Paper SAS1673-2018

Getting More Insight into Your Forecast Errors with the GLMSELECT and QUANTSELECT Procedures

Gerhard Svolba, SAS Institute Inc. Austria

ABSTRACT

Is it sufficient just to monitor the quality of your forecast models over time? Can data science methods identify the drivers for large forecast errors and provide more insights than descriptive statistics? Do demand planners really improve forecast accuracy with their manual overwrites? Using a real-life case study, this paper answers these questions. It shows how you can study the impact of factors like product group, forecast horizons, seasonality, or the forecast model type on forecast accuracy and convert them into actionable results. You learn how univariate methods provide first insights into the structure and relationships of your forecast data. You gain insight into how manual overwrites of the statistical forecast change forecast accuracy in both directions and how you use analytical and graphical methods to illustrate these findings. You see how multivariate analytical methods like linear and quantile regression provide additional relevant insight. You learn how to use the GLMSELECT, QUANTSELECT, and QUANTREG procedures to identify the most important influential factors on the forecast error. You see how you can enhance and interpret the output of these procedures to quantify the effects of the influential factors. You learn how to convert the results from the SAS® procedures into actions to improve your forecasting process. The paper shows an outline of how to use the REGSELECT and QTRSELECT procedures to apply these methods in SAS® Viya®.

INTRODUCTION

APPLY ANALYTICAL METHODS ACROSS DIFFERENT BUSINESS DOMAINS

Analytical methods can leverage the analysis outcome for various business questions. Going one level deeper than simple descriptive methods provides insights in the relationship between influential variables. Analytical methods also help you spot multivariate relationships and enable you to receive an objective and data driven answer to your business questions.

The book *Applying Data Science: Business Case Studies Using SAS®* (Svolba 2017) is dedicated to the application of analytical methods to different types of practical questions. It shows how analytical methods that have been successfully used in certain business domains can and should be applied also to other business areas. For example, you can apply survival analysis techniques to analyze the retention time of employees, or you can use ARIMA methods and multivariate adaptive regression splines to automatically detect breakpoints in your time series data.

CASE STUDY: ANALYZING THE FORECAST ERROR

This paper deals with a case study from the demand forecasting area. The focus is to investigate the forecast error, which is measured as the deviation between the forecasted demand and the actual demand. It shows how analytical methods like regression analysis can be used to identify factors that have an impact on the magnitude of the forecast error.

The case study does not deal with the creation of the statistical forecast itself but with the evaluation of the forecast quality. Typical business questions in forecast quality are discussed and this paper shows how they can be solved with analytical methods, like descriptive analyses or general linear models.

USING REGRESSION ANALYSIS

The statistical tools that are shown here include boxplots, histograms, and descriptive measure like mean, median, and the quartiles, as well as linear regression and quantile regression methods.

The analysis provides insight about the drivers for different levels of forecast quality. It shows that general linear models are perfectly suited to answer business questions related to forecast quality.

- General linear models enable you to automatically select the most important variable for the analysis.
- They provide an answer about the importance of different influential factors.
- They express the mathematical relationship between forecast error and analysis variable.

STUDYING THE EFFECT OF MANUAL OVERRIDES

In the forecasting process, statistical forecasts are often overridden with judgmental forecasts by the forecaster or demand planner. The analysis shows whether the overall forecast quality is improved with manual overrides. A detailed analysis of the characteristics of the manual overrides shows their effect on the forecast quality.

The case study also explains the main assumptions and deliverables of regression analyses and illustrates the main features with results from the business questions.

BUSINESS QUESTIONS FOR THE ANALYSIS

From a business point of the view, the following questions are of interest and are analyzed in this case study.

- What is the distribution of the forecast error over all products?
    - What is the average forecast error?
    - What is the forecast error that is not exceeded by the top 25%, 50%, and 75% of the forecasts?
- Which factors influence the forecast error?
    - Is the forecast error different between product groups or price categories?
    - Does the launch month or the age of the product influence the forecast error?
    - Do different forecast models generate different forecast quality?
    - Do forecasts get better if the target months get closer?
    - Is there a difference in forecast quality between the calendar months or between years?
- How do the manual forecasts compare to statistical forecasts?
    - What is the average improvement of applying judgmental corrections?
    - Are there areas where judgmental corrections have a larger benefit?
    - Are there cases where judgmental corrections decrease forecast quality?
- Are there trends over time in the forecast errors that can be detected?

BUSINESS BACKGROUND OF THE CASE STUDY

NEED FOR DIFFERENT TYPES OF FORECASTS AND ARTICLE SEGMENTATION

In this case study the business department is the operational planning department of an international retail and manufacturing company. For their sales and demand planning, demand forecasts on a monthly basis are needed. These forecasts are generated automatically with analytical models in SAS® Forecast Server and SAS® Enterprise Miner™.

Some of the articles that are sold by the company have been in the assortment already for some years, and other articles remain in the product offering only for a limited period of time, like 6 or 12 months. With part of its product range, the company operates in the fashion business. Here, articles are retired when a new collection comes on the market or when articles do not sell as expected.

Note that in some industries, the term SKU is often used instead of “articles”. SKU is the abbreviation for Stock Keeping Unit. In this case study, the term “article” is used.

Article Segmentation

For the forecasting process, the articles are segmented into LONG and SHORT articles, based their available data history.

- LONG articles have a history of 15 months or more and are forecast with time series forecasting methods like exponential smoothing and ARIMA models.
- SHORT articles have a history up to 14 months and are forecast with a predictive model based on attributes of the product itself.

The future forecast horizon for which forecasts shall be created ranges from 4 to 18 months. These forecasts are used for different purposes:

- The rolling monthly forecasts for sales and demand planning are created for 4-6 months in the future.
- Forecasts of up to 18 months are used for budget planning for the next business year.

Target Months and Create Month

A forecast for a particular month, TARGET_MONTH, is usually created in more than one period of time (CREATE_MONTH). The forecast for the target month July might be created in the create months February, March, April, May, and June. Table 1 illustrates this case.

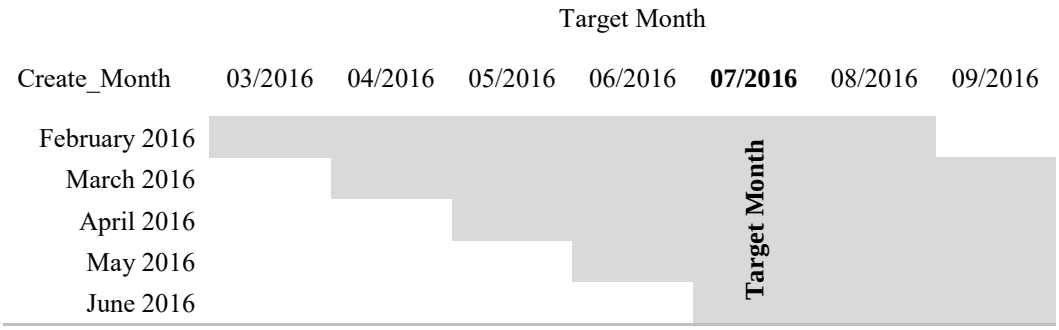


Table 1. Target Month and Create Month

For a particular target month, forecasts from different create months are available in the data. These records have the same TARGET_MONTH but different values of the CREATE_MONTH.

ArticleNum	CreateMonth	TargetMonth	LeadTime	Forecast
15942	2016.02	2016.07	5	1609
15942	2016.03	2016.07	4	1555
15942	2016.04	2016.07	3	1635
15942	2016.05	2016.07	2	1578
15942	2016.06	2016.07	1	1571

Output 1. Forecasts from Different Create Months

Forecasts can also be differentiated by the LEADTIME, which is the interval in the future for which they are created. The lead time of a forecast for July that is created in February, is 5.

MEASURING THE FORECAST ERROR OF STATISTICAL AND JUDGMENTAL FORECAST

Statistical and Judgmental Forecast

The forecasts that are created by analytical methods are called statistical forecasts. For articles in the segment LONG, three different model types are available to create the statistical forecast; for articles in

the segment SHORT, two different models are available. For each article, one of these models is defined as the champion model, and the other models are treated as challenger models for model quality performance monitoring over time.

The forecast can either directly be used for the demand planning, or it can be overwritten by the demand planner and is then called a judgmental forecast.

Usually, the demand planners use information about short-term market trends, promotion activities, or other factors that are not considered in the analytical model for the judgmental forecast.

There is a lot of discussion in forecasting practice about whether manual overrides truly improve forecast quality. See also Goodwin 2009 or Gilliland 2010 for this topic.

The data that are shown here are based on real world data. However, they have been amended for data privacy reasons.

