Paper SAS3448-2019

Monitoring the Relevance of Predictors for a Model Over Time

Ming-Long Lam, Ph.D., SAS Institute Inc.

ABSTRACT

This paper presents a novel approach to monitor model performance over time. Instead of monitoring accuracy of prediction or conformity of predictors' marginal distributions, this approach watches for changes in the joint distribution of the predictors. Mathematically, the model predicted outcome is a function of the predictors' values. Therefore, the predicted outcomes contain intricate information about the joint distribution of the predictors. This paper proposes a simple metric that is coined the Feature Contribution Index. Computing this index requires only the predicted target values and the predictors' observed values. Thus, we can assess the health of a model as soon as the scores are available and raise our readiness for preemptive actions long before the target values are eventually observed. This index is model neutral because it works for any types of models that contain categorical or continuous predictors, and models that generate predicted values or probabilities. Models can be monitored in near real time since the index is computed using simple and time-matured algorithms that can be run in parallel. Finally, it is possible to provide statistical control limits on the index. These limits help foretell whether a particular predictor is a plausible culprit in causing the deterioration of a model's performance over time.

INTRODUCTION

In today's intelligence-driven economy, corporations increasingly rely on analytic models to make their business decisions. Like all tangible assets, models become dated, and their accuracies diminish over time. To stay competitive, corporations constantly monitor their models. When signs of deterioration of model performance appear, stakeholders need to determine if the models must be proactively updated or rebuilt to correct the problems. Since every decision to refresh a model carries risks and can disrupt normal business, a solid business case must be presented to support the request to update or rebuild a model.

Not all models can be monitored or are worth monitoring. In this paper, we focus on monitoring supervised learning models where there is one target variable. Most, if not all, model performance metrics have one thing in common: they measure how well the model predicted values agree with the observed target values. Various model performance metrics have been developed to measure the degree of this agreement. However, we sometimes need to assess the health of a model at the time of scoring when the target values are yet to be observed. If we must wait for the availability of the observed target values, then we might lose the opportunity to make a time-sensitive decision to refresh the model sooner. An example of this need is the fraud detection model. It is known that those who commit fraud game the system to avoid being detected. Since it takes time to diligently investigate fraud, we need some indicators to tell us now if the current system is being gamed. If we find that the system is no longer effective in detecting fraud, then countermeasures must be taken to correct the situation.

Although there are currently various model performance metrics to measure the overall performance of a predictive model, not all metrics are able to pinpoint which predictors might be responsible for the deterioration of model performance. In addition, some metrics are applicable only to certain types of models and computing the metrics might require us to rebuild the model.

This paper proposes a model performance metric coined the Feature Contribution Index. Only the predicted target values and the predictors' values are needed to calculate the index. Thus, we can assess the health of a model as soon as the model scores are available and take preemptive actions long before the target values are observed. This index is model neutral because it works for any types of models that contain categorical or continuous predictors, and models that generate predicted values or predicted probabilities. Models can be monitored in near real time since the index is computed using simple and robust algorithms that can be run in parallel. Finally, statistical control limits on the index can be provided. The statistical control limits can help determine whether a particular predictor is causing the deterioration of a model's health over time.

IDEA CONCEPTION

Loosely speaking, a supervised learning model is an algorithm that takes the values of predictors as inputs and computes the predicted value of the target variable as the output. The algorithm is constructed based on the assumptions made on the probability distribution of the data. The assumptions are the relationship between the target variable and the predictors, the covariance structure of the predictors, and the distribution of the target variable. In a linear regression model, for example, the relationship between the target variable and the predictors is linear, the covariance structure of the predictors is fixed, and the target variable follows a normal distribution that is parametrized by a mean and a constant variance. The mean of the target variable is the predicted value of the linear relationship.

When a supervised learning model does not perform, we mean that the model can no longer be used to describe the probability distribution of the current data. In other words, some assumptions made are no longer valid. Different metrics have been developed to check specific assumptions. For the linear regression model, the lack-of-fit test checks the linear relationship assumption. The test of homogeneity checks the constant target variable variance assumption. The Shapiro–Wilk test checks the normality assumption. In summary, checking the relationship between the target variable and the predictors and determining the probability distribution of the data assumptions requires the observed target values. When the target variable has yet to be observed, which is a common situation in applying the model to new data, these two groups of assumptions cannot be checked. However, we can still check for changes in the covariance structure of the predictors.

To compare the covariance structure of the predictors over time, we can use multivariate tests of equality of covariance matrices such as Box's M test. If we can put aside the argument of whether these tests can apply to our data (due to the assumptions requiring observed target values), these tests can be helpful. However, we often need to know which of the predictors have triggered the differences in the covariance structures in addition to simply knowing that the covariance structures have changed over time.

Let us study the linear regression model to generate ideas. Under this model, the predicted value of the i-th observation is $\hat{y}_i = b_0 + \sum_{r=1}^{k} b_r x_{ir}$ where k is the number of predictors, b_0 is the intercept, $b_1, \dots, b_k$ are the estimated regression coefficients, and $x_{i1}, \dots, x_{ik}$ are the values of the predictors in the i-th observation. Using the fact that $\bar{y} = b_0 + \sum_{r=1}^{k} b_r \bar{x}_r$, it can be shown that

$$(\hat{y}_i - \bar{y})(x_{is} - \bar{x}_s) = \sum_{r=1}^{k} b_r (x_{ir} - \bar{x}_r)(x_{is} - \bar{x}_s) \text{ for } s = 1, \dots, k.$$

Suppose n is the number of observations, then we have

$$\frac{1}{n-1} \sum_{i=1}^{n} (\hat{y}_i - \bar{y})(x_{is} - \bar{x}_s) = \sum_{r=1}^{k} b_r \left(\frac{1}{n-1} \sum_{i=1}^{n} (x_{ir} - \bar{x}_r)(x_{is} - \bar{x}_s) \right).$$

The left-hand side of the equation is the observed covariance between the predicted values and the values of the *s*-th predictor. The right-hand side of the equation is a linear combination of the observed covariances between the values of each predictor and the *s*-th predictor.

