Center for
Economic Theories and Policies
Sofia University St. Kliment Ohridski
Faculty of Economics and Business Administration

ISSN: 2367-7082

Bulgarian Economic Papers

Methods for Simulating Multi-dimensional Data for Financial Services Recommendation

Vasil Marchev & Angel Marchev, Jr.

BEP 02-2021
Publication: February 2021

Online: http://www.bep.bg
Contact for submissions and requests: bep@feb.uni-sofia.bg

Methods for Simulating Multi-dimensional Data for Financial Services Recommendation

Vasil Marchev[1] and Angel Marchev, Jr.[2]

ABSTRACT

This study is part of bigger research about self-learning systems for management of individualized investment portfolios. In this research we present several approaches for generating multi-dimensional synthetic data in conformity with the business logic, correlations, previous datasets, concatenation, neural networks, etc. Each approach is described algorithmically, and a brief comparative analysis is conducted at the conclusion of the paper. All described approaches rely to a different extend on real data as input – whether aggregated distribution or partially available real data to be enriched horizontally or vertically.

JEL: *C63, C81, G29*

Keywords: *self-learning systems, synthetic data, individualized investment portfolios*

[1] University of National and World Economy. Contact: marchev.vasil@unwe.bg

[2] University of National and World Economy. Contact: angel.marchev@unwe.bg

1. INTRODUCTION

This study is part of bigger research about self-learning systems for management of individualized investment portfolios. This paper emphasizes the necessity of multi-dimensional data for financial services in setting up a clustering model. In this research we present several approaches for generating multi-dimensional synthetic data in conformity with the business logic, correlations, previous datasets, concatenation, or neural networks, for instance.

The necessity of specific data assumes the establishment of several algorithms, based on the distributions of the main variables laid down in the business logic of the model. The different approaches aim to consider the models in the momentum of retaining the correlation between the set variables. The observed models are based on the necessity to identify the basic individual characteristics of persons, which implies the establishment of specific variables and their distributions.

This study focuses on alternative approaches in providing the model with the required amount and quality of data. The main stage of creating the whole model is the process of collecting, preparation, and analysis of the input information. The specificity of the data implies difficulties in securing the model with the information. This paper gives the prerequisites to develop the different approaches in the data simulation process. For the purpose of the assigned task, there are several approaches for simulating the necessary data which are discussed in the following tabs:

- Simulation based on a general distributions and business logic,
- Simulation based on general distribution and correlations,
- Inverse copula sampling,
- Data simulation based on a previous data,
- Probabilistic database concatenation,
- Generative adversarial network (GAN).

2. DATA – NECESSITY AND ACCEPTABILITY

The new waves of financial innovations, mainly driven by the emergence of new digital technologies, renewed customer demand in bank retail services, and Big data result in new approaches in customer relationship management through the so-called Fin-tech services. The individualization of investment solutions by AI approaches requires large arrays of personal data, which are typically GDPR protected and cannot be used for marketing purposes. In such a case the only reliable approach to launch a self-perfecting model of such automated service is to synthesize realistic data.

The specificity of the data, its complexity, and the lawfulness of the framework implies difficulties in securing the model with the necessary quantity and quality of information. An alternative approach is used, which aims to give the whole set of new data. These approaches are a simulation of a synthetic dataset. Synthetic data is unreal data, which is generated by statistical dependencies embedded in algorithms. Synthetic data can be used as test data, training data, and model validation. The advantages of this kind of information are the lack of limitations and the opportunity to retain the relationships between the variables

The main disadvantages of this approach that are observed are related to difficulties in the process of data generating. The quality of generated data dependence on the relevance and the quality of the model. Another disadvantage that is observed is the necessity of data validation with a real database. In this case, there is a must to get available real data set for validation.

3. Simulating by Distributions and Reductions by Business Logic

There are data distributions used for each of the selected factors. Distributions are obtained as a result of research, a priori assumptions, banking environment analysis, etc.

