ABSTRACT

Clustering is used by actuaries in a data compression process to make massive or nested stochastic simulations practical to run. A large data set of assets or liabilities is partitioned into a user-defined number of clusters, each of which is compressed to a single representative policy. The representative policies can then simulate the behavior of the entire portfolio over a large range of stochastic scenarios. Such processes are becoming increasingly important in understanding product behavior and assessing reserving requirements in a big-data environment. This article proposes a variety of clustering techniques that can be used for this purpose. Initialization methods for performing clustering compression are also compared, including principal components, factor analysis and segmentation. A variety of methods for choosing a cluster's representative policy is considered. A real data set comprised of variable annuity policies, provided by Milliman, is used to test the proposed methods. It is found that the compressed data sets produced by the new methods, namely model-based clustering, Ward's minimum variance hierarchical clustering and k-medoids clustering, can replicate the behavior of the uncompressed (seriatim) data more accurately than those obtained by the existing Milliman method. This is verified within sample, by examining location variable totals of the representative policies versus the uncompressed data at the five levels of compression of interest. More crucially it is also verified out of sample by comparing the distributions of the present values of several variables after 20 years across 1,000 simulated scenarios based on the compressed and seriatim data, using Kolmogorov-Smirnov goodness-of-fit tests and weighted sums of squared differences.

JEL Classification code: C55 Large Data Sets: Modelling and Analysis.

Keywords: Clustering weighted data; data compression; hierarchical clustering; model-based clustering; stochastic forecasting.

1. INTRODUCTION

1.1 The Need for Data Compression

The use of stochastic scenarios is becoming increasingly popular in actuarial modelling versus deterministic approaches. The current trend is towards the use of nested stochastic scenarios (Reynolds and Man, 2008). Such simulations are useful to insurers who wish to see a robust probabilistic distribution of possible present values across a range of future scenarios. However, it is not computationally practical to run nested stochastic simulations for large data sets, particularly where products have moving parts or heavy optionality. While insurers generally have sufficient computing power to perform seriatim (full data) calculations for single scenario forecasts, or even for a moderate number of scenarios, the use of nested stochastics dramatically increases run time.

Milliman have developed a data compression method using cluster analysis (Freedman and Reynolds, 2008) that makes nested stochastic modelling and massive stochastic runs practical. Millions of assets or liabilities can be well represented by a user-specified number of representative policies, typically a few hundred or a few thousand. The process can produce a good approximation to the results of a seriatim model across a range of economic or experience scenarios. It can be used for any asset class or product type and clustering solutions can be maintained and applied in a consistent manner at subsequent valuation dates.

1.2 Clustering

Clustering means identifying groups of similar objects in a data set, such that objects within clusters are more similar to each other than to objects in different clusters (Anderberg, 1973). In this data compression application each group or cluster is ultimately represented by a single object from the cluster, which is a member of the original data set, scaled up by the total size of all the objects in the cluster.

Similarity between objects, or rather dissimilarity, is measured by Euclidean distance in high-dimensional space according to p appropriately scaled "location variables", which can be any variables that it is desirable for the compressed data set to be able to closely reproduce. Typically when clustering, all observations in a data set are treated equally. However in this application the data are weighted: each object also has a "size variable", typically account value or face amount, meaning that larger objects will have more influence on the cluster locations than smaller ones. Consider observations x_i, where $i = 1,\ldots,n$, each comprising p location variables, and an aim of partitioning the data into G clusters. In this application, all location variables are quantitative and continuous so the dissimilarity between x_i and x_j is given by:

$$d(x_i, x_j) = \{(x_{i1}-x_{j1})^2 + (x_{i2}-x_{j2})^2 + \ldots + (x_{ip}-x_{jp})^2\}^{0.5} \quad \text{I}$$

However, it is possible to extend model-based clustering methods to mixed, ordinal or categorical data (McParland and Gormley, 2014).

1.3 Data Compression by Clustering

A data set of N objects can be partitioned into G clusters, where $G < N$, by any clustering method. Given the cluster membership of each individual object, Milliman's data compression technique proceeds by reducing each cluster to a single representative policy. The size of the representative policy for cluster k, S_k, is the sum of sizes of the objects within the cluster, $S_k = \sum_{i=1}^{N_k} w_i$, where w_i is the size of object i and N_k is the number of objects in cluster k.

The location vector of the representative policy may be determined by several methods. It is required in this application that it be equal to the location of an actual object (original policy) in the cluster for subsequent modeling purposes. The centroid of cluster k is defined as its size-weighted mean location vector:

$$\bar{x}_k = [(\sum_{i=1}^{N_k} w_i x_{i1}, \sum_{i=1}^{N_k} w_i x_{i2}, \ldots, \sum_{i=1}^{N_k} w_i x_{ip}) / \sum_{i=1}^{N_k} w_i] \quad \text{II}$$

The optimal clustering solution is the one that partitions the data into clusters that can be best represented by single objects. Several means of selecting a cluster's representative policy are considered.

1.3.1 Nearest to Centroid Selection

The location vector of the representative policy for cluster k, x_k^*, is set equal to x_i where i is the object in cluster k that minimizes $d(x_i\ \bar{x}_k)$. This constitutes Milliman's default approach and perhaps the most intuitive means by which to represent a cluster by a single policy. For these reasons, and due to the widespread availability of comparative results for the Milliman clustering method under this approach, this selection rule is predominantly considered in the results presented in Section 4. However, this approach potentially underestimates variability by ignoring within-cluster variance. To demonstrate that the quality of compression and the approximation to the true underlying distribution is not critically impacted by this representative policy selection method, additional methods are also tested for the crucial Net Revenue and CTE70 variables and results presented in Sections 4.1.3.5 and 4.1.4 respectively.

1.3.2 Random Selection

The location vector of the representative policy for each cluster is set equal to that of an object selected completely at random from the cluster.

1.3.3 Random Selection Weighted by Size

The location vector of the representative policy for each cluster is set equal to that of an object selected at random from the cluster, with the probability of an object being selected proportional to its size, w_i.

1.3.4 Random Selection Weighted by Distance to Centroid

The location vector of the representative policy for each cluster is equal to that of an object selected at random from the cluster, with the probability of an object being

selected inversely proportional to its distance from the centroid of its cluster, $d(x_i, \bar{x}_k)$. This serves as a proxy for using a policy's contribution to the model likelihood (see Section 3.4) as a weight for its probability of selection.

1.3.5 Modified centroid selection

If the clusters are arranged $(1,2 ..,k,...G)$ in ascending order of total size, $S_1 \leq S_2 \leq ... \leq S_k \leq ... \leq S_G$, then the location vector of the representative policy from cluster k, x_k^*, is equal to x_i where i is the object in cluster k that minimizes $d(x_i , \bar{x}_k - A_{k-1})$, where $A_0 = 0$ and $A_k = x_k^* - (\bar{x}_k - A_{k-1})$. This method reduces the prevalence of trends whereby the representative objects selected are consistently above or below the theoretical cluster centroids for certain location variables.

Figure 1 shows an illustrative data set consisting of 20 objects of varying size, which have just two location variables, generically titled *x* and *y*. These could be, for example, Opening Reserve and Premium for liabilities; or Book\Par Ratio and Yield to Maturity for assets. The 20 observations are ultimately compressed into four representative policies. This method of data compression was developed by Freedman and Reynolds (2008) using a non-parametric, hierarchical agglomerative algorithm to cluster the data. This paper is primarily concerned with the formation of Figure 1(b), the method by which the data are partitioned into clusters. The benefits of alternative non-parametric and model-based clustering methods are explored. The formation of Figure 1(c), the method by which representative policies are derived from clusters, was considered in Section 1.3.

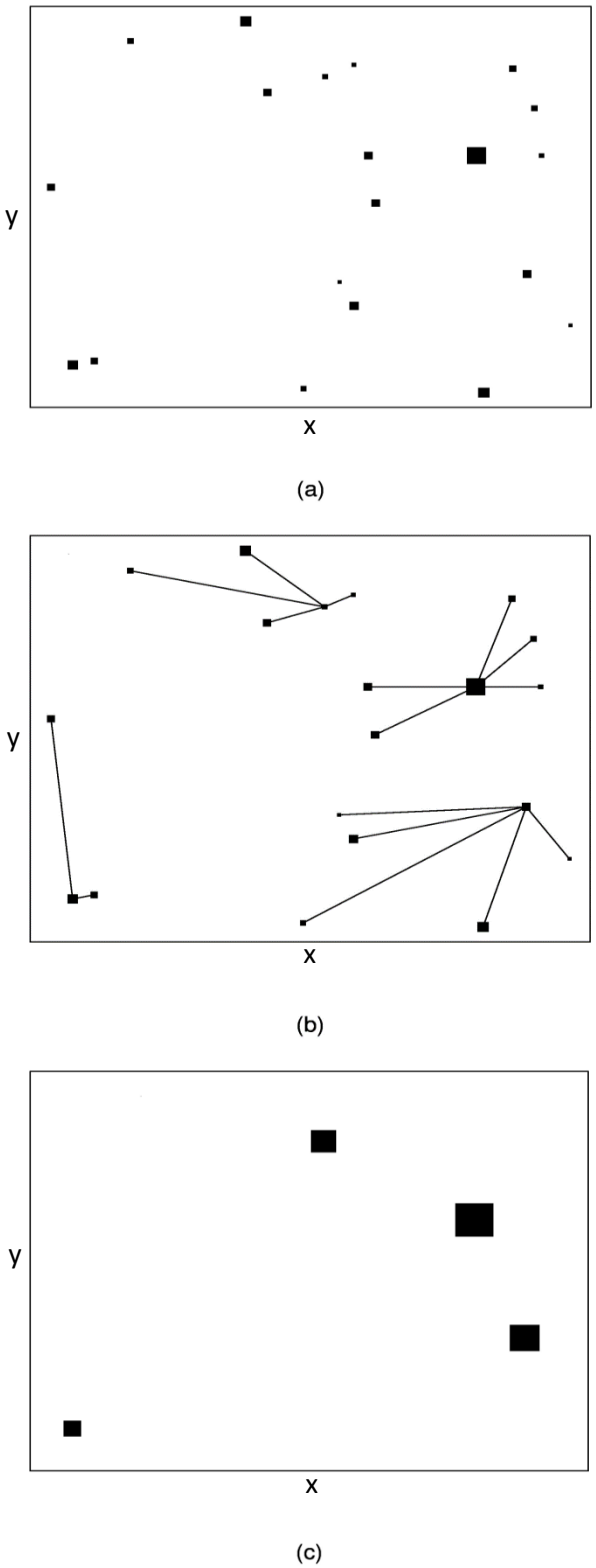


Figure 1: Illustrative data compression by clustering: (a) depicts 20 observations of varying size with two location variables, *x* and *y*; (b) shows the observations being partitioned into clusters by some method; (c) presents the representative policy for each cluster, scaled up to the size of all observations in the cluster.

2. DATA

Milliman have provided a data set of 110,000 variable annuity policies on which to test various clustering methods. New results can be compared to those obtained by the current method at five levels of compression: from the full set of 110,000 policies to 5000, 2500, 1000, 250 and 50 representative policies. The size variable used is Account Value In Force and policies are clustered according to a series of location variables. Optionally, certain location variables can have subjective weights placed on them. This is done when the quality of fit for these variables is felt to be of more importance than others. Variables with larger weights will be more influential when clustering. Milliman pursues this option and consistency with their original approach to using weightings is maintained. The location values and their accompanying weights are detailed below.

Initial Values (each weight 1)

GMDB (Guaranteed Minimum Death Benefit) Ratchet[1], Rollup[2] and Rop Face Value In Force[3] (3 variables)

GMIB (Guaranteed Minimum Income Benefit) Ratchet and Rollup Face Value In Force (2 variables)

GMAB (Guaranteed Minimum Accumulation Benefit) Rop Face Value In Force

Account Value In Force by Fund (7 variables)

General Account Value In Force

Present Values (for 5 calibration scenarios each)

Net Revenue (weight 4)

Commission (weight 2)

Revenue Sharing (weight 2)

Policy Maintenance Expenses (weight 2)

M&E (Mortality & Expense) Fee Income (weight 3)

Net GMDB Costs (weight 3)

Net GMIB Costs (weight 3)

Net GMAB Costs (weight 3)

This gives an overall total of 54 location variables.

The present values of the location variables are calculated for each of five calibration scenarios representing the 2.5%, 20%, 50%, 80% and 97.5% levels of the aggregate average "wealth ratios" across a set of 1000 stochastic scenarios. The aggregate average wealth ratio is calculated based on the value of $1 invested at the start of the projection (in various funds, based on the starting allocation), and left to accumulate for 20 years. It provides a useful summary measure of the overall fund position and is therefore preferable to using any individual location variable. The precise percentiles used (2.5%, 20%, 50%, 80% and 97.5%) are somewhat arbitrary and it is recognized that the path of the development of the indexes, and not just their final average values, will have a substantial impact on results. However, the process does ultimately select a wide range of calibration scenarios that provide a fairly even spread across good, bad and moderate outcomes at which the clustered portfolio can be calibrated. Ultimately these are used to efficiently calculate the distribution of present values across the full underlying range of 1000 scenarios (or larger if so desired).[4]

[1] Ratchet – one means by which benefit bases for variable annuity policyholders can grow. "Ratchet" generally means that a policyholder's benefit base will reset to the maximum of the current value or a set of previous values (as money grows in equity/bond funds). The frequency of these "resets" is specified in policyholder contracts.

[2] Rollup – another means by which benefit bases for variable annuity policyholders can grow. "Rollup" generally implies that a policyholder's benefit base will grow at a specified rate of interest until a specified time or policyholder action, again specified in the policyholder contract.

[3] ROP – stands for "return of premium", a standard guarantee in variable annuity contracts where the policyholder is generally guaranteed a benefit base equivalent to the initial premium he/she paid.

[4] 1000 economic scenarios was the maximum number available for the purposes of the analysis in this paper. However the methods detailed have also been tested and shown to work well across 4000 scenarios in a related piece of research using clustering for mixed actuarial data in conjunction with Aegon See http://mathsci.ucd.ie//docserve?id=146 and use PIN = 6317 for full details.

1000 stochastic modelling scenarios has been deemed industry standard for a long time for applications of this nature. It is possible that the distribution of results from the selected 1000 scenarios is on average (or in the tail) "too high" or "too low" such that, if a set of 10,000 were used, calculated reserves would rise/drop modestly. However it is very likely that the effect of moving from 1000 to 10,000 scenarios would be similar for both the seriatim model and any reasonable compressed model. Hence it is deemed a sufficiently large number by Milliman and many other practitioners for testing and developing cell compression methods.

3. METHODOLOGY

3.1 Scaling the Variables

The location variables used are based on dollar amounts of various initial and present values. Prior to clustering, they are scaled as follows:

1) The values are unitized - each policy is divided by its size so that values are expressed in per-dollar amounts.

2) The variables are standardized - the values for each variable are divided by the size-weighted standard deviation of that variable.

3) Finally, if weights are being used, the values for each variable are then multiplied by the appropriate weight.

3.2 Current Method: Milliman's Non-parametric Clustering

A variety of non-parametric clustering algorithms exist. The one developed and currently used by Freedman and Reynolds (2008), which is tailored to suit weighted data, proceeds as follows, using the size and location variables described:

1) The dissimilarity d_{ij} (Euclidean distance) between every pair of policies is calculated (Equation I).

2) The "importance" of each policy, defined as its size w_i multiplied by the dissimilarity with its nearest neighbour, is calculated. Following the notation set out in Equations I and II, the importance of policy i, I_i, can be expressed as:

$$j' = argmin_j(d_{ij}) \qquad I_i = w_i\, d_{ij}' \qquad \text{III}$$

3) The least important policy, i^*, is identified and mapped away to its nearest neighbour, i'. The nearest neighbour policy retains its original location while its size is updated to the sum of its original size and the size of i^*:

$$i^* = argmin_i(I_i) \qquad x_i' = x_i' \qquad w_i' = w_i' + w_{i^*} \qquad \text{IV}$$

4) The importance values are recalculated for all observations and the overall process is repeated until the desired level of compression is reached.

Figure 2 shows how policies iteratively get mapped away to their nearest neighbours for the illustrative data. The numbers refer to the order in which the mappings occur. The less "important" policies (smaller and closer to their nearest neighbour) are mapped away first.

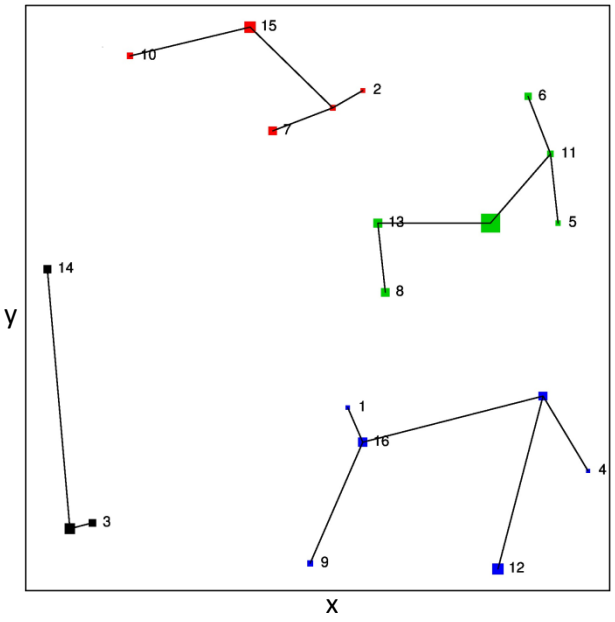


Figure 2: Milliman's clustering method applied to the illustrative data from Figure 1.

3.3 Alternative Non-parametric Clustering Methods

3.3.1 Ward's Minimum Variance Hierarchical Clustering

Milliman's method is a form of hierarchical clustering (Johnson, 1967) i.e. it begins by treating all observations as single clusters and then iteratively merges pairs of similar clusters. Specifically, Milliman's method uses the sizes of policies to define "importance" and at each step merges the "least important" cluster with its nearest neighbour.

