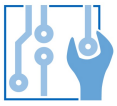


Lehrstuhl für Entwurfsautomatisierung
Technische Universität München
Prof. Dr.-Ing. Ulf Schlichtmann

Internship Report

Evaluation of Statistical Prediction for Non-Gaussian Distributions

Shuhang Zhang

December 22, 2017

Supervisor:
Grace Li Zhang

TABLE OF CONTENTS

1 Introduction

With MOSFET dimensions scaling down gradually, variations are becoming more pronounced, caused by various sources, such as aging, process, temperature, etc. Variations can cause different path delays in a circuit after manufacturing. To combat these challenges, serval methods were proposed. In [1], [2], a design-phase buffer insertion method was proposed to balance timing budgets of critical paths with their neighbors. In [4]-[9], these path delays are modeled as multivariate Gaussian distribution at the design phase.

In a circuit, the number of paths are very large. Therefore, it is expensive if all paths are monitored. The correlation between path delay distributions can be used to select representative paths. There exists work in which representative paths are selected to be monitored whereas other paths are estimated with statistical prediction.

In [3], since correlated variables are difficult to deal with, Principle Components Analysis (PCA) is introduced in multivariate correlated variables decomposition. In this proposed method, multivariate Gaussian distributions can be transformed into a linear combinations of independent univariate Gaussian distributions. In [4], Singular Value Decomposition and QR Decomposition with column pivoting (SVD-QRcp) are introduced for variable selection and prediction. In this method, the prediction is based on the theorem that conditional distributions of a multivariate Gaussian distribution are Gaussian distributions, so the predicted values are the conditional mean values. This overall flow, including PCA decomposition and SVD-QRcp prediction, works efficiently and has been examined in others' works [5]-[9].

Nevertheless, all the previous works are based on the assumption that multivariate correlated variables are multivariate Gaussian distributions. Gaussian distribution can bring us lots of simplifications, when analyzing these variables, because Gaussian distribution naturally does not have high moments. Therefore, this whole prediction flow has not been examined on non-Gaussian distributions yet. In this report, this algorithm is applied on different kinds of non-Gaussian distributions and the results are compared with those of Gaussian distributions separately.

The rest of this report is organized as follows. In Section 2, we give necessary background knowledge for this prediction algorithm. The experimental results are presented in Section 3. The conclusion is drawn in Section 4.

2 Background

In this section, basic knowledge about linear algebra and statistics is introduced. Additionally, some methods used in the next section are also explained here to avoid repeated explanation.

2.1 Multivariate statistics

Compared with univariate analysis, multivariate statistics considers the relations between variables. In this section, a detailed introduction of the multivariate Gaussian distribution and its conditional distributions are given. Furthermore, a general definition of distribution is also discussed.

2.1.1 Multivariate Gaussian Distribution

Multivariate Gaussian distribution is the general expression of the Gaussian distribution, which can be considered as an extension of the univariate Gaussian distribution. The probability density function can be expressed as,

$$f(x_1, x_2, ..., x_k) = \frac{exp(-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^T\boldsymbol{\Sigma}^{-1}(\boldsymbol{x}-\boldsymbol{\mu}))}{\sqrt{|2\pi\boldsymbol{\Sigma}|}} \tag{2.1}$$

where, $\boldsymbol{x}$ is the variable vector, $\boldsymbol{\mu}$ is the mean value vector and $\boldsymbol{\Sigma}$ is covariance matrix. In this report, we focus on the conditional variables, especially the conditional mean and variance, which are very important in predictions. Assume an N dimensional $\boldsymbol{x}$ can be separated into two groups.

$$\boldsymbol{x} = \begin{bmatrix} \boldsymbol{x_1} \\ \boldsymbol{x_2} \end{bmatrix} \text{ with sizes } \begin{bmatrix} q \times 1 \\ (N-q) \times 1 \end{bmatrix} \tag{2.2}$$

Correspondingly, $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$ can be expressed as follows.

$$\boldsymbol{\mu} = \begin{bmatrix} \boldsymbol{\mu_1} \\ \boldsymbol{\mu_2} \end{bmatrix} \text{ with sizes } \begin{bmatrix} q \times 1 \\ (N-q) \times 1 \end{bmatrix} \tag{2.3}$$

$$\boldsymbol{\Sigma} = \begin{bmatrix} \boldsymbol{\Sigma_{11}} & \boldsymbol{\Sigma_{12}} \\ \boldsymbol{\Sigma_{21}} & \boldsymbol{\Sigma_{22}} \end{bmatrix} \text{ with sizes } \begin{bmatrix} q \times q & q \times (N-q) \\ (N-q) \times q & (N-q) \times (N-q) \end{bmatrix} \tag{2.4}$$

Therefore, the conditional variables $\boldsymbol{x_{1|2}}$ satisfy Gaussian distribution, the mean and variance can be expressed as Equation 2.5 and Equation 2.6.

$$\boldsymbol{u_{1|2}} = \boldsymbol{u_1} + \boldsymbol{\Sigma_{12}\Sigma_{22}^{-1}}(\boldsymbol{x_2} - \boldsymbol{\mu_2}) \tag{2.5}$$

$$\boldsymbol{\Sigma_{11|22}} = \boldsymbol{\Sigma_{11}} - \boldsymbol{\Sigma_{12}\Sigma_{22}^{-1}\Sigma_{21}} \tag{2.6}$$

Proof:
Define:

$$\boldsymbol{z} = \boldsymbol{x_1} + \boldsymbol{A}\boldsymbol{x_2}, \tag{2.7}$$

where $\boldsymbol{A} = -\boldsymbol{\Sigma_{12}\Sigma_{22}^{-1}}$.

$$\begin{aligned} cov(\boldsymbol{z}, \boldsymbol{x_2}) &= cov(\boldsymbol{x_1}, \boldsymbol{x_2}) + cov(\boldsymbol{A}\boldsymbol{x_2}, \boldsymbol{x_2}) \\ &= \boldsymbol{\Sigma_{12}} + \boldsymbol{A}var(\boldsymbol{x_2}) \\ &= \boldsymbol{\Sigma_{12}} - \boldsymbol{\Sigma_{12}\Sigma_{22}^{-1}\Sigma_{22}} \\ &= \boldsymbol{0} \end{aligned} \tag{2.8}$$

Therefore $\boldsymbol{z}$ and $\boldsymbol{x_2}$ are uncorrelated.

$$\begin{aligned} E(\boldsymbol{x_1}|\boldsymbol{x_2}) &= E(\boldsymbol{z} - \boldsymbol{A}\boldsymbol{x_2}|\boldsymbol{x_2}) \\ &= E(\boldsymbol{z}|\boldsymbol{x_2}) - E(\boldsymbol{A}\boldsymbol{x_2}|\boldsymbol{x_2}) \\ &= E(\boldsymbol{z}) - \boldsymbol{A}\boldsymbol{x_2} \\ &= \boldsymbol{\mu_1} - \boldsymbol{A}(\boldsymbol{x_2} - \boldsymbol{\mu_2}) \\ &= \boldsymbol{\mu_1} + \boldsymbol{\Sigma_{12}\Sigma_{22}^{-1}}(\boldsymbol{x_2} - \boldsymbol{\mu_2}) \end{aligned} \tag{2.9}$$