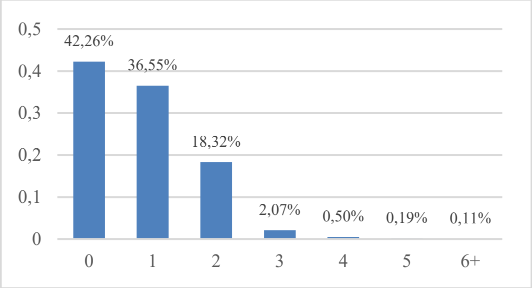
The main stages of the process are selecting variables, analysis of distributions, business logic, simulation, validation of the process. The main sub-elements of which base have developed the model of the simulation are demographic characteristics, personal characteristics, bank, and financial characteristics.

Analysis of Distributions

There are several basic approaches in the analysis of distributions for the purposes of data simulation: enough data, is not enough data, plausible assumptions:

- **When there is enough data for establishing the distributions** – assume unchanged conditions and keep the *posterior* distributions for each of the factors for which we have enough data (see for example figure 1).

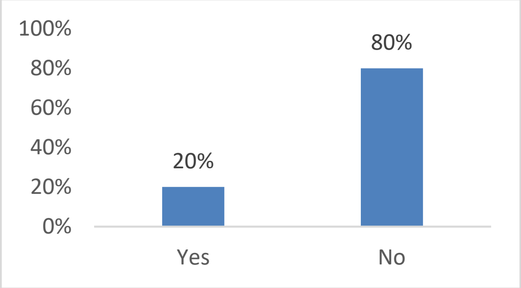
Figure 1: Relative share of families by number of children under 18



Source: National Statistical Institute (2012)

- **When there is not enough data to make distributions** – use the information which is available for the specific distribution parameters such as: mean value, minimum, maximum weighted average, etc. and make a partial simulation to obtain a *posterior* distribution, based on *prior* assumption for the type of distribution (see for example figure 2).

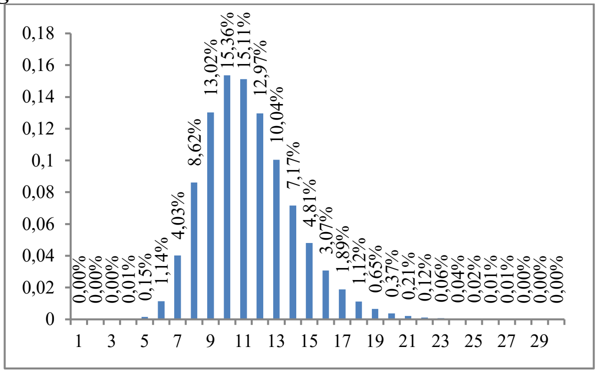
Figure 2: Relative share of individuals by ownership of car leasing, based on data for car ownership and prior assumption for leasing market



Source: Authors, based on National Statistical Institute (2012)

- **Plausible assumptions** – when there is no information to create a relevant distribution, the distribution is based on a plausible assumption on *prior* knowledge of the variables (see for example figure 3).

Figure 3: Relative share of individuals by average number of monthly bank operations, assuming realistic mean value and following log-normal distribution



Source: Authors

Business Logic

Business logic rules (see for example table 1), which are used in the data simulation model is received by selecting potentially interdependent factors, their limitations, possible combinations, and impossible ones. When combining the distributions there is a possibility to generate unattainable or very unlikely values. Trough implementation of the business logic of the model is aimed at removing the unlikely and impossible combinations.

The process of implementation the business logic is as follows:

- Selection of potentially interdependent factors,
- Find the possible and impossible combinations,
- Remove the impossible combinations.

Table 1: Sample business logic rules

Factor	Combining factor	Note
Age	Level of education	Until the age of 24, they are less likely to have tertiary education
Age	Marital status	Under 24 – is less likely to be married, widowed Over 64 years there is higher probability of being a widower
Age	Number of children under 18	Under 24 is less likely to have 1 or more child under 18 Over 65 years age, is unlikely to have children under 18

