Machine Learning Techniques for Variable Annuity Valuation

Guojun Gan
Assistant Professor
Department of Mathematics
University of Connecticut
Storrs, CT, USA
guojun.gan@uconn.edu

Zhiyu Quan
PhD Candidate
Department of Mathematics
University of Connecticut
Storrs, CT, USA
zhiyu.quan@uconn.edu

Emiliano Valdez
Professor
Department of Mathematics
University of Connecticut
Storrs, CT, USA
emiliano.valdez@uconn.edu

***Abstract*—Machine learning refers to a broad class of computational methods that use experience to improve performance or to make accurate predictions. There are two broad categories of machine learning tasks: supervised learning and unsupervised learning. Supervised learning tasks involve labeled data, which consist of inputs and their desired outputs. Unsupervised learning tasks involve unlabeled data, which consist of only inputs. In this paper, we give a brief overview of some machine learning techniques and demonstrate their applications in insurance. In particular, we apply data clustering and tree-based models to address a computational problem arising from the valuation of variable annuity products. Our numerical results show that tree-based models are able to produce accurate predictions and reduce the computational time significantly.**

***Index Terms*—data clustering, regression tree, variable annuity, portfolio valuation**

I. Introduction

The term machine learning was coined by Samuel in 1959 [1] to indicate the field of study where computers have the ability to learn without being explicitly programmed. Nowadays, the term has evolved to indicate the broad field of study where computational methods use experience, which refers to the past information available for analysis, to improve performance or to make accurate predictions [2], [3]. The experience usually takes the form of electronic data collected and made available to the learner. The quality and size of the experience are important to the performance of the learner. Since machine learning algorithms require data to learn, the field is closely related to and has significant overlap with several other areas such as data mining, pattern recognition, statistics, artificial intelligence, and neurocomputing [4].

In machine learning, the goal is to design efficient and accurate prediction algorithms. In particular, Tom M. Mitchell provided the following formal definition of algorithms studied in machine learning [5, p2]:

> A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P, if its performance at tasks in T, as measured by P, improved with experience E.

The quality of the algorithms is measured by their time, space, and sample complexities, which are used to evaluate the amount of time, the amount of memory space, and the sample size required by an algorithm to learn a family of concepts [6].

Major machine learning problems include classification, regression, ranking, clustering, and dimension reduction. In classification, the examples are divided into two or more categories and the learner produces a model to assign unseen examples to one or more of these categories. In regression, the labels assigned to examples are continuous rather than discrete. In ranking, examples are ordered according to some criterion. In clustering, examples do not have labels and the learner partitions the examples into homogeneous groups called clusters. Dimension reduction is also called manifold learning and its goal is to transform examples into a lower-dimensional space while preserving some properties of the examples.

Figure 1 shows a list of common machine learning tasks, which differ in the types of the training data, the order to receive the training data, and the test data. In supervised learning, the learner receives a training sample of labeled data and makes predictions for unseen examples. The aforementioned classification, regression, and ranking problems are associated with supervised learning. Unsupervised learning works with unlabeled data. Clustering and dimension reduction are examples of unsupervised learning problems. In semi-supervised learning, the learner receives a training sample that consists of both labeled and unlabeled data. Classification, regression, and ranking problems sometimes can

978-1-5386-5541-2/18/$31.00 ©2018 IEEE

be formed as semi-supervised learning problems. Transductive inference is similar to semi-supervised learning but with the goal to predict labels only for particular test points. Online learning involves multiple rounds of intermixed training and testing phases. Reinforce learning also involves intermixed training and testing phases and its goal is to maximize the reward over a course of actions and iterations with the environment. In active learning, the learner adaptively or interactively collects training examples with the goal to achieve a performance comparable to standard supervised learning, but with fewer labeled examples.

