Reserving with Machine Learning: Applications for Loyalty Programs and Individual Insurance Claims

Len Llaguno, FCAS

Manolis Bardis, FCAS

Robert Chin

Tina Gwilliam, FCAS

Julie Hagerstrand, FCAS

Evan Petzoldt, ACAS

Abstract

Motivation. Reserving is typically performed on aggregate claim data using familiar reserving techniques such as the chain ladder method. Rich data about individual claims is often available but is not systematically used to estimate ultimate losses. Machine learning techniques are readily available to unlock the benefits of this information, potentially resulting in more accurate reserve estimates.

Method. In this paper we introduce a reserving framework that leverages machine learning to incorporate rich granular information that is not captured when data is analyzed at the aggregate level. The framework relies on the snapshot date triangle as the format for organizing the data, which enables us to incorporate all available information in the prediction of ultimate values. A mix of machine learning algorithms is applied to the snapshot date triangle to create segments of claims with homogeneous development patterns. Standard triangular methods can then be applied on each segment to estimate the ultimate values.

Results. This method was developed in the context of reserving for loyalty programs. Within the loyalty context, reserving refers to estimating future redemption patterns for points issued to-date, producing an estimate of the loyalty program liability. We show how this framework can be used to create segments of members with homogeneous redemption behaviors, which facilitates the reserving exercise.

Conclusions. We see a clear analogy between a loyalty program member's redemption pattern and a claim payment pattern. Consequently, the applicability of this framework for loyalty program reserving suggests there may be an opportunity to apply this framework for insurance claim reserving.

Keywords. Data Mining, Predictive Modeling, Reserving Methods, Individual Claims Reserving, Claims Triage, Loyalty Program Liability, Data Organization, Ultimate Redemption Rate, Breakage Estimation, Snapshot Triangle

---

1. INTRODUCTION

Today, the actuarial reserving exercise is generally performed at the aggregate level by line of business or coverage type (e.g., Auto Liability Bodily Injury vs Property Damage). The rich data associated with individual claims is not typically leveraged in a systematic way to make predictions about future payments. We set out to develop a reserving framework that allows actuaries to leverage this data.

The framework was developed for the application of reserving for loyalty programs. However, there is a strong analogy between this reserving exercise and the more common insurance claim reserving exercise. We will start with a focus on the loyalty context, then make the connection to the insurance context.

Throughout the paper, *granular reserving* will refer to reserving approaches that use granular information (e.g., individual claim/policy information in the insurance context, individual member information in the loyalty program context) as opposed to *aggregate reserving* approaches that use aggregated information.

We stop short of applying the methodology to insurance reserving, but we believe that the applicability of the method in the loyalty context is a good indicator of the applicability in the insurance context.

1.1 Research Context

In the paper "Loyalty Rewards and Gift Card Programs: Basic Actuarial Estimation Techniques" [1], Gault, Llaguno and Menard discuss reserving methods for loyalty programs. All of the methods outlined in this 2012 paper are performed at an aggregate level and do not systematically leverage information at the individual member level.

In the insurance context, there have been several contributions to the literature on reserving using individual claims data. Wüthrich [2] presents a method that leverages machine learning for individual claims reserving. The approach uses both static covariates (i.e., claim characteristics that remain constant throughout the life of the claim, such as reporting delay) and dynamic covariates (i.e., claim characteristics that can change over the life of the claim, such as case reserves or claimant age) as predictors.

In concept, this methodology is flexible, but in practice it can be difficult to implement since it would require the ability to predict dynamic covariates into the future. This can be particularly challenging for what Taylor [3] describes as unpredictable dynamic covariates, which are covariates that change over time in a manner that is not easily predictable (e.g., case reserves).

1.2 Objective

Our research builds on the work done by Wüthrich [2] to illustrate a flexible model for granular reserving that does not require the prediction of dynamic covariates.

1.3 Outline

The remainder of the paper proceeds as follows. Section 2 starts with a description of loyalty programs to set context. Next, the structure of the data used in the analysis is outlined. Finally, the modeling strategy is discussed.

2. BACKGROUND AND METHODS

Our example will be based on the application of this framework to model redemption patterns for loyalty programs. Throughout the paper we will tie the application of this framework to modeling payment patterns for individual insurance claims.

2.1 Loyalty Programs

Loyalty programs are prevalent in many industries. Most people belong to a number of frequent flyer, hotel, retail or credit card loyalty programs. Loyalty programs generally work in the following way:

- A member spends money with the organization sponsoring the loyalty program.
- In return for the purchase, the organization issues a currency to the member. The currency is generally called a point or a mile (we will use the term point in this paper).
- The member can redeem the points for benefits, such as free or discounted flights, hotel stays, or merchandise. The accumulation and redemption of points act as an incentive for the member to be more loyal to the sponsoring organization.