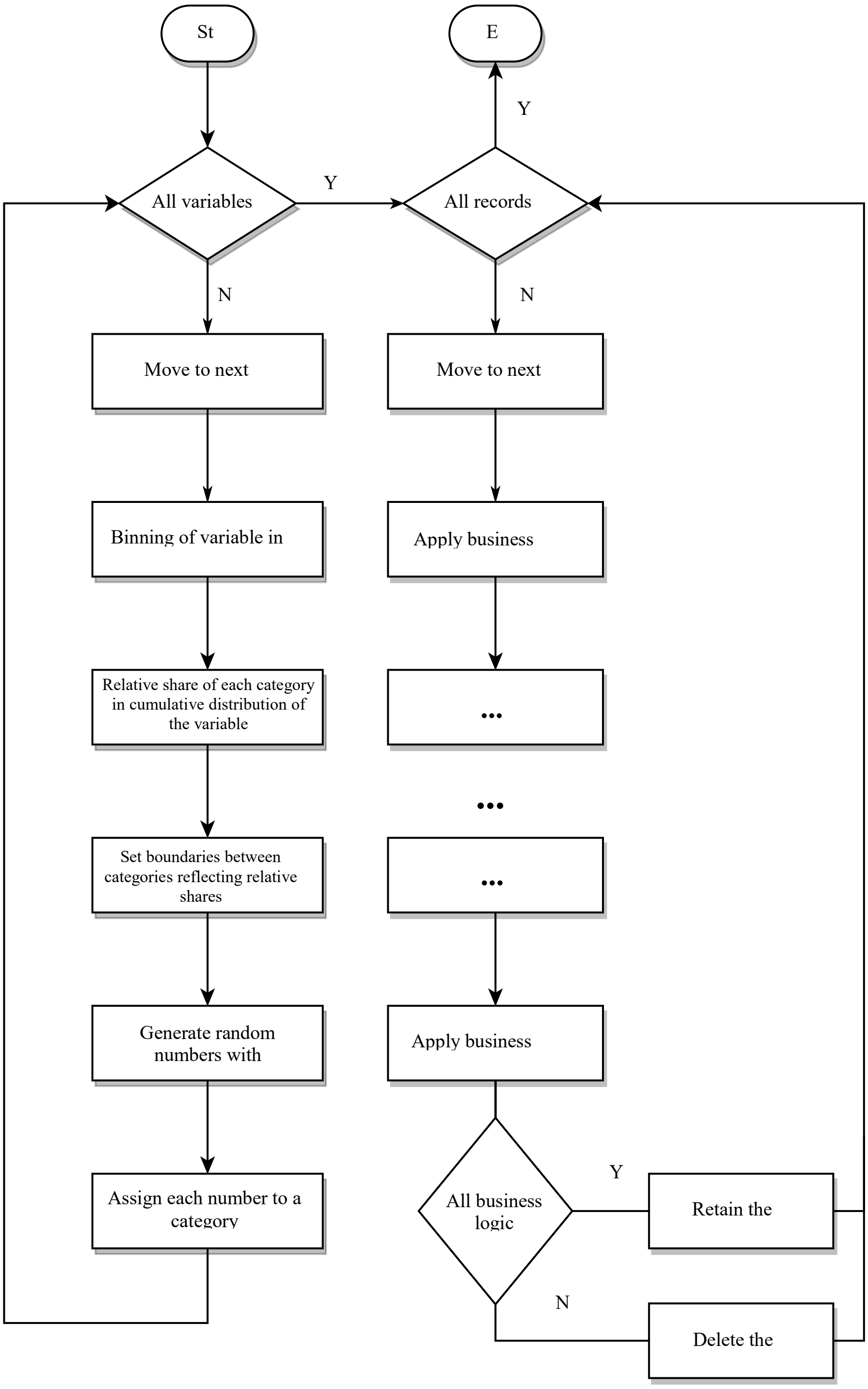
Source: Authors

Simulation

The method which is set up must tune the model about work with the observed distributions. This is a process of preparing algorithms in specialized software, in which the relevant restrictive conditions for the allocation of factors are set.

The process of data simulation (see figure 4) starts with the procedure of getting the univariate random variables from real data and their distributions. The simulated values must be according to the statistical distributions from the real data. This must be equivalent for all variables which are set in the model. When there is newly generated data there is a must to concatenate all of the simulated vectors in one matrix where each column is a different variable. In the process of data generation, it is possible to obtain unattainable or very unlikely values. In this case, the model goes through the reducing phase – the rows who do not correspond to the business rules are reduced from the matrix.