$$
\begin{aligned}
var(\boldsymbol{x1}|\boldsymbol{x2}) &= var(\boldsymbol{z} - \boldsymbol{A}\boldsymbol{x_2}|\boldsymbol{x_2}) \\
&= var(\boldsymbol{z}|\boldsymbol{x_2}) + var(\boldsymbol{A}\boldsymbol{x_2}|\boldsymbol{x_2}) - \boldsymbol{A}cov(\boldsymbol{z}, -\boldsymbol{x_2}) - cov(\boldsymbol{z}, -\boldsymbol{x_2})\boldsymbol{A^T} \\
&= var(\boldsymbol{z}) \\
&= var(\boldsymbol{x_1}) + \boldsymbol{A}var(\boldsymbol{x_2})\boldsymbol{A^T} + \boldsymbol{A}cov(\boldsymbol{x_1}, \boldsymbol{x_2}) + cov(\boldsymbol{x_2}, \boldsymbol{x_1})\boldsymbol{A^T} \\
&= \boldsymbol{\Sigma_{11}} + \boldsymbol{\Sigma_{12}}\boldsymbol{\Sigma_{22}^{-1}}\boldsymbol{\Sigma_{22}}\boldsymbol{\Sigma_{22}^{-1}}\boldsymbol{\Sigma_{21}} - 2\boldsymbol{\Sigma_{12}}\boldsymbol{\Sigma_{22}^{-1}}\boldsymbol{\Sigma_{21}} \\
&= \boldsymbol{\Sigma_{11}} + \boldsymbol{\Sigma_{12}}\boldsymbol{\Sigma_{22}^{-1}}\boldsymbol{\Sigma_{21}} - 2\boldsymbol{\Sigma_{12}}\boldsymbol{\Sigma_{22}^{-1}}\boldsymbol{\Sigma_{21}} \\
&= \boldsymbol{\Sigma_{11}} - \boldsymbol{\Sigma_{12}}\boldsymbol{\Sigma_{22}^{-1}}\boldsymbol{\Sigma_{21}}
\end{aligned}
\tag{2.10}
$$

An interesting fact that can be derived from equation 2.8 is that $\boldsymbol{x_1} - \boldsymbol{\mu_1} - \boldsymbol{\Sigma_{12}}\boldsymbol{\Sigma_{22}^{-1}}(\boldsymbol{x_2} - \boldsymbol{\mu_2})$ and $\boldsymbol{x_2} - \boldsymbol{\mu_2}$ are independent, which builds the theory basis of the prediction algorithm. If some variables are fixed, then $\boldsymbol{x_2} = \boldsymbol{a}$, other variables' distributions are fixed, based on the value a. In this prediction algorithm, the predicted values are actually the conditional means of those variables.

2.1.2 General Representation of Distribution

To derive the general expression of any distribution, a new concept is introduced: moments. For probability on bounded interval, all the moments uniquely determine the distribution. The n-th moment of a distribution function $f(x)$ of a real value c can be expressed as,

$$\mu_n = \int_{-\infty}^{\infty} (x - c)^n f(x) dx. \tag{2.11}$$

In this report, we only consider the first four moments, mean, variance, skewness and kurtosis. The first and second moments are used to measure the average value and variations of a distribution. The third moment, skewness, is used to measure lopsidedness of the distribution and the fourth moment is for measuring the heaviness of the tail of a distribution. A probability distribution function, $f(x)$ can be expressed as,

$$f(x) = F(mean, deviation, skewness, kurtosis), \tag{2.12}$$

where $F(\,\cdot\,)$ is the function generating distribution based on input moments. These four moments are accurate enough for determining any distributions. Therefore, in our

case, any distribution can be transformed into a vector containing these four moments' values. Additionally, using the Pearson system provided by Matlab [10] can generate any distribution random numbers based on manually set four moments.

2.2 Priciple Components Analysis

Principle components analysis (PCA) is a procedure that transforms correlated variables into a linear combination of independent variables considered as principle components. In this procedure, the features contributing to variance most are kept, and so PCA only considers variance and can not handle higher moments. Therefore, PCA is often used in dimension reduction of data set based on variance information. In this report, PCA is widely adapted for the decorrelation of different multivariate distributions using a covariance matrix.

Algorithm 1 PCA

Require: $M, Ratio_{th}$
Ensure: F
1: $N \leftarrow$ Number of rows/columns in M
2: $[V, D] = EIG(M)$
3: $F = V * sqrt(D)$
4: $N_{valid} \leftarrow 1$;
5: **while** $\sum_{i=1}^{N_{valid}} D(i,i) < Ratio_{th} * \sum_{i=1}^{N} D(i,i)$ **do**
6: $N_{valid} \leftarrow N_{valid} + 1$
7: **end while**
8: $F = F(:, 1 : N_{valid})$
9: **return** F

As shown in Algorithm 1, the PCA algorithm only takes two input values, covariance matrix M and $Ratio_{th}$ used for deciding number of valid eigen values. First, eigen decomposition is performed on covariance matrix, which can be expressed as

$$M = V * D * V^{-1} \tag{2.13}$$

where eigenvector matrix V is an $N \times N$ square matrix and matrix D contains the eigenvalue information. For matrix D, eigenvalues are allocated with non-increasing order at the diagonal line. Afterwards, a complete feature matrix can be obtained, but some of the eigenvalues are small and can be ignored. Therefore, a $Ratio_{th}$ is

introduced and set to 0.99 in the experiments. Based on this threshold value, N_{valid} can be decided and then the valid feature matrix by selecting the left N_{valid} columns, since eigenvalues are in non-increasing order, and correspondingly the eigenvectors become less important from left to right.

2.3 SVD-QRcp Algorithm

Algorithm 2 SVD-QRcp Method

Require: $P, Ratio_{th}$
Ensure: P_R, N_R
$N \leftarrow$ Number of rows in P
$[U, \Sigma, V] = SVD(P)$;
$S = diag(\Sigma)$
$N_R \leftarrow 1$;
while $\sum_{i=1}^{N_R} S(i)^2 < Ratio_{th} * \sum_{i=1}^{N} S(i)^2$ **do**
    $N_R \leftarrow N_R + 1$
end while
Select first N_R columns in U, $U_R = U(:, 1 : N_R)$
$[Q, R, \Pi] = QR(U_R^T)$
$P_n = \Pi^T P$;
$P_R = P_n(1 : N_R, :)$
return P_R, N_R

The SVD-QRcp method selects representative variables, P_R, and P_R can be used to predict the values of all other variables. The predicted values $d_{predict}$ can be represented as

$$d_{predict} = PP_R^T(P_R P_R^T)^{-1}(d_R - \mu_R), \tag{2.14}$$

where P denotes unselected variables, P_R means representative variables, D_R denotes real values of representative variables, μ_R represents the mean values of representative variables and $()^{-1}$ represents matrix inverse calculation.
In Algorithm 2, the whole flow of SVD-QRcp method is shown. To select representative variables, the whole set P, M by N matrix, is decomposed into product of three matrices, which can be expressed as

$$P = U\Sigma V^T, \tag{2.15}$$

where matrix $U \in \mathbb{R}^{M \times M}$ and matrix $V \in \mathbb{R}^{N \times N}$ are orthogonal matrices, and Σ is singular value matrix, in which sigular value are placed at diagonal line with non-increasing order. The SVD factorization implies the real rank of decomposed matrix and reveals the importance at different directions by comparing the singular values, which indicates that some small singular values can be ignored with little accuracy loss. Based on this property, the number of selected variables, N_R, can be even smaller than $rank(P)$, so the real N_R can be much smaller than the original number of variables, N.

In order to determine which singular value should be ignored, a threshold ratio is introduced and the square of each singular value can be considered as energy in each direction. Therefore, the ratio of energy can be expressed as

$$Ratio = \frac{\sum_{i=1}^{N_R} \sigma_i^2}{\sum_{i=1}^{N} \sigma_i^2}, \tag{2.16}$$

where N_R is the number of selected variables and σ_i represents one singular value. In this work, $Ratio_{th}$ is set to 0.99, where the N_R representative features can represent the whole feature matrix.

