

Five Things You Should Know about Quantile Regression

Robert N. Rodriguez and Yonggang Yao, SAS Institute Inc.

Abstract

The increasing complexity of data in research and business analytics requires versatile, robust, and scalable methods of building explanatory and predictive statistical models. Quantile regression meets these requirements by fitting conditional quantiles of the response with a general linear model that assumes no parametric form for the conditional distribution of the response; it gives you information that you would not obtain directly from standard regression methods. Quantile regression yields valuable insights in applications such as risk management, where answers to important questions lie in modeling the tails of the conditional distribution. Furthermore, quantile regression is capable of modeling the entire conditional distribution; this is essential for applications such as ranking the performance of students on standardized exams. This expository paper explains the concepts and benefits of quantile regression, and it introduces you to the appropriate procedures in SAS/STAT® software.

Introduction

Students taking their first course in statistics learn to compute quantiles—more commonly referred to as percentiles—as descriptive statistics. But despite the widespread use of quantiles for data summarization, relatively few statisticians and analysts are acquainted with quantile regression as a method of statistical modeling, despite the availability of powerful computational tools that make this approach practical and advantageous for large data.

Quantile regression brings the familiar concept of a quantile into the framework of general linear models,

$$y_i = \beta_0 + \beta_1 x_{i1} + \cdots + \beta_p x_{ip} + \epsilon_i, \quad i = 1, \dots, n$$

where the response y_i for the i th observation is continuous, and the predictors x_{i1}, \dots, x_{ip} represent main effects that consist of continuous or classification variables and their interactions or constructed effects. Quantile regression, which was introduced by Koenker and Bassett (1978), fits specified percentiles of the response, such as the 90th percentile, and can potentially describe the entire conditional distribution of the response.

This paper provides an introduction to quantile regression for statistical modeling; it focuses on the benefits of modeling the conditional distribution of the response as well as the procedures for quantile regression that are available in SAS/STAT software. The paper is organized into six sections:

- Basic Concepts of Quantile Regression
- Fitting Quantile Regression Models
- Building Quantile Regression Models
- Applying Quantile Regression to Financial Risk Management
- Applying Quantile Process Regression to Ranking Exam Performance
- Summary

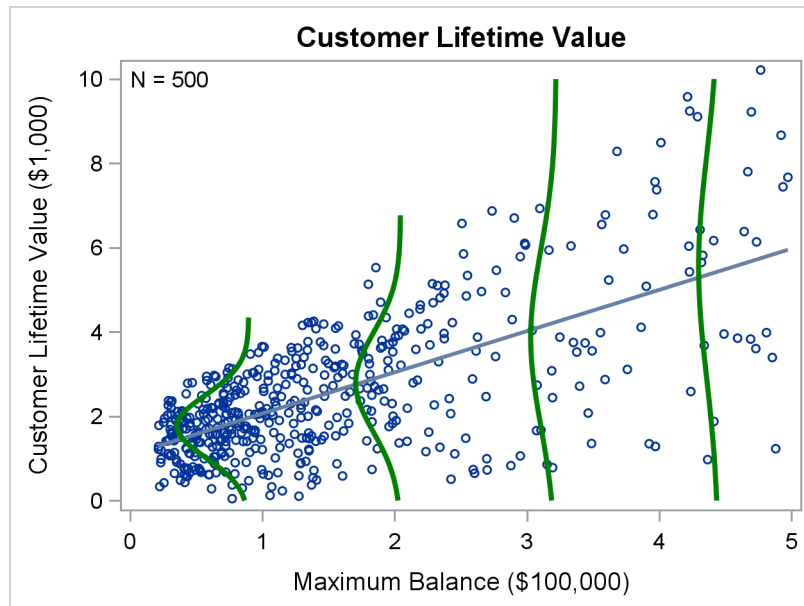
The first five sections present examples that illustrate the concepts and benefits of quantile regression along with procedure syntax and output. The summary distills these examples into five key points that will help you add quantile regression to your statistical toolkit.

Basic Concepts of Quantile Regression

Although quantile regression is most often used to model specific conditional quantiles of the response, its full potential lies in modeling the entire conditional distribution. By comparison, standard least squares regression models only the conditional mean of the response and is computationally less expensive. Quantile regression does not assume a particular parametric distribution for the response, nor does it assume a constant variance for the response, unlike least squares regression.

Figure 1 presents an example of regression data for which both the mean and the variance of the response increase as the predictor increases. In these data, which represent 500 bank customers, the response is the customer lifetime value (CLV) and the predictor is the maximum balance of the customer's account. The line represents a simple linear regression fit.

Figure 1 Variance of Customer Lifetime Value Increases with Maximum Balance

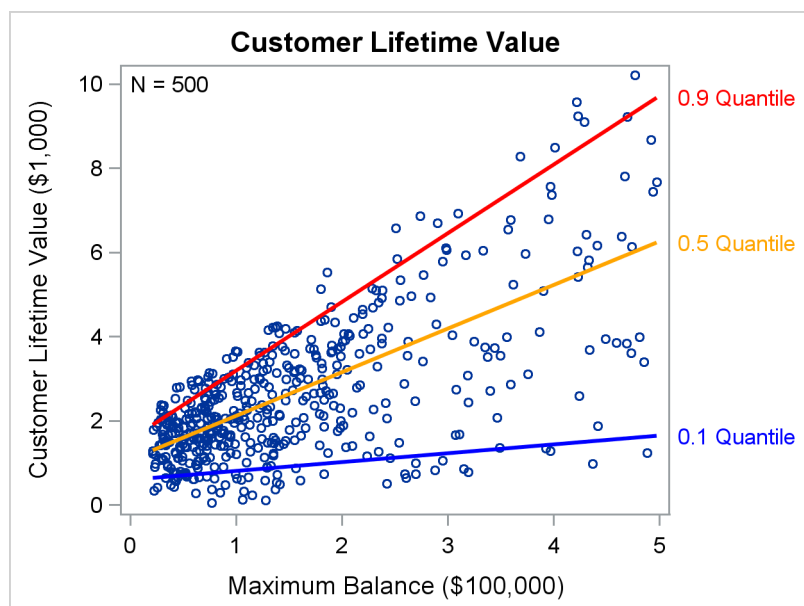


Least squares regression for a response Y and a predictor X models the conditional mean $E[Y|X]$, but it does not capture the conditional variance $\text{Var}[Y|X]$, much less the conditional distribution of Y given X .

The green curves in Figure 1 represent the conditional densities of CLV for four specific values of maximum balance. A set of densities for a comprehensive grid of values of maximum balance would provide a complete picture of the conditional distribution of CLV given maximum balance. Note that the densities shown here are normal only for the purpose of illustration.

Figure 2 shows fitted linear regression models for the quantile levels 0.10, 0.50, and 0.90, or equivalently, the 10th, 50th, and 90th percentiles.