Figure 4: Process of data simulation



St
E
Y
All variables
Y
All records
N
N
Move to next
Move to next
Binning of variable in
Apply business
Relative share of each category in cumulative distribution of the variable
...
...
Set boundaries between categories reflecting relative shares
...
Generate random numbers with
Apply business
Assign each number to a category
All business logic
Y
Retain the
N
Delete the

Source: Authors

Mathematical algorithm

1) Given ω_m to be a univariate finite random variable of real data denoting the $m-th$ (out of s total) quantitative characteristics of a studied system S.

2) Using method of moments, the characterizing distribution $f_m(\omega_m, ; \theta_m)$ of ω_m with parameter set θ_m is obtained as solution to the following SSE

$$\begin{aligned}
\hat{\mu}_{m_1} &= \varphi_{m_1}(\hat{\theta}_{m_1}, \hat{\theta}_{m_2}, \dots, \hat{\theta}_{m_k}) \\
\hat{\mu}_{m_2} &= \varphi_{m_2}(\hat{\theta}_{m_1}, \hat{\theta}_{m_2}, \dots, \hat{\theta}_{m_k}) \\
&\vdots \\
\hat{\mu}_{m_k} &= \varphi_{m_k}(\hat{\theta}_{m_1}, \hat{\theta}_{m_2}, \dots, \hat{\theta}_{m_k})
\end{aligned}$$

Where $\phi_{m_1} \dots \phi_{m_k}$ are known functions and $\hat{\mu}_{m_1} \dots \hat{\mu}_{m_k}$ are the first k moments of f_m as derived from ω_m.

3) Generate values of simulated sample vector x_m with length of t, according to f_m, such that $\{x_m | f_m(x_m, \theta_m) \sim f_m(\omega_m, \theta_m)\}$ and is ordered randomly $x_m = \{x_{m_1}, x_{m_2}, \dots, x_{m_t}\}$.

4) Concatenate x_m in matrix X: $X_{t,s} = [x_1 \quad x_2 \quad \cdots \quad x_s]$.

5) Given a set of v business rules denoted by system of inequalities $A.z \leq b$, where A is $v \times s$ matrix, X is reduced to $\tilde{X}$ such that $\tilde{X}_{n,s} \subseteq X_{t,s}$ and $\tilde{X} = \{X | A.X_{t,*}^T \leq b\}$, where $n \leq t$.

4. Simulation Based on General Distributions and Correlations

This method is based on general distributions, which are available for the analyzed variables, comparing the correlations between them. The method is implemented by IBM SPSS Modeler – see International Business Machines (2017). There are two general ideas in this kind of simulation.

The first moment of this approach is connected with the general distributions of the variables. The data obtained must satisfy the statistical distributions. The simulated data must retain the initial general distribution of the variables.

The second main stage of the process is considered in correlations between variables. The algorithm is based on seated up correlations between variables. The correlations are dependent on the variables which are used for the simulated feature. The crucial elements for this method are the necessity of distribution and correlation between the used for simulation variables.

The algorithm starts with the procedure of getting the univariate random variables from real data and their distributions. These distributions are characterized with the method of moments. When there are the distributions of all variables, there is a necessity to standardize the values between minus one and one with the mean of zero. The generated values must be according to the normal distributions.

According to expert knowledge, there are obtained coefficients of correlation of the new variable according to one of the given variables. The process goes through the phase of the simulation of the new data that is standardized. The new data which is obtained must be de-standardized with scaling parameters which should be given as well. When there is a new column with generated information it must be concatenated with the rest of the matrix of the simulated data.

Mathematical Algorithm

1) Given ω_m to be a univariate finite random variable of real data denoting the $m-th$ (out of s total) quantitative characteristics of a studied system S.