Once N_R is determined, the next step is acquiring the left N_R columns of matrix U and applying QRcp to the transpose of matrix U_R. The QR decomposition can be expressed as

$$U_R^T = QR\Pi^T, \tag{2.17}$$

where matrix Q is unitary, R is an upper triangular matrix, Π is a column permutation matrix implying the importance order of the original feature matrix. By applying permutation matrix to original feature matrix, the rows which are more important are moved to the top. One thing to be noticed is that the original permutation matrix Π is used for column pivoting, so the transpose of matrix Π should be used for row pivoting. After row pivoting, the N_R representative rows are allocated in the first N_R rows.

In conclusion, the SVD-QRcp method consists of two phases: SVD and QRcp. SVD implies the real N_R and QRcp reorders the feature matrix in decreasing order of importance.

2.4 Coupula Method

In this section, we introduce some methods to generate multivariate correlated random numbers. The Coupula method, proposed in [11], can generate any multivariate correlated random numbers.

Algorithm 3 Dependent Random Number Generation

Require: $N_{samples}, \mathbf{\Sigma}$
Ensure: $\boldsymbol{M}$
1: $\boldsymbol{Z} \leftarrow mvnrnd(\mathbf{0}, \mathbf{\Sigma}, N_{samples})$
2: $\boldsymbol{U} \leftarrow normcdf(\boldsymbol{Z})$
3: $\boldsymbol{M} \leftarrow cdfinv(\boldsymbol{U})$
4: **return** $\boldsymbol{M}$

As shown in Algorithm 3, using Coupula method to generate any combination of distributions with given correlation matrix $\mathbf{\Sigma}$ is illustrated. Firstly, $N_{samples}$ determines the number of sets of generated random number, and generated random numbers satisfy the correlation matrix $\mathbf{\Sigma}$. This random number generation algorithm generates correlated multivariate Gaussian random number with zero mean and given covariance matrix ($\mathbf{\Sigma}$) and then those random numbers are transformed to interval $[0, 1]$, using *normcdf* function. In the previous step, those random numbers have been transformed into multidimentional $[0, 1]$ uniform distribution. Then, by using specific cdf inverse functions, multivariate correlated random numbers can be obtained. The multivariate distribution type depends on the cdf inverse functions.

3 Prediction for Non-Gaussian Distributions

In this section, the selection and prediction methods of variables are applied to non-Gaussian distributions. The prediction results of three different non-Gaussian distributions, uniform, exponential and skew normal are compared with those of Gaussian distribution separately. We first start from a scenario where there are only two variables and then multivariate cases can be analyzed similarly.

3.1 Overall flow

The overall flow is shown in Fig 3.1. Firstly, we need to generate random numbers using Copula method, based on the given parameters, e.g. lower bound, upper bound, λ. Then, the mean values, correlation matrix and covariance matrix can be calculated based on these samples and these parameters are used to generate random numbers of non-Gaussian and Gaussian distributions, which ensures the consistency of first and second moments of all distributions. Afterwards, PCA introduced in the previous section is applied on the covariance matrix to convert a set of correlated variables into a set of uncorrelated variables. With PCA, so a feature matrix is obtained based on the effective eigenvalues and eigenvectors of the covariance matrix. Next, the feature matrix is fed to SVD-QRcp method, so all variables can be separated into two groups, representative variables and predicted variables. Because the non-Gaussian distribution and Gaussian distribution have the same covariance matrix, the variable selection is the same. Lastly, using the conditional mean formula explained in the previous section, $d_{predict,k} = \mu_k + \Sigma_{k,t}\Sigma_t^{-1}(d_{real,t} - \mu_t)$, the absolute gap can be expressed as, $\Delta = |d_{predict,k} - d_{real,k}|$, where $d_{predict,k}$ is calculated using that formula and $d_{real,k}$ is the real sample value. Now, we have absolute gap information from all distributions, so the efficiency of PCA and SVD-QRcp algorithm under different distributions can be compared. In the following cases, we only consider,

$$\Delta_{mean} = \frac{1}{N_{samples} \times N_{predict}} \sum_{i=1}^{N_{samples}} \sum_{j=1}^{N_{predict}} \Delta_{i,j}, \tag{3.1}$$

$$\Delta_{max} = max(\Delta_{i,j}), i \in [1, N_{samples}], j \in [1, N_{predict}], \tag{3.2}$$

and the number of points (Inaccuracy Points) whose values are bigger than $0.8\Delta_{max}^{normal}$.

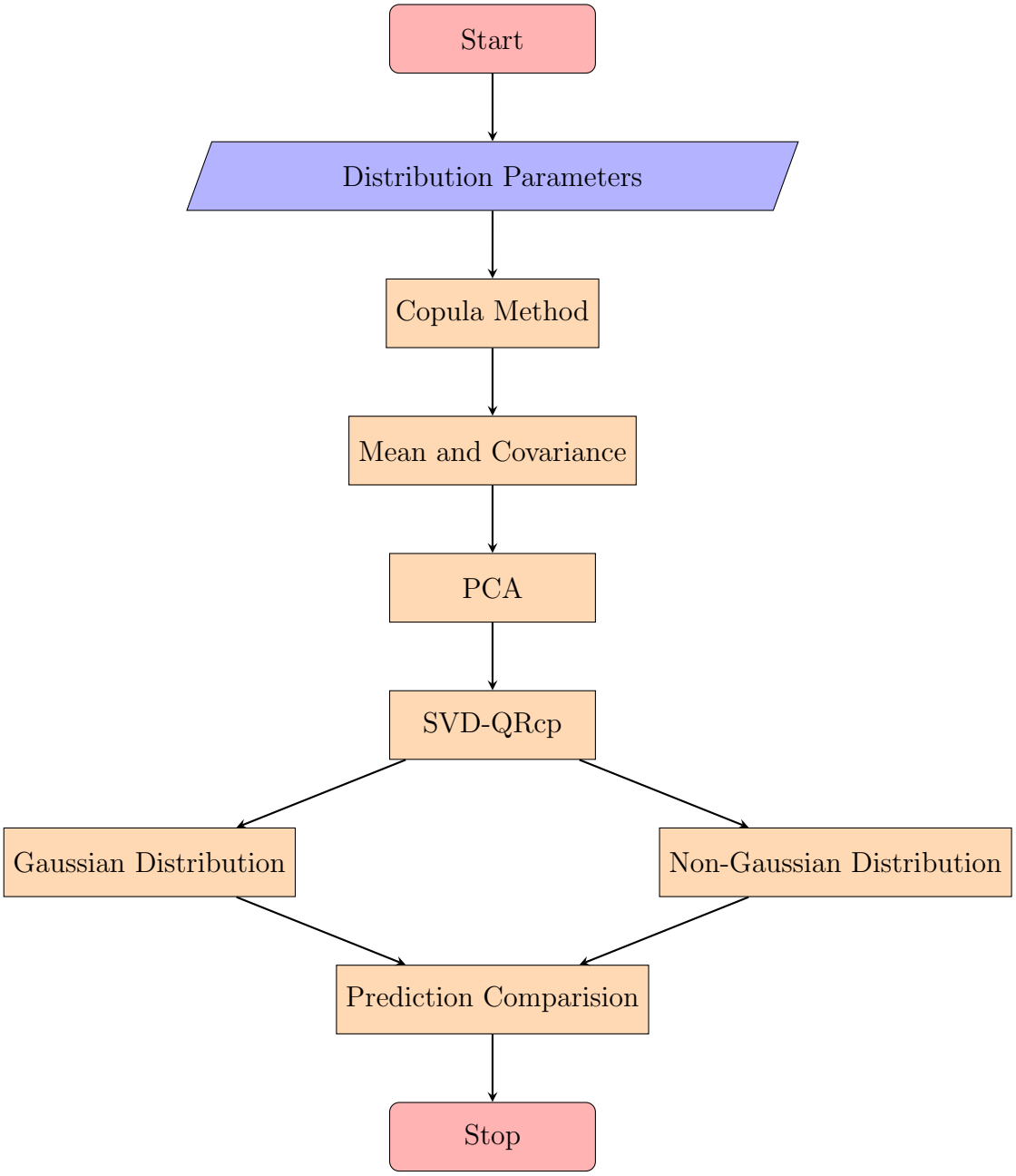


Fig. 3.1: Overall flow.