The sponsoring organization is required to recognize on its balance sheet the future obligation for all points issued that remain outstanding at each financial statement date. This is similar to the requirement for insurance companies to hold reserves related to claims incurred during the insured period.

Depending on the accounting treatment, the balance sheet obligation is either shown as a liability or deferred revenue. For simplicity, we will use the term liability to reference this obligation on the balance sheet.

The liability is calculated as:

$$\text{Liability} = [\text{Outstanding Points Balance}] \times \text{URR} \times \text{CPP} \tag{2.2}$$

The URR is the Ultimate Redemption Rate, which represents the expected percentage of points currently outstanding (i.e., issued but neither redeemed nor expired) that will ultimately be redeemed. The CPP is the Cost Per Point, which represents the expected cost to the program for each currently outstanding point that is expected to be redeemed in the future.

At a high level, estimating the URR is analogous to estimating the ultimate losses in the insurance reserving context. That is, we can develop triangles of cumulative redemption percentages to estimate the redemption pattern and URR using standard actuarial reserving techniques. However, if this is done at an aggregate level then we miss an opportunity to use the rich member level information to inform our estimates. A methodology to leverage this data when estimating the URR is discussed in the following sections.

2.2 Data Organization

2.2.1 Definitions

- Snapshot Date: The date at which we define and begin tracking a given cohort. The definition of the cohort can be anything, so long as it is evaluated using data as of the Snapshot Date. In our case, we define the cohort to be a group of points outstanding as of a particular snapshot date.
- Observation Date: A date subsequent to the Snapshot Date at which we observe some characteristic of the cohort being tracked.
- Observation Age: Observation Date - Snapshot Date. In our example, Observation Ages are counted in months.

For example, suppose the cohort we are tracking is outstanding points as of January 31, 2008 and 10% of these points are redeemed as of March 31, 2008. Then we can say, "10% of points outstanding as of snapshot date January 31, 2008 are redeemed by observation age 2, which represents observation date March 31, 2008."

2.2.2 Aggregate snapshot date triangles

In a snapshot date triangle of redemptions, each row represents a snapshot date and the cohort we are tracking is outstanding points. At each observation age, the metric we are observing is the cumulative percentage of these points that have been redeemed. Redemptions are associated with outstanding points using a FIFO (first in first out) allocation, which is a standard industry approach to allocate redemptions to point balances. The example below illustrates a snapshot date triangle of FIFO redemptions. To save space, we are only showing the January snapshot dates starting in 2008. The latest evaluation date in the example is January 31, 2017.

Table 2.1: Example Snapshot Date Triangle

		Observation Age								
Snapshot Date	Outstanding Points (Billions)	12	24	36	48	60	72	84	96	108
1/31/2008	11.0	31.0%	52.0%	60.3%	61.7%	61.9%	61.9%	61.9%	61.9%	61.9%
1/31/2009	12.2	31.7%	53.4%	61.5%	62.3%	63.4%	64.0%	64.0%	64.0%	
1/31/2010	13.7	32.5%	55.3%	64.5%	65.7%	66.9%	67.5%	67.5%		
1/31/2011	15.9	32.9%	57.5%	66.5%	67.6%	68.8%	69.5%			
1/31/2012	17.8	35.9%	58.4%	67.2%	68.3%	69.6%				
1/31/2013	18.7	38.9%	60.4%	69.5%	70.7%					
1/31/2014	19.9	41.9%	65.1%	74.9%						
1/31/2015	20.9	44.9%	69.7%							
1/31/2016	22.2	47.9%								
1/31/2017	23.6									

We can see that 20.9 billion points were outstanding as of January 31, 2015. Of that 20.9 billion points, 44.9% were redeemed by January 31, 2016 and 69.7% were redeemed by January 31, 2017. Since the FIFO approach is used to tie redemptions to outstanding points, these percentages can never exceed 100%.

An analogous snapshot date triangle in the insurance reserving context would track cohorts of open claims, and the metric observed could be payments (in dollars) on those claims. That is, for a given snapshot date, the cohort of claims open as of that snapshot date would be tracked. At each observation age, the metric observed would be the cumulative payments made on those claims since the snapshot date. This would allow for reserving of incurred but not enough reported claims (IBNER), but would not include a provision for incurred but not yet reported (IBNR) claims or for reopened claims. A separate snapshot date triangle tracking cohorts of *policies* could be used to estimate payments on IBNR claims and reopened claims. Alternatively, the calculation of IBNR claims could employ Frequency/Severity techniques to estimate the future pure IBNR emergence and corresponding claim severity. For simplicity, we will focus our discussion of the analogy to insurance reserving on estimating IBNER.

