Toiling in the Actuarial Vineyards: Accelerating Traditional Experience Analysis With GLM Trees

By Philip Adams

The ever-increasing volume and diversity of data available to an actuary is both exciting and terrifying. Exciting because of the amazing and unexpected findings waiting to be discovered, and terrifying because of the drudgery and disappointment to be endured along the way. And time is of the essence.

Technology rewards us with a mess of data and helps us sift through it with numerous open-source solutions.

BRIEF HISTORY OF ANALYSIS

Actuaries have traditionally analyzed data manually. Even with technology, an actuary is manually reviewing, evaluating and judging the fitness of data for its intended purposes. Generally, the traditional approach often follows a similar recipe for mortality analysis:

1. Look through the dimensions of the data to find statistically significant factors driving mortality.
   a. One dimension is considered at a time.
   b. Optionally, two or more can be considered at the same time.
   c. Filters can be introduced at any point.
2. Develop a set of factors for that dimension/combination of dimensions.
3. Do one of the following:
   a. Adjust experience with the new factors.
   b. Don't adjust.
4. Repeat 1–3 as needed.
5. Finally, check for reasonableness and fit.

The recipe works, but not without shortcomings.

1. Sifting through combinations of factors is labor intensive, and it is probable that some important factors and combinations will be missed.

2. When to stop looking for factors is left to judgment. The risk is that the modeler could either stop too early and miss important factors, or stop too late and waste valuable time.

3. Factors are not fit simultaneously. If there are dependencies, estimates may change when using its related factors. For example, if smoking status isn't distributed identically by gender, the Smoker factor may pick up some signal belonging to Gender.

4. It can happen that different types of models suit different parts of the data. For example, an interaction of dimensions in one section of the data might be unneeded in another.

The manual recipe does not scale with volume. Automated approaches are needed to fill the gap. A variety of predictive models can offer relief, including GLMs (generalized linear models), GAMs (generalized additive models), decision trees/forests, elastic nets, gradient boosting, etc.

This paper introduces a hybrid approach, the GLM tree, to an actuarial audience.

While GLM trees are not the only tool that can deal with these issues, they have the advantage of being intuitive and easy to explain. This cannot be said of random forest, elastic net regression, or boosting methods. As you will see, they get an actuary to an answer more efficiently than other methods all while remaining explainable.

In the early days of my mortality modeling, GLMs and GAMs were the best available tools.

DESSERT BEFORE DINNER

In the fall of 2018, the Society of Actuaries issued a challenge whereby the Individual Life Experience Committee (ILEC) released experience data from 2009 to 2015 and invited parties to submit the best data analysis solution. The top three entrants were awarded fabulous cash prizes.

Consider the following findings for term experience:

1. The mortality experience for term business having at most one preferred class (two-class) deteriorated significantly over the study period.

2. Experience for 3- and 4-risk class systems improved significantly.

3. Term experience with face amounts below $100,000 deteriorated.

GLM TREES

In the early days of my mortality modeling, GLMs and GAMs were the best available tools. In some cases, today's algorithms had not been invented yet (or at least not revealed to a wide audience).

Model fitting with GAMs can be a chore. Stepwise feature selection as implemented for GLMs does not work with splines. Instead, I searched for methods for automated feature/interaction detection. One can find many approaches, including chi-square automatic interaction detection (CHAID) and other decision tree types. CHAID had many of the features I wanted yet appeared to be incompatible with the A/E and qx analyses.

In 2008, Achim Zeileis, Torsten Hothorn and Kurt Hornik introduced a rigorous theoretical framework built on research into combining parametric models such as GLMs with decision tree models. The algorithm relies heavily on **parameter fluctuation tests** (method to detect whether there is unmodeled variation in the residuals).

If you were building a model by hand, you might check how well the model fits the data by examining one or more dimensions. For example, if you fit a constant percentage and then look at fluctuations of actual-to-model mortality, you might note some residual variation for some dimensions. The visual variation may look like noise, or it might show some patterns in the actual-to-model mortality.

