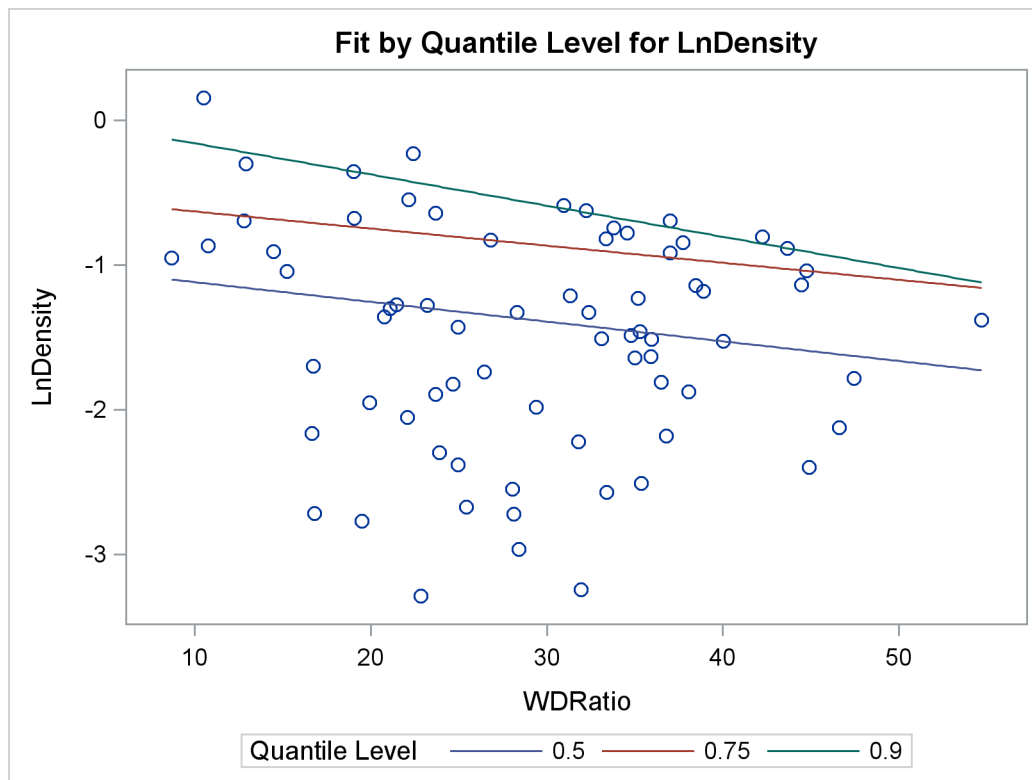
proc quantreg data=trout ci=resampling;
  model LnDensity = WDRatio / quantile=0.5 0.75 0.9 seed=1268;
run;
```

For a single quantile, the confidence limits for the fitted conditional quantiles are also plotted if you specify the `CI=RESAMPLING` or `CI=SPARSITY` option. (See [Figure 98.14](#).) For multiple quantiles, confidence limits are not plotted by default. (See [Figure 98.15](#).) You can add the confidence limits on the plot by specifying the option `PLOT=FITPLOT(SHOWLIMITS)`.

The QUANTREG procedure also provides fit plots for quantile regression splines and polynomials if they are based on a single continuous variable. (See [Example 98.4](#) and [Example 98.5](#) for some examples.)

**Figure 98.14** Fit Plot with Confidence Limits



**Figure 98.15** Fit Plot for Multiple Quantiles

### Quantile Process Plot

A quantile process plot is a scatter plot of an estimated regression parameter against a quantile. You can request this plot by specifying the `PLOT=QUANTPLOT` option in the `MODEL` statement when multiple regression quantiles are computed or when the entire quantile process is computed. Quantile process plots are often used to check model variations at different quantiles, which is usually called model heterogeneity.

By default, panels are used to hold multiple process plots (up to four in each panel). You can use the `UNPACK` option to request individual process plots. [Figure 98.10](#) in the section “[Analysis of Fish-Habitat Relationships](#)” on page 7968 shows a panel that includes two quantile process plots. [Output 98.2.9](#) in [Example 98.2](#) shows a single quantile process plot. [Example 98.3](#) demonstrates more quantile process plots and their usage.

### Distance-Distance Plot

The distance-distance plot (DDPLOT) is mainly used for leverage-point diagnostics. It is a scatter plot of the robust distances against the classical Mahalanobis distances for the continuous independent variables. For more information about the robust distance, see the section “[Leverage Point and Outlier Detection](#)” on page 8009. If a classification variable is specified in the model, this plot is not created.

You can use the `PLOT=DDPLOT` option to request this plot. The following statements use the growth data set in [Example 98.2](#) to create a single plot, which is shown in [Output 98.2.4](#) in [Example 98.2](#):

```
proc quantreg data=growth ci=resampling plot=ddplot;
  model GDP = lgdp2 mse2 fse2 fhe2 mhe2 lexp2
           lintr2 gedy2 Iy2 gcony2 lblakp2 pol2 ttrad2
           / quantile=.5 diagnostics leverage(cutoff=8) seed=1268;
  id Country;
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 98.4](#).

## Residual-Distance Plot

The residual-distance plot (RDPLOT) is used for both outlier and leverage-point diagnostics. It is a scatter plot of the standardized residuals against the robust distances. For more information about the robust distance, see the section “[Leverage Point and Outlier Detection](#)” on page 8009. If a classification variable is specified in the model, this plot is not created.

You can use the PLOT=RDPLOT option to request this plot. The following statements use the growth data set in [Example 98.2](#) to create a single plot, which is shown in [Output 98.2.3](#) in [Example 98.2](#):

```
proc quantreg data=growth ci=resampling plot=rdplot;
  model GDP = lgdp2 mse2 fse2 fhe2 mhe2 lexp2
           lintr2 gedy2 Iy2 gcony2 lblakp2 pol2 ttrad2
           / quantile=.5 diagnostics leverage(cutoff=8) seed=1268;
  id Country;
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 98.4](#).

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

## Histogram and Q-Q Plot

PROC QUANTREG produces a histogram and a Q-Q plot for the standardized residuals. The histogram is superimposed with a normal density curve and a kernel density curve. Using the growth data set in [Example 98.2](#), the following statements create the plot that is shown in [Output 98.2.5](#) in [Example 98.2](#):

```
proc quantreg data=growth ci=resampling plot=histogram;
  model GDP = lgdp2 mse2 fse2 fhe2 mhe2 lexp2
           lintr2 gedy2 Iy2 gcony2 lblakp2 pol2 ttrad2
           / quantile=.5 diagnostics leverage(cutoff=8) seed=1268;
  id Country;
run;
```



---

## Examples: QUANTREG Procedure

---

### Example 98.1: Comparison of Algorithms

This example illustrates and compares the three algorithms for regression estimation available in the QUANTREG procedure. The simplex algorithm is the default because of its stability. Although this algorithm is slower than the interior point and smoothing algorithms for large data sets, the difference is not as significant for data sets with fewer than 5,000 observations and 50 variables. The simplex algorithm can also compute the entire quantile process, which is shown in [Example 98.2](#).

The following statements generate 1,000 random observations. The first 950 observations are from a linear model, and the last 50 observations are significantly biased in the y-direction. In other words, 5% 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 > 950 then y=100 + 10*e;
    else y=10 + 5*x1 + 3*x2 + 0.5 * e;
    output;
  end;
run;
```

The following statements invoke the QUANTREG procedure to fit a median regression model with the default simplex algorithm. They produce the results that are shown in [Output 98.1.1](#) through [Output 98.1.3](#).

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

[Output 98.1.1](#) displays model information and summary statistics for variables in the model. It indicates that the simplex algorithm is used to compute the optimal solution and that the rank method is used to compute confidence intervals of the parameters.

By default, the QUANTREG procedure fits a median regression model. This is indicated by the quantile value 0.5 in [Output 98.1.2](#), which also displays the objective function value and the predicted value of the response at the means of the covariates.

[Output 98.1.3](#) displays parameter estimates and confidence limits. These estimates are reasonable, which indicates that median regression is robust to the 50 outliers.

**Output 98.1.1** Model Fit Information and Summary Statistics from the Simplex Algorithm**The QUANTREG Procedure**

Model Information						
Data Set		WORK.A				
Dependent Variable		y				
Number of Independent Variables		2				
Number of Observations		1000				
Optimization Algorithm		Simplex				
Method for Confidence Limits		Inv_Rank				

Summary Statistics						
Variable	Q1	Median	Q3	Mean	Standard Deviation	MAD
x1	-0.6546	0.0230	0.7099	0.0222	0.9933	1.0085
x2	-0.7891	-0.0747	0.6839	-0.0401	1.0394	1.0857
y	6.1045	10.6936	14.9569	14.4864	20.4087	6.5696

**Output 98.1.2** Quantile and Objective Function from the Simplex Algorithm

Quantile Level and Objective Function	
Quantile Level	0.5
Objective Function	2441.1927
Predicted Value at Mean	10.0259

**Output 98.1.3** Parameter Estimates from the Simplex Algorithm

Parameter Estimates				
Parameter	DF	Estimate	95% Confidence Limits	
Intercept	1	10.0364	9.9959	10.0756
x1	1	5.0106	4.9602	5.0388
x2	1	3.0294	2.9944	3.0630

The following statements refit the model by using the interior point algorithm:

```
proc quantreg algorithm=interior(tolerance=1e-6)
    ci=none data=a;
    model y = x1 x2 / itprint nosummary;
run;
```

The TOLERANCE= option specifies the stopping criterion for convergence of the interior point algorithm, which is controlled by the duality gap. Although the default criterion is 1E-8, the value 1E-6 is often sufficient. The ITPRINT option requests the iteration history for the algorithm. The option CI=NONE suppresses the computation of confidence limits, and the option NOSUMMARY suppresses the table of summary statistics.

Output 98.1.4 displays model fit information.

**Output 98.1.4** Model Fit Information from the Interior Point Algorithm**The QUANTREG Procedure**

Model Information	
Data Set	WORK.A
Dependent Variable	y
Number of Independent Variables	2
Number of Observations	1000
Optimization Algorithm	Interior

Output 98.1.5 displays the iteration history of the interior point algorithm. Note that the duality gap is less than  $1\text{E}-6$  in the final iteration. The table also provides the number of iterations, the number of corrections, the primal step length, the dual step length, and the objective function value at each iteration.

**Output 98.1.5** Iteration History for the Interior Point Algorithm**The QUANTREG Procedure**  
**Quantile Level = 0.5**

Iteration History of Interior Point Algorithm				
Iter	Duality Gap	Primal Step	Dual Step	Objective Function
1	2622.675	0.3113	0.4910	3303.469
2	3214.640	0.0427	1.0000	2461.377
3	1126.899	0.9882	0.3653	2451.134
4	760.887	0.3381	1.0000	2442.810
5	77.102902	1.0000	0.8916	2441.263
6	8.436664	0.9370	0.8381	2441.208
7	1.828685	0.8375	0.7674	2441.199
8	0.405843	0.6980	0.8636	2441.195
9	0.095499	0.9438	0.5955	2441.193
10	0.0066528	0.9818	0.9304	2441.193
11	0.00022482	0.9179	0.9994	2441.193
12	5.44787E-8	1.0000	1.0000	2441.193

Output 98.1.6 displays the parameter estimates that are obtained by using the interior point algorithm. These estimates are identical to those obtained by using the simplex algorithm.

**Output 98.1.6** Parameter Estimates from the Interior Point Algorithm

Parameter Estimates		
Parameter	DF	Estimate
Intercept	1	10.0364
x1	1	5.0106
x2	1	3.0294

The following statements refit the model by using the smoothing algorithm. They produce the results that are shown in Output 98.1.7 through Output 98.1.9.

```
proc quantreg algorithm=smooth(rratio=.5) ci=none data=a;
  model y = x1 x2 / itprint nosummary;
run;
```

The RRATIO= option controls the reduction speed of the threshold. [Output 98.1.7](#) displays the model fit information.

**Output 98.1.7** Model Fit Information from the Smoothing Algorithm

**The QUANTREG Procedure**

Model Information	
Data Set	WORK.A
Dependent Variable	y
Number of Independent Variables	2
Number of Observations	1000
Optimization Algorithm	Smooth

[Output 98.1.8](#) displays the iteration history of the smoothing algorithm. The threshold controls the convergence. Note that the thresholds decrease by a factor of at least 0.5, which is the value specified in the RRATIO= option. The table also provides the number of iterations, the number of factorizations, the number of full updates, the number of partial updates, and the objective function value in each iteration. For details concerning the smoothing algorithm, see Chen (2007).

**Output 98.1.8** Iteration History for the Smoothing Algorithm

**The QUANTREG Procedure**  
**Quantile Level = 0.5**

Iteration History of Smoothing Algorithm					
Iter	Threshold	Refactor- ization	Full Update	Partial Update	Objective Function
1	227.24557	1	1000	0	4267.0988
15	116.94090	4	1480	2420	3631.9653
17	1.44064	4	1480	2583	2441.4719
20	0.72032	5	1980	2598	2441.3315
22	0.36016	6	2248	2607	2441.2369
24	0.18008	7	2376	2608	2441.2056
26	0.09004	8	2446	2613	2441.1997
28	0.04502	9	2481	2617	2441.1971
30	0.02251	10	2497	2618	2441.1956
32	0.01126	11	2505	2620	2441.1946
34	0.00563	12	2510	2621	2441.1933
35	0.00281	13	2514	2621	2441.1930
36	0.0000846	14	2517	2621	2441.1927
37	1E-12	14	2517	2621	2441.1927

[Output 98.1.9](#) displays the parameter estimates that are obtained by using the smoothing algorithm. These estimates are identical to those obtained by using the simplex and interior point algorithms. All three algorithms should have the same parameter estimates unless the problem does not have a unique solution.

**Output 98.1.9** Parameter Estimates from the Smoothing Algorithm

Parameter Estimates		
Parameter	DF	Estimate
Intercept	1	10.0364
x1	1	5.0106
x2	1	3.0294

The interior point algorithm and the smoothing algorithm offer better performance than the simplex algorithm for large data sets. For more information about choosing an appropriate algorithm on the basis of data set size, see Chen (2004). All three algorithms should have the same parameter estimates, unless the optimization problem has multiple solutions.