2.2.3 Snapshot date triangles vs. accident/report year triangles

The concept of a snapshot date triangle in the insurance reserving context is different from accident/report year triangles in two main ways:

- Mutually Exclusive Rows: In an accident/report year triangle, each row tracks a mutually exclusive group of claims based on the year in which the claim occurred or was reported. In a snapshot date triangle, each row is likely not mutually exclusive. For example, claims that are open as of snapshot date January 31, 2012 may still be open as of January 31, 2013 and would therefore be included in the cohorts being tracked in both those rows of a snapshot date triangle.
- Ultimate Values: For both accident/report year and snapshot date triangles, ultimate values for each row can be estimated using standard triangular methods (e.g., chain ladder). In an accident/report year triangle, the total ultimate value as of a given evaluation date is the sum of the ultimate values across every accident/report year (i.e., the sum of the ultimate values across all rows). In a snapshot date triangle, the total ultimate value as of a given evaluation date is the ultimate value for the one row in the triangle where the snapshot date is equal to the evaluation date. Summing across rows in a snapshot date triangle is not appropriate because the lack of mutual exclusivity among rows would result in double counting.

One might wonder why snapshot date triangles are useful. On the surface it appears that there is a reduction of information in a snapshot date triangle because we are effectively collapsing the accident or report year dimension. The value of snapshot date triangles becomes apparent when we consider their application at a granular level.

2.2.4 Granular snapshot date triangles

Table 2.1 is a snapshot date triangle of FIFO redemptions for a hypothetical loyalty program. It is aggregated across all members in the program. A snapshot date triangle can also be produced for each member individually.

For a given member, a snapshot date triangle would have one row for each snapshot date since the member joined the program. Some of these snapshot dates may have zero outstanding points, meaning that the member had never earned points or had either redeemed and/or expired all his points as of that snapshot date. For these rows, the percent redeemed in each cell would be null since there would be no points that could possibly be redeemed at each observation age.

Below is an illustrative example of what this might look like for a member that joins the program on January 15, 2016. The current evaluation date is January 31, 2017 and this member's historical transactions are as follows:

Table 2.2 Example Member Transactions

Transaction Date	Transaction Type	Points
1/15/2016	Earn	1000
3/20/2016	Redeem	500
4/6/2016	Redeem	500
9/29/2016	Earn	2000

This member's corresponding snapshot date triangle would look like this:

Table 2.3: Example Member Snapshot Date Triangle

		Observation Age											
Snapshot Date	Outstanding Points	1	2	3	4	5	6	7	8	9	10	11	12
1/31/2016	1,000	0.0%	50.0%	100.0%	100.0%	100.0%	100.0%	100.0%	100.0%	100.0%	100.0%	100.0%	100.0%
2/29/2016	1,000	50.0%	100.0%	100.0%	100.0%	100.0%	100.0%	100.0%	100.0%	100.0%	100.0%	100.0%	
3/31/2016	500	100.0%	100.0%	100.0%	100.0%	100.0%	100.0%	100.0%	100.0%	100.0%	100.0%		
4/30/2016	0	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL			
5/31/2016	0	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL				
6/30/2016	0	NULL	NULL	NULL	NULL	NULL	NULL	NULL					
7/31/2016	0	NULL	NULL	NULL	NULL	NULL	NULL						
8/31/2016	0	NULL	NULL	NULL	NULL	NULL							
9/30/2016	2,000	0.0%	0.0%	0.0%	0.0%								
10/31/2016	2,000	0.0%	0.0%	0.0%									
11/30/2016	2,000	0.0%	0.0%										
12/31/2016	2,000	0.0%											
1/31/2017	2,000												

The number of cells in this triangle can be expressed using the following formula:

$$[\text{Number of Cells in Triangle}] = N \text{ x } (N-1)/2 = (N^2 - N)/2 \qquad (2.2)$$

Where N = number of snapshot dates in the triangle

Formula 2.2 implies that the number of cells in a snapshot date triangle follows quadratic growth with the number of snapshot dates.

A snapshot date triangle can be represented in table format as a dataset with primary keys [Member ID], [Snapshot Date], and [Observation Age]. Each cell in a snapshot date triangle thus represents a row in this dataset. Consequently, the dataset of member-level snapshot date triangles can become quite large as the number of snapshot dates increases. For programs with millions of members that have been around for decades, the member-level snapshot date triangle dataset could potentially have billions of rows. The question of how to work with this volume of data is a technical consideration when performing this type of analysis but is out of scope for this paper.

