Arius®
Generalized
Linear Model

Milliman

IT TAKES VISION

Information in this document is subject to change without notice. The software described in this manual is furnished under a license agreement. The software may be used or copied only in accordance with the terms of that agreement.

No portion of the contents of this publication may be reproduced or transmitted in any form or by any means without the express written permission of Milliman.

Table of Contents

1. Getting Started with the Arius Generalized Linear Model

Generalized Linear Models (GLM) can play an essential role in claims reserving. While traditional Chain Ladder based methods focus on the exposure and development period dimensions of a loss triangle independently, a GLM framework allows you to account for interactions between these dimensions, as well as changes across the calendar period dimensions. Modeling the impact of trends within these additional dimensions may help with model performance in unstable environments, such as:

- Changes in claim settlement rates,

- Changes in case reserve adequacy,

- Changes in the mix of business by type of risk or type of claim,

- Changes in policy limits or deductibles,

- High rates of inflation or wide fluctuations in inflation rates, and

- Shifts in legal, regulatory, or social-economic environments in which an insurer operates.

In addition to this, simulating insurance claims using statistical models is complicated. One goal when modeling any data is to calibrate the simplest model with the least assumptions and variables but with the most significant explanatory power, also known as the principle of parsimony.

Not using enough parameters could result in not capturing all the essential features of the data. In contrast, over-parameterization may capture features that are just a result of random fluctuations, resulting in a non-predictive model that fits the underlying data well but is not applicable beyond that. With a GLM framework, we have the flexibility to specify only as many parameters as we need to get a robust model.

The purpose of this document is to provide a step-by-step guide for parameterizing and running a GLM model within Arius. For an understanding of the statistical theory underlying GLMs, please refer to Section 1 of the Casualty Actuarial Society's *Practitioner's Guide to Generalized Linear Models*. If you are new to the Arius Stochastic module, please refer to the *Stochastic User Guide* found under HELP | USER DOCUMENTATION.

SETTING UP A NEW MODEL

The primary analysis process looks like this:

1. Open the Arius software and use FILE | NEW to create a new project file. Select **Create a New Arius Project**.

2. Provide necessary information about the data you are entering and working within the **Project Settings** dialog:

   - Use the **Data Structure** tab to change the size and structure of the triangles you are entering into the model;

   - Use the **General** tab to enter project information and notes; and

   - Use the **Segments** tab to add all of the lines of business you plan to work with this model; each line of business or reserving segment has separate data, models, and assumptions.

   - Click **OK** on the bottom of the **Project Settings** dialog to finish creating a new file.

3. Use FILE | SAVE AS to save your file.

4. From the **Home** ribbon, click on the **Model Options** icon in the **Stochastic** area to open the **Model Options** dialog.

  ▪ Use the **Options** tab to change any of the **Global Options** used with all stochastic models.

  ▪ The number of iterations defaults to 10,000, but you can enter any value between 1 and 1,000,000. The iteration count impacts the speed of the model (e.g., you should use a lower number while testing models), but you should set the number of iterations high enough to get a stable result from one run to the next.

  ▪ The seed value defaults to zero, which means that Arius simulates a new set of random values for each simulation. If you would like to utilize the same set of random numbers each time (e.g., to test the impact of a specific model change), you should enter a positive integer in this field between 1 and 2,146,483,648 (=2^{31}).

  ▪ In addition to the **Mean**, **Standard Error**, **Coefficient of Variation**, **Minimum** and **Maximum** values, you can specify which ten percentiles appear in the simulation results tables. By using a Seed Value other than zero, you can specify different percentiles each time you run the model and obtain more than ten percentile outputs.

  ▪ If you select the **Yes, Term Enable Discount Rate** option, then use the **Term Discount** tab to either manually enter, or choose from a lookup table, a discount rate yield curve.

  ▪ If you select **GLM Diagnostics** for the **Save Results to File** option, the system provides a PaidGLM Diagnostics output file under the *C:\Users\username\Documents\Milliman\Arius\Sim_Results* folder that includes the calibrated Design Matrix and Covariance Matrix of the specified parameters. This file generates whenever you **Run Diagnostics**.

  ▪ Under the **Default Model Selection** tab, select the **Generalized Linear Model** option for Paid/Ultimate and any other models you expect to use for every segment. The GLM can also be enabled on a segment by segment basis, using the **Choose Models** dialog.

  ▪ Click **OK** on the bottom of the **Model Options** dialog to save your changes.

5. Below the **Home** ribbon, use the **Segment** dropdown list to select one of the lines of business.

6. In the DATA | INPUTS area of the **Navigation Pane**:

  ▪ Enter Paid Loss data.

  ▪ Enter Earned Premiums or Ultimate Premiums (Optional, only needed if you intend to generate loss ratios).

7. In the STOCHASTIC | GENERALIZED LINEAR MODEL | MODEL ASSUMPTIONS area of the **Navigation Pane**:

  ▪ The default model has the **Use Intercept** option enabled. Before parameterizing the GLM model, we need a baseline model to get an initial estimation of residuals. The Intercept-only model assumes that each predicted incremental paid loss cell has the same expected value. While it is our starting point, it is not likely to be a reasonable model.

  ▪ The Arius GLM model assumes a Log-link function and a Poisson distributed error term. The link function explains the relationship between the underlying linear predictors and the mean of the distribution function. The log-link function estimates logs of additive effects (which, when exponentiated, become multiplicative), and the Poisson distribution assumes the variance increases with the expected value of each observation. We also utilize a Gamma distribution for the Process Variance. These parameters are common in triangular stochastic reserving models and are the only option within the GLM model at this time.

Note:

The simulation speed is a function of the number of parameters, the size of the underlying triangles, the number of iterations, and whether simulation results are saved to a CSV file.
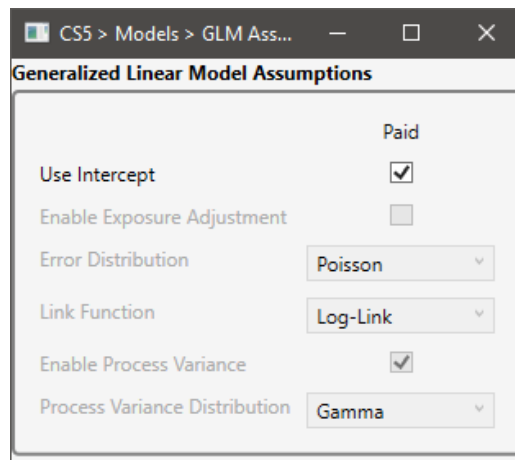
8. From the **Home** ribbon, click on the **Run Diagnostics** icon and select the **Run Diagnostics for** *SegmentAbbr* option to fill the exhibits and graphs with diagnostics.

9. In the STOCHASTIC | GENERALIZED LINEAR MODEL | PARAMETERIZATION area of the **Navigation Pane**:

    - Review the patterns in the **Residual Graph** window and assign parameters to one or more of the objects.

        - Open the **Exposure Period Residual Plot**. You have the option here to:

            - Highlight all columns or a subset of columns (click on a column and scroll to the right to select contiguous cells, such that they are highlighted in blue) and select New Parameter button, or

            - Select the **Quick Add** button (which is equivalent to individually selecting each column of the residual plot and adding a level/constant parameter to each column).

        If you add a new parameter, provide a unique name, and select the type of parameter (e.g., Level/Constant, Index, Index Squared, or Ln(Index)).
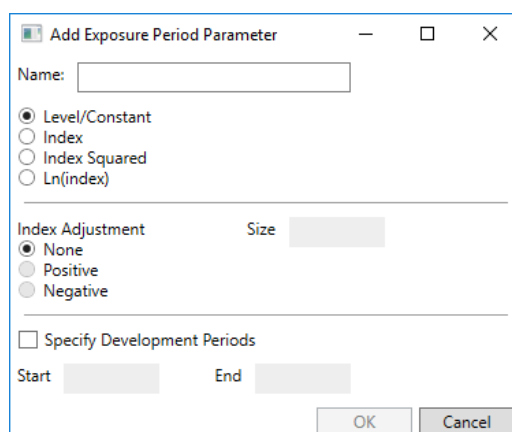
        The type of parameter chosen effects the **Design Matrix -** the matrix of values of the explanatory variables which attempt to explain observed data, which in our case is the incremental paid loss. In Arius, each row of the Design Matrix represents an observation

in the triangle (starting from the upper left corner and moving left-to-right, top-to-bottom) with the successive columns corresponding to the variables and their specific values.