Figure 2 Regression Models for Quantile Levels 0.10, 0.50, and 0.90

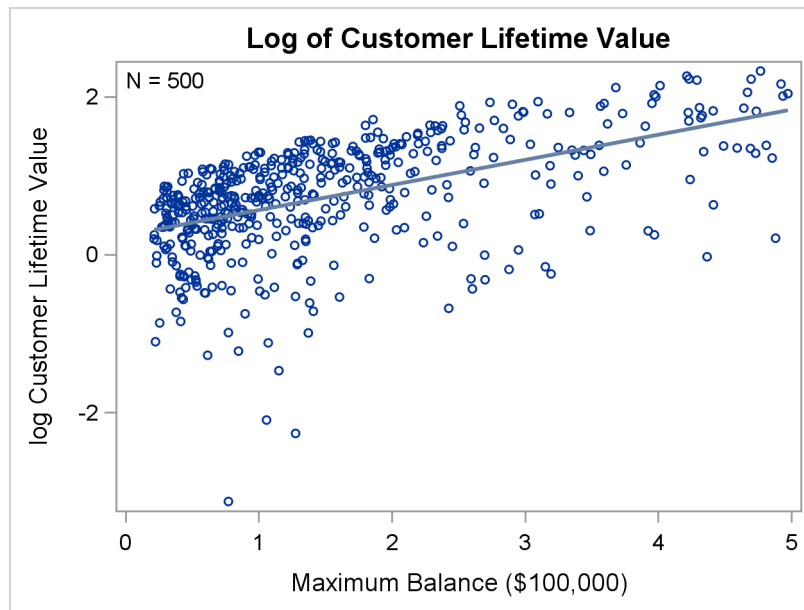


The quantile level is the probability (or the proportion of the population) that is associated with a quantile. The quantile level is often denoted by the Greek letter τ , and the corresponding conditional quantile of Y given X is often written as $Q_\tau(Y|X)$. The quantile level τ is the probability $\Pr[Y \leq Q_\tau(Y|X)|X]$, and it is the value of Y below which the proportion of the conditional response population is τ .

By fitting a series of regression models for a grid of values of τ in the interval $(0,1)$, you can describe the entire conditional distribution of the response. The optimal grid choice depends on the data, and the more data you have, the more detail you can capture in the conditional distribution.

Quantile regression gives you a principled alternative to the usual practice of stabilizing the variance of heteroscedastic data with a monotone transformation $h(Y)$ before fitting a standard regression model. Depending on the data, it is often not possible to find a simple transformation that satisfies the assumption of constant variance. This is evident in Figure 3, where the variance of $\log(\text{CLV})$ increases for maximum balances near \$100,000, and the conditional distributions are asymmetric.

Figure 3 Log Transformation of CLV



Even when a transformation does satisfy the assumptions for standard regression, the inverse transformation does not predict the mean of the response when applied to the predicted mean of the transformed response:

$$E(Y|X) \neq h^{-1}(E(h(Y)|X))$$

In contrast, the inverse transformation can be applied to the predicted quantiles of the transformed response:

$$Q_\tau(Y|X) = h^{-1}(Q_\tau(h(Y)|X))$$

Table 1 summarizes some important differences between standard regression and quantile regression.

Table 1 Comparison of Linear Regression and Quantile Regression

Linear Regression	Quantile Regression
Predicts the conditional mean $E(Y X)$	Predicts conditional quantiles $Q_\tau(Y X)$
Applies when n is small	Needs sufficient data
Often assumes normality	Is distribution agnostic
Does not preserve $E(Y X)$ under transformation	Preserves $Q_\tau(Y X)$ under transformation
Is sensitive to outliers	Is robust to response outliers
Is computationally inexpensive	Is computationally intensive

Koenker (2005) and Hao and Naiman (2007) provide excellent introductions to the theory and applications of quantile regression.

Fitting Quantile Regression Models

The standard regression model for the average response is

$$E(y_i) = \beta_0 + \beta_1 x_{i1} + \cdots + \beta_p x_{ip}, \quad i = 1, \dots, n$$

and the β_j 's are estimated by solving the least squares minimization problem

$$\min_{\beta_0, \dots, \beta_p} \sum_{i=1}^n \left(y_i - \beta_0 - \sum_{j=1}^p x_{ij} \beta_j \right)^2$$

In contrast, the regression model for quantile level τ of the response is

$$Q_\tau(y_i) = \beta_0(\tau) + \beta_1(\tau)x_{i1} + \cdots + \beta_p(\tau)x_{ip}, \quad i = 1, \dots, n$$

and the $\beta_j(\tau)$'s are estimated by solving the minimization problem

$$\min_{\beta_0(\tau), \dots, \beta_p(\tau)} \sum_{i=1}^n \rho_\tau \left(y_i - \beta_0(\tau) - \sum_{j=1}^p x_{ij} \beta_j(\tau) \right)$$

where $\rho_\tau(r) = \tau \max(r, 0) + (1 - \tau) \max(-r, 0)$. The function $\rho_\tau(r)$ is referred to as the check loss, because its shape resembles a check mark.

For each quantile level τ , the solution to the minimization problem yields a distinct set of regression coefficients. Note that $\tau = 0.5$ corresponds to median regression and $2\rho_{0.5}(r)$ is the absolute value function.

Example: Modeling the 10th, 50th, and 90th Percentiles of Customer Lifetime Value

Returning to the customer lifetime value example, suppose that the goal is to target customers with low, medium, and high value after adjusting for 15 covariates (**X1**, ..., **X15**), which include the maximum balance, average overdraft, and total credit card amount used. Assume that low, medium, and high correspond to the 10th, 50th, and 90th percentiles of customer lifetime value, or equivalently, the 0.10, 0.50, and 0.90 quantiles.

The QUANTREG procedure in SAS/STAT software fits quantile regression models and performs statistical inference. The following statements use the QUANTREG procedure to model the three quantiles:

```
proc quantreg data=CLV ci=sparsity ;
  model CLV = x1-x15 / quantiles=0.10 0.50 0.90;
run;
```

You use the QUANTILES= option to specify the level for each quantile.

Figure 4 shows the “Model Information” table that the QUANTREG procedure produces.

Figure 4 Model Information
The QUANTREG Procedure

Model Information	
Data Set	WORK.CLV
Dependent Variable	CLV
Number of Independent Variables	15
Number of Observations	500
Optimization Algorithm	Simplex
Method for Confidence Limits	Sparsity
<hr/>	
Number of Observations Read	500
Number of Observations Used	500

Figure 5 and Figure 6 show the parameter estimates for the 0.10 and 0.90 quantiles of CLV.

Figure 5 Parameter Estimates for Quantile Level 0.10

Parameter Estimates							
Parameter	DF	Estimate	Standard Error	95% Confidence Limits		t Value	Pr > t
Intercept	1	9.9046	0.0477	9.8109	9.9982	207.71	<.0001
X1	1	0.8503	0.0428	0.7662	0.9343	19.87	<.0001
X2	1	0.9471	0.0367	0.8750	1.0193	25.81	<.0001
X3	1	0.9763	0.0397	0.8984	1.0543	24.62	<.0001
X4	1	0.9256	0.0413	0.8445	1.0067	22.43	<.0001
X5	1	0.6670	0.0428	0.5828	0.7511	15.58	<.0001
X6	1	0.2905	0.0443	0.2034	0.3776	6.55	<.0001
X7	1	0.2981	0.0393	0.2208	0.3754	7.58	<.0001
X8	1	0.2094	0.0413	0.1283	0.2905	5.07	<.0001
X9	1	-0.0633	0.0423	-0.1464	0.0199	-1.49	0.1356
X10	1	0.0129	0.0400	-0.0658	0.0916	0.32	0.7473
X11	1	0.1084	0.0421	0.0257	0.1912	2.57	0.0103
X12	1	-0.0249	0.0392	-0.1019	0.0520	-0.64	0.5248
X13	1	-0.0505	0.0410	-0.1311	0.0300	-1.23	0.2182
X14	1	0.2009	0.0548	0.0932	0.3086	3.66	0.0003
X15	1	0.1623	0.0433	0.0773	0.2473	3.75	0.0002

