Data Set Synthesis Based on Known Correlations and Distributions for Expanded Social Graph Generation

LUCIJA PETRICIOLI†, (Member, IEEE), LUKA HUMSKI†, (Member, IEEE), MIHAELA VRANIĆ, (Member, IEEE), AND DAMIR PINTAR, (Member, IEEE)
Faculty of Electrical Engineering and Computing, University of Zagreb, 10000 Zagreb, Croatia
Corresponding author: Lucija Petricioli (lucija.petricioli@fer.hr)

This work was supported by the European Regional Development Fund through the Operational Programme Competitiveness and Cohesion 2014–2020 - in part under the project Integrated Anti-Fraud System (IAFS; KK.01.2.1.01.0041) and in part under the project Advanced Methods and Technologies in Data Science and Cooperative Systems (DATACROSS; KK.01.1.1.01.0009).

ABSTRACT

Nowadays, data created through the usage of different services are most commonly not available to the average researcher. Security and privacy have become a top concern, which has further restricted access to certain real-life data, especially holding true for social networks. This is why synthetic data generators have become a very important area of research, particularly synthetic social graph generators. However, even today, such generators mostly create graphs that contain just the information whether two nodes are connected. Fortunately, there is an existing conceptual solution for an expanded social graph generator that aims to generate synthetic graphs containing multiple weighted edges between nodes, thus showing various types of relationships among those nodes, all based on known real-life data characteristics. One of its proposed steps is the generation of necessary data according to provided distributions and correlations. This paper focuses on the generation of such data by adapting an existing iterative algorithm for non-normal multivariate data simulation to generate synthetic data based on the publicly available distributions and correlations of Facebook interaction parameters. It is shown that the characteristics of the generated synthetic data are similar to the known characteristics of the real-life data, proving that the chosen algorithm, along with the accompanying alterations, can be used as one of the steps within the process of generating a synthetic expanded social graph.

INDEX TERMS

Correlation matrix, data distribution, social graph, synthetic data generation.

I. INTRODUCTION AND MOTIVATION

Data analysis has always greatly depended on the quality and availability of data. Gaining insight into and knowledge of the subject of analysis, creating predictive models which try to predict future events with great accuracy, this all hinges on having access to the right data. Unfortunately, many valuable data sets cannot be used for analysis, impeding researchers’ efforts. There are several common reasons, for example [1]:

- the data are disproportional (e.g., instances of fraud, outliers, etc.);
- the data are confidential (e.g., business contracts, medical records, personal data tied to a bank account, etc.);

The associate editor coordinating the review of this manuscript and approving it for publication was Chunsheng Zhu†.

- the data are expensive (e.g., the equipment needed to collect them is expensive, the collection requires many man-hours, etc.);
- the data are sparse (e.g., rare diseases, errors in complex systems, etc.).

While these cases present problems of varying degrees of difficulty, they all result in the need for an all-encompassing solution which could mitigate their impact. Generators of synthetic data would be one such solution. Data generators vary in construction, but ideally they should be able to generate a viable synthetic data set from sparse data or from their key characteristics, such as the distribution of attributes and correlations between them. A viable synthetic data set is one that retains the characteristics of the starting data set, while having different records. Data generation could possibly solve data sparsity, but that is exceedingly
difficult to test, since there are very few data to begin with, as well as there being very few new data to work with. Fortunately, data generation should help bypass the issues arising from disproportional data, as well as privacy and its protection. Attribute distributions and correlations should be much easier to obtain than existing user data, as they do not contain sensitive private data. Furthermore, a synthetic data set, unlike an anonymized real-life data set, cannot be deanonymized, since it contains no data belonging to a physical user. Therefore, it should allow researchers to carry out any analysis or experiment necessary, without putting others’ personal data at risk.

In general, very few researchers have the opportunity to work with proprietary data, let alone sensitive data pertaining to users. Even when such rarities do occur, the process of acquiring said data is at the very least grueling, as can be seen with the Scottish Longitudinal Study [2] and Eurostat [3]. The situation is further complicated by the advent and implementation of General Data Protection Regulation (GDPR), resulting in fewer data access grants as companies fear breaking the law.

On the other hand, the level of precautionary measures taken to secure user data does not come as a surprise after numerous instances of deanonymization of supposedly well-anonymized publicly shared data sets. A relatively well-known example came after the publication of a Netflix data set containing anonymized movie ratings which was supposed to be used for a data mining contest resulting in an improvement of their recommendation services - instead, Narayanan and Shmatikov developed a method which could reliably discover the identity of certain users with a surprisingly small amount of background knowledge [4]. The same authors later developed another method of deanonymizing anonymized social network graphs, whose efficacy is once again shown by testing on real anonymized data, in this case Twitter and Flickr [5]. Furthermore, with the amounts of personal data created every day as part social network usage, this field is becoming increasingly important. One more recent example of studies connected to data deanonymization is [6], where the authors argue there is often enough information left in an anonymized data set to identify users, chiefly through the still-existing relationships in the set.