- **Level/Constant** – Design matrix receives a value of 1 for the selected range

- **Index** – Design matrix receives a value equal to the period number for the selected range—the impact of the parameter increases/decreases with the period in which it applies (for example, the second column)

- **Index Squared** – Design matrix receives a value equal to the square of the period number for the selected range

- **Log Index** - Design matrix receives a value equal to the natural log of the period number for the selected range

As a simplified example, consider a single row of data. For our purposes, the period can refer to either exposure, calendar, or development age.

Period	1	2	3	4
Value	50	75	100	45

In this example, the predicted incremental at each period (X), when adding a single parameter across all four periods, by type, are:

- **Level/Constant:** $PaidLoss_X = \exp(parameter)$

- **Index:** $PaidLoss_X = \exp(parameter * X)$

- **Index-Squared:** $PaidLoss_X = \exp(parameter * X^2)$

- **Log Index:** $PaidLoss_X = \exp(parameter * \ln(X))$

Note that period number starts at 1 for each of exposure, development, and calendar periods. For more information on the other options within this dialog, see the section below for *Advanced Parameterization Options* below.

- Repeat the steps above for both the **Exposure Residual Plot** and **Calendar Period Residual Plot** as needed.

10. From the **Home** ribbon, click on the **Run Diagnostics** icon and select the **Run Diagnostics for** *SegmentAbbr* option to fill the exhibits and graphs with diagnostics.

11. Review the various diagnostics available within Arius to improve your model fit by adding and removing additional parameters and reducing the influence of outliers.

- Navigate to the Stochastic | Generalized Linear Model | Parameterization or the Stochastic | Generalized Linear Model | Model Summary area of the **Navigation Pane**.

- Open the **GLM Parameter Review** object. This table summarizes parameters added to the GLM model and displays the following, for each parameter:

- **Parameter –** This represents the name of the parameter as specified by the user (or by the quick add functionality). Notice that all parameter names are prefixed with an "a," "b," or "c" for exposure, development, calendar periods, respectively, or "a1-ax", "b1-bx" and "c1-cx" for parameters added via the **Quick Add** button. If you check the **Use Intercept** option, you see an "I" for the intercept parameter, which applies to all observations. Finally, advanced parameters that are applied to subsets of exposure or development periods are prefixed with a "p."

- **Applies to –** The periods that parameter applies to (within the period type specified by the parameter prefix, as outlined above).

- **Coefficient –** The estimate of the coefficient in the generalized linear model.

- **Standard Error –** The standard error of the coefficients.

- **T-Statistic** – The t statistic of each coefficient tests the null hypothesis that the coefficient is zero (against the alternative that it is non-zero) given the other predictors in the model.

- **P**-**Value** – The p-value for the T-Statistic of the hypothesis test is that the corresponding coefficient is equal to zero or not. If the P-Value for a given parameter is higher than α%, that parameter is not significant at the α% significance level, given the other terms in the model.

- To remove parameters, click on the name of the parameter (use control+click or shift+click to select multiple parameters), and click the **Delete** button. Note that deleting a parameter clears out the **Parameter Review** object. Clearing the results is intentional, as the coefficients and diagnostics on the parameters no longer apply without the removed parameter. Running Diagnostics returns parameters and diagnostics on the updated model.

  Note that any other change to the model (e.g., enabling/disabling the intercept or adding parameters) also clears out this table, to ensure consistency between the model specified and that presented in Arius.

- Navigate to the STOCHASTIC | GENERALIZED LINEAR MODEL |MODEL SUMMARY area of the **Navigation Pane**.

  - In addition to the **Parameter Review** object, you can also review the **Model Fit Diagnostics**. This table displays the statistical measures of fit for the model as a whole (versus the parameter specific measures discussed above).

- Navigate to the STOCHASTIC | GENERALIZED LINEAR MODEL | MODEL DIAGNOSTICS area of the **Navigation Pane**.

  - Additional diagnostics for the parameterized model include:

    - **Diagnostic Triangles:**

      - **Fitted Incremental Paid Loss Triangles** – These show the mean result of the calibrated model for each cell of the triangle.
      - **Actual v. Expected Triangle** - the heat maps on this object are useful for indicating poor fit. The goal is a distribution of colors that is random across the triangle. Clumps of one color indicate a poor fit and may indicate the need for additional terms.
      - **Standardized Pearson Residuals/Standardized Deviance Residuals** – These objects provide a way to see the values underlying the residual plots used for model parameterization.

    - **Q-Q Plot for Normality** – This object is used to verify that your residuals are normally distributed, to verify the validity of the parameters and assumptions in your model.

    - **Box Whisker Plot of Residuals** – This object can identify potential outliers that may skew/bias results if not given reduced weight or excluded from the parameterization

12. Navigate to the STOCHASTIC | GENERALIZED LINEAR MODEL | PARAMETERIZATION area of the **Navigation Pane**.

    ▪ Open **Weights** object. To remove an outlier, you can assign a weight of 0 in the corresponding cell. You can also use this weights array to attach more importance to particular cells if warranted. For more information, see the section below for *Reducing Data Point Influence with Weights & Outliers*.

    ▪ After modifying parameters or outliers, you need to use the **Run Diagnostics** icon again to recalculate each of the diagnostic statistics.

    ▪ Navigate to the STOCHASTIC | GENERALIZED LINEAR MODEL | TAIL PARAMETERIZATION area of the **Navigation Pane**:

        ▪ When simulating, the parameterized model applies to the lower right corner of the loss triangle. In many instances, the losses may not be fully paid out by this point. If this is the case, a tail factor can be calibrated based on the parameterized model, which is applied when running the cash flows. For more information, see the section below for *Calibrating a Tail Factor Model*.

13. Once a model (and tail factor model, if applicable) is parameterized, you can run the simulations for this segment using the **Run Simulations** icon and selecting the **Run Simulations for *SegmentAbbr*** option from the Home ribbon.

14. Navigate to the STOCHASTIC | GENERALIZED LINEAR MODEL | RESULTS area of the **Navigation Pane**:

    ▪ The simulation process involves resampling the GLM parameters specified, creating a squared triangle, then applying the tail factor calculation (if applicable) for the selected number of iterations. The output generated by the Arius GLM Simulations is formatted identically to the output from the other simulation models. Please refer to the *Stochastic User Guide* (Chapter 5: Summary Output) found under HELP | USER DOCUMENTATION for more information.

    ▪ Iteratively adjust model parameters or options and re-run the diagnostics or simulations until you are satisfied with the model fit and simulation performance.

15. Finally, repeat steps #5 to 14 for each line of business in the Arius project.

ADVANCED PARAMETERIZATION OPTIONS

Navigate to the STOCHASTIC | GENERALIZED LINEAR MODEL | PARAMETERIZATION area of the **Navigation Pane.**

Index adjustment

When assigning new parameters (step 9 above), you have the option to identify the type of parameter that you are adding to the periods selected on the residual plots. In addition to specifying the type, there are two additional options for each added parameter.

For any **Index** type variables, you can specify an additive adjustment to the index (which applies prior to applying any transformations of the index). For example, if exposure period trend exists within your data within the 5th through 10th exposure period, assigning an **Index** type variable without adjustment to those six exposure periods would result in design matrix entries with values between 5 and 10. Applying a negative additive adjustment of 4 ensures that the first period of this trend has a design matrix entry of 1, rather than 5.

For the GLM purposes, the exposure period numbers start at 1 for the first exposure period and increment by 1 for each additional row, irrespective of the length of the calendar periods. Similarly, the development period numbers start at 1 for the first development period and incremental by 1 for each additional column, irrespective of the length of development periods. The calendar period numbers start at 1 in the upper left-hand corner and progress incrementally through subsequent diagonals. For symmetric triangles (where the length of development periods and exposure periods is the same), the calendar period is equal to the development period number less the exposure period number + 1.

For asymmetrical triangles, Arius accounts for the number of development periods by adjusting the first calendar period for each row. In the picture here (an example of an AxQ triangle), for a given cell, the exposure period is the row number, the development period is the column number, and the calendar period is the black value in the cell.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
2	5	6	7	8	9	10	11	12	13	14	15	16				
3	9	10	11	12	13	14	15	16								
4	13	14	15	16												

Apply parameters to subsets of selections

For parameters added from the exposure period plot, you can set which development periods to which the parameter should apply. The reverse functionality exists for parameters added from the development period plot. These options allow for parameters to only apply to a subset of the exposure/development periods, rather than the entire highlighted column.

These parameters can account for phenomena in specific exposure/development period combinations, such as shifts in the book of business, data availability, and payment pattern changes. This functionality also allows the specification of a parameter for a single cell.