3.2 Bi-correlated Variables

To simplify the bivariate case, we just use one variable to predict the other one, based on the conditional mean formula, $d_{predict,k} = \mu_k + \Sigma_{k,t}\Sigma_t^{-1}(d_{real,t} - \mu_t)$.

3.2.1 Uniform and Gaussian Distributions

As discussed in Section 2, the mean and variance of uniform distribution are determined by its lower and upper bounds. The correlation between two variables can be set manually. In our case, we set the lower bound, -100 and upper bound, 100, so the mean equals 0 and variance equals 3333. For the correlation of the two variables, three values are used, $\rho = 0, 0.5$ and 0.9, to demonstrate the effects resulted from different correlation. In Fig 3.2, histograms of each distribution with 100,000 samples under given parameters are presented.

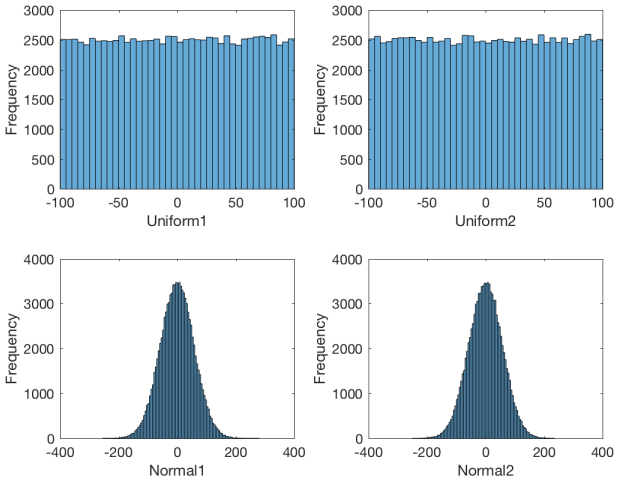


Fig. 3.2: Uniform and normal samples distributions.

3.2.1.1 Theoretical Analysis of Prediction

Firstly, we can start from $\rho = 0$ and calculate the theoretical absolute gap between predicted and sampled values. Δ_{mean}, the mean gap is defined as the average gap among all samples. The maximum gap Δ_{max} is defined as the maximum gap among all samples. In this section, we only consider the scenario when $\rho = 0$, so this equation

can be simplified to $d_{predict,k} = \mu_k$. Therefore, the theoretical absolute gap mean of uniform distribution and Gaussian distribution can be obtained as follows.
Absolute gap for Δ_{mean} with uniform distribution:
Given: a uniform distribution,

$$x \sim U(a, b), \tag{3.3}$$

where a is the lower bound and b is the upper bound.
We use

$$\int_a^b |x - \frac{a+b}{2}| \cdot \frac{1}{b-a} dx, \tag{3.4}$$

to calculate the mean value of absolute gap Δ_{mean}.

$$\int_a^b |x - \frac{a+b}{2}| \cdot \frac{1}{b-a} dx \tag{3.5}$$

$$\Rightarrow \int_a^{\frac{a+b}{2}} (\frac{a+b}{2} - x) \cdot \frac{1}{b-a} dx + \int_{\frac{a+b}{2}}^b (x - \frac{a+b}{2}) \cdot \frac{1}{b-a} dx \tag{3.6}$$

$$\Rightarrow \frac{1}{2}x\Big|_a^{\frac{a+b}{2}} - \frac{1}{2} \cdot \frac{1}{b-a} \cdot x^2\Big|_a^{\frac{a+b}{2}} + \frac{1}{2} \cdot \frac{1}{b-a} \cdot x^2\Big|_{\frac{a+b}{2}}^b - \frac{1}{2}x\Big|_{\frac{a+b}{2}}^b dx \tag{3.7}$$

$$\Rightarrow \frac{1}{4}(b-a) \tag{3.8}$$

Absolute gap for Δ_{mean} with Gaussian distribution:
Given: a Gaussian distribution with mean, μ, and variance, σ^2.

$$x \sim \mathcal{N}(\mu, \sigma^2) \tag{3.9}$$

We use

$$\int_{-\infty}^{\infty} |x - \mu| \frac{1}{\sigma\sqrt{2\pi}} e^{-(x-\mu)^2/2\sigma^2} dx, \tag{3.10}$$

to calculate the mean value of absolute gap Δ_{mean}.

$$\Rightarrow \int_{-\infty}^{\infty} |t| \frac{1}{\sigma\sqrt{2\pi}} e^{-(t)^2/2\sigma^2} dt \qquad //x - \mu \text{ is replaced with } t. \tag{3.11}$$

$$\Rightarrow t \sim \mathcal{N}(0,\, \sigma^2) \qquad //\text{Based on previous formula.} \tag{3.12}$$

$$\Rightarrow \int_{0}^{\infty} t \frac{\sqrt{2}}{\sigma\sqrt{\pi}} e^{-(t)^2/2\sigma^2} dt \qquad //\text{Symmetry on support.} \tag{3.13}$$

$$\Rightarrow \frac{\sqrt{2}\sigma}{\sqrt{\pi}} \qquad //\text{Mean of half normal distribution is known.} \tag{3.14}$$

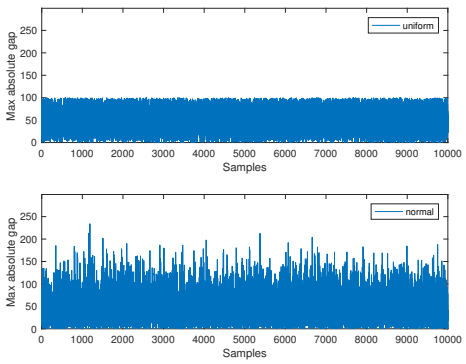
$$\Rightarrow \frac{\sqrt{2}\sqrt{\frac{(b-a)^2}{12}}}{\sqrt{\pi}} \qquad //\text{Mean of half normal distribution is known.} \tag{3.15}$$

We know that σ is determined by a and b, and since $\sigma = \sqrt{\frac{(b-a)^2}{12}}$, so the absolute gap mean of Gaussian distribution is always smaller than the one of uniform distribution. In the following part, we will compare the theoretical absolute gap of maximum values with both distributions. For uniform distribution, Δ_{max} is easy to obtained when $\rho = 0$, $\Delta_{max} = \frac{b-a}{2}$. The predicted value is always the mean, so maximum absolute gap equals half of its valid domain. For Gaussian distribution, Δ_{max} could be ∞, but with extreme low probability. In the experimental part, we will see a large value, but it is still acceptable.

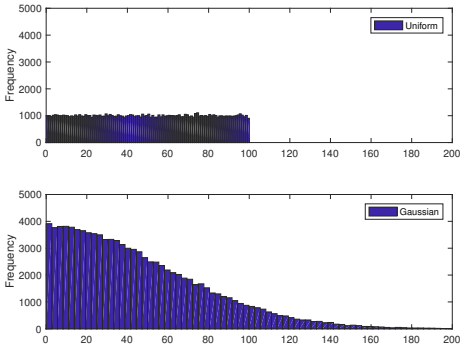
3.2.1.2 Experimental Results

In this section, bivariate cases with different correlation matrices are examined in Matlab and results are shown in the following paragraphs.