Variety of Forecast Measures

There are many different measures to quantify the forecast error. Depending on the industry, the nature of the forecasting problem, and personal preferences, different methods are applied. The methods range in complexity of the calculation method, and some of them are combinations of other basic measures. In the forecasting community, there is no general agreed-upon “best measure” for the forecast error. Gilliland, 2010, for example, discusses the forecast value added (FVA). The FVA is the added value of the forecast in accuracy, compared to a naïve or baseline forecast.

Using the MAPE

In the example shown here, the forecast error is measured with the MAPE, the mean absolute percentage error. There are many critics for using the MAPE.

- The MAPE is asymmetric; a perfect fit results in a MAPE of 0. However, there is no restriction to the upper limit.
- For an observed demand of 0, the MAPE formula causes a division by zero.
- A forecast of 0 leads to a MAPE of 100. Thus, a forecasting model could learn this feature and limit its forecast error by forecasting 0 for all time points.

The advantage of the MAPE, however, is its interpretability, and it is thus very broadly used in business forecasting. The MAPE is calculated with the following steps:

1. Calculate the absolute value of the difference between the forecasted value and the actual value (that’s where the A in MAPE comes from).
2. Convert the absolute difference into a relative difference by dividing it by the actual value. This expresses the forecast error as a percentage of the actual value (that is, the origin of the P and the E in the abbreviation MAPE).
3. Finally, you average these absolute percentage errors over all available time points and receive a Mean Absolute Percentage Error (this is where the M comes from).

Calculating the APE for the Analysis

For the task of analyzing the forecast error per month, only the APE and not the MAPE is calculated. This means that the last step of averaging the forecast errors per article is not performed.

In forecast error analysis, you want to see the deviation for each individual point in time. This provides more detailed insight and also allows analyzing potential seasonal effects in the forecast error. It also provides insight into the change of forecast quality from different forecast create months for a particular target month.

For this case study, the forecast error is the absolute percentage error between the statistical forecast and the actual demand. The abbreviation APE_STAT is used for it.

AVAILABLE DATA AND DATA PREPARATION

OVERVIEW OVER THE AVAILABLE DATA SOURCES

The data table for the analysis is built based on three tables:

Name	Description	Primary Key Columns
STATFC	Contains the statistical forecast. This table is filled from the statistical forecasting process, and it is also used as basis for the judgmental forecasting by the demand planners.	ID, CREATE_MONTH, TARGET_MONTH
MANFC	Contains the forecasts that are finally committed by the demand planner.	ID, CREATE_MONTH, TARGET_MONTH
MATERIAL	Contains the product base data.	ID

Table 2. Data Sources for the Analysis Data Mart

The full process of data preparation and more details about the available data is explained in Svolba 2017, Chapter 10.

FC_ID	ID	Target_Month	Create_Month	Model	Product_Gro...	Price_Index	Launch_Mon...	Target_CalM...	Target_Year	Lead_Time	Product_Age
1	3335539	2009.12	2009.10	LONG S5 Dow...	8	180	1	12	2009	2	120
2	3335539	2009.12	2009.11	LONG S5 Dow...	8	180	1	12	2009	1	120
3	3335539	2010.01	2009.10	LONG S5 Dow...	8	180	1	1	2010	3	120
4	3335539	2010.01	2009.11	LONG S5 Dow...	8	180	1	1	2010	2	120
5	3335539	2010.01	2009.12	LONG S5 Dow...	8	180	1	1	2010	1	120
6	3335539	2010.02	2009.10	LONG S5 Dow...	8	180	1	2	2010	4	120
7	3335539	2010.02	2009.11	LONG S5 Dow...	8	180	1	2	2010	3	120
8	3335539	2010.02	2009.12	LONG S5 Dow...	8	180	1	2	2010	2	120
9	3335539	2010.02	2010.01	LONG S5 Dow...	8	180	1	2	2010	1	120
10	3335539	2010.03	2009.11	LONG S5 Dow...	8	180	1	3	2010	4	120

Output 2. Important Input Variables for the Analysis

Table 3 lists these variables with a short description and the measurement type. The measurement type determines the type of descriptive analysis and graph that can be used and defines how this variable is treated in the regression analysis.

Variable Name	Description	Measurement Type
PRODUCT_GROUP	Product group	Category
PRICE_INDEX	Price index	Interval
LAUNCH_MONTH	Calendar month of product launch	Category
PRODUCT_AGE	Number of months since the article was launched	Interval
MODEL	Model that was used for statistical forecasting	Category
LEAD_TIME	Number of months in the future for which the forecast is created	Interval
TARGET_CALMONTH	Calendar month for which the forecast is created	Category
TARGET_YEAR	Year for which the forecast is created	Interval

Table 3. Important Input Variables for the Analysis

CALCULATING DERIVED VARIABLES

The creation of selected derived variables is illustrated in this section. You see selected code lines from a larger DATA step that prepares the data.

Product and Forecast Process-Related variables

Derived variables from the date variables are created with the YEAR function and the MONTH function.

```
/*** FC-derived variables ***/
Create_CalMonth = month(create_month);
Create_Year     = year(create_month);
Target_CalMonth = month(target_month);
Target_Year     = year(target_month);
```

The LEAD_TIME and the PRODUCT_AGE are calculated as the difference between the two respective date values using the INTCK function. The INTCK function is very convenient to calculate the number of intervals (in this case, months) that lie between two date variables. Compare also Svolba 2006.

```
/*** Lead Times ***/
 Lead_Time = intck('MONTH',create_month,target_month);
 Product_Age = intck('MONTH',launch_date,target_month);
 if Product_Age > 120 then Product_Age=120;
```

Calculating Forecast Error

You create the average percentage error, APE, by calculating the absolute difference between the forecast value and the observed value. Divide this value through the observed value to receive a percentage error.

```
/*** MAPE-Block ***/
format APE_Stat APE_Man APE_Stat_Shift APE_Man_Shift 8.1;
APE_Stat = abs(statfc - actual)/actual * 100;
APE_Man  = abs(JudgmFC - actual)/actual * 100;
```

A variable with shifted APE values is created where extreme large outliers are shifted to a lower value. Otherwise, the graphs and the regression analyses might be dominated by these outliers.

```
ape_stat_shift = min(ape_stat,300);
ape_man_shift  = min(ape_man,300);
```

Calculating the Difference between the Manual and the Statistical Forecast

Two variables are created that describe the difference between the manual and the statistical forecast: APE_DIF and FC_DIF.

APE_DIF contains the difference between the average percentage error of the statistical and the manual forecast. Positive values mean that the APE of the judgmental forecast is larger.

```
APE_DIF = ape_judgm - ape_stat;
```

Extreme values beyond -500 and 500 are shifted toward -500 and 500, respectively.

```
if APE_DIF ne . and APE_DIF < -500 then APE_DIF = - 500;
else if APE_DIF > 500 then APE_DIF = 500;
```

FC_DIF contains the difference between the judgmental forecast and the statistical forecast.

- Positive values mean that the judgmental override increased the forecast.
- Negative values represent a decrease of the forecast through the override.

```
FC_DIF  = JudgmFC-statfc;
```

Extreme values beyond -5,000 and 5,000 are shifted toward -5,000 and 5,000, respectively.

```
if FC_DIF ne . and FC_DIF < -5000 then FC_DIF = - 5000;
else if FC_DIF > 5000 then FC_DIF = 5000;
```

DESCRIPTIVE ANALYSIS OF THE FORECAST ERROR

This section shows that you can gain initial insight into the relationships of your data just by using descriptive methods. Only selected results are shown here. For more insight refer to Svolba 2017, Chapter 10.

CHECKING THE DISTRIBUTION OF THE FORECAST ERROR

The mean of APE_STAT is 86.5 with a standard deviation of 585.8. The median is 40.6. Looking just at the mean and the median, it seems that the forecast quality of this company is quite bad. On average the forecast is 86.5% away from the true demand. And 50% of the forecasts have a forecast error of larger than 40.6%.

However, bear in mind that many products with a very short data history are forecasted in this business example. Products that were just put on the market do not provide a lot of insight in their demand pattern. In many of these cases, only a rough estimate can be created. Rough estimates based on fewer data points have a higher forecast error. The influence of the available history can also be seen later on when the model type or the product age is analyzed.