Assigning a parameter to a single cell is an alternative to excluding outliers from parameterization. Parameters applied to the row, column, or diagonal containing the cell still apply, but with an additional parameter that prevents the outlier from distorting them. For a basic one-dimensional example, consider the table below:

PERIOD	1	2	3	4	5	6	7	8	9	10
Incremental	10	20	30	40	75	85	70	80	90	100

There is an underlying linear trend, where the incremental payment increases by 10 for each additional period, excluding the 5th and 6th. If we were to parameterize a simple linear regression model without accounting for this outlier, we would arrive at an intercept of 5 and a slope of 10. However, if we add a constant parameter for the 5th period, this results in a zero intercept, a slope of 10, and a coefficient of 25 for the period parameter, which is more reflective of the actual pattern. The slope of the line accounts for the normal growth, and the additional parameter estimates the spike.

REDUCING DATA POINT INFLUENCE WITH WEIGHTS & OUTLIERS

Navigate to the STOCHASTIC | GENERALIZED LINEAR MODEL | PARAMETERIZATION area of the **Navigation Pane.**

When reviewing diagnostics to improve your model fit (step 11 above), you have the option to specify, on a cell-by-cell basis, the amount of weight the cell should have in the parameterization of the model. In the **Weights** triangle, you can adjust this (the default is a level weight for each cell). This triangle allows the user to assign more weight to the most recent observations (though the selection of parameters could nullify the impact of these weights), or to give less influence to points exhibiting large residuals (instead of excluding the points or assigning individual parameters to the cells).

The residual diagnostics, QQ-Plots, or Box-Whisker Plots may identify observations with very large residuals. These may be referred to as outliers and can potentially influence the GLM model by shifting the fitted values away from the bulk of the observed incremental paid losses towards the outlier values. You can remove points from the model by assigning them zero weight.

Any excluded point does not appear on the three residual plot (and other diagnostic) objects. Any point given non-zero weight appears on these plots. As the weights don't impact the visualization of the point itself, you should review the weights array alongside the parameterization objects.

CALIBRATING A TAIL FACTOR MODEL

Navigate to the STOCHASTIC | GENERALIZED LINEAR MODEL | TAIL PARAMETERIZATION area of the **Navigation Pane**.

The Arius GLM allows the parameterization of a model to square the triangle, effectively filling in the lower right-hand corner of the paid loss triangle. If development for the segment under review continues beyond the number of development periods in the Arius project, a separate tail factor model can be calibrated based on the results of the parameterized non-tail model.

The Arius GLM utilizes an exponential decay tail factor model, with an option for the user to provide an array of adjustment factors to tweak the results of the selected exponential decay model manually. The tail model requires the following inputs:

- **Number of Periods of Extrapolation:** The number of periods of development expected beyond the triangle (minimum 2)

- **Number of adjustment nodes:** The number of points in the piecewise linear adjustment to apply to the triangle (minimum 2)

- **Periods to use in the tail:** Specified using checkboxes below the first table, this identifies the development periods to use for the exponential decay model, before the piecewise linear adjustment (minimum 2)

- **Matrix of adjustment nodes:** The X and Y values for a piecewise linear adjustment index that are applied to the exponential decay to arrive at the final tail pattern

If you've parameterized a model, the **Deterministic Completed Triangle** section of the tail factor model should be populated. If not, running diagnostics with your specified parameters, updates, and refreshes this box. This area provides the results of the GLM for the squared triangle. The checkboxes allow you to specify which periods calibrate the tail factor model. The tail model requires two or more columns to be selected.

When your columns are specified, running diagnostics updates the **Tail Model Diagnostics** box in the center of this object. The tail factor diagnostics are:

- the exponential decay rate indicated by the checked columns;

- the tail development factor implied by the exponential decay rate and the number of periods of tail extrapolation; and

- the goodness of fit of the exponential decay curve to the selected columns (measured by R-Squared).

The last two diagnostics depend on the values in the **Piecewise Adjustment Factors** section. The piecewise adjustment factor array allows you to tweak the exponential decay model to be as representative of your expectations of tail development as desired. The array takes inputs for pairs of **Tail Periods** and **Adjustment Points**, which derive a piecewise linear pattern of adjustment factors. The adjustment factors are multiplied by the implied exponential decay to arrive at an adjusted exponential decay tail factor.

When the piecewise adjustment factors are equal to 100%, the model uses the unadjusted exponential decay rate. Similarly, when the adjustment factors are 0%, there are no tail losses in that period.

For example, consider a project in which you wish for there to be five periods of extrapolated development beyond the triangle. First, set the number of periods of extrapolation to 5, select the columns to use in the fit, and run diagnostics. For the first pass, the diagnostics should look something like this.

Tail Model Diagnostics	
Implied Unadjusted Exponential Decay Rate	76.44 %
Implied Unadjusted Cumulative Tail Development Factor	1.230
R-Squared	73.43 %
Implied Adjusted Cumulative Tail Development Factor	1.230
Weighted Average Adjustment Applied Factor	100.00 %

Because we haven't changed any of the piecewise adjustment nodes, the adjusted and unadjusted implied tail factors are the same.

For this example, the first tail period losses would be equal to the last development period incremental paid loss times 76.44%. The second tail periods losses would be equal to the last development periods loss times 76.44%^2, and so on.

These unadjusted figures can be modified using the piecewise adjustment nodes. As a simple example, assume we had an adjustment node for each tail period (in other words, five tail periods and five adjustment nodes). For a simple example, leaving the adjustment points all at 100% gives us the following tail decay pattern:

Tail Period	Unadjusted Decay	Adjustment Factor	Adjusted Decay
1	76.44%	100.00%	76.44%
2	58.43%	100.00%	58.43%
3	44.67%	100.00%	44.67%
4	34.15%	100.00%	34.15%
5	26.10%	100.00%	26.10%

Here the adjusted decay pattern is equal to the product of the unadjusted decay and the adjustment factor. Adjustment factors can be set independently for each tail period, or for specific tail periods, in which case linear interpolation derives the adjustment factor for interim periods. In the example below, the adjustment nodes are:

Piecewise Adjustment Factors		
Bend Point	Tail Period	Adjustment Point
1	1	100.00 %
2	3	50.00 %
3	5	0.00 %

The corresponding adjustment factors and decay patterns are:

Tail Period	Unadjusted Decay	Adjustment Factor	Adjusted Decay
1	76.44%	100.00%	76.44%
2	58.43%	75.00%	43.83%
3	44.67%	50.00%	22.33%
4	34.15%	25.00%	8.54%
5	26.10%	0.00%	0.00%

In this case, the 5th tail period would not have any losses (due to the multiplication by zero).

The tail factor is only applicable when running simulations and does not have any impact on the diagnostics or non-tail parameterization object.

EXTRAPOLATING TO FUTURE CALENDAR PERIODS

One of the benefits of the Arius GLM over the other stochastic models is the ability to directly parameterize a variable that operates along the calendar period direction (i.e., the diagonal in a typical triangle view). While the residual plots show representations of all exposure periods and all development periods, there are calendar periods within your analysis not represented in these plots. These periods represent the lower-right component of the triangle.

The parameter associated with the most recent calendar period of the historical data (i.e., the last column of the calendar period residual plot) applies to future calendar periods. Any parameter applicable to this column is extended for enough calendar periods to complete the rectangle. The inherent assumption is that any impact that is present within the last diagonal continues similarly through the end of the triangle.

For example, consider a 5x5 triangle with an intercept term and a single parameter, calibrated by selecting all five columns of the calendar period residual plot, and adding a calendar period trend variable. (All numbers illustrative)

Intercept: 10.000   Cal_Trend: -.5000

Then, for each cell in the triangle (including the lower right forecast period), the expected value of the incremental paid losses can be calculated as $\exp(10 - .5CP)$, where CP represents the calendar period of the cell.

The table below shows the triangle (with forecast) for the above parameterization. Note that while the historical experience only included five calendar periods, the resulting triangle uses calendar periods 6 through 9 to complete the lower-right portion of the triangle.