This paper is a continuation of the research presented in [7], which describes a conceptual solution for an expanded social graph generator. That solution, amongst other prerequisites, requires input in the form of a generated synthetic list of total amounts of users’ interactions on online social networks summarized at the ego-user level. These generated synthetic data (list) should have attribute value distributions and correlations similar to the ones which characterize real-life users’ interactions. However, generating synthetic data according to predefined attribute value distributions and their mutual correlations is a challenge. As a solution, the aforementioned paper proposes using a method, developed by Ruscio and Kaczetow, researchers at the Department of Psychology of the College of New Jersey [8], which simulates non-normal multivariate data.

The research described in this paper, building upon [7], sets out to examine whether the proposed method really is applicable to the described problem, and whether other, more adequate methods can be found. A graphical representation of this approach is shown in Fig. 1. The first step is the analysis of prerequisites, i.e. researching existing methods for synthetic data generation and carrying out a comparative analysis of them, with emphasis on applicability to the posed data generation challenge. Analysis showed that the method proposed in [7] is the best solution that could be found to date, and therefore it is selected as the method this paper will be based upon. However, it still requires certain modifications to suit the posed data generation problem. Namely, one significant drawback of the original, unmodified method is that it requires a real-life data set as input. Instead, the changes made as part of this research enable the use of attribute value distributions and correlations as input for the method, with the output being a synthetic data set with similar key characteristics (distributions and correlations). The modified method is then experimentally evaluated, i.e. synthetic data sets are generated and their key characteristics are compared to those of a real-life data set. Finally, the guidelines for obtaining results whose key characteristics are as similar to the predefined ones as possible are given.

The structure of this paper is as follows: section II defines the terms used, then analyzes and compares the candidate methods for synthetic data generation found in related works; section III details the method chosen and changes made to suit the desired result; section IV gives an experimental evaluation of the modified method; V analyzes the presented results, gives some final remarks and highlights important aspects of the method; section VI concludes what has been achieved and offers ideas towards future work.

II. BASIC TERM DEFINITIONS AND RELATED WORK
In this section, important and frequently used terms are defined. Afterwards, a brief overview of different data generation methods and data generators is given, as is the final choice of method, along with the reasoning why it was selected.
A. BASIC TERM DEFINITIONS AND EXPLANATIONS

A real-life data set is a data set created by collecting physical data (e.g., user data, sensor data, medical data, etc.) through data collection methods (e.g., surveying, measuring, testing, etc.). On the other hand, a generated synthetic data set, synthetic data set, or generated data set, is a data set which is made according to characteristics obtained from a real-life data set. It is important to note that, ideally, this data set should not have any physical connection to a real-life data set, meaning there should be no records from the real-life data set present in the synthetic data set.

A synthetic data generator, or data generator for short, is a method that can produce a synthetic data set. Data generation signifies the process of generating a synthetic data set.

B. DIFFERENT APPROACHES TO DATA GENERATION

Since the aim of this work is to generate a synthetic data set according to predefined distributions and correlations, with the intent of using this solution as a step in the process of generating an expanded social graph, various data generation methods were studied, as was their applicability to the problem at hand. In [7], a method developed by Ruscio and Kaczetow [8] is highlighted as the most suitable. Therefore, this paper assesses whether this suggestion really is the best possible choice or whether a better approach can be found. Consequently, other promising methods were examined as well. The determining factor for a method to be viable is either the method working with key characteristics of the real-life data set (distributions and correlations), or the possibility of a modification being made which would allow the method to work with them.

The available body of work connected to data generation is extensive. Several approaches were identified [7]:

1) using a small real-life data set or a sample from the real-life data set, replicating its contents, and adding noise to the replicated data points [9], [10];
2) using the real-life data set to train a supervised or unsupervised machine learning algorithm, then using the hidden properties of the data the trained model learned to generate new data points [1], [11]–[13];
3) using the real-life data set’s (key) characteristics to generate a synthetic data set, without the need to use the contents of the real-life data set [8], [14], [15].

Unfortunately, both starting groups from the list have a glaring problem: they require at least some records directly from the real-life data set. This automatically invalidates these approaches, since the goal of this research is to generate synthetic data according to the key characteristics of a real-life data set, without having any contact with the records from this data set.

The third group offers some possible solutions to the posed problem. Several promising approaches were examined, along with the solution proposed by the authors of [7], as it belongs into this group.