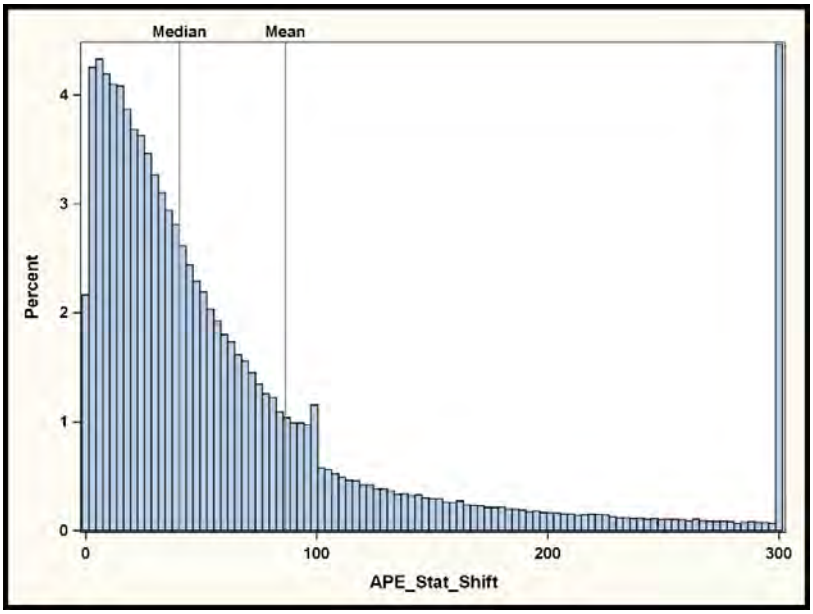
You also see that the distribution of the forecast error is heavily skewed to the right. The mean is twice as large as the median and the standard error is extremely high, due to a few outliers with extreme values. Table 4 also reveals that the maximum forecast error is higher than 230,000.

Quantile	Value
100% Max	238,954.6
95%	276.6
90%	169.5
75% Q3	81.7
50% Median	40.6
25% Q1	18.0
10%	7.0
0% Min	0

Table 4. Quantiles of APE_STAT

The statistics in this table have been created with the following SAS code.

```
proc means data=fc_mart mean std min p10 q1 median q3 p90 p95 max maxdec=1;
 var APE_Stat;
run;
```

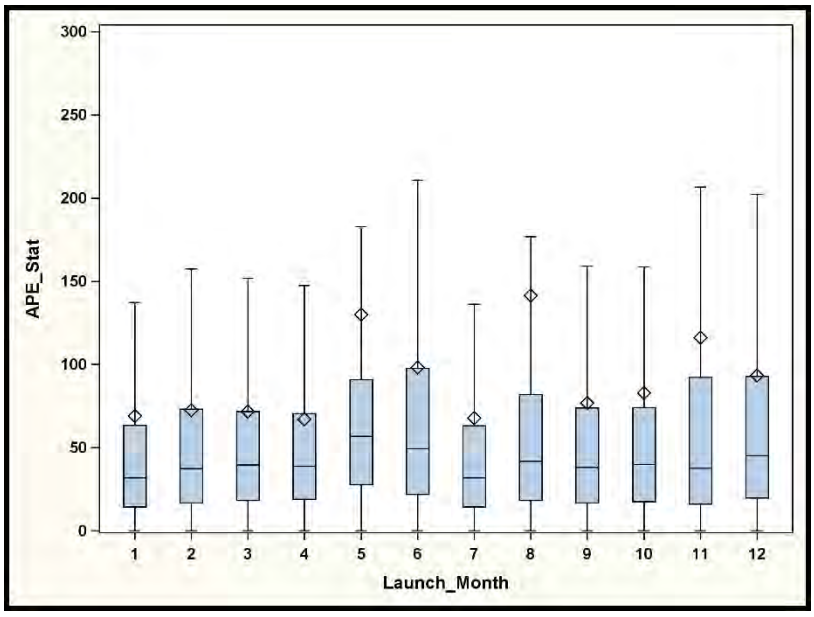


Output 3. Histogram for APE_STAT_SHIFT

Calendar Month of Product Launch

Output 4 shows that the forecast quality also differs by calendar month. You see that products launched in May or June have larger forecast errors compared to products launched in July.

- This might be due to an association between the launch of product groups that are easier to forecast in certain months of the year.
- Another reason might be the interaction between the launch month and the seasonal demand pattern, like the larger demand around Christmas. Products that are launched in July might directly move from a demand peak in the launch phase to a demand peak in the pre-Christmas season.



Output 4. Histogram for APE_STAT_SHIFT by Launch Month

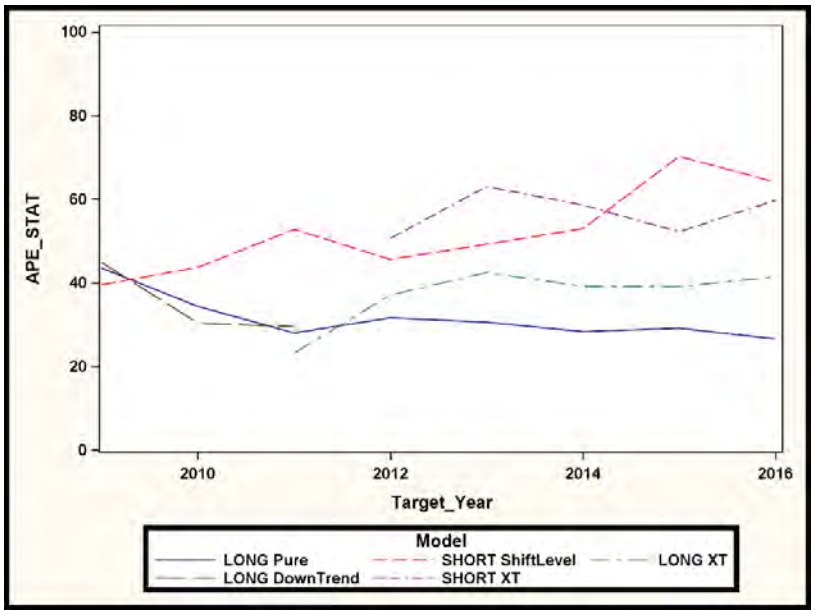
A multivariate analysis with interaction might give more insight into this question. This is shown in the subsequent sections.

DIFFERENT FORECAST MODELS PRODUCE DIFFERENT FORECAST ERRORS OVER TIME

Before interpreting the forecast error of the different model types, you have to bear in mind that for each article, the model that generates the best forecast is selected.

- This also implies that Model A might be selected for the easy-to-forecast articles, while Model B, which is more robust, is selected for articles with a complicated demand pattern.
- In terms of average forecast quality, Model B might look bad compared to Model A, as it is mostly used to forecast complicated articles.

The result is shown in Output 5.



Output 5. Median Course over TARGET_YEAR of Different MODEL Types

Interpretation of the Results

You see that there are different courses of the forecast error over time for different model types. The red dashed line and the violet double-dashed dotted line represent the median course of the forecast error of the models with short demand history.

- You see that there is a slight increase in the forecast error of the SHORT ShiftLevel model over time.
- The SHORT XT model has some variation over time, but stays stable on average.

You see that model LONG DownTrend was discontinued in 2011 and replaced by model LONG XT. It is interesting to see that only in 2011, model LONG XT has a better forecast quality. The forecast error however increases in the later years.

- This might be an indication that model LONG XT is overfitted and only fit well when it was first introduced.
- It might also be the case that the articles with a complicated demand pattern were forecast with model LONG XT. If these articles were forecast with the LONG Pure model, the average forecast

error for this model might have increases as well.

Creating the Line Chart of the Medians

In order to display the context in a single chart, a line plot of the median forecast errors per target year and model type can be created. First you calculate the median forecast error per subgroup with the MEANS procedures and store the results in a data set APE_MEDIAN.

```
PROC MEANS DATA=fc_mart NWAY NOPRINT;
  VAR ape_stat;
  CLASS model Target_year;
  OUTPUT OUT=ape_median mean= median= /autoname;
RUN;
```

Note that the NOPRINT option suppresses printed output in the results window. NWAY specifies that only the lowest level of the subgroup hierarchy, MODEL x TARGET_YEAR, is stored in the output data set.

Next you use the SGPLOT procedure to plot the median course over target year, by model type.

```
PROC SGPLOT DATA=ape_means;
  SERIES X=Target_year Y=ape_stat_median / group=model;
  YAXIS LABEL ="APE_STAT" min=0 max=100;
RUN;
```

THE EFFECT OF MANUAL OVERWRITES

ORIGIN OF MANUAL OVERWRITES

In the operational forecasting process, the statistical forecast is often not used as the final forecast. Demand planners perform a judgmental correction to the statistical forecast.

- This correction is based on their personal experience with the business context.
- They might have additional information available that should influence the forecast value, like a regionally isolated marketing campaign for a certain product that is planned to run next month.
- It might also be the gut feeling of the demand planner.
- The judgmental correction might also have a political reason.
- Sometimes the values of the statistical forecast are just rounded to add a judgmental flavor to it.