In the case of model-based recursive partitioning, that last part about residual variation looking like noise is the key. For an ordered dimension (like age), under the null hypothesis where the residuals are independent, identically distributed random variables, the running sum of the residuals is distributed as a Brownian bridge (a random walk starting and ending at 0), subject to appropriate scaling. In the unordered case (e.g., categories), a Chi-square goodness-of-fit test is applied to the residuals for that dimension.

The dimension with the most variation wins. The algorithm searches for the best binary subdivision for that dimension. For an ordered dimension, each break point in the data is tested sequentially. For an unordered dimension having *n* levels, the algorithm tries all of the binary subdivisions of the dimensions. There are 2^{n-1}-1 possible subset breaks to check. In both cases, the partition that maximizes the likelihood the most wins.

Now that the data have been broken into subsets, the algorithm starts the process over on the smaller pieces. Eventually, the procedure stops, either because there is nothing to improve or the analyst specified a stopping rule.

A CONCRETE EXAMPLE

Since mortality trend is among the most important topics for life insurance, I attempted several model types, both regression and tree-based. As it happens, only GLM trees were able to discover what subsets of the data had meaningful differences in mortality levels and trends.

I used the Poisson regression model:

Number of Deaths ~ β_0+β_1 Experience Year+log(Expected Claims 2015VBT)

The partitioning variables are everything else. Because some variables are insurance plan specific, I carried out the analysis separately for term and perm products. This example follows the algorithm/results for the term analysis. Since this is an exploratory exercise, no training/test split is performed.

Note that I emphasize intuitive understanding in my example and not technical understanding. Therefore, I am combining the parameter fluctuation test with the subset testing.

The algorithm starts by fitting the regression model to the data (Fig. 1). Mean trend is semi-significantly positive.

There are 12 variables to test. We demonstrate the first three and stop with face amount band. The first variable is gender. In Figure 2, we see the models for a potential split. While there is unmodeled trend variation for females, there is less for males. The mean mortality level has small variation. The second is age basis. In Figure 3, this appears to be a promising split, with ANB showing deterioration yet small variation for mean mortality.