---

**Example 98.2: Quantile Regression for Econometric Growth Data**

This example uses a SAS data set named *Growth*, which contains economic growth rates for countries during two time periods: 1965–1975 and 1975–1985. The data come from a study by Barro and Lee (1994) and have also been analyzed by Koenker and Machado (1999).

There are 161 observations and 15 variables in the data set. The variables, which are listed in the following table, include the national growth rates (GDP) for the two periods, 13 covariates, and a name variable (Country) for identifying the countries in one of the two periods.

Variable	Description
Country	Country's name and period
GDP	Annual change per capita in gross domestic product (GDP)
lgdp2	Initial per capita GDP
mse2	Male secondary education
fse2	Female secondary education
fhe2	Female higher education
mhe2	Male higher education
lexp2	Life expectancy
lintr2	Human capital
gedy2	Education/GDP
ly2	Investment/GDP
gcony2	Public consumption/GDP
lblakp2	Black market premium
pol2	Political instability
ttrad2	Growth rate terms trade

The goal is to study the effect of the covariates on GDP. The following statements request median regression for a preliminary exploration. They produce the results that are in [Output 98.2.1](#) through [Output 98.2.6](#).

```
data growth;
  length Country$ 22;
  input Country GDP lgdp2 mse2 fse2 fhe2 mhe2 lexp2 lintr2 gedy2
        ly2 gcony2 lblakp2 pol2 ttrad2 @@;
```

```

    datalines;
Algeria75      .0415 7.330 .1320 .0670 .0050 .0220 3.880 .1138 .0382
               .1898 .0601 .3823 .0833 .1001
Algeria85      .0244 7.745 .2760 .0740 .0070 .0370 3.978 -.107 .0437
               .3057 .0850 .9386 .0000 .0657
Argentina75    .0187 8.220 .7850 .6200 .0740 .1660 4.181 .4060 .0221
               .1505 .0596 .1924 .3575 -.011
Argentina85    -.014 8.407 .9360 .9020 .1320 .2030 4.211 .1914 .0243
               .1467 .0314 .3085 .7010 -.052
Australia75    .0259 9.101 2.541 2.353 .0880 .2070 4.263 6.937 .0348
               .3272 .0257 .0000 .0080 -.016

    ... more lines ...

Zambia75      .0120 6.989 .3760 .1190 .0130 .0420 3.757 .4388 .0339
               .3688 .2513 .3945 .0000 -.032
Zambia85      -.046 7.109 .4200 .2740 .0110 .0270 3.854 .8812 .0477
               .1632 .2637 .6467 .0000 -.033
Zimbabwe75    .0320 6.860 .1450 .0170 .0080 .0450 3.833 .7156 .0337
               .2276 .0246 .1997 .0000 -.040
Zimbabwe85    -.011 7.180 .2200 .0650 .0060 .0400 3.944 .9296 .0520
               .1559 .0518 .7862 .7161 -.024
;

ods graphics on;

proc quantreg data=growth ci=resampling
               plots=(rdplot ddplot reshistogram);
  model GDP = lgdp2 mse2 fse2 fhe2 mhe2 lexp2
             lintr2 gedy2 Iy2 gcony2 lblakp2 pol2 ttrad2
             / quantile=.5 diagnostics leverage(cutoff=8) seed=1268;
  id Country;
  test_lgdp2: test lgdp2 / lr wald;
run;

```

The QUANTREG procedure uses the default simplex algorithm to estimate the parameters and uses the MCMB resampling method to compute confidence limits.

Output 98.2.1 displays model information and summary statistics for the variables in the model. Six summary statistics are computed, including the median and the median absolute deviation (MAD), which are robust measures of univariate location and scale, respectively. For the variable `lintr2` (human capital), both the mean and standard deviation are much larger than the corresponding robust measures (median and MAD), indicating that this variable might have outliers.

### Output 98.2.1 Model Information and Summary Statistics

#### The QUANTREG Procedure

Model Information						
Data Set	WORK.GROWTH					
Dependent Variable	GDP					
Number of Independent Variables	13					
Number of Observations	161					
Optimization Algorithm	Simplex					
Method for Confidence Limits	Resampling					

Summary Statistics						
Variable	Q1	Median	Q3	Mean	Standard Deviation	MAD
<code>lgdp2</code>	6.9890	7.7450	8.6080	7.7905	0.9543	1.1579
<code>mse2</code>	0.3160	0.7230	1.2675	0.9666	0.8574	0.6835
<code>fse2</code>	0.1270	0.4230	0.9835	0.7117	0.8331	0.5011
<code>fhe2</code>	0.0110	0.0350	0.0890	0.0792	0.1216	0.0400
<code>mhe2</code>	0.0400	0.1060	0.2060	0.1584	0.1752	0.1127
<code>lexp2</code>	3.8670	4.0640	4.2430	4.0440	0.2028	0.2728
<code>lintr2</code>	0.00160	0.5604	1.8805	1.4625	2.5491	1.0058
<code>gedy2</code>	0.0248	0.0343	0.0466	0.0360	0.0141	0.0151
<code>ly2</code>	0.1396	0.1955	0.2671	0.2010	0.0877	0.0981
<code>gcony2</code>	0.0480	0.0767	0.1276	0.0914	0.0617	0.0566
<code>lblakp2</code>	0	0.0696	0.2407	0.1916	0.3070	0.1032
<code>pol2</code>	0	0.0500	0.2429	0.1683	0.2409	0.0741
<code>ttrad2</code>	-0.0240	-0.0100	0.00730	-0.00570	0.0375	0.0239
<code>GDP</code>	0.00290	0.0196	0.0351	0.0191	0.0248	0.0237

Output 98.2.2 displays the parameter estimates and 95% confidence limits that are computed with the rank method.

**Output 98.2.2** Parameter Estimates

Parameter Estimates							
Parameter	DF	Estimate	Standard Error	95% Confidence Limits		t Value	Pr >  t
Intercept	1	-0.0488	0.0719	-0.1910	0.0934	-0.68	0.4985
lgdp2	1	-0.0269	0.0039	-0.0346	-0.0192	-6.93	<.0001
mse2	1	0.0110	0.0080	-0.0048	0.0268	1.38	0.1695
fse2	1	-0.0011	0.0088	-0.0186	0.0163	-0.13	0.8970
fhe2	1	0.0148	0.0319	-0.0481	0.0778	0.47	0.6421
mhe2	1	0.0043	0.0265	-0.0481	0.0567	0.16	0.8720
lexp2	1	0.0683	0.0220	0.0249	0.1118	3.11	0.0022
lintr2	1	-0.0022	0.0015	-0.0052	0.0008	-1.45	0.1483
gedy2	1	-0.0508	0.1632	-0.3734	0.2717	-0.31	0.7558
ly2	1	0.0723	0.0240	0.0249	0.1198	3.01	0.0031
gcony2	1	-0.0935	0.0369	-0.1664	-0.0207	-2.54	0.0122
lblakp2	1	-0.0269	0.0085	-0.0438	-0.0101	-3.16	0.0019
pol2	1	-0.0301	0.0094	-0.0486	-0.0115	-3.20	0.0017
ttrad2	1	0.1613	0.0750	0.0130	0.3095	2.15	0.0332

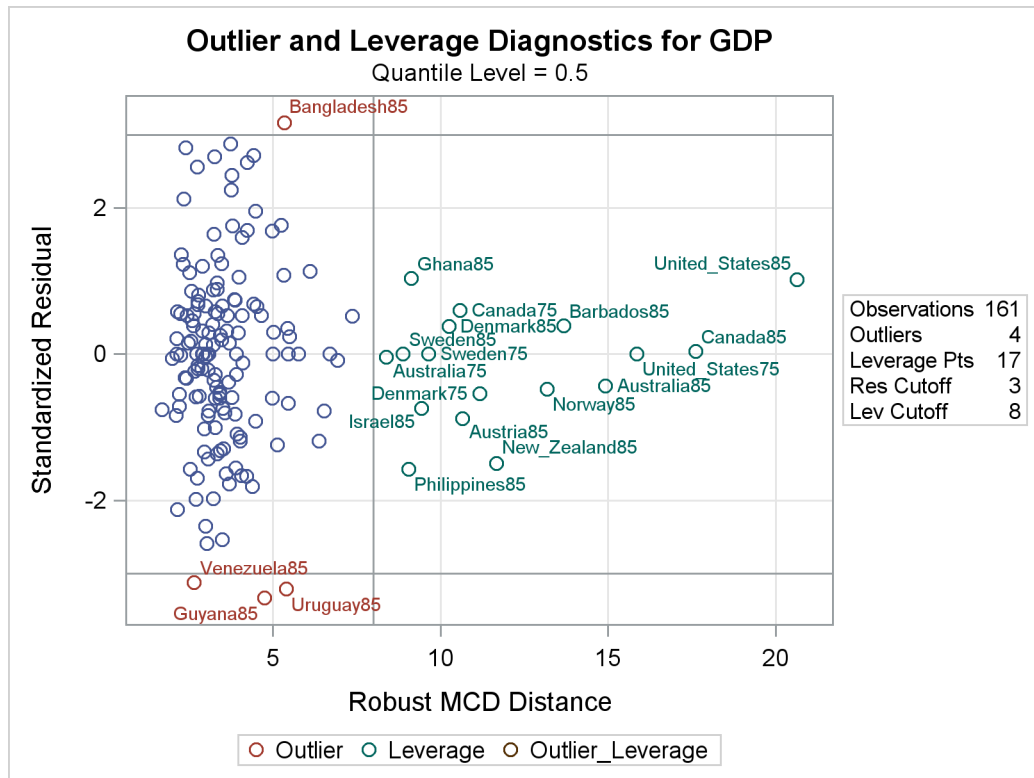
Diagnostics for the median regression fit, which are requested in the PLOTS= option, are displayed in Output 98.2.3 and Output 98.2.4. Output 98.2.3 plots the standardized residuals from median regression against the robust MCD distance. This display is used to diagnose both vertical outliers and horizontal leverage points. Output 98.2.4 plots the robust MCD distance against the Mahalanobis distance. This display is used to diagnose leverage points.

The cutoff value 8, which is specified in the LEVERAGE option, is close to the maximum of the Mahalanobis distance. Eighteen points are diagnosed as high leverage points, and almost all are countries with high human capital, which is the major contributor to the high leverage as observed from the summary statistics. Four points are diagnosed as outliers by using the default cutoff value of 3. However, these are not extreme outliers.

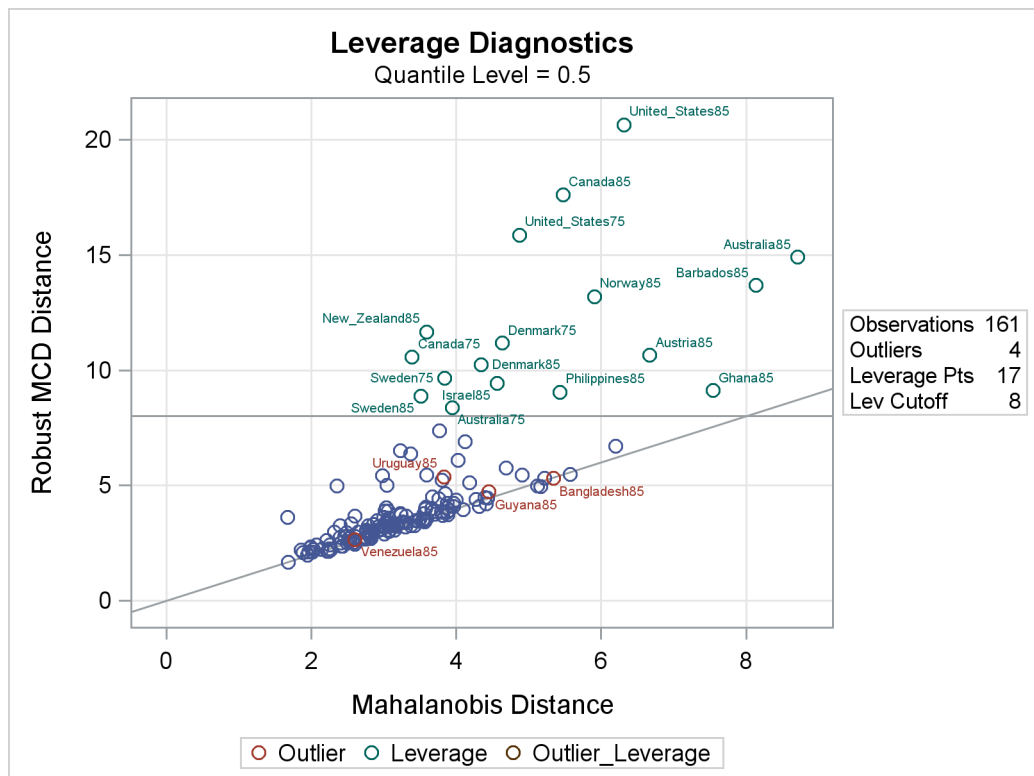
A histogram of the standardized residuals and two fitted density curves are displayed in Output 98.2.5. This output shows that median regression fits the data well.