AWARENESS OF THE DEFINITION OF THE DERIVED VARIABLES

For a clear interpretation of the results, it is a best practice to show and repeat the definition of the derived variables in the comments or in the results file. This makes sure that the value and the sign of the difference can immediately be interpreted by the analyst.

In the case of the difference in percentage error, the following definition is used:

APE_DIF = APE_MAN – APE_STAT

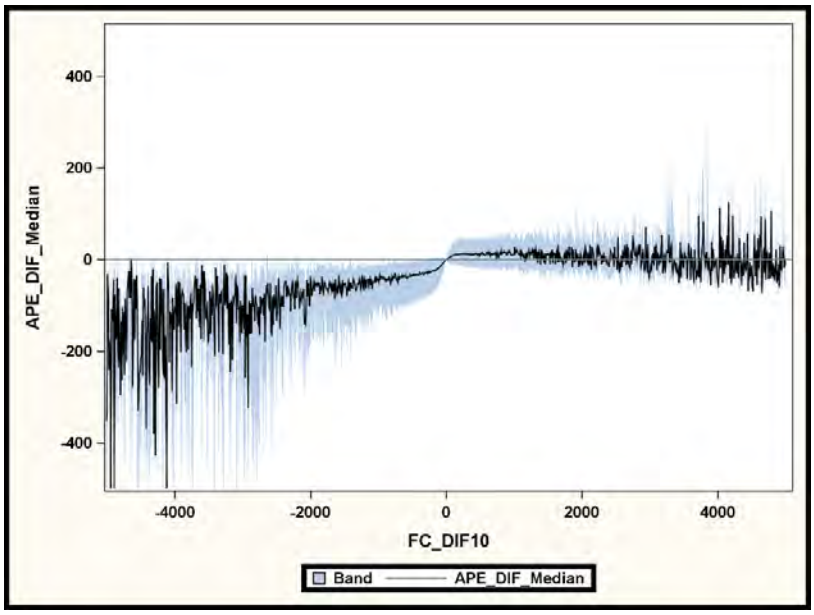
- Thus, a positive value of APE_DIF means that the forecast error got larger because of the manual correction. The manual correction did not improve the accuracy of the forecast.
- A negative value of APE_DIF means that the forecast error got smaller by the manual correction and it improved the accuracy of the forecast.

FC_DIF = JUDGMFC – STATFC

- A positive value indicates that the manual override increased the forecast; a negative value decreased the forecast.

THE BENEFIT OF LARGE OVERWRITES

Output 6 shows the manual override on the X axis and the change in the average percentage forecast error at the Y axis.



Output 6. Relationship between FC_DIF and APE_DIF

Note that the override values at the X axis are rounded to 10 in order to avoid a too busy graph. For each override value at the X axis, you can have several observations. Thus, the median and the first and third quartile are calculated.

- The median is shown by a solid black line.
- The first and third quartile are displayed by a band.

You see that larger changes of the forecast value have on average a much larger effect in decreasing the forecast error. This finding corresponds with the work of Goodwin 2009, who states that small changes to the forecast value usually do not improve forecast quality, while larger changes have a positive effect. Small changes to the forecast are made by the demand planner with less care than large changes. If a large change to the statistical forecast has to be applied, the planner investigates, in much more detail, whether the adjustment shall be made or not.

The consequence should not be to recommend large changes instead of small changes. Rather, the goal is to eliminate the small changes that do not add any benefit and save the time for analyzing whether large changes shall be made.

Again you see that a decrease of the forecast through the manual overwrite had on average a positive effect. You however also see that even large positive changes have a positive effect on the forecast quality. Demand planners obviously put more thought into large changes and apply them only if they are really convinced about it.

Creating the Line Chart and the Band Chart with SGPLOT Procedure

The above graph has been created in two steps. First the median and the first and third quartile have been calculated with the MEANS procedure. Note that the AUTONAME option has been used to automatically create the respective variable names that contain the name of the descriptive statistic:

```
proc means data=fc_mart noprint nway;
 class fc_dif10;
 var ape_dif;
 output out= dif10_mean median= q3= q1= / autoname;
run;
```

Next the data is plotted with the SGPLOT procedure. Note that the BAND statement should precede the SERIES statement for the mean value. Otherwise, the line hides behind the band.

```
proc sgplot data=dif10_mean;
 band x=fc_dif10 lower=ape_dif_q1 upper=ape_dif_q3;
 series x=fc_dif10 y=ape_dif_median;
 refline 0 / axis=y;
run;
```

QUANTIFYING THE EFFECT OF DRIVERS FOR THE FORECAST ERROR WITH THE GLMSELECT PROCEDURE

OVERVIEW

OLS regression enables you to quantify the effect of each explanatory factor, like PRODUCT_AGE or PRODUCT_GROUP on the forecast error. You can run a univariate regression with only one input variable for each influential factor.

This provides insight into the explanatory power of the respective variable on the forecast error. It also allows quantifying this relationship using the regression coefficients.

- If you use an interval input variable, you receive one coefficient.
- If you use a categorical input variable, you receive a coefficient for all categories expect the reference category.

Interpretability versus Statistical Correctness

As in many business analyses, the decision between interpretability and applicability of the results and the statistical correctness needs to be made. From a statistical point of view, the target variable APE_STAT should be definitely log transformed before being used in the regression model.

The price that is paid in this case is that the regression coefficients cannot then be interpreted in units of the target variable. Svolba 2016 shows in more detail that the model fit of the model using the log-transformed variable is not better than those of the untransformed variable. In this case, to be on the safe side, it is better to leave the variable untransformed for better interpretability.

It always makes sense, however, to check both models for their model fit. This enables you to see how much the fit between the two modeling approaches differs.

UNIVARIATE ANALYSIS USING THE GLMSELECT PROCEDURE

The following code example shows how you can perform this analysis for using the GLMSELECT procedure for an interval variable and a categorical variable.

```
PROC GLMSELECT DATA=fc_mart;
 MODEL ape_stat_shift =  product_Age  /  SHOWPVALUES;
RUN;

PROC GLMSELECT DATA=fc_mart;
 CLASS product_group / PARAM=effect;
 MODEL ape_stat_shift =  product_group / SHOWPVALUES;
RUN;
```

Table 5 shows the available input variables ordered by descending R^2. You see that variable MODEL, PRODUCT_AGE, and PRODUCT GROUP are the most influential variables.

Ranking	Input Variable	R-squared linear	Beta linear
1	MODEL	0.0554	
2	PRODUCT_AGE	0.0433	-0.51
3	PRODUCT_GROUP	0.0224	
4	LAUNCH_MONTH	0.0172	
5	TARGET_YEAR	0.0102	4.16
6	TARGET_CALMONTH	0.0084	
7	LEAD_TIME	0.0046	1.68
8	PRICE_INDEX	0.0016	-0.02

Table 5. Input Variables Sorted by Descending Adjusted R-Square

The coefficient of PRODUCT_AGE of -0.51, for example, can be interpreted as the average decrease in forecast error for each additional month of demand history. You can conclude that an additional year of forecast history results on average in a decrease of around 6 percentage points (0.51 times 12 months).

Variables Model Type and Product Age

You see that the two top variables are model type and product age with an R^2 in the linear model of 5.54% and 4.33%, respectively. Variable model type implicitly also contains information about the product age, as the models are separated by short and long data history.

The fact that the explanatory power of the variable model type is higher than those of variable product age indicates that the model type contains more information than just the length of the available data history.

In a multivariate regression model, it is interesting to see whether both variables are still selected or whether the additional explanatory power of the second variable is not high enough to cause the second variable to be added to the model. Using a multivariate regression model enables you to investigate the relative importance of a variable compared to the fact that other variables are already in the model.

MULTIVARIATE ANALYSIS OF THE INFLUENCE ON THE FORECAST ERROR

Code of GLMSELECT Procedure

The following code has been used to perform a multivariate regression analysis with stepwise selection of the input variables:

```
PROC GLMSELECT DATA=fc_mart;
 CLASS product_group launch_month model target_calmonth / PARAM=effect ;
 MODEL ape_stat_shift =
                 product_group|price_index|launch_month|product_age|
                 model|lead_time|target_calmonth|target_year_shift     @1
                     /DETAILS=steps
                      SELECTION=stepwise (SELECT=sl)
                      ORDERSELECT
                      SHOWPVALUES;
RUN;
```

Note the following from the code:

- The CLASS statement, which lists all four categorical variables, is used and the EFFECT coding is requested with the PARAM option.
- The MODEL statement contains the list of input variables. Note that the list of variables could also be specified with blanks between the variables.
- Using the “pipe” | has the advantage that the MODEL statement can be used in a flexible way if, for

example, quadratic terms shall be requested.