Figure 1
First Regression Model on Term Data

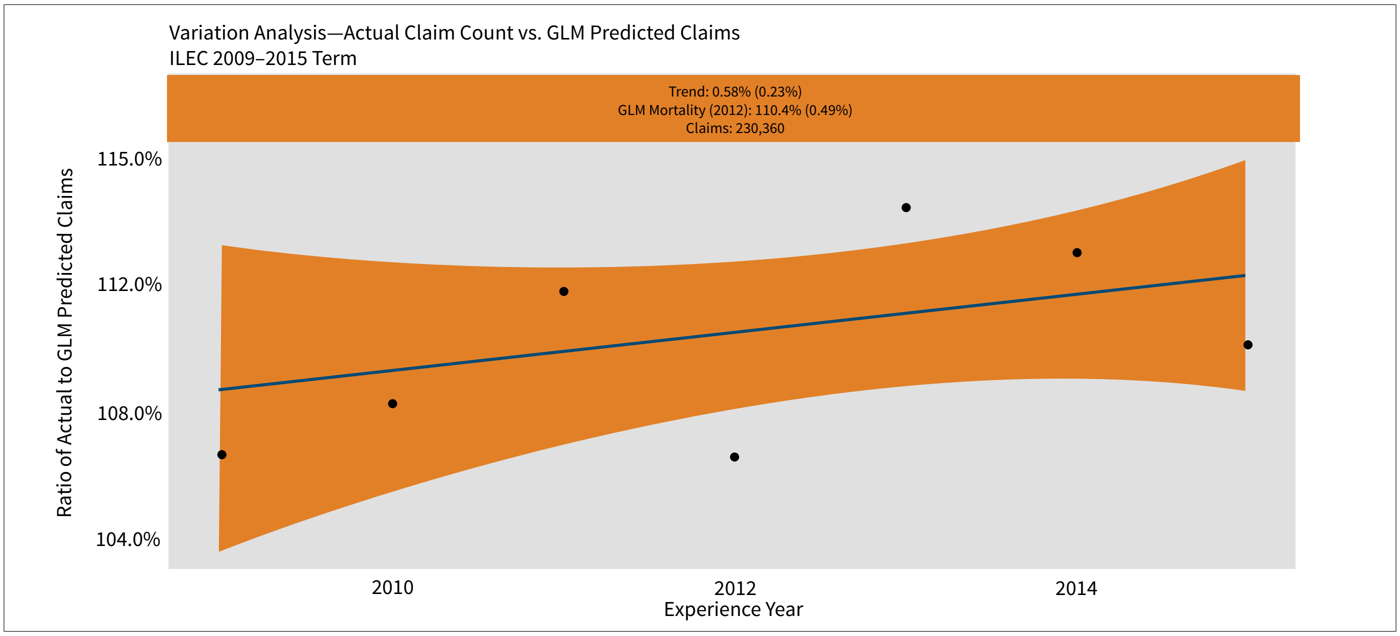


Figure 2