Figure 6 Parameter Estimates for Quantile Level 0.90

Parameter Estimates							
Parameter	DF	Estimate	Standard Error	95% Confidence Limits		t Value	Pr > t
Intercept	1	10.1007	0.1386	9.8283	10.3730	72.87	<.0001
X1	1	0.0191	0.1485	-0.2726	0.3109	0.13	0.8975
X2	1	0.9539	0.1294	0.6996	1.2081	7.37	<.0001
X3	1	0.0721	0.1328	-0.1889	0.3332	0.54	0.5874
X4	1	1.1171	0.1243	0.8728	1.3613	8.99	<.0001
X5	1	-0.0317	0.1501	-0.3266	0.2631	-0.21	0.8326
X6	1	0.1096	0.1581	-0.2010	0.4202	0.69	0.4885
X7	1	0.2428	0.1436	-0.0394	0.5250	1.69	0.0915
X8	1	-0.0743	0.1364	-0.3424	0.1938	-0.54	0.5864
X9	1	0.0918	0.1401	-0.1835	0.3670	0.66	0.5127
X10	1	-0.2426	0.1481	-0.5336	0.0483	-1.64	0.1019
X11	1	0.9099	0.1414	0.6321	1.1878	6.44	<.0001
X12	1	0.7759	0.1353	0.5099	1.0418	5.73	<.0001
X13	1	0.5380	0.1392	0.2645	0.8115	3.87	0.0001
X14	1	0.6897	0.1475	0.3999	0.9796	4.68	<.0001
X15	1	1.0145	0.1516	0.7165	1.3124	6.69	<.0001

Note that the results in Figure 5 and Figure 6 are different. For example, the estimate for X1 is significant in the model for the 0.10 quantile, but it is not significant in the model for the 0.90 quantile. In general, quantile regression produces a distinct set of parameter estimates and predictions for each quantile level.

The QUANTREG procedure provides extensive features for statistical inference, which are not illustrated here. These include the following:

- simplex, interior point, and smooth algorithms for model fitting
- sparsity and bootstrap resampling methods for confidence limits
- Wald, likelihood ratio, and rank-score tests

You can also use PROC QUANTREG to carry out quantile process regression, which fits models for an entire grid of values of τ in the interval (0,1). The following statements illustrate quantile process regression by specifying a grid that is spaced uniformly in increments of 0.02:

```
ods output ParameterEstimates=Estimates;
proc quantreg data=CLV ci=sparsity ;
  model CLV = x1-x15 / quantiles=0.02 to 0.98 by 0.02;
run;
```

The next statements use the parameter estimates and confidence limits that PROC QUANTREG produces to create a quantile process plot for **X5**:

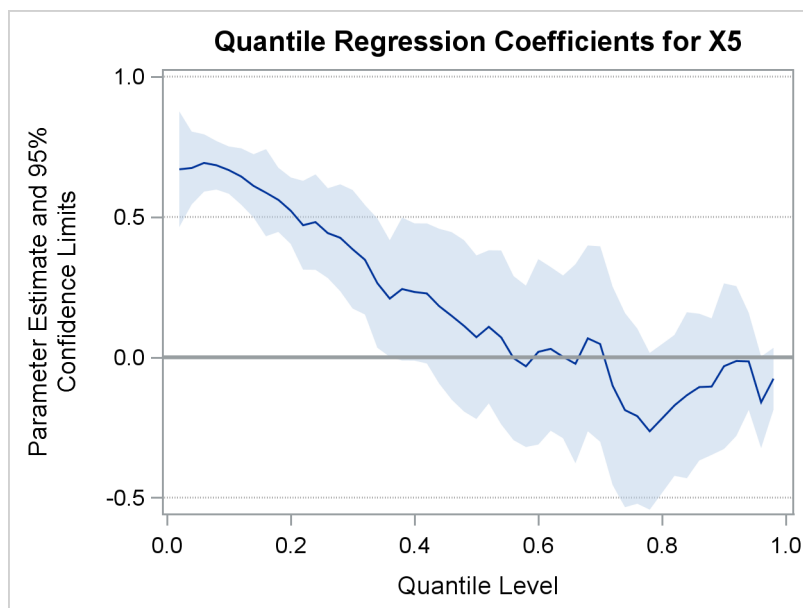
```
%MACRO ProcessPlot(Parm=);
data ParmEst; set Estimates;
  if Parameter EQ "&Parm";
run;

title "Quantile Regression Coefficients for &Parm";
proc sgplot data=ParmEst noautolegend;
  band x=quantile lower=LowerCL upper=UpperCL / transparency=0.5;
  series x=quantile y=estimate ;
  refline 0 / axis=y lineattrs=(thickness=2px);
  yaxis label='Parameter Estimate and 95% Confidence Limits'
  grid gridattrs=(thickness=1px color=gray pattern=dot);
  xaxis label='Quantile Level';
run;
%MEND ProcessPlot;

%ProcessPlot (Parm=X5)
```

The quantile process plot, shown in [Figure 7](#), displays the parameter estimates and 95% confidence limits as a function of quantile level. The plot reveals that **X5** positively affects the lower tail of the distribution of **CLV**, because the lower confidence limits are greater than 0 for quantile levels less than 0.37.

Figure 7 Quantile Process Plot for X5

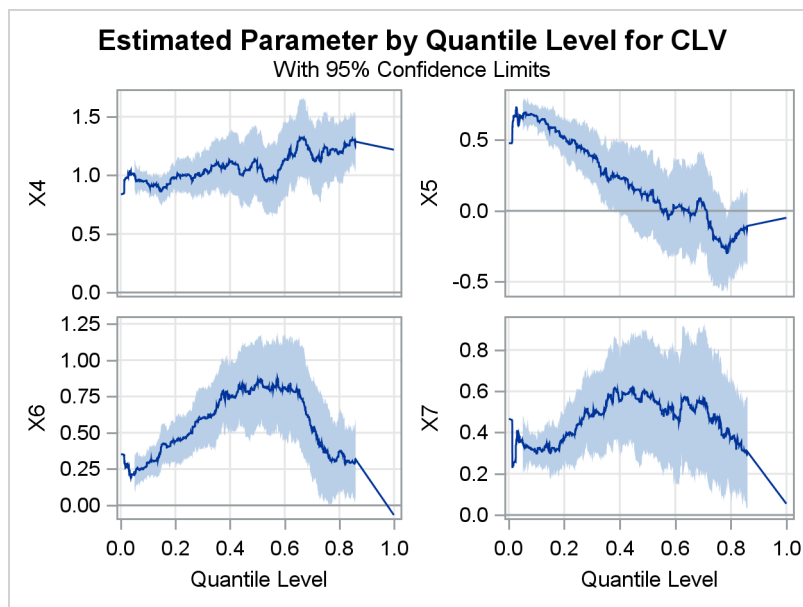


A drawback of specifying an explicit grid for quantile process regression is that the grid resolution might not be optimal for the data. As an alternative, you can search for the optimal grid, which depends on the data, by specifying the QUANTILE=PROCESS option in the MODEL statement. The optimal grid is usually not evenly spaced. The following statements illustrate the option:

```
proc quantreg data=CLV ci=sparsity ;
  model CLV = x1-x15 / quantile=process plot=quantplot;
run;
```

The PLOT=QUANTPLOT option requests paneled displays of quantile process plots for the intercept term and all the predictors. Figure 8 shows the second of the four displays that are produced, which includes the plot for **X5**.

Figure 8 Quantile Process Plots (Panel 2)



The plot for **X5** in Figure 7 is a linearly interpolated low-resolution counterpart of the optimal plot for **X5** in Figure 8. However, computing this low-resolution counterpart is much more efficient than computing the optimal one.