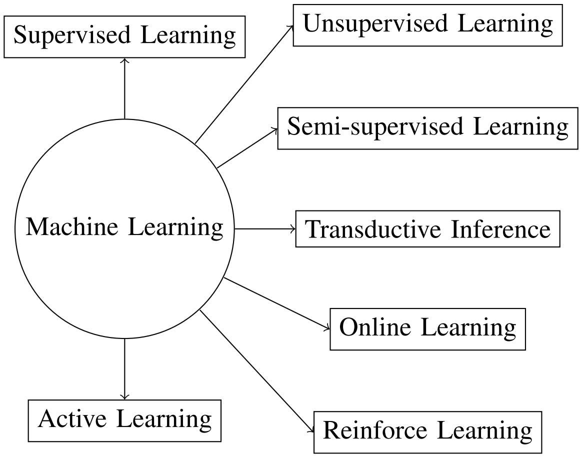

Fig. 1: Categories of machine learning tasks.

In this paper, we give a brief review of some commonly used machine learning algorithms. In particular, we focus on some supervised and unsupervised learning algorithms.

II. Unsupervised Learning

Unsupervised learning algorithms include clustering algorithms, principal component analysis, hidden Markov models, and some neural networks. In this section, we introduce data clustering.

Data clustering, also known as cluster analysis, refers to the process of dividing a dataset into homogeneous groups or clusters such that points in the same cluster are similar and points from different clusters are quite distinct [7], [8]. First originating in anthropology and psychology in the 1930s [9]–[11], data clustering has become one of the most popular tools for exploratory data analysis and has found applications in many scientific areas.

During the past several decades, many clustering algorithms have been proposed. Among these clustering algorithms, the k-means algorithm [12], [13] is perhaps the most widely used algorithm. To describe the k-means algorithm, let $X = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n\}$ be a dataset containing n points, each of which is described by d numerical features. Given a desired number of clusters k, the k-means algorithm aims at minimizing the following objective function:

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

where $U = (u_{il})_{n\times k}$ is an $n \times k$ partition matrix, $Z = \{\mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_k\}$ is a set of cluster centers, and $\|\cdot\|$ is the L^2 norm or Euclidean distance.

The k-means algorithm employs an iterative procedure to minimize the objective function. It updates the partition matrix U and the cluster centers Z alternately by allowing only one to change at a time. During the past several decades, many improvements of the k-means algorithms have been proposed. See [14] for details.

III. Supervised Learning

There exist a large variety of supervised learning algorithms: linear regression models, generalized linear models, support vector machines, neural networks, tree-based models, to just name a few. In this section, we introduce tree-based models.

Tree-based models involve dividing the predictor space (i.e., the space formed by independent variables) into a number of simple regions and using the mean or the mode of the region as the prediction [5], [15]. Tree-based models can be applied to both regression problems and classification problems. When applied to regression problems, the resulting tree model is called a regression tree. When applied to classification problems, the resulting tree model is called a classification tree.

To build a regression tree, the predictor space (i.e., the space formed by independent variables) is divided into non-overlapping regions such that the following objective function

$$f(R_1, R_2, \ldots, R_J) = \sum_{j=1}^{J} \sum_{i=1}^{n} I_{R_j}(\mathbf{x}_i)(y_i - \mu_j)^2 \quad (2)$$

is minimized, where I is an indicator function, R_j denotes the set of indices of the observations that belong to the jth box, μ_j is the mean response of the observations in the jth box, $\mathbf{x}_i$ is the vector of predictor values for the ith observation, and y_i is the response value for the ith observation. Building a classification tree is similar to building a regression tree. However, we use the mode (i.e., the most frequently occurring value) of a region to predict the response of an observation that belongs to the region.

Tree-based models generally do not perform to the level of other regression and classification models in terms of predictive accuracy. However, aggregating many trees has the potential to improve the predictive accuracy significantly [16]. Methods for aggregating trees include bagging, boosting, and random forests.

Bagging and boosting are general-purpose methods for reducing the variance of a statistical learning model. The random forest method uses multiple bootstrapped training samples. However, only a small subset of the predictors is used to fit a tree to a bootstrapped training sample. Tree-based models may outperform linear models when the relationship between the response and the predictors is nonlinear. When the response and the predictors are thought to have an approximately linear relationship, then linear models are likely to outperform tree-based models.

