*DOI: 10.24996/ijs.2020.61.12.20*

**ISSN: 0067-2904**

# A Comparison of Different Estimation Methods to Handle Missing Data in Explanatory Variables

**Manal Jabbar Salman**
Department of Affairs of Higher Studies, University of Baghdad, Baghdad, Iraq

Received: 10/11/2019 Accepted: 29/1/2020

**Abstract**

Missing data is one of the problems that may occur in regression models. This problem is usually handled by deletion mechanism available in statistical software. This method reduces statistical inference values because deletion affects sample size. In this paper, Expectation Maximization algorithm (EM), Multicycle-Expectation-Conditional Maximization algorithm (MC-ECM), Expectation-Conditional Maximization Either (ECME), and Recurrent Neural Networks (RNN) are used to estimate multiple regression models when explanatory variables have some missing values. Experimental dataset were generated using Visual Basic programming language with missing values of explanatory variables according to a missing mechanism at random general pattern and some ratios of missing values (10%, 20%, and 30%) with error variance values of 0.5, 1. 5, and 2, which were included in sample sizes of 25, 50, 100, and 500 and evaluated using Mean Squared Error (MSE). Simulation results show that RNN outperforms the other methods, followed by EM at small sample sizes.

**Keywords**: Missing data, Simulation, Recurrent Neural Networks, Expectation- Maximization, Multicycle –Expectation -Conditional-Maximization, Expectation-Conditional-Maximization-Either

# مقارنة بين طرائق تقدير مختلفة للتعامل مع البيانات المفقودة في المتغيرات التوضيحية

**منال جبار سلمان**

قسم شؤون الدراسات العليا، جامعة بغداد، بغداد، العراق

**الخلاصه**

البيانات المفقودة هي واحدة من المشاكل التي قد تحدث في نماذج الانحدار، تتم معالجة هذه المشكلات عادةً بواسطة آلية الحذف المتوفرة في البرامج الإحصائية. هذه الطريقة تؤدي الى تقليل قيم الاستدلال الإحصائي لأن الحذف يؤثر على حجم العينة. في هذا البحث, تم استخدم خوارزمية تعظيم التوقع (EM) ، خوارزمية تعظيم التوقع الشرطي المتعددة (MC–ECM)، خوارزمية تعظيم التوقع الشرطي الاخر (ECME) ، والشبكات العصبية المتكررة (RNN) المستخدمة لتقدير نموذج الانحدار المتعدد عندما المتغيرات التفسيرية تفقد بعض قيمها. تم توليد مجموعة من البيانات التجريبية باستخدام لغة برمجة Visual Basic بوجود القيم المفقودة في المتغيرات التفسيرية وفقًا لآلية الفقد العشوائي والنمط العام ولبعض نسب الفقد (10٪ و 20٪ و 30٪) لتباين الخطأ قدره (0. 5 ، 1. 5 و 2)

Email: manaljabbar4@gmail.com

ولحجوم عينات (25 و 50 و 100 و 500). لغرض تقييمها تم استخدام المقياس متوسط تربيع الخطأ(MSE) . نتائج المحاكاة اظهرت تفوق RNN على الطرائق الأخرى تتبعها خوارزمية EM لأحجام العينة الصغيرة.

## 1. Introduction

In applied statistics, studying data with missing values is an important topic because analysis of this data gives inaccurate and unreliable results and most specialist programmers rely on deleting these values when analyzing them [1]. Many researchers studied the estimating model in case of an incomplete data problem [2]. Liu and Rubin (1995) used the EM algorithm and its extensions of ECM and ECME to obtain more efficient estimates of maximum likelihood (ML) and in models analyzing factors that may arise in the contexts of educational tests [3]. Bori (2013) used EM algorithm by estimating the mean and the covariance matrix to obtain consistent estimates for the model parameters under study using SAS [4]. Harshanand (2013) studied the rainfall data that suffers missing values using neural networks (NN) and concluded that this method gives strong results, reflecting the uncertainty from missing values [5]. Wellin *et al.* (2018) used NNs to handle missing data by replacing neurons in the hidden layers through their expected values, which showed that this method achieves a cost reduction [6]. The proportions of the missing data are not fixed, so the mechanisms and patterns of missing data differ depending on the probability and position of the missing value in the data. In this paper, different sample sizes were generated, containing missing ratios, using Visual Basic for Applications (VBA) according to general missing pattern and missing at random mechanism. Several algorithms (EM, ECME, MC-ECM, and RNN) were compared. Simulation results showed that the use of RNN generated best results. The remaining sections of the paper are organized as follows: section 2 displays missing values in multiple linear regression, section 3 explains the methods adopted in this paper, section 4 presents simulation results with discussion, and finally the conclusions are drawn in section 5.