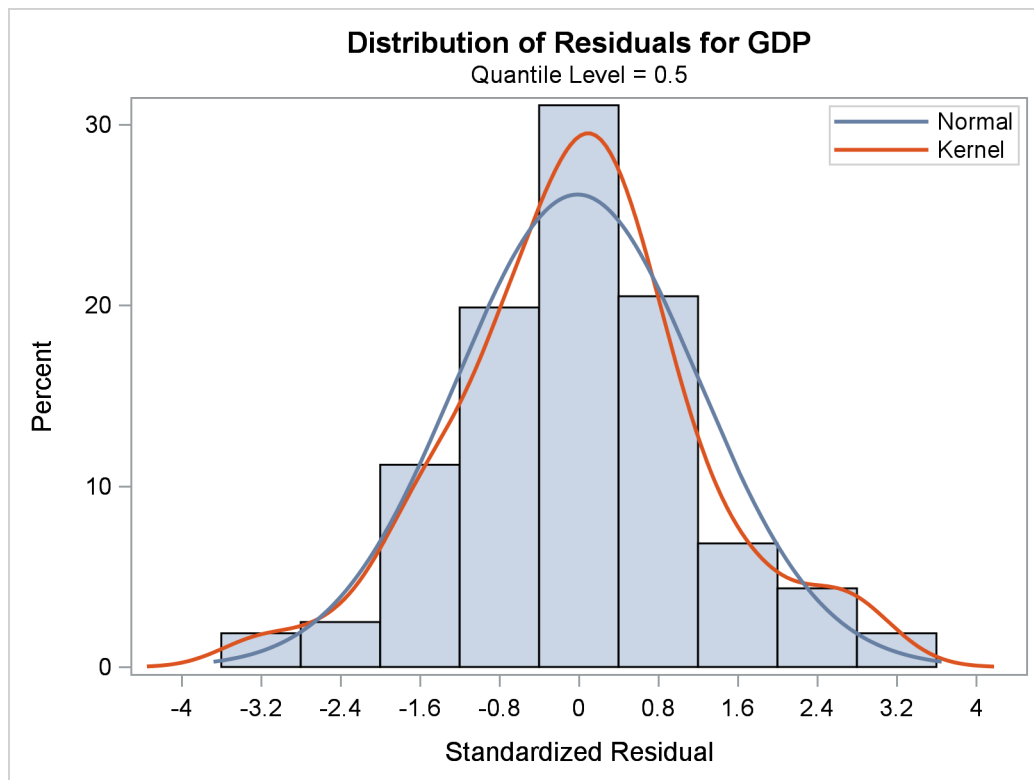


**Output 98.2.3** Plot of Residual versus Robust Distance



**Output 98.2.4** Plot of Robust Distance versus Mahalanobis Distance



**Output 98.2.5** Histogram for Residuals

Tests of significance for the initial per-capita GDP (LGDP2) are shown in [Output 98.2.6](#).

**Output 98.2.6** Tests for Regression Coefficient

Test test_lgdp2 Results				
Test	Test			
	Statistic	DF	Chi-Square	Pr > ChiSq
Wald	48.0009	1	48.00	<.0001
Likelihood Ratio	36.3047	1	36.30	<.0001

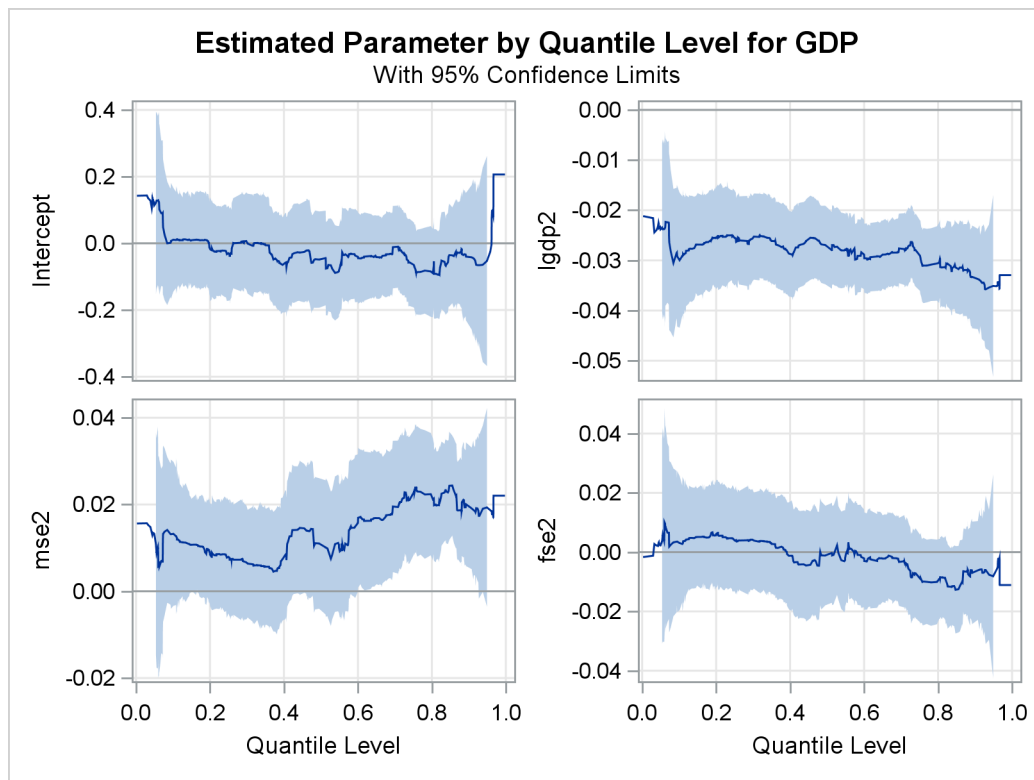
The QUANTREG procedure computes entire quantile processes for covariates when you specify QUANTILE=PROCESS in the MODEL statement, as follows:

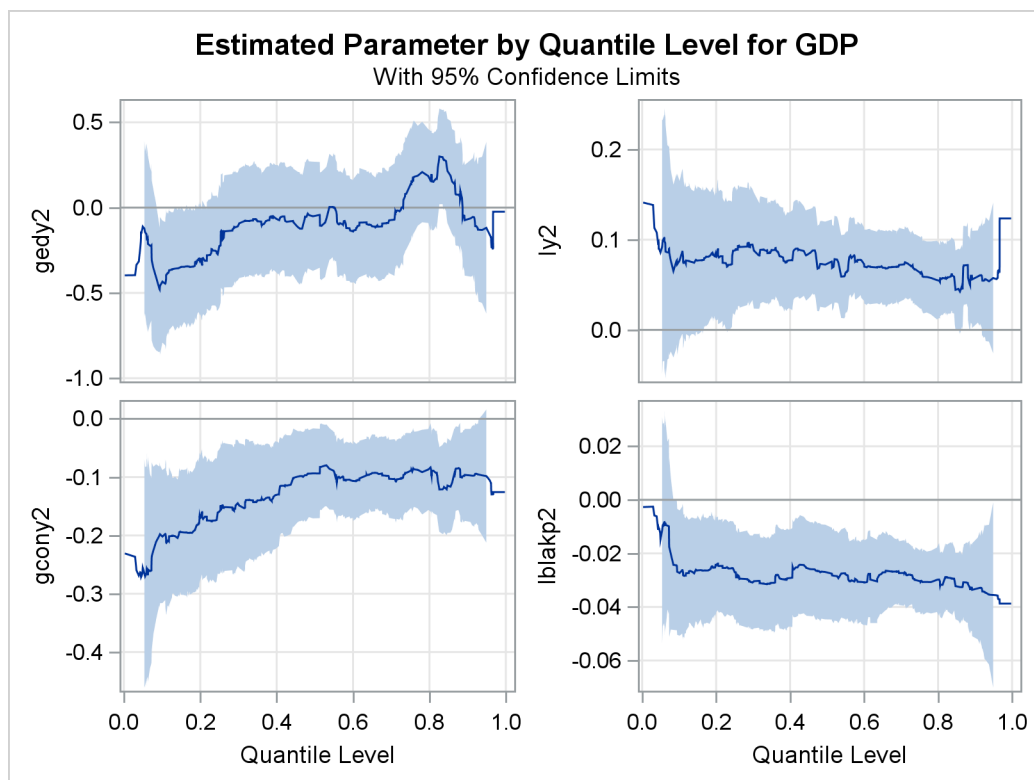
```
proc quantreg data=growth ci=resampling;
  model GDP = lgdp2 mse2 fse2 fhe2 mhe2 lexp2 lintr2
           gedy2 ly2 gcony2 lblakp2 pol2 ttrad2
           / quantile=process plot=quantplot seed=1268;
run;
```

Confidence limits for quantile processes can be computed by using the sparsity or resampling methods. But they cannot be computed by using the rank method, because the computation would be prohibitively expensive.

A total of 14 quantile process plots are produced. [Output 98.2.7](#) and [Output 98.2.8](#) display two panels of eight selected process plots. The 95% confidence bands are shaded.

**Output 98.2.7** Quantile Processes with 95% Confidence Bands



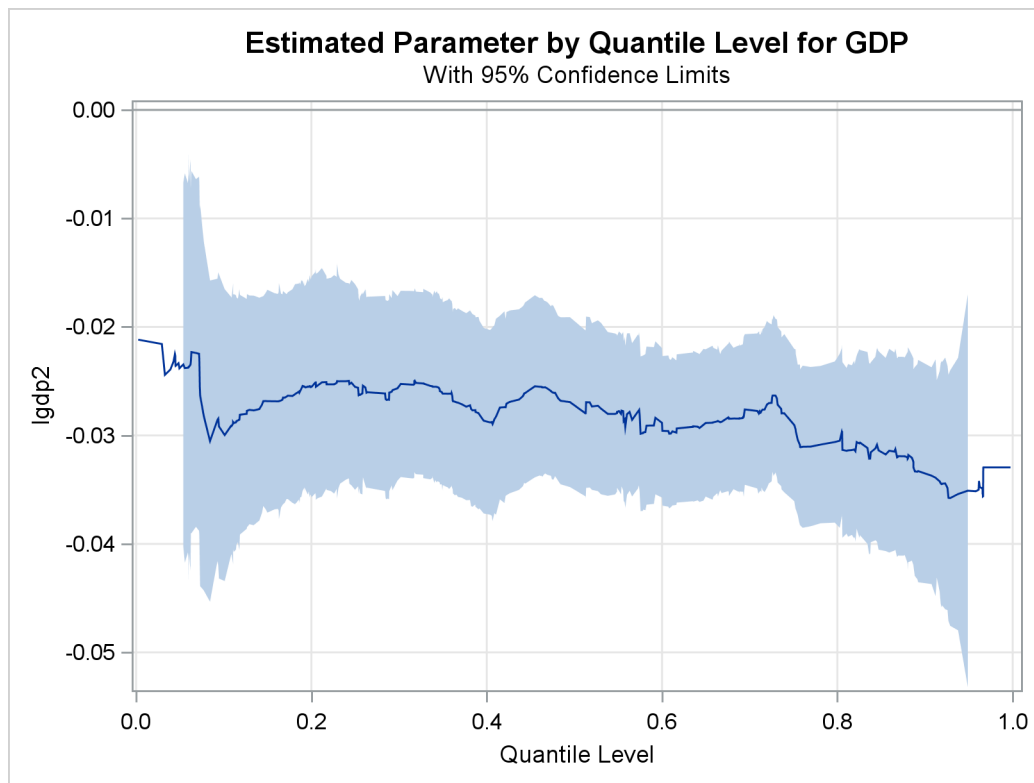
**Output 98.2.8** Quantile Processes with 95% Confidence Bands

As pointed out by Koenker and Machado (1999), previous studies of the Barro growth data have focused on the effect of the initial per-capita GDP on the growth of this variable (annual change in per-capita GDP). The following statements request a single process plot for this effect:

```
proc quantreg data=growth ci=resampling;
  model GDP = lgdp2 mse2 fse2 fhe2 mhe2 lexp2 lintr2
    gedy2 ly2 gcony2 lblakp2 pol2 ttrad2
    / quantile=process plot=quantplot(lgdp2) seed=1268;
run;
```

The plot is shown in [Output 98.2.9](#).

**Output 98.2.9** Quantile Process Plot for LGDP2



The confidence bands here are computed by using the MCMB resampling method. In contrast, Koenker and Machado (1999) used the rank method to compute confidence limits for a few selected points. [Output 98.2.9](#) suggests that the effect of the initial level of GDP is relatively constant over the entire distribution, with a slightly stronger effect in the upper tail.

The effects of other covariates are quite varied. An interesting covariate is public consumption divided by GDP (gcony2) (first plot in second panel), which has a constant effect over the upper half of the distribution and a larger effect in the lower tail. For an analysis of the effects of the other covariates, see Koenker and Machado (1999).

---

### Example 98.3: Quantile Regression Analysis of Birth-Weight Data

This example is patterned after a quantile regression analysis of covariates associated with birth weight that was carried out by Koenker and Hallock (2001). Their study uses a subset of the June 1997 Detailed Natality Data, which was published by the National Center for Health Statistics. The study demonstrates that conditional quantile functions provide more complete information about the covariate effects than ordinary least squares regression provides.

This example is based on Koenker and Hallock (2001); Abreveya (2001); it uses data for live, singleton births to mothers in the United States who were recorded as black or white, and who were between the ages of 18 and 45. For convenience, this example uses 50,000 observations, which are randomly selected from the

qualified observations. Observations that have missing data for any of the variables are deleted. The data are available in the data set Sashelp.BWeight. The following step displays in [Output 98.3.1](#) the variables in the data set:

```
proc contents varnum data=sashelp.bweight;
  ods select position;
run;
```

### Output 98.3.1 Sashelp.BWeight Data Set

#### The CONTENTS Procedure

Variables in Creation Order			
#	Variable	Type	Len Label
1	Weight	Num	8 Infant Birth Weight
2	Black	Num	8 Black Mother
3	Married	Num	8 Married Mother
4	Boy	Num	8 Baby Boy
5	MomAge	Num	8 Mother's Age
6	MomSmoke	Num	8 Smoking Mother
7	CigsPerDay	Num	8 Cigarettes Per Day
8	MomWtGain	Num	8 Mother's Pregnancy Weight Gain
9	Visit	Num	8 Prenatal Visit
10	MomEdLevel	Num	8 Mother's Education Level

The following step creates descriptive labels for the values of the classification variables Visit and MomEdLevel:

```
proc format;
  value vfmt 0 = 'No Visit'          1 = 'Second Trimester'
            2 = 'Last Trimester' 3 = 'First Trimester';
  value efmt 0 = 'High School'      1 = 'Some College'
            2 = 'College'          3 = 'Less Than High School';
run;
```

There are four levels of maternal education. When you specify the ORDER=INTERNAL option, PROC QUANTREG treats the highest unformatted value (3, which represents that the mother's education level is less than high school) as a reference level. The regression coefficients of other levels measure the effect relative to this level. Likewise, there are four levels of prenatal medical care of the mother, and a first visit in the first trimester serves as the reference level.