IV. Application in Variable Annuity Valuation

In this section, we demonstrate the application of machine learning techniques to address a computational problem arising from the valuation of variable annuity (VA) products.

A. Description of the Problem

A VA is a popular insurance product that is created by insurance companies as a tax-deferred retirement vehicle to address concerns many people have about outliving their assets [17]. A main feature of VAs is that they contain guarantees, which include guaranteed minimum death benefit (GMDB), guaranteed minimum accumulation benefit (GMAB), guaranteed minimum income benefit (GMIB), and guaranteed minimum withdrawal benefit (GMWB). These guarantees are financial guarantees that cannot be adequately addressed by traditional actuarial approaches. When the stock market goes down, for example, the insurance companies may lose money on all the VA contracts.

Dynamic hedging is widely adopted by insurance companies to manage the financial risks associated with variable annuities. Dynamic hedging requires calculating Greeks (i.e., sensitivities) of the guarantees embedded in variable annuities. Since the guarantees are complex, their fair market values, which are used to calculate Greeks, cannot be determined in closed form. Monte Carlo simulation is used in practice to calculate the fair market values of these guarantees.

One major drawback of Monte Carlo simulation is that it is computationally intensive. Using Monte Carlo simulation to calculate the fair market values of a large portfolio of VAs may take days or weeks [18], [19].

B. Approach based on Clustering and Tree-based Models

Recently, metamodeling techniques have been proposed to address the computational issues associated with the valuation of large VA portfolios. See, for example, [20]–[31]. The main idea of metamodeling techniques is to build a predictive model based on a small number of representative VA contracts in order to reduce the number of contracts that are valued by Monte Carlo simulation. As a result, a metamodeling technique involves the following four steps:

1) select a small number of representative contracts,
2) use Monte Carlo simulation to calculate the fair market values (or other quantities of interest) of the representative contracts,
3) build a regression model (i.e., the metamodel) based on the representative contracts and their fair market values,
4) use the regression model to estimate the fair market value for every contract in the portfolio.

In the past, data clustering [20], [22], [26], Latin hypercube sampling [21], [26], and conditional Latin hypercube sampling [23] have been used to select representative VA contracts from the portfolio. Ordinary kriging [20], universal kriging [23], and GB2 (Generalized beta of the second kind) regression model [28] have been used to build the metamodel. The advantage of kriging over the GB2 regression model is that the former does not require parameter estimation. However, one drawback of kriging is that the dependent variable (i.e., the fair market value) is assumed to follow a Gaussian distribution. This assumption is not appropriate for the fair market value of the guarantees. Although the GB2 regression model addresses the skewness of the dependent variable, estimating the parameters of the GB2 regression model posed additional challenges.

In this paper, we apply tree-based models to predict the fair market value of the guarantees embedded in a VA contract. To select representative VA contracts, we use the hierarchical k-means algorithm [32], which is efficient in dividing a large portfolio into hundreds of clusters.

C. Description of the Data

The dataset is a synthetic dataset created in [18] and contains 190,000 VA policies, each of which is described by 45 variables. Since some of the variables have identical values, we exclude these variables from the metamodeling process. The explanatory variables used to select representative VA contracts and build the tree-based models are described below:

- `gender` - Gender of the policyholder,
- `productType` - Product type of the VA policy,
- `gmwbBalance` - GMWB balance,
- `gbAmt` - Guaranteed benefit amount,
- `FundValue`i - Account value of the ith fund, for $i = 1, 2, \ldots, 10$,
- `age` - Age of the policyholder, and
- `ttm` - Time to maturity in years.

TABLE I: Summary statistics of the explanatory and the response variables.