- @1 indicates that you want to use these variables only to the power of 1.
- @2 would cause the normal effect and the quadratic effect for each variable.
- This feature is not limited to the GLMSELECT procedure; it can be applied for all regression procedures with MODEL statements.
- A stepwise regression is requested with the SELECTION= stepwise option.
- The SL option specifies that the significance level of each variable to enter or leave the model shall be checked.
- Information about each forecasting step is requested with the DETAILS= steps option.
- The option ORDERSELECT causes the parameters in the final parameter estimates table to be sorted in the order of their inclusion into the model, instead of alphabetic order.
- The SHOWPVALUES options requests that p-values are shown in the parameter estimates table.

Results of the Multivariate Regression

Table 6 shows the list of input variables in their selection order for the linear regression for the non-transformed target error. You see that all available eight variables are selected, even if the last variable only marginally contributes to the improvement of the model fit. This is also due to the large number of observations (> 400,000 records).

Ranking	Input Variable	Adjusted R-square
0	INTERCEPT	0%
1	MODEL	5.46%
2	TARGET_CALMONTH	6.58%
3	PRODUCT_GROUP	7.59%
4	TARGET_YEAR_SHIFT	8.50%
5	PRODUCT_AGE	9.04%
6	LEAD_TIME	9.76%
7	LAUNCH_MONTH	9.90%
8	PRICE_INDEX	9.90%

Table 6. Input Variables Sorted by Adjusted R-Square of the Multivariate Model

The first variable MODEL has been selected, which adds 5.46% of the explanation of the values in variable APE_STAT_SHIFT. Note that in Table 5 you have also seen variable MODEL on top of the list ordered by their univariate contribution, so its selection is intuitive.

Additional Information Is Prioritized

In step 2, however, variable TARGET_CALMONTH has been selected, although it was only at rank 6 of the ordered list in Table 5. It can be assumed that it “overtook” the other variables, because after variable MODEL was selected, the additional explanatory power of the variable TARGET_CALMONTH was higher than that of the others.

At rank 2 of the univariate analysis in Table 5, you saw PRODUCT_AGE. In the multivariate model it is selected only in the fifth step. In the multivariate regression model, the variables are considered in a combined or simultaneous way.

As variable MODEL is already in the regression equation, the additional explanatory power of variable PRODUCT_AGE is not that high anymore. Variable MODEL has already “told” part of its information, for example, that older products, forecasted with “LONG-models” have a lower forecast error than younger products, forecasted with “SHORT-models”.

True Increase of Model Fit

Thus, the relative benefit of variable PRODUCT_AGE is not 4.75% as in the univariate model but only 0.54% (9.04 – 8.50). Variables TARGET_CALMONTH, PRODUCT_GROUP, and TARGET_YEAR are selected first as they obviously can “tell new details”.

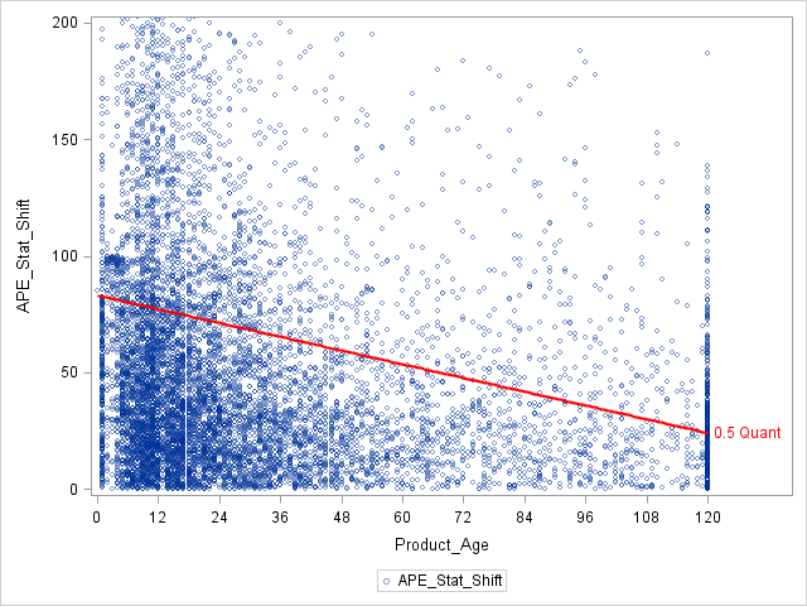
You also see that the additional explanatory power of PRODUCT_GROUP is not 2.24% as shown in Table 5 for the univariate results. It is only around 1%. This indicates that variables MODEL and TARGET_CALMONTH have already contributed more than half of what variable PRODUCT_GROUP could contribute in a univariate model.

STUDYING THE REGRESSION RESULTS VISUALLY

Univariate Analysis of PRODUCT AGE

Output 7 shows the plot of the predicted APE_STAT values from a univariate OLS regression with variable PRODUCT_AGE. The actual values are plotted as blue circles. The predicted values are plotted as a solid red line. You see a decreasing trend of the forecast error over the increasing values of product age.

This result corresponds with the findings shown Table 3. A larger data history for a product decreased the forecast error.



Output 7. Plot of the Predicted APE_STAT Values from the Univariate Regression Model

Multivariate Analysis of Product Age Provide More Insight

Output 8 shows the same plot, however, based on the predicted values of a more detailed regression model. In this model all selected variables have been included. (Compare Table 6.)

You still see a downward trend of the forecast error over product age. However, the relationship is no longer a straight line as the effect of product age is not only measured on its own. It is corrected for the effect of the other available variables.

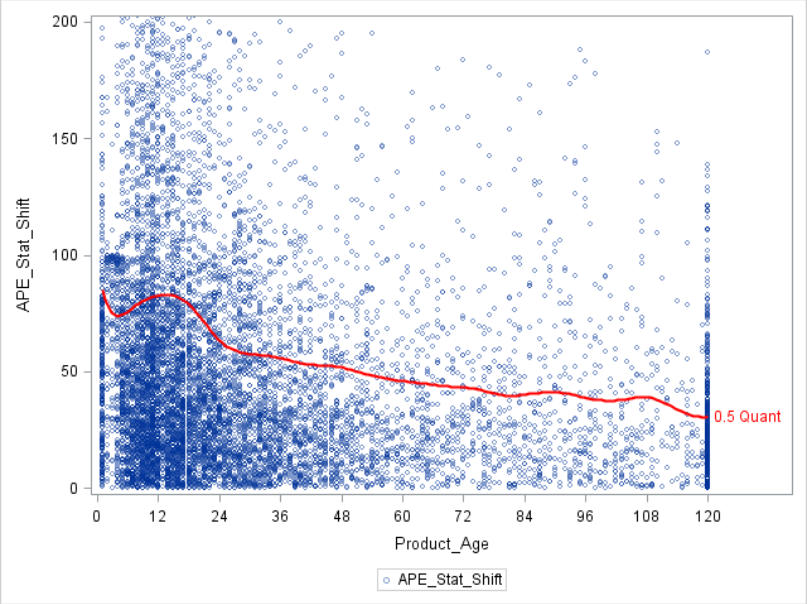
Thus, Output 8 shows the effect of the variable PRODUCT_AGE after correcting for the influence of other co-variables.

This provides much more detailed insight in the effect of variable product age. All other co-influences have already been filtered and this enables you to view only the remaining effect of variable product age.

- You see an interesting drop of the forecast error around month 3, which is hard to explain from a

business point of view.

- You see that additional months of data history have the strongest effect between months 18 and 24, this is when a second full year of data history is achieved.
- You see a rather linear decreasing trend after month 24.



