A Statistical Approach to Money Laundering Detection in Insurance Company Using Genetic Algorithm

Mir Mahdi Seyed Esfahani[1]
Somayeh Molaei[2]
Akbar Esfahanipour[3]

Received: 05. Nov. 2013
Accepted: 21.Apr. 2015

Abstract

Insurance companies are faced with the challenge of money laundering. Money laundering is a complex, dynamic and distributed process which exposes insurance companies to legal, operational and reputational risks. Previous studies in insurance investigate the fraud in insurance and proposed different methods for fraud detection, while money laundering as a crucial phenomenon in insurance, which exposes the insurance company to risk, is neglected. We explore the money laundering in insurance and propose an efficient statistical method to detect it.

In this paper we propose a useful strategy which aimed at stratified sampling instead of exhaustive inspection for detecting money laundering activities. This approach is based on a division of insureds into homogeneous subgroups (strata). For this purpose, we firstly formulate the stratification task as a non-linear restricted optimization problem, in which the variance of overall amount of money laundered due to money laundering activities is minimized. Then we develop the metaheuristic approach namely the genetic algorithm (GA) to compute the optimum number of subgroups. The results show that the near optimum number of strata is 600, which means that we should divide insureds into 600 groups and survey these samples instead of surveying all insureds.

Keywords: *Money Laundering, Life Insurance, Stratified Sampling, Genetic Algorithm, Design of Experiments.*

1. Associated Professor, Amirkabir University of Technology. (msesfahani@aut.ac.ir)
2. PhD Candidate of Industrial Engineering, Iran University of Science and Technology. **Corresponding Author.** (molaei@aut.ac.ir)
3. Assistant Professor, Amirkabir University of Technology. (esfahaa@aut.ac.ir)

Vol. 1, No. 1, Spring 2015 Serial No. 1

1. Introduction

The insurance sector plays an important role in enhancing the economic growth of every country. This sector plays a crucial role as **both institutional investor and loan provider to various industries. This** will in turn be channeled to productive investments and lead to better economic expansion. Despite playing such an important role, it is unfortunately a particularly attractive sector for money launderers (Thanasegaran et al., 2008).

Money laundering is a process in which illegal or dirty money is put through a cycle of transaction (washing), so that it comes out the other end as legal or clean money. In other words, illegally obtained fund is obscured through a succession of transfers and deals in order that those same funds can eventually appear as legitimate income (Umadevi et al., 2012). Today, money laundering is the third largest "Business" in the world following the Currency Exchange and the Automobile Industry. According to researches, the value of laundered money in the recent years ranges from $500 billion to $1 trillion (Baker, 1999).

Researches show that two-thirds of the worldwide cases associated with money laundering in the insurance sector, are related to life insurance products. The availability of modified policies of life insurance enables the laundering of money in large or small sums (Thanasegaran et al., 2008). Transferring large sums is risky, because insurance companies may require reporting high value transactions. Therefore, money launderers usually break the money in smaller amounts and then transfer it to insurance industry (Umadevi et al., 2012).

Previous studies in insurance investigate the fraud in insurance and propose different methods for fraud detection, while money laundering as a crucial phenomenon in insurance, which exposes the insurance company to risk, is neglected. When money laundering takes place, insurers are exposed to legal, operational and reputational risks (Le Khac et al., 2010).To get over this problem, insurance companies have been implementing solutions to fight money laundering activities (Tsantsarova, 2008). Due to high inspection cost and very large number of insureds, an exhaustive inspection is

impractical. Therefore a carefully designed sampling procedure is needed. Stratified sampling is a useful method for solving such problems. This statistical method is a process of dividing members of the target population into homogenous subgroups (strata) before sampling. The approach in stratified sampling is to minimize the variance within groups and maximize it between groups. Given the total number of inspections to be executed, n and the total number of strata, k, the problem is how to allocate the n samples in the k strata, minimizing the statistical variance. A widely used approach to solve this problem is the proportional allocation, in which the n samples are distributed between the strata according to their respective mean consumptions (Garg, 2010). In this paper the allocation scheme is considered as a non-linear restricted optimization problem. In this application the metaheuristic algorithm namely the Genetic Algorithm (GA) is explored to determine the sample size within each stratum.