The following statements fit a regression model for 19 quantiles of birth weight, which are evenly spaced in the interval (0, 1). The model includes linear and quadratic effects for the age of the mother and for weight gain during pregnancy.

```
ods graphics on;

proc quantreg ci=sparsity/iid algorithm=interior(tolerance=5.e-4)
  data=sashelp.bweight order=internal;
  class Visit MomEdLevel;
  model Weight = Black Married Boy Visit MomEdLevel MomSmoke
    CigsPerDay MomAge MomAge*MomAge
    MomWtGain MomWtGain*MomWtGain /
```

```

quantile= 0.05 to 0.95 by 0.05
plot=quantplot;
format Visit vfmt. MomEdLevel efmt.;
run;

```

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

### Output 98.3.2 Model Information and Summary Statistics

#### The QUANTREG Procedure

Model Information						
Data Set	SASHELP.BWEIGHT Infant Birth Weight					
Dependent Variable	Weight Infant Birth Weight					
Number of Independent Variables	9					
Number of Continuous Independent Variables	7					
Number of Class Independent Variables	2					
Number of Observations	50000					
Optimization Algorithm	Interior					
Method for Confidence Limits	Sparsity					

Summary Statistics						
Variable	Q1	Median	Q3	Mean	Standard Deviation	MAD
Black	0	0	0	0.1628	0.3692	0
Married	0	1.0000	1.0000	0.7126	0.4525	0
Boy	0	1.0000	1.0000	0.5158	0.4998	0
MomSmoke	0	0	0	0.1307	0.3370	0
CigsPerDay	0	0	0	1.4766	4.6541	0
MomAge	-4.0000	0	5.0000	0.4161	5.7285	5.9304
MomAge*MomAge	4.0000	16.0000	49.0000	32.9877	39.2861	22.2390
MomWtGain	-8.0000	0	9.0000	0.7092	12.8761	11.8608
MomWtGain*MomWtGain	16.0000	64.0000	196.0	166.3	298.8	88.9561
Weight	3062.0	3402.0	3720.0	3370.8	566.4	504.1

Among the 11 independent variables, Black, Married, Boy, and MomSmoke are binary variables. For these variables, the mean represents the proportion in the category. The two continuous variables, MomAge and MomWtGain, are centered at their medians, which are 27 and 30, respectively.

The quantile plots for the intercept and the other 15 factors with nonzero degrees of freedom are shown in the following four panels. In each plot, the regression coefficient at a given quantile indicates the effect on birth weight of a unit change in that factor, assuming that the other factors are fixed. The bands represent 95% confidence intervals.

Although the data set used here is a subset of the Natality data set, the results are quite similar to those of Koenker and Hallock (2001) for the full data set.

In [Output 98.3.3](#), the first plot is for the intercept. As explained by Koenker and Hallock (2001), the intercept “may be interpreted as the estimated conditional quantile function of the birth-weight distribution of a girl born to an unmarried, white mother with less than a high school education, who is 27 years old and had a weight gain of 30 pounds, didn’t smoke, and had her first prenatal visit in the first trimester of the pregnancy.”

The second plot shows that infants born to black mothers weigh less than infants born to white mothers, especially in the lower tail of the birth-weight distribution. The third plot shows that marital status has a large positive effect on birth weight, especially in the lower tail. The fourth plot shows that boys weigh more than girls for any chosen quantile; this difference is smaller in the lower quantiles of the distribution.

In [Output 98.3.4](#), the first three plots deal with prenatal care. Compared with babies born to mothers who had a prenatal visit in the first trimester, babies born to mothers who received no prenatal care weigh less, especially in the lower quantiles of the birth-weight distributions. As noted by Koenker and Hallock (2001), “babies born to mothers who delayed prenatal visits until the second or third trimester have substantially *higher* birthweights in the lower tail than mothers who had a prenatal visit in the first trimester. This might be interpreted as the self-selection effect of mothers confident about favorable outcomes.”

The fourth plot in [Output 98.3.4](#) and the first two plots in [Output 98.3.5](#) are for variables that are related to education. Education beyond high school is associated with a positive effect on birth weight. The effect of high school education is uniformly around 15 grams across the entire birth-weight distribution (this is a pure location shift effect), whereas the effect of some college and college education is more positive in the lower quantiles than the upper quantiles.

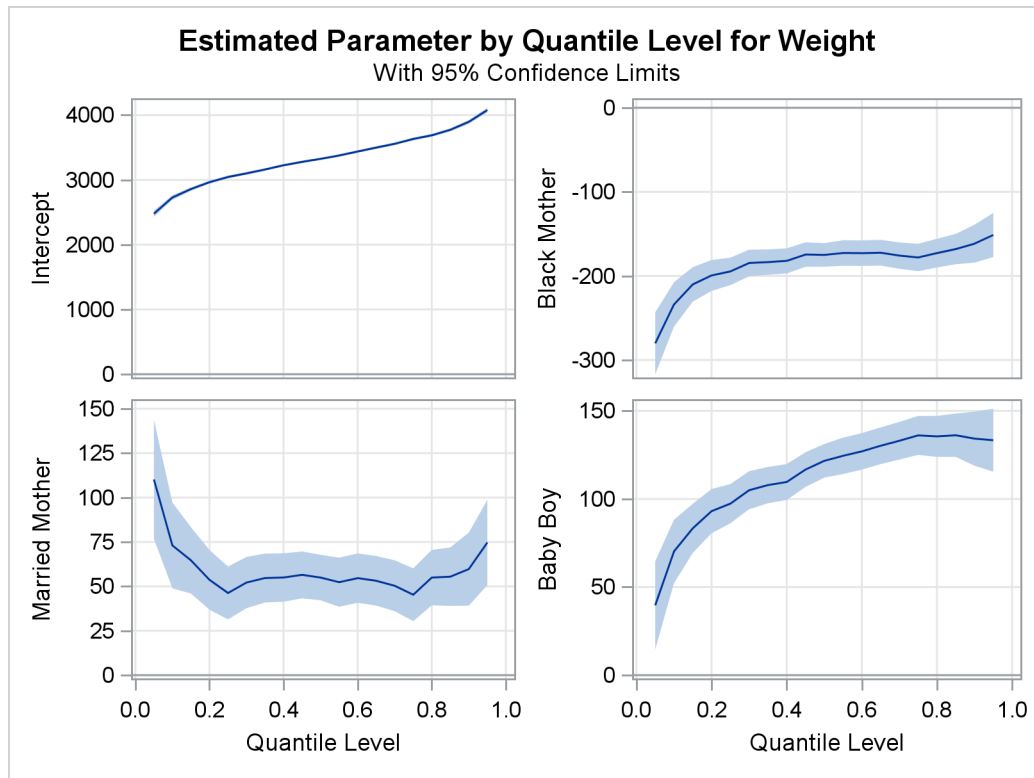
The remaining two plots in [Output 98.3.5](#) show that smoking is associated with a large negative effect on birth weight.

The linear and quadratic effects for the two continuous variables are shown in [Output 98.3.6](#). Both of these variables are centered at their median. At the lower quantiles, the quadratic effect of the mother’s age is more concave. The optimal age at the first quantile is about 33, and the optimal age at the third quantile is about 38. The effect of the mother’s weight gain is clearly positive, as indicated by the narrow confidence bands for both linear and quadratic coefficients.

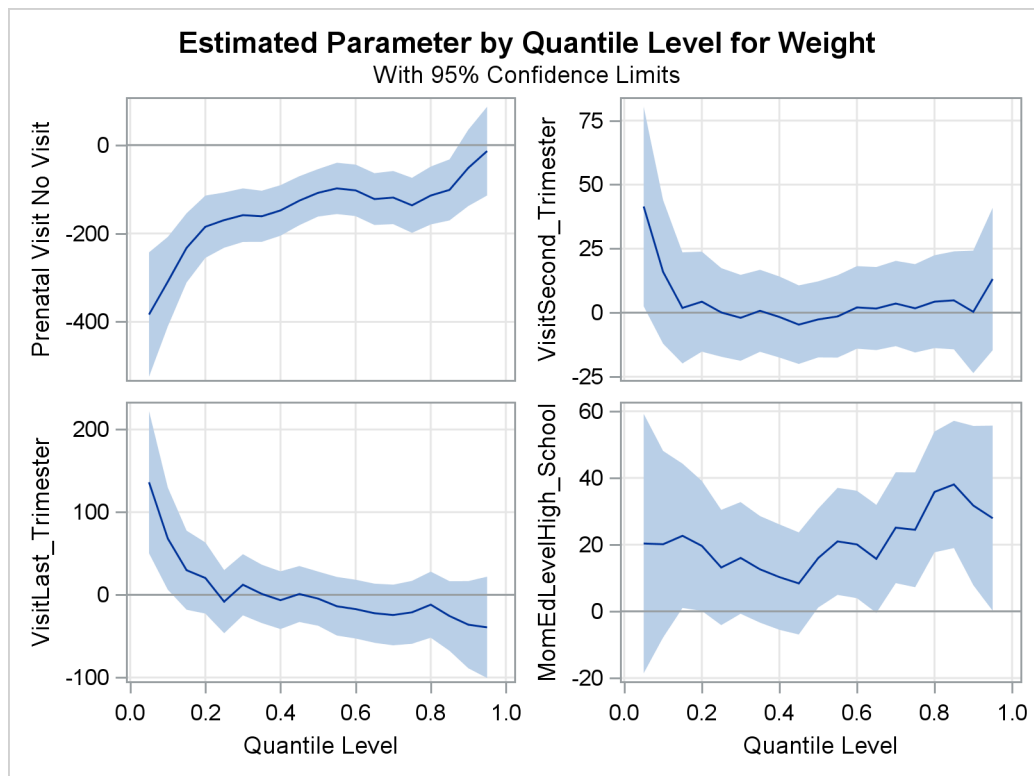
For more information about the covariate effects that are discovered by using quantile regression, see Koenker and Hallock (2001).

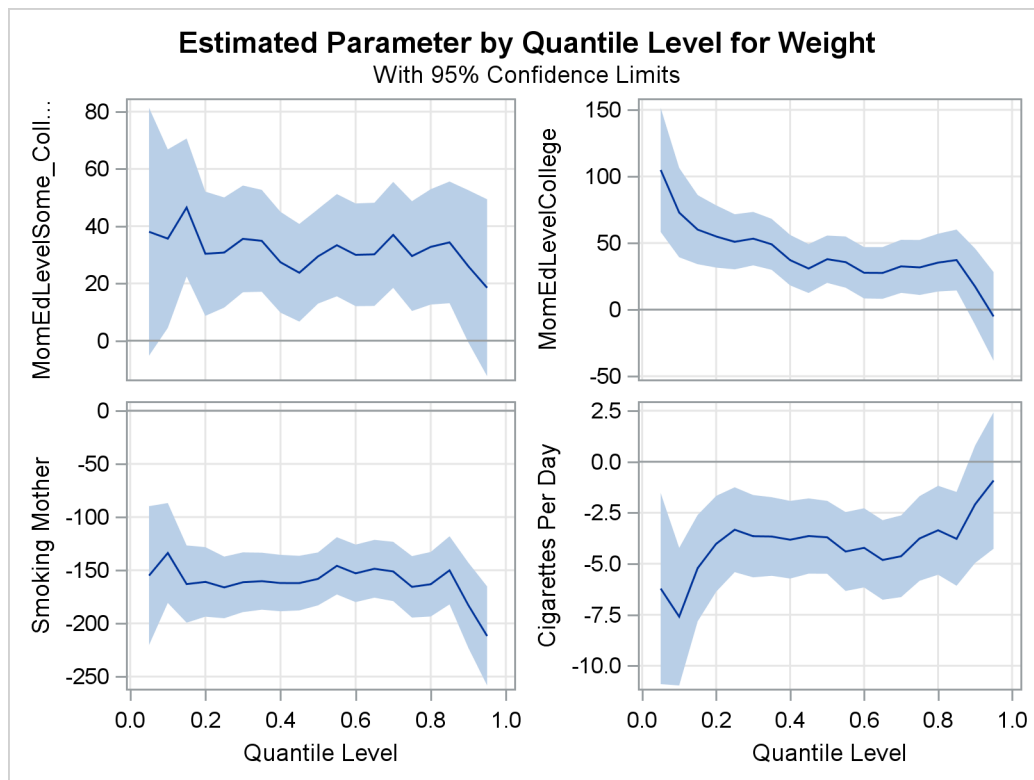
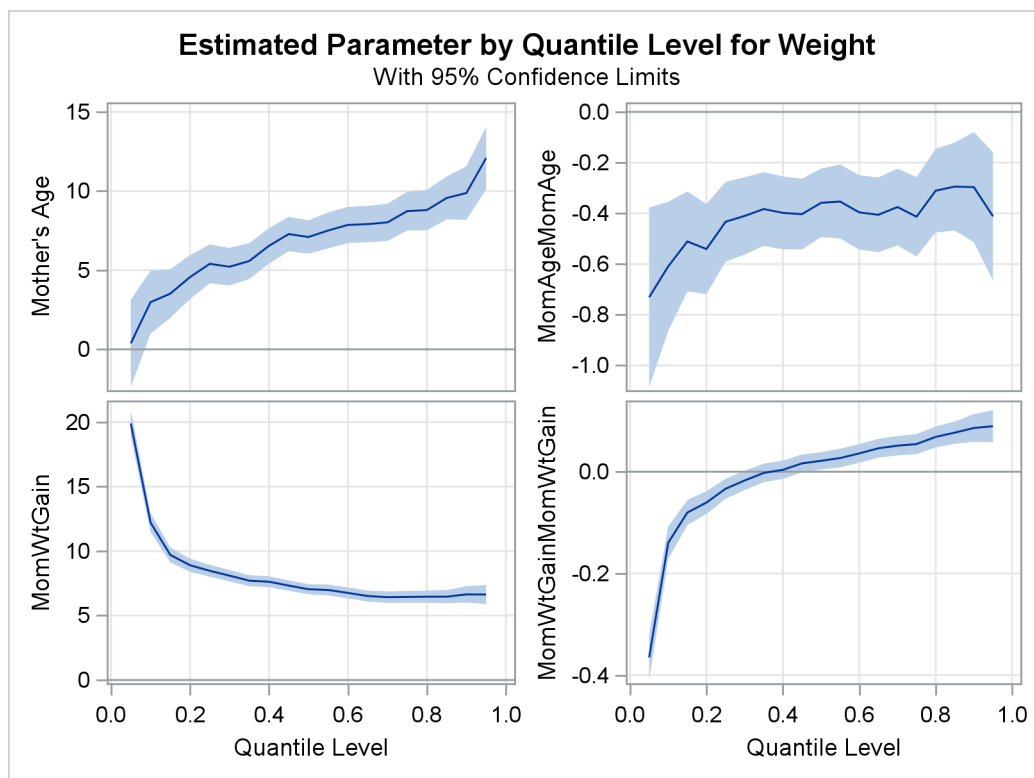


**Output 98.3.3** Quantile Processes with 95% Confidence Bands



**Output 98.3.4** Quantile Processes with 95% Confidence Bands



**Output 98.3.5** Quantile Processes with 95% Confidence Bands**Output 98.3.6** Quantile Processes with 95% Confidence Bands

## Example 98.4: Nonparametric Quantile Regression for Ozone Levels

Tracing seasonal trends in the level of tropospheric ozone is essential for predicting high-level periods, observing long-term trends, and discovering potential changes in pollution. Traditional methods for modeling seasonal effects are based on the conditional mean of ozone concentration. However, the upper conditional quantiles are more critical from a public-health perspective. In this example, the QUANTREG procedure fits conditional quantile curves for seasonal effects by using nonparametric quantile regression with cubic B-splines.