## 2. Missing Values in Multiple Linear Regression

Missing data may occur for intentional or unintentional reasons [1]. To overcome this problem, researchers presented a number of methods, including the list wise deletion method, multiple compensation method, full information maximum likelihood method, and multiple imputation then deletion (MID) method. The best estimate was achieved from the replication EM algorithm [4], which was later developed into upgraded versions such as ECM, MC-ECM, and ECME algorithms. Recently, researchers focused on RNN [4, 7]; before a dealing with missing value, it is important to take a look at missing patterns and mechanisms [1]. Missing patterns depend on a number of variables that include missing values (single missing, bivariate missing, monotone missing, general missing, and symmetric pattern). Missing values may be in one or more variables, which can be either independent or dependent, or both. In addition, the percentage of missing values may sometimes reach 50% or higher [7]. There are many missing data mechanisms to clarify in this research; it is assumed that a matrix R represents data matrix which includes a dependent variable Y and an independent variable X [4, 8]. That is:

$$R=\{X\ Y\}$$

$$R=\begin{cases} R_{\text{miss}} \text{ if observation is missing} \\ R_{\text{obs}} \text{ if observation is not missing} \end{cases}$$

where:

obs: If observation is not missing,

miss: If observation is missing.

Suppose that some elements of R data matrix are missing [7], then matrix M represents the matrix of variables that represent, variables indicating missing of all its values in the variables data matrix R. $M=\begin{cases} 0 \text{ if xij is missing value} \\ 1 \text{ if xij is not missing value} \end{cases}$

$x = (x_{\text{obs}}, x_{\text{miss}})$ can classify missing mechanisms as missing at random (MAR), missing completely at random (MCAR), and missing not at random (MNAR)) [9, 1, 10]

Assume that multiple linear regression at normal error is given by:

$\mathbf{Y}= \boldsymbol{X\beta}+ \boldsymbol{\varepsilon}$……… (1)

Y: Vector dependent variable with dimensions (n * 1).

β: Vector coefficients with dimension (P * 1).

X: Independent variable matrix with dimensions (n * P).

When missing value exists, depending on whether the value exists or not, then the regression equation can be divided into two parts [7]:

$$\begin{bmatrix} Y_{obs} \\ Y_{miss} \end{bmatrix} = \begin{bmatrix} \underline{1} & X_{obs} \\ \underline{1} & X_{miss} \end{bmatrix} \quad \underline{\beta} + \begin{bmatrix} \varepsilon_{obs} \\ \varepsilon_{miss} \end{bmatrix} \quad \text{........(2)}$$

Then we use least squares method to compute regression coefficients of y on x by using complete cases only:

$$\widehat{\boldsymbol{\beta}}= (\mathbf{XX'})^{-1} (\mathbf{X'Y}) \quad \text{…... (3)}$$

where x and y are complete cases [1, 11].

## 3- METHODS

### 3. 1 EM, MC-ECM, and ECME Algorithms

Because many algorithms cannot deal with missing values directly, data were converted to complete, either by compensation or deletion [7]. Among the estimation methods spread in statistical programs, EM algorithm is a complete method that depends on finding maximum likelihood estimation to complete data only. This method involves easy mathematical calculation and stable convergence and consists of the two steps of expectation (E-step) followed by maximum likelihood estimation (M-step) [8]. The steps of the algorithm are listed as follows:

i- Estimated value then complete Group in place missing value (estimation using uncritical means regression analysis and estimation using Monte Carlo Markov series…).

ii- Estimation of coefficients.

iii- Re-estimation of missing values after assuming that the extracted parameters are correct.

Let $\theta$ = ($\mu$, $\sum$) and let $\theta^t$ be a current estimate parameter $\theta$ for E-step (finding expected likelihood). If $\theta$ is $\theta^t$, then let $Q(\theta|\theta^t) = \int L(\theta|Y).f(y_{miss}|y_{obs}\theta = \theta^t)dy_{miss}$. ..….(4)

iv- Re-estimation of the parameter, then iterating continues for a while converging parameter [7].

$\mu^t \text{To} \mu^{t+1}$ and $\sum^t \text{to} \sum^{t+1}$ ……….(5)

$$\left|\underline{\hat{\mu}^{(t+1)}} - \underline{\hat{\mu}^{(t)}}\right| \langle \delta J \quad \text{...........(6)}$$

$$\left|\hat{\sum}^{(t+1)} - \hat{\sum}^{(t)}\right| \langle \delta J \quad \text{………...(7)}$$

where $\delta J$ is a very small number but bigger than zero. Let $Y_i$~N ($X_i\beta$, $\sum$) i = (1, 2..., n), where $X_i$ is defined as (L*p) matrix for the observation, (β) is a vector as (p*1) of unknown regression coefficients, and ($\sum$) is the variance-covariance matrix which has the dimensions (L*L) model (7) [4].