The rest of this paper is organized as follows:

Section II deals with related works on this subject. In section III, the stratified sampling is described in details. Section IV presents the statistical framework that yields the optimization problem. In section V, the metaheuristic method namely genetic algorithm which is employed in this paper is explained in details. The details of experimental setting are summarized in section VI. Finally the summary of the study and the major conclusion are provided in section VII.

2. Related researches

Money laundering in the insurance industry has close association with insurance fraud (Thanasegaran et al, 2008). Although insurance fraud has attracted the greatest attention from researches, there are only few studies on money laundering in insurance, which propose qualitative approaches to detect money laundering activities (Ngai et al., 2011).

Artis et al. have modeled different types of automobile insurance fraud behavior in Spanish insurance (Artis et al.,1999). Brockett and his colleagues have applied principal component analysis to classify the insurance fraud (Brockett et al., 2002). Bermúdez proposed the

Vol. 1, No. 1, Spring 2015 Serial No. 1

Bayesian model to detect the fraudulent activities in insurance (Bermúdez et al., 2008). Artís applied discrete choice model and misclassification claims to detect the fraud in automobile insurance (Artís et al., 2002). Thanasegaran has mentioned that the insurance sector is an attractive avenue for money launderers, therefore the insurance industry must be concerned about money laundering activities (Thanasegaran et al., 2008). Ramage addresses the phenomenon of money laundering and discusses the practical issues in identifying, minimizing and preventing money laundering in different economic sectors (Ramage, 2012).

3.Stratified Sampling

To improve the precision of the statistical surveys, it is advantageous to sample each subpopulation (stratum) independently, if subpopulations within an overall population vary. Stratified sampling is usually employed in this kind of status. Stratification is a process of dividing members of the population into homogeneous subgroups before sampling. The strata should be mutually exclusive and collectively exhaustive. In other words, every element in the population must be assigned to only one stratum and no population element can be excluded. Then a simple random sampling will be applied within each stratum. Simple random sampling is a basic type of sampling, in which, each individual is chosen randomly and entirely by chance. The principle of simple random sampling is that every object has the same probability of being chosen at any stage during the sampling process. It often improves the representativeness of the sample by reducing sampling error and can produce a weighted mean. The weighted mean is similar to an arithmetic mean, where instead of data points contributing equally to the final average, some data points contribute more than others and has less variability than the arithmetic mean (Tipton et al., 2013)

4.Statistical Formulation

It is assumed that the insureds have been divided in k groups (called strata) according to criteria such as the total amount of money paid for purchasing the life insurance, the insured history, the claim

history of insured, and the cancellation period of life insurance.

p_j, μ_j and δ_j^2 are the proportion of money launderers, the mean and the variance of money laundered per money launderers in the jth stratum respectively. For a given sample of $n_1, \dots, n_k$ insureds, if n_j is the number of insureds sampled in the jth stratum, the total amount of money laundered T due to the money laundering activities can be estimated by

$$\hat{T} = \sum_{j=1}^{k} N_j \hat{\mu}_j \hat{p}_j = \sum_{j=1}^{k} \hat{T}(1)$$

where N_j, p_j and μ_j are, respectively, the total number of insureds, the observed proportion of money launderers and the observed amounts of money laundered averaged over the money launderers in the jth stratum. Note that $\hat{T} = N_j \hat{\mu}_j \hat{p}_j$ is the estimated amount of money laundered within the stratum j.

The Objective Function

The variance of $\hat{T}$, $Var(\hat{T})$, has a complex formulation and is obtained by summation of variances of $\hat{T}_j(1, \dots, k)$, which in turn, depend on the variances of the $\hat{p}_j \hat{\mu}_j$. Assuming the independency between p_j and μ_j, it follows that

$$Var\left(\hat{p}_j \hat{\mu}_j\right) = Var(\hat{p}_j)Var(\hat{\mu}_j) + p_j^2 Var(\hat{\mu}_j) + \mu_j^2 Var(\hat{p}_j)(2)$$