The data used here are from Chock, Winkler, and Chen (2000), who studied the association between daily mortality and ambient air pollutant concentrations in Pittsburgh, Pennsylvania. The data set ozone contains the following two variables: Ozone, which represents the daily maximum one-hour ozone concentration (ppm) and Days, which is an index of 1,095 days (3 years).

```
data ozone;
  days = _n_;
  input ozone @@;
  datalines;
0.0060 0.0060 0.0320 0.0320 0.0320 0.0150 0.0150 0.0150 0.0200 0.0200
0.0160 0.0070 0.0270 0.0160 0.0150 0.0240 0.0220 0.0220 0.0220 0.0185
0.0150 0.0150 0.0110 0.0070 0.0070 0.0240 0.0380 0.0240 0.0265 0.0290
0.0310 0.0460 0.0360 0.0260 0.0300 0.0250 0.0280 0.0310 0.0370 0.0325

... more lines ...

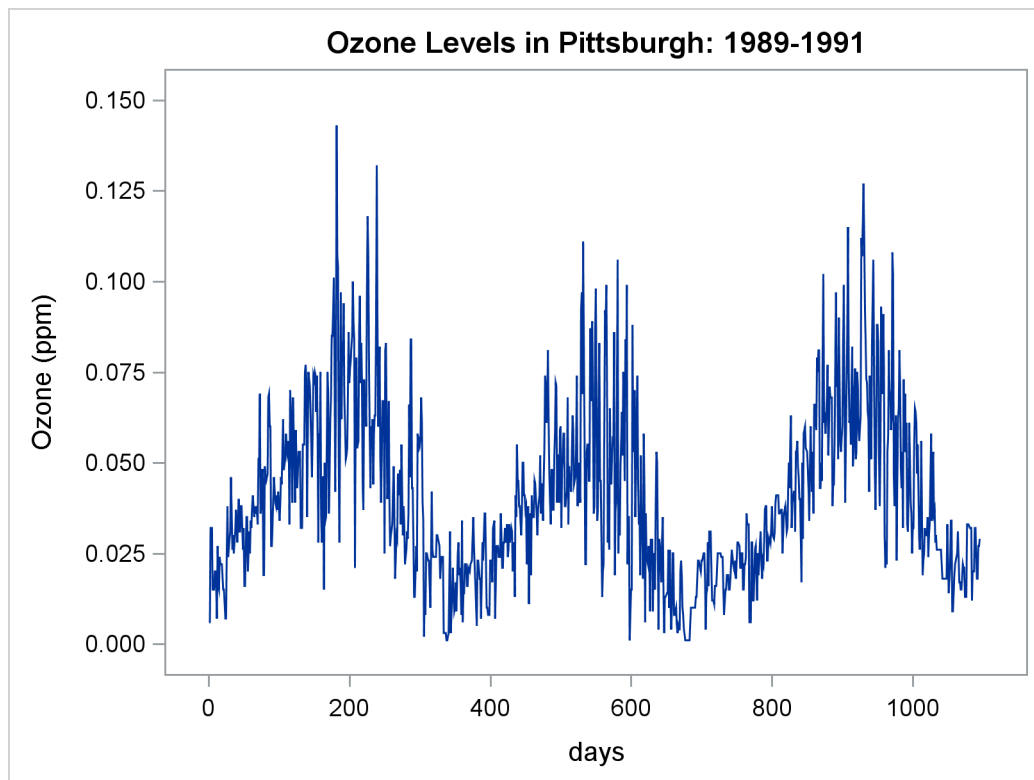
0.0220 0.0210 0.0210 0.0130 0.0130 0.0130 0.0330 0.0330 0.0330 0.0325
0.0320 0.0320 0.0320 0.0120 0.0200 0.0200 0.0200 0.0320 0.0320 0.0250
0.0180 0.0180 0.0270 0.0270 0.0290
;
```

[Output 98.4.1](#), which displays the time series plot of ozone concentration for the three years, shows a clear seasonal pattern.

In this example, cubic B-splines are used to fit the seasonal effect. These splines are generated with 11 knots, which split the 3 years into 12 seasons. The following statements create the spline basis and fit multiple quantile regression spline curves:

```
ods graphics on;

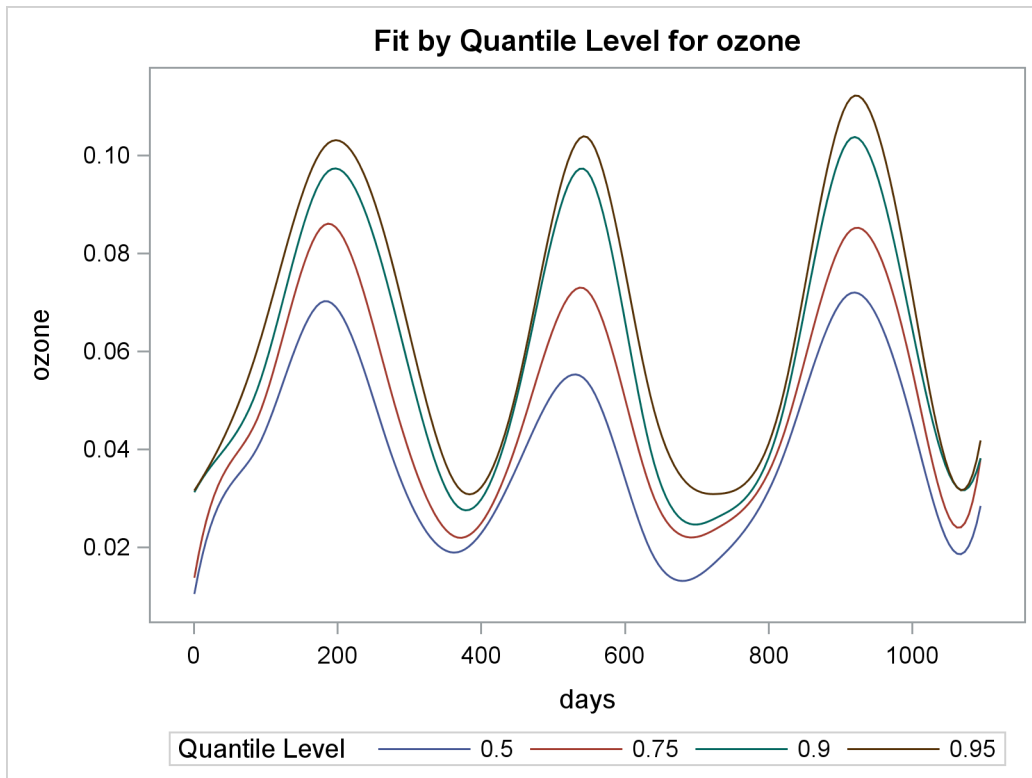
proc quantreg data=ozone algorithm=smooth ci=none plot=fitplot(nodata);
  effect sp = spline( days / knotmethod = list
    (90 182 272 365 455 547 637 730 820 912 1002) );
  model ozone = sp / quantile = 0.5 0.75 0.90 0.95 seed=1268;
run;
```

**Output 98.4.1** Time Series of Ozone Levels in Pittsburgh, Pennsylvania

The **EFFECT** statement creates spline bases for the variable **Days**. The **KNOTMETHOD=LIST** option provides all internal knots for these bases. Cubic spline bases are generated by default. These bases are treated as components of the spline effect *sp*, which is specified in the **MODEL** statement. Spline fits for four quantiles are requested in the **QUANTILE=** option.

When ODS Graphics is enabled, the QUANTREG procedure automatically generates a fit plot, which includes all fitted curves.

**Output 98.4.2** displays these curves. The curves show that peak ozone levels occur in the summer. For the three years 1989–1991, the median curve (labeled 50%) does not cross the 0.08 ppm line, which is the 1997 EPA eight-hour standard. The median curve and the 75% curve show a drop for the ozone concentration levels in 1990. However, for the 90% and 95% curves, peak ozone levels tend to increase. This indicates that there might have been more days with low ozone concentration in 1990, but the top 10% and 5% tend to have higher ozone concentration levels.

**Output 98.4.2** Quantiles of Ozone Levels in Pittsburgh, Pennsylvania

The quantile curves also show that high ozone concentration in 1989 had a longer duration than in 1990 and 1991. This is indicated by the wider spread of the quantile curves in 1989.

---

**Example 98.5: Quantile Polynomial Regression for Salary Data**

This example uses the data set from a university union survey of salaries of professors in 1991. The survey covered departments in US colleges and universities that list programs in statistics. The goal of this example is to examine the relationship between faculty salaries and years of service.

The data include salaries and years of service for 459 professors. The scatter plot in [Output 98.5.1](#) shows that the relationship is not linear and that a quadratic or cubic regression curve is appropriate. [Output 98.5.1](#) shows a cubic curve.

The curve in [Output 98.5.1](#) does not adequately describe the conditional salary distributions and how they change with length of service. [Output 98.5.2](#) shows the 25th, 50th, and 75th percentiles for each number of years, which gives a better picture of the conditional distributions.

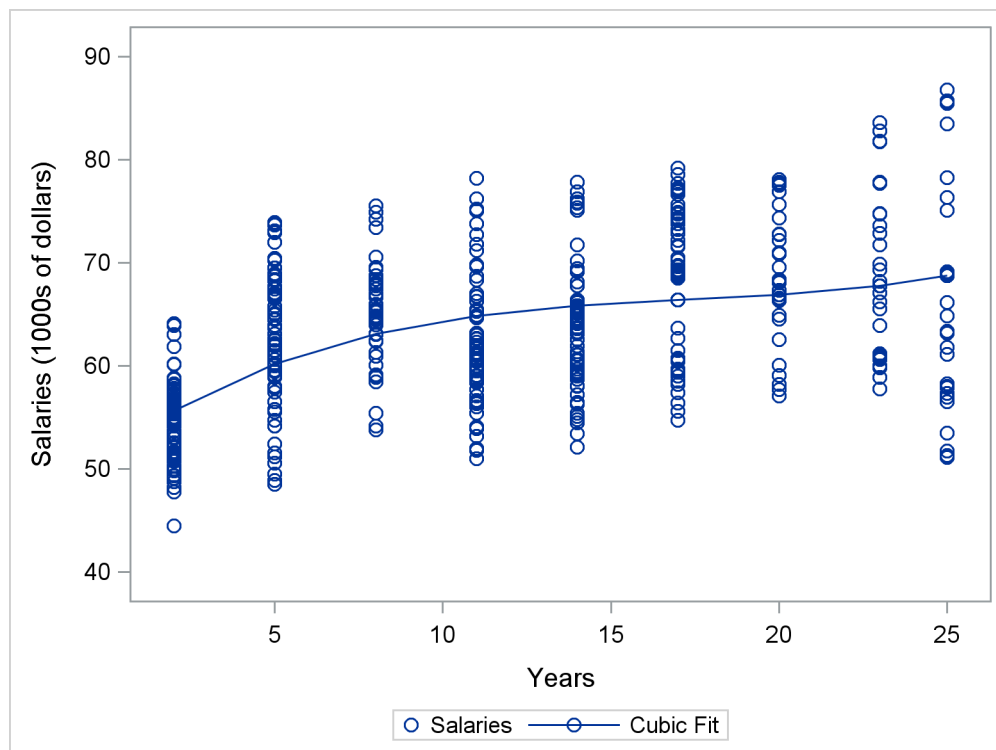
```

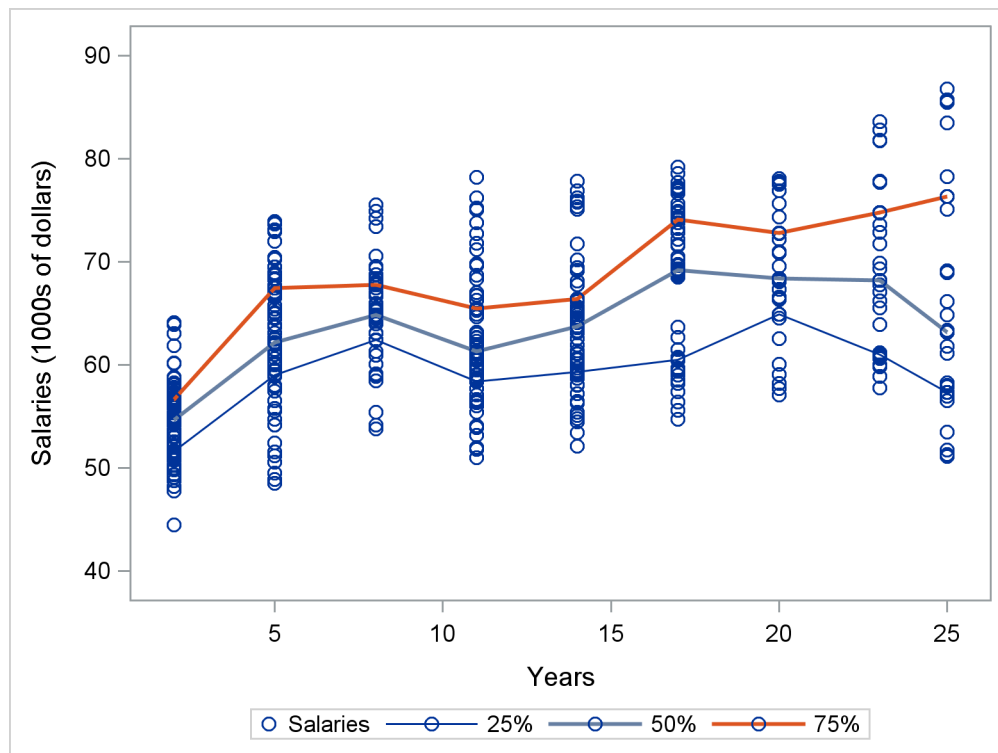
data salary;
  input Salaries Years @@;
  label Salaries='Salaries (1000s of dollars)';
  datalines;
54.94 2 58.24 2 58.11 2 52.23 2 52.98 2 57.62 2
44.48 2 57.22 2 54.24 2 54.79 2 56.42 2 61.90 2
63.90 2 64.10 2 47.77 2 54.86 2 49.31 2 53.37 2
51.69 2 53.66 2 58.77 2 56.77 2 53.06 2 54.86 2
50.96 2 56.46 2 51.67 2 49.37 2 56.86 2 49.85 2

... more lines ...

85.72 25 64.87 25 51.76 25 51.11 25 51.31 25 78.28 25
57.91 25 86.78 25 58.27 25 56.56 25 76.33 25 61.83 25
69.13 25 63.15 25 66.13 25
;

