A Data Mining Framework for Valuing Large Portfolios of Variable Annuities

Guojun Gan
Department of Mathematics
University of Connecticut
341 Mansfield Road
Storrs, CT 06269-1009, USA
guojun.gan@uconn.edu

Jimmy Xiangji Huang
School of Information Technology
York University
4700 Keele Street
Toronto, Ontario M3J 1P3, Canada
jhuang@yorku.ca

ABSTRACT

A variable annuity is a tax-deferred retirement vehicle created to address concerns that many people have about outliving their assets. In the past decade, the rapid growth of variable annuities has posed great challenges to insurance companies especially when it comes to valuing the complex guarantees embedded in these products.

In this paper, we propose a novel data mining framework to address the computational issue associated with the valuation of large portfolios of variable annuity contracts. The data mining framework consists of two major components: a data clustering algorithm which is used to select representative variable annuity contracts, and a regression model which is used to predict quantities of interest for the whole portfolio based on the representative contracts. A series of numerical experiments are conducted on a portfolio of synthetic variable annuity contracts to demonstrate the performance of our proposed data mining framework in terms of accuracy and speed. The experimental results show that our proposed framework is able to produce accurate estimates of various quantities of interest and can reduce the runtime significantly.

CCS CONCEPTS

• **Mathematics of computing** → **Nonparametric statistics**; • **Information systems** → **Data mining**;

KEYWORDS

data mining; data clustering; kriging; variable annuity; portfolio valuation

1 INTRODUCTION AND MOTIVATION

A variable annuity is a life insurance product that is created by insurance companies as a tax-deferred retirement vehicle to address concerns many people have about outliving their assets [26, 31]. Under a variable annuity contract, the policyholder (i.e., the individual who purchases the variable annuity product) agrees to make one lump-sum or a series of purchase payments to the insurance company and in turn, the insurance company agrees to make benefit payments to the policyholder beginning immediately or at some future date. A variable annuity has two phases: the accumulation phase and the payout phase. During the accumulation phase, the policyholder builds assets for retirement by investing the money (i.e., the purchase payments) in several mutual funds provided by the insurance companies. During the payout phase, the policyholder receives payments in either a lump-sum, periodic withdrawals or an ongoing income stream.

A main feature of variable annuities is that they contain guarantees, which can be divided into two main classes: guaranteed minimum death benefit (GMDB) and guaranteed minimum living benefit (GMLB). A GMDB guarantees that the beneficiaries receive a guaranteed minimum amount upon the death of the policyholder. There are three types of GMLB: guaranteed minimum accumulation benefit (GMAB), guaranteed minimum income benefit (GMIB), and guaranteed minimum withdrawal benefit (GMWB). A GMAB is similar to a GMDB except that a GMAB is not triggered by the death of the policyholder. A GMAB is typically triggered on policy anniversaries. A GMIB guarantees that the policyholder receives a minimum income stream from a specified future point in time. A GMWB guarantees that a policyholder can withdraw a specified amount for a specified period of time.

The guarantees embedded in variable annuities are financial guarantees that cannot be adequately addressed by traditional pooling methods [4]. If the stock market goes down, for example, the insurance companies lose money on all the variable annuity contracts. Figure 1 shows the stock prices of five top issuers of variable annuities during the period from 2005 to 2016. From the figure we see that the stock prices of all these insurance companies dove during the 2008 financial crisis. Dynamic hedging is adopted by many insurance companies now to mitigate the financial risks associated with the guarantees.

One major challenge of dynamic hedging is that it requires calculating the fair market values of the guarantees for a large portfolio of variable annuity contracts in a timely manner [8]. Since the guarantees are relatively complex, their fair market values cannot be calculated in closed form except for special cases [13, 21]. In practice, insurance companies rely on Monte Carlo simulation to calculate the fair market values of the guarantees. However, using Monte Carlo simulation to value a large portfolio of variable annuity contracts is extremely time-consuming because every contract needs to be projected over many scenarios for a long time horizon.

KDD'17, August 13–17, 2017, Halifax, NS, Canada.

ACM ISBN 978-1-4503-4887-4/17/08…$15.00
https://doi.org/10.1145/3097983.3098013

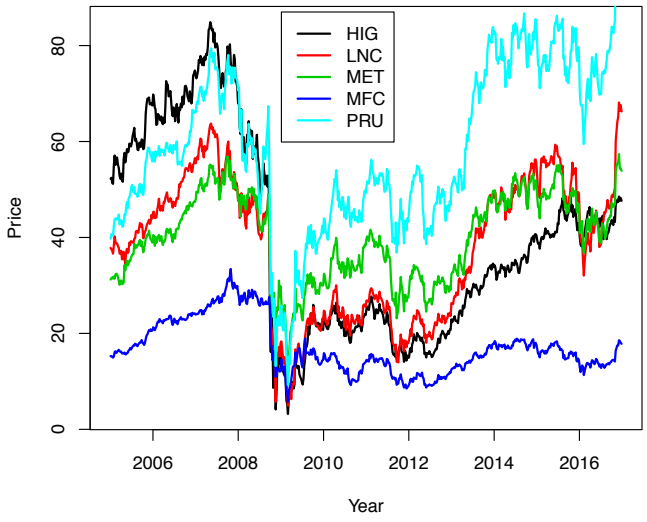

Figure 1: The stock prices of five insurance companies from 2005 to 2016. These insurance companies are top issuers of variable annuities.

In this paper, we propose a data mining framework to address the aforementioned computational issue arising from the insurance industry. The data mining framework consists of two main components: a clustering algorithm for experimental design and a regression model for prediction. The idea is to use a clustering algorithm to select a small number of representative contracts and build a regression model based on these representative contracts to predict the fair market values of all the contract in the portfolio. The data mining framework is able to reduce the valuation time significantly because only a small number of representative contracts are valued by the Monte Carlo simulation method and the whole portfolio of contracts are valued by the regression model, which is much faster than the Monte Carlo simulation method. The details of the framework are presented in Section 3.

The major contributions of this paper are summarized as follows:

- We develop a new framework based on data mining techniques for valuating large portfolios of variable annuity contracts by intergrating a newly proposed data clustering algorithm for experimental design and a new Gaussian process regression model for prediction.
- We show empirically that the data mining framework is able to speed up significantly the valuation of large portfolios of variable annuity contracts and produce accurate estimates.
- In the experimental design step, we propose a new TFCM++ algorithm, which is very efficient and more robust in dividing a large dataset into many clusters, to select representative variable annuity contracts.

2 LITERATURE REVIEW

In this section, we give a brief review of existing methods used to address the computational issue associated valuing the variable annuity guarantees.