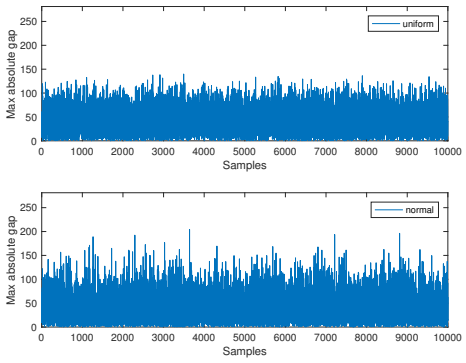
Fig 3.3 shows the absolute gap information with different correlation values. Fig. 3.3 (a), (c) and (e), show the maximum absolute gap values in all samples. The upper one shows Δ_{max} of uniform distribution and the bottom one shows that of Gaussian



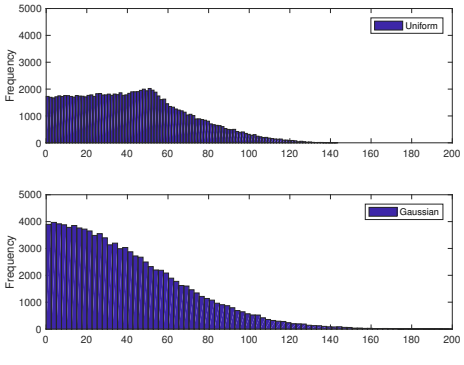
(a) Max gap when $\rho = 0$.



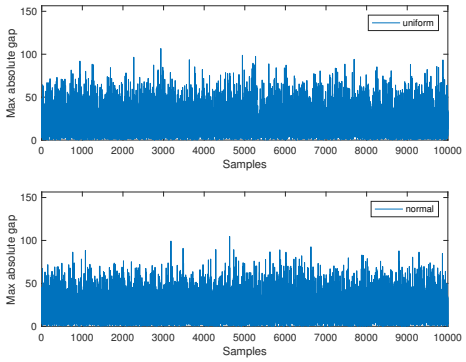
(b) Gap distribution when $\rho = 0$.



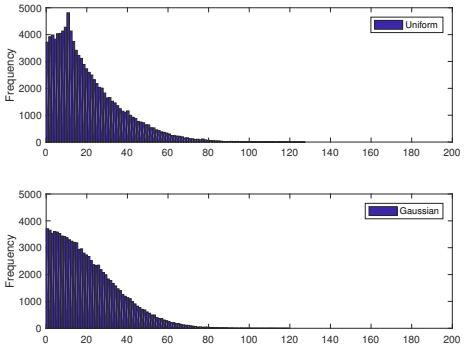
(c) Max gap when $\rho = 0.5$.



(d) Gap distribution when $\rho = 0.5$.



(e) Max gap when $\rho = 0.9$.



(f) Gap distribution when $\rho = 0.9$.

Fig. 3.3: Absolute gap comparisions between uniform and Gaussian distributions.

distribution. Fig 3.3 (b), (d) and (f), show the distributions of absolute gap. The figures on the top are the absolute gap distribution of uniform distribution and the bottom one is that of Gaussian distribution. Based on this figure, the absolute gap mean and maximum decrease gradually with the increase of ρ, The higher correlation coefficient is, the more accurate prediction is.

	$\rho = 0$	$\rho = 0.5$	$\rho = 0.9$
$\Delta_{mean}^{uniform}$	49.9679	42.8226	20.4755
Δ_{mean}^{normal}	45.9321	40.3415	20.8877
$\Delta_{max}^{uniform}$	100.0635	143.5592	127.5513
Δ_{max}^{normal}	277.3576	251.1106	119.6220
$\#Inaccuracy\_Points(uniform)$	0	0	86
$\#Inaccuracy\_Points(normal)$	9	12	25

Tab. 3.1: Prediction comparision between uniform and normal distributions.

Table 3.1 shows the key specifications with different ρ values. In the case of uniform distribution, it should be noted that, the absolute gap mean value of uniform distribution is smaller than that of Gaussian distribution and the maximum absolute gap value is greater than that of Gaussian distribution. This scenario can be explained, that with the increasing of correlation ρ, the conditional deviation is decreasing much faster than that of uniform distribution, so after around $\rho = 0.81$, we can expect an opposite result compared with the result we proved in previous section with $\rho = 0$.

3.2.2 Exponential and Gaussian Distributions

As discussed in Section 2, the mean and variance of exponential distribution is determined by λ. In our case, we set $\lambda = 0.1$, so the mean equals 10 and variance equals 100. In Fig 3.4, the distributions of each variable by sampling are presented. In this case, we use two correlated exponential distributions and set the correlation matrix individually. Three different values of correlation coefficient are used.

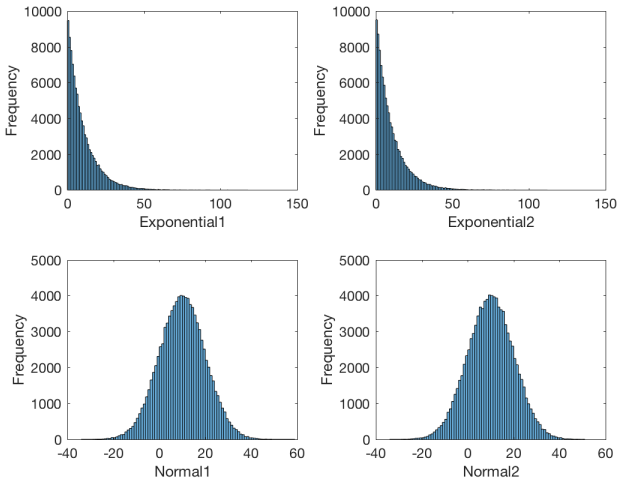


Fig. 3.4: Exponential and normal samples distributions.

3.2.2.1 Theoretical Analysis of Prediction

As presented in previous section, we start from calculating theoretical absolute gap mean, Δ_{mean} and maximum values, Δ_{max}, among all samples, with $\rho = 0$. The theoretical mean value is shown as following. Given: x satisfies exponential distribution.

$$x \sim \lambda e^{-\lambda x} \tag{3.16}$$

We use

$$\int_0^\infty |x - \lambda^{-1}| \cdot \lambda \cdot e^{-\lambda x} dx, \tag{3.17}$$

to calculate absolute gap mean, Δ_{mean}.

$$\Rightarrow \int_0^{\lambda^{-1}} (\lambda^{-1} - x) \cdot \lambda \cdot e^{-\lambda x} dx + \int_{\lambda^{-1}}^\infty (x - \lambda^{-1}) \cdot \lambda \cdot e^{-\lambda x} dx \tag{3.18}$$

$$\Rightarrow -\int_0^{\lambda^{-1}} \lambda \cdot x \cdot e^{-\lambda x} dx + \int_0^{\lambda^{-1}} e^{-\lambda x} dx - \int_{\lambda^{-1}}^\infty e^{-\lambda x} dx + \int_{\lambda^{-1}}^\infty \lambda \cdot x \cdot e^{-\lambda x} dx \tag{3.19}$$