When we apply this linear regression model to new data, the estimated regression coefficients are considered fixed. If the covariances among the predictors of the new data are the same as that of the training data, then the covariances between the predicted values and the values of the predictors of the new data should also be the same as that of the training data. The contraposition of this condition says that if the covariances between the predicted values and the values of the predictors of the new data are different from that of the training data, then the covariances between the predicted values and the values of the predictors of the new data should also be different from that of the training data. Therefore, if we compare the covariances between the predicted values and the values of the predictors with that of the training data, then we might be able to tell if the covariance structures of the predictors have changed.

EXTENSION TO CATEGORICAL VARIABLES

Next, we will attempt to extend the idea in the last section to categorical target variables or predictors. Let's take on the categorical target variable first. If we directly apply the idea in the previous section to the predicted category of a categorical target variable, then we must choose some thresholds for the predicted probabilities of the target categories. Instead of running into the arguments of choosing the "right" thresholds, we will apply the above idea to the predicted probabilities. In other words, we will use the correlation between each of the predicted probabilities with each of the predictors as our metrics.

We cannot break a categorical predictor into its individual levels. Under this constraint, we need to look outside of correlation for our metric. The Eta-Square statistic measures the association between an interval variable and a categorical variable in a general linear model. When the general linear model has only one categorical predictor, the Eta-Square value is equal to the model's R-Square value. The model's R-Square value is the square of the correlation value when the general linear model has only one interval predictor. Therefore, to use the same metric for both categorical predictors and interval predictors, we adopt the R-Square value as our metric and coin it the Feature Contribution Index.

FEATURE CONTRIBUTION INDEX

For a classification model where the target variable is categorical, the model outcome consists of the predicted probabilities. For a regression model where the target variable is continuous, the model outcome is the predicted value. In both types of models, the model outcome consists of one or more numeric values. We measure individual predictors' contribution to model performance by using the following procedure:

1. For each numeric value in the model outcome, perform the main effect analysis of variance on each individual predictor.
2. Measure the contribution of this predictor by the R-square statistic. For a categorical predictor, this is the full Eta-Squared statistic. For an interval predictor, this is the squared Pearson correlation coefficient.
3. For a categorical target, aggregate the contribution indices calculated in step two for each individual predicted probability. The aggregation method is discussed in the next section

The Feature Contribution Index is a numeric value between zero and one inclusively. A value of one indicates that the predictor solely determines the model outcomes. A value of zero indicates that the predictor has no bearing on the model outcomes.

AGGREGATION FOR A CATEGORICAL TARGET

We calculate the Feature Contribution Index for each predicted probability in step two of the steps listed in the previous section. We want to come up with a single index for a categorical target since examining a single index is always more preferred than studying several indices for insights. This single index is a weighted sum of the individual Feature Contribution Indices. Without loss of generality, we require positive weights whose sum is one. We can consider two choices for weights. The first and the non-informative choice sets the weights equal to the reciprocal of the number of predicted probabilities. For example, the weights are 0.5 for a binary target. Another choice sets the weights equal to the observed relative frequencies (that is, proportions) of the target categories.

For a binary target variable, the choice of weight does not matter. Let p be the predicted probability of the event. Then 1 − p is the predicted probability of the non-event. For a predictor x, CORR(x, 1-p) = - CORR(x, p). Therefore, $(\text{CORR}(x, 1-p))^2 = (\text{CORR}(x, p))^2$. Both sides of the equation are the R-Square statistics of the Feature Contribution Indices for the non-event and the event respectively. Since they are equal, any weighted sum of them will result in the same results provided the weights are positive and sum to one.

STATISTICAL CONTROL LIMITS

Since our original goal is to monitor a model over time, we are more interested in studying the change or the trend of the R-Square statistics of each predictor than the R-square statistics themselves. In other words, we are more interested in the change of the R-Square statistics over time compared to a benchmark. An obvious choice for the benchmark is the R-Square statistics that are calculated on the training data.

Our hypothesis is that the predictors' covariance structure in each monitoring data is identical to that of the training data. Under this hypothesis, the R-Square statistics that are calculated on the monitoring data should be ideally the same as the R-Square statistics that are calculated on the training data. In practice, the R-Square statistics are different because of the usual random elements in observing the data. Our question is how much differences among the R-Square statistics can we tolerate before we drop the hypothesis? We address this question by constructing a confidence interval for the R-Square statistics at each time point. If we can agree that the R-square statistic is equal to the Eta-Square statistic for an interval predictor, then we can apply the interval inversion method (Kromrey and Bell 2010 and Steiger 2004) to construct a confidence interval for the Eta-Square statistic.

Let η_0^2 be the Eta-Square statistic (that is, the benchmark value) calculated on the training data that have n observations. The corresponding *F* value is

$F_0 = \eta_0^2/(1-\eta_0^2) \times df_2/df_1$ with two degrees of freedom, df_1 and $df_2 = n - 1 - df_1$.

For an interval predictor, $df_1 = 1$. For a categorical predictor, df_1 is equal to the number of categories of the predictor. Suppose the confidence interval will have $100p\%$ of coverage confidence, then, according to Kromrey and Bell (2010), the interval inversion method finds two values of the non-centrality parameter. One value is such that the observed *F* significance equals $(1 - p)/2$. Another value is such that the observed *F* significance equals $(1 + p)/2$. The FNONCT function in SAS® can calculate these two values of the non-centrality parameter for the *F* distribution. The function takes four arguments in the following order: the observed *F* value, the df_1 value, the df_2 value, and the desired *F* significance value. Since the FNONCT function uses a Newton-type algorithm to iteratively calculate a nonnegative non-centrality value, the function might return a missing value when the algorithm fails to converge. This might happen when the observed *F* value is relatively small.

Let NCP_ETA_L = FNONCT(F_0, df_1, df_2, (1 + p)/2) and NCP_ETA_U = FNONCT(F_0, df_1, df_2, (1 - p)/2). In a one-way analysis of variance, the non-centrality (NC) parameter of the F test is equal to the sum of squares of the model (SSM) divided by the mean squares error (MSE) (MSE = SSE / df_2). The Eta-Square statistic is equal to SSM / (SSM + SSE). Thus, the Eta-Square statistic is equal to NC / (NC + df_2). Using this simple relationship, the lower confidence limit for Eta-Square is NCP_ETA_L / (NCP_ETA_L + df_2) and the upper confidence limit for Eta-Square is NCP_ETA_U / (NCP_ETA_U + df_2). Please beware that the formula that we used is slightly different from that in Kromrey and Bell (2010). In their paper, Kromrey and Bell used the sample size n instead of df_2. Because of this difference, our control limits are slightly wider than the control limits proposed by Kromrey and Bell.