Variable	Min	1st Q	Median	3rd Q	Max
gmwbBalance	0.00	0.00	0.00	0.00	499,708.73
gbAmt	0.00	186,864.95	316,225.98	445,940.63	1,105,731.57
FundValue1	0.00	0.00	12,635.17	49,764.15	1,099,204.71
FundValue2	0.00	0.00	15,107.17	56,882.55	1,136,895.87
FundValue3	0.00	0.00	10,043.96	39,199.69	752,945.34
FundValue4	0.00	0.00	10,383.79	39,519.79	610,579.68
FundValue5	0.00	0.00	9,221.26	35,023.00	498,479.36
FundValue6	0.00	0.00	13,881.41	52,981.06	1,091,155.87
FundValue7	0.00	0.00	11,541.47	44,465.70	834,253.63
FundValue8	0.00	0.00	11,931.41	45,681.16	725,744.64
FundValue9	0.00	0.00	11,562.79	44,302.35	927,513.49
FundValue10	0.00	0.00	11,850.05	44,967.78	785,978.60
age	34.52	42.03	49.45	56.96	64.46
ttm	0.59	10.34	14.51	18.76	28.52
fmv	-94,944.17	-5,142.94	12,488.63	66,814.16	1,536,700.08

Table I shows some summary statistics of the continuous explanatory variables and the response variable fmv. The dataset also has two categories explanatory variables: gender and productType. There are 19 different types of products and there are about 40% female and 60% male for each product type. The number of policies in each product type is the same, i.e., 10,000. The fair market values of the guarantees are calculated by a simple Monte Carlo simulation model [18]. A histogram of the fair market values is shown in Figure 2, which shows that the distribution of the fair market values is positively skewed.

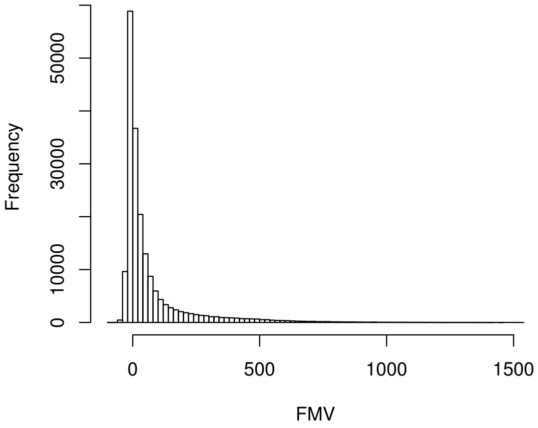

Fig. 2: A histogram of the fair market values (FMV). The fair market values are in 1000s.

D. Numerical Results

To demonstrate the performance of regression trees in predicting the fair market values, we first apply the hierarchical k-means to select k representative VA contracts from the portfolio. Then we build regression trees based on the k representative VA contracts and their fair market values. Finally, we use the resulting regression trees to predict the fair market value for each VA contract in the portfolio.

To measure the accuracy of the regression trees, we use the following two measures:

$$PE = \frac{\sum_{i=1}^{n}(\widehat{y}_i - y_i)}{\sum_{i=1}^{n} y_i}, \quad R^2 = 1 - \frac{\sum_{i=1}^{n}(\widehat{y}_i - y_i)^2}{\sum_{i=1}^{n}(y_i - \mu)^2}, \tag{3}$$

where y_i and $\widehat{y}_i$ denote the fair market values of the ith VA contract obtained from the Monte Carlo simulation model and the tree model, respectively, n is the total number of VA contracts in the portfolio, and μ is the average fair market value, i.e., $\mu = \frac{1}{n}\sum_{i=1}^{n} y_i$. The percentage error PE measures the accuracy of the result at the portfolio level. A lower absolute value of PE indicates a better result. The R^2 measures the accuracy of the result at the individual contract level. A higher value of R^2 means a better result.

We tested the performance of regression trees with different numbers of representative VA contracts. In particular, we tested $k = 340$, where the number 340 is determined to be ten times the number of regressors, which include the dummy binary variables converted from the categorical variables. We also tested $k = 680$ by doubling the number of representative VA contracts to see the impact of the number of representative VA contracts on the performance.