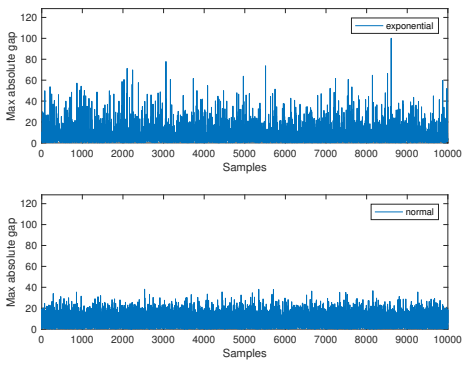
$$\Rightarrow \frac{1}{\lambda} e^{-\lambda x}(\lambda x + 1)\Big|_0^{\lambda^{-1}} - (\frac{1}{\lambda} e^{-\lambda x})\Big|_0^{\lambda^{-1}} + (\frac{1}{\lambda} e^{-\lambda x})\Big|_{\lambda^{-1}}^\infty - (\frac{1}{\lambda} e^{-\lambda x})(\lambda x + 1)\Big|_{\lambda^{-1}}^\infty \tag{3.20}$$

$$\Rightarrow \frac{2}{\lambda}e^{-1} \tag{3.21}$$

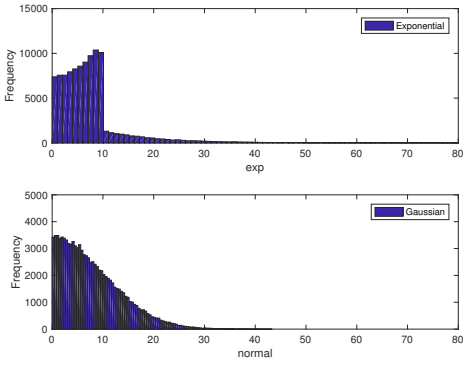
According to the calculated result, the absolute gap mean of exponential distribution is smaller than that of Gaussian distribution, with the result equals $\frac{\sqrt{2}}{\sqrt{\pi}\lambda}$.
For the maximum absolute value, both maximum values of exponential and normal distributions have the extreme low probability to be infinity. Theoretically, both do not have too much difference, but the experimental result shows that the normal distribution is much better than exponential distribution.

3.2.2.2 Experimental Results

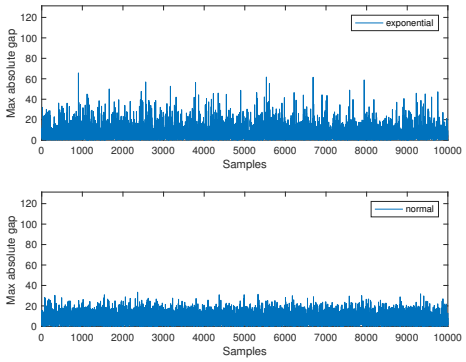
Fig 3.5 shows the absolute gap information of exponential distribution and Gaussian distribution.
With the increase of the correlation coefficient, the absolute gap becomes smaller. From Fig 3.5 (a), (c) and (e), it is obvious that maximum absolute gaps in exponential distribution are much higher than that of Gaussian distribution. From Fig 3.5 (b), (d) and (f), we can observe more inaccuracy points (bigger than $0.8\Delta_{max}^{normal}$) exist in exponential distribution.
Table 3.2 shows the Δ_{mean}, Δ_{max}, and inaccuracy points with different correlation coefficients. This algorithm works well when correlation coefficient increases gradually. In this case, we should pay more attention to these inaccuracy points. The number of inaccuracy points in exponential case are extremely larger than the number in Gaussian case, which means prediction works worse than in Gaussian case.



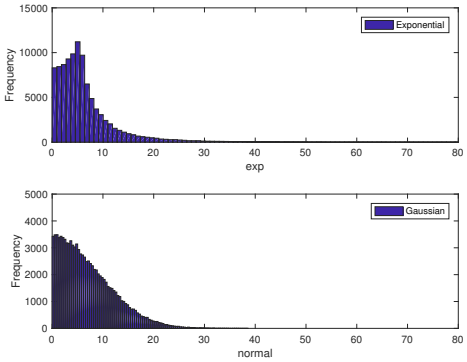
(a) Max gap when $\rho = 0$.



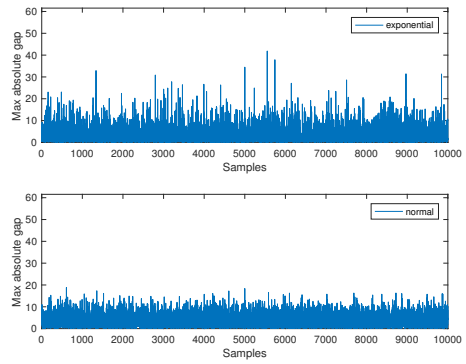
(b) Gap distribution when $\rho = 0$.



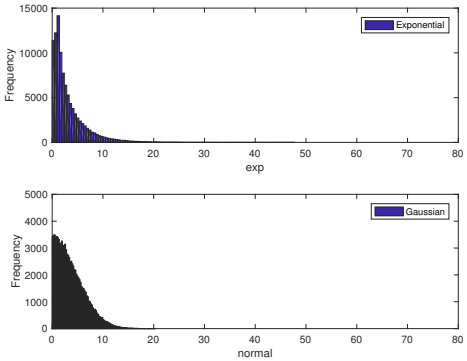
(c) Max gap when $\rho = 0.5$.



(d) Gap distribution when $\rho = 0.5$.



(e) Max gap when $\rho = 0.9$.



(f) Gap distribution when $\rho = 0.9$.

Fig. 3.5: Absolute gap comparisions between exponential and Gaussian distributions.

	$\rho = 0$	$\rho = 0.5$	$\rho = 0.9$
$\Delta_{mean}^{exponential}$	7.4102	6.4331	3.1883
Δ_{mean}^{normal}	8.0258	7.1061	3.7321
$\Delta_{max}^{exponential}$	117.0365	115.7590	51.7589
Δ_{max}^{normal}	46.0854	41.1700	21.1563
$\#Inaccuracy\_Points(exponential)$	974	883	858
$\#Inaccuracy\_Points(normal)$	28	27	33

Tab. 3.2: Prediction comparision between exponential and normal distributions.

3.2.3 Skew Normal and Gaussian Distributions

As introduced in Section 2, the skew normal distribution is determined by mean, variance and shape modification parameter δ. In our case, we set mean equals 100, variance equals 210 and $\delta = -0.99$.

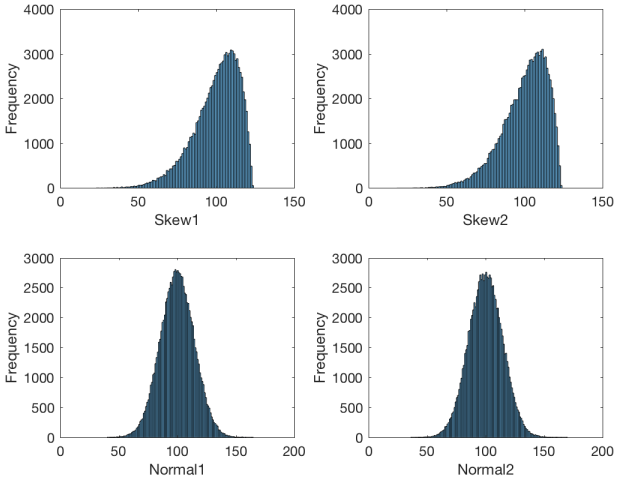
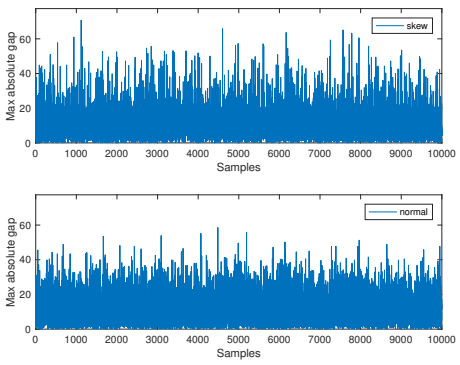
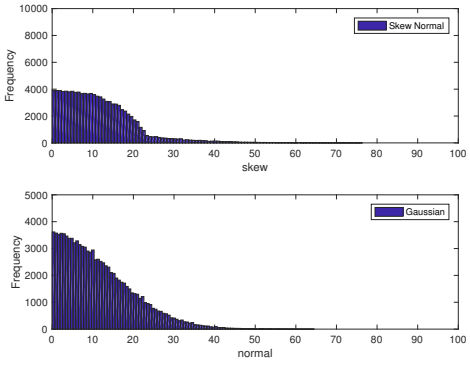


Fig. 3.6: Skew normal and normal samples distributions.

