Two dimensional correlated sampling using alias technique

To cite this article: S Mohanty et al 2012 J. Phys.: Conf. Ser. 368 012045

View the article online for updates and enhancements.
Two dimensional correlated sampling using alias technique

S Mohanty\textsuperscript{1,2}, S Banerjee\textsuperscript{1,3}, J Jose\textsuperscript{1,3}, D Goyal\textsuperscript{1,4}, A K Mohanty\textsuperscript{1,5} and F Carminati\textsuperscript{1}

1 CERN, CH-1211, Geneve 23, Switzerland
2 Department of Computer Science, VNIT, Nagpur, India 440010
3 Department of Computer Science, LNMIIT, Jaipur, India
4 Department of Electronics and Communication Engineering, LNMIIT, Jaipur, India
5 Nuclear Physics Division, Bhabha Atomic Research Centre, Mumbai, India 400085

E-mail: siddhant9892@gmail.com, sbanerje@cern.ch, Johny.Jose@cern.ch, Dushyant.Goyal@cern.ch, ajit@cern.ch, Federico.Carminati@cern.ch

Abstract. Monte-Carlo sampling of two dimensional correlated variables (with non zero covariance) has been carried out using an extended alias technique which was originally proposed by A. J. Walker to sample from an one dimensional distribution. Although, the method has been applied to a correlated two dimensional Gaussian data sample, it is quite general and can easily be extended for sampling from a multidimensional correlated data sample of any arbitrary distribution.

1. Introduction
Monte-Carlo technique enables one to generate random samples from distributions with known characteristics and helps to make probability based inferences of the underlying physical processes. Simulation codes based on Monte-Carlo techniques have become an important aspect in science, technology and business. Fast and efficient Monte-Carlo particle transport code particularly for high energy nuclear and particle physics experiments has become an important tool starting from the design and fabrication of detectors to the modeling of the physics outcome as close as the reality. Quite often Monte-Carlo simulations require multivariate random numbers to be generated from correlated data both from normal and non-normal distributions. Although several techniques exist for multivariate correlated samplings of varying degrees of success, the most practical method is the technique that uses the principal component analysis (PCA) of the given correlation matrix for generating multivariate random numbers with specified inter-correlations. In case of multivariate correlated Gaussian, the distribution can be transformed to a different orthogonal basis using PCA technique\textsuperscript{[1]}. After PCA transformation, the new distribution function can be expressed as the product of \(n\) independent Gaussian distributions with variance \(\lambda_i\) where \(\lambda_i\)'s are distinct eigenvalues of the original covariance matrix \(C\). Monte-Carlo sampling is then carried out independently to build the normal random vector \(Y = (y_1, y_2, \ldots, y_n)^T\) with variance \(\lambda_i\) which can finally be transformed to the original vector \(X = (x_1, x_2, \ldots, x_n)^T\) with mean \(\mu_x = E\{X\}\) through the relation \(X = A^T Y + \mu_x\). Here \(A\) is the matrix consists of eigenvalue vectors of the covariance matrix \(C\). While the above component
analysis is suitable for multivariate normal distribution, it fails when the distribution is non-
Gaussian. Alternate methods have also been proposed for non-Gaussian multivariate analysis
which requires knowledge of higher order moments like skewness and kurtosis in addition to
mean and variance [2] thus adding further computational complexity. In this work, we propose
an extended alias sampling which was originally proposed by A. J. Walker in 1977 [3] to sample
from an one dimensional distribution. This method is quite simple to implement and reproduces
the original distribution well (verified through chi-square, co-variance and Kolmogorov-Smirnov
goodness fit tests). This method is also quite robust and is applicable to all type of multivariate
distribution irrespective of whether the distribution is Gaussian or Non-Gaussian.

Although this method is quite general and can be applied to any dimensions, in this work we
have restricted only to two dimensions. The motivation behind this study has been to develop a
ROOT based Monte-Carlo application package for low energy neutron transport (down to a few
keV) using data from ENDF (Evaluated Nuclear Data File) which is available in ROOT format
(for detail on ROOT, refer CERN web page). Work is in progress to apply this new method of
alias technique to data set where the angle and energy distributions of neutron emissions are
correlated.