In linear regression, assume that ($\sum$) is unrestricted where $\sum$ is known as $\sum=\sum^t$ conditional maximum likelihood estimation of ß that could be obtained simply by least square estimation, as follows:

$$\beta^{t+1}= \{\textstyle\sum_{i=1}^{T} X_i^T (\sum^{(t)})^{-1} X_i\}^{-1} \{\sum_{i=1}^{T} X_i^T (\sum^{(t)})^{-1} Y_i\} \quad \text{........(8)}$$

Similarly, if $\beta=\beta^t$, then we can find the conditional maximum likelihood estimation of $\sum$ from the residual

$$\textstyle\sum^{t+1}= \frac{1}{n}\sum_{i=1}^{n} (y_i - x_i\beta^{t+1}) (y_i - x_i\beta^{t+1})^T \quad \text{………..(9)}$$

Note that, for conditional maximization, the log-likelihood function is increased for each (8) and (9).

That is, the values of β and $\sum$ for the frequency (t+1) are less than (t), as shown below.

$$L(\beta^t\textstyle\sum^t|y) \geq L(\beta^{t+1}\sum^t|y) \ldots\ldots \quad (10)$$

$L\ (\beta^{t+1}\sum^{t+1}|y) \geq L\ (\beta^{t+1}\sum^{t}|y).$ ……(11)

This leads to ECM algorithm in two steps, as those in the EM [10]. E–step is maintained whereas M–Step is replaced in several CM-steps (as in 11 and 12) to maximize CM-steps to find the expected complete-data [2]. CM-step is equal to S=2θ and is a conditional function in each S=2 when maximizing Q $(\theta|\theta^{t})$. Whereas $\theta_1 = \mu$ and $\theta_2 = \sum$, by a sequence of S conditional maximization steps over θ each maximizes θ function, which is defined as:

$Q\ (\theta|\theta^{(t)}) = \int L(\theta|x)f\,(x_{mis}|x_{obs}\theta = \theta^{(t)})dx_{mis} \dots$ ……(12)

$g_s$ (θ) (s=12… S) Modifies previous value [7]. MC-ECM is performed by inserting an E-step before each CM-step rather than before the set of CM-steps (i.e. using the estimated parameters for each iteration based on the next step and so on, until we have least possible difference between previous and subsequent parameters) [12]. The extension of each EM and ECM (which called ECME) is achieved by replacing sequence steps ECM with CM-steps, which leads to maximizing the corresponding constrained L function [9]. For convenience and it more generalizes then ECM call CMQ-step a step of maximizing a constrained Q function and by CML-step, step maximizing a constrained Q functions (i.e. either expected complete data log-likelihood or an observed data log-likelihood) [12]. ECME is characterized by a faster convergence of both of the two algorithms. This algorithm is simple to implement and works very well for the current problem. Implementation of this algorithm does not only require maximizing the Q function in relation to μ and ∑, but also maximizing the observed data log-likelihood for fixed μ and ∑, respectively, and estimating the parameters β [1].

### 3. 2 Recurrent Neural Networks

Neural networks are an attempt to build a machine that will mimic brain activities and be able to usually learn by examples [13], as shown in Figure-1.