From the binomial distribution, it can be assumed that $Var(\hat{p}_j) = p_j(1 - p_j)/n_j$. The variance of $Var(\hat{\mu}_j)$ depends on the number of money launderers detected by the inspectors in the stratum j. If the money laundering were detected at n_j insured, the variance of $\hat{\mu}_j$ would be σ_j^2/n_j. Note that the number of detected money launderers is a random variable itself. In fact, it follows a binomial distribution with parameters n_j and p_j. We have

$$Var(\hat{\mu}_j) = \frac{\sigma_j^2}{1-(1-p_j)} \sum_{k=1}^{n_j} \binom{n_j}{k} (p_j)^k (1 - p_j)^{n_j - k}(3)$$

and therefore

Vol. 1, No. 1, Spring 2015 Serial No. 1

$$Var(\hat{T}) = \sum_{j=1}^{k}(\hat{T}) = \sum_{j=1}^{k} N_j^2 \frac{N_j - n_j}{N_j} \times Var\left(\hat{p}_j \hat{\mu}_j\right) (4)$$

Given the number of strata k, and the number of desired inspection to be executed, n. The important issue is to allocate the n samples into k groups in such a way that $Var(\hat{T})$ is minimized. In other words, the target is to find a combination of $n_1, \dots, n_k$ that minimizes the variance of $\hat{T}$ subject to $n_1 + \cdots + n_k \leq n$. (de O Da Costa et al.,2013)

Due to the difficulty in finding analytical methods to minimize the variance, the use of the GA metaheuristic is suggested to solve this constraint minimizing problem.

5.Genetic Algorithm

The genetic algorithm (GA) is based upon Darwinian evolution theory. This algorithm is modeled on a relatively simple interpretation of the evolutionary process and is proven to be a reliable and powerful optimization technique in a wide variety of applications. It belongs to the larger class of evolutionary algorithms (EA), which generate solutions to optimization and research problems. As an optimization technique, the genetic algorithm simultaneously examines and manipulates a set of possible solutions (Duman et al., 2011).

Genetic algorithm starts with a randomly selected population of chromosomes represented by strings. The GA uses the current population of strings to create a new population such that the strings in the new generation are on average better than those in current population. The selection depends on the fitness value of every individual in the population. The fitness is usually the value of the objective function in the optimization problem being solved. In other words, the selection process determines which string in the current will be used to create the next generation. The new generation of candidate solutions is then used in the next iteration of the algorithm. Usually the algorithm terminates when either a maximum number of generations has been produced, or a satisfactory fitness level has

been reached for the population.

During each algorithm iteration the process of selection, reproduction, crossover and mutation take place to produce the next generation of solutions. Mutation is a genetic operator used to maintain genetic diversity from one generation of genetic algorithm chromosomes to the next. It alters one or more gene values in a chromosome from its initial state. In mutation, the solution may vary entirely from the previous solution. The purpose of mutation in GA is preserving and introducing diversity. Hence GA can provide better solutions by using mutation. Crossover is a genetic operator used to vary the programming of a chromosome or chromosomes from one generation to the next. It is analogous to reproduction and biological crossover, upon which genetic algorithms are based. It is a process of taking more than one parent solutions and producing a child solution from them. Crossover and mutation processes ensure that the GA can explore new features that may not be in the population yet. It makes the entire search space reachable, despite the finite population size. (Rajpaul, 2012).
Algorithm 1 shows the generic implementation of genetic algorithm.

Algorithm 1a generic genetic algorithm

```
1.  Encode solution space
2.  (a) Set pop_size, max_gen, gen = 0
    (b) set cross_rate, mutate_rate;
3.  initialize population
4.  while max_gen ≥ gen
5.  evaluate fitness
                For (i=1 to pop_size)
                 Select (mate1,mate2)
                 if (rnd(0,1)≤ cross_rate)
                          Child = crossover (mate1,mate2)
                 if (rnd(0,1)≤ mutate rate)
                              Child = mutation ();
                  Repair child if necessary
    end for
    Add offspring to new generation
    Gen = gen + 1
    End while
6.  return best chromosomes
```

In this implementation, each individual of GA is represented as a vector x having its size equal to the number of strata. Each

Vol. 1, No. 1, Spring 2015 Serial No. 1

component n_j of the vector refers to the number of insureds to be inspected in stratum j where $j = 1, \dots, k$. In the following subsection the problem constraint and the main stages of GA are explained in details.