When we score the model on a monitoring data set, we assume that the predictors' covariance structure remains unchanged from that of the training data. Under this assumption, we would expect the *actual* Eta-Square values of the monitoring data are not different from the benchmark values. Therefore, we will use the η_0^2 for constructing the limits. The df_1 value stays the same despite the possibility of not observing all the categories of the categorical predictor (otherwise, we unknowingly changed the covariance structure). The df_2 value is equal to the number of observations in the monitoring data. Since we treat η_0^2, which is itself a random variable, as fixed benchmarks for the monitoring data, we should call the limits the control limits to avoid any statistical issues. This is because we cannot ensure that the coverage confidence is actually the nominal value 100p%.

Finally, if the Eta-Square statistics that are calculated on the monitoring data are outside the control limits, then we have reasons to suspect that the predictors' covariance structure might have changed.

SAS MACROS

The following three SAS macros were developed for calculating the Feature Contribution Index:

1. The Compute_FCI_NomPred macro computes the Feature Contribution Indices for a list of categorical predictors.
2. The Compute_FCI_IntPred macro computes the Feature Contribution Indices for a list of interval predictors.
3. The Compute_FCI macro reads input specifications, calls the Compute_FCI_NomPred and the Compute_FCI_IntPred macros to compute the Feature Contribution Indices, and returns the indices in the specified data.

These macros require that the model outcomes are already available in the input monitoring data. You can download these macros from https://support.sas.com/downloads/package.htm?pid=2225 and the accompanying documentation from http://support.sas.com/documentation/prod-p/mdlmgr/14.2/en/PDF/SMM142_FCI_Macros.pdf.

A fourth macro, Create_FCI_Report, was later developed for facilitating the entire process. It bypasses the Compute_FCI macro and directly calls the Compute_FCI_NomPred macro and the Compute_FCI_IntPred macro. In addition, it generates professionally formatted tables and charts. In the future, the Create_FCI_Report macro will be available for download and publish. In the meantime, interested readers can contact the author directly to obtain the Create_FCI_Report macro.

SIMULATION STUDY

After we determine that the predictors' covariance structure has changed over time, our next task is to determine which of the predictors have triggered the change. This task is more difficult than it looks.

Let us use the linear regression example to aid our discussion. Recall that the covariance between a predictor and the predicted value is a linear combination of the individual covariances between two predictors. Mathematically, this is

$$\frac{1}{n-1}\sum_{i=1}^{n}(\hat{y}_i - \bar{y})(x_{is} - \bar{x}_s) = \sum_{r=1}^{k} b_r \left(\frac{1}{n-1}\sum_{i=1}^{n}(x_{ir} - \bar{x}_r)(x_{is} - \bar{x}_s)\right).$$

Without loss of generality, let us consider the scenario where only the covariance between the first two predictors has changed but not their individual variances. This change will affect the covariance between the first predictor and the predicted value. Convoluted with the sign of the regression coefficient of the first predictor in the benchmark model, the covariance between the first predictor and the predicted value will increase or decrease. Similarly, the covariance between the second predictor and the predicted value will increase or decrease. When we notice that only the Feature Contribution Indices of a pair of predictors are outside the control limits, we might conclude that the covariance between that pair of predictors has changed.

In another scenario, the variance of the first predictor has changed. In other words, the distance between $x_{i1} - \bar{x}_1$ has changed for $i = 1, \ldots, n$. This change will affect all the covariances that involve the first predictor. Since either $r = 1$ or $s = 1$, all the covariances between a predictor and the predicted value will be affected. When we notice that all the Feature Contribution Indices are outside the control limits, we need to compare the predictors' variance and their covariances with the benchmark value. An inadequacy of the Feature Contribution Index is the scenario where we do not come to any conclusions after studying these comparisons.

In our final scenario, only the mean of the first predictor has changed. This will not affect any covariances. Thus, we will not see any Feature Contribution Indices outside the control limits. If we are not concerned about the shifts of any means, then this is a good feature of the Feature Contribution Index as we have one thing less to check. Otherwise, this is another inadequacy of the Feature Contribution Index and we must turn to other diagnostic methods instead.

We are going to use a simulation study to illustrate the above three scenarios, demonstrate the steps of calling the macros, and review the results. The predictors are, namely, X1 and X2. The target variable is y whose expectation is E(Y) = -3 + 4 * X1 + 2 * X2.

A normal random noise with zero mean and unit variance is added to E(Y) to obtain the observed Y. In the training data, which contains 1000 observations, the predictors have zero means, unit variances, and are uncorrelated (that is, zero correlations). The ordinary least squares estimate of the model is $\hat{y}$ = -2.9762 + 4.0061 * X1 + 1.9721 * X2.

SCENARIO 1: CHANGE CORRELATION

Seven monitoring data sets, each containing 100 observations, are simulated. The predictors' covariance structures are the same as that in the training data except for the correlation between the two predictors. Here are the changes to the correlations that are simulated in the five monitoring data sets:

1. The correlation between X1 and X2 is 0.0, i.e., no change.
2. The correlation between X1 and X2 is -0.4.

3. The correlation between X1 and X2 is -0.2.
4. The correlation between X1 and X2 is +0.2.
5. The correlation between X1 and X2 is +0.4.

After we score the data sets, our first instinct is to compare the distributions of the predicted values with that of the benchmark training data (the ID column). The box-plots in Figure 1 help us visualize the comparison. At a first glance, although the distributions have different ranges, they are visually indifferent based on metrics such as the means and the medians. If we compare their interquartile ranges, then we may suspect that the distributions in the second and the fifth monitoring data sets have changed because their interquartile ranges are shorter than others. Therefore, visually comparing distributions might not enable us to realize that the predictors' covariance structures have changed.