In Ward's minimum variance method for hierarchical clustering (Ward, 1963) the pair of clusters to be merged at each step is the pair that will lead to the smallest increase in total within-cluster variance. For G clusters, each comprising of N_k objects with p variables, $k = 1,2,3\ldots G$, total within-cluster variance is given by:

$$\sum_{k=1}^{G}\{\sum_{i=1}^{N_k}[\sum_{j=1}^{p}\frac{1}{N_k}(x_{ij} - \bar{x}_{kj})^2]\} \qquad \text{V}$$

where $\bar{x}_{kj}$ is the mean value of the j^{th} variable in the k^{th} cluster. The **hclust.vector** function in the **fastcluster** R package (Müllner, 2013) implements this method efficiently for large data sets. Ward's minimum variance method produces compact, spherical clusters, the latter property meaning that it is equivalent to the EII model-based method for a mixture of Gaussian distributions, if weights are not attached to the observations (see Section 3.3.3).

3.3.2 K-medoids Clustering

K-medoids clustering, or partitioning-around-medoids (Van der Laan et al., 2003), is an algorithm for partitioning data into a fixed number of clusters, k. Given some initial partition, the medoid, or the actual observation from the data set closest to the centroid, of each cluster is identified and objects are reassigned to the cluster whose centroid is closest. This process is repeated until no more observations are moved. Clusters will be similarly sized, linearly separable and approximately spherical. It is preferable to the better-known k-means method in this context because it centres the clusters around actual objects from the data set rather than the locations of the theoretical means.

Typically when clustering, all objects in the data set are treated equally. Ackerman et al. (2012) discusses how a number of algorithms can be adapted to deal with weighted data. The standard software in R for k-medoids, namely the function 'pam' in the package **cluster (**Maechler, et al., 2015), does not deal with weighted data. New R functions have therefore been written to apply these clustering algorithms using the size-weighted mean location of the cluster in place of the pure centroid. The use of the **FNN** (Fast Nearest Neighbours) R package (Beygelzimer et al., 2013) in these functions ensures that they are efficient when dealing with large data sets.

Typically, when k is small, a moderate number of random starting values are used and the solution with the smallest within-cluster variance is selected. This is not practical when k is large as the number of possible starting values is too great. Instead, a hierarchical method such as Ward's or Milliman's should be used to obtain an initial partition.

3.3.3 Model-based Clustering

In model-based clustering (Fraley and Raftery, 2002) the data within each cluster are assumed to follow a multivariate normal distribution. Model-based clustering techniques have been widely used and have shown promising results in many applications involving complex data, including medical diagnosis (De la Cruz-Mesía,et al., 2008), gene expression microarray data (Mar and McLachlan, 2003), imaging (Neumann et al., 2008) and food science (Murphy et al., 2010). Their introduction into the actuarial sphere could potentially

yield significant modelling advances. Briefly, the data $(x_1, \ldots, x_n)$, are assumed to be generated by a mixture model with density

$$\prod_{i=1}^{n}\{\sum_{k=1}^{G}[\tau_k f_k(x_i|\Theta_k)]\} \qquad \text{VI}$$

where τ_k is the probability that observation or policy x_i belongs to cluster k and $f_k(x_i|\Theta_k)$ is a probability distribution with parameters Θ_k. In this application, as is most often the case, f_k is the density of a multivariate normal distribution so each cluster is parameterized by a mean vector μ_k and a covariance matrix Σ_k:

$$f_k(x_i|\mu_k, \Sigma_k) = |2\pi\,\Sigma_k|^{-0.5}\, exp\{-0.5(x_i-\mu_k)'\Sigma_k^{-1}(x_i-\mu_k)\} \quad \text{VII}$$

The covariance matrix Σ_k can be parameterized through eigenvalue decomposition by $\Sigma_k = \lambda_k D_k A_k D'_k$ such that λ_k controls the volume, D_k the orientation and A_k the shape of the k^{th} cluster (Banfield and Raftery, 1993). λ_k and A_k are not identified separately, hence λ_k is defined as the first eigenvalue of Σ_k.

Different levels of constraint can be placed on how the covariance structures are allowed to vary between clusters by keeping any or all of Λ, D and A fixed for all clusters. This leads to the standard model-based clustering notation for models, where cluster behavior is encapsulated using a three letter convention, the letters respectively denoting the volume, shape and orientation of the clusters. Figure 3 depicts some of the model types available with different covariance constraints, while Celeux and Govaert (1995) provide a detailed description of all 14 available model types.

In the most complex and flexible model, VVV, each cluster has a unique size, shape and orientation, while in the simplest model, EII, all the clusters are constrained to have equal volume and to be spherical in shape. K-means clustering and Ward's minimum variance method can be viewed as originating from the application of different estimation methods for the EII model-based clustering approach. [5] Figure 4 shows a VVV clustering solution fitted to the illustrative data set. In this illustrative example, the four clusters identified by the nonparametric method and the model-based method are the same. However, as the size and complexity of the data set increases this will not generally be the case.

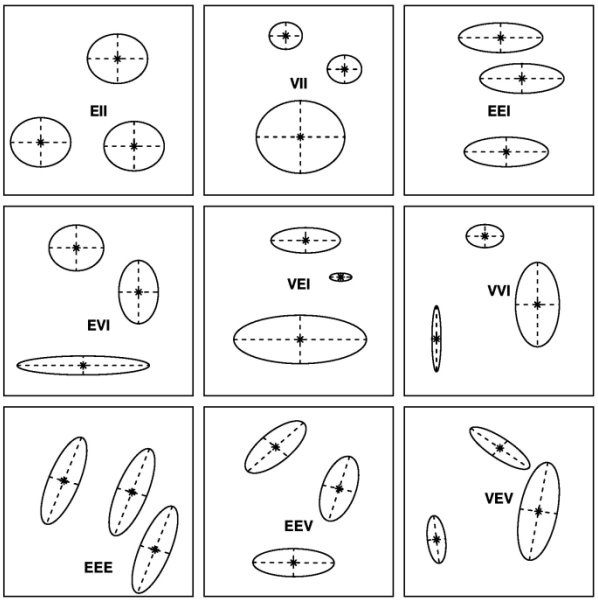


Figure 3: Model types: the letters respectively denote the volume, shape and orientation of clusters. Fixed

[5] Under the k-means approach, observations are assigned to the cluster whose mean is closest in squared Euclidean distance, cluster means are then recalculated, and the process repeated until no observations change cluster membership. Since the arithmetic mean of observations in a cluster is the least squares estimate of the true cluster mean, this approach is minimizing the total within cluster sum of squares at each step. In turn this is equivalent to applying Ward's minimum variance approach. If the k-means algorithm achieves the global minimum within-cluster variance and not just a locally optimal partition, the cluster membership (and corresponding parameter estimates) will be the same as if the EM algorithm is used to fit the EII model-based clustering approach and arrives at the maximum likelihood estimates of the cluster means (in fact Friedman (1989) refers to the EII model as the nearest-means classifier). However k-means requires an initial partition (usually random) of observations into the desired number of clusters and different initializations can produce different clustering solutions at convergence. This is not the case with Ward's method, which will always produce the same partition at convergence.

values across clusters (E), varying values across clusters (V) and set equal to the identity matrix (I) are possible.

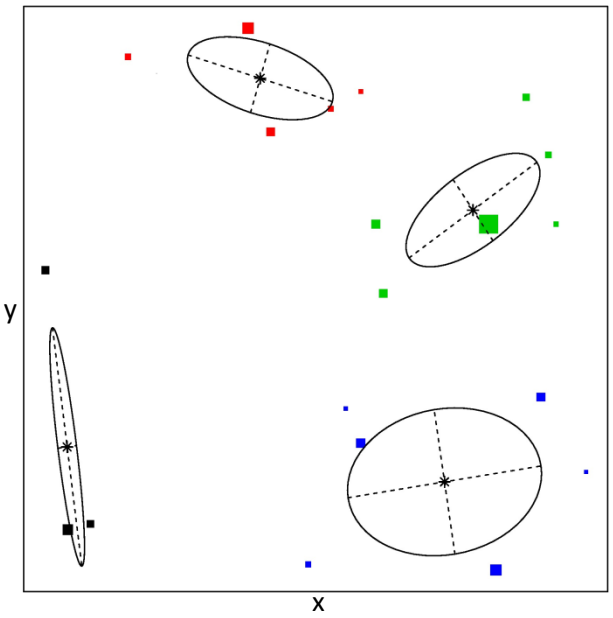


Figure 4: Model-based clustering applied to the illustrative data from Figure 1.

In most applications, the Bayesian Information Criterion (BIC) is used to identify the most suitable model type (Fraley and Raftery, 2002):

$$BIC = 2 \log p(X|\hat{\theta}_k, M_k) - \nu_k \log(n) \qquad \text{VIII}$$

where ν_k is the number of independent model parameters, Θ_k, to be estimated in model M_k, X is the data and n is the number of observations.

The BIC favours simpler models and penalizes better-fitting models for using too many parameters. In this application, the partitioning of the data is all that ultimately matters so there is no theoretical disadvantage to having too complex a model. Large data sets will, by definition, admit more complex models (Fayyad and Smyth, 1995). Therefore the best fitting model will, in theory, be the most flexible, VVV. However, it will be shown in the following sections that the computational cost of fitting complex models can be prohibitive when there are large numbers of clusters and variables and that some constraints can be beneficial when clustering for data compression.

Often, in clustering problems, there is interest in the number of clusters, *G*, present in the data set and in the distinctions between the groups. In this case, however, the level of compression is specified in advance and there is no concern as to whether, for example, a 999 or 1001-cluster solution has a better fit than 1000. Instead, the data are forcibly partitioned into a pre-specified number of groups.

3.4 The EM Algorithm and Weighted Data

The model parameters are estimated by the EM algorithm (Dempster et al., 1977; McLachlan and Krishnan, 1997), wherein each iteration consists of a Maximization (M) step and an Expectation (E) step. In the M-step, the parameters μ_k, Σ_k and τ_k are estimated by maximum likelihood from the data given the conditional probabilities, z_{ik}, that object i belongs to cluster k. In the E-step, the (N x G) Z matrix of conditional probabilities, given the parameters, is calculated. The two steps are repeated iteratively until convergence in the log-likelihood or the parameters is reached.

The likelihood function, L and the complete data likelihood function, L_c, for the finite mixture of normal distributions are specified as

$$L = \prod_{i=1}^{n} \sum_{k=1}^{G} (\tau_k f(x_i|\mu_k, \Sigma_k)) \qquad \text{IX}$$

$$L_c = \prod_{i=1}^{n} \prod_{k=1}^{G} (\tau_k f(x_i|\mu_k, \Sigma_k))^{z_{ik}} \qquad \text{X}$$

Taking natural logarithms, the log-likelihood function l and the complete log-likelihood function l_c are then:

$$l = \sum_{i=1}^{n} log \sum_{k=1}^{G} \tau_k f(x_i|\mu_k, \Sigma_k) \qquad \text{XI}$$

$$l_c = \sum_{i=1}^{n} \sum_{k=1}^{G} z_{ik} log(\tau_k f(x_i|\mu_k, \Sigma_k) \qquad \text{XII}$$

However, since our data are weighted, each policy has a size variable as well as location variables. This is to ensure that the cluster locations are influenced more by

larger policies than smaller ones. A method is described by Murphy and Scrucca (2012) whereby the contribution of each observation to the log-likelihood, and hence to model fit, has a weight. In this case the weighted log-likelihood l^w and complete log-likelihood functions l_c^w are defined as:

$$l^w = \sum_{i=1}^{n} w_i log \sum_{k=1}^{G} \tau_k f(x_i|\mu_k, \Sigma_k) \quad \text{XIII}$$

$$l_c^w = \sum_{i=1}^{n} \sum_{k=1}^{G} z_{ik} w_i log(\tau_k f(x_i|\mu_k, \Sigma_k) \quad \text{XIV}$$

where $\underline{w_i}$ is the size of policy i, scaled to ensure that $max(w_i) = 1$. The **me.weighted** function in the R package **Mclust** (Fraley et al., 2012) performs model-based clustering with the EM algorithm incorporating weights in this manner.

3.4.1 Initialization of the EM Algorithm

The EM algorithm has a linear rate of convergence, which can sometimes be very slow (Fraley et al., 2005) and can result in a solution that is only locally optimal (Lange and Zhou, 2010). Good starting values are required. Typically, these are obtained by hierarchical clustering. The **hclust.vector** function in the **fastcluster** package can again be used for this purpose. Alternatively Posse (2001) describes how minimum spanning trees can be used to obtain an initial partition for model-based hierarchical clustering in large datasets. However, neither of these methods accounts for the weighted nature of the data so the hierarchical clustering method of Freedman and Reynolds (2008), which takes the policy size into account, is preferred. This "Milliman method" corresponds to spherical clusters (the EII covariance structure in model-based clustering) in terms of cluster initialization. Despite this, it works well as an initialization method for fitting models that are not EII, as outlined in Sections 4.1, 4.1.2.1, 4.1.2.3, and 4.1.3. Sampling is often used to obtain an initial estimate of model parameters (Wehrens et al., 2004) but the large sample size required to initialize a set of parameters for thousands of clusters would defeat the purpose of that approach for this endeavour.

3.4.2 Dealing with Large Numbers of Variables

When clustering, policies are envisaged as having locations in high-dimensional space. In this case, with such a large number of location variables (54), the massive volume of space required can make it difficult to fit models to the data (Donoho, 2000).

Figure 5 shows the correlation present between the location variables. Note that there are groups of variables that are strongly correlated - this represents correlation between initial values and values across the five calibration scenarios of certain variables. This is a weighted correlation matrix, i.e. the contribution of each policy to the covariance matrix is weighted by its size.

There is a large number of variables and many of them are strongly correlated with each other, which may pose a challenge in applying model-based clustering. A dimension reduction step such as principal component analysis (Pearson, 1901; Jolliffe, 2002), factor analysis (Spearman, 1904; Harman, 1960) or variable clustering (Sanche and Lonergan, 2006) can be applied. These techniques allow us to re-express multivariate data that has a large amount, p, of observed variables in terms of a much smaller amount, q, of underlying or latent variables. As long as objects that are dissimilar according to the p variables are equivalently dissimilar according to the q variables then it is sufficient to cluster the data according to the q variables.

3.4.2.1 Principal Components Analysis

Principal component analysis (PCA) involves a spectral decomposition of the correlation matrix (with our size-weighted data, the weighted correlation matrix is used) into eigenvectors and eigenvalues. This produces a set of p orthogonal components, which are linear combinations of the original variables, presented in

decreasing order of percentage of variance in the data accounted for. The first q components are retained such that the marginal benefit of having $(q+1)$ components over q is sufficiently small. The PCA function in the **FactoMineR** R package (Lê, et al., 2008; Husson et al., 2014) can be used to perform PCA on weighted data.

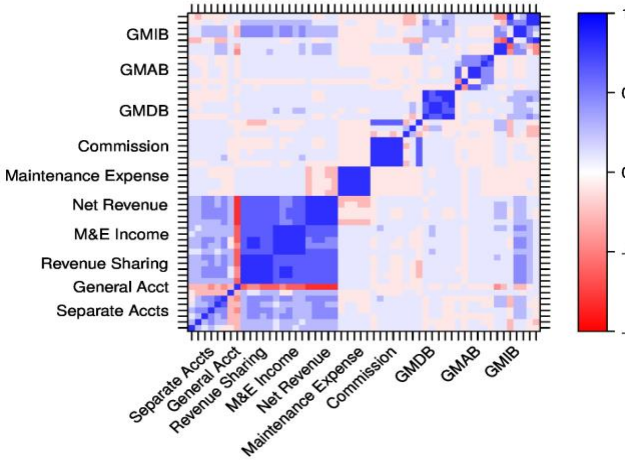


Figure 5: Weighted correlation of location variables.

Fayyad and Smyth (1995) warn that "frequently the goals of a dimension reduction step are not aligned with the overall goals of the analysis, e.g. principal components analysis is a descriptive technique but it does not necessarily help with classification or cluster identification." Additionally, Chang (1983) derives the result that data projection using principal components does not necessarily produce the optimal model-based clustering structure for data generated from a mixture of multivariate normal distributions.

Nevertheless, a moderate number of components, compared to the large number of original variables, can describe dissimilarities between objects well in practise (Ben-Hur and Guyon, 2003). For the motivating data considered, the strong correlation between the raw location variables enables dimension reduction without substantial loss of information. A dimension reduction step is desirable when implementing the model-based clustering methods as the quantity of highly correlated variables makes the parameter estimation computationally difficult. The relatively large number of clusters fitted for the purposes of actuarial data compression and the fact that the goal is the identification of representative policies, rather than the global clustering structure, also aids in insulating the process from substantial information loss. Furthermore, reducing the amount of variables makes it computationally easier to perform model-based clustering. If the use of PCA adversely affects the subsequent clustering compression in some applications, probabilistic PCA (Bellas et al., 2013) provides a potential remedy.

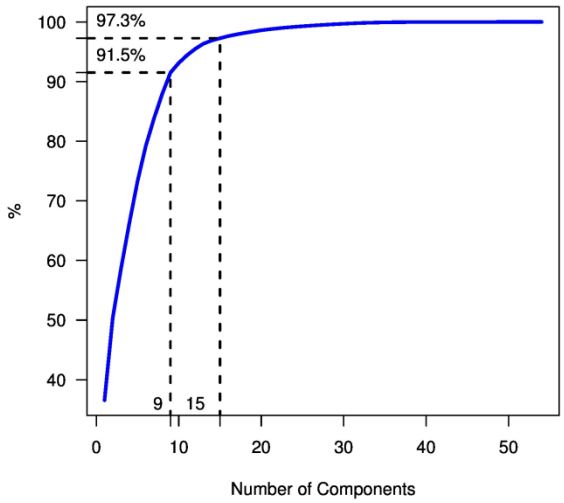


Figure 6: Proportion of variance explained by principal components.

