



SOCIETY OF ACTUARIES

Article from:

The Financial Reporter

June 2008 – Issue No. 73

PBA Reserves and Capital Modeling Efficiency: Representative Scenarios and Predictive Modeling

by Steven Craighead

In modeling principle-based approaches for reserves and capital, it is necessary to conduct extensive scenario testing on each business model. The resultant scenarios are then used in specific ways to determine the proper reserves and impact on overall company capital. This is especially critical for new types of insurance policies with complex options. However, it is also our desire to reduce the computer run-time required to obtain these reserve or capital values. In the past this has been typically done by the use of representative scenarios, where, based on how many scenarios map to each representative, we determine a probability weight associated with each. However, given the relatively simple historically used approaches to the mapping process, the weights obtained may not accurately reflect the character of the scenarios mapped.

If, however, we are able to use representative scenarios to train a separate smaller model to replicate the full business model, we could then use all the scenarios within this less time-expensive model and not use the probability weights at all. In other words, we wish to develop a new technique which combines the use of both representative scenarios with that of predictive modeling.

In the study below we consider two ways of determining representative scenarios, and then we use the results of each in combination with a very effective statistical tool to create predictive models. Although we still have to process our time-expensive business model on the representative scenarios, and separately calibrate the time-inexpensive predictive model, we have found that the results are very good. We conclude that either method of choosing scenarios is effective for reserve calculations, but that one method is superior in estimating capital.

I will briefly discuss the two methods used to select representative scenarios.

Representative Scenarios

There are several (actuarially) published as well as commonly known methods to determine representative scenarios from a larger collection. In our

research we will use one published method and one commonly used by statisticians. We will not introduce any weighting within our scenario selection process, and we will treat the selection process as directly formulated by the sources; however, later we will discuss the use of weighting in the control of bias.

(1) Chueh¹ describes three separate algorithms to select representative scenarios. In particular, the second algorithm uses the following metric to create a distance between two separate interest rate scenarios paths:

$$D_2 = \sqrt{\sum_{t=1}^{30} \left(\prod_{k=1}^t \frac{1}{1+i_k} - \prod_{k=1}^t \frac{1}{1+i_k^P} \right)^2}$$

Here i_t is a one-year rate at time t . i_t^P is the pivot interest rate at time t .

Notice that this is ultimately a sum of squares metric with the interest rates represented as discount factors through time. In her paper, she observed that this metric gave a good overall representation of results both in the center and the tails. We will only use this metric in our analysis below because the required conditional tail expectation (CTE) calculations for reserves is set at 65 percent (CTE65) and capital is set at 90 percent (CTE90). To properly calculate CTE65, we need to have the stochastic results well represented in both the central portion as well as the tail, whereas CTE90's calculation would need scenarios primarily within the tail.

(2) The other algorithm² that is frequently used by statisticians is the CLARA Cluster Algorithm. The CLARA algorithm can either use a sum of squares or a sum of absolute values metric to measure distance. In our work below we will use a sum of absolute



Steven Craighead, ASA, MAAA, CERA, is an actuarial consultant with Towers Perrin. He can be contacted at steven.craighead@towersperrin.com.

continued on page 18 >>

values to indicate distance between separate scenarios. Now let's look briefly at predictive modeling.

Predictive Modeling

Predictive modeling is a means that one can take to design or create a model that can be used to predict an outcome with approximately the same probability that is observed with the actual data. There are many different techniques, but while working independently with the Academy Valuation Basis Table subcommittee, we found one outstanding process. This modeling technique is called Projection Pursuit Regression³ (PPR). See the accompanying PPR appendix, which summarizes the process.

Combination of Methods

In the past, most actuarial research concentrated only on the use of representative scenarios and weighting the results based on the probabilities associated with each representative. We have found that this approach alone does not adequately represent the overall behavior that one obtains when using all of the scenarios. However, the goal of model efficiency is to reduce the entire processing time of the various reserve or capital models. *This has been mostly done in the past by either reducing the number of model points used with the liabilities or assets or by reducing the number of scenarios processed through the model.*

Independently, we have observed that PPR models are very effective, not overly sensitive to outliers within the calibration data, and replicate the overall behavior of high dimensional models well. Another nice feature of PPR is that it is very quick when asked to evaluate additional input besides that of its training data.