2. The alias sampling
Theoretically any random variable \( x \in (a, b) \) with probability density \( p(x) \) satisfying

\[
\int_a^b p(x)dx = 1 \tag{1}
\]

and cumulative probability density,

\[
Q(x) = \int_a^x p(x')dx' \tag{2}
\]

can be sampled using the inverse function method which gives,

\[
x = Q^{-1}(u) \tag{3}
\]

where \( u \) is a random number uniformly distributed in the interval [0, 1]. Numerically, \( Q^{-1}(u) \)
\can be divided into \( N \) equal probable intervals. Sampling is carried out by selecting a bin randomly
between 1 and \( N \) with equal probability \( 1/N \) and with bin number, \( j = Nu_1 + 1 \) and then
randomly selecting a value within the bin with an uniform probability \( x = (1 - u_2)x_{j+1} + u_2x_j \)
where \( u_1 \) and \( u_2 \) are random numbers uniformly distributed between 0 and 1. This method is
known as Equal Probable Bin (EPB) method which is quite fast and has been adopted in most
of the Monte-Carlo programs.

However, it is inaccurate as much of the original distribution is lost by the necessary
assumption of uniform likelihood within each bin. Another popular method is alias sampling
which is as fast as the equal probable bin and can be made as accurate as table look up.
Initially, it was proposed by A. J. Walker in 1977 to sample from a discrete data set [3] which
was subsequently modified to generate continuous distribution through the technique known as
linear alias sampling [4].

2.1. Alias sampling for discrete distributions
The alias method described by Walker, generates random variables from any discrete distribution
with a definite number of outcomes. The alias method very closely resembles the rejection
method of generating discrete random numbers, however instead of rejecting a number, a number
is either accepted or it is replaced with its alias value (defined later) . The alias technique works
by constructing an alias table for a given distribution. Consider we are given a distribution describing the probability of occurrence of $N$ discrete events. The alias technique converts this distribution to a distribution of $N$ separate events which occur with the equal probability of $\frac{1}{N}$. Each of these events $E_i$ consists of two possible outcomes, the standard outcome $x_i$ with probability $p_i$ and the alias outcome $\Lambda_i$ with the probability of it’s occurrence $\Pi_i$.

As suggested by Walker, we construct this table by considering any two events with probabilities such that one is more and other is less than the average likelihood $\frac{1}{N}$. In the following, we give an example which has been adopted from [4]. Consider a data set having $N(= 6)$ discrete tabulated probability values $p_i$ such that $\sum p_i = 1$ (see table 1 below). The first and second column of the table 1 show the original value of $i$ (in case $i$ represents a bin number, $x_i$ represents the bin content) and it’s normalized probability $p_i$. In Walker’s notation, $i$ and $p_i$ are called non-alias outcome and probability. In this example, the average likelihood value is $\frac{1}{6} = 0.1667$. Consider two non-alias events $i = 1$ and $i = 2$. The probability of event 2 is less from the average by an amount $0.1667 - 0.08 = 0.0867$ which is subtracted from the probability of event 1 to get a new probability $0.24 - 0.0867 = 0.1533$. The alias probability $\Pi$ is then calculated by multiplying $N$ with 0.08 i.e to the probability of event 2 that gives $\Pi = 0.48$. The corresponding alias outcome $\Lambda$ is taken as 1 which is the donor event (reduced by donating a value 0.0867). Thus, in the first step, we get alias outcome $\Lambda = 1$ corresponding to the alias probability $\Pi = 0.48$. The corresponding non-alias outcome $i$ and probability $p_i$ are 2 and 0.08 respectively. Next, event 2 is removed from the table and the second iteration begins with remaining events (except 2) with event 1 replaced by the reduced probability of 0.1533. The process continues and gets completed after six iterations (see appendix for remaining iterations).

The third and fourth column of table 1 show the alias outcome $\Lambda_i$ and alias probability $\Pi_i$ obtained after six iterations.