Table II shows the performance of four tree-based models: the regular regression tree model, the bagged model, the boosted model, and the random forest model. The runtime measures the time used to fit the model and make all predictions. From the table, we see that the regression tree model is the fastest model among the four models. The R^2 values show that the bagged model and the boosted model have higher accuracy at the individual contract level. Since the random forest method use a subset of explanatory variables to decides each split, its R^2 is lower. At the portfolio level, the prediction error is within 5% for all models. When we

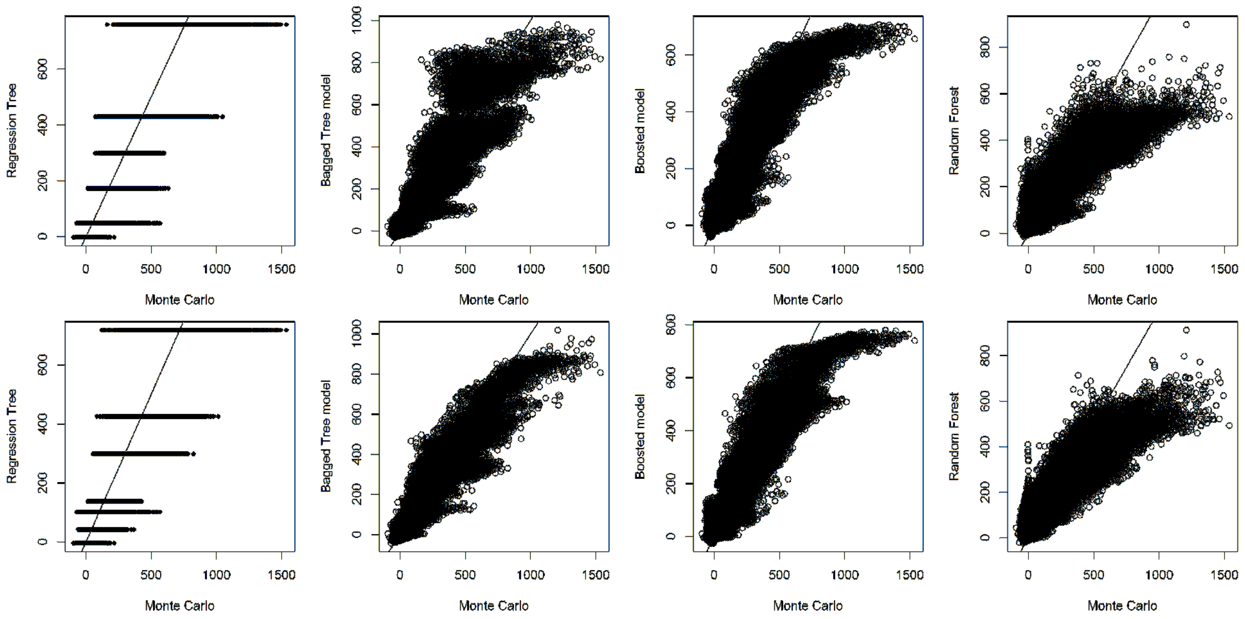

Fig. 3: Scatter plots of the fair market values calculated by Monte Carlo and those estimated by tree-based models. The four plots in the top are based on $k = 340$; the four plots in the bottom are based on $k = 680$.

TABLE II: Accuracy and speed of tree-based models. Here RT and RF denote regression tree and random forests, respectively. The runtime is in seconds.

(a) $k = 340$

	RT	Bagged	Boosted	RF
PE	-3.85%	-0.06%	2.15%	4.33%
R^2	0.81	0.88	0.88	0.78
Runtime	0.34	7.07	33.57	6.69

(b) $k = 680$

	RT	Bagged	Boosted	RF
PE	-3.19%	-3.11%	-1.10%	0.80%
R^2	0.84	0.92	0.90	0.82
Runtime	0.38	10.31	31.95	8.42

double the number of representative VA contracts, we see similar patterns for the accuracy.

Table II also shows the runtime of the tree-based models. The runtime includes the time used for fitting the tree-based models as well as the time used by the tree-based models to make all the predictions. From the table, we see that all the fitting and predictions can be done within 1 minute. Note that it took Monte Carlo simulation about 4 hours to calculate the fair market values for all the 190,000 VA contracts in the portfolio. Using the hierarchical k-means algorithm to select $k = 340$ and 680 representative VA contracts took about 130 and 136 seconds, respectively. To fit the tree-based models, we only require running Monte Carlo simulation for the selected representative VA contracts. This can be done within a few minutes. As a result, the approach based on clustering and tree-based models can reduce the runtime significantly.