The dataset of member-level snapshot date triangles would have a variable for [FIFO Cumulative Percent Redeemed], which is evaluated as of the [Observation Age] for a given [Snapshot Date]. Characteristics of each member as of a [Snapshot Date] can also be added to this dataset. For example, we can derive a member's [Outstanding Points], [Cumulative Earned Points], [Cumulative Redeemed Points], [Cumulative Expired Points], and [Time since Last Activity] for each member as of each [Snapshot Date]. Note that all of these are dynamic covariates since their value can change over time. We will assume that these variables are added to the dataset. We refer to these variables collectively as [Member Characteristics]. We will refer to this dataset as the Snapshot Triangle Dataset.

[Member Characteristics] beyond those listed above can be added to the dataset so long as they can be derived as of the [Snapshot Date]. This requirement is important because the [Member Characteristics] will be used as predictors for redemption behavior. Since [Member Characteristics] are dynamic covariates, if any factor is evaluated as of a date after the [Snapshot Date], then it will include future information that was not available as of the given [Snapshot Date] and may overstate the predictiveness of the model for that given [Snapshot Date] and make it less useful for projections for a new [Snapshot Date].

The analogy of the Snapshot Triangle Dataset in the insurance context would be a snapshot date triangle for each open claim as opposed to each member. That is, for a given claim, the Snapshot Triangle Dataset would have a row for every combination of [Snapshot Date] and [Observation Age] since that claim was reported. The key would be [Claim Number], [Snapshot Date], and [Observation Age]. The metric being tracked as of each [Observation Age] could be cumulative payments on that claim subsequent to the [Snapshot Date]. The [Member Characteristics] would be replaced with [Claim Characteristics], which could include variables such as [Accident Month], [Report Month], [Claim Type] and various other available claim details. Again, it is important that all [Claim Characteristics] be derived as of the [Snapshot Date].

In section 2.2.3 we indicated that the aggregate snapshot date triangle appears to result in a loss of information relative to an accident/report year triangle. This is not true for claim-level snapshot date triangles. In fact, there is an increase in information since [Accident Month] and [Report Month] are readily available as well as many other details about the claim found in [Claim Characteristics].

An actuary trying to model payment patterns for individual claims may be tempted to structure the modeling dataset into a claim-level report month triangle given the profession's familiarity with organizing data in this way. That is, a triangle that is tracking payments on claims since the last day of the month in which they were reported. The dataset representation of this format would have key [Claim Number], [Development Age].

Organizing the data in this way would require you to restrict yourself to [Claim Characteristics] evaluated as of the [Report Month] (i.e., [Development Age] = 0). Using [Claim Characteristics] evaluated at any date subsequent to the [Report Month] would be using future information not available as of the [Report Month] and would therefore overstate the predictiveness of your model. Therefore, structuring your data into claim level report month triangles would be useful for predicting payment patterns for claims when they are first reported but would less useful for predicting future payments on reported claims after they have been reported and observed for a few development ages.

For example, if a claim is 18 months old, a model developed on a claim-level report month triangle would not allow you to use information gathered over those 18 months to predict payments from [Development Age] 19 to ultimate. Hence, claim-level report month triangles result in a loss of information when predicting future payments on claims that have been reported and observed for a few development ages.

Claim-level report month triangles represent the subset of the rows in a Snapshot Triangle Dataset where [Snapshot Date] = [Report Date]. Additional observations are added to the Snapshot Triangle Dataset for every subsequent [Snapshot Date] and [Observation Age] that we have actually observed in the historical data.

For example, a claim reported on January 31, 2016 would have the following records in the Snapshot Triangle Dataset assuming the current evaluation date is April 30, 2016.

Table 2.4: Example Rows in Snapshot Triangle Dataset

Row Number	Claim	Report Date	Snapshot Date	Observation Date	Development Age	Observation Age	[Member Characteristics]	Cumulative Payments Since Snapshot Date
(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)
1	A	1/31/2016	1/31/2016	2/29/2016	1	1	XXX	100
2	A	1/31/2016	1/31/2016	3/31/2016	2	2	XXX	500
3	A	1/31/2016	1/31/2016	4/30/2016	3	3	XXX	800
4	A	1/31/2016	2/29/2016	3/31/2016	2	1	YYY	400
5	A	1/31/2016	2/29/2016	4/30/2016	3	2	YYY	700
6	A	1/31/2016	3/31/2016	4/30/2016	3	1	ZZZ	300

Notes:
(6) Development Age = Observation Date - Report Date
(7) Observation Age = Observation Date - Snapshot Date

The purple shaded rows represent the subset of observations that would be included in a claim-level report month triangle. Rows 4 and 5 allow us to take a snapshot of the claim at development age 2 (i.e., snapshot date 2/29/2016) and see all the subsequent payments that occur on that claim (i.e., payments as of observation dates 3/31/2016 and 4/30/2016). Similarly, row 6 allows us to take a snapshot of the claim at development age 3 (i.e., snapshot date 3/31/2016) and see all subsequent payments that occurred on that claim (i.e., payments as of observation date 4/30/2016). Since the [Claim Characteristics] are evaluated as of the [Snapshot Date] we gain the ability to use these variables to predict [Cumulative Payments Since Snapshot Date] for claims that are reported and observed for a few development ages.