```

**Output 98.5.1** Salary and Years as Professor: Cubic Fit

**Output 98.5.2** Salary and Years as Professor: Sample Quantiles

These descriptive percentiles do not clearly show trends with length of service. The following statements use polynomial quantile regression to obtain a smooth version.

```
ods graphics on;

proc quantreg data=salary ci=sparsity;
  model salaries = years years*years years*years*years
    /quantile=0.25 0.5 0.75
    plot=fitplot(showlimits);

  test years/QINTERACT;

run;
```

The results are shown in [Output 98.5.3](#) and [Output 98.5.5](#). [Output 98.5.3](#) displays the regression coefficients for the three quantiles, from which you can see a difference among the estimated parameters of the variable *years* across the three quantiles. To test whether the difference is significant, you can specify the option `QINTERACT` in the `TEST` statement. [Output 98.5.4](#) indicates that the difference is not significant (the *p*-value is greater than 0.05).

**Output 98.5.3** Regression Coefficients**The QUANTREG Procedure**  
**Quantile Level = 0.25**

Parameter Estimates							
Parameter	DF	Estimate	Standard Error	95% Confidence Limits		t Value	Pr >  t
Intercept	1	48.2509	1.3484	45.6011	50.9007	35.78	<.0001
Years	1	2.2234	0.5455	1.1514	3.2953	4.08	<.0001
Years*Years	1	-0.1292	0.0500	-0.2275	-0.0308	-2.58	0.0101
Years*Years*Years	1	0.0024	0.0013	-0.0001	0.0049	1.86	0.0634

Parameter Estimates							
Parameter	DF	Estimate	Standard Error	95% Confidence Limits		t Value	Pr >  t
Intercept	1	50.2512	1.2812	47.7334	52.7690	39.22	<.0001
Years	1	2.7173	0.5947	1.5485	3.8860	4.57	<.0001
Years*Years	1	-0.1632	0.0632	-0.2873	-0.0390	-2.58	0.0101
Years*Years*Years	1	0.0034	0.0018	-0.0002	0.0070	1.85	0.0647

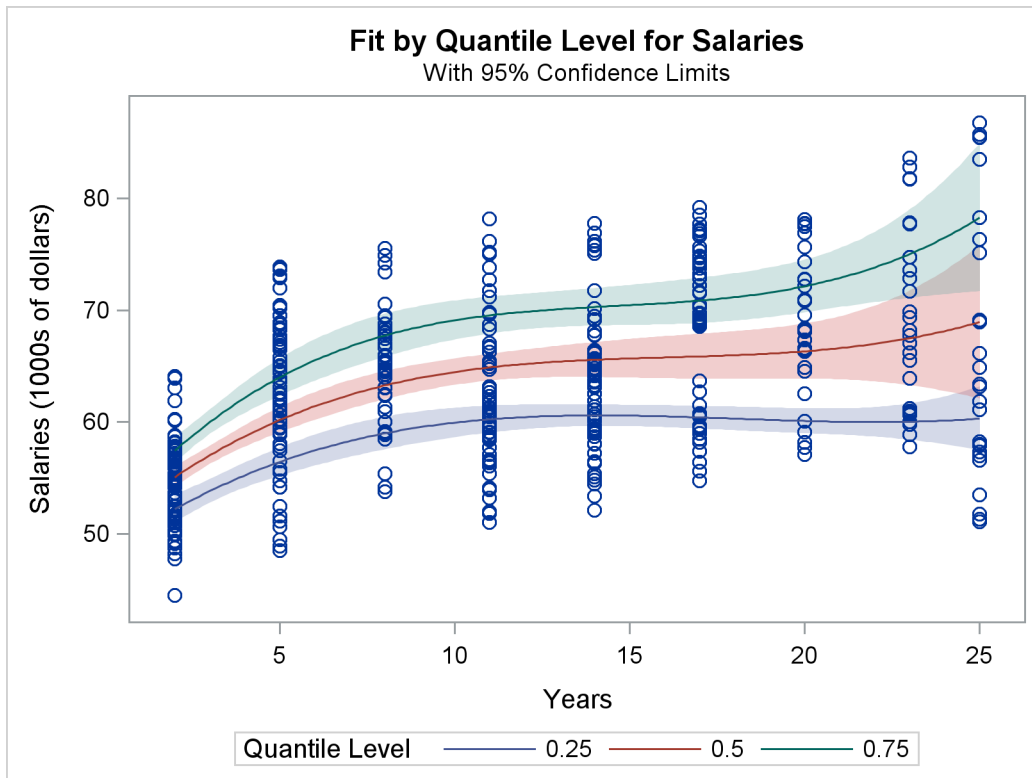
Parameter Estimates							
Parameter	DF	Estimate	Standard Error	95% Confidence Limits		t Value	Pr >  t
Intercept	1	51.0298	1.5886	47.9078	54.1517	32.12	<.0001
Years	1	3.6513	0.7594	2.1590	5.1436	4.81	<.0001
Years*Years	1	-0.2390	0.0764	-0.3892	-0.0888	-3.13	0.0019
Years*Years*Years	1	0.0055	0.0021	0.0013	0.0096	2.60	0.0098

**Output 98.5.4** Tests for Heteroscedasticity

Test Results Equal Coefficients Across Quantiles			
Chi-Square			
DF	Pr > ChiSq		
2	0.1825		

The three fitted quantile curves and their 95% confidence limits in the [Output 98.5.5](#) clearly show that salary dispersion increases gradually with length of service. After 15 years, a salary more than \$70,000 is relatively high, whereas a salary less than \$60,000 is relatively low. Percentile curves of this type are useful in medical science as reference curves (Yu, Lu, and Stander 2003).



**Output 98.5.5** Salary and Years as Professor: Regression Quantiles

## Example 98.6: Fast Quantile Process Regression

Quantile process regression fits quantile regression models for the entire range of quantile levels from 0 to 1, thus estimating the entire probability distribution of the response conditional on the covariates. Fast quantile process regression can efficiently approximate quantile process regression on a quantile-level grid. This example demonstrates how you can conduct fast quantile process regression analysis by using the QUANTILE=FQPR option.

### Parameter Estimates for Quantile Process Regression

The following statements simulate the data set analysisData, where the distribution of the response  $y$  conditional on the covariates  $x_1$  and  $x_2$  gradually changes from a normal distribution to an exponential distribution:

```
%let seed=123;
%let n=6001;
%let model=x1 x2;
data analysisData;
  do i=1 to &n;
    x1=(i-1)/(&n-1);
    x2=(1-x1)*(1-x1);
    y =x1*ranexp(&seed)+x2*(rannor(&seed)-3);
    output;
  end;
run;
```

The following statements use the QUANTILE=FQPR option to fit the fast quantile process regression model:

$$y = \beta_0 + x_1\beta_1 + x_2\beta_2:$$

```
proc quantreg data=analysisData;
  ods output ProcessEst=fqprEst
             AvgParameterEst=fqprAvgEst;
  model y = &model / quantile=fqpr(n=500);
run;
```

The N=500 suboption in the QUANTILE=FQPR option specifies an equally spaced grid of 500 quantile levels for the fast quantile process regression. The ODS OUTPUT statement outputs the data set fqprEst for the quantile process parameter estimates table and the data set fqprAvgEst for the average parameter estimates table.

[Output 98.6.1](#) shows the average parameter estimates of the fitted fast quantile process, and [Output 98.6.2](#) shows the quantile levels and the average objective function value.

#### **Output 98.6.1** Average Parameter Estimates for Quantile Process

##### **The QUANTREG Procedure**

Average Parameter Estimates		
Parameter	DF	Estimate
Intercept	1	-0.2248
x1	1	1.2701
x2	1	-2.7625

#### **Output 98.6.2** Average Objective Function Values for Quantile Process

Quantile Levels and Average Objective Function	
Number of Quantile Levels	500
Minimum Quantile Level	0.0010
Maximum Quantile Level	0.9990
Average Objective Function	1122.1630
Average Predicted Value at Mean	-0.5106

The following statements print the first 10 parameter estimates of the fast quantile process, as shown in [Output 98.6.3](#).

```
proc print data=fqprEst (obs=10);
run;
```

For more information about parameter estimates for the fast quantile process, see the section “[Fast Quantile Process Regression](#)” on page 8003.

**Output 98.6.3** Quantile Process Parameter Estimates

Obs	QuantileLabel	QuantileLevel	Intercept	x1	x2
1	t0	0.001000	0.2954	-0.2964	-6.0015
2	t1	0.003000	0.5281	-0.5487	-6.1878
3	t2	0.005000	0.2224	-0.2230	-5.4191
4	t3	0.007000	0.3619	-0.3677	-5.5338
5	t4	0.009000	0.4732	-0.4846	-5.6295
6	t5	0.011000	0.4946	-0.5070	-5.5893
7	t6	0.013000	0.4666	-0.4737	-5.4900
8	t7	0.015000	0.5064	-0.5137	-5.5052
9	t8	0.017000	0.5468	-0.5572	-5.4800
10	t9	0.019000	0.5706	-0.5802	-5.4826

**Observationwise Distribution Estimation**

Quantile process regression can estimate the entire distribution of a response variable conditional on its covariates. The following statements use the IML procedure in SAS/IML to create macro variables for observation indices, observation quantile levels, observation mean predictions, and limits of quantile predictions and to create a data set, `distData`, that contains all quantile levels and quantile predictions for the specified observations:

```
proc iml;

    /* Specify observations to be plotted */
    Obs = {1 3001 6001};
    nObs = ncol(Obs);
    call symputx("nObs",nObs);

    /* Load training data */
    use analysisData;
    read all var {&model} into x;
    read all var {"y"} into y;
    close analysisData;

    /* Load parameter estimates for the quantile process */
    use fqprEst;
    read all var {"Intercept"} into beta0;
    read all var {&model} into beta;
    read all var {"QuantileLevel"} into qLev;
    close fqprEst;

    /* Load average parameter estimates for the quantile process */
    use fqprAvgEst;
    read all var {"Estimate"} into avgBeta;
    close fqprAvgEst;

    nTau = nrow(qLev);
    qPrCs = j((nTau*nObs),3);
    obsInfo = j(nObs,6);

    /* Make macro variable names */
```

```

obsIndex = "_Obs1": "_Obs&nObs";
levNames = "_qLev1": "_qLev&nObs";
qtNames = "_qt1": "_qt&nObs";
qmNames = "_qMean1": "_qMean&nObs";
gLNames = "_gridL1": "_gridL&nObs";
gUNames = "_gridU1": "_gridU&nObs";

/* Define a function for computing observation quantile level */
start QntLev(y, sample);
  call sort(sample, 1);
  nL = ncol(loc(sample < y));
  nH = ncol(loc(sample > y));
  nr = nrow(sample);
  if      nL=0      then qtlev = 0.25/nr;
  else if nH=0      then qtlev = 1-(0.25/nr);
  else if nr > (nL+nH) then qtlev = 0.5*(nL+nr-nH)/nr;
  else qtlev = (nL-0.5+(y-sample[nL])/(sample[nL+1]-sample[nL]))/nr;
  return qtlev;
finish;

do j=1 to nObs;
  iObs = Obs[j];

  /* Compute quantile process prediction for each specified observation */
  Quantiles = beta0 + beta*t(x[iObs,]);

  /* Compute average of the quantile process prediction */
  Average = avgBeta[1]+x[iObs,]*avgBeta[2:3];

  /* Compute observation quantile level */
  QuantLev = QntLev(y[iObs], Quantiles);

  /* Save quantile process prediction */
  qPrCs[((j-1)*nTau+1):(j*nTau), 1]=iObs;
  qPrCs[((j-1)*nTau+1):(j*nTau), 2]=qLev;
  qPrCs[((j-1)*nTau+1):(j*nTau), 3]=Quantiles;

  /* Make macro variables */
  call symputx(obsIndex[j], iObs);
  call symputx(qtNames[j], y[iObs]);
  call symputx(levNames[j], QuantLev);
  call symputx(qmNames[j], Average);
  call symputx(gLNames[j], Quantiles[1]);
  call symputx(gUNames[j], Quantiles[nTau]);

  /* Save observation quantile level and average of
     the quantile process prediction */
  obsInfo[j, 1]=iObs;
  obsInfo[j, 2]=y[iObs];
  obsInfo[j, 3]=x[iObs, 1];
  obsInfo[j, 4]=x[iObs, 2];
  obsInfo[j, 5]=QuantLev;
  obsInfo[j, 6]=Average;
end;

```

```

/* Print observation information */
obsInfoColName = {"Index" "Response Value" "x1" "x2"
                  "Quantile Level" "Mean Prediction"};
obsInfoLabel = {"Information for Specified Observations"};
print obsInfo[colname=obsInfoColName label=obsInfoLabel];

/* Store all quantile process predictions into a SAS data set */
create distData from qPrCs[colname={"iObs" "qLev" "Quantiles"}];
append from qPrCs;
close distData;
quit;

```

The function QntLev in this SAS/IML program computes the quantile level of a specified value according to a sample. Let  $\{s_1, s_2, \dots, s_n\}$  denote the sample. QntLev first sorts the sample in ascending order as  $\{s_{(1)} \leq s_{(2)} \leq \dots \leq s_{(n)}\}$ . Then the quantile level of the specified value  $y$  is computed as follows:

$$\hat{\tau}(y) = \begin{cases} 0.25/n & \text{if } y < s_{(1)} \\ 1 - (0.25/n) & \text{if } y > s_{(n)} \\ \left( i - 0.5 + \frac{y - s_{(i)}}{s_{(i+1)} - s_{(i)}} \right) / n & \text{if } s_{(i)} < y < s_{(i+1)} \\ 0.5(i + j - 1) / n & \text{if } s_{(i)} < y = s_{(i+1)} = \dots = s_{(j-1)} < s_{(j)} \\ & \text{where } s_{(0)} = -\infty \text{ and } s_{(n+1)} = \infty \text{ for simplicity} \end{cases}$$

Output 98.6.4 shows the observation information table for the specified observations.