The first of these approaches was the one developed by Lee et al. [14]. The authors’ data generator was to be used as a way of testing a military tracking system which should identify potential threats moving towards a location. Classification is key to this approach; each simulated object is created from certain rules tied to one of three possible classes of monitored means of transport in a military setting.

The next approach that was taken into account was that of Vale and Maurelli [15]. The authors proposed a method for simulating non-normal multivariate data by combining two approaches that had been developed some time before their published paper. The first approach was devised by Fleishman, and uses a normal distribution to simulate a non-normal one [16]. Each value is obtained by drawing a number from a normal distribution, and then using it to calculate the result of a formula of Fleishman’s making. To expand this approach to multivariate distributions, the authors of [15] combined it with the work of Kaiser and Dickman, who worked out a matrix decomposition method [17]. The two combined approaches allow multivariate non-normal data generation via the first 4 moments of each attribute’s distribution and the underlying correlation matrix tying the attributes together.

Finally, the approach proposed by [8] was studied in detail. Similarly to the aforementioned approach [15], it too uses an underlying correlation matrix which shows the relationships between attributes of a data set. However, this approach skips the calculation of distribution moments, and instead opts for an iterative way of determining the correct distributions of the data. The iteration continues until no significant improvement can be made to the measure of similarity between the original real-life data set’s characteristics and the characteristics of the synthetic data set being generated.

Once all the methods had been surveyed, their applicability to the task at hand had to be determined. The first listed method was immediately discarded, since it heavily relies on expert knowledge of man-made transportation and classification, whereas the posed data generation problem requires a method that could generate synthetic data according to attribute correlations and distributions. Even if the level of expert knowledge could be replicated in another setting, the data whose key characteristics would be used have to be classifiable, a requirement that would be severely limiting.

The choice between the other two aforementioned methods seemed somewhat more complicated at first, seeing that they should ultimately produce a very similar result. However, some clear advantages to the method described in [8] were found listed in that same paper. First of all, for Vale and Maurelli’s method [15] to function at all, the first 4 moments of any attribute’s distribution have to be defined (for mathematical proof, see [8]), whereas Ruscio and Kaczetow’s method has no such restriction. Furthermore, Headrick and Kowalchuk [18] found that, for a wider range of distribution functions (e.g., lognormal, $\chi^2$), the values of skew and kurtosis had to belong to certain intervals for the Vale-Maurelli method to work properly, a problem that is not present in the Ruscio-Kaczetow method. Also, the moments used for
calculation do not define a function uniquely, meaning the Vale-Maurelli method can replicate only one of the distributions sharing the 4 moments, once again a limitation which does not occur with the Ruscio-Kaczetow method. Finally, the Vale-Maurelli method requires the data it is being used on to be continuous, placing yet another restriction on what can be achieved by using this method. While the Ruscio-Kaczetow method does have its own limitations in this regard (the data have to be minimally ordinal by nature), they are nowhere near as restrictive as the ones present in the aforementioned method. Even after considering the intensity of the extra calculations needed for the Ruscio-Kaczetow method (which have become a much lesser problem in the years after the study was published), the aforementioned properties resulted in the decision to use it. This was further aided by the fact that the authors provided a documented reference implementation which could be used to generate synthetic data sets. It was also determined that, with all the information provided, the method really could be modified to work with key data characteristics instead of real-life data sets.

III. METHODOLOGY
This section starts off with the description of the algorithm for non-normal multivariate data simulation developed by Ruscio and Kaczetow [8] before any necessary alterations are made. Afterwards, any and all changes and additions are described in detail.

A. DESCRIPTION OF THE ORIGINALLY PROPOSED METHOD
Once the various approaches to data generation were investigated, and the approach by Ruscio and Kaczetow [8] was chosen, it was important to understand the inner workings of their method in order to decide which modifications should be made. Without modifying the code provided by the authors, the method works as follows:

The user provides the program with a (real-life) data set $D_{input}$ whose characteristics they wish to replicate in the synthetic data set. The number of columns $k$ of the supplied data set is the number of attributes present in it, whereas the number of rows $N$ indicates the number of cases (data points). The attribute values of this data set have to be minimally ordinal in nature for the method to function properly. Next, a distribution of values is generated for each attribute by sampling with replacement from the supplied data set $D_{input}$, with the same number of samples as there are cases in the set ($N$). All the sampled distributions are stored in a matrix $D_{distribution}$ of the same size as the supplied data set ($N \times k$), and will be used to produce the synthetic one. It is important to note that each of the distributions is sorted in an ascending order. A matrix $D_{simulated}$ of dimension $N \times k$ is also initialized to serve as a container for the simulated data.