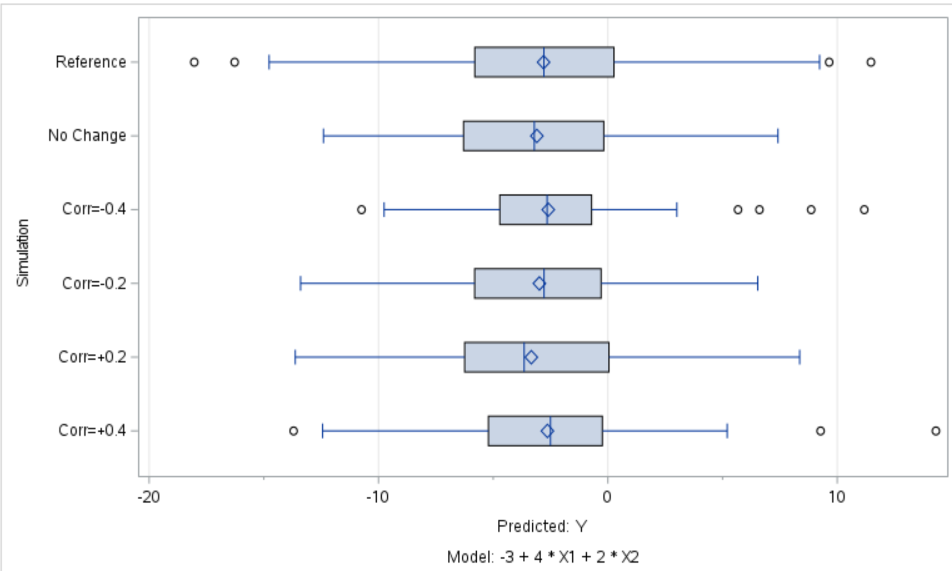

Figure 1. Box Plots of Predicted Values Across Monitoring Data Sets for Scenario 1

The panel chart in Figure 2 shows the Feature Contribution Indices of the predictors across the monitoring data sets. The benchmark index is subtracted from the indices and the control limits so that all the graphs are drawn using the same scale. Therefore, the vertical axis tells the change from the benchmark index. Finally, the graphs are shown in descending level of the benchmark indices (the value of the level is inside the parentheses of the individual chart titles). This enables us to focus on the predictors that contribute more to the model outcome in the benchmark data.

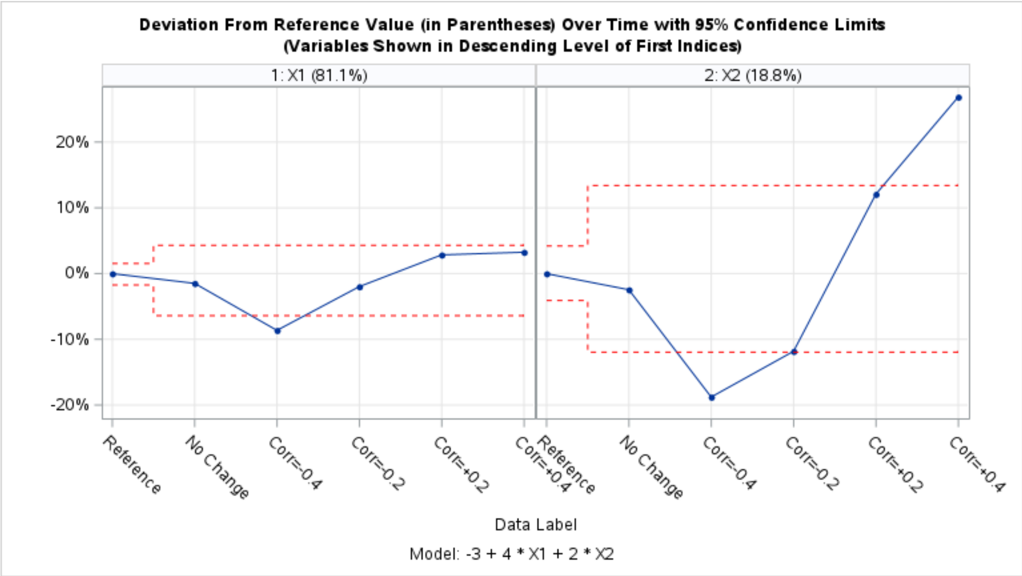

Figure 2. Feature Contribution Indices of Predictors Across Monitoring Data Sets for Scenario 1

Table 1 shows the Feature Contribution Indices of the five monitoring data sets and the benchmark training data (the Reference column). The out-of-bound values are highlighted in red.

Table 1. Feature Contribution Indices of Predictors for Scenario 1

Predictor	Reference	No Change	Corr=-0.4	Corr=-0.2	Corr=0.2	Corr=0.4
X1	81.1%	79.6%	72.5%	79.1%	84.0%	84.4%
X2	18.8%	16.3%	0.0%	7.9%	30.9%	45.6%

Ideally, we want to see that the indices of both perturbed predictors in each monitoring data set are outside the control limit (for example, X1 and X2 in the fourth monitoring data set that is labeled Corr=0.2). Since the monitoring data has only 100 observations, which is one-tenth the number of observations in the training data, the ideal results may not occur in every scenario unless the correlations become apparently stronger.

SCENARIO 2: CHANGE STANDARD DEVIATION

Four monitoring data sets, each containing 100 observations, are simulated. The predictors' covariance structures are the same as that in the training data except for the standard deviations of the predictors. Here are the changes to the standard deviation that are simulated in the four monitoring data sets:

1. The standard deviations of X1 and X2 are 1, i.e., No Change.
2. The standard deviation of X1 is 0.8 and that of X2 is 1.
3. The standard deviation of X1 is 1 and that of X2 is 1.25.
4. The standard deviation of X1 is 0.8 and that of X2 is 1.25.

A comparison of the distributions shows more distinct differences than what was found in the first scenario (Figure 3). The ranges and the interquartile ranges are visually different. However, the medians are seemingly the same.