In past experience, we have also observed that the CLARA algorithm is very effective in selecting representative scenarios. This is due to the fact that the process discovers a majority of the extreme scenarios, which contribute to the tail of the reserve or the capital distribution.

When one uses representative scenarios and then uses probability weighting of results, the final results are very dependent upon how those weights are obtained or used. However, if one does not use these weights at all, but only uses the representatives as training data for a predictive model, we can then process all scenarios through the resulting predictive model. For this to work well, we hope that the number of representative scenarios will be rich enough to adequately span the high dimensional business

model. Also, we hope that the predictive model will also adequately model the business model as well.

We will now test the hybrid approach of using representative scenarios as training data and then processing all the scenarios through the predictive model.

Next, we briefly discuss what data we use in our analysis.

Data Sources

Craighead⁴ describes and models from over 100 insurance-related datasets. In our work below, we will concentrate on his 1993 dataset associated with business model 4 and the associated 10,000 interest rate scenarios that were used in the generation of these values. Craighead discusses the generation process of these scenarios as well. We have restricted ourselves to this specific dataset because it was determined within that this specific data set has such complex behavior, that if one is able to adequately model the underlying data, the remaining datasets are very easily modeled.

Now let's discuss how we will conduct our experiments.

Process Outline

Using the basic information of the 10,000 scenarios and the 10,000 associated capital values mentioned above, we conduct 100 separate experiments using random samples of 5,000 scenarios for each representative set size. On each of these scenario sets, we apply either the Chueh Algorithm (with a specific modification discussed below) or the CLARA Algorithm and choose separate representative subsets. Once a specific representative set is selected, the representative scenarios, in addition to their associated capital values, are used as the training data for a PPR model. Once the PPR model is trained (or calibrated), the entire sample of 5,000 scenarios is then projected using the resultant PPR model. Using the PPR model, we calculate both the CTE65 and CTE90 (we refer to these as Model). Also, based on the specific sample of 5,000 capital values, we also calculate the CTE65 and CTE90, (we refer to these as Actual). We then calculate the relative error associated between Actual and Model, by the formula: $RE = (Actual - Model)/Actual$.

In our analysis with the Chueh Algorithm, we modify the distance formula to include not only the 90-day rates but also the 10-year rates, by the following formula:

$$D_2 = \sqrt{\sum_{t=0}^{20} \left(\prod_{k=0}^t \frac{1}{1+i_{90k}} - \prod_{k=0}^t \frac{1}{1+i_{90k}^P} \right)^2}$$

$$+ \sum_{t=0}^{20} \left(\prod_{k=0}^t \frac{1}{1+i_{10k}} - \prod_{k=0}^t \frac{1}{1+i_{10k}^P} \right)^2$$

Where the CLARA algorithm uses the following distance formula:

$$\sum_{t=0}^{20} (|i_{90:t} - i_{90:t}^P| + |i_{10:t} - i_{10:t}^P|)$$

Note: We use the 90:t or 10:k notation to indicate 90-day rates or the 10-year rates in year t or k.

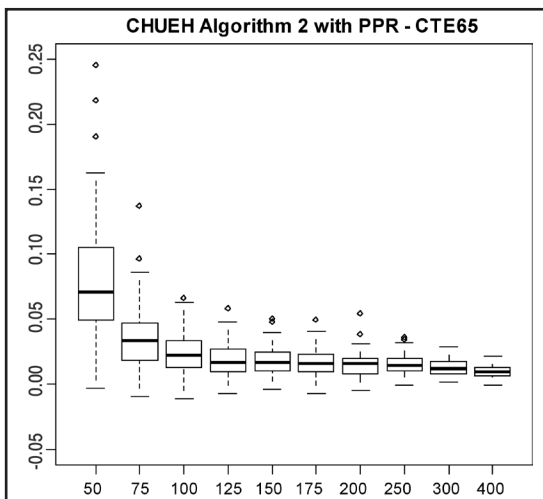
Results

Regarding the Chueh Algorithm experiments, we examine separate sets of representative scenarios. The various representative set sizes are 50, 75, 100, 125, 150, 175, 200, 250, 300 and 400.