	1	2	3	4	5
1	$exp(10 - .5 * 1)$	$exp(10 - .5 * 2)$	$exp(10 - .5 * 3)$	$exp(10 - .5 * 4)$	$exp(10 - .5 * 5)$
2	$exp(10 - .5 * 2)$	$exp(10 - .5 * 3)$	$exp(10 - .5 * 4)$	$exp(10 - .5 * 5)$	$exp(10 - .5 * 6)$
3	$exp(10 - .5 * 3)$	$exp(10 - .5 * 4)$	$exp(10 - .5 * 5)$	$exp(10 - .5 * 6)$	$exp(10 - .5 * 7)$
4	$exp(10 - .5 * 4)$	$exp(10 - .5 * 5)$	$exp(10 - .5 * 6)$	$exp(10 - .5 * 7)$	$exp(10 - .5 * 8)$
5	$exp(10 - .5 * 5)$	$exp(10 - .5 * 6)$	$exp(10 - .5 * 7)$	$exp(10 - .5 * 8)$	$exp(10 - .5 * 9)$

REVIEWING THE DESIGN MATRIX

While the design matrix for the parameterized model is not viewable in the Arius desktop software, it can be extracted to a CSV file and analyzed in other software of your choosing. The matrix can aid in ensuring that the terms of the model are as expected. To extract the design matrix (as well as the parameter covariance matrix), click on **Model Options** on the ribbon and select **GLM Diagnostics** from the **Save Results to File?** dropdown.

With this option selected, **Run Diagnostics**, which creates a CSV file in the C:\Users\YourUserName\Documents\Milliman\Arius\Sim_Results folder, with the naming convention *ProjectName_SegmentAbbreviation__PaidGLM_Diagnostics.csv.* This file contains the design matrix.

The observations in the CSV file go down the triangle column by column. In other words, for a 10x10 triangle, the first ten rows of the design matrix represent the first development period for the ten exposure periods. The next nine rows represent the second development period for the first nine development periods, and so on.

Below the design matrix is a P by P square, which is the covariance matrix of the **P** parameters specified by the model, provided for informational purposes.

Note that the CSV generates the design matrix as specified by the *selected* parameters, not by the *derived* parameters. For example, if you add two identical parameters, only one of them has coefficients returned, but both appear in the design matrix. This can be easily identified in the covariance matrix, which has rows & columns entirely of zeros for parameters that were *specified* but not utilized in the model.

2. GLM Case Studies

INTRODUCTION

The following case studies are included with the Arius 2020b installation, in
C:\Users*username*\Documents\Milliman\Arius\DemoFiles\GLM_Case_Studies.apj.

These case studies serve as an introduction to potential applications of the GLM framework. The model
parameterized within each case study is not necessarily the optimal model for that dataset, nor are the
models applicable as out of the box solutions to other datasets.

REPLICATING THE ODP BOOTSTRAP MODEL

Several case studies highlight enhancements to the ODP Bootstrap model framework within the GLM
context. This section walks you through reproducing the ODP Bootstrap (Paid Chain Ladder)
methodology within the GLM.

The Overdispersed-Poisson (ODP) Bootstrap Paid Chain Ladder model is a particular case of a GLM with
a Level/Constant parameter for each exposure period and each development period (technically, all
but one exposure period, which serves as the base period). Therefore, replicating the ODP Paid Chain
Ladder within the Arius GLM is very straightforward:
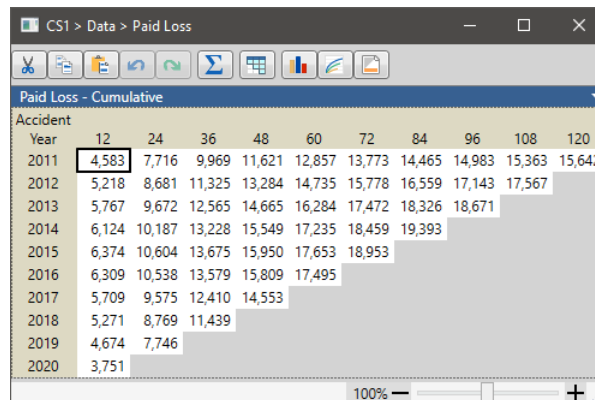
- Quick-add parameters on the exposure period plot,

- Quick-add parameters on the development period plot,

- Remove the first development year parameter on the parameter review table; and

- Remove the intercept parameter, if applied.

- Run Diagnostics.

These steps replicate the ODP Paid Chain Ladder model (in terms of model specification). Simulated
results and degree of variability differ slightly, as the GLM model relies on parameter resampling for
simulation, while the ODP models utilize residual resampling.

CS1 - ENHANCING YOUR ODP BOOTSTRAP MODEL BY MODELING CATEGORICAL VARIABLES AS CONTINUOUS (PARAMETER REDUCTION)

The typical ODP Bootstrap model utilizes a large number of parameters, one for each exposure period and one for each development period (less one). For a 10x10 triangle, this amounts to 19 parameters for the 55 data points. In some situations, the parameters implicit in the ODP Bootstrap model are not statistically significant, which implies a better model can be developed simply by excluding the parameter. In other situations, simple continuous functions exist that closely approximate the parameters, which enables parameter reduction.

Data:
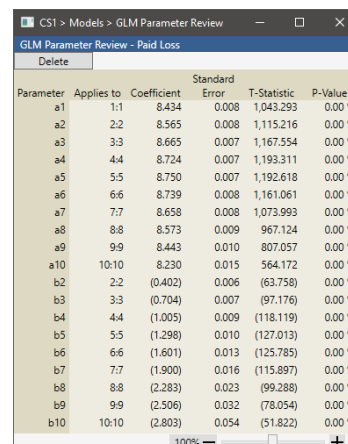
Analysis:

To get insight as to the value of the underlying parameters of the ODP bootstrap model, we first need to reproduce it in the Arius GLM framework, as described above.

Now, running diagnostics provides the parameters associated with the replicated ODP Bootstrap model, shown to the right.

Users interested in reducing parameters in this model may take the natural next step of plotting the coefficients, to see if any can be bucketed, or otherwise reduced. Plotting the exposure period parameters (in external software) shows that the ten individual parameters are well represented by a parabola, which means that we can likely replace these ten parameters with three (a quadratic equation).

To make this adjustment, first, remove all ten exposure period parameters on the **Parameter Review** table. Next, navigate to the **Exposure Period Residual Plot.** Highlight all of the columns of the plot, and click **New Parameter.** The first new parameter's coefficient applies to the square of the column number. We'll name this parameter "SQParam" to inform future users as to the intent of this parameter. This parameter is of type **Index Squared**, which means that the values in the design matrix are equal to the exposure period number squared. No index adjustment is necessary, and this parameter should apply to all development periods.

Next, we'll repeat this process, but add an **Index** parameter, named "Param." This parameter is similar to the "SQParam" just added. However, it is a parameter of type **Index**, which means the values in the design matrix are equal to the exposure period number. Again, no index or range adjustments are necessary.

The final component of the parabola is the intercept term, we repeat the process one more time, adding a parameter of type **Level/Constant**, which means the values in the design matrix are equal to one. Adding this parameter is identical to navigating to **GLM Assumptions** and checking the **Use Intercept** button. Whichever your approach, click **Run Diagnostics**. Reviewing the **Parameter Review** table shows the coefficients of these new parameters:

Parameter	Applies to	Coefficient	Standard Error	T-Statistic	P-Value
b2	2:2	(0.401)	0.006	(61.839)	0.00 %
b3	3:3	(0.704)	0.007	(94.335)	0.00 %
b4	4:4	(1.005)	0.009	(114.359)	0.00 %
b5	5:5	(1.296)	0.011	(122.661)	0.00 %
b6	6:6	(1.601)	0.013	(121.723)	0.00 %
b7	7:7	(1.901)	0.017	(112.217)	0.00 %
b8	8:8	(2.284)	0.024	(96.029)	0.00 %
b9	9:9	(2.504)	0.033	(75.452)	0.00 %
b10	10:10	(2.797)	0.056	(50.076)	0.00 %
aSqTerm	1:10	(0.020)	0.000	(50.166)	0.00 %
aLinearTerm	1:10	0.199	0.004	48.621	0.00 %
aConstTerm	1:10	8.249	0.010	844.123	0.00 %

This parameterization effectively replaces the unique intercept parameters for each exposure period with a quadratic equation:

$$ExposurePeriodFactor = -0.020k^2 + 0.199k + 8.249$$

Comparing the **Model Fit Diagnostics** before and after this adjustment below, we see an improvement in model fit (based on AIC, BIC, and GCV metrics), using fewer parameters, making this parabolic parameter replacement a change worth incorporating into the model.