Hence, claim level snapshot date triangles allow for more information to be used than claim level report month triangles because the predictors can incorporate all information through the snapshot date rather than the report date.

2.3 Modeling Strategy

Once the snapshot date triangles have been built, the modeling strategy has three main steps. First, we select various [Observation Ages] and build models to predict [FIFO Cumulative Percent Redeemed] at those ages using [Member Characteristics]. Second, we cluster members into groups based on the predicted values from those models. Third, we summarize results by cluster into snapshot date triangles and apply triangular methods to estimate ultimate values.

2.3.1 Build models to predict FIFO cumulative percent redeemed

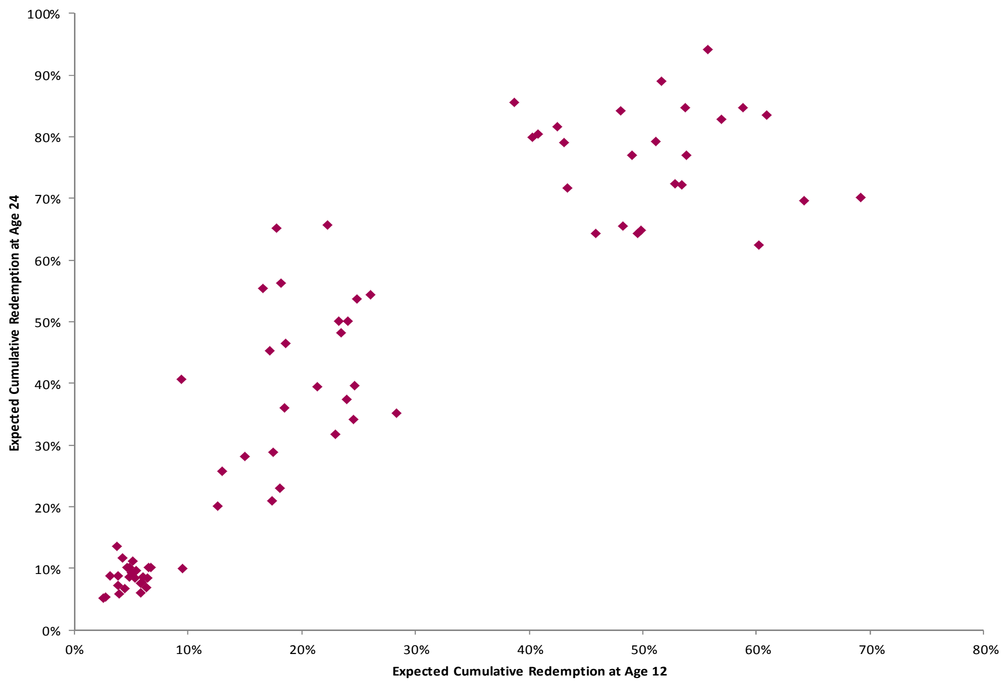
Using the Snapshot Triangle Dataset, the first step of the modeling strategy is to build models to predict [FIFO Cumulative Percent Redeemed] at various [Observation Ages] using [Member Characteristics] as predictors. Any modeling technique, such as decision trees or regression, can be used so long as it performs well.

For the sake of simplicity, we will use an example where we build two models:

- Model 1: Target is [FIFO Cumulative Percent Redeemed] at [Observation Age] = 12. Predictors are [Member Characteristics]. We denote P(12) as the predicted cumulative percent redeemed at age 12.
- Model 2: Target is [FIFO Cumulative Percent Redeemed] at [Observation Age] = 24. Predictors are [Member Characteristics]. We denote P(24) as the predicted cumulative percent redeemed at age 24.

After the models are built, they are applied to every member at every snapshot date to produce a predicted value P(12) and P(24). A scatter plot of the expected values may look something like this:

Graph 2.1: Illustrative example of scatter plot of P(12) and P(24)