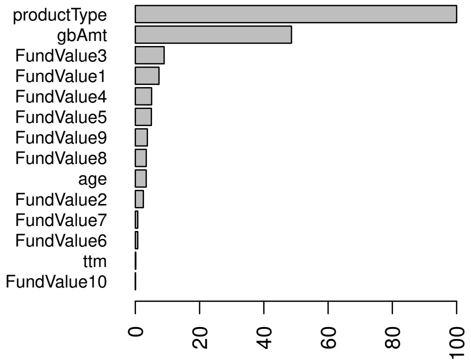

Fig. 4: Variable importance obtained from the regression tree.

Figure 3 shows the scatter plots of the fair market values calculated by Monte Carlo simulation and those predicted by the tree-based models. The figures show that the predictions made by the bagged model, the boosted model, and the random forest model are more accurate than those made by the regression tree model. If we compare the scatter plots in the top row and those in the bottom row, we see that doubling the number of representative VA contracts does not improve significantly the accuracy at the individual contract level.

Figure 4 shows the variance importance obtained from the tree model. From the figure, we see that the most important variables are productType and gbAmt. The least important variables are ttm and FundValue10. This makes sense because productType determines how the guarantee payoffs are calculated and gbAmt determines the amount of guarantee payments. The investment fund 10 is a balanced fund and has less impact the guarantee payoffs than do the equity funds such as the investment funds 1 and 3.

In summary, our numerical results show that tree-based models can make accurate predictions for the fair market values of guarantees embedded in VA contracts.

V. Concluding Remarks

Machine learning is a field that is closely related to and has significant overlap with several other fields such as data mining, pattern recognition, statistics, and artificial intelligence. In this paper, we provided a brief introduction to machine learning, with a focus on supervised and unsupervised learning. We also demonstrated the application of machine learning techniques in life insurance by using hierarchical k-means and tree-based models to address a computational problem arising from the valuation of VA products.

Acknowledgments

This work is supported by a CAE (Centers of Actuarial Excellence) grant (http://actscidm.math.uconn.edu) from the Society of Actuaries.

References

[1] A. L. Samuel, "Some studies in machine learning using the game of checkers," *IBM Journal of Research and Development*, vol. 3, no. 3, pp. 210–229, 1959.

[2] C. M. Bishop, *Pattern Recognition and Machine Learning*. New York, NY: Springer, 2007.

[3] M. Kubat, *An Introduction to Machine Learning*, 2nd ed. New York, NY: Springer, 2017.

[4] M. Mohammed, M. B. Khan, and E. B. M. Bashier, Eds., *Machine Learning: Algorithms and Applications*. Boca Raton, FL: CRC Press, 2017.

[5] T. M. Mitchell, *Machine Learning*. McGraw-Hill, 1997.

[6] M. Mohri, A. Rostamizadeh, and A. Talwalkar, *Foundations of Machine Learning*. Cambridge, MA: MIT Press, 2012.

[7] G. Gan, C. Ma, and J. Wu, *Data Clustering: Theory, Algorithms, and Applications*. Philadelphia, PA: SIAM Press, 2007.

[8] G. Gan, *Data Clustering in C++: An Object-Oriented Approach*, ser. Data Mining and Knowledge Discovery Series. Boca Raton, FL, USA: Chapman & Hall/CRC Press, 2011.

[9] H. E. Driver and A. L. Kroeber, "Quantitative expression of cultural relationships," *University of California Publications in American Archaeology and Ethnology*, vol. 31, no. 4, pp. 211–256, 1932.

[10] J. Zubin, "A technique for measuring like-mindedness," *Journal of Abnormal and Social Psychology*, vol. 33, no. 4, pp. 508–516, 1938.

[11] R. C. Tryon, *Cluster analysis; correlation profile and orthometric (factor) analysis for the isolation of unities in mind and personality*. Ann Arbor, MI: Edwards brother, Inc., 1939.

[12] G. S. Sebestyen, "Pattern recognition by an adaptive process of sample set construction," *IRE Transactions on Information Theory*, vol. 8, no. 5, pp. 82–91, 1962.