Paneled quantile process plots help you to readily identify which predictors are associated with different parts of the response distribution.

Building Quantile Regression Models

One of the most frequently asked questions in the framework of standard regression is this: “I have hundreds of variables—even thousands. Which should I include in my model?” The same question arises in the framework of quantile regression.

For standard regression, the flagship SAS/STAT procedure for model building is the GLMSELECT procedure. This procedure selects effects in general linear models of the form

$$y_i = \beta_0 + \beta_1 x_{i1} + \cdots + \beta_p x_{ip} + \epsilon_i, \quad i = 1, \dots, n$$

where the response y_i is continuous. The predictors x_{i1}, \dots, x_{ip} represent main effects that consist of continuous or classification variables and their interactions or constructed effects.

The QUANTSELECT procedure performs effect selection for quantile regression. Like the GLMSELECT procedure, it is designed primarily for effect selection, and it does not include regression diagnostics or hypothesis testing, which are available in the QUANTREG procedure.

If you have too many predictors, the model can overfit the training data, leading to poor prediction when you apply the model to future data. To deal with this problem, the QUANTSELECT procedure supports a variety of model selection methods, including the lasso method; these are summarized in Table 2.

Table 2 Effect Selection Methods in the QUANTSELECT Procedure

Method	Description
Forward selection	Starts with no effects and adds effects
Backward elimination	Starts with all effects and deletes effects
Stepwise selection	Starts with no effects; effects are added and can be deleted
Lasso	Adds and deletes effects based on a constrained version of check loss where the ℓ_1 norm of the β s is penalized
Adaptive lasso	Constrains sum of absolute weighted β s; some β s set to 0

The QUANTSELECT procedure offers extensive capabilities for customizing model selection by using a wide variety of selection and stopping criteria, including significance-level-based criteria and information criteria. The procedure also enables you to use validation-based criteria by partitioning the data into subsets for training, validation, and testing.

The following example illustrates the use of the QUANTSELECT procedure.

Example: Predicting the Close Rates of Retail Stores

The close rate of a retail store is the percentage of shoppers who enter the store and make a purchase. Understanding what factors predict close rate is critical to the profitability and growth of large retail companies, and a regression model is constructed to study this question.

The close rates of 500 stores are saved in a data set named **Stores**. Each observation provides information about a store. The variables available for the model are the response **Close_Rate** and the following candidate predictors:

- **X1**, ..., **X20**, which measure 20 general characteristics of stores, such as floor size and number of employees
- **P1**, ..., **P6**, which measure six promotional activities, such as advertising and sales
- **L1**, ..., **L6**, which measure special layouts of items in six departments

In practice, close rate data can involve hundreds of candidate predictors. A small set is used here for illustration.

By building a standard regression model, you can answer questions such as the following:

How can I predict the close rate of a new store?

Which variables explain the average close rate of a store?

By building a quantile regression model, you can answer a different set of questions:

How can I predict a high close rate, such as the 90th percentile of the close rate distribution?

Which variables explain a low close rate, such as the 10th percentile of the close rate distribution?

Are there variables that differentiate between low and high close rates?

The following statements use the QUANTSELECT procedure to build quantile regression models for levels 0.1, 0.5, and 0.9:

```
proc quantselect data=Stores plots=Coefficients seed=15531;
  model Close_Rate = X1-X20 L1-L6 P1-P6 / quantile = 0.1 0.5 0.9
                                selection=lasso(sh=3);
  partition fraction(validate=0.3);
run;
```

The SELECTION= option specifies the lasso method with a stop horizon of 3. The PARTITION statement reserves 30% of the data for validation, leaving the remaining 70% for training.

Figure 9 summarizes the effect selection process for quantile level 0.1. The lasso method generates a sequence of candidate models, and the process chooses the model that minimizes the average check loss (ACL) computed from the validation data. The process stops at Step 14.

Figure 9 Selection Summary for Quantile Level 0.1

The QUANTSELECT Procedure
Quantile Level = 0.1

Selection Summary				
Step	Effect Entered	Effect Removed	Number Effects In	Validation ACL
0	Intercept		1	0.1578
1	X2		2	0.1667
2	X4		3	0.1566
3	P3		4	0.1380
4	P1		5	0.1326
5	P2		6	0.1119
6	P4		7	0.1104
7	X20		8	0.1113
8	X3		9	0.1111
9	P5		10	0.1096
10		P5	9	0.1111
11	P5		10	0.1096
12		X3	9	0.1083*
13	L1		10	0.1105
14	X3		11	0.1117

The coefficient progression plot in Figure 10 visualizes the selection process. The variables **X2** and **X4** are the first to enter the model.

Figure 10 Coefficient Progression for Quantile Level 0.1

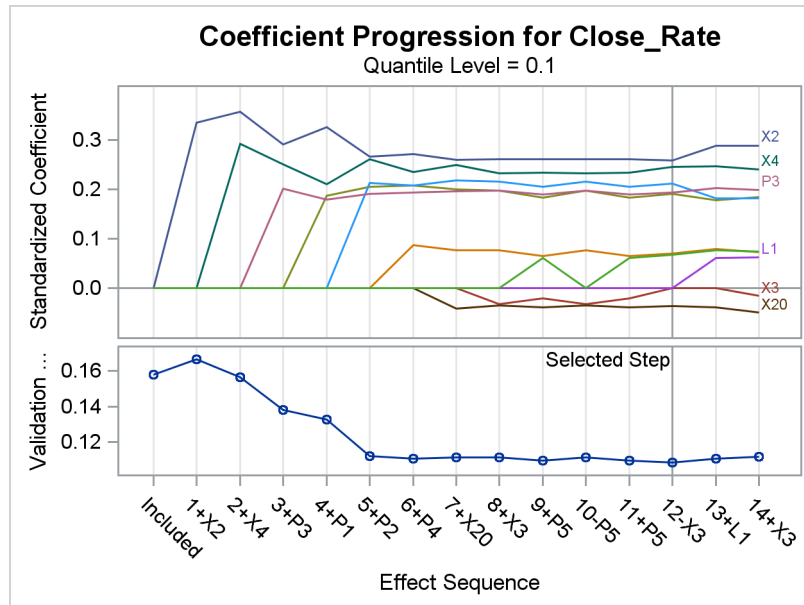


Figure 11 shows the fit statistics and parameter estimates for the final model for quantile level 0.1. The QUANTSELECT procedure produces parallel but distinct sets of results for quantile levels 0.5 and 0.9.

Figure 11 Fit Statistics and Parameter Estimates for Model Selected for Quantile Level 0.1

The QUANTSELECT Procedure
Quantile Level = 0.1

Fit Statistics			
Objective Function		36.17929	
R1		0.38327	
Adj R1		0.36909	
AIC		-1616.52369	
AICC		-1616.00496	
SBC		-1581.62407	
ACL (Train)		0.10134	
ACL (Validate)		0.10826	

Parameter Estimates			
Parameter	DF	Estimate	Standardized Estimate
Intercept	1	60.097618	0
X2	1	0.953402	0.258498
X4	1	0.933705	0.245902
X20	1	-0.140895	-0.035981
P1	1	0.724145	0.190798
P2	1	0.783880	0.211752
P3	1	0.696274	0.193163
P4	1	0.260641	0.069442
P5	1	0.242147	0.067135

Figure 12 and Figure 13 show the parameter estimates for the final models for quantile levels 0.5 and 0.9.