Existing methods can be divided into two groups: hardware methods and software methods. Hardware methods try to speed up the computation from the perspective of hardware. For example, GPUs (Graphics Processing Unit) have been used to value variable annuity contracts [27, 29]. One drawback of hardware methods is that they are not scalable. In other words, if the number of variable annuity contracts doubles, then the insurance company needs to double the number of computers or GPUs in order to complete the calculation within the same time interval. Another drawback of hardware methods is that they are expensive. Buying or renting many computers or GPUs can cost the insurance company a lot of money every year.

Software methods try to speed up the computation from the perspective of software by developing efficient algorithms and mathematical models. One type of software methods involves constructing replicating portfolios by using standard financial instruments such as futures, European options, and swaps [9, 11, 28]. Under this type of software methods, the replicating portfolio is constructed to match the cash flows of the variable annuity guarantees. Then the portfolio of variable annuity contracts is replaced by the replicating portfolio and closed-form formulas are employed to calculate quantities of interest. However, constructing a replicating portfolio of a large portfolio of variable annuities is time-consuming because the cash flows of the portfolio at each time step and each scenario must be projected by an actuarial projection system.

Another type of software methods involves reducing the number of variable annuity contracts that go through Monte Carlo simulation. Vadiveloo [32] proposed a method based on replicated stratified sampling and only these sample policies are valuated. Gan [14] used the k-prototypes algorithm to select a small set of representative variable annuity contracts and used the ordinary kriging method [24] to predict the fair market values based on those of the representative contracts. Since the k-prototypes algorithm is extremely slow when used to divide a large dataset into many clusters, the portfolio of variable annuity contracts was split into many subsets and the k-prototypes algorithm was applied to these subsets.

To address the inefficiency of the k-prototypes algorithm in dividing a large portfolio of variable annuity contracts into many clusters, Gan [15] proposed to use the Latin hypercube sampling method to select representative contracts. Since the fair market values of the guarantees embedded in variable annuities are skewed and have fat-tails, Gan and Valdez [19] proposed to use the GB2 (Generalized Beta of the Second Kind) regression model to capture the skewness. In [19], the conditional Latin hypercube sampling was used to select representative variable annuity contracts. However, it is a great challenge to estimate the parameters of the GB2 regression model.

3 A DATA MINING FRAMEWORK

Data mining refers to a computational process of exploring and analyzing large amounts of data in order to discover useful information [1, 6, 7, 10]. There are four main types of data mining tasks: association rule learning, clustering, classification, and regression. There are two types of data: labelled and unlabelled. Labelled data has a specially designated attribute and the aim is to use the given

data to predict the value of that attribute for new data. Unlabelled data does not have such a designated attribute. The first two data mining tasks, association rule learning and clustering, work with unlabelled data and are known as unsupervised learning [23]. The last two data mining tasks, classification and regression, work with labelled data and are called supervised learning [22].

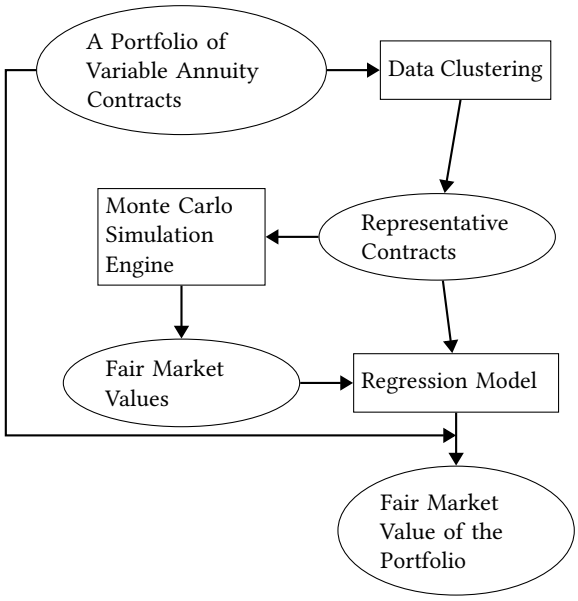

Figure 2: A data mining framework for estimating the fair market values of guarantees embedded in variable annuities.

Figure 2 shows the data mining framework proposed to speed up the calculation of the fair market values of guarantees for a large portfolio of variable annuity contracts. The data mining framework consists of four major steps:

(1) Use a data clustering algorithm to divide the portfolio of variable annuity contracts into clusters in order to find representative contracts. The clustering algorithm should produce spherical shaped clusters. In each cluster, the contract that is closest to the cluster center is selected as a representative contract.
(2) Run the Monte Carlo simulation engine to calculate the fair market values (or other quantities of interest) of the guarantees for the representative contracts.
(3) Create a regression model by using contract features as explanatory variables and the fair market value (or other quantities of interest) as response variable.
(4) Use the regression model to predict the fair market values (or other quantities of interest) of the guarantees for all contracts in the portfolio.

The Monte Carlo simulation engine is not part of the framework but is used to produce the fair market values of guarantees for the representative contracts. In fact, the data mining framework treats the Monte Carlo simulation engine as a black box and creates a regression model to replace it. Since the regression model is much faster than the Monte Carlo simulation engine, using the regression model to estimate the fair market values for the whole portfolio has the potential to reduce the runtime significantly.

In this section, we introduce the clustering algorithm and the regression model used in the data mining framework in detail. The Monte Carlo simulation engine is specific to particular variable annuity products and will not be discussed here. Interested readers are referred to [16] for a simple example of Monte Carlo simulation engines.

3.1 The TFCM++ Algorithm

Typically, the portfolio contains hundreds of thousands of contracts and we need many (e.g., 100 to 500) representative contracts in order to build a regression model that can produce accurate estimate of the fair market value of the portfolio. Since we select only one contract from each cluster as representative contract, we need to divide the portfolio into many clusters. However, most existing clustering algorithms do not scale to divide a large dataset into many clusters [5].

The literature on optimizing clustering algorithm running time for dividing a large dataset into many clusters is scarce. Relevant work includes the WAND-k-means algorithm [5] and the TFCM (Truncated Fuzzy c-means) algorithm [17]. The WAND-k-means algorithm was proposed by Broder et al. [5] to divide efficiently millions of webpages into thousands of categories. In each iteration, the WAND-k-means utilizes a "centers picking points" approach instead of the "points picking centers" approach normally used by k-means. Since webpages are documents, an inverted index over all the points (i.e., webpages) is created before clustering. During the clustering process, the current centers are used as queries to this index to decide on cluster membership. The TFCM algorithm is a variant of the fuzzy c-means (FCM) algorithm [3, 12] proposed by Gan et al. [17] to divide a large dataset into many clusters.