2) Using method of moments, the characterizing distribution $f_m(\omega_m;\theta_m)$ of ω_m with parameter set θ_m is obtained as solution to the following SSE

$$\begin{aligned}\hat{\mu}_{m_1} &= \varphi_{m_1}(\hat{\theta}_{m_1},\hat{\theta}_{m_2},\dots,\hat{\theta}_{m_k})\\ \hat{\mu}_{m_2} &= \varphi_{m_2}(\hat{\theta}_{m_1},\hat{\theta}_{m_2},\dots,\hat{\theta}_{m_k})\\ &\vdots\\ \hat{\mu}_{m_k} &= \varphi_{m_k}(\hat{\theta}_{m_1},\hat{\theta}_{m_2},\dots,\hat{\theta}_{m_k})\end{aligned}$$

Where $\phi_{m_1}\dots\phi_{m_k}$ are known functions and $\hat{\mu}_{m_1}\dots\hat{\mu}_{m_k}$ are the first k moments of f_m as derived from ω_m.

3) Standardize ω_m values such that their mean is 0 and standard deviation of 1 unit by $\bar{\omega}_m = \frac{\omega_m-\hat{\mu}_{m_1}}{\hat{\mu}_{m_2}}$.

4) Generate values of simulated sample random vector w with length of n, according to normal distribution $\dot{f}$, such that $\{w|\dot{f}(w,\theta_{m_1}=0,\theta_{m_2}=1)\}$.

5) Given correlation coefficient ρ measuring with real data $\bar{\omega}_m$, simulate new data for standardized values of x_l, such as $\bar{x}_l = \rho.\bar{\omega}_m + \sqrt{1-\rho^2}.w$.

6) De-standardize $\omega_m = \bar{\omega}_m.\hat{\mu}_{m_2} + \hat{\mu}_{m_1}$ and $x_l = \bar{x}_l.\gamma_{l_2} + \gamma_{l_1}$, where $\gamma_{l_1}$and $\gamma_{l_2}$are scaling parameters, given for x_l.

7) Concatenate ω_m and x_l in matrix $\tilde{X}$: $\tilde{X}_{n,s} = [\omega_1 \quad \cdots \quad \omega_k \quad x_1 \quad \cdots \quad x_l]$, where $s = k + l$ columns.

5. INVERSE COPULA SAMPLING

A copula is a multivariate cumulative distribution function for which the marginal probability distribution of each variable is uniform on the interval [0, 1] (see Nielsen, 2006).

This method describes dependencies among variables and provides opportunities to create distributions that model correlated multivariate data. The algorithm provides an approach for constructing a multivariate distribution based on specifying optimal univariate distributions. Based on a copula the algorithm gives correlations structure between set variables. In this algorithm, it is possible to use bivariate distributions, as well as distributions in higher dimensions.

This algorithm uses one of the properties of a copula function where there is a necessity to fit the copula to the multivariate distribution and afterwards the algorithm inverse something of the copula function. The resulting values are going to be normalized between zero and one, so there is a necessity to de-normalize them by scaling parameters. The last stage of the process is concatenation of the matrix with the de-normalized data.

Mathematical Algorithm

1) Given $\varPhi = f_1, \dots f_s(\omega_1 \dots \omega_s)$ be a multivariate joint distribution of m finite random real data variables ω denoting m number of quantitative characteristics of a studied system S. Then the cumulative distribution function is defined as $\varPhi = P(\omega_1 \leq \widehat{\omega}_1, \omega_2 \leq \widehat{\omega}_2, \dots, \omega_s \leq \widehat{\omega}_s)$, where P denotes the probability that all random variables ω_i is less than or equal to the corresponding values $\widehat{\omega}_i$.

2) Use copula function C to generate simulated normalized sample vectors $(x_1, x_2, \dots, x_m)$ with length n from $\varPhi$ as

$$C(\bar{x}_1, \bar{x}_2, \dots, \bar{x}_s) = \varPhi[\omega_1 \leq {f_1}^{-1}(\bar{x}_1), \omega_2 \leq {f_2}^{-1}(\bar{x}_2), \dots, \omega_s \leq {f_s}^{-1}(\bar{x}_s)].$$

3) De-normalize $x_s = \bar{x}_s . \gamma_{s_2} + \gamma_{s_1}$, where $\gamma_{s_1}$and $\gamma_{s_2}$are scaling parameters, given for x_s.

4) Concatenate x_m in matrix $\tilde{X}$: $\tilde{X}_{n,s} = [x_1 \quad x_2 \quad \cdots \quad x_s]$.