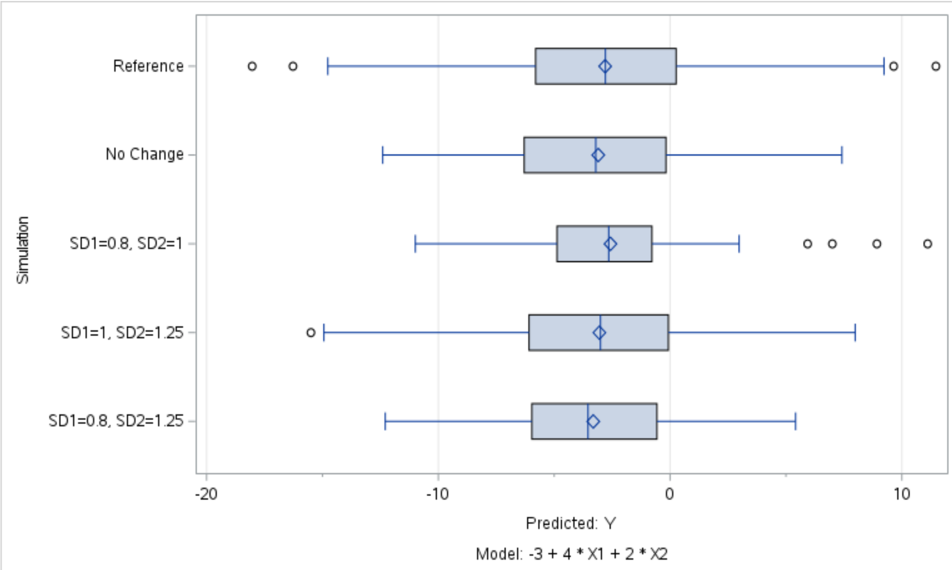

Figure 3. Box Plots of Predicted Values Across Monitoring Data Sets for Scenario 2

The panel chart in Figure 4 shows the Feature Contribution Indices of the predictors across the four monitoring data sets. Since X1 has the highest benchmark index, a change in the standard deviation of any predictor (including X1) in the monitoring data set will trigger a stronger ripple effect on the X1's index. On the contrary, X2 has the lowest benchmark index, only a change in its own standard deviation plus another change of X1's standard deviation in the monitoring data set can affect its index.

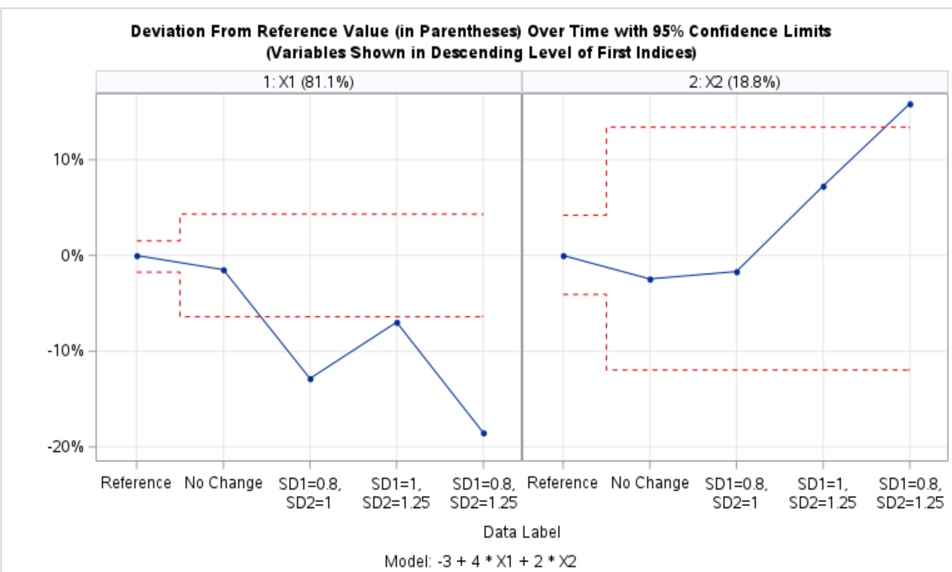

Figure 4. Feature Contribution Indices of Predictors Across Monitoring Data Sets for Scenario 2

Table 2 shows the Feature Contribution Indices of the four monitoring data sets and the benchmark training data (the Reference column). The out-of-bound values are highlighted in red.

Table 2. Feature Contribution Indices of Predictors for Scenario 2

Predictor	Reference	No Change	SD1 = 0.8 & SD2 = 1.0	SD1 = 1.0 & SD2 = 1.25	SD1 = 0.8 & SD2 = 1.25
X1	81.1%	79.6%	68.3%	74.1%	62.6%
X2	18.8%	16.3%	17.1%	26.0%	34.6%

Although we have not proved this fact mathematically, we notice that changing the standard deviation of a predictor will drastically magnify or shrink its Feature Contribution Index. For example, changing only the standard deviation of X2 from 1 to 1.25 in the third monitoring data set will magnify its index approximately 1.4 times (from 18.8% to 26.0%). The magnitudes of the changes might be even bigger when the standard deviations of other predictors also change.

SCENARIO 3: CHANGE THE MEAN

Four monitoring data sets, each containing 100 observations, are simulated. The predictors' covariance structures are the same as that in the training data except for the means of the predictors. Here are the simulated changes to the mean in the four monitoring data sets:

1. The means of X1 and X2 are both 0, i.e., No Change.
2. The mean of X1 is -10 and that of X2 is 0.
3. The mean of X1 is 0 and that of X2 is 20.
4. The mean of X1 is -10 and that of X2 is 20.

The box-plots in Figure 5 show that the distributions have very different medians, but the ranges are similar. This is expected since we changed only the means of the predictors but not their covariance structures.