Figure 6 shows the proportion of the total variance in the data that can be explained by various numbers of principal components. The choice of how many components to retain is subjective and is based on a trade-off between a loss of information (too few) and the computational cost of fitting more complex models (too many). While there is no clear kink in the curve, which would suggest an optimal cut-off point, it is

clear, for example, that using 9 principal components effectively means retaining 91.5% of the information in the data set; while using 15 retains 97.3%. Clustering is performed using both 9 and 15 principal components. While the principal components are uncorrelated with each other globally, there may be correlation within clusters.

3.4.2.2 Factor Analysis

Factor analysis is a more elaborate statistical method for describing data with p variables in terms of q^* underlying factors, where $q^*<p$, that makes more assumptions than PCA (Harman, 1960). The p location variables are each modelled as linear combinations of q^* factors, plus Gaussian error terms. The initial data are in the $(n \times p)$ location matrix $(X-\mu)$ where μ is the vector of the size-weighted means of the location variables.

The orthogonal factor analysis model assumes that each factor f follows a zero-mean, unit-variance, Gaussian distribution. The $(q^* \times p)$ factor loadings matrix Λ is calculated such that:

$$X-\mu = F\Omega + \epsilon \qquad \text{XV}$$

$$Cov(X-\mu, F) = \Omega \qquad \text{XVI}$$

$$Cov(X-\mu) = Cov(F\Omega + \epsilon) = \Lambda'\Lambda + \psi \qquad \text{XVII}$$

where ϵ is a p-dimensional zero-mean Gaussian noise vector with diagonal covariance matrix ψ.

The value ω_{jk} in the matrix Ω is the loading of the k^{th} observed variable on the j^{th} unobserved factor. There is no unique solution for Ω. The varimax factor rotation (Kaiser, 1958) is used to ensure that the solution is easily interpretable in the sense that each variable can be represented by one or two factors. To perform cluster analysis, X is replaced by the $(n \times q^*)$ matrix of factor scores F. Figure 7 depicts Ω, with 15 factors, in terms of their correlation with the original 54 location variables. This shows, for example, that the second factor represents commission across the five calibration scenarios and, similarly, the third factor represents the maintenance expenses variables.

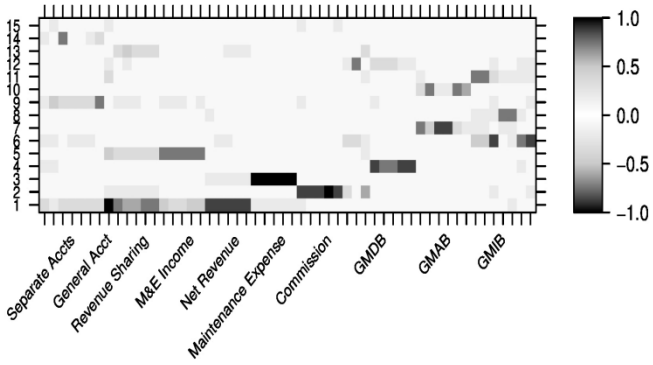


Figure 7: Interpretation of factors.

Factor scores are calculated for each observation or policy such that each one can be expressed in terms of q^* factors in place of the p original location variables. Clustering is then performed using 9 and 15 factors and the results are compared to those obtained by clustering with the same numbers of principal components. While PCA is a distribution-free descriptive technique, factor analysis assumes that the location variables are caused by unobservable factors. Brown (2009, p. 26-30) recommends using factor analysis when theoretical ideas about relationships between variables exist and suggests that PCA should be used if the goal of the researcher is to explore patterns in their data. In this case it is known, for example, that the present value of commission in one calibration scenario has a strong relationship with the present value of commission in the other four scenarios, suggesting that the factor analysis approach is valid.

3.4.2.3 Variable Clustering

Variable clustering (Sanche and Lonergan, 2006) is an alternative dimension-reduction technique used by

actuaries. It is more appropriate for reducing thousands of location variables and is not used here.

3.4.3 Dealing with Large Numbers of Clusters

The most widely used model-based clustering software is the R package **Mclust**. This was developed more for answering questions such as "does this moderately-sized data set have four clusters, or five?" than for partitioning large data sets into thousands of clusters. Direct application of model-based clustering to large datasets with large numbers of clusters can be prohibitively expensive in terms of computer time and memory (Fraley et al., 2005). For example, a VVV solution with 5000 clusters and 15 location variables requires the estimation of hundreds of thousands of parameters. This is not computationally feasible, especially bearing in mind that the original aim of this application is to ease the computational burden of analysis. It will therefore be necessary to impose some of the model constraints on the covariance matrices, for example only considering EII (equal volume, spherical) solutions.

The EM algorithm can be implemented directly for any model type for 50 clusters with appropriate starting values via laptop processing. With 250 clusters this is slow but possible. However, doing so for 1000 clusters or more, even for EII models, has massive computational requirements. A common remedy is to apply segmentation in advance of the data compression process. Alternatively, a novel adaptation of the model-based method relying on resampling techniques can be used. These methods are detailed in Section 3.4.3.1 and 3.4.3.2.

3.4.3.1 Segmentation

Segmentation reduces the number of clusters that need to be fitted at a particular iteration. This means splitting the data into a small number of segments and clustering within each segment. Insurers will often need to segment by line of business, asset class or some other categorical variable for reporting purposes in any case. Alternatively, it is possible to identify a small number of clusters in the data set, by either a model-based or a non-parametric method, and to use these as an initial segmentation before proceeding to cluster within each segment. It is necessary to decide in advance how many clusters are in each segment. Generally it is sensible to maintain the same compression ratio, i.e. if compressing N policies to G clusters overall, and there are S_i policies in segment i, then the aim is to have (S_i x G/N) clusters in segment i.

The idea is that policies from different segments should be unlikely to end up in the same clusters. In this way segmentation allows more flexible models to be fitted by easing the computational burden without compromising model fit. Any model-type constraints, e.g. that clusters must are equally shaped, only need to hold within segments. However, using a large amount of segments can separate similar policies that should otherwise be grouped together, leading to markedly worse fit. The effectiveness of segmentation therefore depends on the underlying distribution of the data.

3.4.3.2 Feedback Sampling

Feedback sampling is a novel model-based method developed in this paper for partitioning objects in a data set into G clusters, where G can be large, without ever having to run an E-step or an M-step for the whole data set. It takes advantage of the size-weighted nature of the data, the ability to merge similar objects and the fact that there is ultimately no interest in the values of Σ_k or τ_k. It has parallels with the ideas of data-squashing (DuMouchel et al., 1999), particularly in a likelihood-based context (Madigan et al., 2002). The process involves identifying G theoretical cluster centres, μ_k, through repeated sampling combined with model-based clustering. With this method, there is never a need to compute an (N x G) matrix of conditional probabilities

or a full set of parameters for all clusters. Feedback sampling can be implemented as follows:

1) Take a random sample of 2500 observations from the data.

2) Partition the sample into a moderate number of clusters (20-50) using weighted **Mclust**. BIC can be used to select the optimum model type and number of clusters *G*.

3) Treat the resulting cluster centres as *G* individual objects, scaled up by the sums of the sizes of the objects in each cluster.

4) Replace the sampled objects in the data set with these *G* scaled-up cluster centres, thus reducing the size of the data set by (*2500-G*).

5) Repeat until the desired number of objects remains.

6) Assign each original full data policy to the cluster whose centre is closest. Once the clusters are formed in this manner, selection of the representative policy for the cluster can proceed via the chosen selection method.

In Step 1, pure random sampling is not used but rather objects are sampled with probability inversely proportional to their size. This ensures that any objects that escape the sampling are likely to be large enough to merit ending up as cluster centres. Wehrens et al. (2004) suggest that a sample size of 2500 is sufficient and that there is no marked advantage to using larger samples than this size. The final step, which is non-parametric, can again be implemented efficiently in R using the **FNN** package. This is equivalent to a single step in an iteration of k-medoids clustering.

The range for the number of clusters fitted to each sample should include the sample size multiplied by the overall *G/N* compression ratio. If a specific number of final clusters is required, then it is necessary to fix *G* in the final iteration. This method can be applied to non-weighted data by initially assigning each object a weight of 1 and allowing them to be merged subsequently.

4. RESULTS

The aim of the application is to use cluster analysis to produce a compressed data set that replicates the behavior of the seriatim (full) data set as closely as possible. The representative policies produced by the cluster analysis will then be used to perform a series of stochastic simulations. Results obtained by various clustering methods at the five levels of compression are compared and contrasted.

4.1 Assessing Clustering Methods: Weighted Sums of Squares

Recall that, prior to scaling, the location variables are expressed in dollar amounts. The data are clustered based on scaled per-dollar values, so that policies within a cluster have, for example, similar present values of net revenue per dollar account value and will be merged. A policy's size, in dollars, affects its influence when clustering. Each cluster is ultimately reduced to a single representative policy, which takes the per-dollar location of the cluster's representative policy and the size of the sum of the sizes of all the policies in the cluster.

Hence for each location variable, the sum totals of the dollar values for the *N* policies should be closely matched by the sum totals of the dollar values of the *J* representative policies. The goodness of fit of a clustering solution can be described by measuring how closely these values match. The weighted sum of squared errors for a clustering solution is a single-figure summary statistic defined as:

$$WSS = \sum_{k=1}^{p} w_k\{1-(\sum_{j=1}^{J} x_{jk}^{comp} / \sum_{i=1}^{N} x_{ik}^{ser})\}^2 \qquad \text{XVIII}$$

where there are p variables, each with weight w_k, x_{jk}^{comp} is the dollar value of the k^{th} variable for the j^{th} representative policy in the compressed data set and x_{ik}^{ser} is the dollar value of the k^{th} variable for the i^{th}

policy in the seriatim data set. A lower value for the WSS indicates better fit.

The WSS is used to check the appropriateness of the Milliman EII initialization method for model-based clustering with non-EII models. The Milliman method proved to perform very well. For a random sample of 10,000 policies clustered into 50 representative policies, the WSS for the VVV model initialized using the Milliman method was 22.1 versus 76.0 for VVV hierarchical clustering initialization. The corresponding results were 5.1 versus 79.5 and 35.8 versus 54.3 for the VII and EII cases respectively.

Section 4.1.1 compares the effectiveness of the different variable reduction techniques used. Section 4.1.2 explores goodness-of-fit of the compressed data sets based on the within sample data used to perform the clustering. The compressed data sets are subsequently used to simulate present values of a number of variables across a range of out-of-sample scenarios. Section 4.1.3 examines the accuracy of these simulations relative to those based on the seriatim data.

4.1.1 Variable Reduction

The computational costs of performing model-based clustering on a large data set with 54 variables can be prohibitive at some levels of compression. However, since many of the variables are highly correlated, a dimension reduction technique can be used. Therefore both principal component analysis (PCA) and factor analysis (FA) were used to express the data in terms of 15 variables prior to clustering. With PCA, 15 components accounts for 97.3% of the variation in the data. In Figure 8, the WSS is calculated for solutions with 300, 400, 500, … 7,000 clusters based on Ward's minimum variance method for hierarchical clustering using the PCA data, the FA data and the full data. Results obtained using 9 instead of 15 principal components or using factor analysis were of much lower quality. However factor analysis may provide a useful alternative to PCA for alternative data sets.

Figure 8 represents three key findings: firstly that using 15 principal components is as good as using the full set of 54 location variables to cluster the data, while using the 15 factors obtained by orthogonal factor analysis is not. Figure 8 also shows that there is no substantial decrease in WSS as the number of clusters rises above 3000. This suggests that when using clustering for data compression, 3000 clusters should provide a good representation of the full data set and substantially better representations cannot necessarily be obtained even by using 5000 clusters or more. In this sense, Figure 8 can be viewed as a tool to help practitioners select the "optimal" compression level, if one is to be chosen, where for this data set 3000 representative policies appears to strike a good balance between computational burden and ensuring that the compressed data set contains representative policies that give an accurate portrayal of the full data. Finally, while the WSS generally decreases as the number of clusters increases, it does not do so smoothly, particularly for smaller numbers of clusters. This is because part of the error measured by the WSS statistic, which is due to discrepancies between the theoretical cluster centres and the actual representative objects, is random. PCA variable reduction with 15 components is hence the preferred method employed in the remainder of the paper.

4.1.2 Goodness of Fit Within Sample

The goodness of fit of compressed data sets obtained by different clustering methods is measured according to the location variable totals at each level of compression.

4.1.2.1 50 CLUSTERS

When model-based clustering is performed a model type must be chosen. With 50 clusters it is possible to implement the EM algorithm directly for any model type. Milliman's method is used for initialization. Log-

likelihood and BIC as well as WSS can be compared for each model type before deciding on the most suitable covariance structure. The log-likelihood is a measure of the probability of the data assuming that the data follow the Gaussian distributions in the models. Increased model complexity generally leads to higher log-likelihood. So if two proposed models have a similar likelihood the BIC will favour the simpler model. The WSS is based on the aggregate squared distances between the locations of the representative objects and the theoretical cluster centres.

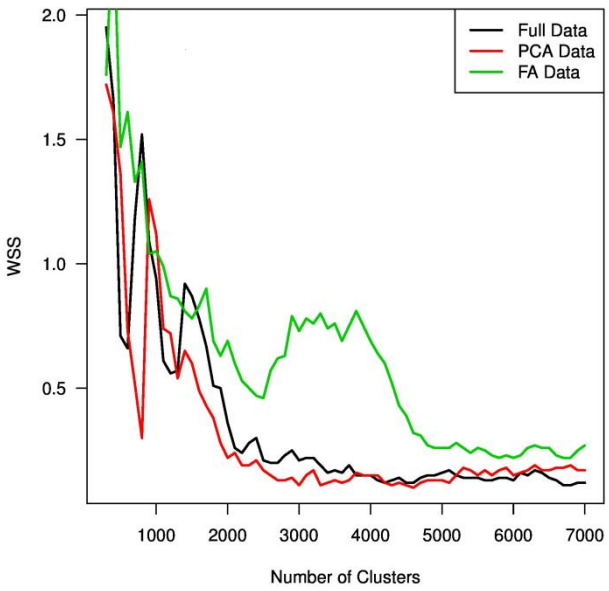


Figure 8: WSS under Ward's minimum variance hierarchical clustering.

While the WSS is a more relevant measure for a clustering solution in this particular application, it has a random element and should be treated as an indicative rather than a decisive measure.

Figure 9 shows how the different model types score according to the three criteria. The non-parametric methods -- Ward's, Milliman's, k-means and k-medoids -- are not probabilistic and therefore do not have a log-likelihood or BIC. Most of the model-based clustering solutions outperform the non-parametric methods according to WSS at this level of compression. This demonstrates that the Milliman method of initialization (corresponding to an EII model) performs well even when applied to non-EII model-based approaches: the best WSS results of 3.56 and 3.89 belong to the EEI and EEV models respectively.

The interesting finding is that models with equal volume (EEI, EEV, EVI and EII) all do better according to WSS than according to BIC or log-likelihood. This appears to be because, if clusters are of varying size, there is a possibility of having some very large clusters. These are more likely to have a larger distance between their mean locations and the nearest actual policy, which contributes a higher error to the WSS when policy nearest centroid selection is used, as is the case here. The equal volume constraint ensures that no representative policies are likely to be too far away from their cluster centres and is therefore useful when clustering for data compression with policy nearest centroid as the representative policy selection strategy. Table A1 shows all location variable totals for a variety of 50-cluster models as percentages of the seriatim values.

Most of the model-based methods outperformed the non-parametric methods at this level. Figure 10 compares the best clustering solution according to WSS, which is the EEI (equal volume, equal shape and parallel with the axes) model-based solution, with Milliman's method, according to the location variable totals. The y-axis measures the sum of each variable's values for the compressed data representative policies as a percentage of its sum across all policies in the full (seriatim) data set. So, for example, if the portfolio total net revenue in one of the calibration scenarios is $200m according to the seriatim data, and $202m according to a compressed data set, the graph shows 101% for that variable for the clustering method used to produce that

compressed data set. In such a graph an optimal solution would produce a flat line at 100% while a spike means that a particular variable is not well represented by the cluster-compressed data set.

As was remarked for Figure 8, information captured in the form of Figure 10 can also be used as a guide for practitioners seeking an optimal compression level for a data set. If the user has an a priori acceptable upper limit on the percentage variation in variable totals in the seriatim versus the compressed data set, they can establish how many representative policies are needed before the threshold is satisfied. So, for example, comparing Figure 10 and Figure 11 we see that if a maximum 25% disparity in variable totals between seriatim and compressed data set is permitted, then 50 clusters (representative policies) does not suffice but 250 clusters (or above) does suffice, for this data set.

4.1.2.2 250 CLUSTERS

With 250 clusters, Ward's method resulted in the lowest WSS. Only the simplest model-based method, EII, could be fitted directly for this data. More complex types such as EEV were fitted using segmentation. Where segmentation is used, the data are split into four roughly equally-sized segments according to the categorical variable IB Reinsurance Treaty. Model-based clustering is then used to partition to each segment. Table A2 shows all location variable totals for a variety of 250-cluster models as percentages of the seriatim values.

At this level of compression, the non-parametric approaches (Ward, k-means and k-medoids) outperform the model-based clustering approaches, the best of which still outperforms Milliman's method (see Figure 11). The segmentation approach performed particularly poorly. It appears that policies that ought to have been clustered together were artificially kept apart by the segmentation. This has important ramifications for insurers, many of whom frequently assume that such categorical variables give good policy separation.

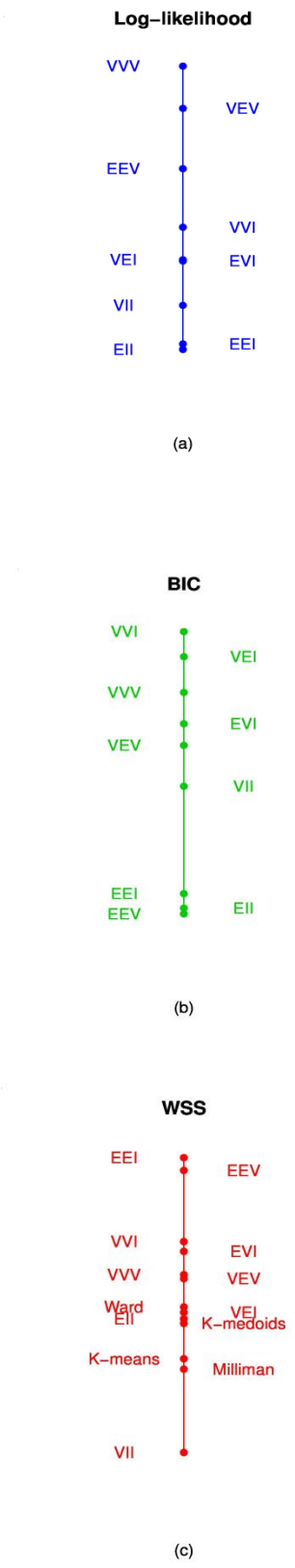


Figure 9: Scaled values of (a) Log-likelihood (b) BIC (c) WSS for the different model-types.