BEFORE PARAMETER REDUCTION   **AFTER PARAMETER REDUCTION**

GLM Fit Diagnostics - Paid Loss	
Metric	Values
Parameters	19
Degrees of Freedom	36
Scale Parameter/Dispersion	0.798
Mean Squared Error	1,748.434
Unscaled Deviance	28.607
Akaike Information Criterion	586.840
Bayesian Information Criterion	624.979
Generalized Cross Validation	7,840.730
Sum of Squared Errors	62,943.636
Regression Sum of Squares	157,291,269.219
Total Sum of Squares	157,456,350.109
F-Statistic	630,330.930
F-Test P-Value	0.00 %
Ordinary R^2	99.96 %
Adjusted R^2	99.94 %

GLM Fit Diagnostics - Paid Loss	
Metric	Values
Parameters	12
Degrees of Freedom	43
Scale Parameter/Dispersion	0.857
Mean Squared Error	2,524.629
Unscaled Deviance	36.685
Akaike Information Criterion	580.918
Bayesian Information Criterion	605.006
Generalized Cross Validation	4,857.481
Sum of Squared Errors	108,559.061
Regression Sum of Squares	157,203,832.037
Total Sum of Squares	157,456,350.109
F-Statistic	929,249.825
F-Test P-Value	0.00 %
Ordinary R^2	99.93 %
Adjusted R^2	99.91 %

Parameter reduction can also similarly be accomplished by expressing predictor variables as linear and exponential (using the natural log adjustment) functions in a similar manner to that described above.

CS2 - ENHANCING YOUR ODP BOOTSTRAP MODEL BY INCORPORATING CALENDAR YEAR EFFECTS (SUDDEN CHANGE)

The ODP Bootstrap model sets parameters for each Development Period and each Exposure Period. However, there may be instances in which changes in claims payment (either amount, timing, or both) do not follow either of these temporal directions, but instead follow the calendar year direction, effectively creating a dependency between exposure period and development period.

Attempting to parameterize and use an ODP model that does not account for these types of changes can introduce omitted variable bias in your reserve or distribution estimate. This example highlights a situation where there is a sudden change in the paid triangle that persists for multiple exposure periods.

CS2 > Data > Paid Loss

Paid Loss - Incremental

Accident Year	12	24	36	48	60	72	84	96	108	120
12-2000	3,011	2,540	2,253	1,808	1,480	1,030	860	733	606	512
12-2001	3,641	3,072	2,752	2,164	1,506	1,283	1,094	895	740	
12-2002	4,064	3,463	3,071	2,075	1,681	1,404	1,173	970		
12-2003	3,011	2,489	1,896	1,568	1,283	1,020	895			
12-2004	6,568	4,901	4,135	3,352	2,772	2,247				
12-2005	6,750	5,869	5,102	4,219	3,420					
12-2006	3,856	3,157	2,800	2,247						
12-2007	6,108	5,364	4,617							
12-2008	3,704	3,253								
12-2009	6,956									

Analysis:

As with the first case study, we reproduce the ODP Bootstrap parameters in the GLM framework.

On the surface, this appears to be a reasonable model. However, investigation of the residual plots and the Actual to Expected triangles (either for the ODP module or its recreation within GLM) reveals that we're potentially missing something with this model. Looking at the calendar year plot (to the right) reveals some concerns:
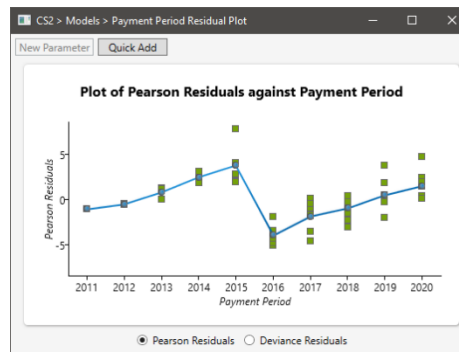
There is some trend in the residuals for the first few exposure periods, which then repeats itself five periods into the analysis. This plot is a visible sign of some calendar year activity. The actual to expected triangle can provide some additional insight. The ODP model actual to expected shows a clear trend upwards for the first 5 diagonals, then a sudden drop reversion, followed by a subsequent upward trend. Perhaps a new system was implemented after 5 calendar years to attempt to correct for the upward trend.

Regardless of the cause, we account for this by implementing a regime variable. This variable serves as a flag to identify the regime in which each cell is located. To do this, we use the payment period plot to identify the periods in which the new regime exists. In the above plot, this refers to calendar periods 2005 and subsequent[1]. On the payment period plot, we highlight these five columns and add a new **Level/Constant** parameter. Here, we call this "Regime2."
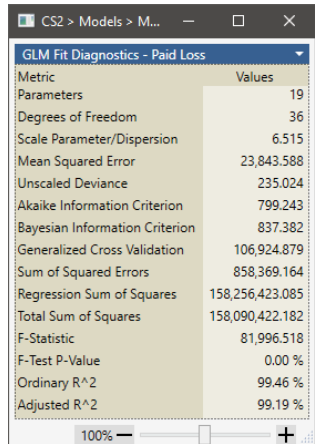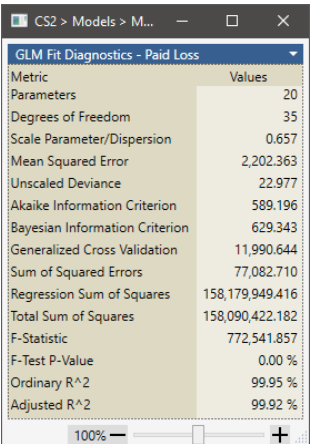
There is no need to add a parameter for the first regime. The logic is that the ODP model applies to this entire triangle after we apply an adjustment factor to the last five periods to account for the calendar period affects. If we were to add a level/constant parameter for the first five periods, the results of the model would be similar, with one additional parameter, indicative of a weaker model.

Reviewing the calendar year residual plot, we see a more favorable dispersion of residuals.

---
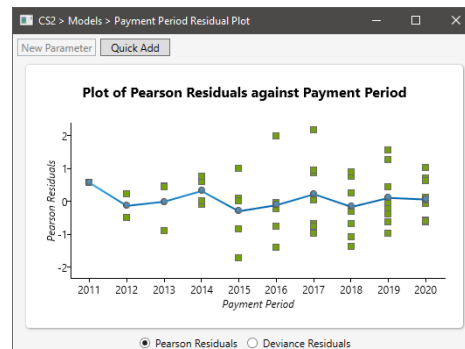
[1] Recall that any parameter applying to the last column of the payment period plot will be assumed to continue for the remainder of the development periods in the triangle.

A review of the model fit diagnostics before and after the introduction of this variable shows an improvement in fit, despite the additional parameter.

BEFORE REGIME VARIABLE	AFTER REGIME VARIABLE

GLM Fit Diagnostics - Paid Loss (BEFORE REGIME VARIABLE):

Metric	Values
Parameters	19
Degrees of Freedom	36
Scale Parameter/Dispersion	6.515
Mean Squared Error	23,843.588
Unscaled Deviance	235.024
Akaike Information Criterion	799.243
Bayesian Information Criterion	837.382
Generalized Cross Validation	106,924.879
Sum of Squared Errors	858,369.164
Regression Sum of Squares	158,256,423.085
Total Sum of Squares	158,090,422.182
F-Statistic	81,996.518
F-Test P-Value	0.00 %
Ordinary R^2	99.46 %
Adjusted R^2	99.19 %

GLM Fit Diagnostics - Paid Loss (AFTER REGIME VARIABLE):

Metric	Values
Parameters	20
Degrees of Freedom	35
Scale Parameter/Dispersion	0.657
Mean Squared Error	2,202.363
Unscaled Deviance	22.977
Akaike Information Criterion	589.196
Bayesian Information Criterion	629.343
Generalized Cross Validation	11,990.644
Sum of Squared Errors	77,082.710
Regression Sum of Squares	158,179,949.416
Total Sum of Squares	158,090,422.182
F-Statistic	772,541.857
F-Test P-Value	0.00 %
Ordinary R^2	99.95 %
Adjusted R^2	99.92 %
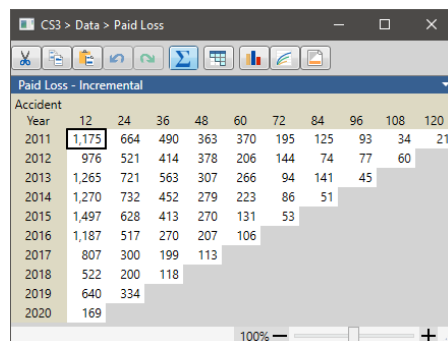
CS3 - ENHANCING YOUR ODP BOOTSTRAP MODEL BY INCORPORATING CALENDAR YEAR EFFECTS (GRADUAL CHANGE)

The previous example introduced a treatment of calendar year impacts with a binary application, either a cell was adjusted (in the new regime), or it wasn't. However, in many situations, the calendar year impact is not an all or nothing venture.