Problem constraint

Three constraints are considered. When there are multiple objects around which a function accumulates, the inferior and superior limits extract the smallest and largest of them. Therefore, the first two constraints refer to the superior and inferior limits to the number of inspections in each stratum. Hence each stratum must have at least one inspection and the number of inspections must be less than N, the total number of the insureds of the strata.

The third constraint is related to operational situation, in which there is a maximum number of inspections due to the time and cost of execution. That is $\sum_{j=1}^{k} n_j = M$, where M represents the maximum number of inspections.

The number of inspections in each stratum is adjusted by algorithm 2 in a proportional manner. That is, given a vector of allocations, the algorithm guaranties that the values are in the limits stipulated by the lower and upper bounds.

Algorithm 2 Proportional Adjustment Algorithm

```
1.  procedure ADJUSTMENT (x, k, M)
2.      // fixing limits
3.      For i =1 → k do
4.  x [i]←max (min(x[i], N[i]), 1 )
5.  // proportional adjustment
6.  if sum(x) ≠ M then
7.  dif← M – sum(x); comp ← 0
8.  for i=1 → k do
9.  fix ← round (dif* (x[i] / sum (x)))
10. if x[i] + fix ≥ 1 and x[i] + fix ≤ N[i] then
11. x[i]←x[i] + fix; comp ←+ fix
12. // rounding adjustment
13. Ac ← dif - comp
14. Len ← length (x)
15. If ac > 0 then
16. while ac ≠ o do          >add inspections
17. for I in sample (range (1,len),len) do
18. if x[i] < N[i] then
19. x[i] ←x[i] + 1; ac ← ac -1
20. if ac = o then
21. break
22. else
```

```
23. while ac ≠ 0 do          >remove inspections
24. for I in sample (range(1,len),len) do
25. if x[i] > 1 then
26. x[i] ← x[i]-1; ac ← ac+1
27. if ac = 0 then
28. Break
```

Initialization of the genetic algorithm

Initial populations for the GA are generated randomly from the uniform probability distribution to create a representative set of individuals that tries to cover the whole search space. It creates a random vector at each call, which satisfies the constraints of the problem. In addition, a vector with proportionately defined components are inserted in each initial population, that is, a vector for which each component is of the form $n_j = M\hat{\mu}_j / \sum_{j=1}^{k} \hat{\mu}_j$, where $\hat{\mu}_j$is the average observation of money laundered in stratum j.

Generation of new vectors

In the preliminary tests for GA, several crossover and mutation operations are considered. Given two individuals, A and B, n positions are randomly selected and two new individuals, copies of A and B, receive their values at the selected positions from B and A, respectively. While the limits are automatically obeyed, an adjuster is again needed to keep the sum of the components correct.

The mutation operation substituted a value at a randomly selected position by a value with zero mean and a given variance. The rate of mutation determines for how many times the operation is called. An elitist selection, which is the best individual retained in a generation unchanged in the next generation, is applied in generating a new generation.

Stopping Criteria

Convergence is the stopping criterion applied in the GA algorithm. The convergence criterion is defined on the basis of the similarity of the elements in the population. The current population P is considered as converged, if for any two distinct elements p and q in P, dist $(p, q) < D_{min}$. Here dist(p, q) is defined as Euclidean distance between p and q and $D_{min} = 10$. This value is selected on the basis of the preliminary tests with several different D_{min}values.

Vol. 1, No. 1, Spring 2015 Serial No. 1

6.Experiments and Results

Insureds are divided in k strata by sorting them based on their key criteria mentioned in section IV, in k parts of equal size. The values of k are taken from the set $\{2,5,7,10,20,50,80,100,150,200,\dots,1000\}$. The values of the mean $\hat{\mu}_j$ and the variance σ_j^2 of money laundered are calculated for each stratum $j = 1,\dots,k$. The $\hat{\mu}_j$ and σ_j^2 are the mean and variance of money paid for purchasing the life insurance, in the stratum j. The values of the proportion of detected money launderers (p_j) in the stratum j are determined on the basis of previous experiments. Because of the intrinsic random nature of the GA algorithm, the experiments with each algorithm are conducted 30 times for each number of strata k, hence minimizing the stochastic effects.