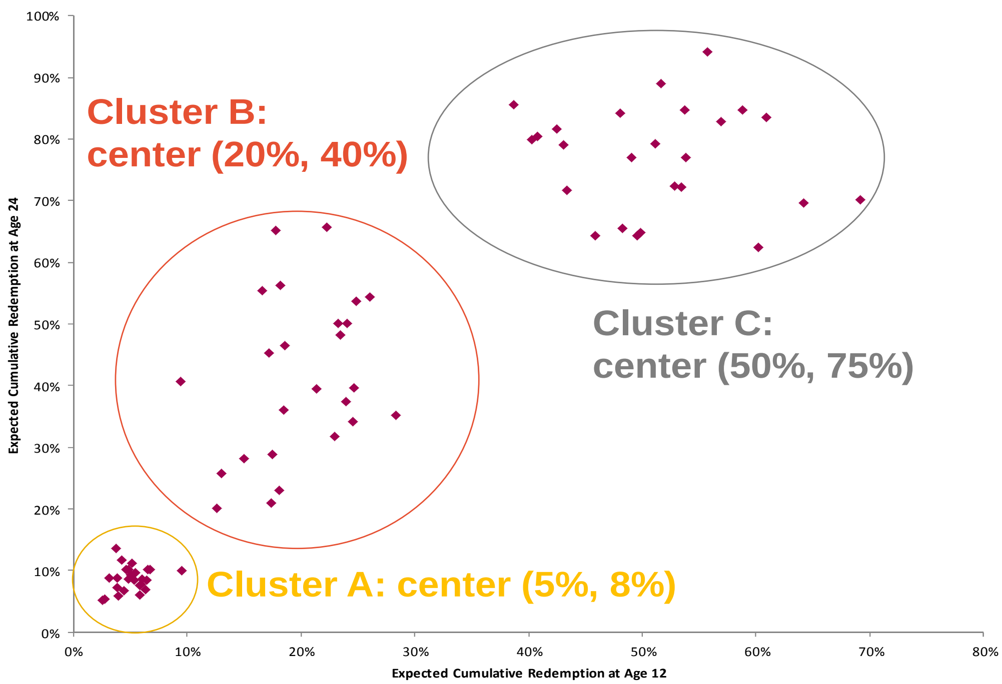


2.3.2 Cluster based on predicted values

A clustering algorithm is applied to cluster across the P(12) and P(24) dimensions. Clustering is performed on the expected values because we are trying to identify clusters with similar expected redemption behavior, which should always be greater than zero. Using actual values instead of expected will result in a large cluster centered at the origin for all members that have not redeemed any points, incorrectly suggesting that these members would have an expected redemption pattern of zero at 12 and 24.

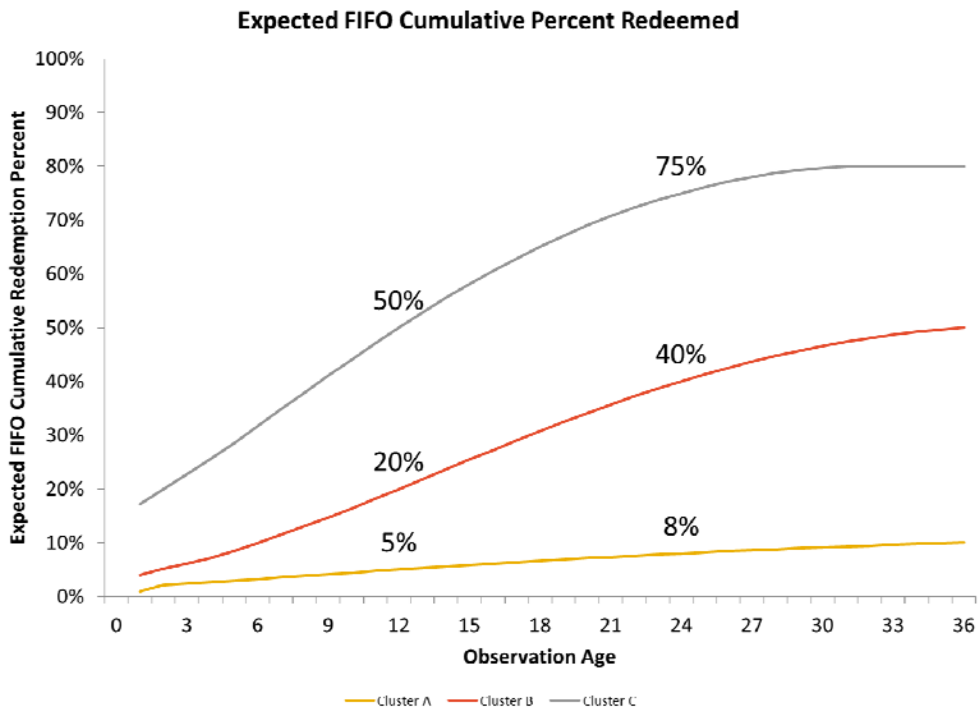
The resulting cluster centers when clustering on P(12) and P(24) may look something like this:

Graph 2.2: Illustrative Example of Clusters Produced on P(12) and P(24)



The clustering algorithm groups every member at every snapshot date into clusters with similar expected cumulative redemption patterns at observation ages 12 and 24. For example, the expected cumulative redemption patterns for each cluster may look like this:

Graph 2.3: Illustrative Example of Expected Cumulative Redemption Patterns



If model 1 and model 2 are performing well, then we would anticipate that the actual [FIFO Cumulative Percent Redeemed] for each cluster would be very close to the expected lines above at age 12 and 24.