For example, new claims systems can take many calendar periods to propagate to the entire triangle. While including a binary flag for these types of system changes is better than ignoring them, they may not be the best approach for accounting for them.

This case study showcases a loss triangle with a gradual slowdown in incremental paid losses, introduced in the 5th calendar period, and continuing through the end of the triangle.
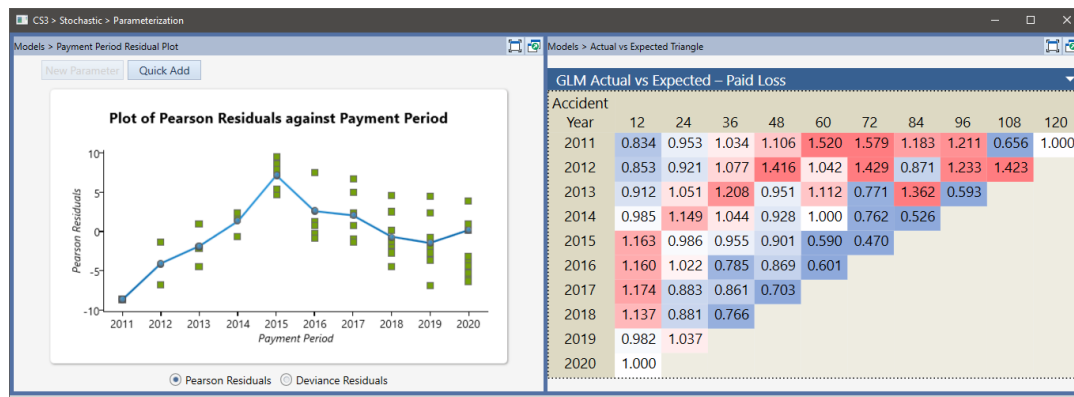
Data:

Paid Loss - Incremental

| Accident Year	12	24	36	48	60	72	84	96	108	120
2011	1,175	664	490	363	370	195	125	93	34	21
2012	976	521	414	378	206	144	74	77	60	
2013	1,265	721	563	307	266	94	141	45		
2014	1,270	732	452	279	223	86	51			
2015	1,497	628	413	270	131	53				
2016	1,187	517	270	207	106					
2017	807	300	199	113						
2018	522	200	118							
2019	640	334								
2020	169									

Analysis:

After replicating the ODP bootstrap model framework, the calendar year residual plot and the actual to expected triangle showcase a similar phenomenon to the previous case study:

In this case, after payment year 2015, we see a continual decrease in the actual versus expected, after an initial ramp-up from payment year 2011 through 2015. First, we'll add a "Regime2" variable. We add this based on the data observed, but parameters could also be added based on outside information. Say the claims department implemented a payment slowdown initiative starting in 2016. If this was the case, it may make sense to introduce parameters a priori, and remove them if not statistically significant.

First, we'll introduce this parameter as a **Level/Constant** parameter, as we did in the previous regime change case study:
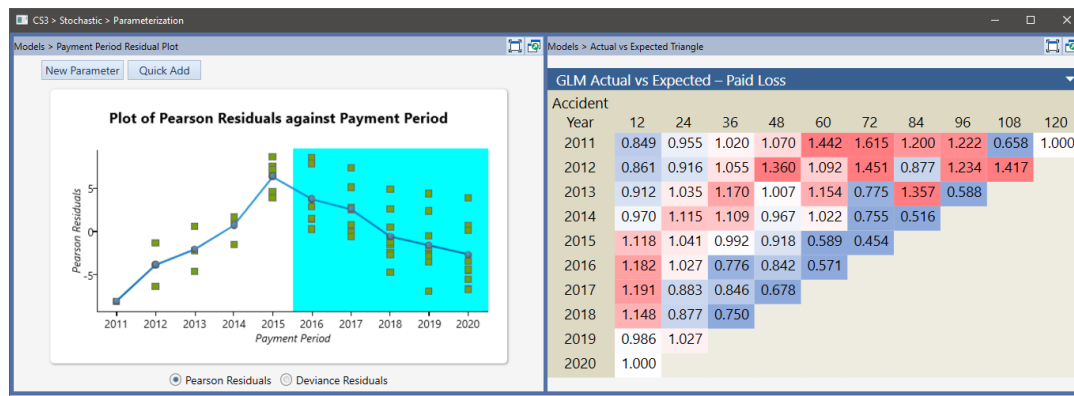
As shown in the above plot and table, this new parameter did not materially impact the payment period residual plot, and the Actual vs. Expected is virtually unchanged as well. Reviewing the **Parameter Review Table**, we notice that the constant regime change parameter is not statistically significant. So, we delete this parameter and account for the calendar year impact in another way.

Rather than having a 1/0 flag (in the design matrix) for the updated regime, it may make sense for the first period of the new regime would receive a 1, the second, a 2, continuing through the highlighted periods. Solving for the parameter in this situation would mimic the introduction of a regime over some time.

We highlight the 6th through the 10th. Click **Add Parameter**. Here, we name the parameter Regime2IP and select **Index**.

At this point, we could click **OK**. However, this would introduce the parameter Regime2IP starting at calendar period 6, with a factor of 6 x coefficient (in other words, design matrix entries for this parameter would start at 6 in the appropriate period). To adjust this to reflect the first period of the new regime, we turn on the negative adjustment and shift by five periods (to start at 1):

(Note that the **Index Squared** and **LN(Index)** options allow further manipulation of the design matrix entries, either squaring the adjusted index value or taking the natural log, respectively)

Running diagnostics gives us our adjusted model parameters. Reviewing the calendar year plot and AvE once more, we see a much better fit (though as noted earlier, not likely an optimal model for this dataset):

Finally, we review the model fit diagnostics, which indicate a better fit with this updated model, despite the increase in the number of parameters
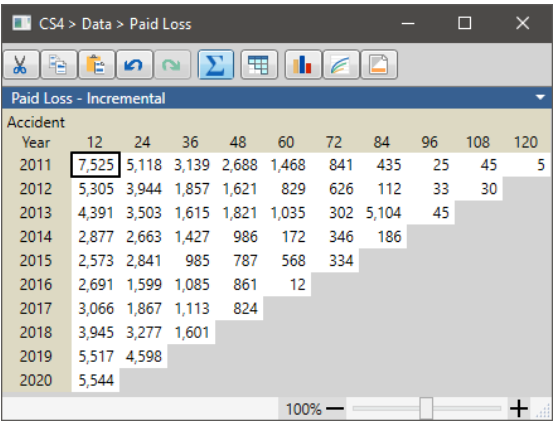
BEFORE REGIME VARIABLE	AFTER REGIME VARIABLE

GLM Fit Diagnostics - Paid Loss (Before)

Metric	Values
Parameters	19
Degrees of Freedom	36
Scale Parameter/Dispersion	17.607
Mean Squared Error	8,033.222
Unscaled Deviance	637.834
Akaike Information Criterion	1,081.120
Bayesian Information Criterion	1,119.259
Generalized Cross Validation	36,024.415
Sum of Squared Errors	289,195.999
Regression Sum of Squares	7,003,921.939
Total Sum of Squares	7,056,502.937
F-Statistic	2,617.102
F-Test P-Value	0.00 %
Ordinary R^2	95.90 %
Adjusted R^2	93.85 %

GLM Fit Diagnostics - Paid Loss (After)

Metric	Values
Parameters	20
Degrees of Freedom	35
Scale Parameter/Dispersion	4.672
Mean Squared Error	910.973
Unscaled Deviance	160.177
Akaike Information Criterion	605.463
Bayesian Information Criterion	645.610
Generalized Cross Validation	4,959.744
Sum of Squared Errors	31,884.067
Regression Sum of Squares	7,052,683.959
Total Sum of Squares	7,056,502.937
F-Statistic	9,400.984
F-Test P-Value	0.00 %
Ordinary R^2	99.55 %
Adjusted R^2	99.30 %

CS4 - ENHANCING YOUR ODP BOOTSTRAP MODEL BY ACCOUNTING FOR "SHOCK" CLAIMS/OUTLIERS

Within the ODP Bootstrap model, significant "shock" losses can be excluded from the model parameterization by excluding the cell in which the loss occurs. However, this is not always the ideal treatment of these losses, and care must be taken to understand the ramifications of completely excluding a data point from model calibration.

The Arius GLM tool offers two different approaches to handling these types of losses. The first approach is similar to the ODP approach, merely excluding the point from model calibration. The second is unique to the Arius GLM and can provide additional flexibility into your stochastic models.