The WAND-k-means requires an inverted index and thus cannot be applied to select representative contracts, which are not documents. The TFCM algorithm is sensitive to initial cluster centers and we need to run the TFCM algorithm multiple times in order to select the best clustering result. In this section, we present a modified version of the TFCM algorithm, called the TFCM++ algorithm, to select representative variable annuity contracts. The TFCM++ algorithm uses the method of the k-means++ algorithm [2] to initialize cluster centers. Since the TFCM++ algorithm is more robust than the TFCM algorithm, we only need to run the TFCM++ algorithm once to select representative variable annuity contracts.

We first describe the TFCM algorithm. Let $X = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n\}$ be a dataset containing n points. Let k be the desired number of clusters. Let T be an integer such that $1 \le T \le k$ and let $\mathcal{U}_T$ be the set of fuzzy partition matrices U such that each row of U has at most T nonzero entries, that is, $U \in \mathcal{U}_T$ if U satisfies the following conditions

$$u_{il} \in [0,1], \quad i = 1, 2, \ldots, n, \; l = 1, 2, \ldots, k, \tag{1a}$$

$$\sum_{l=1}^{k} u_{il} = 1, \quad i = 1, 2, \ldots, n, \tag{1b}$$

$$|\{l : u_{il} > 0\}| \le T, \quad i = 1, 2, \ldots, n, \tag{1c}$$

where $|\cdot|$ denote the number of elements in a set.

The TFCM algorithm aims at finding a truncated fuzzy partition matrix U and a set of cluster centers Z to minimize the following objective function:

$$P(U,Z)=\sum_{i=1}^{n}\sum_{l=1}^{k}u_{il}^{\alpha}\left(\|\mathbf{x}_i-\mathbf{z}_l\|^2+\epsilon\right), \tag{2}$$

where $\alpha>1$ is the fuzzifier, $U\in\mathcal{U}_T$, $Z=\{\mathbf{z}_1,\mathbf{z}_2,\ldots,\mathbf{z}_k\}$ is a set of cluster centers, $\|\cdot\|$ is the L^2-norm or Euclidean distance, and ϵ is a small positive number used to prevent division by zero.

Similar to the original FCM algorithm, the TFCM algorithm uses an alternative updating scheme in order to minimize the objective function. Theorem 3.1 and Theorem 3.2 describe how to update the fuzzy membership U given the cluster centers Z and how to update the cluster centers Z given the fuzzy membership U, respectively.

Theorem 3.1. *For a fixed set of centers Z, the fuzzy partition matrix $U\in\mathcal{U}_T$ that minimizes the objective function* (2) *is given by*

$$u_{il}=\frac{\left(\|\mathbf{x}_i-\mathbf{z}_l\|^2+\epsilon\right)^{-\frac{1}{\alpha-1}}}{\sum_{s\in I_i}\left(\|\mathbf{x}_i-\mathbf{z}_s\|^2+\epsilon\right)^{-\frac{1}{\alpha-1}}},\quad 1\le i\le n,\ l\in I_i, \tag{3}$$

where I_i is the set of indices of the T centers that are closest to $\mathbf{x}_i$.

Theorem 3.2. *For a fixed fuzzy partition matrix $U\in\mathcal{U}_T$, the set of centers Z that minimizes the objective function* (2) *is given by*

$$z_{lj}=\frac{\sum_{i=1}^{n}u_{il}^{\alpha}x_{ij}}{\sum_{i=1}^{n}u_{il}^{\alpha}}=\frac{\sum_{i\in C_l}u_{il}^{\alpha}x_{ij}}{\sum_{i\in C_l}u_{il}^{\alpha}}, \tag{4}$$

for $l=1,2,\ldots,k$ and $j=1,2,\ldots,d$, where d is the dimension of the dataset, z_{lj} is the jth component of $\mathbf{z}_l$, and $C_l=\{i:u_{il}>0\}$.

The TFCM algorithm uses random sampling to initialize cluster centers. In the TFCM++ algorithm, we use the method of the k-means++ algorithm [2] to select initial cluster centers. In the method of the k-means++ algorithm, cluster centers are initialized with probabilities that are dependent on the shortest distances between centers already selected and points not yet selected. The pseudo-code of the TFCM++ algorithm is shown in Algorithm 1.

The TFCM++ algorithm requires several parameters: k, T, α, δ, and N_{max}. The parameter k specifies the desired number of clusters and corresponds to the number of representative variable annuity contracts. The parameter T specifies the number of clusters to which a data point may belong. Selecting a value for the parameter T is a trade-off between runtime and accuracy. When a larger value is used for T, the clustering result will be closer to that of the original FCM algorithm. However, a larger value for T makes the algorithm slower. A good start point to select a value for the parameter T is to use $T=d+1$, where d is the dimensionality of the underlying dataset. In a d-dimensional dataset, a simplex has $d+1$ vertices and a points might be equidistant from the centers of $d+1$ sphere-shaped clusters. The parameter α is called the fuzzifier and takes values in $(1,\infty)$. The last two parameters δ and N_{max} are used to terminate the algorithm. Default values of these parameters are given in Table 1.

The time complexity of the proposed TFCM++ algorithm is $O((n-\frac{k+1}{2})k+nT^2)$. This is because (1) the time complexity of initialization is $O((n-\frac{k+1}{2})k)$ and (2) it takes the TFCM++ algorithm $O(nT^2)$ floating point operations to update the fuzzy partition matrix U [25].

Algorithm 1: Pseudo-code of the TFCM++ algorithm.

Input: $X=\{\mathbf{x}_1,\mathbf{x}_2,\ldots,\mathbf{x}_n\}$, k, T, δ, N_{max}, α
Output: U, Z

Select an initial center $\mathbf{z}_1$ uniformly at random from X and let $Z=\{\mathbf{z}_1\}$;
for $l=2$ *to* k **do**
  Calculate the distances between $\mathbf{z}_{l-1}$ and points in $X\backslash Z$;
  Let I_{l-1} be the indices of the T points in $X\backslash Z$ that are closest to $\mathbf{z}_{l-1}$;
  Select an initial center $\mathbf{z}_l=\mathbf{x}'$ from X with probability $\frac{D(\mathbf{x}')^2}{\sum_{\mathbf{x}\in X}D(\mathbf{x})^2}$, where $D(\mathbf{x})$ denotes the shortest distance between $\mathbf{x}$ and the selected centers;
  $Z\leftarrow Z\cup\{\mathbf{z}_l\}$;
end
Calculate the distances between $\mathbf{z}_k$ and points in $X\backslash Z$;
Let I_k be the indices of the T points in $X\backslash Z$ that are closest to $\mathbf{z}_{l-1}$;
$s\leftarrow 0$, $P\leftarrow 0$;
while *True* **do**
  for $i=1$ *to* n **do**
    Select T indices J_i in $\{1,2,\ldots,k\}/I_i$ randomly;
    Calculate the distance between $\mathbf{x}_i$ and centers with indices in $I_i\cup J_i$;
    Update I_i with the indices of the T centers that are closest to $\mathbf{x}_i$;
    Update the weights u_{il} for $l\in I_i$ according to Equation (3);
  end
  Update the set of cluster centers Z according to Equation (4);
  $P^*\leftarrow P$, $P\leftarrow P(U,Z)$, $s\leftarrow s+1$;
  if $\frac{|P-P^*|}{P^*}<\delta$ *or* $s\ge N_{max}$ **then**
    Break;
  end
end
Return U and Z;