Output 8. Plot of the Predicted APE_STAT Values from the Multivariate Regression Model

SAS Code

The code for the multivariate analysis is shown here. Note that an OUTPUT statement is added to the GLMSELECT procedure to output the predicted values from the regression model. These predicted values are then used in the SGPLOT procedure.

```
PROC glmselect DATA=FC_Mart_10smp ;
  partition rolevar=_ROLE_ (train = 'TRN' validate='VAL');
  CLASS product_group launch_month model target_calmonth / PARAM=effect ;
  MODEL ape_stat_shift = product_group|price_index|launch_month|
                         product_age|model|lead_time|
                         target_calmonth|target_year @1
                    /selection=stepwise(choose=validate slentry=0.001);;
    output out=LinReg1Pred p=APE_STAT_PRED
RUN;
```

The SGPLOT procedure is used to combine a SCATTER plot for the actual data value with a SPLINE plot for the predicted values.

```
proc sgplot data=LinReg1Pred;
 scatter y=ape_stat_shift x=product_age/
             markerattrs=(size=5) transparency=0.5
             filledoutlinedmarkers;
 pbspline y=APE_STAT_PRED x=product_age/
             lineattrs=(thickness=2 color=red) nknots=20 nolegfit
             curvelabel="0.5 Quant" curvelabelattrs=(color=red) nomarkers;
   xaxis values= (0 to 120 by 12) ;
   yaxis max=200;
   where product_age ne 0;
run;
```

GETTING ADDITIONAL INSIGHT WITH QUANTILE REGRESSION

BASIC IDEA OF QUANTILE REGRESSION

Idea of Linear Regression

With a linear regression model, as presented in the previous section, an important implicit assumption is made: The conditional mean of the dependent variable is modeled.

In linear regression, the model equation is:

$$Y_i = x_i'\beta + \varepsilon_i$$

And the vector β is determined by minimizing the errors:

$$\min \sum_{i=1}^{n} \varepsilon_i^2$$

In many cases the conditional mean is what should be modeled and predicted and not a lot of thought is put into that fact. There are, however, some cases where you are not interested in a model that explains or predicts the conditional mean of the distribution of the dependent variable, but you are rather interested in specific quantiles.

Ordinary least squares regression 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 idea of regression models to conditional quantiles of the response variable, such as the 90th percentile (0.9 quantile).

Quantile Regression

Here the quantile regression comes into play. It allows you to model selected conditional quantiles. You receive a model that predicts the value of a certain quantile instead of the mean. The model equation for a quantile τ is the following:

$$Q(\tau|X = x) = x'\beta(\tau) + \varepsilon_i$$

Here the following expression is minimized,

$$\min \sum_{i=1}^{n} \rho_\tau|\varepsilon_i| + \sum_{i=1}^{n} (1 - \rho_\tau)|\varepsilon_i|$$

where $\rho_\tau|\varepsilon_i|$ and $(1 - \rho_\tau)|\varepsilon_i|$ are the penalty terms for over and under estimation.

In the case of the forecast errors analysis, this might answer the following business questions: “What influences the 1st quartile of the target variable?”

- Do you want to see the predictors for a better quarter of the forecast errors?
- Quantile regression shows you the list of variables and their parameters that are related to the forecast error value, which is not exceeded by the 25% of time series with the lowest forecast error.

You can perform the same procedure for the 3rd quartile to get insight on those variables that relate to the upper range of the forecast errors.

- You also might want to know whether the list of influential factors for the upper and the lower quarter of forecast error differs.

QUANTILE REGRESSION FOR THE STATISTICAL FORECAST ERROR

Rationale of Using Quantile Regression

Quantile regression allows you to better understand the influence of independent variables for different quantiles of the statistical forecast error. The 0.75 quantile regression enables you to identify and parameterize those factors that influence whether a certain forecast error is not exceeded by 75% of the time series.

Linear regression only deals with the analysis of the average location of the forecast error and the variables that influence this average error.

Quantile regression enables you to make assumptions about the extreme areas. “Which forecast error is not exceeded, if I have data history of more than 24 months?” From a business point of view, the information that “75% of the time series have a forecast error smaller than x” is more important than the average forecast error.

The same applies to the 0.25 quantile: You receive information about the forecast error that is not exceeded by your best 25% of the time series, if you increase the available time history.

Quantile Regression for Selected Quartiles

In this example, the quantile regression for the 0.1, 0.25, 0.5, 0.75 and 0.9 quantile of the statistical forecast error is performed using the following SAS code:

```
PROC QUANTSELECT DATA=fc_mart_10smp;
  CLASS product_group launch_month model target_calmonth / PARAM=effect ;
  MODEL ape_stat_shift = product_group|price_index|launch_month|
                         product_age|model|lead_time|
                         target_calmonth|target_year @1
                    /quantile= 0.1 0.25 0.5 0.75 0.9
                     selection=stepwise(choose=validate slentry=0.001);
  ods select SelectionSummary;
RUN;
```

Note that the QUANTILE option in the MODEL statement is used to specify the quantiles of interest.

Variables Selected for Different Quantiles

Tables 7–9 show the variables that have been selected by different quantile regressions. You see that these sets differ. This indicates that for different quantiles, different combinations of influential factors are relevant.

- You see that variables model type, product age, product group, and lead time are selected in every model.
- You also see that the model for the 0.25 quantile uses a smaller set of variables than the model for the 0.5 and the 0.75 quantiles.
- You could also study the coefficients of the parameter for each model. In that case, you would see that the coefficients differ for each model, even if the same set of variables is selected.

Step	EffectEntered	SBC
1	Model	36293.1995
2	Product_Group	36356.7482
3	Lead_Time	36356.6815
4	Product_Age	36330.2975

Table 7. Variables Selected for the 0.25 Quantile

Step	EffectEntered	SBC
1	Model	43049.0936
2	Launch_Month	43118.2647
3	Lead_Time	43087.3543
4	Product_Group	43094.7320
5	Product_Age	43077.2793
6	Target_CalMonth	43123.6095
7	Target_Year	43125.2832

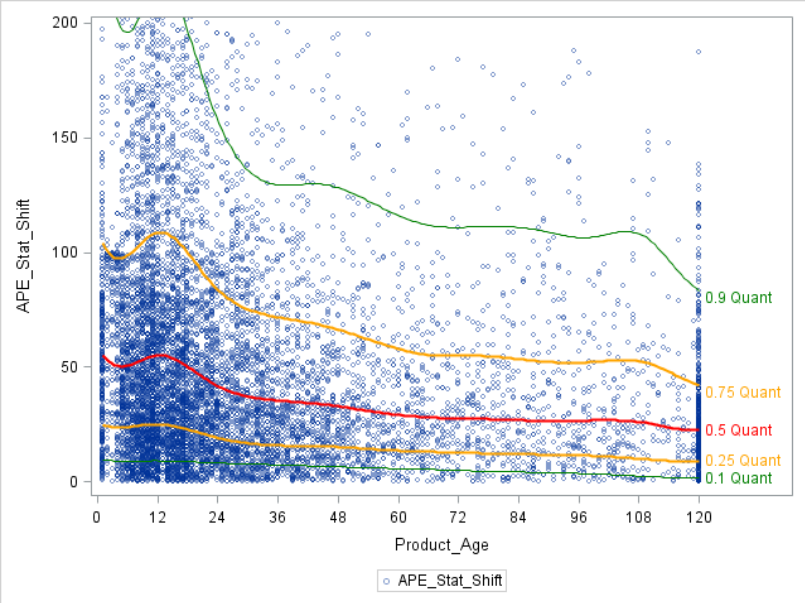
Table 8. Variables Selected for the 0.5 Quantile

Step	EffectEntered	SBC
1	Model	44077.9745
2	Target_CalMonth	44027.8505
3	Launch_Month	44054.7556
4	Lead_Time	43978.5909
5	Product_Age	43929.8234
6	Product_Group	43931.3299
7	Target_Year	43882.6948

Table 9. Variables Selected for the 0.75 Quantile

Displaying the Results Visually

Output 9 displays the results of the multivariate quantile regression in the same way as shown in Output 8 for the OLS regression.



Output 9. Plot of the Predicted APE_STAT Values from the Multivariate Quantile Regression Model

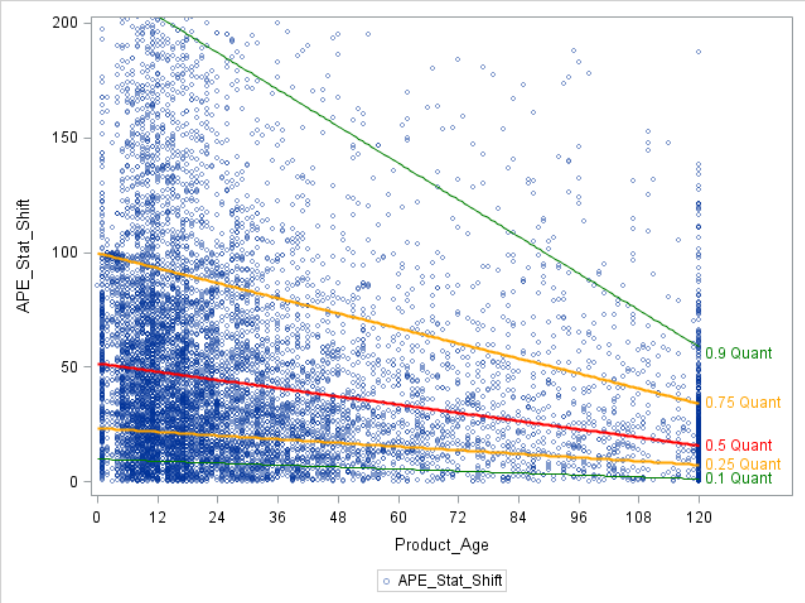
The blue circles represent a scatter plot for PRODUCT_AGE and APE_STAT of the actual data. You see solid lines in different colors for the predicted values of APE_STAT from the multivariate quantile regression for different quantiles. The relationship of product age, however, is not only measured on its own. It is corrected for the effect of the other available variables as multivariate regression model is used.