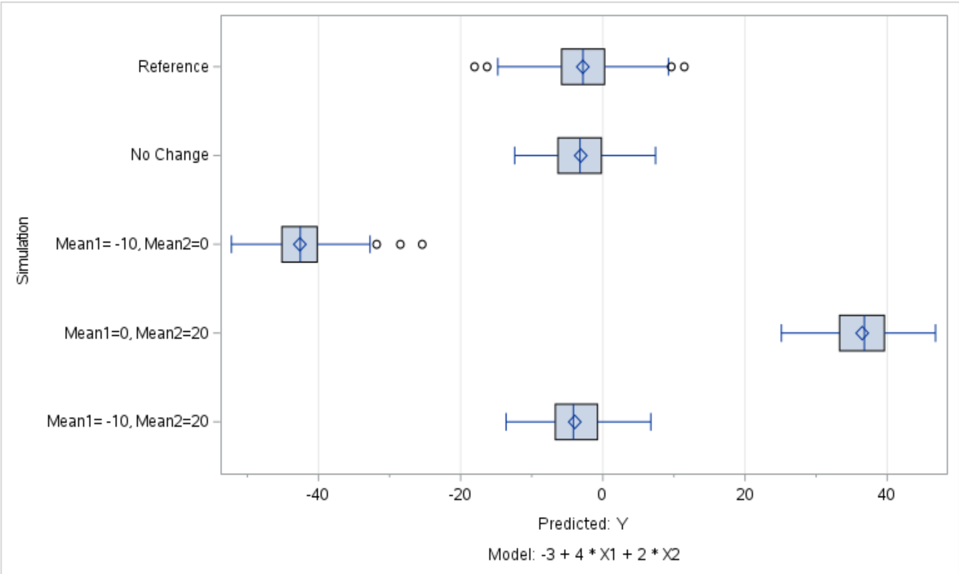

Figure 5. Box Plots of Predicted Values Across Monitoring Data Sets for Scenario 3

The panel chart in Figure 6 shows the indices of the predictors across the data sets. All the indices are within the control limits. We do expect this because the indices are designed to detect changes in the covariance structures (including standard deviations and correlations), but not changes in the means.

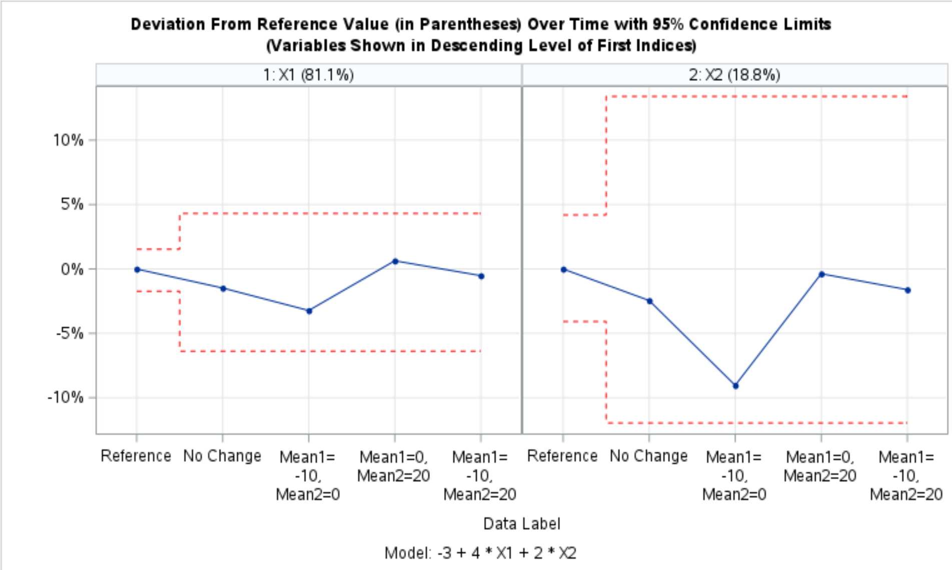

Figure 6. Feature Contribution Indices of Predictors Across Monitoring Data Sets for Scenario 3

Table 3 shows the Feature Contribution Indices of the four monitoring data sets and the benchmark training data (the Reference column). The out-of-bound values, if any, are highlighted in red.

Table 3. Feature Contribution Indices of Predictors for Scenario 3

Predictor	Reference	No Change	Mean1 = -10 & Means = 0	Mean1 = 0 & Means = 20	Mean1 = -10 & Means = 20
X1	81.1%	79.6%	77.9%	81.8%	80.6%
X2	18.8%	16.3%	9.7%	18.4%	17.2%

ANALYSIS EXAMPLE

Finally, we will illustrate our method using a real-life data set. The data in this example describes the historical usage patterns along with the weather data about the bike rental demand in the Capital Bikeshare program in Washington, D.C. The data is available on the Kaggle site[1]. The original data was provided by Fanaee-T and Gama (2014).

For the sake of discussion, we are going to build a Poisson regression model to predict the total number of rentals (that is, count). The data originally covers 10,886 rental records dated from January 1, 2011 to December 31, 2012. We create the training data and four monitoring data sets based on the rental dates. The training data consists of all rentals in 2011 and it has 5,422 observations. The first monitoring data set consists of rentals in the first quarter of 2012 (that is, January to March) and has 1,363 observations. The second monitoring data set consists of rentals in the second quarter of 2012 (that is, April to June) and has 1,366 observations. The third monitoring data set consists of rentals from the third quarter of 2012 and it has 1,368 observations. Finally, the fourth monitoring data set consists of rentals from the fourth quarter of 2012 and has 1,367 observations.

A few categorical variables are created so that the Poisson regression model is more predictive. For example, the rental_hour_group variable is created by grouping values of the rental_hour variable. The grouping is determined mostly due to business reason. The

[1] https://www.kaggle.com/c/bike-sharing-demand.

Poisson regression algorithm converged. Plotting the predicted counts versus the observed counts assure us that the model fits the data well (Figure 7). Thus, we will use this model result for our discussion.

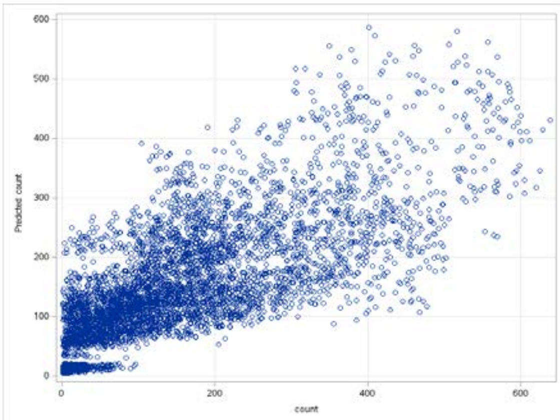

Figure 7. Predicted Counts Versus the Observed Counts of the Poisson Regression Model

Table 4 shows the parameter estimates of the Poisson regression model. The type III likelihood ratio tests of all the predictors are significant at the 0.01% level.