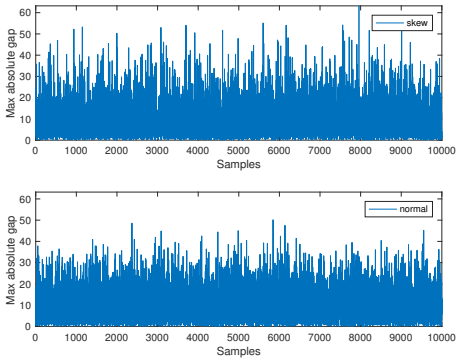
In Fig 3.6, the distributions of each variable by sampling are presented. In this case, we use two correlated left skew normal distributions and set the correlation matrix individually. The correlated random numbers are also generated by Copula method. As previous section, three different values of correlation coefficients are experimented.



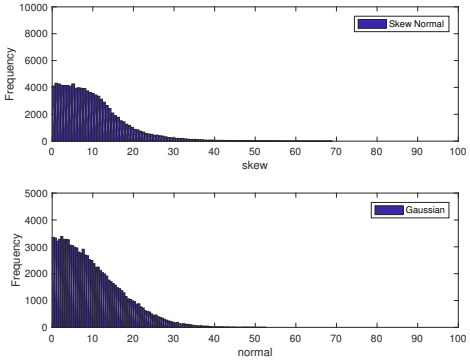
(a) Max gap when $\rho = 0$.



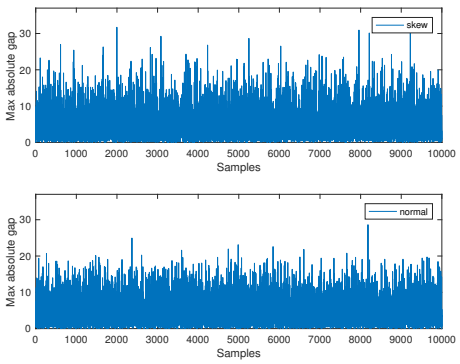
(b) Gap distribution when $\rho = 0$.



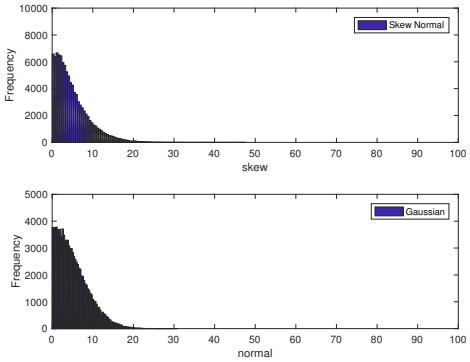
(c) Max gap when $\rho = 0.5$.



(d) Gap distribution when $\rho = 0.5$.



(e) Max gap when $\rho = 0.9$.



(f) Gap distribution when $\rho = 0.9$.

Fig. 3.7: Absolute gap comparision between skew normal and Gaussian distributions.

3.2.3.1 Experimental Results

In Fig 3.7, the absolute gap mean, Δ_{mean}, with different correlation coefficients are shown. Fig 3.7 (a), (c) and (e) illustrate the maximum value of each sample. The

upper one represents maximum absolute gap of skew normal distribution and the bottom one represents that of Gaussian distribution. It is obvious that the maximum values of skew normal distribution are larger than that of the normal distribution. Fig 3.7 (b), (d) and (f) illustrate the absolute gap distributions with different correlation coefficients and it is obvious the skew normal distribution has more possibility in high gap interval.

	$\rho = 0$	$\rho = 0.5$	$\rho = 0.9$
Δ_{mean}^{skew}	11.5307	9.9504	4.9381
Δ_{mean}^{normal}	11.5434	10.0813	5.1491
Δ_{max}^{skew}	76.3818	68.9650	47.7216
Δ_{max}^{normal}	64.5271	52.6806	30.8328
$\#Inaccuracy\_Points(skew)$	321	471	132
$\#Inaccuracy\_Points(normal)$	46	91	17

Tab. 3.3: Prediction comparision between skew normal and normal distributions.

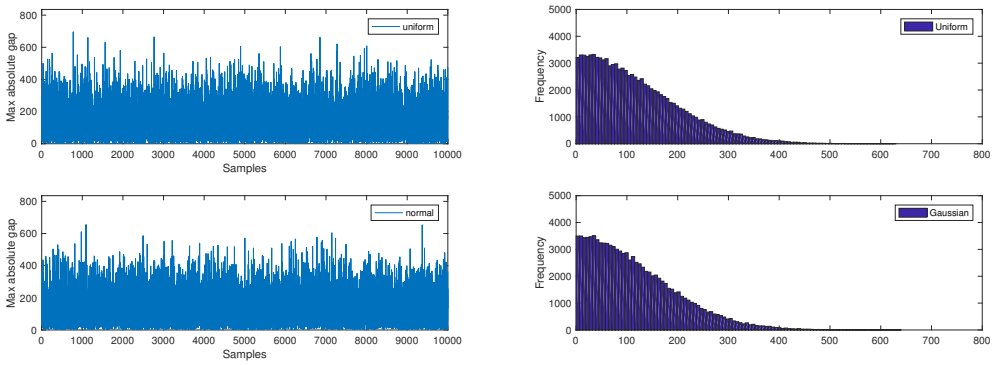
In Table 3.3, we can observe that this algorithm works more efficiently in the case of normal distribution than that of skew normal distribution, because the maximum value of skew normal distribution is higher than normal distribution and the number of inaccuracy points is also larger.

3.3 Multi-correlated Variables

The bivariate cases have been introduced and the results also have been compared with normal distribution individually. From now on, bivariate cases are extended to multivariate cases to check if the results obtained previously are still valid. In multivariate scenario, the PCA and SVD-QRcp methods are necessary, because these methods select most representative variables from huge amount of variables that can be used to predict other variables accurately. The initial parameters of distributions are kept the same as the one in bivariate cases. In addition, the correlation matrix, is set randomly. The minimum value of this correlation matrix is about 0.6 and mean value of this correlation matrix is about 0.8. Although there could be multiple variables to be predicted, the result from only one variable will be presented, because those variables have the same initial parameters and similar prediction results.

3.3.1 Multivariate uniform and normal distributions

In Fig 3.8, the absolute gap information of one predicted variable is presented. The left two pictures are the maximum gap values in each samples and the right ones are the gap distribution. The difference is not obvious enough, because the correlation matrix values are around 0.8.



(a) One variable max gap. (b) One variable gap distribution.

Fig. 3.8: One variabe absolute gap information of predicted variables.

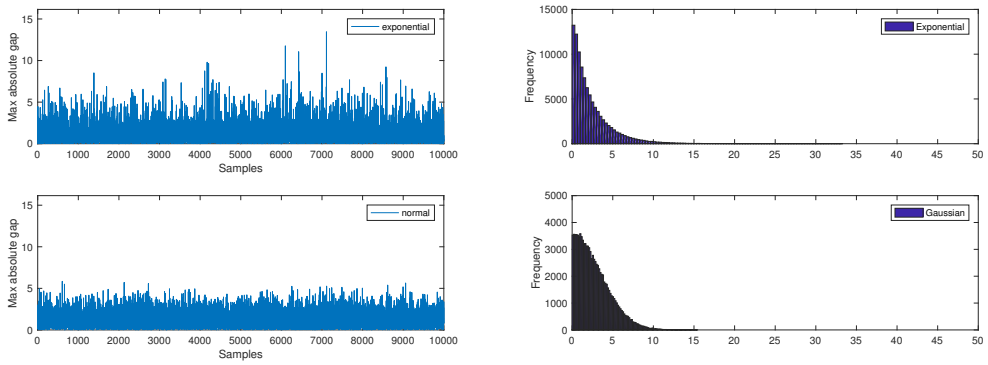
	$U(-50, 50)$	$U(-100, 100)$	$U(-200, 200)$
$\Delta_{mean}^{uniform}$	53.3586	141.3696	222.2706
Δ_{mean}^{normal}	50.9899	136.9007	212.6690
$\Delta_{max}^{uniform}$	443.2584	1091.4	1812.6
Δ_{max}^{normal}	436.2044	1062.5	1741.5
$\#Inaccuracy\_Points(uniform)$	39	70	89
$\#Inaccuracy\_Points(normal)$	27	52	62

Tab. 3.4: Prediction comparision between skew normal and normal distributions.