Adjusting parameters

Finding good parameters is an important issue in applying a metaheuristic method. Therefore a proper method, namely Design of Experiments (DOE) is used to choose the parameters of the algorithm. DOE deals with planning, conducting, analyzing and interpreting controlled tests to evaluate the factors that control the value of a parameter or group of parameters (Anderson et al.,1994)

Genetic Algorithm

The GA requires six parameters to be set. The main parameters are the population and those related to crossover and mutation. The remaining parameters are set to a constant value for all experiments, such as the best individuals retained in a generation unchanged in the next generation (elites), number of crossover points and standard deviation of normal curve. Table 1 illustrates the parameter values found by Design of Experiment (DOE) for each instance of the problem.

Table 1.Parameter values found by DOE

Strata (k)	Crossover Rate	Mutation Rate	Population Size
2	0.75	0.11	100
5	0.75	0.50	200
7	0.60	0.50	200
10	0.90	0.50	200
20	0.67	0.6	100
50	0.75	0.11	90
80	0.65	0.06	95
100	0.60	0.07	100
150	0.75	0.06	80
200	0.8	0.05	50
250	0.70	0.04	70
300	0.8	0.03	80
350	0.65	0.04	60
400	0.5	0.05	50
450	0.75	0.03	80
500	0.8	0.02	90
550	0.85	0.01	130
600	1.0	0.001	180
650	0.90	0.001	100
700	0.95	0.01	70
750	0.85	0.01	60
800	0.95	0.01	80
850	0.90	0.01	60
900	0.95	0.001	50
950	0.90	0.001	130
1000	0.85	0.001	180

Vol. 1, No. 1, Spring 2015 Serial No. 1

Table2 summarizes the results produced by GA with the parameter values adjusted manually and using Design of Experiment.

Table 2.results produced by GA

Strata	0bjective values($\times 10^{13}$)				CPU times (s)			
(k)	Min	Max	Avg	S.D	Min	Max	Avg	S.D
2	4.570	4.570	4.570	**0.000**	3	5	4	0
5	2.629	2.629	2.629	**0.000**	88	348	190	60
7	2.245	2.245	2.245	**0.000**	38	110	65	17
10	1.938	1.938	1.938	**0.000**	80	1275	987	344
20	1.438	1.234	1.629	**0.000**	129	1380	980	300
50	1.237	1.237	1.237	**0.000**	187	1471	975	392
80	1.180	1.120	1.119	**0.000**	130	1570	1110	200
100	1.112	1.112	1.112	**0.000**	999	1735	1617	143
150	1.110	1.098	1.090	0.000	543	1890	1416	380
200	1.039	1.043	1.040	0.001	279	1914	1314	483
250	1.023	1.034	1.029	0.001	1694	2280	1980	180
300	1.014	1.016	1.015	**0.000**	2437	2672	2573	60
350	1.010	1.018	1.015	0.001	2456	2700	1599	62
400	1.007	1.020	1.015	0.003	2489	2729	2605	63
450	1.005	1.020	1.009	0.001	2957	3289	3380	78
500	1.002	1.008	1.005	0.001	3527	3871	3724	90
550	1.001	1.005	1.003	0.001	2959	3950	3410	483
600	**0.990**	**0.998**	**0.992**	0.002	2215	4237	3078	526
650	1.001	1.005	1.003	0.001	3695	4340	3870	240
700	1.002	1.007	1.004	0.001	4076	4474	4285	108
750	1.005	1.010	1.010	0.002	4283	4987	4790	118
800	1.008	1.019	1.014	0.003	4777	5226	5015	128
850	1.010	1.070	1.046	0.002	961	405	690	58
900	1.038	1.087	1.066	0.012	72	199	110	33
950	1.020	1.005	1.076	0.002	1943	3576	3980	349
1000	0.993	1.004	0.998	0.003	3207	5451	4204	534

The objective values (variance) related to the proportional allocation is shown in table 3.