Candidate Models for Gender Split

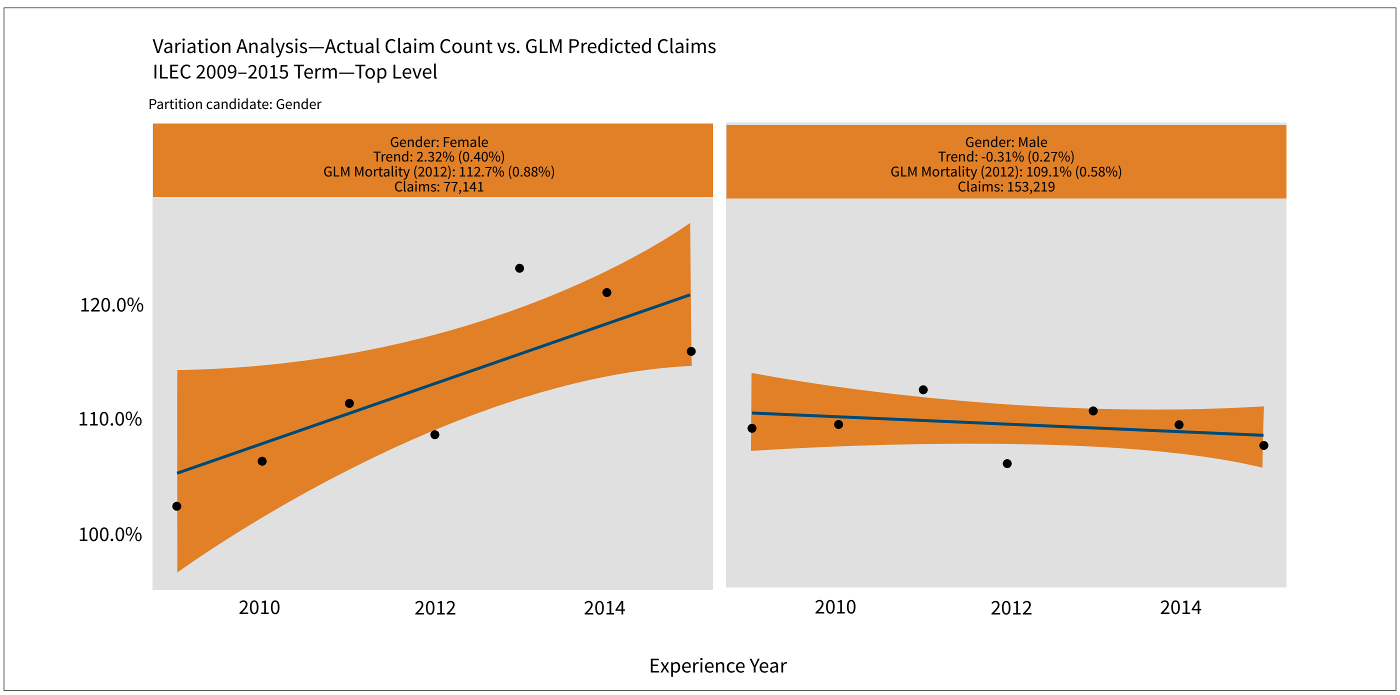


Figure 3
Candidate Models