Regarding, the CLARA Algorithm, we use representative set sizes of 50, 75, 125, 175, 200, 250, 300, 400 and 500.

As mentioned before, for each of these representative set sizes, we repeat the random sampling of 5,000 scenarios 100 times. By conducting this repeated sampling we can observe the effectiveness of the overall process and approximate the sample error associated in our tests.

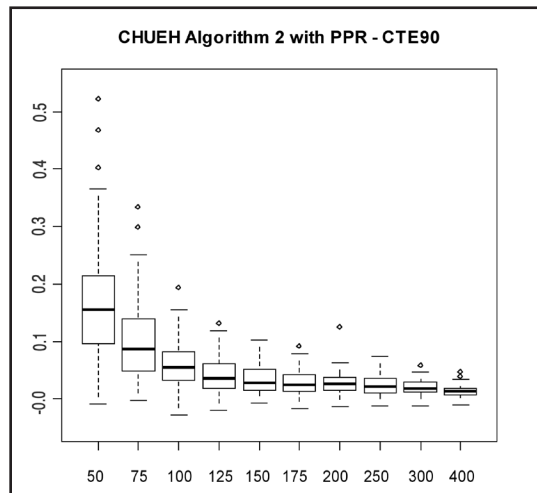
The next two box-whisker graphs as the results of the experiments associated with the Chueh distance. Following each graph are the associated statistics for each set of 100 tests. See the Wikipedia⁵ discussion on how to interpret box-whisker plots.



| | Min. | 1st Qu. | Median | Mean | 3rd Qu. | Max. |
|-----|----------|----------|----------|----------|---------|---------|
| 50 | -0.00351 | 0.0497 | 0.0709 | 0.07809 | 0.1041 | 0.2453 |
| 75 | -0.00928 | 0.01813 | 0.03358 | 0.03522 | 0.04689 | 0.1372 |
| 100 | -0.01093 | 0.01309 | 0.022 | 0.02356 | 0.03324 | 0.06634 |
| 125 | -0.00691 | 0.009314 | 0.01696 | 0.0184 | 0.02673 | 0.05787 |
| 150 | -0.00395 | 0.01067 | 0.01695 | 0.01802 | 0.02449 | 0.05027 |
| 175 | -0.00752 | 0.009906 | 0.01561 | 0.01607 | 0.02322 | 0.04944 |
| 200 | -0.00456 | 0.008315 | 0.01554 | 0.01497 | 0.02025 | 0.05378 |
| 250 | -0.00115 | 0.009923 | 0.01409 | 0.0148 | 0.01956 | 0.03615 |
| 300 | 0.001184 | 0.008013 | 0.0117 | 0.01257 | 0.01735 | 0.02898 |
| 400 | -0.00084 | 0.005982 | 0.009175 | 0.009458 | 0.01271 | 0.02113 |

Notice how the median (the dark heavy) line moves down as the number of representative scenarios increase. Notice also how the relative errors remain mostly positive. This indicates that the Actual CTE65 values are larger than the Model CTE65 values. This indicates that this approach is liberally biased, by an average of 94bp for the 400 representative samples. Notice as the number of representatives increases, how both the box and separately the whiskers narrows around the median.