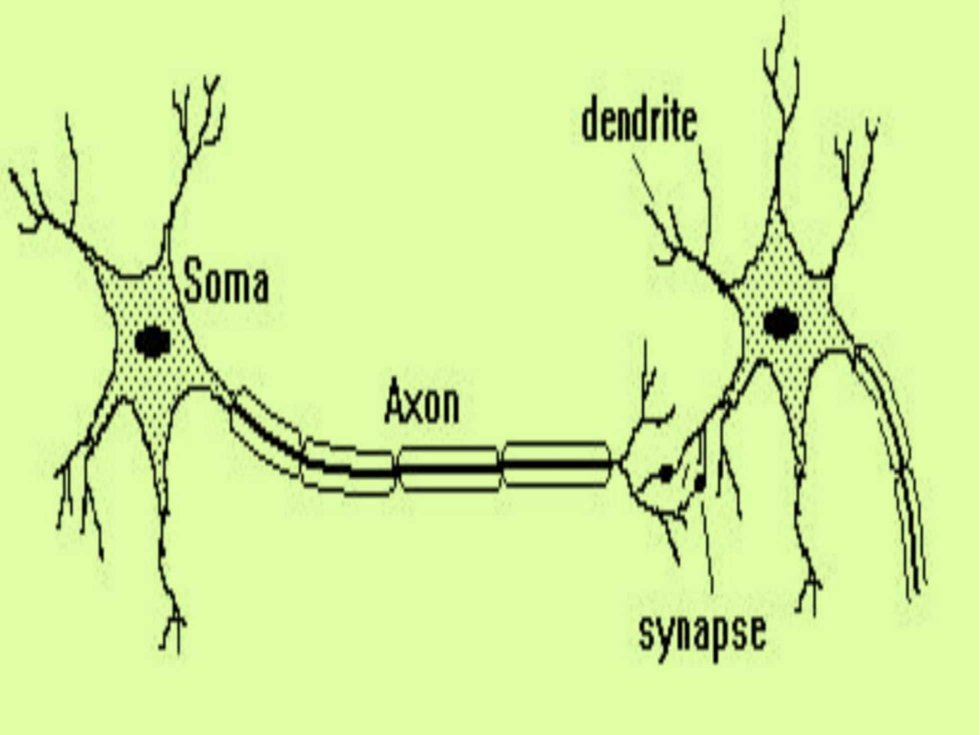

**Figure 1-**Shape of a neurons [12].

The aim of this approach is to enhance the processing in terms of speed, accuracy, and effectiveness in order to gives values approximating the real values [11]. For an estimate, parameters of multiple linear regression models for incomplete data-repeated neural networks were used. This method is generally composed of three layers; a first input layer, a second hidden layer which includes mathematical steps where data is processed in memory, and a third output layer. In the second layer, the missing values are compensated to achieve complete data usage by different estimation methods [14]. This is dependent on probability functions to compensate and adjust different weights between hidden nodes by deriving errors to model for each parameter Controlled preference adjusted weights by use errors in determining it [8]. The number of hidden nodes depends on the type of the problem to be studied and the amount of data which are relied on to improve network efficiency [15, 16]. After initial weights are determined randomly, they are divided into four sections:

i) Feed-forward computation.

ii) Back propagation to the output layer if the condition is met.

iii) Back propagation to the hidden layer if the condition is not met.

iv) Weights update.

The process is called real-time recurrent learning (RTRL), when the parameters are updated after a sequence of forward and backward passes is completed [11], such that:

RNN: X $\rightarrow$ F(X) $\rightarrow$ Y.

The idea behind continuously updating weights during neural network training, after each repetition, is to be able to obtain each time new weights which are better than the previous weight sets. The process continues until the goal is reached, which is to obtain weights that reduce errors, which can be written as in the following relation:

$W_j = \beta_j$

Weights used are based on the partial derivative parameters in linear regression formula or error square formula, which is recalculated continuously. For each row of complete data, changing all weights should be attempted simultaneously in which missing values are estimated in case that error box is larger than the less value [17, 18]:

$$W_j = \frac{\partial \varepsilon^2}{\partial \beta_j} = \frac{\partial}{\partial \beta_j}(y_i - \hat{y}_i)^2 = 2(y_i - \hat{y}_i)\frac{\partial\,(\,(y_i - \hat{y}_i)}{\partial \beta_j} \quad \text{……..(13)}$$

If neurons know xi is input and output it produces has $Y_j$ given Aims to for output unit the perception learning algorithm updates weights, according to the following equation.

$$W_{ij\,new} = W_{ijold} + C\,(Y_{ij} - \hat{Y}_{ij})\,X_{ij} \ldots \quad \text{…….(14)}$$

where C is a constant, called learning rate, which works on the product of activations. Specifically, weights update depends on a mistake Move it [19]. Weights are updated only if a value obtained is not correct, otherwise we move to the other row and so on while getting lower weights, which results in a min error $= \frac{1}{n}\sum_{i=1}^{n} e_i$ [5].

## 4- Simulation Results and Discussion

In this research work, we use sample sizes of n=25, 50, 100, and 500 parameters ($\beta_0, \beta_1, \beta_2, \beta_3$, respectively) of regression coefficient are imposed (1, 0. 25, 1. 2, 0. 5) and e ~ (0, $\sigma_\varepsilon^2$) ; $\sigma_\varepsilon^2$=0. 5, 1. 5, 2, which were generated using VBA program. Three independent variables were analyzed using Monte Carlo method according to the uniform distribution. The dependent variable was calculated as in Equation (15).

$$Y_i = 1X_{0i} + 0.25\,X_{1i} + 1.2\,X_{2i} + 0.5\,X_{3i} + e_i \quad i=1\ 2\ 3\ldots\ n. \quad \text{.......... (15)}$$

Let missing ratios (10%, 20%, and 30%), Missing value in exploratory variables according missing mechanism (MAR) and general pattern then applied (EM, MC-ECM, and ECME) algorithms and RNN with a frequency (500) iterations [7] Finally, calculate Mean Square error MSE for comparison then take average for all iterations and for all duplicates MMSE, as shown in Tables 1-3.

### 4. 1 Comparison of estimation methods when error = 0. 5

From the results shown in Table-1 and Figure- 2, it was found that, when the error variance is 0. 5, the algorithm ECME in a sample size of n=50 and 100 and missing ratios of 30% and 10%, respectively, had the lower value among the other methods. Likewise, at the sample size of n=100 and missing ratio of 30%, the MC-ECM algorithm was the best according to the results obtained from the simulation program the rest sample to missing ratios, RNN recorded the lower minimum mean square error values as compared to EM, MC-ECM, and ECME algorithms. That is, from the above, the RNN method is the best among the methods used in the research.