Data:

Paid Loss - Incremental

Accident Year	12	24	36	48	60	72	84	96	108	120
2011	7,525	5,118	3,139	2,688	1,468	841	435	25	45	5
2012	5,305	3,944	1,857	1,621	829	626	112	33	30	
2013	4,391	3,503	1,615	1,821	1,035	302	5,104	45		
2014	2,877	2,663	1,427	986	172	346	186			
2015	2,573	2,841	985	787	568	334				
2016	2,691	1,599	1,085	861	12					
2017	3,066	1,867	1,113	824						
2018	3,945	3,277	1,601							
2019	5,517	4,598								
2020	5,544									

Analysis:

Notice the substantial payment in the 84th development month of Accident Year 2013. This point is an outlier. In a deterministic model, it is highly unlikely that you would include the loss development

factor implied by this period in your statistics. Within the ODP Bootstrap model, you can attempt to adjust the model for this using outliers. The GLM model allows for additional functionality.

As with previous case studies, we'll start by replicating the ODP Bootstrap model. Next, we use the **Weights** object to assign no weight to this significant outlier point (2002, 84 months of development). This approach is similar to assigning the point as an outlier in the ODP model (see the table at the end of this section)

An alternative approach would be to allow the attritional losses for this data point to be allowed to influence the model separately than the shock claim. By assigning a parameter unique to the cell containing this large payment, the Arius GLM can accommodate this.

First, return each point to equal weights in the **Weights** object. Then, in the development year plot, highlight the 84th month of development, and click **New Parameter**. This parameter is **Level/Constant**, named "BulkLoss." However, we don't want to apply this parameter to the entire period. Use the **Specify Exposure Periods** feature to limit this to the third exposure period (note, the same parameter can be specified from the Exposure Period plot, where you'd identify specific development periods):

Now, for that cell of the triangle, the model projects incurred losses as a function of the exposure date parameter, the development age parameter, and a "bulk loss" parameter.

Running diagnostics here shows a similar model to the zero weight model, as one would reasonably expect. This "Bulk Loss" adjustment is extensible to any contiguous combinations of exposure and development periods, making it a better option if these "bulk" payments persist for multiple periods.

While the table shows the addition of the bulk weight parameter is not necessarily a better model than the zero weight situation, it does provide additional value, specifically about the cell that contained the outlier.

In the **Equal-Weight to All Points** scenario, the incremental paid losses for the cell in question is predicted by an accident year and exposure period parameter:

$$\exp(a3 + b7) = \exp(8.664 - 5.126) = 1,676$$

In the zero weight scenario, this projection is:

$$\exp(a3 + b7) = \exp(8.425 - 3.058) = 214$$

Finally, in the "BulkLoss" scenario, we have:

$$\exp(a3 + b7 + BulkLoss) = \exp(8.425 - 3.058 + 3.171) = 5,104$$

Of course, because we're specifying this parameter specifically on one point, this accurately replicates the empirical paid loss triangle. However, it provides additional insight, using the first two parameters. Because these parameters are calibrated on the entire row (or column) of the triangle, they provide an estimate for the cell, ignoring the impact of the bulk loss. This estimate is equal to the value for the cell in the zero-weight scenario.

Here, the "BulkLoss" scenario can be split into attritional and large losses:

$$\exp(a3 + b7) + \exp(BulkLoss) = \exp(8.425 - 3.058) + \exp(3.171) = 214 + 4,890 = 5,104$$

The 4,890 adjustment for the substantial loss can be compared to actual claim records.

EQUAL WEIGHT TO ALL POINTS	ZERO WEIGHT TO OUTLIER	BULKLOSS PARAMETER

EQUAL WEIGHT TO ALL POINTS

CS4 > Models > M...   —   □   ✕

GLM Fit Diagnostics - Paid Loss   ▼

Metric	Values
Parameters	19
Degrees of Freedom	36
Scale Parameter/Dispersion	347.135
Mean Squared Error	628,431.599
Unscaled Deviance	11,986.623
Akaike Information Criterion	12,498.895
Bayesian Information Criterion	12,537.034
Generalized Cross Validation	2,818,156.927
Sum of Squared Errors	22,623,537.552
Regression Sum of Squares	162,255,619.050
Total Sum of Squares	180,173,354.109
F-Statistic	1,042.658
F-Test P-Value	0.00 %
Ordinary R^2	87.44 %
Adjusted R^2	81.17 %

100% — ▭ +

ZERO WEIGHT TO OUTLIER

CS4 > Models > M...   —   □   ✕

GLM Fit Diagnostics - Paid Loss   ▼

Metric	Values
Parameters	19
Degrees of Freedom	35
Scale Parameter/Dispersion	49.553
Mean Squared Error	68,928.598
Unscaled Deviance	2,091.755
Akaike Information Criterion	2,593.651
Bayesian Information Criterion	2,631.442
Generalized Cross Validation	329,834.112
Sum of Squared Errors	2,412,500.932
Regression Sum of Squares	171,173,594.336
Total Sum of Squares	170,106,382.537
F-Statistic	6,996.751
F-Test P-Value	0.00 %
Ordinary R^2	98.58 %
Adjusted R^2	97.85 %

100% — ▭ +

BULKLOSS PARAMETER

CS4 > Models > M...   —   □   ✕

GLM Fit Diagnostics - Paid Loss   ▼

Metric	Values
Parameters	20
Degrees of Freedom	35
Scale Parameter/Dispersion	49.553
Mean Squared Error	68,928.598
Unscaled Deviance	2,091.755
Akaike Information Criterion	2,606.027
Bayesian Information Criterion	2,646.173
Generalized Cross Validation	375,277.923
Sum of Squared Errors	2,412,500.932
Regression Sum of Squares	181,240,565.908
Total Sum of Squares	180,173,354.109
F-Statistic	7,022.322
F-Test P-Value	0.00 %
Ordinary R^2	98.66 %
Adjusted R^2	97.93 %

100% — ▭ +

CS5 - REPRODUCING TAYLOR CHAPTER 7 WITH THE ARIUS GLM

Within this case study, we use the Arius software to reproduce the Generalized Linear Model found in the CAS Monograph #3--*Stochastic Loss Reserving Using Generalized Linear Models* by Taylor & McGuire[2] (ultimately, we reproduce formula 7-10 within the case study[3]). While the diagnostics available within Arius are not identical to those found in the Taylor monograph, there is enough insight available to walk through this example from data to results.

Data:

The data used for this case study is the same as that found in Taylor. *The triangle has been obtained from the database of Meyers and Shi[4] (2011).* The triangle represents workers' compensation data from the New Jersey Manufacturers Group. The exposure periods in the case study do not match those from the database.

---

[2] *Stochastic Loss Reserving Using Generalized Linear Models*, Greg Taylor and Gráinne McGuire, 2016
  https://www.casact.org/pubs/monographs/papers/03-Taylor.pdf

[3] The WC segment of the Arius_Stochastic_Sample.apj file reproduces Taylor formula 7-14 (Table 7-5).

[4] Meyers, G. G., and P. Shi. 2011. *The Retrospective Testing of Stochastic Loss Reserve Models.* Casualty Actuarial Society E-Forum, Summer
  2011. http://www.casact.org/ pubs/forum/11sumforum/Meyers-Shi.pdf.

Analysis:

Taylor's Chapter 7 builds on what he calls the Cross Classified model, with a parameter for each accident period and each exposure period. The Cross Classified model is effectively the ODP bootstrap model. To reproduce this table, we first **Run Diagnostics** with an intercept only model to activate the residual plots, then **Quick Add** parameters on both the Exposure Period and Development Period plots.

To reproduce this exactly, we elect to alias the first development period parameter, by deleting the "b1" parameter from the **Parameter Review table**. If we did not make this selection, results would be similar, but Arius would select which parameter to alias. In the interest of replication, we make this determination in advance.

Next, run diagnostics. After the run is completed, the **GLM Parameter Review** table exactly matches Table 5-1 in the Taylor monograph (see screenshot to the right).

Taylor's next step is to plot the exposure period parameters (those prefixed with an "a" against the exposure period. As with our first case study above (Parameter Reduction), these exposure period parameters can be approximated by a quadratic curve.

We make the same replacement as we did in the first case study. First, remove all ten exposure period parameters on the **Parameter Review** table. Next, navigate to the **Exposure Period Residual Plot.** Highlight all of the columns of the plot, and click **New Parameter.** The first new parameter's coefficient is applied to the square of the column number. We'll name this parameter "SqTerm" to inform future users as to the intent of this parameter. The parameter is of type **Index Squared**, which means that the values in the design matrix are equal to the exposure period number squared. No index adjustment is necessary, and this parameter should apply to all development periods.