Now observe the Chueh CTE90 results:



| | Min. | 1st Qu. | Median | Mean | 3rd Qu. | Max. |
|-----|----------|----------|---------|---------|---------|---------|
| 50 | -0.00876 | 0.09702 | 0.1555 | 0.1663 | 0.213 | 0.5213 |
| 75 | -0.00218 | 0.04898 | 0.08644 | 0.09739 | 0.1393 | 0.3344 |
| 100 | -0.02871 | 0.03181 | 0.05445 | 0.05984 | 0.08232 | 0.193 |
| 125 | -0.01989 | 0.01837 | 0.03506 | 0.041 | 0.06137 | 0.1316 |
| 150 | -0.00775 | 0.01512 | 0.02786 | 0.03508 | 0.05139 | 0.1021 |
| 175 | -0.01698 | 0.01273 | 0.02499 | 0.02686 | 0.04207 | 0.09204 |
| 200 | -0.01434 | 0.015 | 0.02569 | 0.02674 | 0.03762 | 0.1256 |
| 250 | -0.01233 | 0.01032 | 0.0212 | 0.02454 | 0.03543 | 0.07377 |
| 300 | -0.01203 | 0.01183 | 0.01737 | 0.01974 | 0.02955 | 0.05846 |
| 400 | -0.01069 | 0.006189 | 0.01342 | 0.01289 | 0.01795 | 0.04684 |

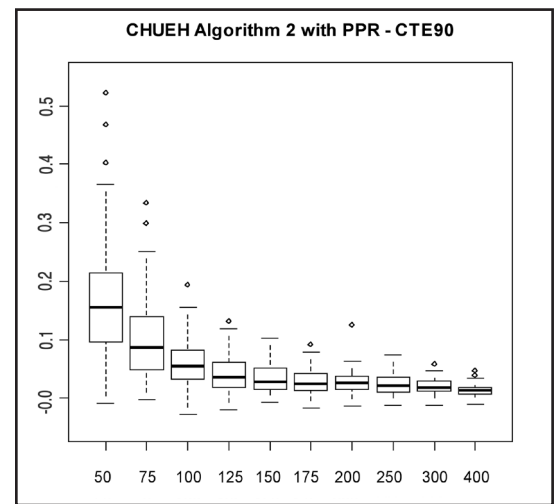
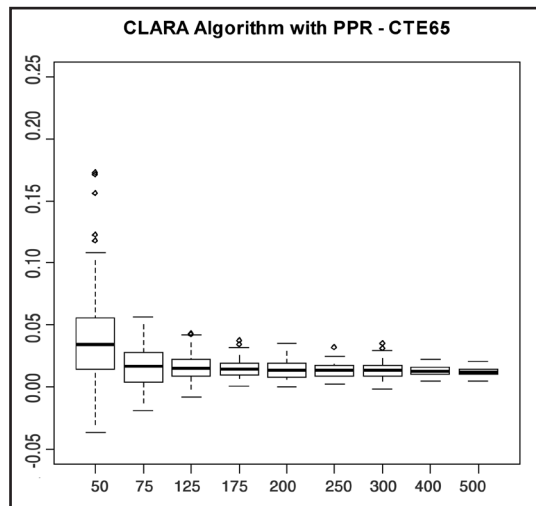
continued on page 20 >>

Notice how the collective CTE90 results are not as good as the collective CTE65 results, where using 400 representatives the median relative error is 134 bp versus that of 92 bp. Also observe the absolute distance from the maximum and the minimum. Again, note that these results are so liberally biased. Note also how the small circles above the top whiskers indicate that there are two outliers in the plot of the 400 representatives.

around the median that that of the Chueh. Now the CTE90 results:

| | Min. | 1st Qu. | Median | Mean | 3rd Qu. | Max. |
|-----|----------|----------|----------|----------|----------|---------|
| 50 | -0.03344 | 0.04874 | 0.08013 | 0.1027 | 0.1484 | 0.4273 |
| 75 | -0.03633 | 0.002239 | 0.02523 | 0.02672 | 0.0496 | 0.09426 |
| 125 | -0.02789 | 0.006763 | 0.02003 | 0.01907 | 0.03103 | 0.08259 |
| 175 | -0.01918 | 0.003258 | 0.01248 | 0.01382 | 0.02327 | 0.06151 |
| 200 | -0.02165 | 0.001014 | 0.01065 | 0.01294 | 0.0219 | 0.05468 |
| 250 | -0.01053 | 0.003376 | 0.009998 | 0.01089 | 0.01766 | 0.03989 |
| 300 | -0.01077 | 0.002906 | 0.007679 | 0.008806 | 0.01451 | 0.04095 |
| 400 | -0.00422 | 0.001636 | 0.005093 | 0.005152 | 0.008281 | 0.01846 |
| 500 | -0.00379 | 0.000105 | 0.002869 | 0.003092 | 0.005355 | 0.01331 |

Now examine the corresponding CLARA Algorithm results. The next two graphs display the box-and-whisker plots based on the same process except that the CLARA Algorithm is used to select the representative scenarios.



| | Min. | 1st Qu. | Median | Mean | 3rd Qu. | Max. |
|-----|----------|----------|---------|---------|---------|---------|
| 50 | -0.03666 | 0.01428 | 0.03394 | 0.03885 | 0.05481 | 0.1731 |
| 75 | -0.01942 | 0.004485 | 0.01637 | 0.0158 | 0.02794 | 0.05635 |
| 125 | -0.00832 | 0.009057 | 0.01548 | 0.01636 | 0.02218 | 0.0428 |
| 175 | 0.000665 | 0.009482 | 0.01404 | 0.01523 | 0.01897 | 0.03734 |
| 200 | -0.00041 | 0.008229 | 0.01337 | 0.01446 | 0.01932 | 0.03488 |
| 250 | 0.002103 | 0.009004 | 0.01372 | 0.01372 | 0.01747 | 0.03151 |
| 300 | -0.00138 | 0.009033 | 0.01319 | 0.01361 | 0.01737 | 0.0354 |
| 400 | 0.004923 | 0.00996 | 0.01287 | 0.0131 | 0.01612 | 0.02231 |
| 500 | 0.004738 | 0.01029 | 0.01179 | 0.01231 | 0.01441 | 0.02046 |

For the CTE90 results, we again see that these results are liberally biased, but median results as well as the minimum and maximum relative errors are very tight and vastly superior to those of the Chueh Algorithm. We have noted from prior experience using the CLARA algorithm (set to use the sum of absolute value of the differences metric), that the algorithm chooses more tail scenarios than any other technique.

Issues Regarding Bias

Though our process is positively biased and understates the reserves and capital, we see that the average error is reasonable given the speed enhancement. Of course one may increase the size of the representative set and this will reduce bias. Donald Krouse (AEGON) has given some insight into what a practitioner may take to reduce this bias. Based on his suggestions, one could introduce weights to the scenario selection process or by experimenting with other various metrics. This may help, because the bias may arise from the fact that the training scenarios may over- or under-emphasize certain attributes within the scenarios. Also, the PPR model itself can lead to

Note how the CTE65 results have a tighter interquartile range than the CTE65 Chueh results. Also, note that the process is liberally biased just like the Chueh technique. Comparing the median values for 400, we see that the median value is 129 bp versus that of 92 bp from the Chueh CTE65. Note though even with a higher median value, the difference between the maximum and the minimum is tighter

biased results just due to the fact of how it calibrates (as discussed within the appendices). The practitioner may want to introduce weighting to the calibration process or manipulate other settings (as briefly outlined within the Techniques and Diagnostics for PPR appendix) to see if bias can be eliminated. Currently, we have used other predictive models such as neural networks and other types of machine learning, to eliminate the bias, but we have found that PPR is still superior because it does not suffer from the curse of dimensionality. Furthermore, it simulates the underlying structure of the complex capital model quite well, where these other techniques poorly calibrate to the representative sets.