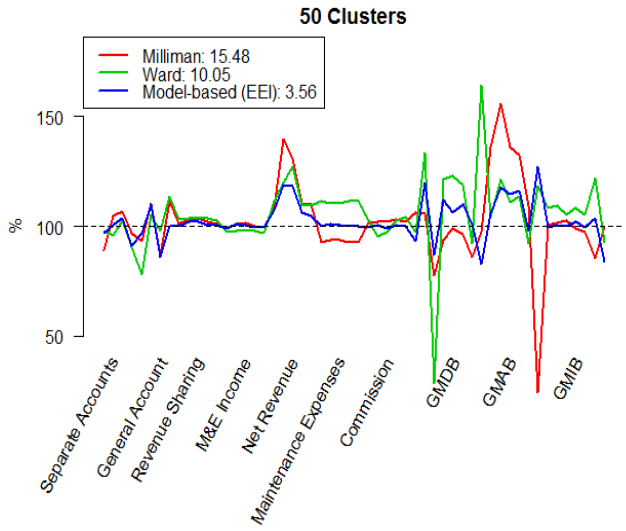


Figure 10: The best model-based method versus Ward and Milliman's methods, 50-clusters.

4.1.2.3 1000 CLUSTERS

With 1000 clusters it is not possible to perform model-based clustering directly via laptop processing. Model-based solutions were obtained using a high-power computer for two model types: EII and EEV. At this level, feedback sampling is implemented as an indirect approach to model-based clustering. Table A3 shows all location variable totals for a variety of 1000 cluster models as percentages of the seriatim values.

The best solution at this level is the one obtained by model-based clustering using feedback sampling (see Figure 12). The EII solution fitted directly with a high-powered computer is as good as the EEV solution fitted with the segmentation constraints. Both achieve low WSS values of 0.87 under the Milliman initialization method, again pointing to its suitability for use in preceding both the EII model to which it maps directly, but also covariance structures. Interestingly, the EEV solution fitted directly is not as good as either of these. Segmentation improves fit by removing the equal volume and shape constraints between different segments but also worsens fit by preventing policies from different segments from being clustered together.

It appears that, as the number of clusters increases, so too does the number of constraints that can be included. It is possible that the EEV solution obtained by the high-powered computer is only locally optimal and that a better solution may be obtained using different starting values.

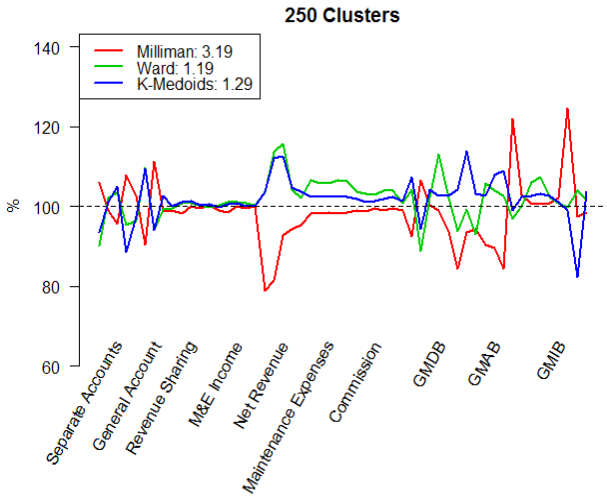


Figure 11: The best non-parametric and model-based methods versus Milliman's, 250 clusters.

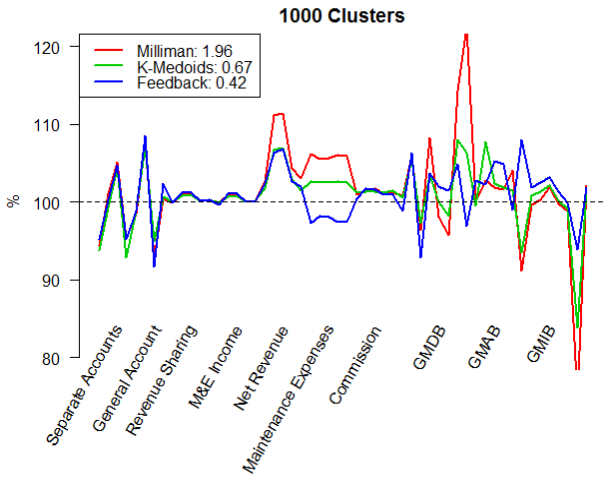


Figure 12: The best non-parametric and model-based methods versus Milliman's, 1000 clusters.

4.1.2.4 2500 CLUSTERS

For 2500 clusters and above, EII was the only model-based method that could be implemented directly. Ward's method is the best according to WSS, while the EII partition, obtained by a high-powered computer using feedback sampling, is better than the remaining non-parametric methods (see Figure 13). Table A4 shows all location variable totals for a variety of 2500-cluster models as percentages of the seriatim values.

4.1.2.5 5000 CLUSTERS

At this level of compression Ward's method is the best according to WSS, while feedback sampling with the model-based approach outperforms the other non-parametric methods (see Figure 14). Table A5 shows all location variable totals for a variety of 5000-cluster models as percentages of the seriatim values.

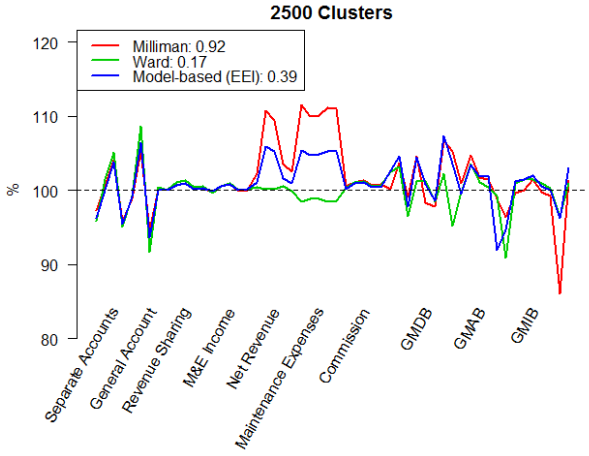


Figure 13: The best non-parametric and model-based methods versus Milliman's, 2500 clusters.

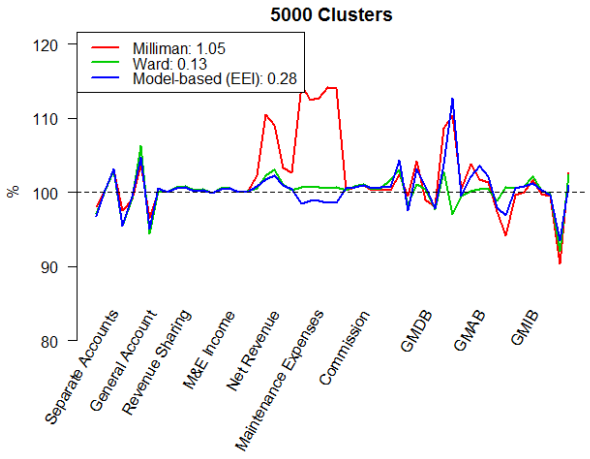


Figure 14: The best non-parametric and model-based methods versus Milliman's, 5000 clusters.

4.1.3 Goodness of Fit Out of Sample

Tables A1-A5 and Figures 10-14 measure the goodness of fit of a clustering solution relative to the location variables used to fit it. However, the critical test of the quality of a clustering method is how well the resulting compressed data set performs over a wide range of stochastic scenarios, distinct from those used to produce the representative policies. In this case, the present values of nine variables (net revenue, commission, revenue sharing, policy maintenance expenses, M&E fee income, net GMDB costs, net GMIB costs, net GMAB costs and Worst Surplus) at the end of a twenty-year period are calculated for each of 1000 scenarios. The distributions of these present values of each variable according to the synthesised representative policies can be compared with the distributions according to the seriatim data for each clustering technique for each variable.

These present values have been acquired from the seriatim data set, which contains 110,000 policies, and from compressed data sets, formed using Milliman's and Ward's clustering methods. In Sections 4.1.3.3 and 4.1.3.4 the distributions of the present values of two of the variables are examined in detail. Full sets of results

for Kolmogorov-Smirnov (KS) p-values, test statistics and scaled sum of squared differences (SSSD) can be found in Tables A6-A8. These metrics are documented in Sections 4.1.3.1 and 4.1.3.2. Notably, for 50 clusters where a variety of model-based covariance structures were tested, EEI and EEV outperformed EII for 5 of the 9 variables and matched it for the other 4 according to KS; and outperformed EII for 8 of the 9 variables according to the SSSD. Again it seems there is no negative implication for goodness of fit using the model-based approach with covariance structures different from the EII structure implied by Milliman's method, which was used for initialization purposes.

4.1.3.1 KOLMOGOROV-SMIRNOV TEST

The two-sample Kolmogorov-Smirnov test compares the distributions of data from two samples. The null hypothesis is that both come from the same probability distribution. In this case the first sample will be the present values of a particular variable in the 1000 scenarios according to the seriatim data and the second sample the same values according to a compressed data set formed by some clustering technique.

A high p-value indicates that the two samples come from the same distribution, and hence the compressed data set is a good representation of the seriatim data for that variable. If the p-value is low the null hypothesis is rejected and the conclusion is that the compressed data set is a poor representation of the seriatim data for that variable. The Kolmogorov-Smirnov test makes no assumptions about the data other than that observations in each sample are independently identically distributed from some continuous distribution, that is to say that the 1000 scenarios are independent. The test statistic quantifies the maximum absolute difference between the two empirical sample cumulative distribution functions (CDFs) over the range of values in the samples. It is sensitive to both the shape and location of the CDFs and so is useful for comparing distributions.

4.1.3.2 SCALED SUM OF SQUARED DIFFERENCES

A Scaled Sum of Squares (*SSS)* statistic is produced for each variable *j* for each compressed data set formed by clustering method *m*. The *SSS* is calculated using

$$T_{sj}^{m} = \sum_{i=1}^{N} x_{sij}^{m} \qquad \text{XIX}$$

where x_{sij}^{m} is the present value, in dollars, of variable *j* for representative policy *i* according to model *m*, in scenario *s*. It follows that:

$$SSS_{j}^{m} = \sum_{s=1}^{1000} (T_{sj}^{m} - T_{sj}^{seriatim})^{2} \qquad \text{XX}$$

The scenarios are ordered from lowest to highest for each variable. Note that this order is not necessarily constant between variables or models. The interest lies in the differences between the model and seriatim in the ultimate distribution of present values across the 1000 scenarios rather than on a scenario-by-scenario basis. Next these *SSS* values are scaled so that the standard deviation for each variable is one, producing the *SSS* results in Table A8. This allows the calculation of the "Total" column, a single-figure summary for each clustering method:

$$SSS_{Total} = \sum_{j=1}^{p} SS_{j}^{m} \qquad \text{XXI}$$

Since the statistics are based on the sum of squared differences between the modelled values and the seriatim values, the better-performing methods are those with lower *SSS* statistics. It can be seen that model-based clustering performs best across all variables for 50 and 5000 clusters respectively (the latter employing feedback sampling). Ward's method is optimal at 250 clusters and k-medoids has the lowest SSS for 1000 and 2500 clusters.

4.1.3.3 M&E FEE INCOME

Kolmogorov-Smirnov tests were performed to compare each of the compressed data sets to the seriatim values.

Table 1: Kolmogorov-Smirnov p-values for present value of M&E fee income.

Number of Clusters	5000	2500	1000	250
Milliman KS p-value	0.24	0.24	0.31	0.01
Ward KS p-value	**1.00**	0.98	0.29	**0.98**
Model-based KS p-value	**1.00**	**1.00**	0.89	0.06
K-Medoids KS p-value	N/A	0.95	**0.97**	N/A

The higher p-values for Ward's method (see Table 1) indicate that it has resulted in a better fit than Milliman's method for this variable when compressing to 250 clusters. Model-based and k-medoids methods perform well for larger numbers of clusters. Figure 15 shows the distribution of the present value of this variable across the 1000 scenarios for the seriatim data and for the approximations based on Ward's and Milliman's methods with 250 clusters. Figure 15(b) shows the full distribution while Figures 15(a) and 15(c) focus on the tails. Figures 16 and 17 depict the equivalent information for models using 2500 and 5000 clusters respectively.

The cluster compressed data based on Ward's and model-based methods are generally closer to seriatim than Milliman's for this variable. Sections 4.1.3.4 – 4.1.3.5 contain similar analysis for further variables.

4.1.3.4 Maintenance Expenses

For the policy maintenance expenses variable, the p-value from the Kolmogorov-Smirnov test is almost zero in all but two of the fitted models (see Table 2). While this implies that most of these models poorly represent this variable, the p-value is based on a test statistic that can still be examined to give an indication of relative goodness of fit. A lower value of the test statistic indicates better fit, meaning that Ward's method provides a better representation of this variable than Milliman's, particularly in the 2500 and 250-cluster solutions (see Table 3). The model-based approach based on feedback sampling is optimal for 1000 and 5000 cluster solutions.

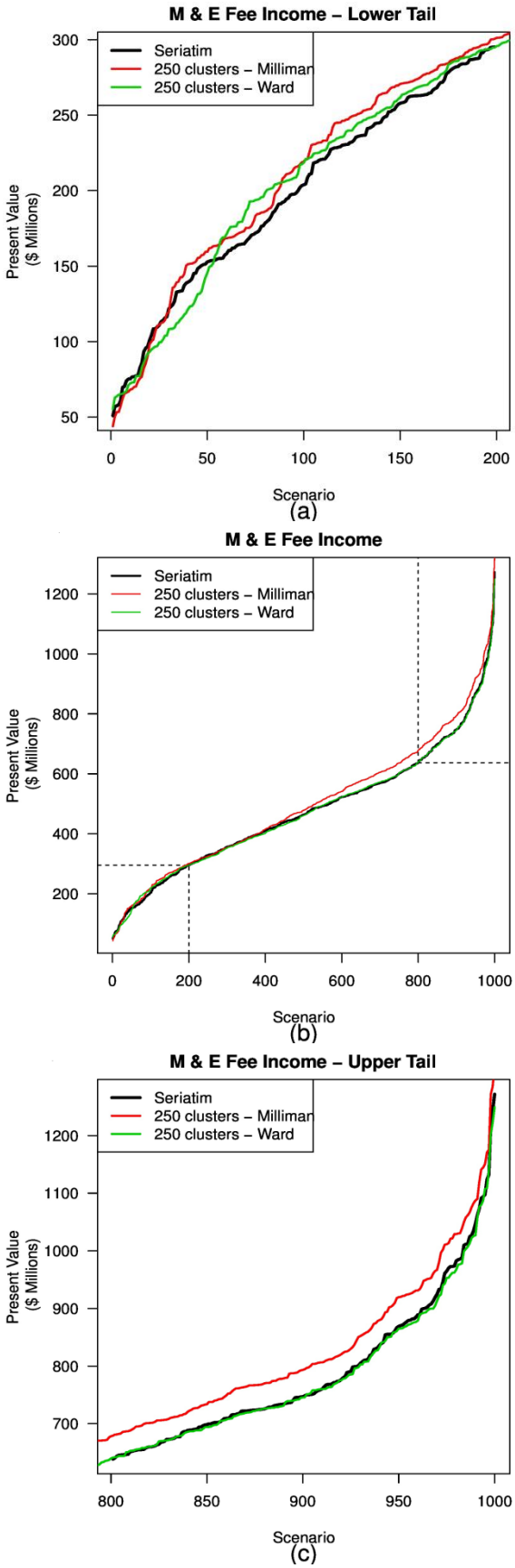


Figure 15: Present value M&E fee income for seriatim and 250 cluster models.

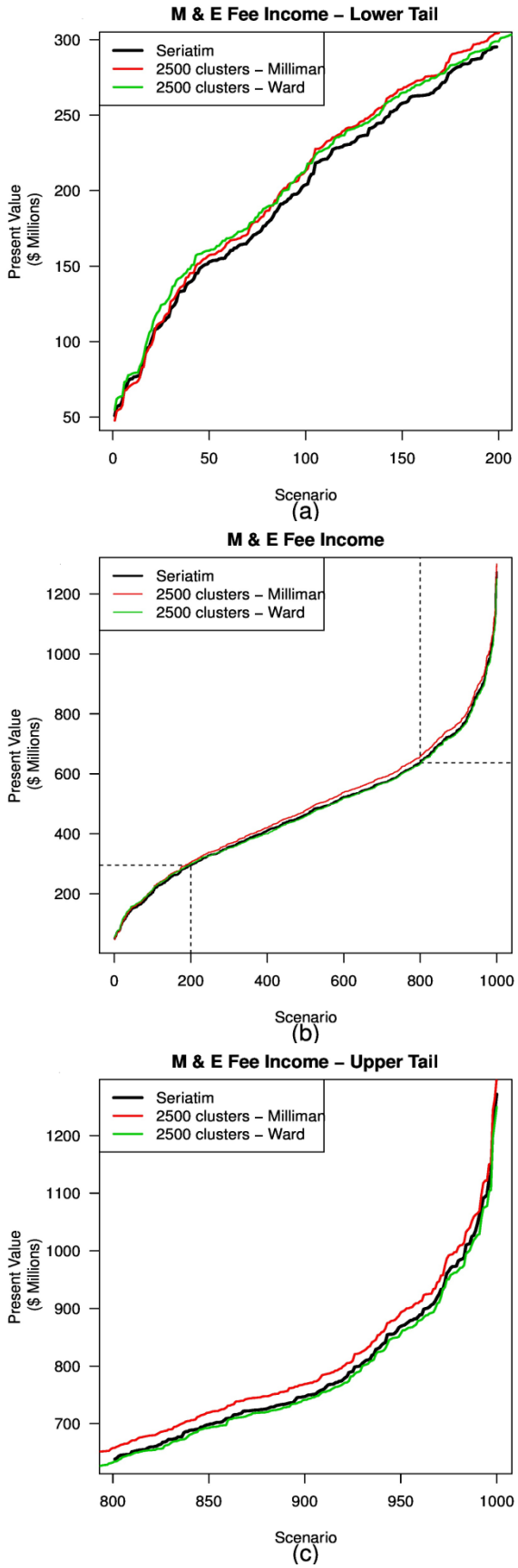


Figure 16: Present value M&E fee income for seriatim and 2500-cluster models.