Parameter	Default Value	Parameter	Default Value
T	d + 1	δ	10^{-3}
α	2	N_{max}	100

Table 1: Default values of some parameters required by the TFCM++ algorithm. Here d is the dimensionality of the underlying dataset.

3.2 The Orindary Kriging Method

A regression model is another important component of the data mining framework. We use the ordinary kriging method [24] to predict the fair market values of the guarantees and other quantities

of interest such as deltas and rhos. The ordinary kriging method is also known as a Gaussian process regression model [30]. In this section, we give a brief description of the ordinary kriging method.

Let $X = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n\}$ be a portfolio of n variable annuity contracts. and let $\mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_k$ be the representative contracts obtained from the clustering process. For every $j = 1, 2, \ldots, k$, let y_j be some quantity of interest of $\mathbf{z}_j$ that is calculated by the Monte Carlo simulation method. Quantities of interest include fair market values, deltas, and rhos, where deltas refer to the sensitivities of the fair market values to the underlying equity prices and rhos refer to the sensitivities of the fair market values to the interest rates. Under the ordinary kriging method, the quantity of interest of the variable annuity contract $\mathbf{x}_i$ as

$$\hat{y}_i = \sum_{j=1}^{k} w_{ij} \cdot y_j, \tag{5}$$

where $w_{i1}, w_{i2}, \ldots, w_{ik}$ are the kriging weights.

The kriging weights $w_{i1}, w_{i2}, \ldots, w_{ik}$ are obtained by solving the following linear equation system

$$\begin{pmatrix} V_{11} & \cdots & V_{1k} & 1 \\ \vdots & \ddots & \vdots & \vdots \\ V_{k1} & \cdots & V_{kk} & 1 \\ 1 & \cdots & 1 & 0 \end{pmatrix} \cdot \begin{pmatrix} w_{i1} \\ \vdots \\ w_{ik} \\ \theta_i \end{pmatrix} = \begin{pmatrix} D_{i1} \\ \vdots \\ D_{ik} \\ 1 \end{pmatrix}, \tag{6}$$

where θ_i is a control variable used to make sure the sum of the kriging weights is equal to one,

$$V_{rs} = \alpha + \exp\left(-\frac{3}{\beta} D(\mathbf{z}_r, \mathbf{z}_s)\right), \quad r, s = 1, 2, \ldots, k, \tag{7}$$

and

$$D_{ij} = \alpha + \exp\left(-\frac{3}{\beta} D(\mathbf{x}_i, \mathbf{z}_j)\right), \quad j = 1, 2, \ldots, k. \tag{8}$$

Here $D(\cdot, \cdot)$ is the Euclidean distance function. Before calculating the distances between variable annuity contracts, we convert all categorical variables (e.g., gender and product type) into dummy binary variables and use the Min-Max normalization method to scale all variables to the interval $[0, 1]$.

In Equations (7) and (8), $\alpha \geq 0$ and $\beta > 0$ are two parameters. In practice, we can set $\alpha = 0$ and set β to be the 95th percentile of all the distances between pairs of the k representative variable annuity contracts [24].

Since $D(\mathbf{z}_r, \mathbf{z}_s) > 0$ for all $1 \leq r < s \leq k$, the linear equation system given in Equation (6) has a unique solution [24]. Solving many linear equation systems to calculate the individual estimates $\hat{y}_1, \hat{y}_2, \ldots, \hat{y}_n$ is time-consuming. However, we can avoid this by observing that the matrix in the left hand side of Equation (6) is independent of i. In fact, we can calculate the following vector once:

$$M = (y_1, y_2, \ldots, y_k, 0) \cdot \begin{pmatrix} V_{11} & \cdots & V_{1k} & 1 \\ \vdots & \ddots & \vdots & \vdots \\ V_{k1} & \cdots & V_{kk} & 1 \\ 1 & \cdots & 1 & 0 \end{pmatrix}^{-1}. \tag{9}$$

Then we can calculate $\hat{y}_i$ as follows:

$$\hat{y}_i = (y_1, y_2, \ldots, y_k, 0) \cdot \begin{pmatrix} w_{i1} \\ w_{i2} \\ \vdots \\ w_{ik} \\ \theta_i \end{pmatrix} = M \cdot \begin{pmatrix} D_{i1} \\ \vdots \\ D_{ik} \\ 1 \end{pmatrix}. \tag{10}$$

In this way, we do not need to solve a linear equation system for calculating an individual $\hat{y}_i$. Instead, we only need to calculate the inner product of two vectors, thus making significant efficiency gain.

4 EMPIRICAL EVALUATION

In this section, we evaluate the data mining framework experimentally by using a synthetic portfolio of variable annuity contracts.

4.1 A Synthetic Portfolio

To evaluate the performance of the data mining framework, we create a portfolio of synthetic variable annuity contracts based on the following properties of portfolios of real variable annuity contracts:

- A portfolio of real variable annuity contracts contains different type of variable annuity products.
- A real variable annuity contract allows the contract holder to invest the money in multiple funds.
- Real variable annuity contracts are issued at different dates and have different time to maturity.

The portfolio contains 10,000 synthetic variable annuity contracts, each of which is described by 18 features including two categorical features. A description of the features can be found in [18], [19], and [20]. Figure 3 shows a histogram of the fair market values, deltas, and rhos of the guarantees embedded in the 10,000 synthetic variable annuity contracts. From the histogram, we see that the distribution of the fair market values is skewed to the right. Deltas measure the sensitivities of the fair market values of the guarantees to the underlying stock prices. Most of the deltas are negative because the guarantees are similar to put options, which have negative deltas. Rhos measure the sensitivities of the fair market values of the guarantees to the level of interest rates. Most of the rhos are also negative because when interest rates go up, the fair market values of the guarantees go down. These quantities are calculated by a simple Monte Carlo simulation method [16]. It took the Monte Carlo simulation method 72,234.12 seconds to calculate these quantities for all 10,000 variable annuity contracts. In the simple Monte Carlo simulation, we used 5,000 scenarios with monthly steps to project cash flows for 40 years.