Conclusions

We see that the Chueh technique and the CLARA technique are relatively comparable when calculating the CTE65 results while examining the 400 representative sets. Here we see that the median relative error for the Chueh algorithm is 92 bps with a range of results of 220 bps. Also, note that the CLARA CTE65 has a median of 129 bp with a tighter range of 156 bp. However, we observe that the CLARA technique is vastly superior to the Chueh technique when conducting the CTE90. This is because the median CTE90 relative error for the Chueh algorithm is 134 bp with a range of 575 bp, which is unreasonable, whereas the CLARA algorithm has a median relative error of 51 bp and range of 237 bp.

Our recommendation is to use either representative scenario selection process when calculating reserves, but to limit the approach strictly to the CLARA technique for making capital CTE estimations.

APPENDICES

Projection Pursuit Regression (PPR)

In linear regression, one fits a response variable Y to a collection of n predictor variables X_i in the familiar form:

$$Y = \alpha + \sum_{i=1}^n \beta_i X_i + \varepsilon$$

In additive models, the $\beta_i X_i$ are replaced with various functions $f_i(X_i)$, with this form:

$$Y = \alpha + \sum_{i=1}^n f_i(X_i) + \varepsilon$$

Projection Pursuit Regression (PPR), introduced by Friedman and Stuetzle,⁶ is a modification of this structure in that there are:

- M different f_i .
- Each f_i acts on a different linear combination of all n of the X_k .
- A specific coefficient of these linear combinations is denoted by α_{ik} .
- Each f_i is multiplied by a β_i .
- The constant term is the average of the response variable.

So PPR takes on the following form:

$$Y = \bar{y} + \sum_{i=1}^M \beta_i f_i \left(\sum_{k=1}^n \alpha_{ik} X_k \right) + \varepsilon$$

or in vector format:

$$Y = \bar{y} + \sum_{i=1}^M \beta_i f_i(\alpha_i \cdot X) + \varepsilon$$

where $X = (X_1, X_2, \dots, X_n)$ is the predictor vector, and $\alpha_i = (\alpha_{i1}, \alpha_{i2}, \dots, \alpha_{in})$.

The term "Projection" in PPR comes from the projection of X on to the directional vector α_i for each i .

The "Pursuit" arises from the algorithm that is used to determine optimal direction vectors $\alpha_1, \alpha_2, \dots, \alpha_M$.

Each f_i is called a ridge function. This is because they only have values in the α_i direction and are considered constant elsewhere. Effectively, what occurs is that the overall PPR model is a linear combination (β_i are the coefficients) of the ridge functions. These functions only take on values that arise from the projection of the predictors against the direction vectors, and the functions are assumed to take on a constant value in any other direction. So, each ridge function is like the profile of a mountain range, and we linearly combine these functions along all different ridges (as pointed out by the α_i).

On a formal basis, Y and X are assumed to satisfy the following conditional expectation:

$$\begin{aligned} E[Y | X_1, X_2, \dots, X_n] &= \\ &= \mu_y + \sum_{i=1}^M \beta_i f_i(\alpha_i \cdot X) \end{aligned}$$

continued on page 22 >>

PPR models can take extremely large amounts of data and create a very good model of the underlying data.

with $\mu_y = E[Y]$ and the f_i have been standardized to have zero mean and a unit variance. That is: $E[f_i(\alpha_i \cdot X)] = 0$ and $E[f_i^2(\alpha_i \cdot X)] = 1$, where i takes on values from 1 to M . We assume that the realized sample values for the random

variables Y and $X = (X_1, X_2, \dots, X_n)$ are independent and identically distributed to the distributions of Y and X , respectively.

The ppr algorithm in the R stats library⁷ estimates the best β_i , f_i and the α_i by minimizing the following target function for the mean square error:

$$E \left[Y - \mu_y - \sum_{i=1}^M \beta_i f_i(\alpha_i \cdot X) \right]^2$$

across all the data samples for Y and X . Note: This expectation can be a weighted average.

A powerful trait of PPR models, since the predictor vector X is projected, is that interactions between different X_j and X_k are included within the model, whereas other model algorithms cannot do this without user intervention.