for Age Basis

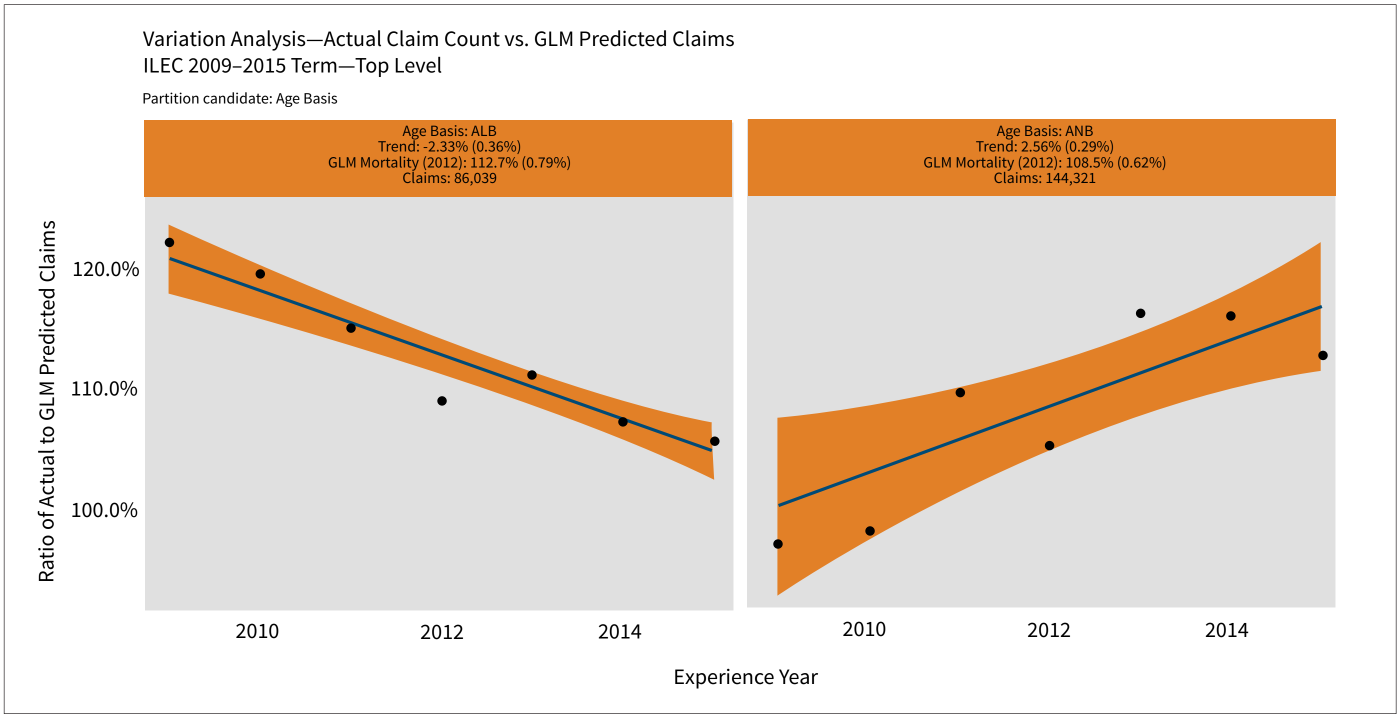
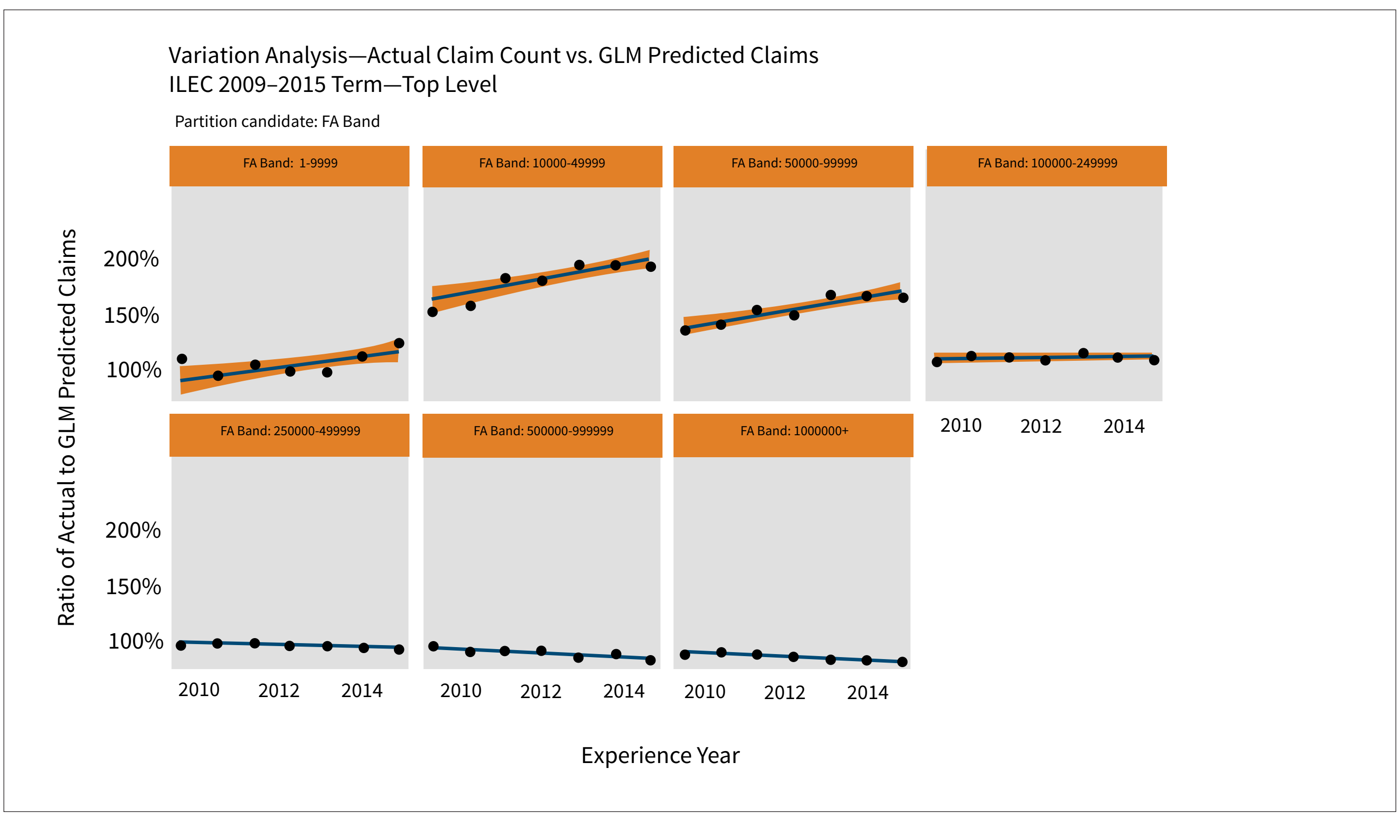