- It is remarkable to see that for the 0.1 and the 0.25 quantile of the forecast error (the value that is not exceeded by your top time series) the relationship between the product age and forecast error is very flat.
- You see that the increase in forecast errors around months 12 (which is hard to explain from a business point of view) increases with larger quantiles. It is almost not present for quantile 0.25 and 0.5.
- You also see that additional data history after month 36 has only a marginal effect, except for quantile 0.9, where additional time history does matter for higher product age values.

Results from Univariate Quantile Regression for Variable PRODUCT_AGE

Output 10 shows the results from a univariate quantile regression model that uses only variable PRODUCT_AGE. Similar to Output 7, you see only linear trends, as the influence is not corrected for other available variables. You clearly see that the slope for the trend curve is different for the different quantiles. Larger quantiles have a steeper curve than the lower quantiles.

This leads to the interpretation that additional available data history has a much stronger positive effect on the higher quantiles than the lower quantiles. The times series with a smaller forecast error do not benefit as much from additional data history as those that are in general harder to predict.



Output 10. Plot of the Predicted APE_STAT Values from the Univariate Quantile Regression Model

Create the SCATTER and SPLINE Plot

The plot is generated in a similar way as shown above. Here you use a separate PBSPLINE statement for each quantile.

```
proc sgplot data=QuRegPred;
 scatter y=ape_stat_shift x=product_age/
         markerattrs=(size=5)   transparency=0.5 filledoutlinedmarkers;
 pbspline y=APE_STAT_PRED1 x=product_age/
        lineattrs=(thickness=1 color=green ) nolegfit
        curvelabel="0.1 Quant" curvelabelattrs=(color=green) nomarkers;
 pbspline y=APE_STAT_PRED2 x=product_age/
       lineattrs=(thickness=2 color=orange) nolegfit
      curvelabel="0.25 Quant" curvelabelattrs=(color=orange)  nomarkers;
 pbspline y=APE_STAT_PRED3 x=product_age/
```

```
       lineattrs=(thickness=2 color=red) nolegfit
       curvelabel="0.5 Quant" curvelabelattrs=(color=red) nomarkers ;
 pbspline y=APE_STAT_PRED4 x=product_age/
       lineattrs=(thickness=2 color=orange) nolegfit
       curvelabel="0.75 Quant" curvelabelattrs=(color=orange) nomarkers;
 pbspline y=APE_STAT_PRED5 x=product_age/
       lineattrs=(thickness=1 color=green ) nolegfit
       curvelabel="0.9 Quant" curvelabelattrs=(color=green) nomarkers;
   xaxis values= (0 to 120 by 12) ;
   yaxis max=200;
run;
```

CREATING A PROCESS PLOT FOR THE PARAMETER ESTIMATES

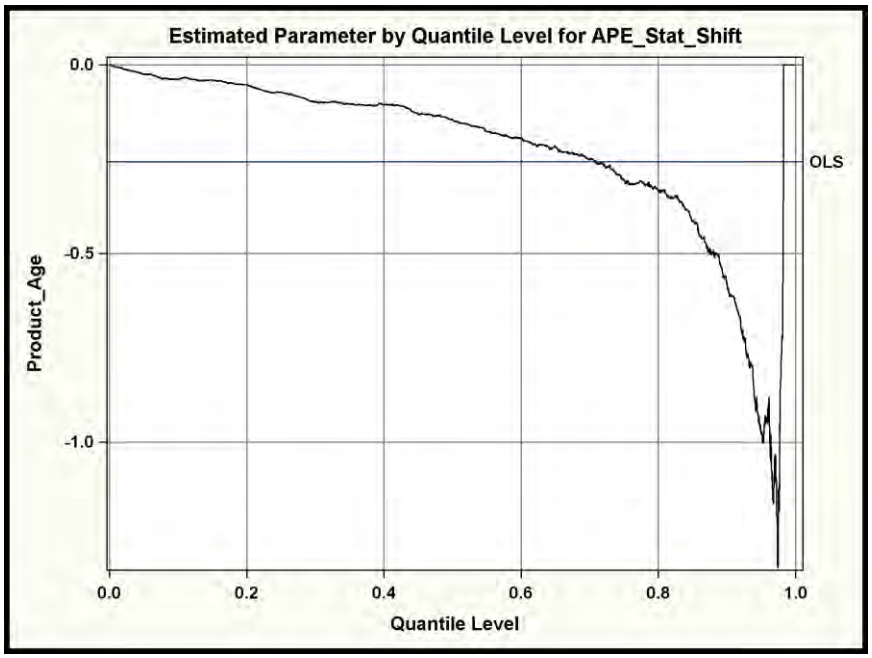
General Idea

Output 10 shows that with increasing quantiles, the coefficient of variable PRODUCT_AGE changes. This "quantile process" can be easily analyzed with the QUANTSELECT and the QUANTREG procedure. However, only the QUANTREG procedure enables you to create the process plot that enables you to visually study the effects.

A quantile process means that a quantile regression for all quantiles from 0 to 1 is performed. The results enable you to study the relationship of an independent variable on a target variable across the quantiles. This provides insight into the influence of the independent variable on different quantile levels.

Process Plot and Interpretation

The process plot for variable product age is shown in Output 11.



Output 11. Process Plot for PRODUCT_AGE

- From the plot, you see the range of quantiles on the horizontal axis and the parameter estimate for product age on the vertical axis.
- The value for the ordinary least squares estimator is shown as a horizontal line and is labeled as OLS. Note that the OLS estimate for the (conditional) mean is constant around -0.3. It does not

depend on the quantile level.

You see that the estimate in the quantile regression for parameter product age decreases over the quantile process.

- For larger quantiles of the forecast error, the estimate of product age decreases to -1. Here an additional month decreases the forecast error by 1 percentage point.
- One additional month of product age decreases the conditional 0.2 quantile of the forecast error by around 0.1 percentage points.
- One additional month of product age decreases the conditional 0.8 quantile of the forecast error by around 0.35 percentage points.

You learn that the influence of product age is not constant over the quantile process. Higher quantiles of the forecast error benefit much more (by getting smaller) from an additional month of product age available in the data.

Using the QUANTREG Procedure

The QUANTREG procedure can be used to create a quantile process and the process plot with the following code:

```
PROC QUANTREG data=fc_mart_smp10000 algorithm=simplex ;
 CLASS Product_Group  model  ;
 MODEL ape_stat_shift =  product_group Product_Age model target_year_shift
                        /   QUANTILE=process
                            PLOT=quantplot(Product_Age) /UNPACK OLS
                             ;
RUN;
```

Note that the shifted average percentage error is used here. Otherwise, the outliers in the 0.99 and 1.00 percentile would also cause extreme values in the estimate of product age for these quantiles.

- You specify the value PROCESS with the QUANTILE = option to request the analysis of the quantile process for quantiles from 0 to 1.
- The PLOT = QUANTPLOT option requests the quantile plot for PRODUCT_AGE.
    - The UNPACK option creates an individual process plots.
    - It enables you to specify the OLS option that shows the ordinary least squares estimate as a horizontal line in the plot for comparison.

RUN TIMES, SAMPLING, AND DATA PARTITIONING

SAMPLING THE DATA FOR QUANTILE REGRESSION

The following run times are measured on 4 core Intel i7 processor with 2.3 GHz. Running stepwise linear regression on the entire data set of 411743 observations takes 4 seconds. Running stepwise quantile regression for the median on a sample of 10.000 observations takes 29 seconds real time and 1 minute 16 seconds CPU time. Note that the real time differs from the total CPU time as the procedure distributes computing across the available nodes.

This result shows that quantile regression is very compute intense. It thus makes sense to sample the available data. SAS®9 provides the SURVEYSELECT procedure to sample the data.

```
proc surveyselect data=FC_MART method=srs smpsize=10000
                  seed=60502    out=FC_Mart_10smp;
run;
```

You specify method SRS for simple random sampling. The SEED option enables you to fix the seed for the random sampling to generate reproducible results with every run.