**Output 98.6.4** Information for Observations 1, 3001, and 6001

Information for Specified Observations						
	Index	Response Value	x1	x2	Quantile Level	Mean Prediction
ROW1	1	-2.34428	0	1	0.749	-2.987244
ROW2	3001	-0.557309	0.5	0.25	0.343017	-0.280339
ROW3	6001	0.9299095	1	0	0.579	1.0453271

The SAS/IML program creates macro variables for the specified observations (\_Obs1, \_Obs3001, and \_Obs6001), their observation quantile levels (\_qt1, \_qt3001, and \_qt6001), and their associated cumulative probabilities (\_qLev1, \_qLev3001, and \_qLev6001). The following statements use these macro values to plot the conditional cumulative distribution functions (CDFs) for the specified observations:

```

data distData;
  set distData;
  label iObs = "Observation Index"
        qLev = "Cumulative Probability"
        Quantiles = "Quantile";
run;

/* Make CDF plot for specified observations */
%macro plotCDF;
  proc sgplot data=distData;
    series y=qLev x=Quantiles/group=iObs;

```

```

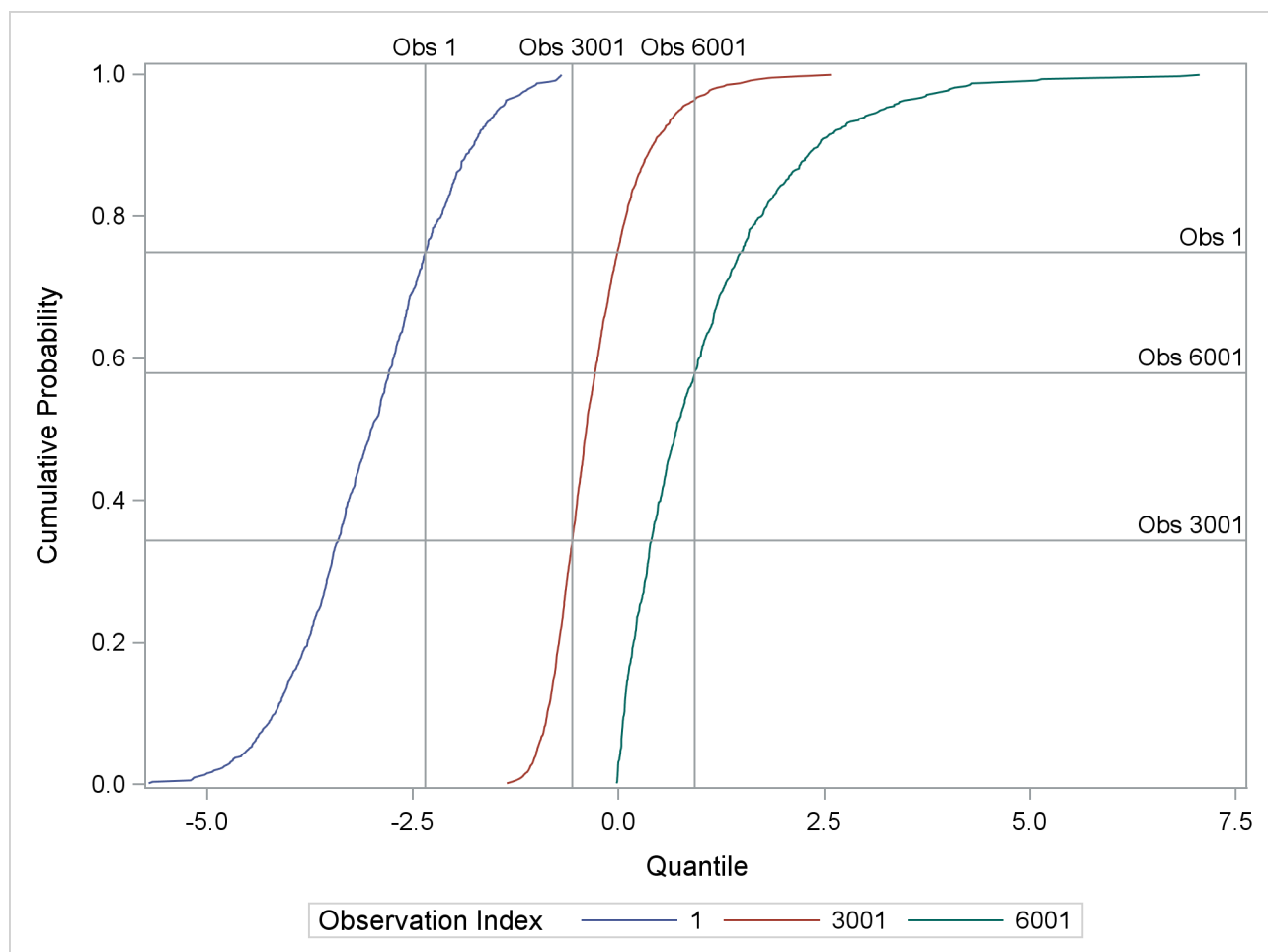
%do j=1 %to &nObs;
  refline &&_qLev&j/label=("Obs &&_Obs&j")
              axis=y labelloc=inside;
  refline &&_qt&j/ label=("Obs &&_Obs&j")
              axis=x;
%end;
run;
%mend;

%plotCDF;

```

Output 98.6.5 displays the conditional CDFs for the three specified observations. Each observation has a vertical reference line for its observed response value and a horizontal reference line for its quantile level.

**Output 98.6.5** Cumulative Distribution Functions



You can also estimate the conditional probability density function (PDF) for the specified observations by using the KDE procedure in SAS/STAT. The following SAS/IML code creates the data set probData, which contains the probability estimates for each predicted quantile of the specified observations in variables Obs1, Obs3001, and Obs6001, and then uses a macro that runs PROC KDE to plot their PDFs:

```

proc iml;
  /* Load quantile process predictions */
  use distData;
  read all var {"iObs"} into iObs;
  read all var {"qLev"} into qLev;
  read all var {"Quantiles"} into Y;
  close distData;

  nObs = &nObs;
  nTau = nrow(qLev)/nObs;
  pProb = j(nTau, 3+nObs);

  pProb[,1]          = t(1:nTau);
  pProb[,2]          = qLev[1:nTau];

  /* Compute weights for quantile predictions */
  pProb[1,3]          = (qLev[1]+qLev[2])/2;
  pProb[nTau,3]       = 1-(qLev[nTau-1]+qLev[nTau])/2;
  pProb[2:(nTau-1),3] = (qLev[3:nTau]-qLev[1:(nTau-2)])/2;

  do j=1 to nObs;
    jump = (j-1)*nTau;
    pProb[, (3+j)] = Y[(jump+1):(jump+nTau)];
  end;

  /* Store all quantile process predictions and their weights
     into a SAS data set */
  create probData from pProb[colname={"obs" "quantLev" "pProb"
                                     "Obs1" "Obs3001" "Obs6001"}];

  append from pProb;
  close probData;
quit;

/* Make PDF plot for specified observations */
%macro plotPDF;
  proc kde data=probData;
    weight pProb;
    univar
      %do j=1 %to &nObs;
        Obs&&_Obs&j (gridL=&&_gridL&j gridU=&&_gridU&j)
      %end;
    / plots=densityoverlay;
  run;
%mend;

%plotPDF;

```

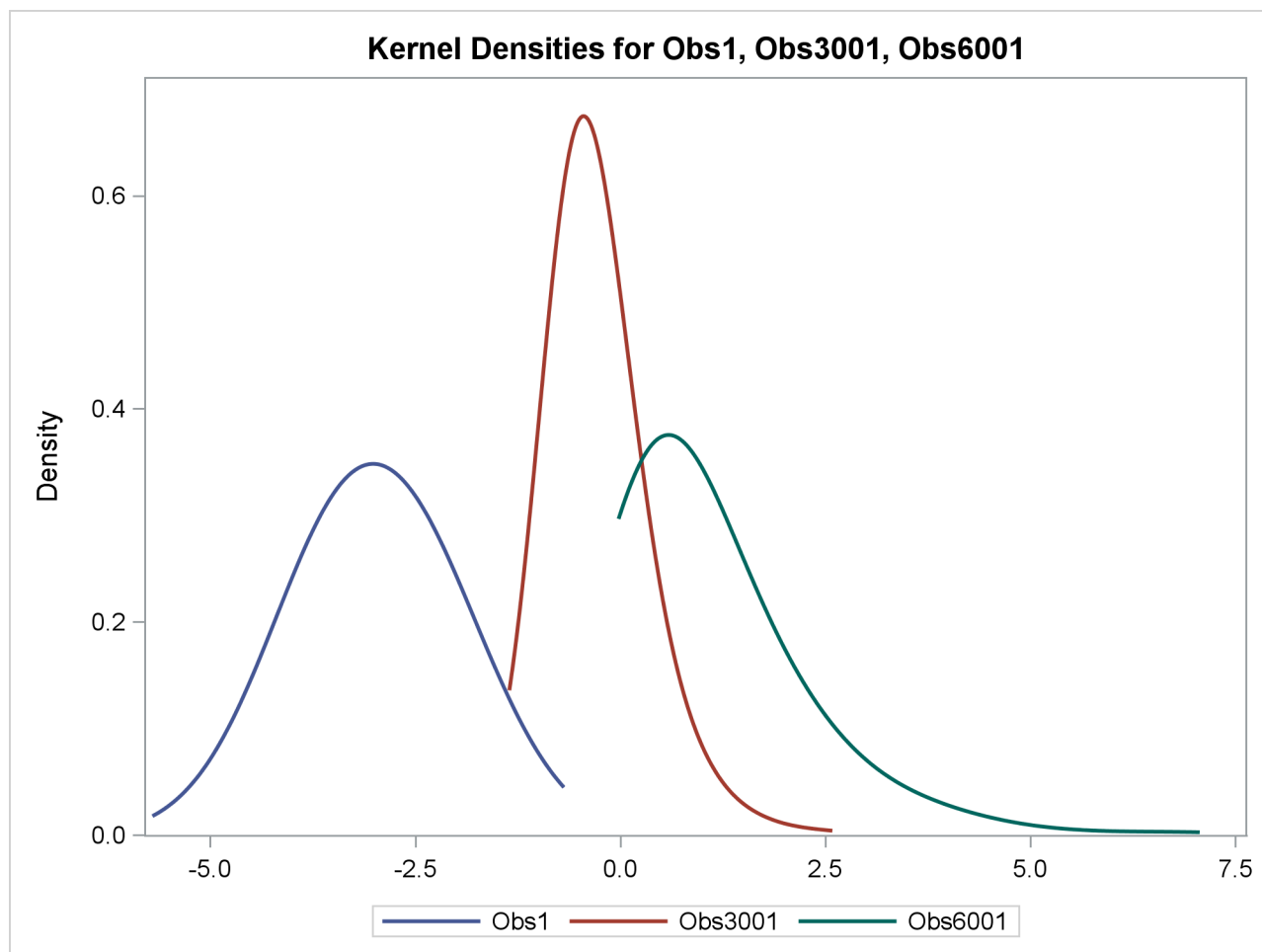
Let  $\{\tau_1 < \tau_2 < \dots < \tau_q\}$  denote a quantile-level grid, which is not necessarily equally spaced. The probability estimate of a quantile prediction at quantile level  $\tau_i$  for the relevant quantile process is defined as follows:

$$\hat{\tau}(y) = \begin{cases} 0.5(\tau_1 + \tau_2) & \text{if } i = 1 \\ 1 - 0.5(\tau_{n-1} + \tau_n) & \text{if } i = n \\ 0.5(\tau_{i+1} - \tau_{i-1}) & \text{otherwise} \end{cases}$$

These probability estimates for the quantile process and are then used as weights in the WEIGHT statement of the KDE procedure.

Output 98.6.6 displays the density estimation plots for the specified observations. You can see that, approximately, the PDF for Observation 1 is a normal PDF, the PDF for Observation 3001 is slightly right-skewed, and the PDF for Observation 6001 is an exponential PDF.