4.2 Validation Measures

To assess the accuracy of the data mining framework, we use the following two validation measures: the percentage error at the portfolio level and the R^2.

For $i = 1, 2, \ldots, n$, let y_i and $\widehat{y_i}$ be the fair market value of the ith variable annuity contract obtained from the Monte Carlo simulation model and that estimated by the ordinary kriging method, respectively. Then the percentage error at the portfolio level is

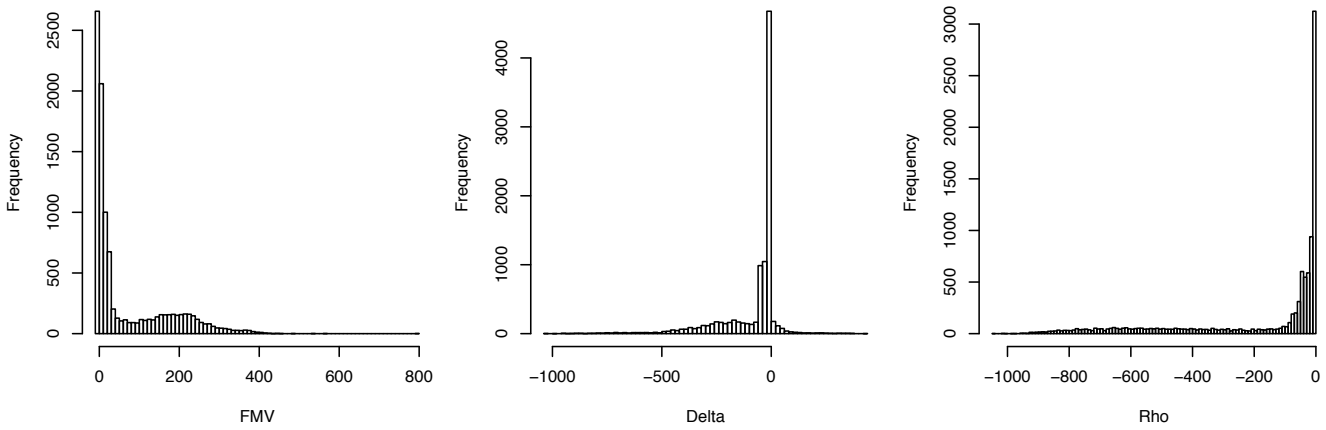

Figure 3: A histogram of the fair market values, deltas, and rhos of the guarantees embedded in variable annuity contracts.

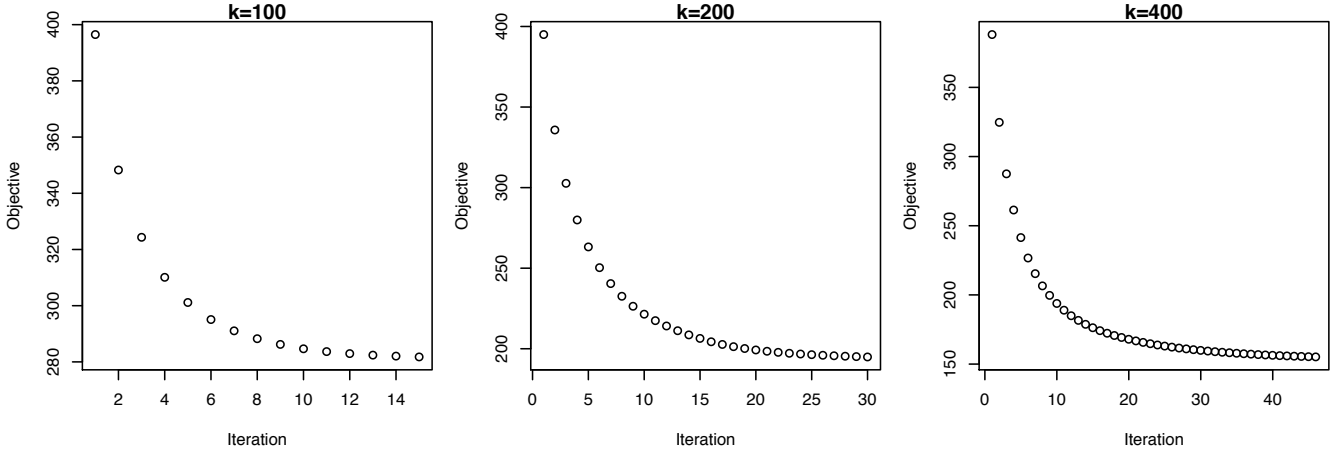

Figure 4: Convergence of the objective function of the TFCM++ algorithm.

defined as

$$PE = \frac{\sum_{i=1}^{n}(\widehat{y}_i - y_i)}{\sum_{i=1}^{n} y_i}. \tag{11}$$

The R^2 is defined as

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(\widehat{y}_i - y_i)^2}{\sum_{i=1}^{n}(y_i - \mu)^2}, \tag{12}$$

where μ is the average fair market value, i.e.,

$$\mu = \frac{1}{n}\sum_{i=1}^{n} y_i.$$

The percentage error at the portfolio level measures the aggregate accuracy of the result because the errors at the individual contract level can offset each other. If the absolute value of PE is closer to zero, then the result is more accurate. The R^2 measures the accuracy of the result without offsetting the errors at the individual contract level. The higher the R^2, the more accurate the result.

4.3 Experimental Results

We test the performance of the data mining framework with $k = 100$, $k = 200$ and $k = 400$ clusters. In our tests, we use the default values for other parameters of the TFCM++ algorithm (see Table 1). Since the dataset has 21 dimensions, we used $T = 22$ as suggested in Section 3.1. Figure 4 shows the objective function values of the TFCM++ algorithm at each iteration. From this figure, we can see that the TFCM++ algorithm converges pretty fast. When $k = 100$ is used, the TFCM++ algorithm converges in 14 iterations. When $k = 400$ is used, it converges in 46 iterations. When k is larger and T is the same, it takes the TFCM++ algorithm more iterations to converge.

Table 2 shows the validation measures used to assess the accuracy of the data mining framework. From this table, we see that in general, the accuracy increases when the number of clusters increases. For example, the absolute value of the percentage error for the fair market value decreases from 6.22% to 4.78% when k increases from 100 to 400. The R^2 always increases when k increases, indicating that the larger the k, the better the fit.

	FMV	Delta	Rho
PE	-6.22%	-4.12%	-0.91%
R^2	0.6794	0.6160	0.8154

(a) $k = 100$.