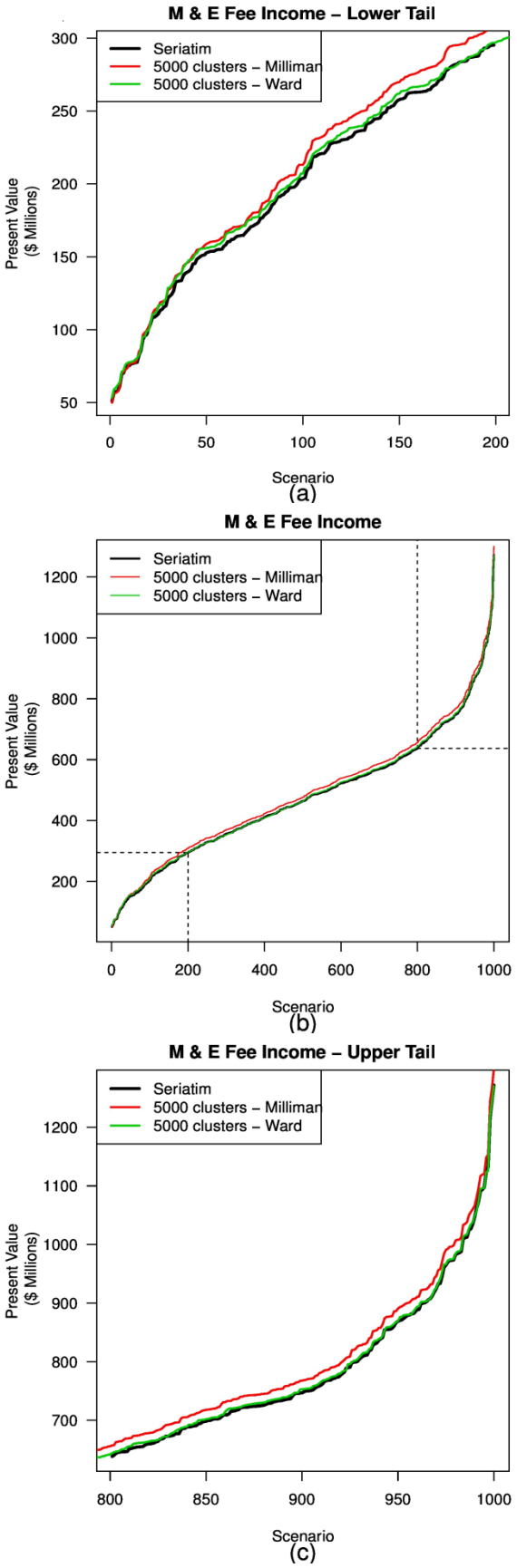


Figure 17: Present value M&E fee income for seriatim and 5000-cluster models.

Table 2: Kolmogorov-Smirnov p-values for present value of policy maintenance expenses.

Number of Clusters	5000	2500	1000	250
Milliman KS p-value	<0.001	<0.001	<0.001	<0.001
Ward KS p-value	<0.001	0.02	<0.001	**0.79**
Model-based KS p-value	**<0.001**	<0.001	**0.34**	<0.001
K-Medoids KS p-value	N/A	**0.03**	<0.001	N/A

Table 3: Kolmogorov-Smirnov test statistics for present value of policy maintenance expenses.

Number of Clusters	5000	2500	1000	250
Milliman KS statistic	0.11	0.12	0.12	0.12
Ward KS statistic	0.11	0.07	0.19	**0.03**
Model-based KS statistic	**0.08**	0.10	**0.04**	0.13
K-Medoids KS statistic	N/A	**0.06**	0.11	N/A

The distributions of the present values of this variable are plotted in Figures 18-20 for the 250, 2500 and 5000-cluster models respectively. Again, figure (b) in each case shows the full distributions while figures (a) and (c) show the tails. Ward's and model-based methods respectively are closer to the seriatim outcome than Milliman's method for policy maintenance expenses, apart from in the lower tail.

4.1.3.5 NET REVENUE

Net revenue is regarded as the most important variable for this data set (partly evidenced in it holding the largest variable weighting as set out by Milliman in Section 2.) Consequently it is analysed both through the prism of clustering methodology, as for the previous variables, but also representative policy selection method.

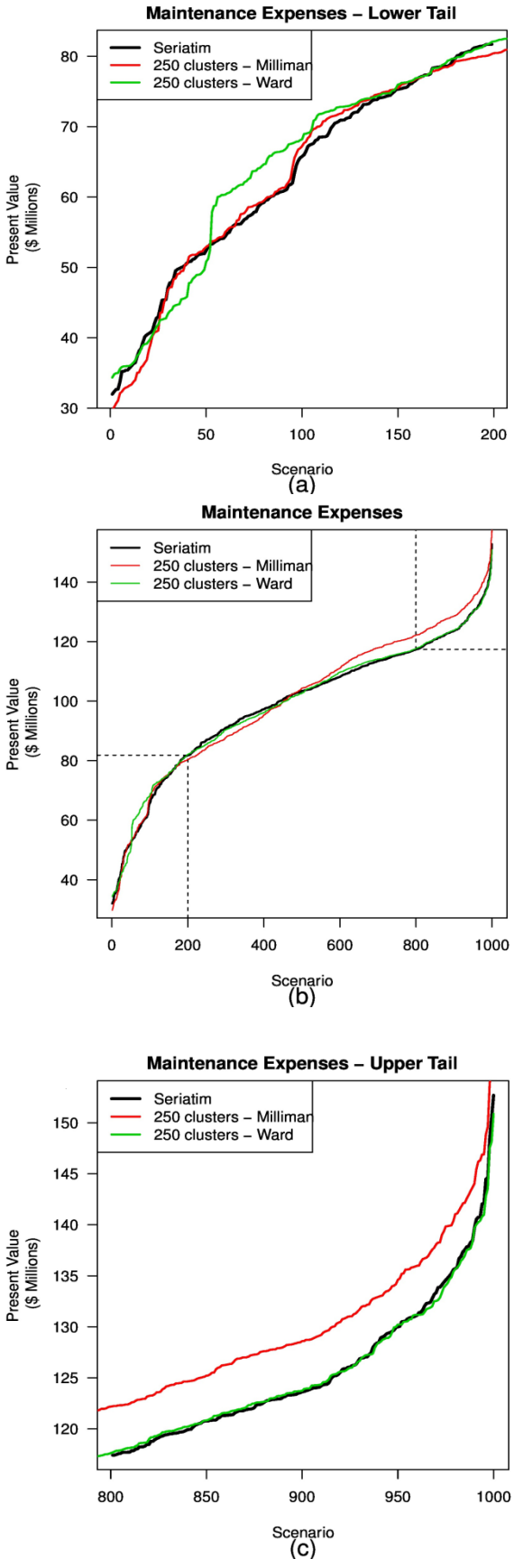
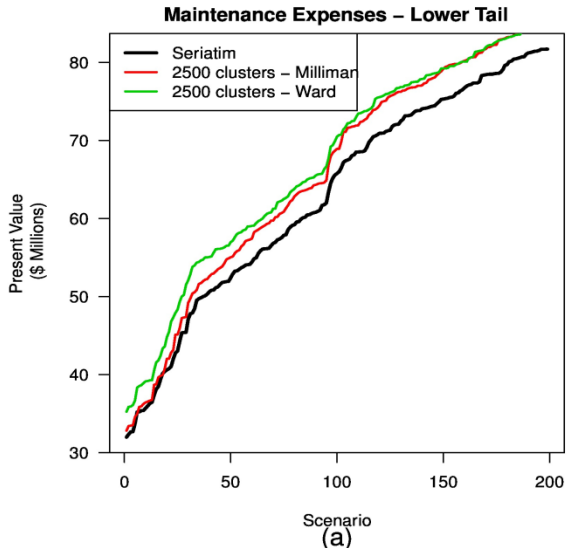
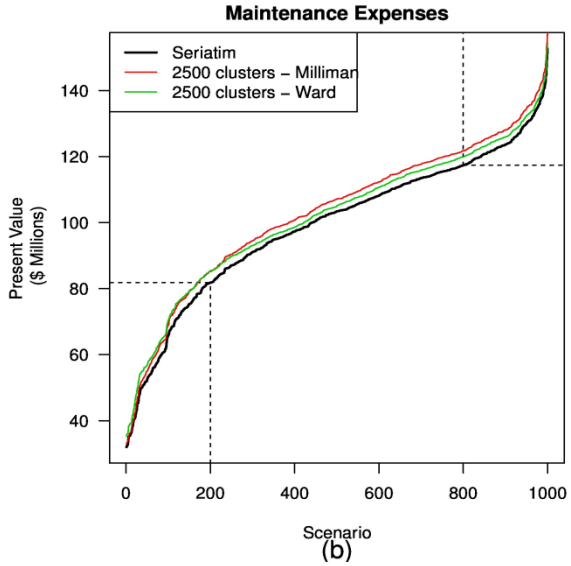


Figure 18: Present value policy maintenance expenses for seriatim and 250 cluster models.





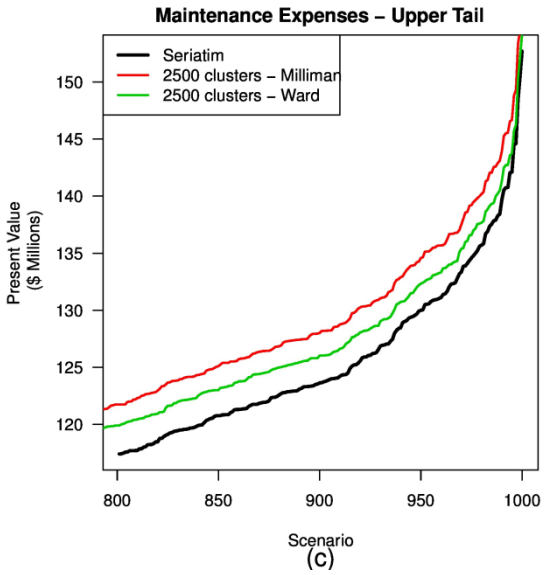
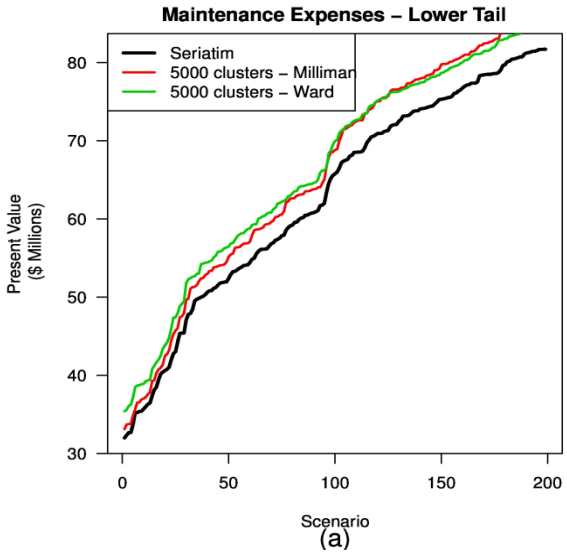
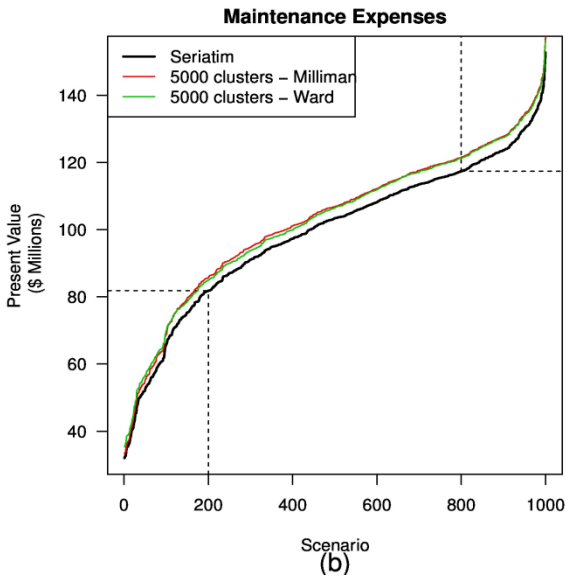


Figure 19: Present value policy maintenance expenses for seriatim and 2500-cluster models.





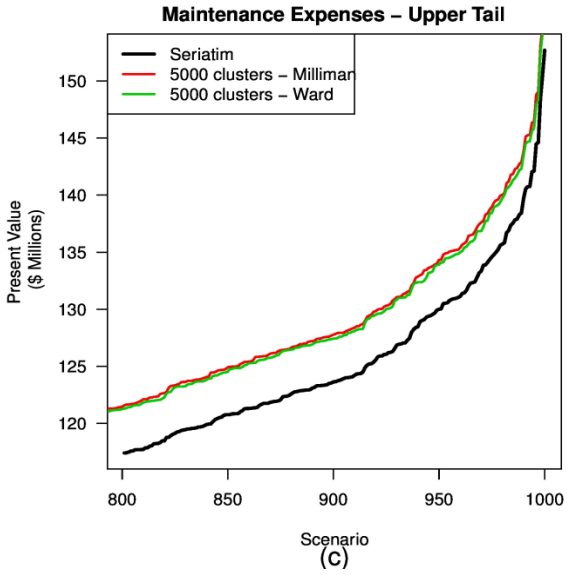


Figure 20: Present value policy maintenance expenses for seriatim and 5000-cluster models.

Table 4: Kolmogorov-Smirnov test statistics for present value of net revenue.

Number of Clusters	5000	2500	1000	250
Milliman KS statistic	**0.116**	0.133	0.289	0.296
Ward KS statistic	0.163	0.115	0.354	**0.125**
Model-based KS statistic	**0.116**	0.183	0.076	0.267
K-Medoids KS statistic	N/A	**0.111**	**0.056**	N/A

Figure 21 and Table 4 illustrate strongly that the optimal compression technique and the quality of compression can vary markedly according to the number of clusters/representative policies required. For 5000 representative policies Milliman's approach and the model-based approach are tied for accuracy according to the KS statistic, since both experience their single biggest departure from the seriatim in the lower tail. However over the full range of scenarios the model-based approach (implemented via feedback sampling) is superior. For 2500 clusters k-medoids provides the most accurate set of representative policies. At the 1000 cluster level of compression k-medoids is again the optimal approach, by a very narrow margin versus the model-based method, both of which are substantially more accurate than the Milliman methods (though Ward is best for compression to 250 clusters).

Figure 22 illustrates the present value of net revenue across 1000 economic scenarios for 1000 clusters, where the clusters are identified using k-medoids and the representative policy is chosen via all potential methods outlined in Section 1.3. The default approach of using the policy nearest the centroid (denoted "Centroid") is the best-performing selection method for the k-medoids approach for this variable and compression level (KS statistic = 0.056). However it is closely followed by the random selection method with weights proportional to policy size ("RandSize", KS statistic = 0.059) and random selection using weights proportional to distance from centroid ("RandLoc", KS statistic = 0.087); and to a lesser extent by complete random selection ("Random", KS statistic = 0.162). Hence it is reassuring to observe that, if there is a concern as to the potential underestimation of variance by the policy nearest centroid selection method, random selection of representative policy using one of the three available sets of weights provides a viable alternative that can still provide a high quality compression.

It is interesting to note that the modified centroid approach ("ModCent", KS statistic = 0.325) performs poorly in conjunction with the k-medoids approach for Net Revenue. This is generally true across location variables when the modified centroid approach to representative policy selection is combined with nonparametric approaches to clustering. Conversely the modified centroid approach generally works well in conjunction with model-based clustering. Consider for example Net Revenue at 5000 clusters, where model-based clustering with feedback ties Milliman as the optimal compression method: a further reduction in the KS statistic from 0.116 to 0.073 can be achieved by moving from the policy nearest centroid to the modified centroid selection method. At the other end of the compression spectrum, at 250 clusters, where the Ward approach (equivalent to model-based clustering with EII covariance for unweighted data) is optimal, the KS statistic can be reduced from 0.125 to 0.107 by switching to the modified centroid representative policy selection approach.

4.1.4 CTE70

The CTE (conditional tail expectation) 70 is a summary figure often used by actuaries to represent the average present value across the worst 30% of scenarios. It is defined as such to mirror a key reserving requirement set by industry regulators (Junus and Motiwalla, 2009). Here the CTE70 for the worst present value of the end of year surpluses through to the end of year 20 is considered. This type of tail analysis - only considering

the worst case scenarios - is important as it drives reserves for this type of product.

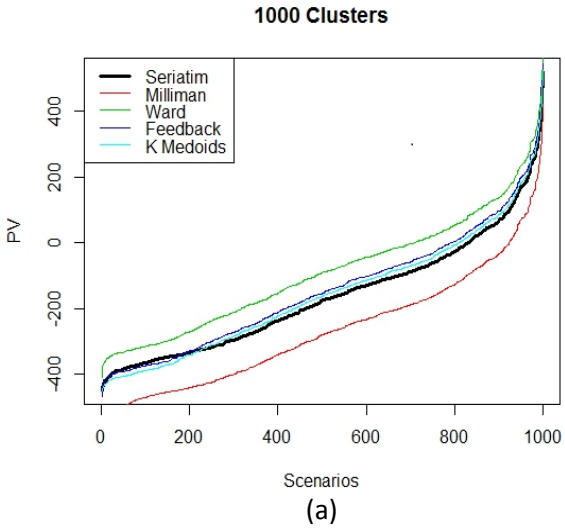
For each scenario, the present value of surplus for the portfolio is calculated at the end of each of the next 20 years. The worst of these is taken for that scenario. The CTE70 is the average of these worst present values across the 300 worst scenarios. With this product, initial surplus is always zero and in about 80% of scenarios surplus is never negative so the worst present value of surplus is actually zero in all but about 200 of the 1000 scenarios. Hence the average of 300 scenarios contains approximately 100 zero values.