**Table 1**-Minimum mean squared errors for each method in 500 iterations classified according to sample size and ratio of missing value for simulation experiments when $\sigma_{\varepsilon}^{2}$ = 0. 5.

| Sample size | Missing rates | EM | MC-ECM | ECME | RNN |
|---|---|---|---|---|---|
| 25 | 10% | 4. 71E-02 | 1. 68E-01 | 6. 70E-02 | ∗2.39E−03 |
| | 20% | 4. 70E-02 | 1. 87E-01 | 6. 73E-02 | ∗2.47E−03 |
| | 30% | 2. 40E-02 | 1. 02E-01 | 3. 72E-02 | ∗1.32E−03 |
| 50 | 10% | 1. 19E-01 | 1. 07E-03 | 1. 07E-03 | ∗6.35E−04 |
| | 20% | 9. 12E-02 | 9. 49E-04 | 9. 67E-04 | ∗6.01E−04 |
| | 30% | 7. 73E-02 | 6. 63E-04 | ∗9.18E−04 | 9. 27E-04 |
| 100 | 10% | 5. 21E-02 | 5. 28E-04 | ∗5.26E−04 | 2. 57E-04 |
| | 20% | 5. 30E-02 | 4. 72E-04 | 4. 69E-04 | ∗2.34E−04 |
| | 30% | 2. 34E-04 | ∗1.51E−06 | 1. 75E-06 | 3. 04E-02 |
| 500 | 10% | 5. 29E-02 | 5. 41E-03 | 4. 32E-03 | ∗1.50E−05 |
| | 20% | 5. 30E-02 | 4. 35E-03 | 4. 34E-03 | ∗1.76E−05 |
| | 30% | 5. 31E-02 | 4. 38E-03 | 4. 37E-02 | ∗1.90E−05 |

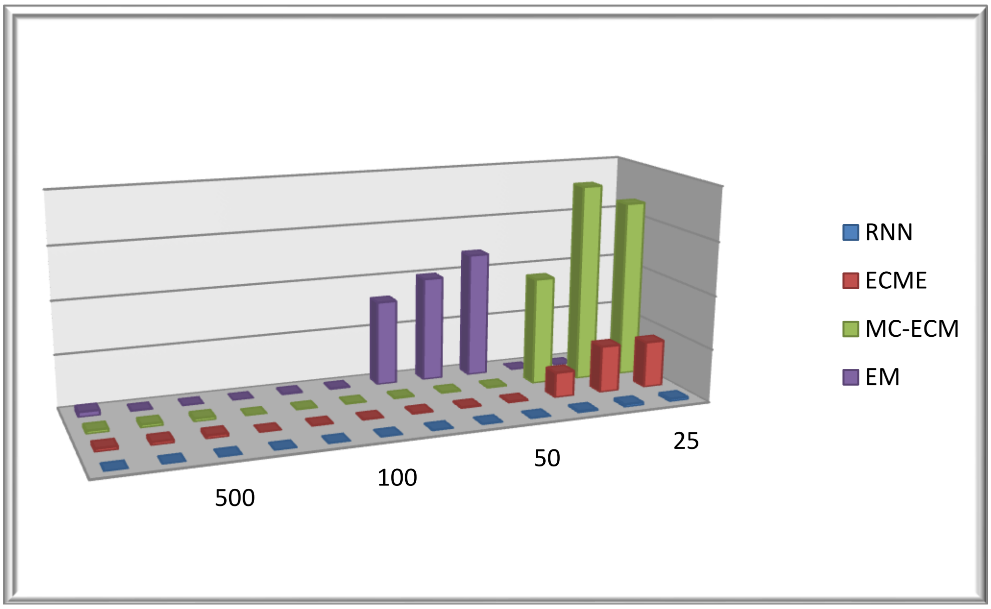

**Figure 2**-Minimum mean squared errors for all sample sizes and ratios of missing values when error = 0.5.

### 4. 2 Comparison of estimation methods when error = 1. 5 and 2

From the results shown by Tables- 2 and 3 and Figures- 3 and 4, it is evident that RNN is superior in large and medium sample sizes (n= 50, 100, and 500) when error variance values are equals to 1.5 and 2 and for all missing ratios (10%, 20%, and 30%). The best results were recorded using this algorithm, except for the error variance value of 1.5, sample size of n = 100, and missing ratio of 10%. The ECME algorithm recorded the lower mean square error value for small sample sizes (n=25). The EM algorithm had the best value for all missing ratios From the above. Overall, the results suggest that the RNN algorithm is superior to the other algorithms, followed by the EM algorithm.