	FMV	Delta	Rho
PE	-5.54%	-3.52%	-3.36%
R^2	0.7553	0.6824	0.8792

(b) $k = 200$.

	FMV	Delta	Rho
PE	4.78%	-0.35%	2.77%
R^2	0.8231	0.7918	0.9057

(c) $k = 400$.

Table 2: Accuracy of the data mining framework. Here FMV denotes fair market value.

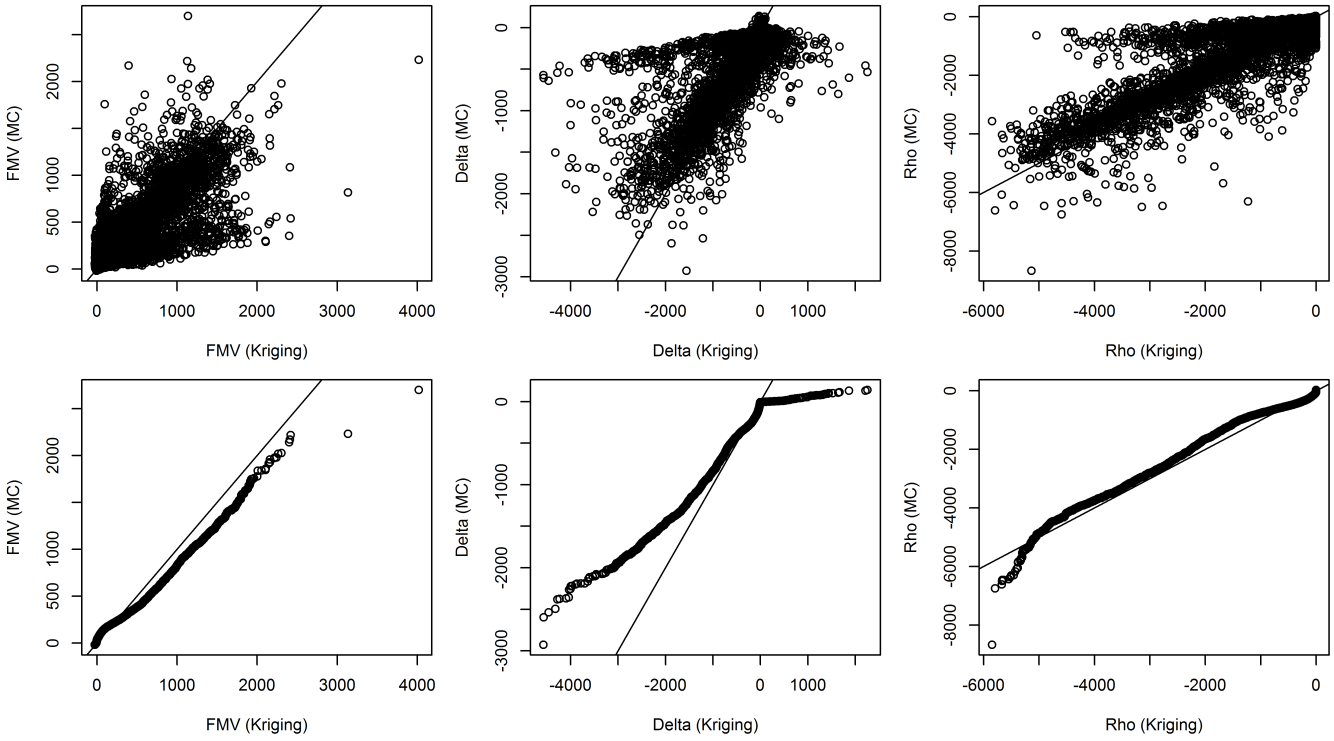

Figure 5: Scatter plots and QQ plots of the quantities calculated by Monte Carlo and those obtained by the data mining framework when 100 clusters are used.

Figures 5, 6, and 7 show the scatter plots and QQ (Quantile-Quantile) plots of the quantities calculated by Monte Carlo simulation and those estimated by the data mining framework. The scatter plots show that the ordinary kriging method does not produce very accurate estimates at the individual contract level. The QQ plots show that the ordinary kriging method does not fit the tails well, especially for the fair market values and the deltas. The reason is the the ordinary kriging method assumes that the response variable follows a normal distribution. From the histograms in Figure 3, we can see that the fair market values, deltas, and rhos are not normally distributed. Hence it is expected that the ordinary kriging method will not produce accurate estimates at the individual contract level or good fit of tails.

However, the ordinary kriging method is able to produce accurate estimates at the portfolio level as shown in Table 2. The errors of individual contracts offset each other. In practice, the goal is to produce accurate estimates at the portfolio level because risk management is done for the whole portfolio rather than individual contracts.

Table 3 shows the runtime of the three major steps of the data mining framework. We can see from the table that the runtime is dominated by the Monte Carlo simulation engine. It took the Monte Carlo simulation engine 72,234.12 seconds or 20 hours to compute the fair market values, deltas, and rhos for the whole portfolio, which contains 10,000 variable annuity contracts. When $k = 100$ was used, it took the data mining framework 780.61 seconds or 13 minutes to estimate those quantities for the whole portfolio. It took the TFCM++ algorithm 55.20 seconds to divide the portfolio into 100 clusters. The ordinary kriging method was pretty fast. The efficiency gain of the data mining framework is significant.

In summary, the experiments show that the data mining framework is able to produce accurate estimates of various quantities of interest and can save significant runtime.