The outcome is sampled by first randomly selecting an equal probable event $i$,

$$i = [Nu_1 + 1], \quad (4)$$

where $u_1$ is an uniform random number between $[0,1]$. The method then compares a second uniform random number $u_2$ against the alias probability to select either the alias or non-alias event. If $u_2 \leq \Pi_i$, then non-alias event $i$ is chosen. Otherwise an alias event $\Lambda_i$ is selected. Note that in case of $i$ (or $\Lambda_i$) represents a bin number, the bin content $x$ is selected by,

$$x = \begin{cases} x[i] & \text{if } u_2 \leq \Pi_i \\ x[\Lambda_i] & \text{otherwise} \end{cases} \quad (5)$$

| $i$ | $p_i$ | $\Lambda_i$ | $\Pi_i$ |
|-----|------|-------------|--------|
| 1   | 0.24 | 3           | 0.92   |
| 2   | 0.08 | 1           | 0.48   |
| 3   | 0.28 | 3           | 1.0    |
| 4   | 0.12 | 3           | 0.72   |
| 5   | 0.12 | 3           | 0.72   |
| 6   | 0.16 | 3           | 0.96   |
2.2. Alias sampling for continuous distributions

Though the above method is meant to generate random variables which has a discrete distribution, it can be extended to sample continuous probability densities as well [4]. Given a probability distribution function $y(x)$, we estimate the discrete tables through the integral,

$$F_i = \int_{x_i}^{x_{i+1}} y(x) \, dx$$  \hspace{1cm} (6)

We now construct the alias table for the discrete distribution $(i, F_i)$ and carry out alias sampling using the set $\{F_i\}$ as discussed above which results in an interval $[x_i, x_{i+1}]$. Assuming that the distribution is piece-wise linear, a third uniform random number $u_3$ (between $[0, 1]$) is used to estimate $x_1$ and $x_2$ given by,

$$x_1 = (1 - u_3)x_i + u_3x_{i+1}$$ \hspace{1cm} (7)

$$x_2 = u_3x_i + (1 - u_3)x_{i+1}$$ \hspace{1cm} (8)

Using a fourth uniform random number $u_4$, we generate the required random variable $x$,

$$x = \begin{cases} x_1 & \text{if } u_4(y_i + y_{i+1}) \leq (1 - u_3)y_i + u_3y_{i+1} \\ x_2 & \text{otherwise} \end{cases}$$ \hspace{1cm} (9)

Thus, it can be noticed that the EPB method requires two random numbers where as the alias sampling requires four random numbers. Another difference is that the alias method stores two tables II and Λ. Although extra storage requirement is a disadvantage as compared to EPB method, the linear alias method is easy for numerical implementation. This alias technique can also be used for interpolation between two tabulated discrete data sets [5].

In the following, we carry out a comparative study using both EPB and linear alias samplings. As an example, we have considered RENDF/B-VI, file 5 for LF=1 which gives the secondary neutron energy probability distribution as an arbitrary tabulated probability. We select the reaction MT=91 for neutron interacting with $^{238}\text{U}$ via inelastic continuum reaction. Figure 1 shows a typical sampling at $E_n = 8$ MeV and figure 2 shows the similar plot at $E_n = 15$ MeV using both EPB and alias samplings. The filled circles (blue color) are the ENDF tabulated data points. The red curve is the result of EPB samplings which divides the entire range into 25 equal probable bins of width 0.04 each. The blue curve is the result of linear alias sampling which passes exactly though the data points which is by construction. However, in between two data points, a linear alias sampling is carried out as discussed before. If the curve is linear interpolable between two data points, the linear alias sampling will give accurate result. On the other hand, the EPB sampling will not be able to reproduce the distributions if the rise is very fast (see figure 1(b) on log scale) or the distribution has long tail.