**Table 2**-Minimum mean squared errors for each method in 500 iterations classified based on sample size and ratio of missing values for simulation experiments when $\sigma_{\varepsilon}^{2} = 1.5$.

| Sample size | Missing rates | EM | MC-ECM | ECME | RNN |
|---|---|---|---|---|---|
| 25 | 10% | ∗4.51E–04 | 1. 67E-01 | 4. 80E-02 | 2. 35E-03 |
| | 20% | ∗5.32E–04 | 1. 92E-01 | 4. 84E-02 | 2. 47E-03 |
| | 30% | ∗5.65E–04 | 2. 08E-01 | 5. 18E-02 | 2. 64E-03 |
| 50 | 10% | 1. 19E-01 | 1. 07E-03 | 1. 07E-03 | *6. 54E-04 |
| | 20% | 9. 92E-02 | 9. 51E-04 | 9. 69E-04 | *6. 49E-04 |
| | 30% | 8. 04E-02 | 9. 22E-04 | 9. 51E-04 | *6. 55E-04 |
| 100 | 10% | 1. 04E-03 | 5. 28E-04 | ∗5.27E–04 | 7. 90E-04 |
| | 20% | 9. 75E-06 | 2. 08E-06 | 1. 65E-06 | *5. 95E-07 |
| | 30% | 1. 88E-03 | 4. 72E-04 | 4. 69E-04 | *2. 77E-04 |
| 500 | 10% | 5. 29E-02 | 5. 41E-03 | 4. 32E-03 | *1. 53E-05 |
| | 20% | 5. 30E-02 | 4. 35E-03 | 4. 34E-03 | *1. 76E-05 |
| | 30% | 5. 31E-02 | 4. 38E-03 | 4. 37E-03 | *1. 90E-05 |

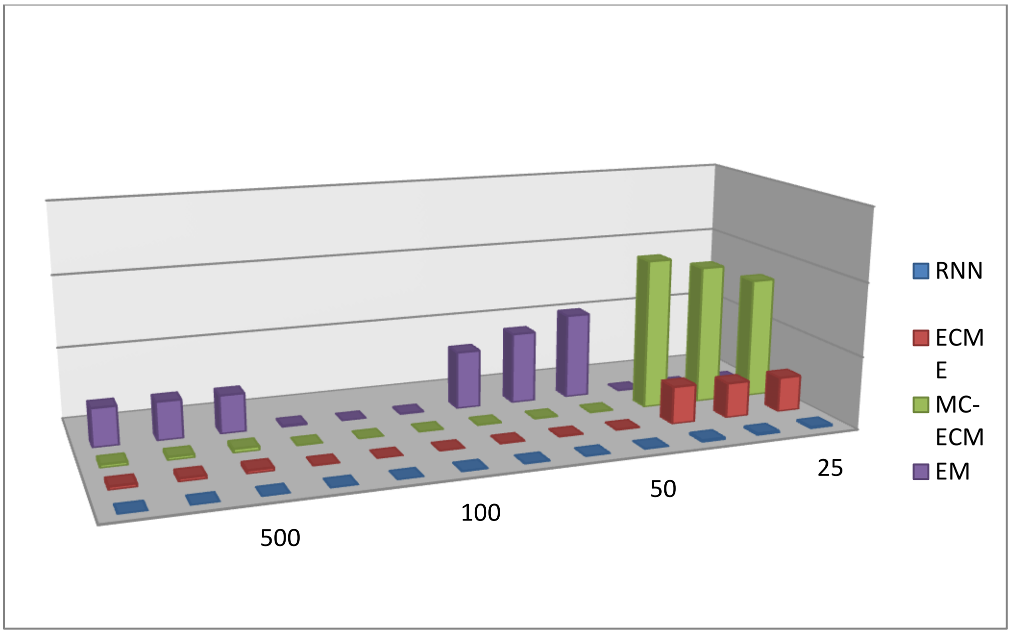

**Figure 3**-Minimum mean squared errors for each sample size and ratio of missing value when error= 1.5.

**Table 3**-Minimum mean squared errors for each method in 500 iterations classified based on sample size and ratio of missing value for simulation experiments when $\sigma_{\varepsilon}^{2} = 2$.

| Sample size | Missing rate | EM | MC-ECM | ECME | RNN |
|---|---|---|---|---|---|
| 25 | 10% | *2. 64E-04 | 1. 68E-01 | 4. 38E-02 | 2. 39E-03 |
| | 20% | *3. 38E-04 | 1. 87E-01 | 4. 41E-02 | 2. 47E-03 |
| | 30% | *1. 90E-04 | 1. 02E-01 | 2. 36E-02 | 1. 32E-03 |
| 50 | 10% | 1. 19E-01 | 1. 07E-03 | 1. 07E-03 | *5. 03E-04 |
| | 20% | 9. 92E-02 | 9. 51E-04 | 9. 69E-04 | *5. 07E-04 |
| | 30% | 8. 04E-02 | 9. 20E-04 | 9. 47E-04 | *4. 94E-04 |
| 100 | 10% | 8. 38E-04 | 5. 28E-04 | 5. 27E-04 | *2. 15E-04 |
| | 20% | 8. 38E-04 | 5. 28E-04 | 5. 26E-04 | *2. 15E-04 |
| | 30% | 9. 07E-06 | 2. 08E-06 | 1. 65E-06 | *6. 19E-07 |
| 500 | 10% | 2. 69E-04 | 2. 81E-03 | 2. 81E-03 | *1. 53E-05 |
| | 20% | 4. 23E-04 | 2. 84E-03 | 2. 83E-03 | *1. 76E-05 |
| | 30% | 4. 70E-03 | 2. 85E-03 | 2. 85E-03 | *1. 90E-05 |