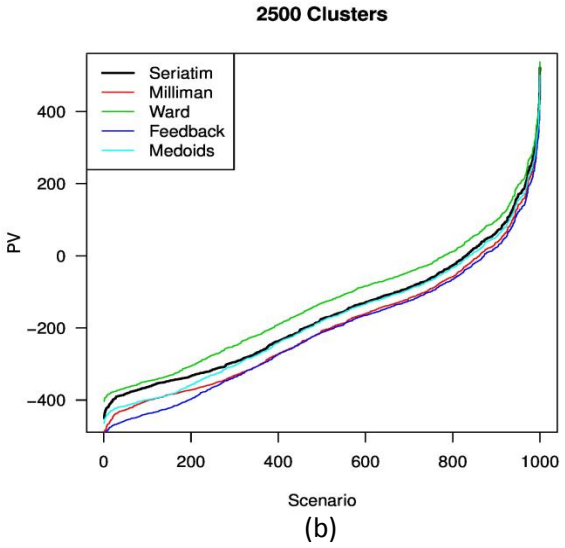
The Alternative results in Table 5 are model-based for 50 and 250 clusters, k-medoids for 1000 and 2500 clusters and model-based with feedback sampling for 5000 clusters. These methods generally perform well, yielding results that are close to but not quite as good as Milliman's method according to the CTE70 metric, except for the 1000 cluster case in which k-medoids clustering outperforms the Milliman method. This is despite the methods having better quality of fit overall across all variables. All Alternative results use the policy nearest centroid as the representative policy for each cluster, in accordance with the Milliman approach.

The Alternative* results in Table 5 allow both the method of clustering *and* the means of selecting a representative policy to vary: policy nearest centroid, random selection, random selection with weights based on size, random selection with weights based on distance from centroid and the modified centroid approaches are all considered (see Section 1.3).

Table 5: CTE70 results for worst present value of surplus.

	Seriatim	Milliman	Alternative	Alternative*
5000 clusters	100.0%	99.3%	98.7%	98.7%
2500 clusters	100.0%	99.2%	99.4%	99.4%
1000 clusters	100.0%	98.6%	97.7%	100.3%
250 clusters	100.0%	97.9%	97.8%	97.8%
50 clusters	100.0%	95.2%	105.7%	105.7%





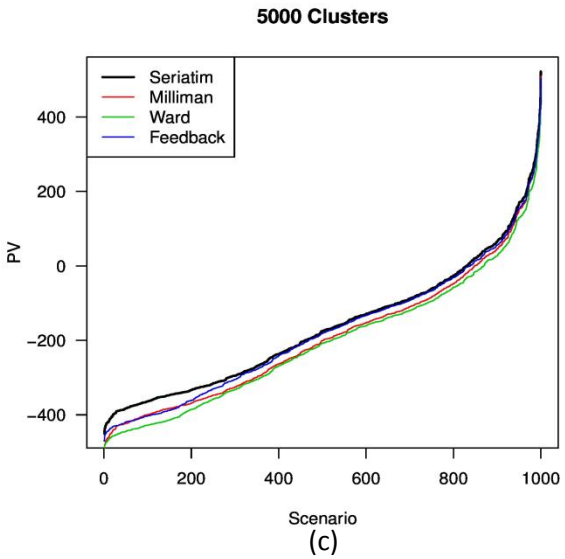


Figure 21: Present value of net revenue for (a) 1000, (b) 2500 and (c) 5000-cluster models.

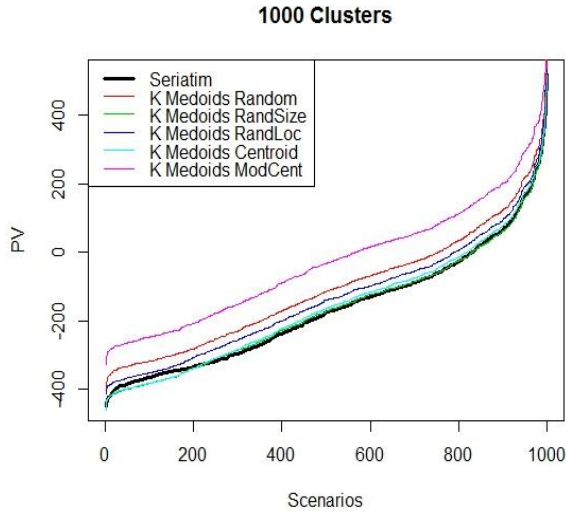


Figure 22: Present value of net revenue for 1000 clusters using k-medoids with a variety of representative policy selection methods.

Despite this, the default selection method of policy nearest centroid prevails at all compression levels except 1000 clusters for CTE, where a slightly better result for the K-medoids model can be obtained by selecting representative policies randomly within clusters using weights proportional to policy size (and generally performs well for CTE70 for other compression levels). This suggests that selecting the representative policy using the standard policy nearest centroid approach is a robust approach for optimizing the critical CTE metric and, by extension, for the application as a whole. Should one wish to avoid this type of representative policy, random selection with weights proportional to policy size appears to provide a good alternative.

The CTE70 results can also act as a useful tool for practitioners seeking to identify an "optimal" number of representative policies for a given data set. For example, for this data set there is an improvement in accuracy of CTE70 results as the number of representative policies increases to 2500, but little to be gained from increasing to 5000 policies, according to this metric. This out of sample result (across all scenarios as opposed to only the calibration scenarios) tallies with the in-sample interpretation of the optimal number of representative policies gleaned using the Weighted Sum of Squares metric in Figure 8.

5. CONCLUSIONS

A large number of clustering algorithms exist. The most appropriate one depends on the nature of the data, the purpose of the clustering and the level of compression.

Within sample, the model-based, k-medoids and Ward methods proposed have outperformed Milliman's clustering method at all five levels of compression tested, according to the location variable totals for the sample data set. When the clustering methods are compared across stochastic simulations out of sample, the method with the lower Weighted Sum of Squares has generally produced representative policies with superior accuracy over the full range of variables, as expected. To this end, model-based, Ward and k-medoid clustering show great promise as alternative clustering compression methodologies for stochastic simulations. In terms of the quality of fit of compressed data points for the Worst Surplus variable that informs the CTE70 metric, the existing Milliman methodology is generally most precise for the data set tested, though by a small margin in some cases.

When the number of clusters is very small or very large, model-based clustering appears to generally outperform the non-parametric methods. An advantage of the model-based method is that the clusters identified are Gaussian, i.e. symmetric and bell-shaped, and are therefore better represented, on average, by single policies near their centroids than clusters obtained by nonparametric methods, which may be skewed, heavy-tailed or otherwise irregularly shaped. Compact clusters with low variance that have real objects close to their centroids are usually ideal. These may not exist in a data set when a small number of clusters are sought. In

this situation a model-based approach with the equal volume constraint appears best for partitioning the data.

Although large data sets, in theory, admit much more complex models, it is not computationally practical to directly perform unconstrained model-based clustering with large numbers of clusters and variables. Spherical model constraints such as EII can be suitable when partitioning data into large numbers of clusters. Indeed, the equal volume constraint appears essential. Feedback sampling is an alternative way of using the model-based approach indirectly to obtain a good partition even for extremely large data sets with large numbers of clusters. Segmentation can be used to good effect to fit flexible models if the distribution of the data so permits. However, for moderate to large numbers of clusters, non-parametric methods such as k-medoids and Ward's method seem to be effective and are easy to implement.

The results obtained by Ward's minimum variance hierarchical clustering method and k-medoids were generally good at most compression levels for the variable annuities data. Both of these non-parametric clustering algorithms can be implemented efficiently in R. It is useful to perform a dimension reduction step prior to clustering if there are many location variables, particularly if some of the variables are strongly correlated. Principal component analysis proved to be more suitable than orthogonal factor analysis for this actuarial application. The alternative representative policy selection methods based on randomness and randomness using weights based on policy size or distance from the centroid show promise if it is desirable not to use the default policy nearest centroid method, but the latter does perform well in general.

6. FURTHER WORK

6.1 Ward's method for Weighted Data

Ward's minimum variance method performs well in this application. Since each cluster is ultimately represented by a single object, in the optimal solution objects within clusters are as close as possible to the representative object. By minimizing the variance in each cluster, the algorithm minimizes the loss of information in the data compression.

When Ward's method has been applied here, all objects have been treated equally and the weighted nature of the data has been ignored. This is because adapting this method to account for weighted data is not possible with the available software. Ward's method, as implemented, minimizes total within-cluster variance:

$$\sum_{k=1}^{G} \sum_{i=1}^{N_k} \sum_{j=1}^{p} \frac{1}{n_k} (x_{ij} - \bar{x}_{kj})^2 \qquad \text{XXII}$$

But, in order to account for policy size, the aim should instead be to minimize the total within-cluster *size-weighted* variance:

$$\sum_{k=1}^{G} \sum_{i=1}^{N_k} \sum_{j=1}^{p} \frac{1}{n_k} \; w_i (x_{ij} - \bar{x}_{kj}^{w})^2 / \sum_{i=1}^{n_k} w_i \qquad \text{XXIII}$$

where w_i is the size of policy i and $\bar{x}_{kj}^{w}$ is the *weighted* mean value of variable j for the objects in cluster k. Performing Ward's minimum variance hierarchical clustering in this manner would be likely to lead to further gains in quality of the compressed data points.

6.2 Variable Weights

Prior to clustering, each variable was assigned a weight. When analysing the compressed data sets, it is apparent that some variables are more accurately represented than others. When clustering for a specific purpose, e.g. for the calculation of the CTE70, which focuses only on the worst-case scenarios, that purpose should be reflected in the variable weights.

A technique such as boosting (Bauer and Kohavi, 2009), which is used in classification algorithms, can be used to optimize the variable weights. After an initial clustering solution is obtained, the worst-fitting variables can be identified. These can then be given larger weights and the clustering performed again until all variables appear sufficiently well represented.

6.3 Bayesian Model Averaging

It is possible to obtain partitions from a variety of clustering methods. If, for one level of compression, a number of solutions are obtained and it is not clear that one is the best (according to BIC, log-likelihood, WSS or any other measure), Bayesian Model Averaging (BMA) may be used to take an average across several viable solutions (Hoeting et al., 2009). To do this, consider a partition in terms of the Z matrix where z_{ik} is the probability that object i belongs to cluster k. Each object x_i will be assigned to the cluster for which it has the highest probability of membership. If there are M viable solutions, $Z_1 \ldots Z_M$, then Z_{BMA} is a weighted average of $Z_1 \ldots Z_M$. The BIC associated with Z_a, $a=1 \ldots M$, is used to calculate its corresponding weight, which is proportional to $exp(BIC_a/2)$.

7. REFERENCES

Ackerman, M. et al. (2012). Weighted Clustering. *In Proceedings of the 26th AAAI Conference on Artificial Intelligence*.

Anderberg, M.R. (1973). Cluster Analysis for Applications. Academic Press.

Banfield, J.D. and Raftery, A.E. (1993). Model-based Gaussian and non-Gaussian clustering. *Biometrics* 49:803-821.

Bauer, E. and Kohavi, R. (1999). An Empirical Comparison of Voting Classification Algorithms: Bagging, Boosting, and Variants. *Machine Learning* 36(1-2):105-139.

Bellas, A., Bouveyron, C., Cottrell, M. and Lacaille, J. (2013), Model-based Clustering of High-Dimensional Data Streams with Online Mixture of Probabilistic PCA, *Advances in Data Analysis and Classification*, 7 (3):281-300.

Ben-Hur, A. and Guyon, I. (2003). Detecting Stable Clusters using Principal Component Analysis. *In Functional Genomics: Methods and Protocols*, ed. M.J. Brownstein and A. Kohodursky, pp. 159-182. Humana press.

Beygelzimer, A., Kakadet, S., Langford, J., Arya, S., Mount, D., Li, S. and Li, M.S. (2013). FNN: Fast Nearest Neighbor Search Algorithms and Applications. R package version 1.1.

Brown, J.D. (2009). Statistics Corner Questions and Answers about Language Testing Statistics: Principal Components Analysis and Exploratory Factor Analysis—Definitions, differences, and choices. *Shiken: JALT Testing & Evaluation SIG Newsletter*, 13(1): 26-30.

Celeux, G. and Govaert, G. (1995). Gaussian Parsimonious Clustering Models. *Pattern Recognition* 28(5):781-793.

Chang, W.C. (1983). On Using Principal Components Before Separating a Mixture of Two Multivariate Normal Distributions. *Journal of the Royal Statistical Society, Series C (Applied Statistics)*, 32(3): 267–275.

De la Cruz-Mesía, R., Quintana, F.A. and Marshall, G. (2008). Model-based Clustering for Longitudinal Data. *Computational Statistics and Data Analysis* 52(3):1441-1457.

Dempster, A.P., Laird, N.M. and Rubin, D.B. (1977). Maximum Likelihood for Incomplete Data via the EM Algorithm. *Journal of the Royal Statistical Society* 39:1-38.

DuMouchel, W., Volinsky, C., Johnson, T., Cortes, C. and Pregibon, D. (1999). Squashing Flat Files Flatter. *In Proceedings of the Fifth ACM Conference on Knowledge Discovery and Data Mining*, pp. 6–15.

Donoho, D.L. (2000). High-Dimensional Data Analysis: The Curses and Blessings of Dimensionality. *AMS Math Challenges Lecture*, pp. 1-32

Fayyad, U. and Smyth, P. (1995). From Massive Data Sets to Science Catalogs: Applications and Challenges. *In Proceedings of the Workshop on Massive Data Sets, National Research Council*, pp. 129-142.

Fraley, C. and Raftery, A.E. (2002). Model-based Clustering, Discriminant Analysis and Density Estimation. *Journal of the American Statistical Association* 97:611-631.

Fraley, C., Raftery, A.E. and Scrucca, L. (2012). **mclust** version 4 for R: Normal Mixture Modeling for Model-Based Clustering, Classification, and Density Estimation. *Technical Report No. 597*, Department of Statistics, University of Washington.

Fraley, C., Raftery, A.E. and Wehrens, R. (2005). Incremental Model-Based Clustering for Large Datasets with Small Clusters. *Journal of Computational and Graphical Statistics* 14(3):529-546.

Freedman, A. and Reynolds, C. (2008). Cluster Analysis: A Spatial Approach to Actuarial Modelling. Milliman Research Report. Online at http://www.milliman.com/insight/research/insurance/Cluster-analysis-A-spatial-approach-to-actuarial-modeling/?lng=1048578.

Friedman, J. (1989). Regularized Discriminant Analysis. *Journal of the American Statistical Association* 84: 165-175.

Harman, H.H. (1960). *Modern Factor Analysis*. Oxford, England: University of Chicago Press.

Hoeting, J.A., Madigan, D., Raftery, A.E. and Volinsky, C.T. (1999). Bayesian Model Averaging: a Tutorial. *Statistical Science* 14(4):382-401.

Husson, F., Josse, J., Le, S. and Mazet, J. (2014). FactoMineR: Multivariate Exploratory Data Analysis and Data Mining with R. R package version 1.26.

Jolliffe, I. (2002). *Principal Component Analysis*. Springer. 2nd Edition.

Johnson, S. (1967). Hierarchical Clustering Schemes. *Psychometrika* 32(3):241-254.

Junus, N. and Motiwalla, Z. (2009). A Discussion of Actuarial Guideline 43 for Variable Annuities.
Kaiser, H.F. (1958). The Varimax Criterion for Analytic Rotation in Factor Analysis. *Psychometrika* 23:187-200.

Lange, K.L. and Zhou, H. (2010). On the Bumpy Road to the Dominant Mode. *Scand Stat Theory Appl*. 37(4):612-631.

Lê, S., Josse, J. and Husson, F. (2008). FactoMineR: An R Package for Multivariate Analysis. *Journal of Statistical Software*. 25(1)**:**1-18.

MacLachlan, G.J. and Krishnan, T. (1997). *The EM Algorithm and Extensions*. New York: Wiley.

Madigan, D., Raghavan, N., Dumouchel, W., Nason, M., Posse, C. and Ridgeway, G. (2002). Likelihood-Based Data Squashing: A Modeling Approach to Instance Construction. *Data Mining and Knowledge Discovery* 6(2):173–190.

Maechler, M., Rousseeuw, P., Struyf, A., Hubert, M. and Hornik, K. (2015). cluster: Cluster Analysis Basics and Extensions. R package version 2.0.3.

Mar, J.C. and McLachlan, G.J. (2003). Model-based Clustering in Gene Expression Microarrays: an Application to Breast Cancer Data. In *Proceedings of the First Asia-Pacific Bioinformatics Conference on Bioinformatics* 19.

Massey, F.J. (1951). The Kolmogorov-Smirnov Test for Goodness of Fit. *Journal of the American Statistical Association* 46(253):68-78.

McParland, D. and Gormley, I. (2014). Model Based Clustering for Mixed Data: clustMD. *Advances in Data Analysis and Classification*. Online at http://dx.doi.org/10.1007/s11634-016-0238-x

Müllner, D. (2013). fastcluster: Fast Hierarchical, Agglomerative Clustering Routines for R and Python. *Journal of Statistical Software* 53(9):1-18. R package version 1.1.16.

Murphy, T.B. and Scrucca, L. (2012). Using Weights in **mclust**.

Murphy, T.B., Dean, N. and Raftery, A.E. (2010). Variable Selection and Updating In Model-Based Discriminant Analysis for High Dimensional Data with Food Authenticity Applications. *Annals of Applied Statistics* 4:396-421.

Neumann, J., Cramon, D. and Lohmann, G. (2008). Model-Based Clustering of Meta-Analytic Functional Imaging Data. *Human Brain Mapping* 29(2):177-192.