The target correlation matrix $C_{target}$ for the synthetic data set is calculated from the supplied data set $D_{input}$. This correlation matrix is also stored as the intermediate correlation matrix $C_{intermediate}$, which will be altered as the algorithm iterates. If the user did not specify the number of factors (underlying variables) $f$ to be used to reproduce the correlations present in the target matrix $C_{target}$, it is calculated using parallel analysis.

Afterwards, two matrices are created and filled with random standard normal values. The first one ($S$) has the same number of rows as the supplied data set, and the same number of columns as there are factors ($N \times f$). It contains values that will be partly shared by all the attributes, their shared components. The second one ($U$) is the same size as the supplied data set ($N \times k$), and contains values that will contribute the unique (or error) component of each attribute. These matrices, combined with the weights calculated in the next step, serve to construct the simulated normal data set necessary for data generation.

The iterative part of the method begins with the calculation of the weights that will be applied to the shared and unique component matrices, $S$ and $U$ respectively. The shared component weights $W_{shared}$ (referred to as loadings in [8]) are calculated through a factor analysis of the intermediate correlation matrix $C_{intermediate}$ using the number of latent factors $f$. The dimension of this shared component weight matrix $W_{shared}$ is $k \times f$. After fine-tuning the weights (e.g. replacing all weights greater than 1 with 1, etc.), they are used to calculate the unique component weights $W_{unique}$ of size $k \times 1$ (the exact formula can be found in [8] under (2)). To calculate each of the simulated attribute values, all the calculated weights ($W_{shared}$ and $W_{unique}$) are used to weight the shared and unique components ($S$ and $U$) as follows:

$$D_{simulated,i} = (SW_{shared})_{s,i} + U_{*,i}W_{unique,i}$$

for $1 \leq i \leq k$, (1)

where $D_{simulated,i}$ is the $i$-th column of the matrix of simulated values $D_{simulated}$, $(SW_{shared})_{s,i}$ the $i$-th column of the matrix product of the shared component matrix $S$ and shared weight matrix transposed $W_{shared}^{T}$. $U_{*,i}$ the $i$-th column of the unique component matrix $U$, and $W_{unique,i}$ is the $i$-th value of the single column unique weight matrix $W_{unique}$. The matrix of simulated values $D_{simulated}$ now contains simulated normal values. Once it has been populated completely, its values are sorted and replaced with cases from the matrix of sampled distributions $D_{distribution}$. The sorting is done successively, column by column. After a column has been selected, the rows are ordered by its values in an ascending order. It is then replaced by the corresponding column from the $D_{distribution}$ matrix, after which the next column is selected. Once the process finishes, the simulated data set matrix $D_{simulated}$ contains simulated non-normal data.

In the next step, the correlation matrix $C_{reproduced}$ of the simulated data set $D_{simulated}$ is calculated. This matrix of reproduced correlations is then subtracted from the target correlation matrix ($C_{target} - C_{reproduced}$), the result of which is the correlation matrix of residual values $C_{residual}$. This in turn is used to calculate the root mean square residual (RMSR)
The simulated data set \( D \) is given, the modified version can only produce as many real-life data set or from theoretical distributions. Since the supplied data set \( D \) method [8] is changed, so it would accept distributions synthetic data according to the attribute value distributions These are the reasons why the method is modified to generate based on predefined distributions and correlations as its input. An expanded social graph [7] requires a synthetic data set concerns. Additionally, the conceptual solution for generating a real-life data set on hand is often not realistic due to various it requires a real-life data set as its input. However, having

\[
RMSR = \sqrt{\frac{\sum r_{\text{residual}}^2}{k(k-1)}},
\]

where \( r_{\text{residual}} \) signifies each of the values below the main diagonal of the matrix of residuals \( C_{\text{residual}} \), and \( k \) is the number of attributes of the supplied data set \( D_{\text{input}} \). If the calculated RMSR correlation is the smallest one yet, the current intermediate correlation matrix \( C_{\text{intermediate}} \) and RMSR correlation are stored as the best correlation matrix \( C_{\text{best}} \) and achieved RMSR correlation \( RMSR_{\text{best}} \) so far, a trial counter is reset (this counter gives a limit to calculations without improvement), the intermediate matrix \( C_{\text{intermediate}} \) is slightly altered by adding a small percentage of the values from the residual correlation matrix \( C_{\text{residual}} \), and the process is started again (weight calculation).

In case the RMSR correlation was not small enough, the trial counter is incremented. If the maximal number of trials without improvement has been reached, the program proceeds to the final step. Otherwise, a percentage of the residual correlation matrix \( C_{\text{residual}} \) is added to the best correlation matrix so far \( C_{\text{best}} \), and stored as the intermediate correlation matrix \( C_{\text{intermediate}} \), after which the program returns to weight calculation.