Figure 12 Parameter Estimates for Model Selected for Quantile Level 0.5

Parameter Estimates			
Parameter	DF	Estimate	Standardized Estimate
Intercept	1	60.950579	0
X2	1	1.508595	0.409029
X4	1	0.710687	0.187168
P3	1	0.361047	0.100163
P4	1	0.669943	0.178491
P5	1	0.544278	0.150902

Figure 13 Parameter Estimates for Model Selected for Quantile Level 0.9

Parameter Estimates			
Parameter	DF	Estimate	Standardized Estimate
Intercept	1	61.079231	0
X2	1	0.982776	0.266463
X4	1	1.118507	0.294572
L2	1	1.027725	0.297930
L3	1	0.859988	0.240257
L5	1	0.672210	0.186588
P5	1	0.192967	0.053500

A sparse model that contains only six variables (**X2**, **X4**, **L2**, **L3**, **L5**, and **P5**) is selected as the best model for predicting the 90th percentile. The layout variables **L2**, **L3**, and **L5** are in this model, but not in the models for the 10th and 50th percentiles. The variables **X2** and **X4** are common to all three models. These results give you insights about store performance that you would not obtain directly from standard regression methods.

Applying Quantile Regression to Financial Risk Management

Although quantile regression can model the entire conditional distribution of the response, it often leads to deep insights and valuable solutions in situations where the most useful information lies in the tails. This is demonstrated by the application of quantile regression to the estimation of value at risk (VaR).

Financial institutions and their regulators use VaR as the standard measure of market risk. The quantity VaR measures market risk by how much a portfolio can lose within a given time period, with a specified confidence level $(1 - \tau)$, where τ is often set to 0.01 or 0.05. More precisely, the value at risk at time t (denoted by VaR_t) is the conditional quantile of future portfolio values that satisfies the equation

$$\Pr[y_t < -\text{VaR}_t] = \tau, \quad 0 < \tau < 1$$

where $\{y_t\}$ is the series of asset returns and Ω_τ , the information available at time t , includes covariates and values of past asset returns.

Commonly used methods of estimating VaR include copula models, ARCH models, and GARCH models (GARCH stands for generalized autoregressive conditional homoscedasticity). SAS/ETS[®] software provides a number of procedures for fitting these models; see the *SAS/ETS 14.2 User's Guide*.

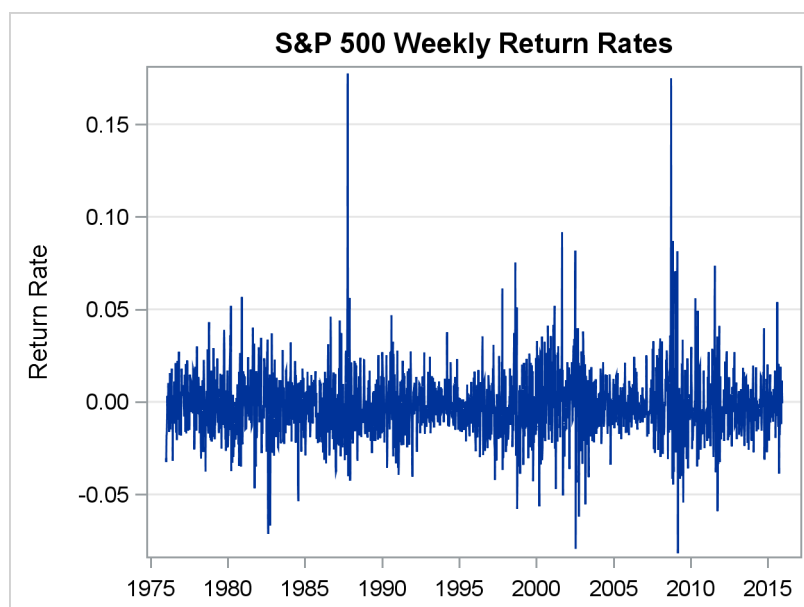
ARCH and GARCH models assume that financial returns are normally distributed. However, as pointed out by Xiao, Guo, and Lam (2015, p. 1144), the distributions of financial time series and market returns often display skewness and heavy tails. Extreme values of returns can bias estimates of VaR that are produced using ARCH and GARCH models.

Autoregressive quantile regression provides a robust alternative for estimating VaR that does not assume normality (Koenker and Zhao 1996; Koenker and Xiao 2006; Xiao and Koenker 2009). This is illustrated by the next example, which is patterned after the analysis of equity market indexes by Xiao, Guo, and Lam (2015, pp. 1159–1166).

Example: Computing Value at Risk for S&P 500 Return Rates

Figure 14 displays weekly return rates of the S&P 500 Composite Index.

Figure 14 Weekly Return Rates of the S&P 500 Index



The following statements compute predicted 0.05 quantiles for the weekly return rate by fitting a standard GARCH(1,1) model, which assumes that the rate is normally distributed:

```
%let VaRQtLevel=0.05; /* 95% confidence */

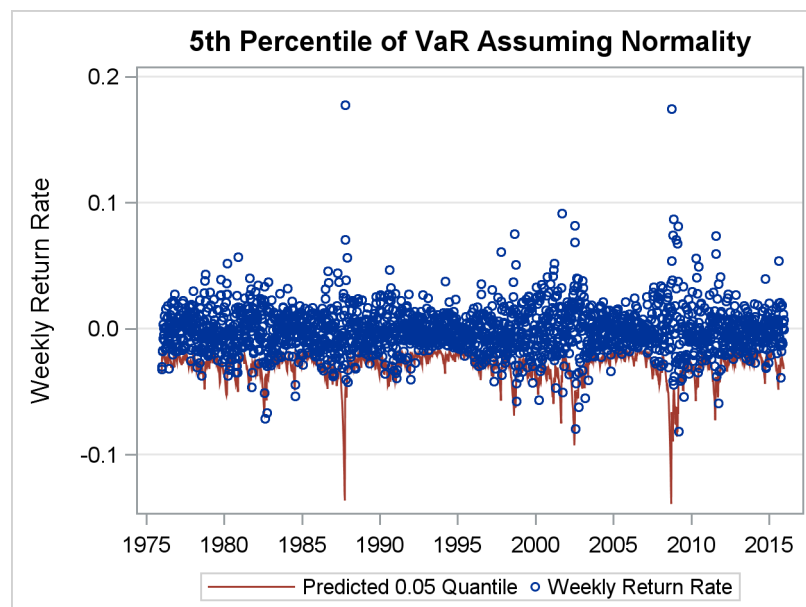
proc varmax data=SP500;
  model ReturnRate;
  garch form=ccc subform=garch p=1 q=1;
  output out=g11 lead=1;
  id date interval=week;
run;

data g11;
  set g11;
  qt=for1 + std1*quantile('normal',&VaRQtLevel);
run;

title "%sysevalf(&VaRQtLevel*100)th Percentile of VaR Assuming Normality";
proc sgplot data=g11;
  series y=qt          x=date / lineattrs=graphdata2(thickness=1);
  scatter y=ReturnRate x=date / markerattrs=(size=5);
  yaxis grid;
  xaxis display=(nolabel) type=linear %tick offsetmax=0.05 ;
  label ReturnRate = "Weekly Return Rate"
         qt        = "Predicted &VaRQtLevel Quantile";
run;
```

The results are plotted in Figure 15. The proportion of observed return rates that are less than the predicted quantiles (highlighted in red) is less than 0.05, because the model assumes that the rate distribution is symmetric when it is actually skewed in the high direction. Therefore, the predicted 0.05 quantile based on this model overestimates the risk.