2.3. A 2D correlated alias sampling

We have extended the above alias technique to sample from a multi dimensional correlated data set. To demonstrate how it works, we consider a correlated two dimensional Gaussian given by,

$$f(x, y) = e^{-[(x-x_0)^2+(y-y_0)^2+\alpha(x-x_0)(y-y_0)]/\sigma^2}$$ \hspace{1cm} (10)

The above function is uncorrelated Gaussian when $\alpha = 0.0$ and can be written as the product of two independent Gaussian. However for non-zero value of $\alpha$, the above function can not be factorized. This corresponds to a situation when co-variance is non zero. Therefore, we
have proposed a new sampling scheme based on the alias technique as described below. In this example, we generate 100 discrete data points $p[i][j]$ corresponding to $i = 0, 1, 2, 3, 4, 5, 6, 7, 8, 9$ and $j = 0, 1, 2, 3, 4, 5, 6, 7, 8, 9$ for discrete sampling. First, we choose a row in random based on the distribution $px[i]$ (using alias technique) where $px[i] = \sum_j p[i][j]$. Once the row $I$ is chosen the corresponding $j$ values are chosen depending on the distribution $p[I][j]$. In the present work, we have considered alias sampling as this method helps to get back the original distribution at each input points. So we can carry out $\chi^2$, covariance and Kolmogorov-Smirnov (K-S) test to study the quality of the sampling. It can be mentioned here that other methods like equal probability bin can also be implemented with this algorithm.

Figure 3 shows the original and generated distributions after sampling $n = 10^6$ events for $\alpha = 1.0$.

We define $\chi^2$ as

$$\chi^2 = \frac{1}{N} \sum \left[ \frac{p[i][j] - g[i][j]}{p[i][j]} \right]^2,$$

where $g$ is the sampled distribution over the $N$ data points. In addition to the $\chi^2$ test, we carry
Figure 3. The comparison between actual inputs (circles) and generated data (shown as continuous line) using alias sampling for $\alpha = 1.0$. The X and Y axes are bin numbers (0-9) whereas the height represents the normalized probability distribution of the correlated Gaussian as described in the text.

The covariance test which is defined as,

$$\sigma_{xy} = \frac{1}{n} \sum (x - \bar{x})(y - \bar{y})$$

(12)

We have also carried out goodness of fit test using similar procedure as that of K-S (one sample test) where we compute the maximum absolute difference $D_n$ of the observed and expected cumulative distribution functions (cdf) $F_o(i)$ and $F_e(i)$,

$$D_n = \max |F^n_o(i) - F_e(i)|.$$  

(13)

where cdfs $F^n_o(i)$ and $F_e(i)$ are given by,

$$F^n_o(i) = \frac{1}{n} \sum_{j=1}^{n} I_{j\leq i}$$

$$F_e(i) = \sum_{j=0}^{i} p[i].$$

(14)

In the above, $I_{j\leq i}$ is the indicator function equal to 1 if $j \leq i$ and equal to zero otherwise [6]. Note that since our data set is discrete, we estimate the cdfs at each bin $i$. As a goodness of fit, we expect $D_n \to 0$ when $n \to \infty$. Thus, $\chi^2$ test makes the comparison using the two pdfs where as $D_n$ test is done using cdfs.

Table 2. The table for $\chi^2$ test using TRandom3 of ROOT package, $N = 100$ is the number of data points and $n$ is the sample size. The parameters of the Gaussian are $x_0 = y_0 = 5$ and $\sigma^2 = 4$.

| $\alpha$ | $\chi^2/N, n = 10^2K$ | $\chi^2/N, n = 10^3K$ | $\chi^2/N, n = 10^4K$ |
|----------|----------------------|----------------------|----------------------|
| 0.0      | 0.1449               | 0.0328               | 0.0032               |
| 1.0      | 0.512                | 0.112                | 0.034                |
| 2.0      | 0.38                 | 0.418                | 0.108                |
| 3.0      | 1.453                | 0.266                | 0.491                |
Table 3. The table for $D_n$ test using TRandom3 of ROOT package, $N = 100$ is the number of data points and $n$ is the sample size. The parameters of the Gaussian are $x_0 = y_0 = 5$ and $\sigma^2 = 4$.