The final step of the program is to construct the final simulated data set \( D_{\text{simulated}} \) employing the process of weight calculation, shared and unique component weighting (according to (1)), and substitution of normal with non-normal data. However, the best achieved correlation matrix \( C_{\text{best}} \) is used instead of the intermediate correlation matrix \( C_{\text{intermediate}} \). The simulated data set \( D_{\text{simulated}} \) obtained by this procedure is returned as the final result of the described method, along with several other measures (e.g. its RMSR correlation, sample size, number of iterations, etc.).

**B. DESCRIPTION OF THE INTRODUCED ALTERATIONS**

As can be noted from the original method’s description, it requires a real-life data set as its input. However, having a real-life data set on hand is often not realistic due to various concerns. Additionally, the conceptual solution for generating an expanded social graph [7] requires a synthetic data set based on predefined distributions and correlations as its input. These are the reasons why the method is modified to generate synthetic data according to the attribute value distributions and correlations of real-life data, circumventing the absence of the real-life data set itself.

First and foremost, the input of the Ruscio-Kaczetow method [8] is changed, so it would accept distributions \( D_{\text{distribution}} \) and correlations \( C_{\text{target}} \) as its input instead of a supplied data set \( D_{\text{input}} \). It is assumed the distributions will be provided as distribution quantiles, either sampled from a real-life data set or from theoretical distributions. Since the original method produces the same amount of records as it is given, the modified version can only produce as many records as there are in the quantile matrix. As the generated data set should often have (many) more records than the number of provided quantiles, an interpolation module is written to make generating data sets of desired sizes possible. The quantile matrix output by the interpolation module can then be input into the modified method as \( D_{\text{distribution}} \), and a synthetic data set can be created.

The significance of the interpolation module is twofold:

1. it bypasses the issue of the method producing a synthetic data set that is the same size as the data set it is given;
2. it produces “new” (synthetic) records that fit the distributions of the data set the quantiles were extracted from.

Essentially, the module allows the method to produce any amount of generated synthetic records. Its design and functionality are described in the following paragraphs.

The interpolation module requires a set of distribution quantiles \( Q \), a desired set size \( N \), and a dilation factor \( d \). Each data point is created according to the following formula:

\[
\hat{Q}_{i,*} = \frac{Q_{i,*} + Q_{i+1,*}}{2} \quad \text{for} \quad 1 \leq i \leq M - 1,
\]

where \( \hat{Q}_{i,*} \) denotes a new row in the quantile matrix \( Q \) containing interpolated quantile values, and \( M \) the number of rows of the quantile matrix at the moment of calculation. The process of creating such interpolated values is repeated until the number of rows in the quantile matrix (now used as the interpolated quantile matrix) reaches a size greater or equal to \( N * d \). Once this condition is met, \( N \) rows are pseudorandomly sampled from the quantile matrix, and returned as the result. This result is to be used as the \( D_{\text{distribution}} \) matrix of the previously described method. The values of these data points will not be changed by the Ruscio-Kaczetow method [8], but they will be rearranged to achieve correlations that are as similar as possible to the predefined correlations.

After all the additions and alterations are made, the modified method requires a set of distribution quantiles, a correlation matrix, and a desired number of records in the soon-to-be-generated data set as its input, and gives a generated synthetic data set of desired size as its output.

**IV. EXPERIMENTAL EVALUATION**

After the development of the modules, data sets are generated to test the quality of the solution. The Facebook key data set properties (distributions and correlations) provided with [7] are used as the goal key characteristics. The output of the solution are generated data sets combined with their final RMSR correlation measures. The RMSR correlation measures are used to compare different generated synthetic data sets, and to analyze whether the proposed method and alterations are a viable data generation solution. The RMSR correlation is especially used when choosing the synthetic data set with key characteristics that are the most similar to the predefined ones. The lower the RMSR correlation value, the closer the values in the two correlation matrices are.
A. CREATING THE INTERPOLATED QUANTILE MATRIX
As Ruscio and Kaczetow note, their developed method works better with a greater number of data points [8]. Initial tests of the developed solution showed this conclusion held true for the interpolated quantile matrix that was to be fed to the algorithm. This led to the decision to work with an interpolated quantile matrix of 10,000 data points. To have a better dispersal of values that the attributes would assume, the quantile data points are sampled from a set with a size more than 20 times greater than the desired one.

B. DATA SET GENERATION AND QUALITY ASSESSMENT
Generated synthetic data set records are determined by the interpolated quantile matrix (prepared in the previous step), a predefined correlation matrix, the interpolated quantile matrix’s order of attribute columns, and a seed1. The first two are fixed characteristics of the soon-to-be-generated data set, whereas the second pair are parameters that provide randomness within the model. To check how seed changes and attribute column permutations impact the final result, different data sets are generated by using different seeds and various attribute column order permutations. The impact of these alterations is checked both individually and simultaneously, and it is noted they result in generated data sets with completely different records with slight differences in RMSR correlation values.