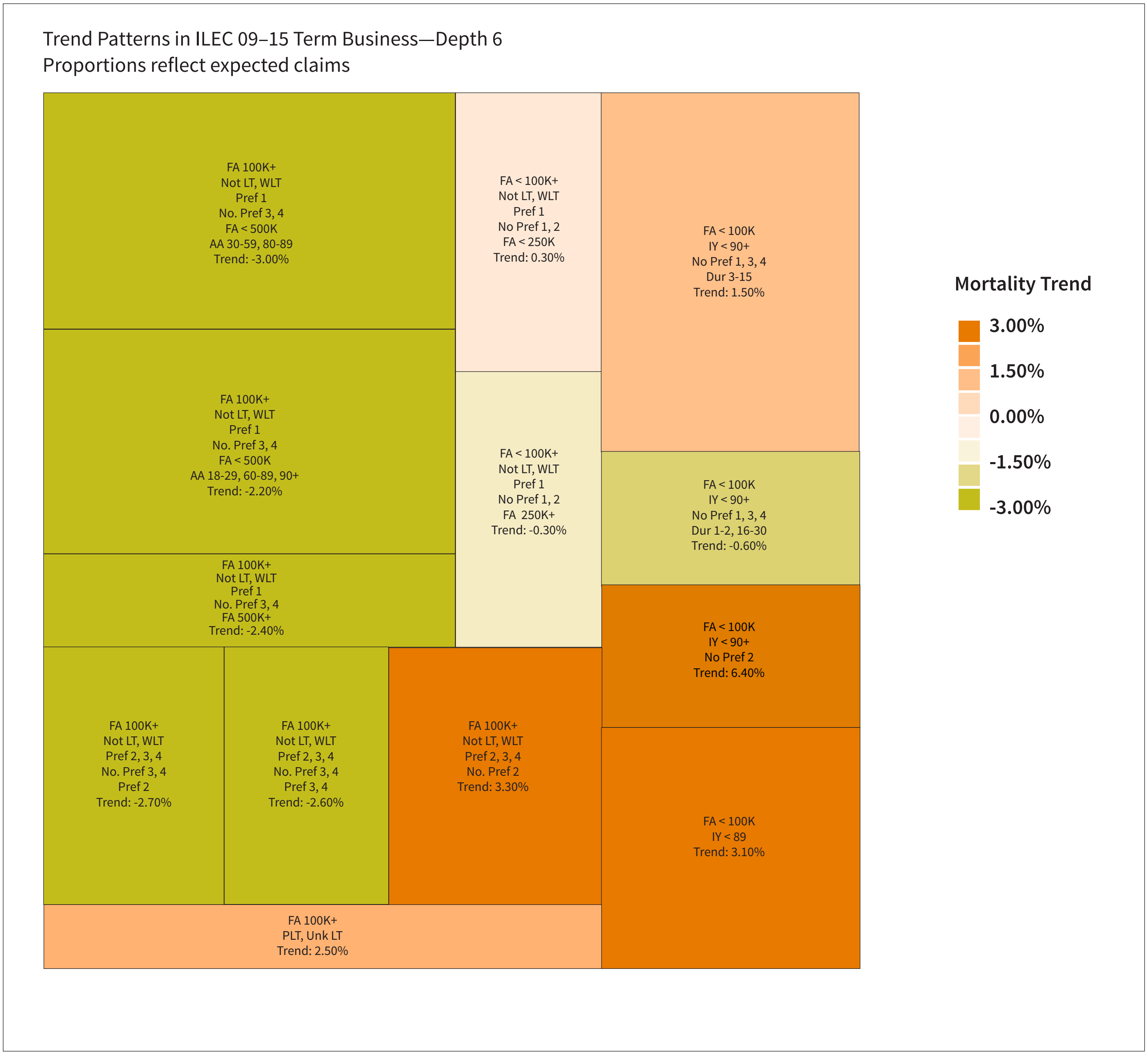
Figure 4
Candidate Models for Face Amount Band