Table 4. Parameter Estimates of the Poisson Regression Model

Measurement Level	Parameter	Level	DF	Estimate
	Intercept		1	3.9253
Nominal	holiday	0	0	0
		1	1	0.0539
	rental_weekday	1	1	-0.0270
		2	1	-0.0665
		3	1	-0.0562
		4	1	-0.1220
		5	1	-0.0518
		6	1	0.0144
		7	0	0
	rental_hour_group	2AM - 5AM	1	-1.9459
		6AM - 8AM	1	0.6220
		9AM - 11AM	1	0.4495
		12NOON - 4PM	1	0.6151
		5PM - 7PM	1	1.0955
		8PM - 1AM	0	0
	weather	1	0	0
		2	1	-0.0602
		3	1	-0.4314
Interval	temp		1	0.0406
	humidity		1	-0.0014
	windspeed		1	-0.0037

Next, we will apply this model to the four monitoring data sets. Figure 8 shows the Feature Contribution Indices of all predictors in the training data and the four monitoring data sets. Overall, there are no drastic changes among the indices. The more noticeable changes are at the humidity (the relative humidity), the season (the season indicator), and the temp (the hourly temperature) predictors. Since the monitoring data sets are characterized by the rental dates (for example, in the first monitoring data set, season equals 1 for all observations, and humidity and temp varies within the winter-characterized ranges), the spreads of the humidity, the season, and the temp predictors in a monitoring data might be narrower than that in the benchmark data.

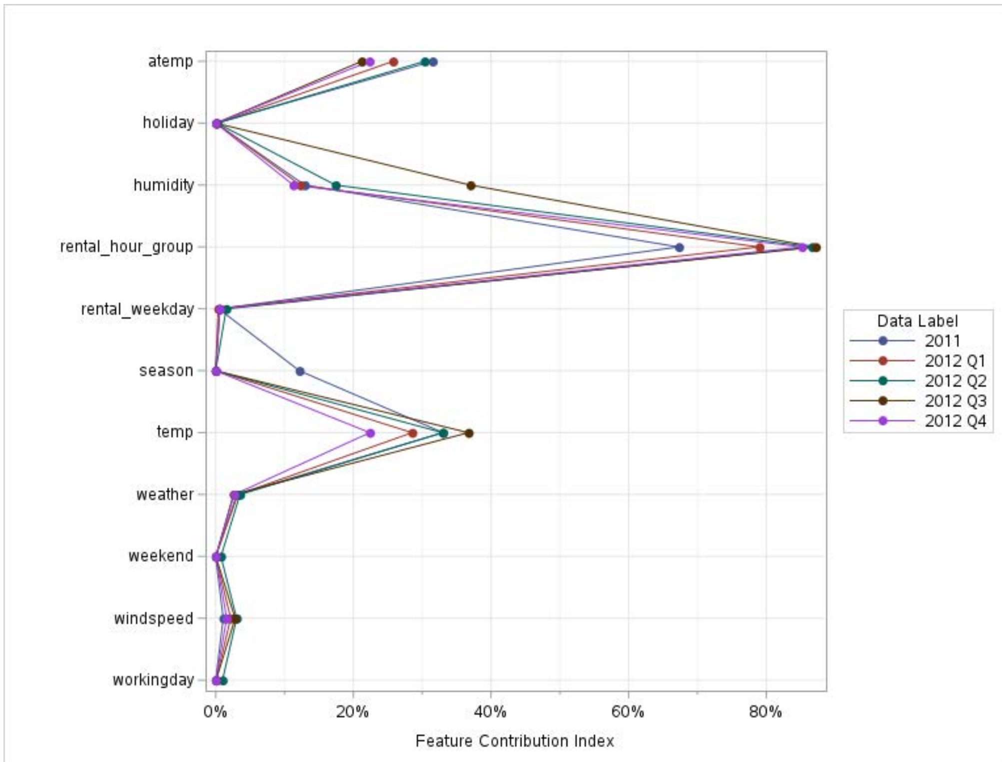

Figure 8. Feature Contribution Indices of the Monitoring Data for the Analysis Example

We will next review the Feature Contribution Indices of the predictors individually (Figure 9). You should notice that the weekend predictor does not have control limits. The LOG messages show that the FNONCT functions ran into computational problems and could not return values. Since the Feature Contribution Index of the weekend predictor is almost zero in the benchmark training data, we do anticipate this problem. Therefore, we can safely put this undesirable result aside.

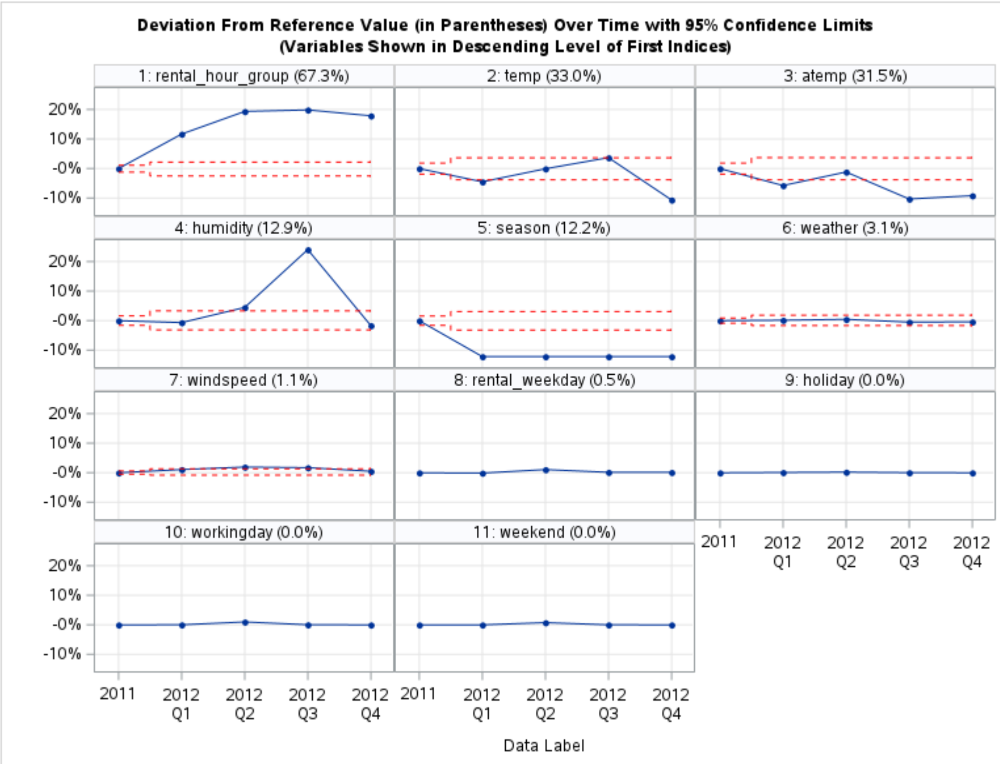

Figure 9. Feature Contribution Indices of Predictors Across Monitoring Data Sets for the Analysis Example

Table 5 shows the Feature Contribution Indices of the four monitoring data sets and the benchmark training data (the ID column). The out-of-bound values are highlighted in red.