Figure 15 Analysis Based on GARCH and Normal Quantile Regression Models



The robustness of quantile regression makes it an attractive alternative for modeling the heavy-tailed behavior of portfolio returns. Xiao, Guo, and Lam (2015, p. 1161) discuss an approach that uses an AR(1)–ARCH(7) quantile regression model for the return rate at time t .

The following statements implement a similar approach in two steps, the first of which fits an AR(1)–ARCH(7) model by using the VARMAX procedure in SAS/ETS software:

```
proc varmax data=SP500;
    model ReturnRate / p=1;
    garch form=ccc subform=garch q=6;
    output out=ala7 lead=1;
    id date interval=week;
run;
```

The MODEL statement specifies an AR(1) (autoregressive order one) model for the mean,

$$r_t = \alpha_0 + \alpha_1 r_{t-1} + u_t$$

where $u_t = \sigma_t \epsilon_t$. The GARCH statement specifies the ARCH(7) component:

$$\sigma_t = \gamma_0 + \gamma_1 |u_{t-1}| + \cdots + \gamma_6 |u_{t-6}|$$

No parametric distribution is assumed for ϵ_t . The VARMAX procedure creates an output data set named **A1A7** that saves the standard error of prediction in the variable **STD1**.

The second step fits a quantile regression model for level τ of VaR_t , which conditions on lagged values of the standard error that was estimated by PROC VARMAX:

```
data ala7;
    set ala7;

    /* Lagged predictors for quantile regression */
    STD2=lag1(std1);
    STD3=lag2(std1);
    STD4=lag3(std1);
    STD5=lag4(std1);
    STD6=lag5(std1);
    STD7=lag6(std1);
run;

proc quantreg data=ala7 ci=none;
    model ReturnRate = std1-std7 / quantile=&VarQtlLevel;
    output out=qr p=p;
    id date;
    label ReturnRate = "Return Rate";
run;

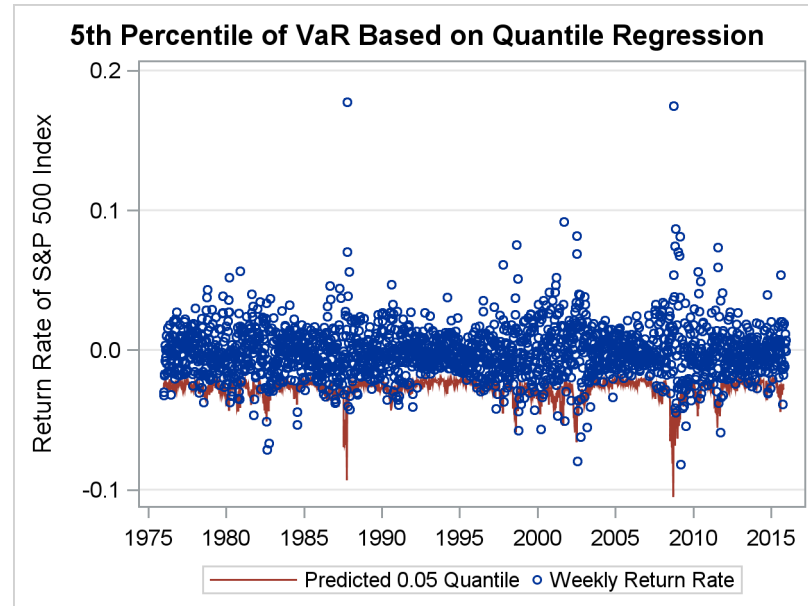
title "%sysevalf(&VarQtlLevel*100)th Percentile of VaR Based on Quantile Regression";
proc sgplot data=qr;
    series y=p x=date / lineattrs=graphdata2(thickness=1);
    scatter y=ReturnRate x=date / markerattrs=(size=5);
    yaxis label="Return Rate of S&P 500 Index" grid;
    xaxis display=(nolabel) type=linear %tick offsetmax=0.05 ;
    label p ="Predicted &VarQtlLevel Quantile";
    label ReturnRate="Weekly Return Rate";
run;
```

The form of the model is

$$Q_\tau(\text{VaR}_t) = \gamma_0(\tau) + \gamma_1(\tau)|u_{t-1}| + \cdots + \gamma_6(\tau)|u_{t-6}|$$

The QUANTREG procedure computes the predicted 0.05 quantiles of the return rates on the AR(1)–ARCH(7) variance predictions. This guarantees that precisely 5% of the observed return rates lie below the predicted 0.05 quantiles of VaR_t , which are plotted in [Figure 16](#).

Figure 16 Analysis Based on Quantile Regression AR(1)–ARCH(7) Model



Applying Quantile Process Regression to Ranking Exam Performance

In the applications of quantile regression that have been discussed so far in this paper, the goal has been to predict conditional quantiles for specified quantile levels. However, in many applications—such as ranking the performance of students on exams—the goal is to predict conditional quantile *levels* for specified observations. You can use quantile process regression for this purpose because it predicts the entire conditional distribution of the response, and quantile levels are simply probabilities that can be computed from this distribution.

Consider a student named Mary who scored 1948 points on a college entrance exam. You cannot rank her performance unless you know the distribution of scores for all students who took the exam. Mary, her parents, and her teachers are primarily interested in her quantile level, which is 0.9. This informs them that she performed better than 90% of the students who took the exam.

Mathematically, if Y denotes the score for a randomly selected student who took the exam, and if $F(y)$ denotes the cumulative distribution function (CDF) of Y , then the CDF determines the quantile level for any observed value of Y . In particular, Mary's quantile level is $F(1948) = \Pr[Y \leq 1948] = 0.9$.

In practice, the quantile levels of a response variable Y must often be adjusted for the effects of covariates X_1, \dots, X_p . This requires that the quantile levels be computed from the conditional distribution $F(y \mid X_1 = x_1, \dots, X_p = x_p)$.

To see why such an adjustment makes a difference, consider a second student named Michael, who took the exam and scored 1617 points. Michael's quantile level is $F(1617) = 0.5$, so you might conclude that Mary performed better than Michael. However, if you learn that Mary is 17 and Michael is 12, then the question becomes, How did Mary and Michael perform relative to the other students in their age groups? The answer is given by their respective conditional quantile levels, which are $F(1948 \mid \text{Age} = 17)$ and $F(1617 \mid \text{Age} = 12)$.

With sufficient data, quantile process regression gives you a flexible method of obtaining adjusted quantile levels that does not require you to assume a parametric form for the conditional distribution of the response. The following example illustrates the computations.

Example: Ranking Exam Scores

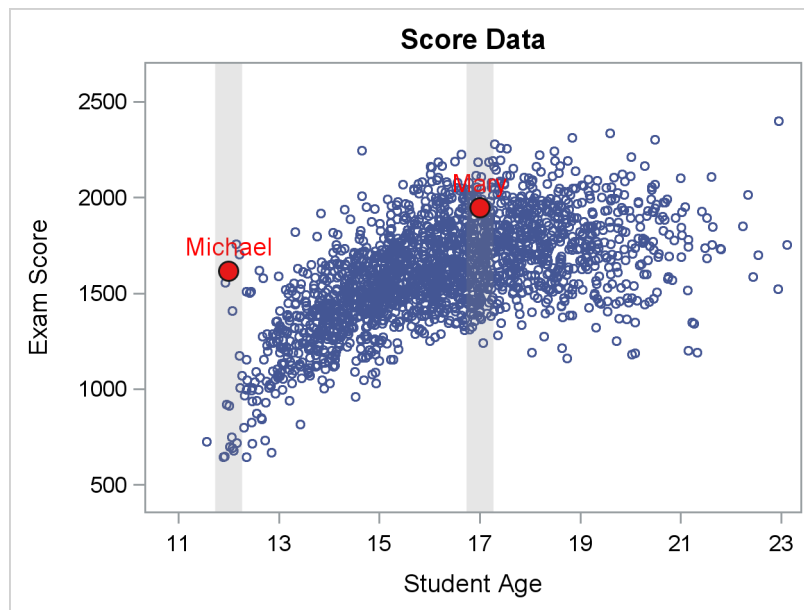
A SAS data set named **Score** contains three variables, **Name**, **Age**, and **Score**, which provide the names, ages, and scores of the 2,000 students who took the exam, including Mary and Michael. Figure 17 lists the first five observations.