In Table 3.4, different uniform distribution bounds are used to test this algorithm. From the table, we can observe that the mean value, max value and the number of inaccuracy points are very similar, because the correlation matrix is near the turning point mentioned in bivariate uniform-normal case. Although the difference is slight, we can still observe that the algorithm works better on Gaussian distribution than uniform distribution.

3.3.2 Multivariate exponential and normal distributions

In Fig 3.9, as usual, the left two figures are the maximum gap in each sample and the right figures are the gap distributions. In this case, it is obvious that the maximum values of exponential distribution is much higher than normal distribution.



(a) One variable max gap. (b) One variable gap distribution.

Fig. 3.9: One variabe absolute gap information of predicted variables.

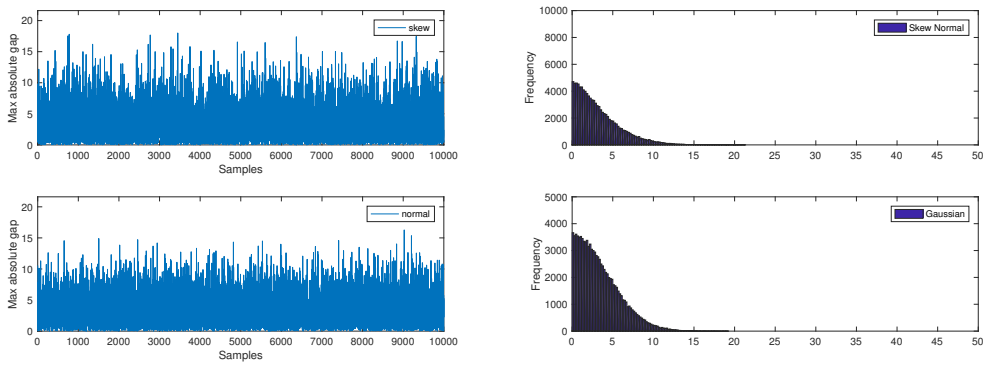
In Table 3.5, we can observe that the mean value of normal distribution is larger, and the maximum value of exponential distribution is bigger. Regarding the inaccuracy points, the number from exponential distribution is much higher than the one from normal distribution, which matches the case of bivariate scenario and verifies the inefficiency of this algorithm applied to exponential distribution.

	$\lambda = 0.05$	$\lambda = 0.1$	$\lambda = 0.2$
Δ_{mean}^{exp}	4.3619	2.2649	1.0550
Δ_{mean}^{normal}	4.9624	2.5685	1.2058
Δ_{max}^{exp}	80.4636	45.1862	20.9646
Δ_{max}^{normal}	38.3540	20.0484	9.4574
$\#Inaccuracy\_Points(exp)$	7766	6743	7322
$\#Inaccuracy\_Points(normal)$	39	26	42

Tab. 3.5: Prediction comparision between exponential and normal distributions.

3.3.3 Multivariate skew normal and normal distributions

In Fig 3.10, similar to the exponential case, the maximum value of skew normal distribution is higher than that of the normal distribution.



(a) One variable max gap. (b) One variable gap distribution.

Fig. 3.10: One variabe absolute gap information of predicted variables.

In Table 3.6, different δ values are tested. With δ approaching 0, the skew normal distribution is same with normal distribution. Therefore, the results are same.

	$\delta = -0.5$	$\delta = -0.9$	$\delta = -0.99$
Δ_{mean}^{skew}	3.4081	3.4544	3.3105
Δ_{mean}^{normal}	3.4081	3.4883	3.4461
Δ_{max}^{skew}	26.4660	36.7420	30.1133
Δ_{max}^{normal}	30.9757	32.4698	24.4375
$\#Inaccuracy\_Points(skew)$	7	37	1525
$\#Inaccuracy\_Points(normal)$	6	10	143

Tab. 3.6: Prediction comparision between skew normal and normal distributions.

4 Conclusion

In the previous section, the prediction algorithm is applied to three different non-Gaussian distributions, uniform, exponential and skew normal distributions respectively. According to the prediction results, non-Gaussian distributions have much higher number of inaccuracy points, so the algorithm works much better on Gaussian distribution than non-Gaussian distributions.
As introduced in Section 2, any distribution is determined by its all moments. However, this algorithm only considers first two moments and ignores the higher moments, which introduced inaccuracy in this algorithm.
In the future work, this algorithm could be improved by considering higher moments or using machine learning techniques in the variable selection phase.

REFERENCES

[1] G. L. Zhang, B. Li, and U. Schlichtmann, "Sampling-based buffer insertion for post-silicon yield improvement under process variability, " in *Design, Automation and Test in Europe (DATE)*, March 2016, pp. 1457-1460.

[2] G. L. Zhang, B. Li, J. Liu, Y. Shi, and U. Schlichtmann, "Design-phase buffer allocation for post-silicon clock binning by iterative learning, " in *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 2018, doi: 10.1109/TCAD.2017.2702632.

[3] H. Chang, SS. Sapatnekar, "Statistical timing analysis considering spatial correlations using a single PERT-like traversal," in *IEEE Proceedings International Conference on Computer-Aided Design (ICCAD)*, November 2013, pp. 621-625.

[4] L. Xie and A. Davoodi, "Representative path selection for post-silicon timing prediction under variability," in *ACM/IEEE Proceedings Design Automation Conference (DAC)*, January 2010, pp. 386-391.

[5] S. Wang, J. Chen, and M. Tehranipoor, "Representative critical reliability paths for low-cost and accurate on-chip aging evaluation," in *IEEE Proceedings International Conference on Computer-Aided Design (ICCAD)*, October 2012, pp. 736-741.

[6] F. Firouzi et al, "Representative critical-path selection for aging-induced delay monitoring," in *Proceedings of the IEEE International Test Conference (ITC)*, Septmber 2013, pp. 1-10.

[7] F. Firouzi et al, "Aging- and variation-aware delay monitoring using representative critical path selection," in *ACM Transactions on Design Automation of Electronic Systems*, June 2015, vol 20, no 3, article 39.

[8] G. L. Zhang, B. Li, and U. Schlichtmann, "EffiTest: Efficient delay test and statistical prediction for configuring post-silicon tunable buffers," in *ACM/IEEE Proceedings Design Automation Conference (DAC)*, June 2016, pp. 60:1-60:6.

[9] G. L. Zhang, B. Li, Y. Shi, J. Hu and U. Schlichtmann, "EffiTest2: Efficient Delay Test and Prediction for Post-Silicon Clock Skew Configuration under Process Variations, " in *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 2018.

[10] MathWorks, "Generating Data Using Flexible Families of Distributions," *https://de.mathworks.com/help/stats/generating-data-using-flexible-families-of-distributions.html*, 2017.

[11] MathWorks, "Simulating Dependent Random Variables Using Copulas," *https://de.mathworks.com/help/stats/examples/simulating-dependent-random-variables-using-copulas.html*, 2017.