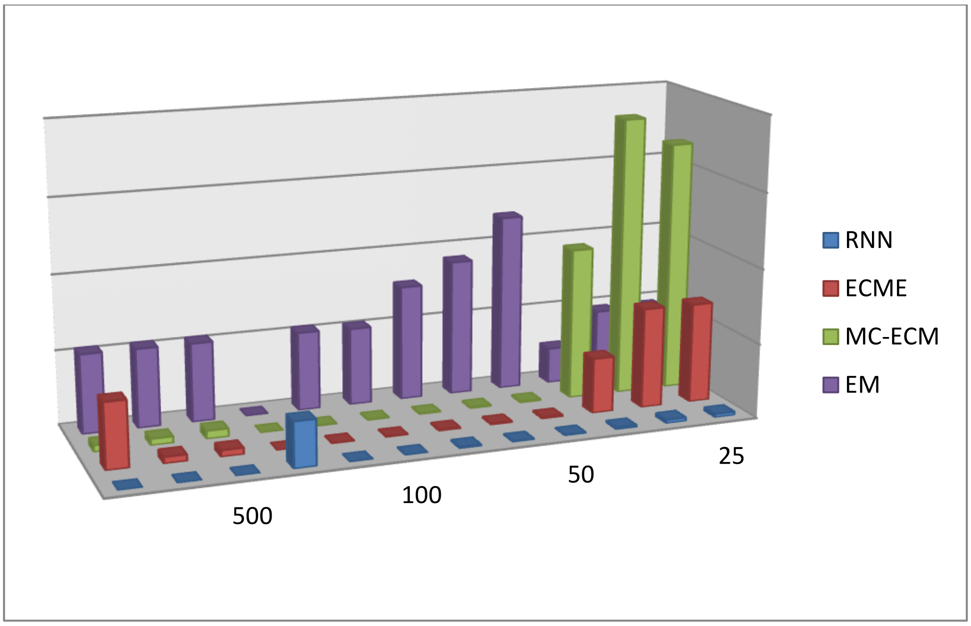

**Figure 4**-Minimum mean squared errors for each sample size and ratio of missing value when error = 2. * The minimum mean squared error for each percentage is missing.

**5- Conclusions**

Based on Tables-(1-3) and Figure-5, using MMSE for the regression models is superior to other algorithms when there is missing values in explanatory variables in large sample sizes (N=100, 500) and for all error variance and missing ratios. RNN achieved the best results in simulation, except for sample size of 100, ratio of missing values of 10%, and error variance of 0. 5 and 1. 5. ECME algorithm was the best besides MC-ECME algorithm when e=0. 5 and the ratio of missing values is 30% for medium sample size (n = 50). When the error variance is small (0. 5), and for all ratios of missing values, RNN showed lower efficiency than that of MSE. For the remaining values of error variance and for a ratio of missing values of 30%, ECME algorithm was superior for small sample size (n=25). EM algorithm outperformed the other algorithms for all ratios of missing values and error variance values of 1.5 and 2,. For small error variance (0. 5), RNN outperformed the other algorithms. We conclude that recurrent neural networks gave the best results in simulation, followed by EM algorithm for a small sample size.