In Figure 4, there is substantial variation for both trend and mean mortality with increasing face amount band. Trend ranges from positive to negative with increasing face amount, and with the exception of face amounts under 10,000, mean mortality declines with increasing face amount. After testing the other nine variables, face amount was the first dimension along which to split the data. Because it has seven levels, there are 63 splits. To lessen computation, face amount band is treated as an ordered factor, reducing this to six splits. The chosen split was at 100,000.

The algorithm then builds a tree recursively defined by split conditions with a GLM at each node/leaf of the tree. The partykit package expresses the results as a traditional tree. To get around the limitations of the default output, I expressed the results as a tree map as in Figure 5. The minimum node size was 10,000 expected claims.

Figure 5
Tree Map of GLM Output for Trend



If you let your eyes wander, some findings emerge:

1. Face amounts less than 100,000 exhibited deterioration (the right third of the square). In the instance of 2-class preferred systems, deterioration was 6.4 percent (SE 0.42 percent) on average per year. One possible exception was the light green block, but this is statistically not significant (SE 0.44 percent).

2. Face amounts 100,000 and higher witnessed improvement in general, with two exceptions.

    a. The lower left corner contains post-level term and unknown level-term business. The trend here is potentially contaminated with slope misalignment. The net deterioration is 2.5 percent (SE 0.28 percent) per year on average.

    b. The angry red block above it is residual standard of 2-class preferred systems including the non-level term and within level term business. Within level term dominates the block. There was substantial deterioration of 3.3 percent on average per year (SE 0.43 percent).

3. Right next to the angry red block is a very green block that contains residual standard of 3- and 4-class preferred systems. For these lives, there was substantial improvement on average of 2.6 percent per year (SE 0.32 percent), partly offsetting the deterioration of the 2-class systems.

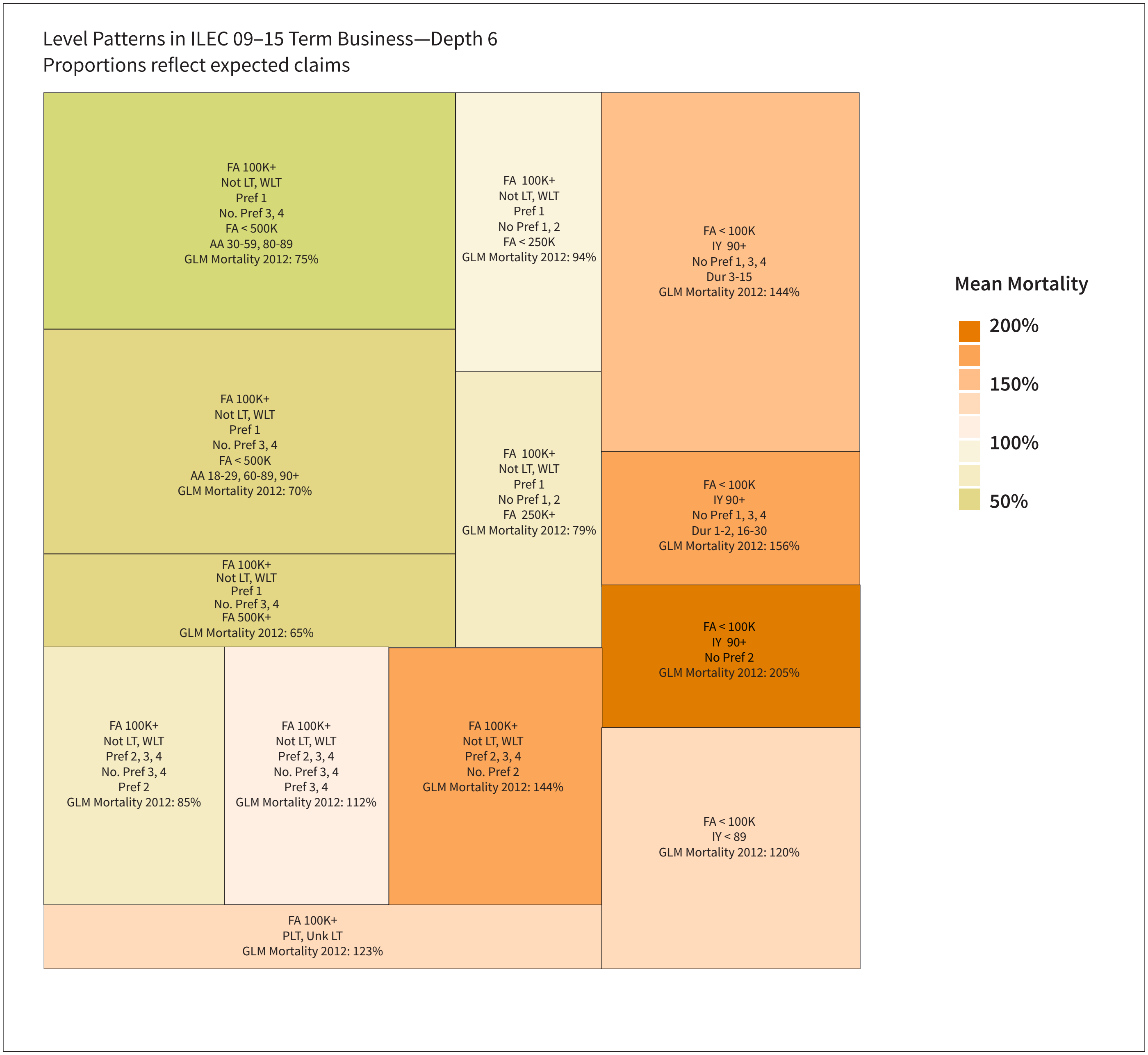
The same plot can be had for adjusted mean mortality (centering at 2012). In Figure 6, we see that many relationships are as we expect: higher face amounts have better mortality, better preferred has better mortality, post-level term has worse mortality. Standing out is the high mortality for 2-class preferred systems with face < $100,000.