Table 3. Objective values for different strata

Strata (k)	Objective values ($\times 10^{13}$)
2	4.595
5	2.778
7	2.385
10	2.061
20	2.020
50	1.290
80	1.200
100	1.171
150	1.134
200	1.115
250	1.110
300	**1.105**
350	**1.105**
400	**1.105**
450	1.112
500	1.114
550	1.120
600	1.126
650	1.132
700	1.137
750	1.143
800	1.151
850	1.158
900	1.165
950	1.172
1000	1.177

7. Conclusion and future research

In this paper, for the first time we have explored the money laundering in insurance and proposed an efficient statistical method to detect money laundering activities. In this study an exhaustive inspection is identified as impractical due to the high inspection cost and very large number of insureds. Therefore we have proposed a dynamic method to solve this problem. By Stratified Sampling we have categorized all the insureds into k stratas and minimized the variance of money laundered due to money laundering activities using genetic algorithm metaheuristic. The results show that the near optimum number of strata is 600, which means that we should divide

the insureds into 600 groups and survey these samples instead of surveying them all.

In future researches we can develop other metaheuristic algorithms, and compare their performance with genetic algorithm.

References

1. Anderson, M.J. and Whitcomb, P.J., 1974. *Design of experiments*. John Wiley & Sons, Inc.
2. Artís, M., Ayuso, M. and Guillén, M., 1999. Modelling different types of automobile insurance fraud behaviour in the Spanish market. *Insurance: Mathematics and Economics*, *24*(1), pp. 67-81.
3. Artís, M., Ayuso, M. and Guillén, M., 2002. Detection of automobile insurance fraud with discrete choice models and misclassified claims. *Journal of Risk and Insurance*, *69*(3), pp. 325-340.
4. Baker, R.W., 1999. The biggest loophole in the free market system. *Washington Quarterly*, 22(4), pp. 27-46.
5. Bermúdez, L., Pérez, J. M., Ayuso, M., Gómez, E., and Vázquez, F. J., 2008. A Bayesian dichotomous model with asymmetric link for fraud in insurance. *Insurance: Mathematics and Economics*, 42(2), pp.779-786.
6. Brockett, P. L., Derrig, R. A., Golden, L. L., Levine, A. and Alpert, M., 2002. Fraud classification using principal component analysis of RIDITs. *Journal of Risk and Insurance*, 69(3), pp. 341-371.
7. De O da Costa, E., Fabris, F., Rodrigues Loureiros, A., Ahonen, H., Varejao, F. M. and Ferro, R. M., 2013. Using GA for the stratified sampling of electricity consumers. In *Evolutionary Computation (CEC), 2013 IEEE Congress on* (pp. 261-268).IEEE.
8. Duman, E. and Ozcelik, M. H., 2011. Detecting credit card fraud by genetic algorithm and scatter search. *Expert Systems with Applications*, *38*(10), pp.13057-13063.
9. Garg, P., 2010. A Comparison between Memetic algorithm and Genetic algorithm for the cryptanalysis of Simplified Data Encryption Standard algorithm .*arXiv preprint arXiv,1004.0574.*
10. Le Khac, N.A., Markos, S. and Kechadi, M.T., 2010. A data

mining-based solution for detecting suspicious money laundering cases in an investment bank. In: *Advances in Databases Knowledge and Data Applications (DBKDA), 2010 Second International Conference on* (pp. 235-240).IEEE.

11. Ngai, E. W. T., Hu, Y., Wong, Y. H., Chen, Y. and Sun, X., 2011. The application of data mining techniques in financial fraud detection: A classification framework and an academic review of literature. *Decision Support Systems*, *50*(3), pp. 559-569.
12. Rajpaul, V., 2012. Genetic algorithms in astronomy and astrophysics. *arXiv preprint arXiv, 1202.1643.*
13. Ramage, S., 2012. Information technology facilitating money laundering. *Information & Communications Technology Law*, 21(3), pp. 269-282.
14. Thanasegaran, H. and Shanmugam, B., 2008. Exploitation of the insurance industry for money laundering: the Malaysian perspective. *Journal of Money Laundering Control*, 11(2), pp. 135-145.
15. Tipton, E., Hedges, L., Vaden-Kiernan, M., Borman, G., Sullivan, K. and Caverly, S., 2014. Sample selection in randomized experiments: A new method using propensity score stratified sampling. *Journal of Research on Educational Effectiveness*, 7(1), pp.114-135
16. Tsantsarova, D., 2008. *VersicherungsbetrugzumNachteil der Versicherung*.Universitat Wien
17. Umadevi, P. and Divya, E., 2012. Money laundering detection using TFA system. *In: Software Engineering and Mobile Application Modelling and Development (ICSEMA 2012) International Conference on* (pp. 1-8). IET.