**Output 98.6.6** Probability Density Functions





## References

- Abreveya, J. (2001). "The Effects of Demographics and Maternal Behavior on the Distribution of Birth Outcomes." *Journal of Economics* 26:247–257.
- Barro, R., and Lee, J. W. (1994). "Data Set for a Panel of 138 Countries." Discussion paper, National Bureau of Econometric Research. <http://admin.nber.org/pub/barro.lee/readme.txt>.
- Barrodale, I., and Roberts, F. D. K. (1973). "An Improved Algorithm for Discrete  $l_1$  Linear Approximation." *SIAM Journal on Numerical Analysis* 10:839–848.
- Bassett, G. W., and Koenker, R. (1982). "An Empirical Quantile Function for Linear Models with iid Errors." *Journal of the American Statistical Association* 77:401–415.
- Cade, B. S., and Noon, B. R. (2003). "A Gentle Introduction to Quantile Regression for Ecologists." *Frontiers in Ecology and the Environment* 1:412–420.
- Chen, C. (2004). "An Adaptive Algorithm for Quantile Regression." In *Theory and Applications of Recent Robust Methods*, edited by M. Hubert, G. Pison, A. Struyf, and S. V. Aels, 39–48. Basel: Birkhäuser.
- Chen, C. (2005). "Growth Charts of Body Mass Index (BMI) with Quantile Regression." In *Proceedings of 2005 International Conference on Algorithmic Mathematics and Computer Science*, edited by H. R. Arabnia and I. A. Ajwa, 114–120. Bogart, GA: CSREA Press.
- Chen, C. (2007). "A Finite Smoothing Algorithm for Quantile Regression." *Journal of Computational and Graphical Statistics* 16:136–164.
- Chock, D. P., Winkler, S. L., and Chen, C. (2000). "A Study of the Association between Daily Mortality and Ambient Air Pollutant Concentrations in Pittsburgh, Pennsylvania." *Journal of the Air and Waste Management Association* 50:1481–1500.
- Dunham, J. B., Cade, B. S., and Terrell, J. W. (2002). "Influences of Spatial and Temporal Variation on Fish-Habitat Relationships Defined by Regression Quantiles." *Transactions of the American Fisheries Society* 131:86–98.
- Gutenbrunner, C., and Jureckova, J. (1992). "Regression Rank Scores and Regression Quantiles." *Annals of Statistics* 20:305–330.
- Gutenbrunner, C., Jureckova, J., Koenker, R., and Portnoy, S. (1993). "Tests of Linear Hypotheses Based on Regression Rank Scores." *Journal of Nonparametric Statistics* 2:307–331.
- He, X., and Hu, F. (2002). "Markov Chain Marginal Bootstrap." *Journal of the American Statistical Association* 97:783–795.
- Huber, P. J. (1981). *Robust Statistics*. New York: John Wiley & Sons.
- Karmarkar, N. (1984). "A New Polynomial-Time Algorithm for Linear Programming." *Combinatorica* 4:373–395.
- Koenker, R. (1994). "Confidence Intervals for Regression Quantiles." In *Asymptotic Statistics*, edited by P. Mandl and M. Huskova, 349–359. New York: Springer-Verlag.

- Koenker, R. (2005). *Quantile Regression*. New York: Cambridge University Press.
- Koenker, R., and Bassett, G. W. (1978). "Regression Quantiles." *Econometrica* 46:33–50.
- Koenker, R., and Bassett, G. W. (1982a). "Robust Tests for Heteroscedasticity Based on Regression Quantiles." *Econometrica* 50:43–61.
- Koenker, R., and Bassett, G. W. (1982b). "Tests of Linear Hypotheses and  $l_1$  Estimation." *Econometrica* 50:1577–1583.
- Koenker, R., and d'Orey, V. (1994). "Remark AS R92: A Remark on Algorithm AS 229: Computing Dual Regression Quantiles and Regression Rank Scores." *Journal of the Royal Statistical Society, Series C* 43:410–414.
- Koenker, R., and Hallock, K. (2001). "Quantile Regression: An Introduction." *Journal of Economic Perspectives* 15:143–156.
- Koenker, R., and Machado, A. F. (1999). "Goodness of Fit and Related Inference Processes for Quantile Regression." *Journal of the American Statistical Association* 94:1296–1310.
- Koenker, R., and Zhao, Q. (1994). "L-Estimation for Linear Heteroscedastic Models." *Journal of Nonparametric Statistics* 3:223–235.
- Kuczmariski, R. J., Ogden, C. L., and Guo, S. S. (2002). "2000 CDC Growth Charts for the United States: Methods and Development." *Vital and Health Statistics* 11:1–190.
- Lustig, I. J., Marsten, R. E., and Shanno, D. F. (1992). "On Implementing Mehrotra's Predictor-Corrector Interior-Point Method for Linear Programming." *SIAM Journal on Optimization* 2:435–449.
- Madsen, K., and Nielsen, H. B. (1993). "A Finite Smoothing Algorithm for Linear  $L_1$  Estimation." *SIAM Journal on Optimization* 3:223–235.
- Parzen, M. I., Wei, L. J., and Ying, Z. (1994). "A Resampling Method Based on Pivotal Estimating Functions." *Biometrika* 81:341–350.
- Portnoy, S., and Koenker, R. (1997). "The Gaussian Hare and the Laplacian Tortoise: Computation of Squared-Error vs. Absolute-Error Estimators." *Statistical Science* 12:279–300.
- Roos, C., Terlaky, T., and Vial, J. (1997). *Theory and Algorithms for Linear Optimization*. Chichester, UK: John Wiley & Sons.
- Rousseeuw, P. J., and Van Driessen, K. (1999). "A Fast Algorithm for the Minimum Covariance Determinant Estimator." *Technometrics* 41:212–223.
- Yao, Y. (2017). "Fast Quantile Process Regression." Paper presented at the 2017 International Conference on Robust Statistics, Wollongong, NSW, Australia.
- Yu, K., Lu, Z., and Stander, J. (2003). "Quantile Regression: Application and Current Research Areas." *Journal of the Royal Statistical Society, Series D* 52:331–350.

# Subject Index

- affine step
  - QUANTREG procedure, [7998](#)
- centering step
  - QUANTREG procedure, [7999](#)
- complementarity
  - QUANTREG procedure, [7998](#)
- computational resources
  - QUANTREG procedure, [8011](#)
- INEST= data sets
  - QUANTREG procedure, [8010](#)
- infeasibility
  - QUANTREG procedure, [7998](#)
- Karush-Kuhn-Tucker (KKT) conditions
  - QUANTREG procedure, [7997](#)
- ODS Graphics names
  - QUANTREG procedure, [8013](#)
- options summary
  - EFFECT statement, [7985](#)
  - ESTIMATE statement, [7987](#)
- OUTEST= data sets
  - QUANTREG procedure, [8010](#)
- output table names
  - QUANTREG procedure, [8011](#)
- primal-dual with predictor-corrector algorithm
  - QUANTREG procedure, [7998](#)
- QUANTREG procedure, [7964](#)
  - affine step, [7998](#)
  - centering step, [7999](#)
  - complementarity, [7998](#)
  - computational resources, [8011](#)
  - INEST= data sets, [8010](#)
  - infeasibility, [7998](#)
  - Karush-Kuhn-Tucker (KKT) conditions, [7997](#)
  - ODS Graphics names, [8013](#)
  - ordering of effects, [7981](#)
  - OUTEST= data sets, [8010](#)
  - output table names, [8011](#)
  - primal-dual with predictor-corrector algorithm, [7998](#)
- QUANTTREG procedure
  - syntax, [7978](#)
- syntax
  - QUANTTREG procedure, [7978](#)



# Syntax Index

- ALGORITHM option
  - PROC QUANTREG statement, [7979](#)
- ALPHA= option
  - PROC QUANTREG (QUANTREG), [7980](#)
- BY statement
  - QUANTREG procedure, [7984](#)
- CI option
  - PROC QUANTREG statement, [7980](#)
- CLASS statement
  - QUANTREG procedure, [7984](#)
- CORRB option
  - MODEL statement (QUANTREG), [7988](#)
- COVB option
  - MODEL statement (QUANTREG), [7989](#)
- CPUCOUNT option
  - PERFORMANCE statement (QUANTREG), [7993](#)
- CUTOFF option
  - MODEL statement (QUANTREG), [7989](#)
- DATA= option
  - PROC QUANTREG statement, [7981](#)
- DETAILS option
  - PERFORMANCE statement (QUANTREG), [7993](#)
- DIAGNOSTICS option
  - MODEL statement (QUANTREG), [7989](#)
- EFFECT statement
  - QUANTREG procedure, [7985](#)
- ESTIMATE statement
  - QUANTREG procedure, [7986](#)
- ID statement
  - QUANTREG procedure, [7987](#)
- INEST= option
  - PROC QUANTREG statement, [7981](#)
- ITPRINT option
  - MODEL statement, [7989](#)
- KAPPA= option
  - PROC QUANTREG statement, [7979](#)
- keyword= option
  - OUTPUT statement (QUANTREG), [7992](#)
- LEVERAGE keyword
  - OUTPUT statement (QUANTREG), [7992](#)
- LEVERAGE option
  - MODEL statement, [7989](#)
- LR option
  - TEST statement (QUANTREG), [7994](#)
- MAHADIST keyword
  - OUTPUT statement (QUANTREG), [7992](#)
- MAXIT= option
  - PROC QUANTREG statement, [7979](#)
- MAXSTATIONARY= option
  - PROC QUANTREG statement, [7979](#)
- MODEL statement
  - QUANTREG procedure, [7988](#)
- NAMELEN= option
  - PROC QUANTREG statement, [7981](#)
- NODIAG option
  - MODEL statement (QUANTREG), [7989](#)
- NOINT option
  - MODEL statement (QUANTREG), [7989](#)
- NOSUMMARY option
  - MODEL statement (QUANTREG), [7989](#)
- NOTHEADS option
  - PERFORMANCE statement (QUANTREG), [7993](#)
- OPTION statement
  - QUANTREG procedure, [7988](#)
- ORDER= option
  - PROC QUANTREG statement, [7981](#)
- OUT= option
  - OUTPUT statement (QUANTREG), [7992](#)
- OUTEST= option
  - PROC QUANTREG statement, [7982](#)
- OUTLIER keyword
  - OUTPUT statement (QUANTREG), [7992](#)
- OUTPUT statement
  - QUANTREG procedure, [7992](#)
- PERFORMANCE statement
  - QUANTREG procedure, [7993](#)
- PP= option
  - PROC QUANTREG statement, [7984](#)
- PREDICTED keyword
  - OUTPUT statement (QUANTREG), [7992](#)
- PROC QUANTREG statement, *see* QUANTREG procedure
- QUANTILE option

- MODEL statement (QUANTREG), 7990
- QUANTILES keyword
  - OUTPUT statement (QUANTREG), 7992
- QUANTREG procedure
  - EFFECT statement, 7985
- QUANTREG procedure, BY statement, 7984
- QUANTREG procedure, CLASS statement, 7984
  - TRUNCATE option, 7985
- QUANTREG procedure, EFFECT statement, 7985
- QUANTREG procedure, ESTIMATE statement, 7986
- QUANTREG procedure, ID statement, 7987
- QUANTREG procedure, MODEL statement, 7988
  - CORRB option, 7988
  - COVB option, 7989
  - CUTOFF option, 7989
  - DIAGNOSTICS option, 7989
  - ITPRINT option, 7989
  - LEVERAGE option, 7989
  - NODIAG option, 7989
  - NOINT option, 7989
  - NOSUMMARY option, 7989
  - PLOT= plot option, 7989
  - QUANTILE option, 7990
  - SCALE option, 7991
  - SINGULAR= option, 7991
- QUANTREG procedure, OPTION2 statement, 7988
- QUANTREG procedure, OUTPUT statement, 7992
  - keyword= option, 7992
  - LEVERAGE keyword, 7992
  - MAHADIST keyword, 7992
  - OUT= option, 7992
  - OUTLIER keyword, 7992
  - PREDICTED keyword, 7992
  - QUANTILES keyword, 7992
  - RESIDUAL keyword, 7992
  - ROBDIST keyword, 7993
  - SPLINE keyword, 7993
  - SRESIDUAL keyword, 7993
  - STD\_ERR keyword, 7993
- QUANTREG procedure, PERFORMANCE statement, 7993
  - CPUCOUNT option, 7993
  - DETAILS option, 7993
  - NOTHEADS option, 7993
  - THREADS option, 7993
- QUANTREG procedure, PROC QUANTREG statement, 7978
  - ALGORITHM option, 7979
  - ALPHA= option, 7980
  - CI option, 7980
  - DATA= option, 7981
  - INEST= option, 7981
  - KAPPA= option, 7979
  - MAXIT= option, 7979
  - MAXSTATIONARY= option, 7979
  - NAMELEN= option, 7981
  - ORDER= option, 7981
  - OUTEST= option, 7982
  - PLOT= plot option, 7982
  - PP option, 7984
  - RRATIO= option, 7980
  - TOLERANCE= option, 7979, 7980
- QUANTREG procedure, TEST statement, 7994
  - LR option, 7994
  - RANKSCORE option, 7994
  - WALD option, 7994
- QUANTREG procedure, WEIGHT statement, 7994
- QUANTREG procedure, MODEL statement
  - SEED option, 7991
- RANKSCORE option
  - TEST statement (QUANTREG), 7994
- RESIDUAL keyword
  - OUTPUT statement (QUANTREG), 7992
- ROBDIST keyword
  - OUTPUT statement (QUANTREG), 7993
- RRATIO= option
  - PROC QUANTREG statement, 7980
- SCALE option
  - MODEL statement (QUANTREG), 7991
- SEED option
  - MODEL statement (QUANTREG), 7991
- SINGULAR= option
  - MODEL statement (QUANTREG), 7991
- SPLINE keyword
  - OUTPUT statement (QUANTREG), 7993
- SRESIDUAL keyword
  - OUTPUT statement (QUANTREG), 7993
- STD\_ERR keyword
  - OUTPUT statement (QUANTREG), 7993
- TEST statement
  - QUANTREG procedure, 7994
- THREADS option
  - PERFORMANCE statement (QUANTREG), 7993
- TOLERANCE= option
  - PROC QUANTREG statement, 7979, 7980
- TRUNCATE option
  - CLASS statement (QUANTREG), 7985
- WALD option
  - TEST statement (QUANTREG), 7994
- WEIGHT statement
  - QUANTREG procedure, 7994