We justify this by using an algebraic demonstration based on the S-Plus Guide to Statistics⁸ recast into our notation as follows:

Suppose that the actual data model is $E[Y | X_1, X_2] = X_1 X_2$.

Let $Y = 0$, $M = 2$, $\beta_1 = \beta_2 = .25$ and assume that $\alpha_1 = (1, 1)$ and $\alpha_2 = (1, -1)$. Furthermore assume that $f_1(t) = t^2$ and $f_2(t) = -t^2$. Let $X = (X_1, X_2)$

Now

$$f_1(\alpha_1 \cdot X) = (X_1 + X_2)^2 = X_1^2 + 2X_1X_2 + X_2^2.$$

and similarly

$$f_2(\alpha_2 \cdot X) = -X_1^2 + 2X_1X_2 - X_2^2.$$

and finally,

$$\sum_{i=1}^M \beta_i f_i(\alpha_i \cdot X) = X_1X_2.$$

So, we can see that if the β_i , f_i and α_i are optimally selected and the underlying model has

interactions between different predictors, PPR should capture this.

Advantages and Disadvantages of PPR

The following is a list of advantages of using PPR as a model:

- The model is a continuous function. According to Venables and Ripley,⁹ they cite Diaconis and Shahshahani¹⁰ and say that given a large enough number of ridge functions, PPR can approximate arbitrary continuous functions.
- It is the best possible fit since every component is solved for the minimization of the weighted least squares.
- Each ridge function does not extrapolate outside of its specific domain. If the specific $\alpha_i X$ is outside the domain, the relevant domain endpoint is used.
- The model handles the interactions between the different predictors as we saw in the last section.
- PPR models categorical predictors as easily as continuous predictors.
- PPR models can take extremely large amounts of data and create a very good model of the underlying data. One can also adjust the model to distinguish between model fit and model smoothness.
- PPR does not suffer from the curse of dimensionality (COD). COD arises from the increased complexity of a multi-dimensional surface. Since PPR optimally is solved one ridge function at a time, the difficulty of trying to locate global optimal values for model calibration is eliminated.

The disadvantages are:

- The range of a PPR model may be outside of the range of acceptable values. For instance, if one were using PPR to model mortality, model results could fall below zero or above one. However, PPR will not extrapolate outside the existing ridge functions, so if any predictor projects on a specific α with a value outside the

domain of a specific ridge function, the ridge function takes on the value either at the furthest point on the right hand side or left hand side. This no extrapolation rule can lead to biased results.

- All of the parameters are point estimates, and there is no distributional consideration given to the significance of a specific parameter. Because one is unable to create a confidence interval using the R ppr function around each of the α_{ik} or the β_i , one is not able to determine if a specific parameter is significant to the model. In fact, one is unable to test if the actual model is significant, other than the use of the goodness of fit statistic. There are complex methods that have been developed using spherical statistics to overcome this, but these require an understanding of advanced Banach Algebra in functional analysis and have not been included within the R ppr function.
- One can easily overfit or overexplain the data. See Venables and Ripley for a further discussion.
- The model can be *too flexible*, which can make interpretation of the PPR model difficult. Again, see Venables and Ripley.

Techniques and Diagnostics for PPR

The procedure when using the R ppr algorithm is as follows:

First, one specifies that M should range between $M_{MIN}=1$ and some positive integer M_{MAX} . The ppr algorithm then creates a PPR model for each M from M_{MAX} to M_{MIN} in a descending fashion, and at the same time produces a goodness of fit statistic for each value of M . Scanning this list of goodness of fit values should display a local minimum. If this local minimum is at M_{MAX} one should reprocess the experiment with a larger M_{MAX} . Once one determines the local minimum, say s , reset $M_{MIN} = s$ and reprocess the ppr algorithm with the same M_{MAX} as before. The resultant model arising from the backward iteration from M_{MAX} to M_{MIN} will then be the best PPR model.