PARTITIONING THE DATA

In order to partition the data into training and validation data, you can use the PARTITION statement in the GLMSELECT or QUANTSELECT procedure. Here you can either split the data randomly or split the data according to a predefined ROLE variable.

Using a predefined ROLE variable makes sense if you want to use a fix seed for the portioning. As in the case so sampling, fixing the seed makes sense if you want to generate reproducible results with every run.

The PARTITION statement in the GLMSELECT or QUANTSELECT procedure does not enable you to specify the seed. Using a SAS DATA step you can use a fixed seed to partition the data.

```
data FC_Mart_10smp;
 set FC_Mart_10smp;
 format _ROLE_ $3.;
 call streaminit(seed=2311);
 if rand('Uniform') > 0.3 then _ROLE_ = 'TRN';
 else _ROLE_ = 'VAL';
run;
```

You initialize the random number generation with the CALL STREAMINIT statement, where you specify the seed. You assign the TRAIN or VALID role depending on the values of the random number generated with the RAND function.

SAS Viya provides the REGSELECT procedure for linear regression and the QTRSELECT procedure for quantile regression.

LINEAR AND QUANTILE REGRESSION WITH SAS VIYA

Sampling and Partitioning the Data

SAS Viya allows distributed high performance computing for large-scale data. SAS Viya provides the REGSELECT procedure for ordinary least squares regression and the QTRSELECT procedure for quantile regression.

SAS Viya provides very good performance. If you still want to sample your data in SAS Viya, you can use the PARTITION procedure in a similar way as shown above with the SURVEYSELECT procedure.

```
proc partition data=FC_MART samppct=10 seed=60502;
   output out=FC_Mart_10smp;
run;
```

Different to SAS9 the PARTITION statement in the SAS Viya QTRSELECT procedure enables you to specify a seed and thus fix the partitioning for reproducible results.

The QTRSELECT procedure

The following code shows how the QTRSELECT procedure can be called in SAS Viya to run the same quantile regression as presented in the previous chapter. You see that the code is very similar to the code used for the QUANTSELECT procedure.

```
PROC qtrselect DATA=FC_Mart_10smp ;
 partition fraction(validate=0.3 seed=2311);
 CLASS product_group launch_month model target_calmonth / PARAM=effect ;
 MODEL ape_stat_shift =
                  product_group|price_index|launch_month|product_age|
                  model|lead_time|target_calmonth|target_year_shift @1
                    /quantile= 0.1 0.25 0.5 0.75 0.9;
    selection method=stepwise(choose=validate slentry=0.001)  ;
    output out=cas1.QuRegPred
```

```
            copyvars=(price_index product_age ape_stat_shift)
            p=APE_STAT_PRED
            role=Role;
    ods output selectionsummary=work.selectionsummary;
RUN;
```

Note that when using the OUTPUT statement, you have to explicitly specify the variables that shall be copied to the output data set.

CONCLUSION

You have seen that the application of analytical methods provides many relevant insights to help you make better business decisions. This is not only the case for the analysis of the forecast error as presented in this paper, but also for many other business questions. Svolba 2016 presents a collection of examples and SAS code where relevant business questions are analyzed with analytical methods.

In the example presented here, you have seen that the descriptive method also provides a lot of insight. Using linear regression enables you to better quantify the importance of different influential factors and to assess the strength and the direction of different categories. You see that the multivariate analysis provides a more comprehensive picture than the isolated univariate analysis of influential factors.

Quantile regression enables you get a clearer picture about the extremes of your distribution. You learn which influential factors trigger the fact that forecast errors do not exceed a certain limit. In the above example you have seen that some variables are important to explain the higher quantiles of the outcome but not the lower quantiles of the outcome.

The SAS platform with SAS9 and SAS Viya procedures provides a comprehensive set of analytical methods that enable you gain more insight in the relationships between your data and your business processes.

REFERENCES

Svolba, Gerhard. 2017. *Applying Data Science: Business Case Studies Using SAS®*. Cary, NC: SAS Institute Inc.

Fildes R., P. Goodwin, M. Lawrence, and K. Nikolopoulos K. 2009. “Effective Forecasting and Judgmental Adjustments: An Empirical Evaluation and Strategies for Improvement in Supply-Chain Planning.” *International Journal of Forecasting* 25(1): 3–23 (DOI: 10.1016/j.ijforecast.2008.11.010).

Gilliland, M. 2010. *The Business Forecasting Deal*. Hoboken, NJ: Wiley.

Svolba, Gerhard. 2006. *Data Preparation for Analytics Using SAS®*. Cary, NC: SAS Institute Inc. Available http://www.sascommunity.org/wiki/Data_Preparation_for_Analytics.

Svolba, Gerhard. 2012. *Data Quality for Analytics Using SAS®*. Cary, NC: SAS Institute Inc. Available http://www.sascommunity.org/wiki/Data_Quality_for_Analytics.

APPENDIX

THE %CALC_REFERENCE_CATEGORY MACRO

Introduction

The %CALC_REFERENCE_CATEGORY macro enables you to calculate the “hidden” coefficients of the reference category in dummy coding when the EFFECT parameterization has been used.

When using the EFFECT parameterization, the coefficient of the reference group (the “missing coefficient”) can be calculated by summing the coefficients of the other categories and changing the sign. This is also referred to as the negative sum of the coefficients of the other categories.

Although it is technically possible to perform these calculations by hand, it is more convenient and efficient to use a program to do this automatically.

Prerequisites for the Macro

The macro has the following prerequisites.

- In the model, the EFFECT coding has been used for the creation of the dummy variables.
- The dummy variables have been automatically created using the CLASS statement. This is, for example, possible in the GLMSELECT and the QUANTSELECT procedure.
- Note that the DMREG procedure in SAS Enterprise Miner also creates dummy variables based on EFFECT coding. However, the macro has not been tested for the output of the DMREG procedure.
- The parameter estimates file contains the p-value for each parameter. This can be requested with the option SHOWPVALUES in the MODEL statement.
- The macro assumes that the input table that is used in the macro call has been created using the ODS OUTPUT statement in the respective regression procedure.
- The ODS objects PARAMETERESTIMATES and CLASSLEVELINFO can be created with the following ODS OUTPUT statement in the respective regression procedure:

```
ODS OUTPUT ParameterEstimates= ParameterEstimates
           ClassLevelInfo    = ClassLevelInfo;
```

Limitations of the Macro

The current version of the macro has the following limitations:

- The categories of the CLASS variables must not contain blanks. For example, a category value “Model 1” is invalid. It needs to be transformed, for example, to “Model1” or “Model_1” before the regression analysis is run.
- Note that the macro ignores interaction terms in the output table. The reason for this is that the effect names that are created from the interaction terms are often abbreviated and cannot be reproduced by the macro from the ClassLevelsList.

Macro Parameters

The following parameters can be specified with the macro.

- **ParmEst**: The name of the data set that contains the ParameterEstimates, created with the ODS OUTPUT statement. Default = ParameterEstimates.
- **ClassLevels**: The name of the data set that contains the ClassLevelInfo, created with the ODS OUTPUT statement. Default = ClassLevelInfo.
- **OutputDS**: The name of the data set that shall contain the output data set. Default = _ParmEst_XT_.

Refer to Svolba 2016, Chapter 12 for a comprehensive explanation of the macro and its functionality. The macros can be downloaded here: http://www.sascommunity.org/wiki/Data_Quality_for_Analytics_--_Download_Page.

ACKNOWLEDGMENTS

Many people have helped and inspired me to write and to complete this paper: Bob Rodriguez, Paul Goodwin, Mike Gilliland, Mihai Paunescu, Albert Tösch, and Robin Langford.

RECOMMENDED READING

Svolba, Gerhard. 2006. “Efficient ‘One-Row-per-Subject’ Data Mart Construction for Data Mining.” *Proceedings of the Thirty-First Annual SAS Users Group International Conference*. Cary, NC: SAS Institute Inc. Available http://www2.sas.com/proceedings/sugi31/078-31.pdf.

Svolba, Gerhard. 2015. “Want an Early Picture of the Data Quality Status of Your Analysis Data? SAS® Visual Analytics Shows You How.” *Proceedings of the SAS Global Forum 2015 Conference.* Cary, NC: SAS Institute Inc. Available http://support.sas.com/resources/papers/proceedings15/SAS1440-2015.pdf.

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author at:

Gerhard Svolba
SAS Institute Inc. Austria
Mariahilfer Strasse 116, A-1070 Wien
Email: mailto: Sastools.by.gerhard@gmx.net
Web: http://www.sascommunity.org/wiki/Gerhard_Svolba