Fig. 2 shows how the RMSR correlation changes when different pseudorandomly selected seeds are fed to the algorithm. Changing the seed results in both the shared component matrix and unique component matrix from the method described in [8] having different element values than the ones obtained with another seed. Subsets of the present attributes are used for each measurement (the exact number is denoted on the x-axis) in order to check the influence seed changes have on data sets of different size. The attributes for each subset are once again chosen pseudorandomly, taking care that each larger subset is in fact a superset of the one preceding it. After a subset is taken, it is input into the algorithm 20 times; each time with a different seed (the same 20 seeds were used for each attribute subset).

The same subsets are then used to test how much difference the permutation of attributes makes (Fig. 3). The attributes are permuted by changing the order in which they are present in the data set. For the first two subsets there are 2 and 6 permutations, respectively. The rest are limited to 20, since the number of possible permutations grows in a factorial fashion. The seed used for all the measurements is one of the seeds used in the previous experiment.

As it can be observed that both seed changes and quantile matrix permutations have a level of influence on the generated data set and its calculated RMSR correlations, a combination of the two variations is tested. Fig. 4 shows the density of calculated RMSR correlations that are the result of 400 seeds being applied to each of 400 different quantile matrix permutations made for the task. The results of this hefty computation are then searched for the generated synthetic data set with the smallest RMSR correlation. The residual correlation matrix $C_{\text{residual}}$ (absolute values) for the generated data set with the lowest RMSR correlation can be seen in Fig. 5 (the values are represented as colored squares, where the size of the square reflects the size of the number within a color bracket - the actual number is written under the main diagonal; the meaning of the abbreviations and the description of the attributes can be found in [7]).

To further examine whether the generated data set’s key characteristics truly mimic the predefined ones, the quantiles of each of the attributes are calculated, then plotted onto the same graph with the predefined quantiles (Fig. 6; 4 of the 16 attributes present are plotted for the sake of brevity and clarity; their exact meaning can once again be found in [7]).

V. DISCUSSION
As can be seen in the previous section, the method presented in [8], together with the previously described additions and modifications, gives entirely satisfactory results. It has been shown that the method, in compliance with the suggestions made in [7], can be used as part of a future expanded social graph generator. Especially the residual correlation matrix plot (Fig. 5) and the quantile plot (Fig. 6) show the principal similarity of the predefined key properties and the ones of the generated data set. The key characteristics of the generated synthetic data sets do not have a 100% overlap with the predefined key characteristics, which they neither can nor should have. Namely, no exact key characteristics of a phenomenon exist. They are an approximation, which means there are certain deviations depending on the sample they were extracted from. Thus, variance among the key characteristics of generated synthetic data sets is both expected and desirable, as is the variance between the predefined key characteristics and the ones from synthetic data sets. Only a domain expert can gauge how great these differences can be for the synthetic data set to still be considered relevant. It is worth noting that the generated data sets usually have completely different records even when the starting sampled interpolated quantile matrix is the same. This is a fine feature because it means the method can generate a variety of data sets with similar key properties.

It is very important to note that the similarity between the generated data set’s key characteristics and the predefined ones is influenced by seemingly trivial changes - the change of the seed the program is given (Fig. 2) and the permutation of the quantile matrix (Fig. 3). Those behaviors should be taken into account when generating a synthetic data set. If the first generated set does not exhibit a lower measure of RMSR correlation, it is entirely possible the seed or quantile matrix permutation choice was not favorable, resulting in a low score.

After such a large number of experiments is run, the results can be used to determine an approximate number of times a synthetic data set should be created with different

---

1 The seed is a number which incorporates randomness into the model.
FIGURE 2. RMSR correlation change in relation to various algorithm seeds for differently sized attribute subsets.

FIGURE 3. RMSR correlation change in relation to attribute order permutation for differently sized attribute subsets.
FIGURE 4. RMSR correlation density for 160 000 different combinations of quantile matrix permutations and seeds. The vertical line represents the mean of the set of values, while the colored areas show the values that are one standard deviation and two standard deviations away from the mean.