Two other components that are implemented in ppr are the concept of “bass” and “optlevel.” “Bass” is Friedman’s super smoother bass tone control¹¹ that

is used with automatic span selection. It is used in ppr to smooth the results. The range of values allowed with this component is from 0 to 10. To increase smoothing within the data, increase this value. The default is 0, and this setting gives the best fit to the underlying data. Bass is similar to the h smoothness parameter used within the Whitaker-Henderson graduation formula.

“Optlevel” is an integer from zero to three, which determines the optimization thoroughness. The best models usually are obtained if this is set to three. At level zero, the ridge functions are not refitted. At level one, the projection directions are not refitted, but the ridge functions and the regression coefficients are. Levels two and three refit everything, but level three takes pains to re-balance each regressors’ contribution at each step and so reduces the chance of converging to a saddle point in the sum of squares.


One diagnostic aid in PPR model building is to plot the ridge functions. If these ridge functions are very noisy or discontinuous, you should expect that the resultant PPR model will behave oddly.

Another effective diagnostic aid is to both plot the fitted \hat{Y} against the actual Y and do a simple linear regression of Y against \hat{Y} , assuming no intercept. The scatterplot should display symmetry around the 45 degree line and the coefficient of the regression should be approximately one. These two diagnostics will indicate how well the PPR model will perform as a predictive model.

Note: A PPR model does not extrapolate outside of the sample data. So, frequently the resultant fitted values from PPR model will hit a maximum value and will not grow any larger no matter how one manipulates the predictors. This is not the case for linear regression models, where there are no natural limits placed on how one sets any respective X_i . However, one may revise the prediction object to conduct extrapolations. However, one must first feel comfortable with the continuity of the separate ridge

Two other components that are implemented in ppr are the concept of “bass” and “optlevel.” “Bass” is Friedman’s super smoother bass tone control that is used with automatic span selection.

continued on page 24 >>

functions. If these functions are very noisy or appear not to be differentiable, you might want to avoid all extrapolation. 

Endnotes

- ¹ Chueh, Y. 2002. "Efficient Stochastic Modeling for Large and Consolidated Insurance Business: Interest Rate Sampling Algorithms." *NAAJ*, pp. 88-103, available at http://www.cwu.edu/~chueh/naaj0207_8.pdf.
- ² CLARA algorithm, see: <http://rweb.stat.umn.edu/R/library/cluster/html/clara.object.html>
- ³ Projection pursuit Regression, see: <http://www.slac.stanford.edu/pubs/slacpubs/2250/slac-pub-2466.pdf>
- ⁴ Craighead, S. 2000. "Insolvency Testing: An Empirical Analysis of the Generalized Beta Type 2 Distribution, Quantile Regression, and a Resampled Extreme Value Technique," *ARCH*, pp. 13-149, available at <http://www.soa.org/library/research/actuarial-research-clearing-house/2000-09/2000/arch-2/arch00v26.pdf>.
- ⁵ Box-Whisker Plot, see http://en.wikipedia.org/wiki/Box_plot.
- ⁶ Friedman, J.H., and Stuetzle, W. 1981. "Projection Pursuit Regression." *Journal of the American Statistical Association* 76: 817-823.
- ⁷ R Development Core Team. 2006. "R: A language and environment for statistical computing," R Foundation for Statistical Computing, Vienna, Austria. ISBN 3-900051-07-0, URL <http://www.R-project.org>.
- ⁸ S-Plus 6 for Windows Guide to Statistics. 2001. Vol 1, Insightful Corporation, Seattle, Wash.
- ⁹ Venables, W.N., and Ripley, B.D. 2002. *Modern Applied Statistics with S, Fourth Edition* Springer.
- ¹⁰ Diaconis, P., and Shahshahani, M. 1984. "On Non-linear Functions of Linear Combinations." *SIAM Journal of Scientific and Statistical Computing* 5, pp. 175-191.
- ¹¹ Friedman, J.H. 1984. "A Variable Span Scatterplot Smoother." *Stanford University Technical Report No. 5* Laboratory for Computational Statistics.