Figure 17 Partial Listing of **Score**

Obs	Name	Age	Score
1	Michael	12.0	1617
2	Mary	17.0	1948
3	Yonggang	15.3	1661
4	Bob	15.3	1517
5	Youngjin	13.1	1305

The scatter plot in [Figure 18](#) highlights the observations for Mary and Michael. Note that the distribution for 12-year-olds is different from the distribution for 17-year-olds. For a fair comparison, the quantile levels for Mary and Michael should be adjusted for the effect of age.

Figure 18 Exam Score versus Age



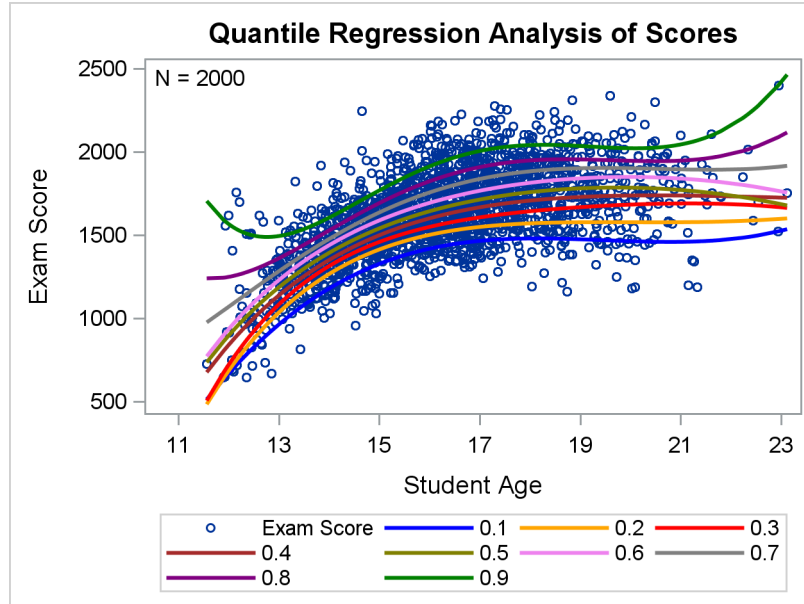
The first step in making this comparison is to fit a model that adequately describes the conditional score distribution. To account for the nonlinearity in the data, the following statements fit a quantile regression model that has four predictors, three of which are derived from **Age**. To examine the fit, it suffices to specify nine equally spaced quantile levels in the MODEL statement for PROC QUANTREG.

```
data Score;
  set Score;
  Age2  = Age*Age;
  Age3  = Age2*Age;
  AgeInv = 1/Age;
  label Score = "Exam Score"
         Age  = "Student Age";
run;

proc quantreg data=Score;
  model Score = Age Age2 Age3 AgeInv / quantile = 0.10 to 0.90 by 0.1;
  output out=ModelFit p=Predicted;
  label Score = "Exam Score"
         Age  = "Student Age";
run;
```

The fit plot in [Figure 19](#) shows that the model adequately captures the nonlinearity.

Figure 19 Conditional Quantile Regression Models for Exam Scores



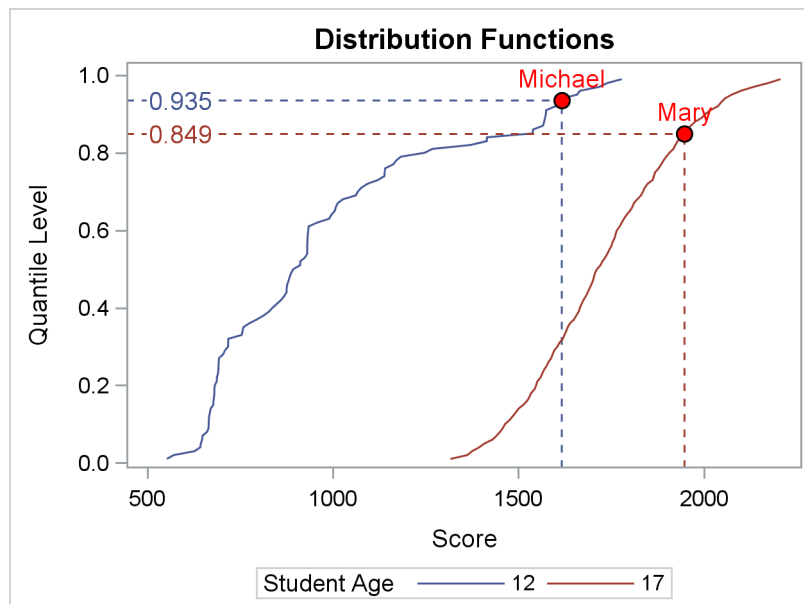
In the next statements, the model variables serve as input to the QPRFIT macro, which refits the model for an extensive grid of quantile levels ($\tau = 0.01, 0.02, \dots, 0.99$). The macro then forms sets of predicted quantiles that condition on the values of **Age** for Mary and Michael, whose observations are identified by **Name** in the IDDATA= data set. From each set, the macro constructs a conditional CDF, which is used to compute the adjusted quantile levels.

```
data ScoreID;
    Name='Mary';    output;
    Name='Michael'; output;
run;

%qprFit(data=Score, depvar=Score, indvar=Age Age2 Age3 AgeInv, onevar=Age,
        nodes=99, iddata=ScoreID, showPDFs=1, showdist=1)
```

The INDVAR= option specifies the predictors **Age**, **Age2**, **Age3**, and **AgeInv**. The ONEVAR= option indicates that the last three predictors are derived from **Age**. As shown in Figure 20, the macro plots the CDFs for 12-year-old and 17-year-old students.

Figure 20 Conditional Distribution Functions of Scores for Ages 12 and 17



The drop lines indicate the scores and quantile levels for Mary and Michael. The macro also produces the table shown in [Figure 21](#), which summarizes the results.