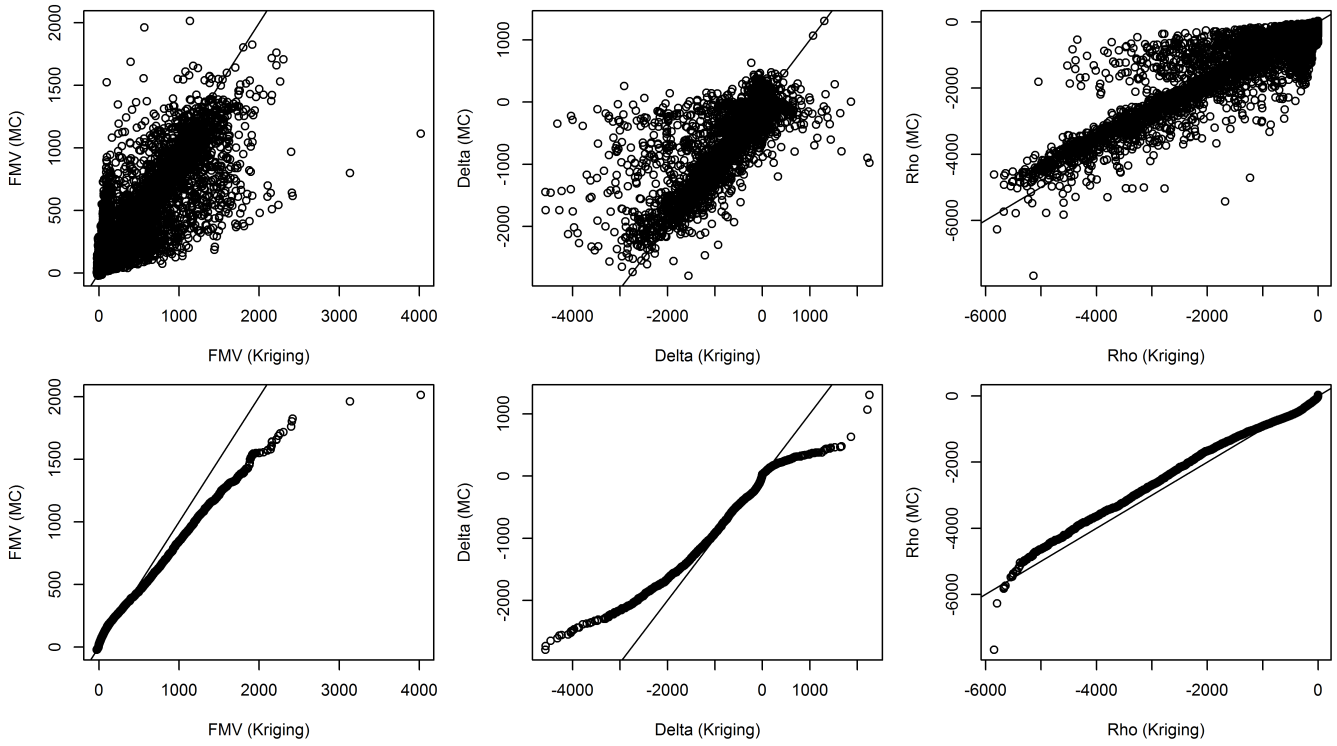

Figure 6: Scatter plots and QQ plots of the quantities calculated by Monte Carlo and those obtained by the data mining framework when 100 clusters are used.

	Data Mining			Portfolio
	100	200	400	10000
TFCM++	55.20	129.98	235.60	-
Monte Carlo	722.34	1,444.68	2,889.36	72,234.12
Kriging	3.07	7.30	14.79	-
Total	780.61	1,581.96	3,139.75	72,234.12

Table 3: Runtime of major steps of the data mining framework.

5 CONCLUSIONS AND FUTURE WORK

In this paper, we propose a novel data mining framework to address the computational issue associated with the valuation of large portfolios of variable annuity contracts. The proposed data mining framework consists of two major components: a data clustering algorithm and a regression model. The data clustering algorithm is used to select representative variable annuity contracts from the portfolio and the regression model is used to predict quantities of interest for the whole portfolio based on the representative contracts. Since only a small number of representative contracts are valued by the Monte Carlo simulation engine, the data mining framework is able to make significant gain in efficiency.

Our numerical experiments on a portfolio of synthetic variable annuity contracts show that the data mining framework is able to produce accurate estimates of various quantities of interest and can also reduce the runtime significantly. This data mining framework has the potential to help insurance companies that have a variable annuity business to make risk management decisions on a timely basis and save money on computer hardware. In future, we would like to investigate more efficient clustering algorithms to divide a large dataset into many clusters.

6 ACKNOWLEDGMENTS

This work is supported by a CAE (Centers of Actuarial Excellence) grant [1] from the Society of Actuaries. This research is also supported by the Natural Sciences and Engineering Research Council (NSERC) of Canada, an NSERC CREATE award in ADERSIM [2], the York Research Chairs (YRC) program and an ORF-RE (Ontario Research Fund-Research Excellence) award in BRAIN Alliance [3].

REFERENCES

[1] C. C. Aggarwal. *Data Mining: The Textbook.* Springer, New York, NY, 2015.
[2] D. Arthur and S. Vassilvitskii. *k*-means++: The advantages of careful seeding. In *Proceedings of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '07, pages 1027–1035, Philadelphia, PA, USA, 2007. Society for Industrial and Applied Mathematics.
[3] J. Bezdek. *Pattern Recognition with Fuzzy Objective Function Algorithms.* Kluwer Academic Publishers, Norwell, MA, USA, 1981.
[4] P. Boyle and M. Hardy. Reserving for maturity guarantees: Two approaches. *Insurance: Mathematics and Economics*, 21(2):113–127, 1997.

[1] http://actscidm.math.uconn.edu
[2] http://www.yorku.ca/adersim
[3] http://brainalliance.ca