WHAT ABOUT PERM? ANALYSIS BY AMOUNT?

Perm has been omitted from this paper for brevity. The claim count is nearly 10x as large as for term, with much longer issue year horizons and more insurance plan types.

Analysis by amount has a few differences. For parameters, the GLM family is changed to a Tweedie distribution with parameter 1.2. Minimum size depends on the specified weighting vector. The minimum size is set to 10,000 * $50,000, or $500,000,000, and the maximum depth tree depth is set to six. All but one of the resulting leaves has at least 10,000 claims.

Figure 6
Tree Map of GLM Output for Mean Mortality



LIMITATIONS

The intent of the analysis was to unravel some of the riddles around trend in the ILEC 2009–2015 dataset. Since the minimum claim size per node was so large, it is likely that more insights can be gained by allowing the algorithm to drill deeper or changing the GLM model used for each node.

I encountered a few problems when applying the GLM tree function in the partykit package to the data. The first was data sparsity with depth; there must be enough diversity in the data to support fitting a GLM within any proposed node. An early attempt at GLM trees was to have the GLM model template be an interaction between attained age group and duration group. The result would be an optimal subdivision of the data with a custom select-and-ultimate mortality table for each node of the tree. However, not every combination of age and duration will be available in every subset, or there may be no claims.

The second was weighting. The GLM tree function applies the same weights parameter to the GLM fitting and the parameter fluctuation tests. This is problematic when using an offset in a Poisson model. If a weights vector is specified, the resulting GLMs will be skewed. If no weight is specified, the individual GLMs are fine, but the parameter fluctuation tests will weight equally each row of the data. Thus, it was necessary to customize the code to allow separate weights for the GLM fitting steps.

The third was lack of accommodation for splines. I had attempted to build a "GAM tree" function where the models within each node were GAMs. Adapting the spline parameters to parameter fluctuation tests proved challenging, and I ultimately set the task aside for later research.

FUTURE DIRECTIONS

I offer GLM trees as a valuable tool that helps to bridge the gap between the needs of actuarial analysis and the potential of data science methods. As an exploratory tool, it can illuminate structures in datasets. Using the typical recipe with training and test data, it can be applied as a predictive model. It can also be a point of departure for additional analysis, such as exposing where to focus further analysis or as a point of departure for more sophisticated models. ■

Philip Adams, FSA, CERA, is an AVP and actuary, Biometric Research, for Munich American Reassurance Company. He can be contacted at *padams@munichre.com.*