FIGURE 5. Residual correlation matrix $C_{\text{residual}}$ of a generated data set with the lowest RMSR correlation measure within the run experiments.
parameters (seeds and permutations) to achieve a high probability of it having a calculated correlation matrix whose RMSR correlation would be small. The formula that calculates the probability of at least one favorable outcome or success in a number of independent experiments is as follows:

\[ P(1 \leq X \leq n) = 1 - (1 - p)^n, \quad (4) \]

where \( X \) signifies a number of experiments, \( n \) the total number of experiments that were run, and \( p \) is the probability of the favorable outcome in question. However, (4) is only valid for binomial distributions, which means the 160,000 RMSR correlations have to be observed as a set of values of which part are favorable, and the rest unfavorable. As this division could be made arbitrarily, it was decided that the values should be subjected to a more palpable criterion than a random number. To this end, the mean and standard deviation of all the computed values are calculated. These are then used to construct criteria for two different ways of dividing the large set of values. The reasoning behind choosing two divisions is that one should be more general, and the other refined, so future researchers could decide how fine their calculations should be. The two criteria are

\[ \text{RMSR} < \mu - \sigma, \quad (5) \]

and

\[ \text{RMSR} < \mu - 2 \sigma, \quad (6) \]

respectively, where RMSR is an RMSR correlation value, \( \mu \) the mean, and \( \sigma \) the standard deviation of the RMSR correlations as a whole. Every value that fulfills the given conditions is considered a favorable event. Applying (5) to the values results in 28,559 of 160,000 being classified as successful experiments, and the rest as unsuccessful. The more restrictive rule (6) results in 837 of 160,000 values. After this conversion, the question of the probability of an experiment being favorable or not remains. This is solved by calculating the experimental probability of the defined favorable event, given by the following formula:

\[ \hat{p} = \frac{m}{N}, \quad (7) \]

where \( \hat{p} \) is the experimental probability, \( m \) the number of favorable occurrences within the experiments, and \( N \) the number of experiments over all. Therefore, after expressing (4) in terms of \( n \) and substituting \( p \) with \( \hat{p} \) from (7), the following expression is obtained:

\[ n = \frac{\log(1 - P(1 \leq X \leq n))}{\log(1 - \hat{p})} = \frac{\log(1 - P(1 \leq X \leq n))}{\log(1 - m/N)}. \quad (8) \]

This equation is then solved, with \( m \) having the value of 28,559 and 837 for the less and more refined models respectively, and \( P(1 \leq X \leq n) \) set at 99%, to achieve a very high probability of a successful experiment. The result for the less refined division is \( n \approx 24 \), and for the more refined one it is \( n \approx 879 \). As can be seen, the number of the least number of experiments necessary is fairly low when compared to the number of experiments run for this paper. Even the higher number of experiments should not take longer than an afternoon.
VI. CONCLUSION AND FUTURE WORK

As can be seen from the results presented in this paper, the described extension of the method originally developed by Ruscio and Kaczetow [8] yields very promising results. The analysis of the generated data sets shows that these data sets replicate the predefined key characteristics well.

It is believed that greater data availability can be achieved by using the presented method. This would allow replacing many large, restricted data sets with surrogate data of adequate quality, resulting in more experts being able to run analyses on data which would otherwise be unobtainable.

To generate a synthetic data set with key characteristics very similar to predefined data set characteristics, the following guidelines should be followed:

1) the values which are present in the sampled interpolated quantile matrix have to be sorted to avoid a decline of the result quality, a side effect of the inner workings of the original method (presented in [8]);

2) the method should be run at least 25 or 1000 times using different seed and attribute order pairs (precise numbers for the case described in this paper can be seen in section V), depending on the desired level of similarity to the predefined key characteristics.

Ultimately, adapting the method proposed in [8] to replicate predefined attribute distributions and correlations between them has been successful. This represents an important step in a larger effort to develop an enriched social graph generator. Such a generator, which could create a social graph that would show more attributes of a connection or relationship between people, would be a great contribution to any social study, social network analysis, or the dynamics of an organization or company. It could also be used for fraud detection in telecommunications, where the structure of a generated social graph could help simulate suspicious activity and subsequently help model effective countermeasures.

REFERENCES

[1] M. Robnik-Šikonja, “Data generators for learning systems based on RBF networks,” *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 27, no. 5, pp. 926–938, May 2016.

[2] *Step-by-Step Guide to Accessing SLS Data*. Accessed: Jun. 11, 2019. [Online]. Available: https://sls.bcs.ac.uk/guides-resources/step-bystep-guide-to-accessing-sls-data/

[3] *Access to Microdata*. Accessed: Aug. 27, 2019 [Online]. Available: https://ec.europa.eu/eurostat/web/microdata/overview

[4] A. Narayanan and V. Shmatikov, “Robust de-anonymization of large sparse datasets,” in *Proc. IEEE Symp. Secur. Privacy*, May 2008, pp. 111–125. [Online]. Available: https://www.cs.utexas.edu/~shmat/summa/ok08netflix.pdf

[5] A. Narayanan and V. Shmatikov, “De-anonymizing social networks,” in *Proc. 30th IEEE Symp. Secur. Privacy*, May 2009, pp. 173–187. [Online]. Available: http://ieeexplore.ieee.org/document/5207644/

[6] D. Yin, Y. Shen, and C. Liu, “Attribute couplet attacks and privacy preservation in social networks,” *IEEE Access*, vol. 5, pp. 25295–25305, 2017.