Next, we'll repeat this process, but add an **Index** parameter, named "LinearTerm." This parameter is identical to the "SqTerm" just added, but with the parameter type **Index**, which means the values in the design matrix is equal to the exposure period number. Again, no index or range adjustments are necessary.

Finally, the Taylor paper forgoes the inclusion of an intercept term on this quadratic, electing instead to restore "b1" as a free parameter (remember we had aliased this parameter above). We do this by clicking the **Quick Add** button on the development period residual plot.

Then, click **Run Diagnostics**. Reviewing the **Parameter Review** table shows the coefficients of these new parameters. The model represented by Taylor's formula 7-2 has coefficients, as seen in the screenshot to the left.

The next enhancement Taylor makes to the model is to apply similar parameter reduction techniques to the development year parameters.

Taylor notes that a linear spline (with a knot at 7.5) well represents these parameters. The R^2 value of a linear fit to the first seven parameters is 99.4%, and the R^2 of a linear fit to the last three parameters is 99.9%. Fitting a line to all ten parameters (removing the knot) yields an R^2 of 97.7%. To reproduce Taylor, we replicate the spline, which is defined by two parameters:

$$b_x \cong \begin{cases} SplineA(x-1) \text{ for } x \leq 7.5 \\ SplineB(x-7.5) \text{ for } x > 7.5 \end{cases}$$

Building this spline into the Arius GLM requires two development year parameters. The first applies to the first seven development periods, and the second applies to the remaining three. As in Taylor, we parameterize the second spline as being an adjustment to the first. Formulaically, this is:

$$b_x \cong \begin{cases} SplineA(x-1) \text{ for } all\ X \\ SplineA(x-1) + \ SplineB(x-7.5) \text{ for } x > 7.5 \end{cases}$$

To parameterize the first, highlight all ten columns of the development period plot, and click **New Parameter**. This new parameter (Spline A) is an index parameter, where the first highlighted column receives an entry of 1 in the design matrix, with each subsequent column increasing by 1. However, to force this parameter to start at zero (for the first column), we perform a negative index adjustment of size 1.
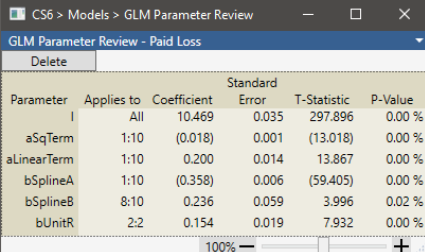
For Spline B, we perform a similar mechanic, first highlighting the last three columns of the development period plot, and adding a new index parameter with a negative index adjustment of size 7.5.
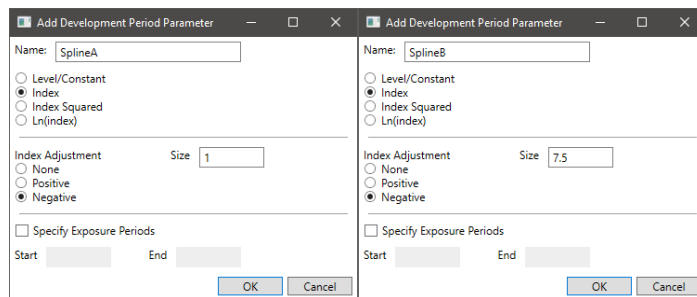
Next, Taylor introduces the intercept term at this point. Navigate to the **GLM Model Assumptions** window and check the **Use Intercept** box. Taylor also includes a unit regressor (in development period 2) in the model. In Arius, this can be done by adding a parameter to development period 2 by highlighting that column of the **development period plot**, clicking **Add New**, and naming a level/constant parameter.

Lastly, we delete the individual development year parameters from the **Parameter Review** table and **Run Diagnostics**.

We now have a model that replicates Taylor's Formula 7-10, and the parameters within Table 7-3:

3. Frequently Asked Questions

Why do some of my specified parameters have zero coefficients and blank goodness of fit measures?

This situation is caused by duplication or redundancies in the parameters. The Arius GLM performs automatic aliasing using QR factorization with the Householder transform to determine how many independent columns there are in the design matrix and then fits that many parameters.

Why do I need two-tail periods and adjustment nodes for my tail factor model?

The piecewise adjustment fit requires two or more points for proper specification. For a one-period tail model, set the number of periods of extrapolation to two and set the second adjustment node to zero. A tail set in this manner results in one (non-zero) period of extrapolated results beyond your triangle.

Why did I receive an error message stating "Iteration limit is reached. The model did not converge. The validity of the model fit is questionable"?

The flexibility of the Arius GLM introduces the possibility of developing a model that does not converge within the number of permutations performed by the system. This can happen in the rare case when there is a parameter assigned to a range of cells all containing an incremental paid loss of 0. When you receive this error, your model requires simplification. Ensure that all specified parameters apply to ranges of cells containing at least one non-zero incremental paid loss.

What happens when I modify structure (e.g., roll-forwards, compression)?

As of Arius 2020b, the GLM model is cleared whenever the underlying data structure of the model is changed.

Can I weight the GLM with existing models? Can I weight multiple GLMs together?

While the GLM model is parameterized independently of the other stochastic models, and users are restricted to a single GLM in a given segment (at a time), these tasks can both be accomplished using the Arius Stochastic Compilation Module. Results from one (or more) GLMs can be saved to a CSV file using the **All Incrementals, by Iteration** option, which generates a file formatted for use in the compilation module.

How does the Arius GLM account for a partial last diagonal or a shorter first development age?

Arius automatically accounts for these situations *when simulating*. However, it does not account for these when parameterizing. Arius requires you to set a **Level/Constant** parameter that applies only to the last diagonal in the event of a partial last diagonal. Arius does not require a separate parameter for a short first development age. However, not including one causes a shorter incremental payment period to be treated the same as the remainder of the triangle, so it is highly recommended.

How does the Arius GLM account for the All Prior row?

Arius does not distinguish between the all prior row and the remainder of the triangle within the

GLM. Therefore, care must be taken to ensure any all prior row does not adversely influence the model fit. The prior row can be accommodated by excluding the all-prior row (by assigning those points zero weight), or by specifying parameters unique to the all prior row.

I assigned zero weight to points in my triangle; why do I still see residuals for that cell?

Unlike the ODP model, where excluding a point results in no residual being displayed and calculated for an excluded cell, the GLM calculates a residual for each point in the triangle, as a function of the parameters specified. When excluding points, they are being excluded from the derivation of the parameters, and the parameters specified generates a predicted value for that cell.

Can I use the GLM as a Deterministic Method?

While the Arius GLM is only available to Arius Stochastic licensees, these users can utilize the GLM as an indication of ultimate loss for deterministic reserving. Simply calibrate your GLM as usual and copy the **Ultimate Loss** column from the **Deterministic Calculations** object into a User Defined Loss Method, and incorporate the GLM within your **Comparison of Ultimate Loss** reports.

Why do empty triangle cells result in two empty residual/fitted cells within the GLM diagnostics?

This situation is caused by the conversion of a cumulative loss triangle to an incremental loss triangle within the calculation engine. If a cumulative data point is empty, the subsequent populated cell is not assumed to represent an actual incremental paid loss value, which is what the GLM model is framed around.

Why doesn't my RSS, AIC, or BIC match the ODP model when replicating the parameterization?

The two model families rely on different calculations for these components. The RSS in the ODP Bootstrap model is used to determine how well the residuals fit a normal distribution, not how well the model fits the data. Within the GLM, the RSS measures the fit to the data.

The AIC/BIC differences are also due to this difference, the RSS used in the AIC/BIC calculations under the ODP model determines how well the residuals fit a normal distribution. Again, the GLM AIC/BIC measures the fit to the data.

Why do I see 1.0 or some negative value in my Fitted Incrementals triangle?

If you see unrealistic values in your fitted incremental triangle (1's or negative numbers), it is likely due to there being no parameter assigned to that cell of the triangle. Users should take care to ensure that each cell has at least one parameter applied to it, and reviewing the **Fitted Incrementals** object is a quick way to see if a parameter may be missing.

How can I validate my model?

In a triangular reserving model, it is impossible to perform out of sample testing because all of the data is used in model calibration. The Model Diagnostics object provides a summary of the model's goodness of fit. For those looking for a more traditional GLM validation tool, the Generalized Cross-Validation metric provides an approximation for leave-one-out K-fold cross-validation.