Pearson, K. (1901). On Lines and Planes of Closest Fit to Systems of Points in Space. *Philosophical Magazine* 2(11): 559-572.

Posse, C. (2001). Hierarchical Model-Based Clustering for Large Datasets. *Journal of Computational and Graphical Statistics* 10:464-486.

Reynolds, C. and Man, S. (2008). Nested Stochastic Pricing: The Time Has Come. Product Matters. *Society of Actuaries* 71 (2008): 16-20.

Sanche, R. and Lonergan, K. (2006). Variable Reduction for Predictive Modelling with Clustering. *Casualty Actuarial Society Forum* pp. 89-100.

Spearman, C. (1904). General Intelligence, Objectively Determined and Measured. *American Journal of Psychology* 15 (2):201-292.

Van der Laan, M., Pollard, K. and Bryan, J. (2003). A new Partitioning Around Medoids Algorithm. *Journal of Statistical Computation and Simulation* 73(8):575-584.

Ward, J.H. (1963). Hierarchical Grouping to Optimize an Objective Function. *Journal of the American Statistical Association* 58(301):236-244.

Wehrens, R., Buydens, L.M., Fraley, C. and Raftery, A.E. (2004). Model-Based Clustering for Image Segmentation and Large Datasets via Sampling. *Journal of Classification* 21(2):231-253.

APPENDIX A: ADDITIONAL RESULTS

Table A1: Location variable totals for 50 clusters.

	Weight	Milliman	Ward	EEI	EEV	VVV	K-medoids
GMDB Ratchet	1	106.2%	97.7%	93.3%	96.1%	90.3%	**100.4%**
GMDB Rollup	1	105.6%	133.6%	120.1%	95.3%	**98.6%**	121.7%
GMDB ROP	1	77.5%	28.2%	86.8%	99.7%	**100.0%**	86.0%
GMIB Ratchet	1	101.7%	109.5%	100.4%	101.8%	**100.0%**	105.8%
GMIB Rollup	1	100.8%	108.3%	**99.8%**	**100.2%**	100.9%	103.9%
GMAB ROP	1	136.1%	104.1%	105.9%	91.5%	87.6%	103.6%
Separate Acct 1	1	89.2%	**97.8%**	97.0%	92.7%	87.7%	88.2%
Separate Acct 2	1	104.6%	96.0%	**100.7%**	96.4%	101.2%	101.3%
Separate Acct 3	1	107.0%	**102.9%**	104.0%	107.4%	107.4%	104.1%
Separate Acct 4	1	96.8%	90.5%	91.1%	96.1%	**98.8%**	98.3%
Separate Acct 5	1	93.4%	78.0%	96.8%	92.5%	**100.2%**	94.1%
Separate Acct 6	1	110.6%	**105.2%**	110.1%	113.2%	110.7%	109.5%
Separate Acct 7	1	86.8%	**98.2%**	86.1%	91.6%	90.8%	92.4%
General Acct	1	111.4%	113.5%	**100.0%**	107.4%	100.6%	100.1%
Net Revenue 1	4	108.4%	111.6%	107.2%	**100.0%**	100.3%	108.0%
Commissions 1	2	101.7%	102.8%	**99.4%**	103.1%	96.8%	105.8%
Revenue Sharing 1	2	100.9%	103.3%	**100.0%**	102.6%	101.3%	100.5%
Maintenance Expense 1	2	92.5%	111.7%	**99.8%**	100.5%	105.7%	98.9%
M & E Income 1	3	99.2%	97.6%	99.1%	99.4%	96.7%	**100.4%**
Net GMAB Cost 1	3	155.9%	121.6%	117.6%	114.8%	**97.7%**	130.3%
Net GMDB Cost 1	3	93.7%	121.3%	111.8%	93.2%	**96.9%**	126.5%
Net GMIB Cost 1	3	102.8%	105.2%	**100.0%**	95.8%	91.8%	106.7%
Net Revenue 2	4	139.7%	119.8%	118.7%	**100.2%**	94.8%	125.4%
Commissions 2	2	102.0%	95.2%	100.3%	**99.9%**	96.4%	104.4%
Revenue Sharing 2	2	103.0%	103.7%	102.3%	105.1%	105.4%	**101.8%**
Maintenance Expense 2	2	93.7%	110.5%	100.8%	101.4%	106.0%	**99.5%**
M & E Income 2	3	101.2%	97.7%	**100.9%**	101.4%	**100.9%**	101.5%
Net GMAB Cost 2	3	135.9%	111.1%	114.8%	105.2%	**99.0%**	113.9%
Net GMDB Cost 2	3	**98.8%**	123.3%	106.2%	102.1%	92.3%	115.2%
Net GMIB Cost 2	3	98.8%	108.6%	102.3%	100.5%	107.1%	**100.8%**
Net Revenue 3	4	131.0%	127.3%	118.6%	106.0%	**95.6%**	126.8%
Commissions 3	2	102.1%	97.6%	98.9%	**100.3%**	97.1%	104.6%
Revenue Sharing 3	2	103.5%	103.7%	102.2%	105.2%	104.8%	**102.0%**
Maintenance Expense 3	2	93.9%	110.7%	100.8%	101.2%	105.9%	**99.5%**
M & E Income 3	3	101.7%	97.7%	100.8%	101.5%	**100.2%**	101.6%
Net GMAB Cost 3	3	132.3%	113.4%	116.2%	108.1%	99.4%	108.1%
Net GMDB Cost 3	3	96.2%	119.0%	110.1%	95.1%	96.1%	**102.9%**
Net GMIB Cost 3	3	97.7%	105.1%	99.6%	99.1%	107.6%	**99.8%**
Net Revenue 4	4	110.6%	109.7%	106.3%	103.2%	**102.6%**	110.2%
Commissions 4	2	103.2%	102.8%	**100.4%**	102.2%	97.4%	106.1%
Revenue Sharing 4	2	101.9%	103.6%	**100.8%**	103.0%	102.7%	101.0%
Maintenance Expense 4	2	92.6%	111.6%	**100.0%**	100.4%	106.0%	99.0%
M & E Income 4	3	100.3%	98.0%	**99.8%**	99.6%	98.0%	100.8%
Net GMAB Cost 4	3	109.3%	92.4%	**98.0%**	62.1%	55.7%	104.3%
Net GMDB Cost 4	3	86.0%	92.4%	**101.1%**	84.2%	128.1%	136.3%
Net GMIB Cost 4	3	85.4%	122.2%	103.9%	104.1%	89.9%	73.5%
Net Revenue 5	4	110.1%	109.7%	104.9%	**100.0%**	102.3%	107.8%
Commissions 5	2	102.1%	104.0%	**100.1%**	103.5%	98.1%	105.9%
Revenue Sharing 5	2	101.2%	102.7%	**100.6%**	102.9%	101.6%	**100.6%**
Maintenance Expense 5	2	92.6%	111.5%	**100.1%**	100.6%	105.7%	98.7%
M & E Income 5	3	99.3%	96.9%	99.5%	99.4%	96.9%	**100.2%**
Net GMAB Cost 5	3	24.3%	**118.1%**	127.0%	59.3%	46.6%	58.7%
Net GMDB Cost 5	3	**97.7%**	164.4%	82.7%	83.3%	151.0%	159.9%
Net GMIB Cost 5	3	**100.2%**	92.5%	84.1%	91.4%	94.7%	102.8%
WSS		15.48	10.05	**3.56**	3.89	8.03	11.27

Table A2: Location variable totals for 250 clusters

	Weight	Milliman	Ward	EII	EEV	K-medoids
GMDB Ratchet	1	98.9%	100.7%	101.4%	99.8%	101.3%
GMDB Rollup	1	92.4%	104.3%	114.8%	**101.4%**	107.4%
GMDB ROP	1	106.6%	88.7%	93.9%	88.1%	94.3%
GMIB Ratchet	1	100.6%	107.4%	103.8%	**100.3%**	103.2%
GMIB Rollup	1	**100.6%**	105.9%	102.7%	**100.6%**	102.7%
GMAB ROP	1	94.3%	92.9%	99.2%	**99.3%**	103.1%
Separate Acct 1	1	106.1%	90.0%	93.0%	87.4%	**93.4%**
Separate Acct 2	1	99.1%	102.0%	101.4%	99.3%	101.0%
Separate Acct 3	1	95.6%	**103.4%**	104.5%	103.6%	104.9%
Separate Acct 4	1	108.0%	**95.4%**	93.5%	84.4%	88.5%
Separate Acct 5	1	102.6%	96.4%	**98.5%**	98.1%	97.4%
Separate Acct 6	1	90.4%	109.7%	111.6%	109.3%	109.4%
Separate Acct 7	1	111.3%	94.0%	91.2%	**101.6%**	93.9%
General Acct	1	98.8%	99.2%	**100.3%**	102.1%	102.6%
Net Revenue 1	4	78.8%	103.5%	**102.2%**	103.1%	103.6%
Commissions 1	2	**99.1%**	103.6%	105.1%	97.4%	102.0%
Revenue Sharing 1	2	98.9%	99.6%	99.6%	101.3%	**100.0%**
Maintenance Expense 1	2	98.2%	106.5%	**99.5%**	98.8%	102.4%
M & E Income 1	3	98.9%	100.2%	99.7%	98.9%	99.7%
Net GMAB Cost 1	3	90.4%	105.7%	113.8%	116.8%	102.6%
Net GMDB Cost 1	3	**100.4%**	102.1%	101.9%	103.8%	104.3%
Net GMIB Cost 1	3	100.7%	102.4%	**99.8%**	105.7%	102.7%
Net Revenue 2	4	81.4%	113.6%	115.1%	**90.2%**	112.1%
Commissions 2	2	98.7%	103.3%	104.2%	93.8%	**101.0%**
Revenue Sharing 2	2	98.3%	**100.8%**	100.9%	103.8%	101.0%
Maintenance Expense 2	2	98.2%	105.8%	**100.0%**	100.3%	102.3%
M & E Income 2	3	98.5%	101.1%	100.8%	101.2%	**100.6%**
Net GMAB Cost 2	3	89.6%	104.0%	108.4%	**101.1%**	107.9%
Net GMDB Cost 2	3	**99.1%**	113.2%	106.3%	103.4%	102.6%
Net GMIB Cost 2	3	102.1%	100.9%	99.8%	106.2%	101.0%
Net Revenue 3	4	92.8%	115.7%	114.7%	**99.9%**	112.6%
Commissions 3	2	**99.4%**	102.9%	104.3%	95.1%	101.3%
Revenue Sharing 3	2	**99.9%**	100.8%	100.9%	103.2%	101.3%
Maintenance Expense 3	2	98.5%	105.8%	99.9%	**100.0%**	102.3%
M & E Income 3	3	**99.9%**	101.1%	100.7%	100.6%	100.7%
Net GMAB Cost 3	3	84.3%	102.7%	106.1%	**100.2%**	108.9%
Net GMDB Cost 3	3	93.8%	102.1%	105.6%	98.5%	102.6%
Net GMIB Cost 3	3	124.6%	**99.9%**	98.2%	104.1%	98.9%
Net Revenue 4	4	94.4%	104.3%	**103.6%**	104.4%	104.6%
Commissions 4	2	**98.9%**	104.1%	105.2%	96.7%	101.9%
Revenue Sharing 4	2	99.6%	100.3%	**100.0%**	101.3%	100.3%
Maintenance Expense 4	2	98.3%	106.5%	**99.6%**	98.8%	102.4%
M & E Income 4	3	99.6%	100.8%	**100.0%**	98.8%	99.9%
Net GMAB Cost 4	3	121.9%	96.9%	93.5%	92.8%	**99.0%**
Net GMDB Cost 4	3	84.4%	93.9%	94.3%	116.7%	**104.2%**
Net GMIB Cost 4	3	**97.4%**	103.9%	90.7%	75.7%	82.3%
Net Revenue 5	4	95.4%	102.1%	**101.9%**	103.7%	103.8%
Commissions 5	2	**99.6%**	103.9%	105.6%	99.0%	102.4%
Revenue Sharing 5	2	100.2%	**99.9%**	100.2%	101.3%	100.5%
Maintenance Expense 5	2	98.4%	106.2%	**99.7%**	98.8%	102.4%
M & E Income 5	3	100.1%	100.3%	100.1%	98.8%	**100.0%**
Net GMAB Cost 5	3	102.8%	**100.0%**	71.1%	100.4%	102.3%
Net GMDB Cost 5	3	93.5%	**99.1%**	103.8%	150.7%	113.9%
Net GMIB Cost 5	3	**98.5%**	101.7%	96.1%	97.9%	103.8%
WSS		3.19	**1.19**	2.12	3.85	1.29

Table A3: Location variable totals for 1000 clusters.

	Weight	Milliman	Ward	Feedback	EII	EEV	K-Medoids
GMDB Ratchet	1	100.8%	101.8%	**98.9%**	101.6%	98.5%	100.6%
GMDB Rollup	1	105.9%	104.9%	106.3%	106.7%	**101.8%**	105.5%
GMDB ROP	1	96.4%	96.5%	92.8%	94.9%	95.9%	97.3%
GMIB Ratchet	1	100.2%	100.8%	102.6%	102.2%	**100.0%**	101.3%
GMIB Rollup	1	99.6%	**100.3%**	101.9%	101.9%	99.6%	100.8%
GMAB ROP	1	**100.0%**	98.8%	102.8%	98.6%	101.6%	99.5%
Separate Acct 1	1	94.4%	94.2%	95.2%	**96.5%**	92.8%	93.8%
Separate Acct 2	1	101.0%	102.0%	**100.3%**	100.8%	102.6%	99.2%
Separate Acct 3	1	105.2%	106.1%	104.7%	104.5%	**102.7%**	104.4%
Separate Acct 4	1	**95.3%**	95.2%	95.2%	**95.3%**	92.8%	92.8%
Separate Acct 5	1	98.4%	97.7%	98.8%	97.7%	**101.0%**	98.9%
Separate Acct 6	1	107.6%	108.8%	108.5%	108.5%	**106.4%**	107.1%
Separate Acct 7	1	92.8%	92.2%	91.7%	91.9%	**95.2%**	95.0%
General Acct	1	**100.4%**	100.5%	102.3%	100.7%	101.8%	100.7%
Net Revenue 1	4	102.7%	101.3%	102.3%	101.7%	**101.4%**	101.8%
Commissions 1	2	100.9%	102.0%	**100.4%**	101.4%	102.0%	101.2%
Revenue Sharing 1	2	99.9%	99.9%	99.8%	100.2%	100.7%	**100.0%**
Maintenance Expense 1	2	106.2%	99.1%	97.2%	101.9%	**100.5%**	102.6%
M&E Income 1	3	99.8%	99.5%	99.6%	99.7%	99.4%	**99.9%**
Net GMAB Cost 1	3	102.7%	101.6%	102.3%	115.0%	106.0%	107.7%
Net GMDB Cost 1	3	108.3%	100.7%	103.6%	105.1%	**99.4%**	103.6%
Net GMIB Cost 1	3	101.8%	103.7%	103.1%	104.1%	**100.5%**	102.1%
Net Revenue 2	4	111.2%	104.0%	106.3%	105.7%	**98.0%**	106.7%
Commissions 2	2	101.5%	102.4%	101.7%	101.7%	**100.0%**	101.3%
Revenue Sharing 2	2	101.0%	101.0%	101.2%	101.1%	102.4%	**100.8%**
Maintenance Expense 2	2	105.5%	**99.7%**	98.2%	101.9%	101.0%	102.5%
M&E Income 2	3	100.9%	**100.6%**	101.1%	100.7%	101.3%	100.7%
Net GMAB Cost 2	3	101.8%	102.8%	105.2%	107.5%	**101.0%**	102.3%
Net GMDB Cost 2	3	98.1%	101.7%	101.9%	99.8%	100.7%	**100.0%**
Net GMIB Cost 2	3	99.7%	**99.9%**	101.4%	100.4%	102.3%	100.2%
Net Revenue 3	4	111.3%	105.7%	106.8%	105.6%	**103.5%**	107.0%
Commissions 3	2	101.7%	103.0%	101.6%	101.9%	**100.8%**	101.4%
Revenue Sharing 3	2	101.1%	101.4%	101.2%	101.4%	102.0%	**100.9%**
Maintenance Expense 3	2	105.7%	**99.7%**	98.1%	102.0%	100.7%	102.5%
M&E Income 3	3	101.1%	101.0%	101.1%	100.9%	**100.8%**	**100.8%**
Net GMAB Cost 3	3	**101.5%**	102.1%	104.8%	106.9%	**101.5%**	101.8%
Net GMDB Cost 3	3	95.7%	95.6%	101.4%	96.8%	95.0%	98.1%
Net GMIB Cost 3	3	98.8%	98.9%	99.8%	99.7%	**100.4%**	99.1%
Net Revenue 4	4	104.3%	102.0%	102.6%	102.5%	**101.9%**	102.9%
Commissions 4	2	101.0%	102.3%	**100.9%**	101.6%	101.7%	101.2%
Revenue Sharing 4	2	**100.1%**	100.2%	100.2%	100.5%	**101.1%**	100.2%
Maintenance Expense 4	2	106.0%	99.2%	97.4%	101.8%	**100.4%**	102.6%
M&E Income 4	3	100.1%	99.9%	**100.0%**	**100.0%**	99.9%	100.1%
Net GMAB Cost 4	3	104.1%	97.5%	99.0%	**99.7%**	93.8%	101.4%
Net GMDB Cost 4	3	114.2%	**102.3%**	104.8%	107.0%	94.7%	107.9%
Net GMIB Cost 4	3	74.8%	73.5%	**93.8%**	83.7%	91.9%	83.8%
Net Revenue 5	4	103.1%	101.2%	102.0%	**101.0%**	101.3%	101.4%
Commissions 5	2	101.2%	102.5%	**101.0%**	101.9%	102.4%	101.4%
Revenue Sharing 5	2	100.3%	100.3%	100.3%	100.6%	100.7%	100.3%
Maintenance Expense 5	2	105.9%	99.2%	97.4%	101.9%	**100.5%**	102.6%
M&E Income 5	3	100.1%	99.9%	99.9%	100.1%	99.4%	100.1%
Net GMAB Cost 5	3	91.1%	91.2%	107.9%	94.3%	74.2%	93.5%
Net GMDB Cost 5	3	122.5%	**97.7%**	96.9%	105.8%	102.9%	106.3%
Net GMIB Cost 5	3	102.1%	106.6%	101.9%	104.5%	96.6%	**101.8%**
WSS		1.96	0.94	**0.42**	0.87	0.87	0.67

Table A4: Location variable totals for 2500 clusters.