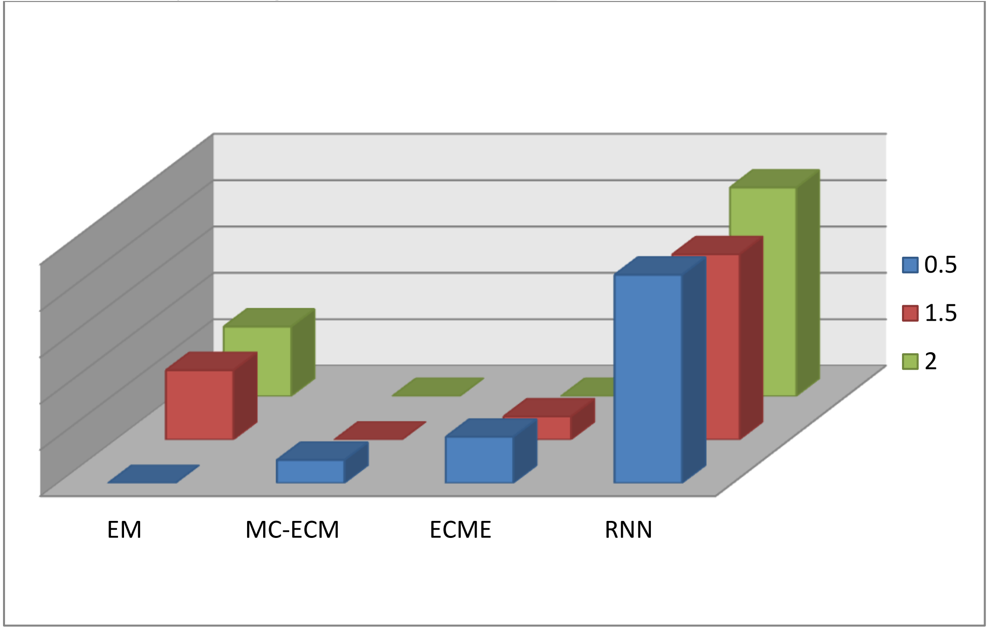

**Figure** 5-Frequency of the number of times each method shown according to the variance of error.

## References

**1.** Watanabe, W. and Yamaguchi, K. **2004**. *The EM Algorithm and Related Statistical Models* ; New York NYMarcel Dekker, Inc

**2.** Kariuki, shN., Gichuhi, AW. And Wanjoya, AK. **2015**. Comparison of Methods of Handling Missing Data: A Case Study of KDHS 2010 Data, *American Journal of Theoretical and Applied Statistics* ; **4**(3): 192-200

**3.** Liu, C.H. and Rubin, D. B. **1995**. ML Estimation of The t Distribution using EM and its extensions ECM and ECME, *statistica sinica*, **5**: 19-39

**4.** Bori M. S. **2013**. Dealing with missing data: Key assumptions and methods for applied analysis, Boston University, Technical Report No4

**5.** Harshanandgd. G. **2013**. Artificial Neural Network Method For Estimation Of Missing Data, *International Journal of Advanced Technology in Civil Engineering*, **2**(1).

**6.** Liu, CH. **1997**. ML Estimation of the Multivariate t Distribution and the EM Algorithm, *Journal of Multivariate Analysis Article*, **63**: 296-312

**7.** Suraphee S., Raksmanee Ch., Busaba J., Chaisorn Ch. and Nakornthai W. **2006**. A Comparison of Estimation Methods for Missing Data in Multiple Linear Regression with Two Independent Variables, *Thailand Statistician*; **4**:13-26.

**8.** Dekel, Oand Shamiro, O. **2008**. Learning to Classify with Missing and Corrupted Features, Proceedings of the 25 th International Conference on Machine Learning.

**9.** MENG, X-L. and RUBIN, DB. **1993**. Maximum likelihood estimation via the ECM algorithm: A general framework, *Biometrika*, **80**(2): 267-78.

**10.** Lopes N. and Ribeiro B. **2012**. Handling Missing Values Via A Neural Selective Input Model, *Neural Network World*, **4**(12): 357-370.

**11.** Ogunmolu, O., Gu, X., Jiang, S. and Gans, N. **2016**. Nonlinear Systems Identification Using Deep Dynamic Neural Networks, This work was supported by the Radiation Oncology Department UT Southwestern Dallas Texas USAarXiv:1610. 01439v1 [cs. NE] 5 Oct 2016.

**12.** Nitithumbundit Th. and Chan J. S. K. **2015**. An ECM algorithm for Skewed Multivariate Variance Gamma Distribution in Normal Mean-Variance, *Cornell university*, arXiv:1504. 01239v2 [stat. ME].

**13.** Cilimkovic M. **2010**. Neural Networks and Back Propagation Algorithm, *Institute of Technology Blanchardstown, Ireland*

**14.** Webb, S., Tom, T., The, YW. and MKumar, M. P. **2019**. A Statistical Approach To Assessing Neural Network Robustness, Published as a conference paper at ICLR 2019.

**15.** Bocco M., Willington E. and Arias M. **2010**. Comparison Of Regression And Neural Networks Models To Estimate Solar Radiation, *Chilean Journal Of Agricultural Research*, **70**(3): 428-435.

**16.** Abidin NZ. and Ismail A. R. Performance Analysis of Machine Learning Algorithms for Missing Value Imputation, *(IJACSA) International Journal of Advanced Computer Science and Applications*, **9**(6).

**17.** Smieja, M., Struski, L., Tabor, J., Zielinski, B. and Spurek, P. **2018**. Processing of missing data by neural networks Faculty of Mathematics and Computer Science, *Lojasiewicza,* **6**: 30-348.

**18.** Shiruru K. **2016**. Introduction To Artificial Neural Network, *International Journal Of Advance Research And Innovative Ideas In Education*, *IJARIIE-ISSN* (O) -2395-4396

**19.** Santos T. S., Mendes, D. and Torres R. R. **2016**. Artificial neural networks and multiple linear regression model using principal components to estimate rainfall over South America, *Nonlinear Processes Geophys*, **23**: 13–20.