6. DATA SIMULATION BASED ON A PREVIOUS DATA

Data simulation based on previous data uses basic, anonymized data, through which new variables and their corresponding records are synthesized. The main stages of the process are:

Real data – for the purposes of this approach there is a necessity of several variables of real data.

Anonymization of a real database – at this stage there is a necessity of encrypting the existing dataset. The process of encrypting the data set provides the removal of the names, personal numbers, emails, addresses, etc. In other words, the whole set of personal information, except the demographic one. This part of the process is a must in consideration of the GDPR, bank, and laws of the EU.

Simulation algorithm about new variables – the conditions according to which synthetic data is generated are programmed.

Different weights for basic variables – the model uses scaling factors and scaling coefficients obtained by the experts which satisfy the business rules. Different weights are used for each of the new variables that are generated. The weights depend on the correlation between the available variables and those to be simulated. There is a generated new matrix where all the original data and the data that has been simulated for exist.

Mathematical Algorithm

1) Given ω to be a multivariate finite random variable of real data denoting k number of quantitative characteristics of a studied system S.

2) Generate values of l simulated vectors x_l with length of n, according to $x_l = \omega.B$, where B is a column vector of coefficients satisfying a set of v business rules denoted by system of inequalities $A.(\omega.B) \leq b$, where A is $v \times l$ matrix.

3) Concatenate ω and x_l in matrix $\tilde{X}$: $\tilde{X}_{n,s} = [\omega \quad x_1 \quad \cdots \quad x_l]$, where $s = k + l$ columns.

7. Probabilistic Database Concatenation

This method is based on the concatenation of two or more existing real datasets. The approach is considered in the condition of connecting two or more real datasets, which are matched by a unique identification key. The identification key is created from two or more variables, which are present in both of the matrices.

In the essence of the algorithm, it occurs the necessity of analysis of the independent datasets. In the analysis, one observes of the existing real different datasets and their variables. When the datasets are observed there appears information that must be encrypted. The process of encrypting the data set is discussed in the previous section.

A combination of existing closed datasets goes through the creation of a unique identification key based on two or more variables. Based on this key the model creates an outer join and ends up with the full combination of all variants of the records.

In the process of probabilistic database concatenation, some of the generated values are impossible or repeated. In the new matrix, the data must be appropriate for modeling so it must be cleaned by removing the records that are repeated from the left matrix. The result is a new matrix reduced by deleting rows.

Mathematical Algorithm

1) Given ω to be one multivariate finite random variable of real data denoting k number of quantitative characteristics of a studied system S, while ω includes n rows of data (so it is $n \times k$ matrix).

2) Given λ to be another distinct multivariate finite random variable of real data denoting l number of quantitative characteristics of a studied system S, while λ includes t rows of data (so it is $t \times l$ matrix).

3) Given that there is an intersection between ω and λ of at least two columns (while $n \leq t$): $\{\omega\} \cap \{\lambda\} = \{x_1 \dots x_p | x_1 \dots x_p \in \omega \wedge x_1 \dots x_p \in \lambda \wedge p \geq 2\}$.

4) Perform left outer join of ω with λ (strictly in that order) $\omega \bowtie \lambda = \{\tau_{i,j} | i = n \wedge j = k + l - p\}$ by matching corresponding rows keyed on values of $\hat{x}_1, \dots, \hat{x}_p$, to obtain a concatenated matrix X: $X_{t,s} = [\omega \quad \lambda]$.

5) If there are matching subsets $\dot{\lambda}$ and $\dot{\omega}$, $\{\dot{\lambda} \subseteq \lambda | \hat{x}_1, \dots, \hat{x}_p \subseteq \dot{\lambda}\} \wedge \{\dot{\omega} \subseteq \omega | \hat{x}_1, \dots, \hat{x}_p \subseteq \dot{\omega}\}$, where $\dot{\lambda} > \dot{\omega}$, pick random row to be matched from $\dot{\lambda}$, so that X is reduced to $\tilde{X}$ such that $\tilde{X}_{n,s} \subseteq X_{t,s}$ and $\tilde{X} = \{\tilde{X} | \exists! \tilde{X}_{n,*}\}$, where $n \leq t$.