Note that, for simplicity, we only model two observation ages (12 and 24) in the above example. All other points on the lines in graph 2.3 are based on an arbitrary interpolation/extrapolation and shown simply to illustrate what the cumulative pattern could look like at other ages. In practice, it is ideal to model as many observation ages as possible, which would provide a more complete definition of the expected cumulative redemption patterns.

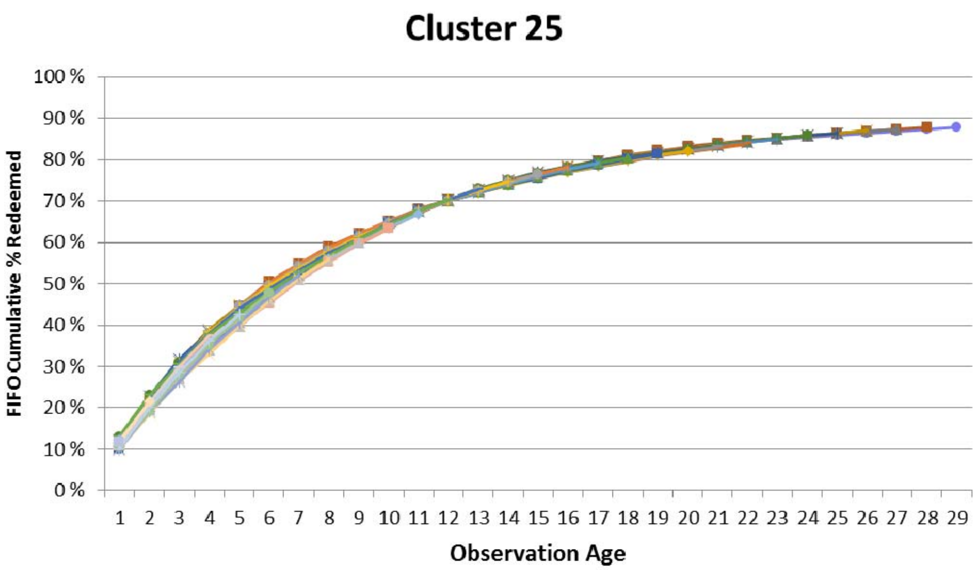
2.3.3 Summarize results by cluster

Using the models derived above, we assign each [Member ID] at each [Snapshot Date] in the Snapshot Triangle Dataset to a cluster. We then produce an aggregate snapshot date triangle of actual [FIFO Cumulative Percent Redeemed] for each cluster (as opposed to expected [FIFO Cumulative Percent Redeemed]).

Following is a graphical example of the resulting snapshot date triangle of actual [FIFO

Cumulative Percent Redeemed] for a single cluster when this methodology is thoroughly applied (i.e., as opposed to our simplified two dimensional example).

Graph 2.4: Graph of Actual FIFO Cumulative % Redeemed for Cluster 25



Each line represents a row in the snapshot date triangle for cluster 25. That is, each line represents a snapshot date, and we are tracking the actual FIFO cumulative percent redeemed on points outstanding for members who are assigned to cluster 25 as of that snapshot date. Each of the lines then varies in length, depending on the number of observation ages we have accumulated as of the current snapshot date.

The lines are very consistent, with each line sitting almost perfectly on top of the other lines. The only thing that each line has in common is that the members included in each line belonged to Cluster 25 as of the snapshot date (i.e., as of observation age zero). This graph illustrates that every time we saw a member in cluster 25 their subsequent average redemption behavior was generally the same. The consistency of this average behavior gives us confidence to predict, on average, how a member in cluster 25 today is going to redeem his or her currently outstanding points in the future.

It is not surprising that we see this level of consistency since the clusters were designed such that all members in the cluster have the same average cumulative redemption patterns. If the models were well calibrated and performed well on hold-out data, then we would anticipate that the actual [FIFO Cumulative Percent Redeemed] by cluster would produce consistent results.

At this point, snapshot date triangles aggregated by each cluster can be produced, and triangular methods (e.g., chain ladder) can be used to derive an expected redemption pattern and extrapolate a tail beyond the observed data. The point at which this expected pattern flattens is the URR (i.e., the projected ultimate cumulative redemptions, including the tail).

Note that this methodology does not require us to predict any [Member Characteristics] (i.e., dynamic covariates) into the future in order to predict the expected future redemptions. Organizing the data into the Snapshot Triangle Dataset allows us to use all available information from [Member Characteristics] as of the snapshot date to predict all future redemptions without having to explicitly state how those [Member Characteristics] will change in the future. This can potentially make the modeling exercise much easier than the method proposed by Wüthrich [2].