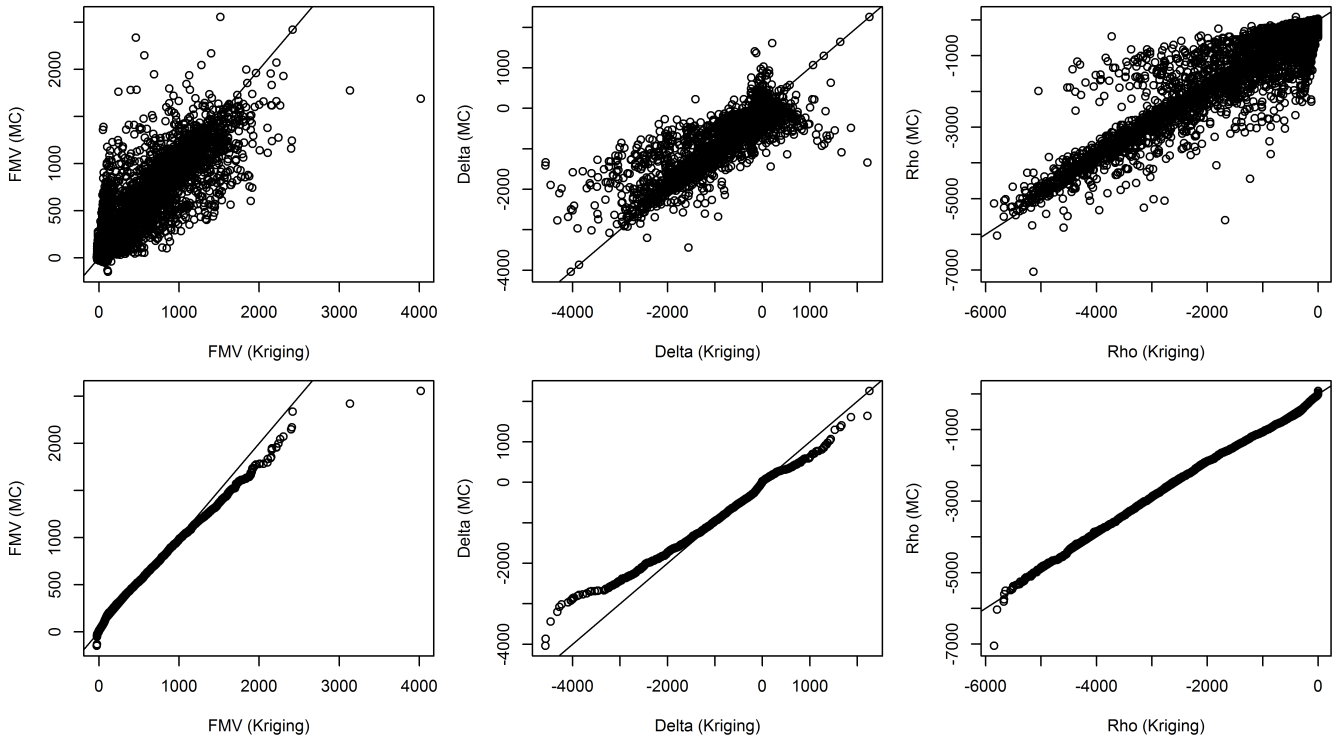

Figure 7: Scatter plots and QQ plots of the quantities calculated by Monte Carlo and those obtained by the data mining framework when 100 clusters are used.

[5] A. Broder, L. Garcia-Pueyo, V. Josifovski, S. Vassilvitskii, and S. Venkatesan. Scalable k-means by ranked retrieval. In *Proceedings of the 7th ACM International Conference on Web Search and Data Mining*, WSDM '14, pages 233–242. ACM, 2014.

[6] M. Chen, S. Mao, Y. Zhang, and V. C. Leung. *Big Data: Related Technologies, Challenges and Future Prospects*. Springer, New York, NY, 2014.

[7] P. Cichosz. *Data Mining Algorithms: Explained Using R*. Wiley, Hoboken, NJ, 2015.

[8] T. Dardis. Model efficiency in the U.S. life insurance industry. *The Modeling Platform*, (3):9–16, 2016.

[9] S. Daul and E. G. Vidal. Replication of insurance liabilities. *RiskMetrics Journal*, 9(1), 2009.

[10] J. Dean. *Big Data, Data Mining, and Machine Learning: Value Creation for Business Leaders and Practitioners*. Wiley, Hoboken, NJ, 2014.

[11] R. Dembo and D. Rosen. The practice of portfolio replication: A practical overview of forward and inverse problems. *Annals of Operations Research*, 85:267–284, 1999.

[12] J. C. Dunn. A fuzzy relative of the ISODATA process and its use in detecting compact well-separated clusters. *Journal of Cybernetics*, 3(3):32–57, 1973.

[13] R. Feng and H. Volkmer. Analytical calculation of risk measures for variable annuity guaranteed benefits. *Insurance: Mathematics and Economics*, 51(3):636–648, 2012.

[14] G. Gan. Application of data clustering and machine learning in variable annuity valuation. *Insurance: Mathematics and Economics*, 53(3):795–801, 2013.

[15] G. Gan. Application of metamodeling to the valuation of large variable annuity portfolios. In *Proceedings of the Winter Simulation Conference*, pages 1103–1114, 2015.

[16] G. Gan. A multi-asset Monte Carlo simulation model for the valuation of variable annuities. In *Proceedings of the Winter Simulation Conference*, pages 3162–3163, 2015.

[17] G. Gan, Q. Lan, and C. Ma. Scalable clustering by truncated fuzzy c-means. *Big Data and Information Analytics*, 1(2/3):247–259, 2016.

[18] G. Gan and E. A. Valdez. An empirical comparison of some experimental designs for the valuation of large variable annuity portfolios. *Dependence Modeling*, 4(1):382–400, 2016.

[19] G. Gan and E. A. Valdez. Regression modeling for the valuation of large variable annuity portfolios. Submitted to North American Actuarial Journal, July 2016.

[20] G. Gan and E. A. Valdez. Modeling partial greeks of variable annuities with dependence. Submitted to Insurance: Mathematics and Econocmics, 2017.

[21] H. Gerber and E. Shiu. Pricing lookback options and dynamic guarantees. *North American Actuarial Journal*, 7(1):48–67, 2003.

[22] X. Huang, Y. R. Huang, M. Wen, A. An, Y. Liu, and J. Poon. Applying data mining to pseudo-relevance feedback for high performance text retrieval. In *Proceedings of the 6th IEEE International Conference on Data Mining (ICDM 2006), 18-22 December 2006, Hong Kong, China*, pages 295–306, 2006.

[23] X. Huang, F. Peng, A. An, D. Schuurmans, and N. Cercone. Session boundary detection for association rule learning using n-gram language models. In *Advances in Artificial Intelligence, 16th Conference of the Canadian Society for Computational Studies of Intelligence, AI 2003, Halifax, Canada, June 11-13, 2003, Proceedings*, pages 237–251, 2003.

[24] E. Isaaks and R. Srivastava. *An Introduction to Applied Geostatistics*. Oxford University Press, Oxford, UK, 1990.

[25] J. F. Kolen and T. Hutcheson. Reducing the time complexity of the fuzzy c-means algorithm. *IEEE Transactions on Fuzzy Systems*, 10(2):263–267, 2002.

[26] M. C. Ledlie, D. P. Corry, G. S. Finkelstein, A. J. Ritchie, K. Su, and D. C. E. Wilson. Variable annuities. *British Actuarial Journal*, 14(2):327–389, 2008.

[27] NVIDIA. People like VAs like GPUs. *Wilmott magazine*, 2012(60):10–13, 2012.

[28] J. Oechslin, O. Aubry, M. Aellig, A. Käppeli, D. Brönnimann, A. Tandonnet, and G. Valois. Replicating embedded options in life insurance policies. *Life & Pensions*, pages 47–52, 2007.

[29] P. Phillips. Lessons learned about leveraging high performance computing for variable annuities. In *Equity-Based Insurance Guarantees Conference*, Chicago, IL, 2012.

[30] C. E. Rasmussen and C. K. I. Williams. *Gaussian Processes for Machine Learning*. MIT Press, Cambridge, MA, 2006.

[31] The Geneva Association Report. Variable annuities - an analysis of financial stability. Available online at: https://www.genevaassociation.org/media/618236/ga2013-variable_annuities.pdf, 2013.

[32] J. Vadiveloo. Replicated stratified sampling - a new financial modeling option. *Actuarial Research Clearing House*, 1:1–4, 2012.