Table 5. Feature Contribution Indices of the Predictors for the Analysis Example

Measurement Level	Predictor	2011	2012 Q1	2012 Q2	2012 Q3	2012 Q4
Interval	atemp	31.5%	25.8%	30.4%	21.2%	22.3%
	humidity	12.9%	12.3%	17.4%	37.0%	11.3%
	temp	33.0%	28.6%	33.0%	36.7%	22.4%
	windspeed	1.1%	2.2%	3.0%	2.8%	1.6%
Nominal	holiday	0.0%	0.1%	0.2%	0.1%	0.0%
	rental_hour_group	67.3%	79.0%	86.7%	87.2%	85.2%
	rental_weekday	0.5%	0.4%	1.5%	0.6%	0.6%
	season	12.2%	0.0%	0.0%	0.0%	0.0%
	weather	3.1%	3.2%	3.5%	2.6%	2.7%
	weekend	0.0%	0.0%	0.8%	0.0%	0.0%
	workingday	0.0%	0.0%	1.0%	0.1%	0.0%

The most obvious change occurs in the rental_hour_group predictor. The indices of other interval predictors that describe the climate of the quarters showed some changes. Our common sense tells us that these interval predictors are correlated, and their covariance structures do depend on the quarters. Figure 10 shows the association structure between the rental_hour_group and the humidity predictors by quarter.

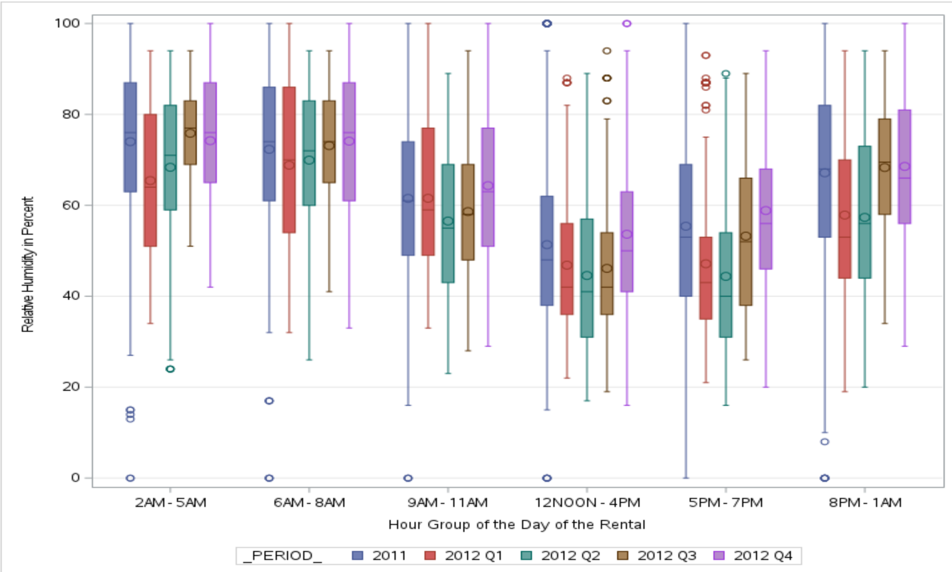

Figure 10. Association Structure Between the rental_hour_group and the humidity Predictors by Quarter

At first glance, the humidity predictor has a smaller range in the quarterly data than in the training data. In addition, the rental hour group seems to affect the interquartile ranges. For example, the interquartile ranges during 5PM – 7PM of 2012 Q1 overlaps the least with that of 2012 Q4. During other times of day, the interquartile range of 2012 Q1 overlaps more with that of 2012 Q4.

Finally, the Feature Contribution Indices of the season drop to zero in the four monitoring data sets. This is no surprise because the season predictor is practically constant in each monitoring data set. Thus, it has no relevance to the model outcome except to raise or lower the overall mean of the predicted rental counts.

CONCLUSION

We have introduced the Feature Contribution Index and we attempted to interpret the meanings of the index using a few simulated studies and the Bike Share Demand data. The Feature Contribution Index idea has plenty of room for improvement as we are not fully able to make conclusions based on their values. We welcome others to join us in further studying, improving, and interpreting the Feature Contribution Index.

REFERENCES

Fanaee-T, H., and Gama, J. (2014). "Event labeling combining ensemble detectors and background knowledge", *Progress in Artificial Intelligence,* 2(2-3), 113-117, Berlin Heidelberg, Germany: Springer-Verlag. https://doi.org/10.1007/s13748-013-0040-3.

Kromrey, J. D., and Bell, B. A. (2010). "ES_ANOVA: A SAS Macro for Computing Point and Interval Estimates of Effect Sizes Associated with Analysis of Variance Models," *Proceedings of the SouthEast SAS Users Group Conference (SESUG 2010)*, paper PO-05. Cary, NC: SAS Institute Inc. https://analytics.ncsu.edu/sesug/2010/PO05.Kromrey.pdf.

Steiger, J. H. (2004). "Beyond the *F* Test: Effect Size Confidence Intervals and Tests of Close Fit in the Analysis of Variance and Contrast Analysis," *Psychological Methods*, 9(2), 164-182. http://dx.doi.org/10.1037/1082-989X.9.2.164.

CONTACT INFORMATION

Your comments and questions are welcomed, encouraged, and valued. In addition, you can request the SAS codes that generate the above results. Please contact the author at:

Ming-Long Lam
SAS Institute Inc.
(312) 995-5865
ming-long.lam@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.