3. RESULTS AND DISCUSSION

Organizing member-level data into snapshot date triangles is beneficial because it allows us to use all information available as of a snapshot date to make predictions about future behavior. However, this method for organizing data comes at a cost, since the size of the dataset follows quadratic growth with each additional snapshot date.

The modeling strategy described in this paper uses individual member characteristics to sort members at each [Snapshot Date] into groups with consistent redemption patterns. This consistency gives us confidence in our predictions of future redemption behavior for members in a given cluster today.

A key benefit of the approach is that it does not require the ability to predict how the [Member Characteristics] (i.e., dynamic covariates) will change in the future in order to predict future redemptions. This simplifies the modeling exercise relative to that proposed by Wüthrich [2].

4. CONCLUSIONS

We hope to build on the work done by Wüthrich [2] to illustrate a flexible model for granular reserving that does not require the ability to predict dynamic covariates. The modeling framework outlined in this paper illustrates how machine learning techniques applied to granular snapshot date triangles achieves this goal.

There is a strong analogy between loyalty program redemption patterns and insurance claim payment patterns, so we believe there is an opportunity to apply this framework to model payment patterns on individual claims. Reserving at the individual claim level has been a topic in the actuarial profession for many years, but the practical application of the proposed methods has not become commonplace. Our hope is that the flexibility of this framework could help make individual claims reserving more accessible.

Acknowledgment

The authors acknowledge Ann Conway and Sean McDermott for the gracious support and encouragement as we developed this framework. Their leadership and mentorship has been invaluable. Lastly, a special thank you to Emilie Dubois for her thorough review and valuable insights as we finalized this paper.

5. REFERENCES

[1] Gault, T., Llaguno, L., and Menard, M., "Loyalty Rewards and Gift Card Programs: Basic Actuarial Estimation Techniques," *CAS E-Forums*, Summer 2012

[2] Wüthrich, M., "Machine Learning in Individual Claims Reserving," *Swiss Finance Institute Research Paper Series*, No. 16-67

[3] Taylor, G., McGuire, G., and Sullivan, J. "Individual Claim Loss Reserving Conditioned by Case Estimates," *Annals of Actuarial Science*, **2008**, *Vol. 3, Issue 1-2*, 215-256.

Abbreviations and notations

URR, Ultimate Redemption Rate
CPP, Cost Per Point
FIFO, First In First Out
IBNR, Incurred But Not Reported
IBNER, Incurred But Not Enough Reported

Biographies of the Authors

Len Llaguno is a co-leader of Willis Towers Watson's Global Loyalty Analytics practice. His area of expertise is in the application of advanced predictive analytics to enable leaders of loyalty programs to predict behaviors of their membership, allowing them to find opportunities to drive better financial performance for their loyalty programs. He has an Honors Bachelor of Science degree, magna cum laude, in computer science and actuarial science from the University of Toronto. He is a Fellow of the CAS and a Member of the American Academy of Actuaries.

Manolis Bardis is a co-leader of Willis Towers Watson's Global Loyalty Analytics practice. He has assisted loyalty programs in various industries including hospitality, aviation, coalition and financial institutions, managing their financial risk and providing advice on planned program changes. Mr. Bardis has a Ph.D. from the University of Notre Dame and an M.B.A. from the University of New Hampshire. He is a Fellow of the CAS and a Member of the American Academy of Actuaries.

Robert Chin is a consultant with the Willis Towers Watson's Global Loyalty Analytics practice. His area of expertise is in the application, development and delivery of advanced predictive analytics of loyalty programs. He holds a Bachelor of Science degree, summa cum laude, in actuarial science from St. John's University in New York.

Tina Gwilliam is a co-leader of Willis Towers Watson's Global Loyalty Analytics practice. For over a decade she has helped loyalty programs across a spectrum of industries manage their financials risks. Ms. Gwilliam graduated cum laude from Duke University with a Bachelor of Science in Mathematics. She is a Fellow of the CAS and a Member of the American Academy of Actuaries.

Julie Hagerstrand is a consultant with the Willis Towers Watson's Global Loyalty Analytics practice. She holds a Bachelor of Arts degree, summa cum laude, in mathematics and economics from Washington University in St. Louis. She is a Fellow of the CAS and a member of the American Academy of Actuaries.

Evan Petzoldt is a consultant at Willis Towers Watson. In addition to his contributions to the Global Loyalty Analytics practice, he has expertise in pricing and reserving in the property and casualty insurance market. He holds a Bachelor of Science degree, summa cum laude, in actuarial science from Maryville. He is an Associate of the CAS and a member of the American Academy of Actuaries.