Figure 21 Regression-Adjusted and Univariate Quantile Levels for Mary and Michael

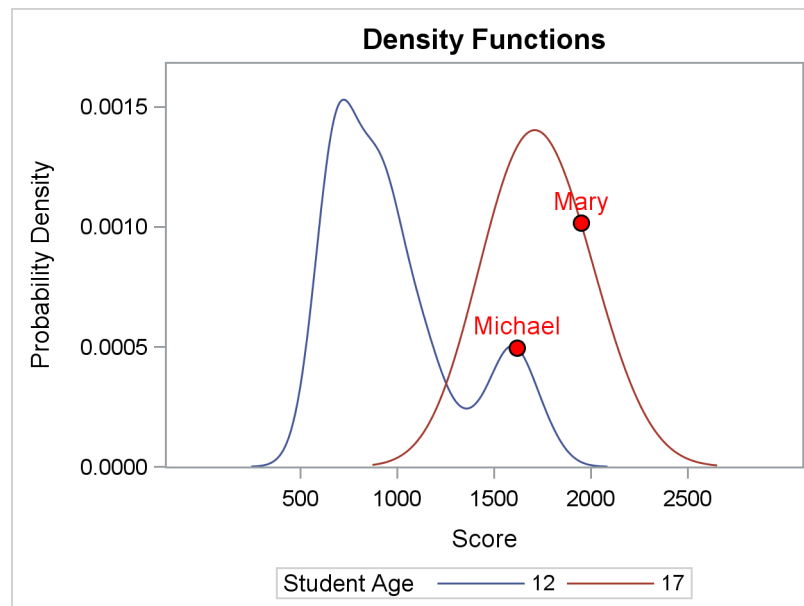
Statistics for the Highlighted Observations

Obs	Name	Score	Age	Mean	Median	Regression Quantile Level	Sample Quantile Level
1	Michael	1617	12	971.43	893.45	0.93500	0.50075
2	Mary	1948	17	1709.94	1712.36	0.84851	0.90025

Based on the regression-adjusted quantile levels, Michael is at the 93.50 percentile for 12-year-olds, and Mary is at the 84.85 percentile for 17-year-olds.

The SHOWPDFS=1 option requests the density estimates shown in [Figure 22](#).

Figure 22 Conditional Density Functions of Exam Scores for Ages 12 and 17



The Appendix explains the QPRFIT macro in more detail.

Summary

This paper makes five key points:

1. Quantile regression is a highly versatile statistical modeling approach because it uses a general linear model to fit conditional quantiles of the response without assuming a parametric distribution.
2. Quantile process regression estimates the entire conditional distribution of the response, and it allows the shape of the distribution to depend on the predictors.
3. Quantile process plots reveal the effects of predictors on different parts of the response distribution.
4. Quantile regression can predict the quantile levels of observations while adjusting for the effects of covariates.
5. The QUANTREG and QUANTSELECT procedures give you powerful tools for fitting and building quantile regression models, making them feasible for applications with large data.

Note that SAS/STAT software also provides the QUANTLIFE procedure, which fits quantile regression models for censored data, and the HPQUANTSELECT procedure, a high-performance procedure for fitting and building quantile regression models that runs in either single-machine mode or distributed mode (the latter requires SAS® High-Performance Statistics). SAS® Viya™ provides the QTRSELECT procedure, which fits and builds quantile regression models.

Appendix: The QPRFIT Macro

The QPRFIT macro fits a quantile process regression model and performs conditional distribution analysis for a subset of specified observations. The macro is available in the SAS autocall library starting with SAS® 9.4M4, and it requires SAS/STAT and SAS/IML® software. You invoke the macro as follows:

```

%macro qprFit(
    data=_last_,
    depvar=,
    indvar=,
    onevar=,
    nodes=19,
    peData=qprPE,
    iddata=,
    showPDFs=0,
    showdist=1,
);
    /*-----*/
    /* Quantile regression specialized output.      */
    /*-----*/
    /* Input data set.                               */
    /* Dependent or response variable.                */
    /* Independent or explanatory variables.          */
    /* 1, y, Y, t, T - show fit and scatter plots,   */
    /* which are appropriate for a single independent */
    /* variable. (Only the first character is checked.) */
    /* Other nonblank - do not show fit plot.        */
    /* By default, ONEVAR is true when there is a    */
    /* single independent variable and false otherwise. */
    /* Set ONEVAR= to true when there are multiple  */
    /* independent variables but they form a polynomial */
    /* or other nonlinear function of a single      */
    /* variable. When ONEVAR is true, the first     */
    /* independent variable is used in the fit and   */
    /* scatter plots.                                */
    /* Quantile process step size is 1 / (1 + NODES). */
    /* The default step size is 0.05.               */
    /* Output parameter-estimates data set for the  */
    /* quantile process regression model.            */
    /* This data set is used in the qprPredict macro. */
    /* Data set with ID variable for the observations */
    /* to highlight. Only one variable is permitted in */
    /* the data set, and the same variable must be in  */
    /* the DATA= data set.                          */
    /* 1, y, Y, t, T - show probability density     */
    /* function plot.                                */
    /* Other nonblank - do not show density plots.   */
    /* 1, y, Y, t, T - show distribution functions plot. */
    /* Other nonblank - do not show this plot.      */
    /*-----*/

```

You specify the dependent variable by using the DEPVAR= option and the independent variables by using the INDVAR= option. You specify ONEVAR=0 if there are two or more independent variables. You specify ONEVAR=1 if there is a single independent variable or if the INDVAR= list includes variables that are derived from a single independent variable (see the example on page 14).

For data that contain a dependent variable Y and independent variables X_1, \dots, X_p , the QPRFIT macro uses the QUANTREG procedure to fit the conditional quantile regression model

$$Q_\tau(y_i|x_{i1}, \dots, x_{ip}) = \beta_0(\tau) + \beta_1(\tau)x_{i1} + \dots + \beta_p(\tau)x_{ip}, \quad i = 1, \dots, n$$

for t equally spaced quantile levels: $\tau_1 = 1/(t+1), \tau_2 = 2/(t+1), \dots, \tau_t = t/(t+1)$. You specify t by using the NODES= option. Estimates for $\beta(\tau_1), \dots, \beta(\tau_t)$ are saved in an output data set that you can name in the PEDATA= option. The default output data set is named **QPRPE**.

Let y_{i_1}, \dots, y_{i_m} denote the values of Y for a subset of m observations that you identify in the IDDATA= data set, and let $x_{i_1 1}, \dots, x_{i_m p}$ denote the corresponding covariate values. For observation i_j , the macro forms the set Q_{i_j} of predicted quantiles. These quantiles are sorted and used to construct a conditional cumulative distribution function (CDF) that corresponds to the covariate values $x_{i_1 1}, \dots, x_{i_m p}$. When you specify SHOWDIST=1, the macro plots the CDFs that correspond to the covariate values and the predicted quantile levels for the specified observations, which it computes from the CDFs; see Figure 20 for an example. When you specify SHOWPDFS=1, the macro plots smooth density estimates that correspond to the covariate values; see Figure 22 for an example.

REFERENCES

- Hao, L., and Naiman, D. Q. (2007). *Quantile Regression*. London: Sage Publications.
- 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 Xiao, Z. (2006). "Quantile Autoregression." *Journal of the American Statistical Association* 101:980–1006.
- Koenker, R., and Zhao, Q. (1996). "Conditional Quantile Estimation and Inference for ARCH Models." *Econometric Theory* 12:793–813.
- Xiao, Z., Guo, H., and Lam, M. S. (2015). "Quantile Regression and Value at Risk." In *Handbook of Financial Econometrics and Statistics*, edited by C.-F. Lee and J. Lee, 1143–1167. New York: Springer.
- Xiao, Z., and Koenker, R. (2009). "Conditional Quantile Estimation for Generalized Autoregressive Conditional Heteroscedasticity Models." *Journal of the American Statistical Association* 104:1696–1712.

Acknowledgments

The authors thank Warren Kuhfeld for assistance with the QPRFIT macro and the graphical displays in this paper. The authors also thank Ed Huddleston for editorial assistance.

Contact Information

Your comments and questions are valued and encouraged. You can contact the authors at the following addresses:

Robert N. Rodriguez	Yonggang Yao
SAS Institute Inc.	SAS Institute Inc.
SAS Campus Drive	SAS Campus Drive
Cary, NC 27513	Cary, NC 27513
Bob.Rodriguez@sas.com	Yonggang.Yao@sas.com

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.