	Weight	Milliman	Ward	EII	Feedback	K-medoids
GMDB Ratchet	1	**100.2%**	102.5%	102.6%	100.6%	100.5%
GMDB Rollup	1	103.7%	**103.1%**	104.6%	104.8%	103.8%
GMDB ROP	1	99.1%	96.4%	97.8%	97.0%	**99.2%**
GMIB Ratchet	1	**100.0%**	101.5%	101.5%	101.2%	100.9%
GMIB Rollup	1	**99.6%**	100.9%	101.2%	101.0%	100.6%
GMAB ROP	1	100.9%	**99.7%**	99.5%	98.0%	100.5%
Separate Acct 1	1	**97.2%**	95.8%	96.2%	96.5%	96.8%
Separate Acct 2	1	100.7%	101.4%	100.1%	99.8%	**100.0%**
Separate Acct 3	1	104.0%	105.1%	103.8%	103.5%	**102.7%**
Separate Acct 4	1	**95.8%**	95.0%	95.4%	94.1%	94.5%
Separate Acct 5	1	98.8%	99.2%	99.1%	**100.0%**	99.0%
Separate Acct 6	1	**105.4%**	108.6%	106.4%	105.9%	105.5%
Separate Acct 7	1	94.5%	91.7%	93.6%	93.7%	**96.2%**
General Acct	1	100.2%	100.4%	**100.1%**	100.7%	100.3%
Net Revenue 1	4	102.2%	**100.5%**	101.1%	101.6%	101.6%
Commissions 1	2	100.5%	100.2%	**100.1%**	100.7%	100.5%
Revenue Sharing 1	2	**100.0%**	**100.0%**	**100.0%**	**100.0%**	100.1%
Maintenance Expense 1	2	111.5%	**98.5%**	105.4%	97.0%	108.3%
M & E Income 1	3	99.8%	99.7%	99.9%	99.7%	**100.0%**
Net GMAB Cost 1	3	104.7%	103.5%	**103.4%**	105.9%	104.5%
Net GMDB Cost 1	3	104.6%	**101.1%**	104.5%	103.6%	102.9%
Net GMIB Cost 1	3	**101.5%**	**101.5%**	101.9%	102.7%	101.7%
Net Revenue 2	4	110.7%	**100.1%**	105.9%	103.9%	107.0%
Commissions 2	2	101.1%	101.0%	101.0%	101.1%	**100.9%**
Revenue Sharing 2	2	100.7%	101.0%	100.7%	100.7%	**100.6%**
Maintenance Expense 2	2	109.9%	**98.8%**	104.8%	97.7%	107.2%
M & E Income 2	3	**100.5%**	100.6%	100.6%	**100.5%**	**100.5%**
Net GMAB Cost 2	3	101.8%	**101.1%**	101.8%	104.4%	101.5%
Net GMDB Cost 2	3	98.3%	101.3%	100.8%	101.1%	100.6%
Net GMIB Cost 2	3	99.7%	100.9%	100.5%	100.5%	**100.2%**
Net Revenue 3	4	109.4%	**100.2%**	105.3%	104.5%	106.8%
Commissions 3	2	101.4%	101.2%	**101.1%**	101.2%	**101.1%**
Revenue Sharing 3	2	100.9%	101.4%	100.9%	100.9%	**100.7%**
Maintenance Expense 3	2	110.1%	**98.9%**	104.9%	97.7%	107.3%
M & E Income 3	3	100.8%	100.9%	100.8%	**100.6%**	**100.6%**
Net GMAB Cost 3	3	101.4%	**100.4%**	101.9%	103.8%	100.7%
Net GMDB Cost 3	3	97.8%	98.5%	98.5%	97.0%	**99.4%**
Net GMIB Cost 3	3	99.2%	100.2%	**100.0%**	99.8%	99.5%
Net Revenue 4	4	103.5%	**100.5%**	101.6%	101.9%	102.4%
Commissions 4	2	100.6%	100.6%	**100.4%**	101.0%	100.6%
Revenue Sharing 4	2	**100.1%**	100.4%	100.2%	100.2%	100.2%
Maintenance Expense 4	2	111.1%	**98.5%**	105.2%	97.1%	108.1%
M & E Income 4	3	99.9%	**100.0%**	100.1%	99.9%	100.1%
Net GMAB Cost 4	3	98.8%	99.3%	92.0%	92.6%	98.2%
Net GMDB Cost 4	3	106.7%	102.2%	107.4%	103.0%	105.9%
Net GMIB Cost 4	3	86.1%	**96.4%**	96.3%	90.6%	91.3%
Net Revenue 5	4	102.5%	**99.8%**	101.0%	100.8%	101.4%
Commissions 5	2	100.8%	100.7%	**100.4%**	101.0%	100.8%
Revenue Sharing 5	2	**100.2%**	100.5%	100.3%	100.3%	100.4%
Maintenance Expense 5	2	111.1%	**98.5%**	105.2%	97.2%	108.0%
M & E Income 5	3	**100.0%**	100.1%	100.1%	**100.0%**	100.2%
Net GMAB Cost 5	3	**96.4%**	90.8%	94.6%	95.7%	95.4%
Net GMDB Cost 5	3	105.2%	95.2%	**103.5%**	110.3%	96.5%
Net GMIB Cost 5	3	101.3%	101.0%	103.1%	101.2%	**100.6%**
WSS		0.92	**0.17**	0.39	0.44	0.47

Table A5: Location variable totals for 5000 clusters.

	Weight	Milliman	Ward	Feedback	K-medoids
GMDB Ratchet	1	**100.2%**	101.7%	100.7%	101.0%
GMDB Rollup	1	102.4%	103.0%	104.3%	102.4%
GMDB ROP	1	**99.5%**	98.3%	97.5%	99.3%
GMIB Ratchet	1	**100.0%**	100.8%	100.9%	100.4%
GMIB Rollup	1	99.6%	100.6%	100.5%	**100.1%**
GMAB ROP	1	100.4%	99.5%	99.5%	**100.2%**
Separate Acct 1	1	98.0%	97.2%	96.7%	98.2%
Separate Acct 2	1	100.2%	**100.1%**	100.2%	**100.1%**
Separate Acct 3	1	103.0%	102.8%	103.1%	**102.2%**
Separate Acct 4	1	**97.5%**	95.4%	95.4%	96.0%
Separate Acct 5	1	98.9%	99.1%	99.4%	98.7%
Separate Acct 6	1	103.9%	106.3%	104.7%	**103.0%**
Separate Acct 7	1	96.4%	94.4%	95.0%	**97.6%**
General Acct	1	**100.2%**	**100.2%**	100.6%	100.3%
Net Revenue 1	4	102.2%	**100.6%**	100.8%	101.4%
Commissions 1	2	100.3%	100.2%	100.5%	100.3%
Revenue Sharing 1	2	100.1%	100.1%	**100.0%**	100.1%
Maintenance Expense 1	2	114.6%	**100.7%**	98.5%	110.8%
M & E Income 1	3	99.8%	99.8%	99.8%	**100.0%**
Net GMAB Cost 1	3	103.8%	**100.2%**	102.1%	102.7%
Net GMDB Cost 1	3	104.2%	**101.0%**	103.2%	101.7%
Net GMIB Cost 1	3	101.6%	102.1%	**101.2%**	101.3%
Net Revenue 2	4	110.4%	102.3%	**101.7%**	107.2%
Commissions 2	2	100.7%	100.8%	100.7%	100.6%
Revenue Sharing 2	2	100.6%	100.7%	**100.5%**	**100.5%**
Maintenance Expense 2	2	112.4%	**100.7%**	98.8%	109.2%
M & E Income 2	3	**100.4%**	100.5%	**100.4%**	**100.4%**
Net GMAB Cost 2	3	101.7%	**100.4%**	103.6%	101.7%
Net GMDB Cost 2	3	98.9%	100.4%	100.9%	**100.1%**
Net GMIB Cost 2	3	99.8%	100.2%	100.3%	**100.1%**
Net Revenue 3	4	109.0%	103.0%	**102.3%**	106.4%
Commissions 3	2	100.9%	101.0%	100.9%	100.7%
Revenue Sharing 3	2	100.8%	100.8%	100.7%	100.6%
Maintenance Expense 3	2	112.7%	**100.7%**	98.8%	109.4%
M & E Income 3	3	100.6%	100.6%	**100.5%**	**100.5%**
Net GMAB Cost 3	3	101.4%	**100.4%**	102.1%	101.4%
Net GMDB Cost 3	3	98.0%	97.6%	97.7%	**99.5%**
Net GMIB Cost 3	3	99.5%	99.8%	99.6%	**99.7%**
Net Revenue 4	4	103.3%	101.0%	**100.9%**	102.2%
Commissions 4	2	100.3%	100.4%	100.6%	100.3%
Revenue Sharing 4	2	100.2%	100.2%	**100.1%**	100.2%
Maintenance Expense 4	2	114.1%	**100.6%**	98.5%	110.4%
M & E Income 4	3	99.9%	**100.0%**	**100.0%**	**100.0%**
Net GMAB Cost 4	3	97.4%	98.8%	97.9%	**99.0%**
Net GMDB Cost 4	3	108.6%	**102.7%**	103.8%	103.2%
Net GMIB Cost 4	3	90.4%	92.0%	93.5%	**94.8%**
Net Revenue 5	4	102.6%	**100.3%**	100.4%	101.6%
Commissions 5	2	**100.4%**	100.6%	100.7%	**100.4%**
Revenue Sharing 5	2	100.3%	100.4%	**100.2%**	100.3%
Maintenance Expense 5	2	114.0%	**100.6%**	98.5%	110.3%
M& E Income 5	3	100.0%	100.1%	**100.0%**	100.1%
Net GMAB Cost 5	3	94.2%	**100.7%**	96.8%	97.3%
Net GMDB Cost 5	3	110.4%	97.0%	112.7%	**97.5%**
Net GMIB Cost 5	3	102.6%	102.4%	100.9%	**100.7%**
WSS	_	1.05	**0.13**	0.28	0.43

Table A6: Kolmogorov-Smirnov p-values for present value of all variables.

Representative Policies	Method	Commission	IMFProf	MaintExp	M&E	GMAB	GMDB	GMIB	Net Revenue	Worst Surplus
110000	Seriatim	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
5000	Milliman	0.536	0.341	<0.001	0.241	0.020	**1.000**	0.288	<0.001	1.000
5000	Ward	0.432	0.573	<0.001	1.000	0.007	0.500	0.685	<0.001	1.000
5000	Feedback	**0.997**	**0.936**	**0.003**	**1.000**	**0.370**	0.648	**0.888**	<0.001	1.000
2500	Milliman	0.400	0.288	0.000	0.241	**0.794**	**0.988**	0.181	<0.001	1.000
2500	Ward	0.794	0.969	0.020	0.980	0.219	0.913	**1.000**	<0.001	1.000
2500	Feedback	0.999	0.648	0.000	**1.000**	0.087	**0.988**	0.794	<0.001	1.000
2500	K-medoids	**1.000**	**1.000**	**0.033**	0.954	0.007	0.685	0.888	<0.001	1.000
1000	Milliman	0.794	0.219	<0.001	0.314	0.000	**0.999**	0.043	<0.001	1.000
1000	Ward	0.828	0.029	<0.001	0.288	0.006	0.913	0.055	<0.001	1.000
1000	Feedback	0.859	**0.859**	**0.341**	0.888	0.001	0.794	**0.759**	0.006	1.000
1000	K-medoids	**0.994**	0.794	0.000	**0.969**	**0.500**	0.536	0.241	0.010	1.000
250	Milliman	0.263	0.008	<0.001	0.013	0.007	0.043	0.001	<0.001	1.000
250	Ward	0.011	**0.500**	**0.794**	**0.980**	<0.001	<0.001	**0.936**	<0.001	1.000
250	EII	**0.536**	0.134	<0.001	0.062	**0.370**	**0.000**	0.314	<0.001	1.000
50	Milliman	<0.001	0.001	**<0.001**	**0.002**	**<0.001**	0.000	0.026	<0.001	1.000
50	EEI	**0.013**	<0.001	<0.001	0.001	<0.001	**0.001**	**0.003**	<0.001	1.000
50	EEV	<0.001	**0.002**	<0.001	<0.001	<0.001	<0.001	<0.001	<0.001	1.000
50	EII	<0.001	<0.001	<0.001	<0.001	<0.001	<0.001	<0.001	<0.001	0.954

Table A7: Kolmogorov-Smirnov test statistics for present value of all variables.

Representative Policies	Method	Commission	IMF Prof	Maint Exp	M&E	GMAB	GMDB	GMIB	Net Revenue	Worst Surplus	Total
110000	Seriatim	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.00
5000	Milliman	0.036	0.042	0.111	0.046	0.068	**0.016**	0.044	**0.116**	**0.005**	0.48
5000	Ward	0.039	0.035	0.105	0.016	0.075	0.037	0.032	0.163	0.007	0.51
5000	**Feedback**	**0.018**	**0.024**	**0.081**	**0.014**	**0.041**	0.033	**0.026**	**0.116**	0.008	**0.36**
2500	Milliman	0.040	0.044	0.115	0.046	**0.029**	**0.020**	0.049	0.133	**0.005**	0.48
2500	**Ward**	0.029	0.022	0.068	0.021	0.047	0.025	**0.015**	0.115	0.008	**0.35**
2500	Feedback	0.017	0.033	0.098	**0.016**	0.056	**0.020**	0.029	0.183	0.007	0.46
2500	K-medoids	**0.016**	**0.013**	**0.064**	0.023	0.075	0.032	0.026	**0.111**	**0.005**	0.37
1000	Milliman	0.029	0.047	0.116	0.043	0.118	**0.017**	0.062	0.289	0.006	0.73
1000	Ward	0.028	0.065	0.188	0.044	0.076	0.025	0.060	0.354	0.009	0.85
1000	**Feedback**	0.027	**0.027**	**0.042**	0.026	0.087	0.029	**0.030**	0.076	0.010	**0.35**
1000	K-medoids	**0.019**	0.029	0.108	**0.022**	**0.037**	0.036	0.046	**0.073**	**0.009**	0.38
250	Milliman	0.045	0.074	0.116	0.071	0.075	0.062	0.085	0.296	0.008	0.83
250	**Ward**	0.072	**0.037**	**0.029**	**0.021**	0.157	0.151	**0.024**	**0.125**	**0.007**	**0.62**
250	EII	**0.036**	0.052	0.132	0.059	**0.041**	**0.095**	0.043	0.267	**0.007**	0.73
50	**Milliman**	0.220	0.086	**0.139**	**0.082**	**0.203**	0.145	0.066	**0.112**	0.010	**1.06**
50	EEI	**0.071**	0.103	0.171	0.088	0.255	**0.087**	**0.080**	0.243	0.015	1.11
50	EEV	0.109	**0.082**	0.169	0.100	0.320	0.202	0.115	0.407	**0.009**	1.51
50	EII	0.229	0.165	0.253	0.184	0.304	0.276	0.118	0.214	0.023	1.77

Table A8: Scaled sum of squared differences between model and seriatim values.

Representative Policies	Method	Commission	IMFProf	MaintExp	M&E	GMAB	GMDB	GMIB	Net Revenue	Worst Surplus	Total
110000	Seriatim	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.00
5000	Milliman	0.060	0.225	0.271	0.177	0.026	**0.010**	0.297	0.067	0.011	1.14
5000	Ward	0.059	0.103	0.224	0.008	**0.021**	0.016	0.085	0.153	**0.052**	0.72
5000	**Feedback**	**0.009**	**0.037**	**0.110**	**0.006**	0.054	0.012	**0.061**	**0.030**	0.115	**0.43**
2500	Milliman	0.081	0.221	0.267	0.172	0.020	0.019	0.391	0.104	**0.013**	1.29
2500	Ward	0.013	0.046	0.135	0.033	0.051	0.015	**0.029**	0.125	0.130	0.58
2500	Feedback	0.010	0.105	0.180	**0.008**	0.054	**0.004**	0.087	0.205	0.098	0.75
2500	**K-medoids**	**0.005**	**0.008**	**0.068**	0.028	**0.033**	0.011	0.063	**0.025**	0.031	**0.27**
1000	Milliman	0.034	0.273	0.208	0.172	0.050	0.012	0.731	0.933	**0.051**	2.46
1000	Ward	0.034	0.791	1.143	0.246	0.033	0.021	0.573	2.036	0.153	5.03
1000	**Feedback**	0.016	**0.045**	**0.015**	0.023	0.083	**0.011**	**0.089**	0.055	0.348	**0.69**
1000	K-medoids	**0.010**	0.077	0.194	**0.016**	**0.022**	0.034	0.277	**0.024**	0.097	0.75
250	Milliman	0.133	0.761	0.193	0.502	**0.057**	**0.151**	1.163	1.204	**0.091**	4.25
250	**Ward**	0.225	**0.155**	**0.044**	**0.032**	0.307	1.029	**0.161**	**0.145**	0.211	**2.31**
250	EII	**0.027**	0.364	0.291	0.198	0.081	0.288	0.428	0.711	0.111	2.50
50	Milliman	3.705	1.144	0.939	**0.711**	**0.320**	1.225	**1.268**	**0.125**	**0.315**	9.72
50	**EEI**	**0.167**	2.189	**0.619**	1.107	2.112	**0.361**	1.684	0.358	0.922	**9.52**
50	EEV	0.547	**0.679**	1.016	0.899	3.790	1.051	1.598	4.081	0.775	14.43
50	EII	2.655	4.072	4.422	4.358	1.767	4.231	4.163	0.258	4.416	30.34