[13] J. Macqueen, "Some methods for classification and analysis of multivariate observations," in *Proceedings of the 5th Berkeley Symposium on Mathematical Statistics andProbability*, L. LeCam and J. Neyman, Eds., vol. 1. Berkely, CA, USA: University of California Press, 1967, pp. 281–297.

[14] D. Steinley, "k-means clustering: A half-century synthesis," *British Journal of Mathematical and Statistical Psychology*, vol. 59, pp. 1–34, 2006.

[15] L. Breiman, J. Friedman, C. J. Stone, and R. Olshen, *Classification and Regression Trees*. Raton Boca, FL: Chapman and Hall/CRC, 1984.

[16] G. James, D. Witten, T. Hastie, and R. Tibshirani, *An Introduction to Statistical Learning: with Applications in R*. New York, NY: Springer, 2013.

[17] M. Hardy, *Investment Guarantees: Modeling and Risk Management for Equity-Linked Life Insurance*. Hoboken, New Jersey: John Wiley & Sons, Inc., 2003.

[18] G. Gan and E. A. Valdez, "Valuation of large variable annuity portfolios: Monte Carlo simulation and synthetic datasets," *Dependence Modeling*, vol. 5, pp. 354–374, 2017.

[19] ——, "Nested stochastic valuation of large variable annuity portfolios: Monte carlo simulation and synthetic datasets," *Data*, vol. 3, no. 3, p. 31, 2018.

[20] G. Gan, "Application of data clustering and machine learning in variable annuity valuation," *Insurance: Mathematics and Economics*, vol. 53, no. 3, pp. 795–801, 2013.

[21] ——, "Application of metamodeling to the valuation of large variable annuity portfolios," in *Proceedings of the Winter Simulation Conference*, 2015, pp. 1103–1114.

[22] G. Gan and X. S. Lin, "Valuation of large variable annuity portfolios under nested simulation: A functional data approach," *Insurance: Mathematics and Economics*, vol. 62, pp. 138 – 150, 2015.

[23] ——, "Efficient greek calculation of variable annuity portfolios for dynamic hedging: A two-level metamodeling approach," *North American Actuarial Journal*, vol. 21, no. 2, pp. 161–177, 2017.

[24] S. A. Hejazi and K. R. Jackson, "A neural network approach to efficient valuation of large portfolios of variable annuities," *Insurance: Mathematics and Economics*, vol. 70, pp. 169 – 181, 2016.

[25] G. Gan and E. A. Valdez, "Modeling partial greeks of variable annuities with dependence," *Insurance: Mathematics and Econocmics*, vol. 76, pp. 118–134, 2017.

[26] ——, "An empirical comparison of some experimental designs for the valuation of large variable annuity portfolios," *Dependence Modeling*, vol. 4, no. 1, pp. 382–400, 2016. [Online]. Available: http://ssrn.com/abstract=2830879

[27] S. A. Hejazi, K. R. Jackson, and G. Gan, "A spatial interpolation framework for efficient valuation of large portfolios of variable annuities," *Quantitative Finance and Economics*, vol. 1, no. 2, pp. 125–144, 2017.

[28] G. Gan and E. A. Valdez, "Regression modeling for the valuation of large variable annuity portfolios," *North American Actuarial Journal*, vol. 22, no. 1, pp. 40–54, 2018.

[29] G. Gan and J. Huang, "A data mining framework for valuing large portfolios of variable annuities," in *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2017, pp. 1467–1475.

[30] W. Xu, Y. Chen, C. Coleman, and T. F. Coleman, "Moment matching machine learning methods for risk management of large variable annuity portfolios," *Journal of Economic Dynamics and Control*, vol. 87, pp. 1 – 20, 2018.

[31] G. Gan, "Valuation of large variable annuity portfolios using linear models with interactions," *Risks*, vol. 6, no. 3, p. 71, 2018.

[32] D. Nister and H. Stewenius, "Scalable recognition with a vocabulary tree," in *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, vol. 2, 2006, pp. 2161–2168.