[7] L. Humski, D. Pintar, and M. Vranic, “Analysis of Facebook interaction as basis for synthetic expanded social graph generation,” *IEEE Access*, vol. 7, pp. 6622–6636, 2019.

[8] J. Ruscio and W. Kaczetow, “Simulating multivariate nonnormal data using an iterative algorithm,” *Multivariate Behav. Res.*, vol. 43, no. 3, pp. 355–381, Sep. 2008.

[9] N. Itikhar, X. Liu, F. E. Nordbjerg, and S. Danalachi, “A prediction-based smart meter data generator,” in *Proc. 19th Int. Conf. Network-Based Inf. Syst. (NBiS)*, Dec. 2016, pp. 173–180.

[10] R. A. Hazwani, N. Wahida, S. I. Shafikah, and P. N. Ellyza, “Automatic artificial data generator: Framework and implementation,” in *Proc. 1st Int. Conf. Inf. Commun. Technol. (ICICTM)*, Mar. 2017, pp. 56–60.

[11] S. Yang, Y. Zhou, Y. Guo, R. A. Farneth, I. Marsic, and B. S. Randall, “Semi-synthetic trauma resuscitation process data generator,” in *Proc. IEEE Int. Conf. Healthcare Informat. (ICHI)*, Sep. 2017, p. 573.

[12] R. Liu, B. Fang, Y. Y. Tang, and P. P. Chan, “Synthetic data generator for classification rules learning,” in *Proc. 7th Int. Conf. Cloud Comput. Big Data (CCBD)*, Jul. 2017, pp. 357–361.

[13] B. Nowok, G. M. Raah, and C. Dibben, “synthpop: Bespoke creation of synthetic data in R,” *J. Stat. Softw.*, vol. 74, no. 11, pp. 1–26, Oct. 2016.

[14] J. Lee, J. Hong, B. Hong, and J. Ahn, “A generator of test data set for tactical moving objects based on velocity,” in *Proc. IEEE Int. Conf. Big Data, Big Data*, Dec. 2016, pp. 4011–4013.

[15] C. D. Vale and V. A. Maurelli, “Simulating multivariate nonnormal distributions,” *Psychometrika*, vol. 48, no. 3, pp. 465–471, Sep. 1983.

[16] A. I. Fleishman, “A method for simulating non-normal distributions,” *Psychometrika*, vol. 43, no. 4, pp. 521–532, Dec. 1978.

[17] H. F. Kaiser and K. Dickman, “Sample and population score matrices and sample correlation matrices from an arbitrary population correlation matrix,” *Psychometrika*, vol. 27, no. 2, pp. 179–182, Jun. 1962.

[18] T. C. Headrick and R. K. Kowalchuk, “The power method transformation: Its probability density function, distribution function, and its further use for fitting data,” *J. Stat. Comput. Simul.*, vol. 77, no. 3, pp. 229–249, Mar. 2007.

LUCIA PETRICIOLI (Member, IEEE) received the B.S. and M.S. degrees in computer science from the Faculty of Electrical Engineering and Computing, Zagreb. She is currently pursuing a Ph.D. degree at the Faculty of Electrical Engineering and Computing, University of Zagreb. She is also a Young Researcher at the Faculty of Electrical Engineering and Computing, University of Zagreb. Her research interests include synthetic data generation, social network analysis, data mining, machine learning, and electronic business. He works as a researcher on various science-related projects.

LUKA HUMSKI (Member, IEEE) received the master’s and Ph.D. degrees (both with summa cum laude) from the Faculty of Electrical Engineering and Computing, University of Zagreb. He is currently a Postdoctoral Researcher at the Faculty of Electrical Engineering and Computing. His research interests include synthetic data generation, social network analysis, data mining, machine learning, and electronic business. He works as a researcher on various science-related projects.

MIHAELA VRANIC (Member, IEEE) is currently an Assistant Professor at the Faculty of Electrical Engineering and Computing, University of Zagreb. She is also a Postdoctoral Researcher at the Faculty of Electrical Engineering and Computing. Her research interests include data generation, social network analysis, machine learning, and data mining. She has been involved in many industry projects both as a researcher and as a project manager.

DAMIR PINTAR (Member, IEEE) is currently an Associate Professor at the Faculty of Electrical Engineering and Computing, University of Zagreb. His interests include data management, machine learning, data mining and big data technologies, with a particular focus on predictive modeling, and practical implementations of data science principles in industry and academia. He is actively involved in a number of data science-related projects in which he acts both as a researcher and as the project leader.