8. Generative Adversarial Networks (GAN)

The essence of the model as proposed by Goodfellow *et al.* (2014), the data simulation based on two adversative neural networks. The first neural network is a discriminator. The purposes of the discriminator are to classify samples from real and simulated data in the most appropriate way. The second neural network in GAN is a data generator. The generator is trained to mislead the discriminator as much as possible.

The generator is a neural network that models the transformation function. Input is a random variable that must be retrained to reach the optimal distribution of generated data.

Discriminator takes as an input starting value and returns as an output the probability that this point lies on the optimal distribution. A parameterized model is used to express both the generator and the discriminator.

The data generation model aims to mislead the discriminator. The generating neural network is trained to increase the maximum classification error (between real and generated data). The discriminator aims to detect incorrect generated data so that the discriminating neural network is trained to minimize the final classification error.

The main stages of the process are:

- The necessity of real data sets which is already obtained.
- Based on the real data the model uses a generative artificial neural network that generates values that are similar to the real dataset which are synthetic.
- On the next stage, the model uses another neural network which is discriminator. This neural network must discriminate whether the values that are available are real or seem real.

This is not an appropriate method to obtain values when there isn't a real data set. This is an applicable method to simulate synthetic data when there is real information about the variables. In this research, the method could be used as a validation method on how good the simulated data is.

Mathematical Algorithm

1) Given ω to be a multivariate finite random variable of real data denoting s number (out of s total) of quantitative characteristics of a studied system S, and n number of rows.

2) Using generative ANN, generate values of simulated matrix $\tilde{X}_{n,s} = \{\tilde{X}_{n,s} | \tilde{X}_{n,s} \neq \omega\}$.

3) Using discriminative ANN, discriminate for $\tilde{X}_{n,s} \sim \omega$.

9. CONCLUSION

As a conclusion the survey of the typical use case for all of the reviewed methods emerges as in Table 2. The first four algorithms should be used when there is some knowledge about the variables of the real data. The algorithms are considering some extend of posterior data - real distributions, and/or parameter of real distributions, and/or correlations, and/or real data for some of the variables. The fifth method should be used when there are several pieces of real data or simulated one and the data must be pieced together. This is a concatenation method, considering matching data specifics.

The last method is used for anonymizing real data by generating statistically similar data. It could also be used for validation how appropriate the simulated data is from any of the other methods in comparison to real data.

Table 2: Comparison table - all reviewed approaches

Comparative analysis	General distributions & business logic	General distributions & correlations	Inverse copula sampling	Partial Previous data	Probabilistic database concatenation	Generative adversarial networks
Essence	Generate data based on one-dimensional general distributions	one dimensional general distributions & correlations	Generate data based on multi-dimensional general distributions	Simulation-based on previous data and coefficients	Concatenation of two or more datasets	Self-learning system - based on an initial set of information
Required data	Distributions of the variables and business logic	Distributions of the variables and correlations	Copulas of multi-dimensional real data	Partial, real, anonymized data and business rules	Arrays of public anonymized data	Whole real dataset
Horizontal generation	-	Yes	Yes	Yes	-	Yes
Vertical generation	Yes	Yes	Yes	-	-	Yes
Synchronization & interconnection	Rules and business logic	Correlation between the variables	Copula	- Coefficients - If/Then rules	Probabilistic concatenation	Competition between neural networks
Main difficulties	Need for assumptions for some distributions. Post factum editing dataset	Need to clarify correlations	Extracting copula from real data	-Partial, real, anonymized data. -Generation of data based on already generated data	Arrays of anonymized data - free to use	A difficult and slow process of training neural networks
Final result	Simulation	Simulation	Simulation	Partial simulation	concatenated dataset	Simulation

Source: Authors

REFERENCES

1. National Statistical Institute (2012), Population census and the housing fund in 2011, National Statistical Institute, Sofia.

2. International Business Machines (2017), “Simgennode properties”, “IBM SPSS Modeler 18.1.1 Python Scripting and Automation Guide”, International Business Machines LLC., pp. 93.

3. Nelsen, R. B. (2006). An Introduction to Copulas (Second ed.). New York: Springer.

4. Goodfellow, I. J., J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, Y. Bengio, (2014), “Generative Adversarial Networks”, ArXiv.org @ Cornell University, https://arxiv.org/abs/1406.2661