| $\alpha$ | $D_n$, $n = 10^2K$ | $D_n$, $n = 10^3K$ | $D_n$, $n = 10^4K$ |
|----------|---------------------|---------------------|---------------------|
| 0.0      | 1.8E-3              | 6.5E-4              | 4.0E-4              |
| 1.0      | 1.6E-3              | 1.2E-3              | 3.0E-4              |
| 2.0      | 2.0E-3              | 7.0E-4              | 2.5E-4              |
| 3.0      | 1.0E-3              | 7.4E-4              | 2.9E-4              |

The tables 2 and 3 show the $\chi^2$ and $D_n$ values for several $\alpha$ with different sample size $n$. The decreasing values of $\chi^2$ and $D_n$ with increasing sample size indicate that the method is working well. Please note that for a given sample size $n$, the dependence of both $\chi^2$ and $D_n$ on $\alpha$ is bit erratic. This should not be interpreted as the failure of the method as with increasing $\alpha$, the tail of the distribution decreases sharply and the sample size needs to be increased accordingly for meaningful comparison. For example, the ratio of the lowest to highest probability becomes as low as $10^{-5}$ when $\alpha = 0$ and $10^{-12}$ when $\alpha = 3$. Therefore, for statistical comparisons as a function of $\alpha$, the sample size should be at least $10^{12}$ and more. Table 4 shows the co-variance test of the actual inputs versus the sampled outputs. The second and third columns show the input variance and co-variance whereas the fourth and fifth columns show the corresponding sampled variances which are in good agreement.

Table 4. The table for co-variance test using TRandom3 of ROOT package, $N = 100$ is the number of data points and $n$ is the sample size which is fixed at 100$K$. The parameters of the Gaussian are $x_0 = y_0 = 5$ and $\sigma^2 = 4$.

| $\alpha$ | $\sigma^2$ | $\sigma^2_{xy}$ | $\sigma^2$ | $\sigma^2_{xy}$ |
|----------|------|----------------|------|----------------|
| 0.0      | 1.985 | -3.32E-17     | 1.977 | -0.0008        |
| 1.0      | 2.579 | -1.264        | 2.569 | -1.255         |
| 2.0      | 6.856 | -5.968        | 6.845 | -5.978         |
| 3.0      | 16.30 | -15.94        | 16.31 | -15.94         |

3. Conclusion

In conclusion, we have carried out alias sampling from a discrete data set generated using a two dimensional correlated Gaussian distribution. In case of multivariate correlated Gaussian distribution, principal component analysis is a well known technique quite often used for Monte-Carlo samplings, although the method fails when distribution is strongly non-Gaussian. The present alias sampling is definitely not a superior method as compared to PCA, but quite robust and convenient for numerical implementation. It may be mentioned here that unlike PCA method, this alias technique is quite general and can be applied to any data set irrespective of whether the distribution is Gaussian or non-Gaussian. At present, our approach is confined only to two dimensional discrete sampling. Work is in progress to extend this method to higher dimensions and also to generate continuous distributions using linear alias principle.
Appendix

For the computer implementation, consider the following probability distribution (for N=6):

\[
\{p/i\} = \{0.24/1, 0.08/2, 0.28/3, 0.12/4, 0.12/5, 0.16/6\}
\]

Consider any two events having probability greater and smaller than \(1/N = 0.1667\) (say events 1 and 2). Event 2 is less by an amount \(.0867\) from the average which can be subtracted from event 1 \(.24 - .0867 = 0.1533\). Thus, in the step 1, the original distribution is updated to:

\[
\{p/i\} = \{0.1533/1, 0/2, 0.28/3, 0.12/4, 0.12/5, 0.16/6\}
\]

with alias representation,

\[
\{\Pi/\Lambda\} = \{-, 0.48/1, -, -, -, -\}
\]

In the above, the alias probability \(\Pi\) has been estimated by multiplying \(N\) to the original non alias probability 0.08 and the alias outcome \(\Lambda\) is the donor event 1 whose probability is reduced to 0.1533. Next, we can consider events 1 and 3. Since event 1 is less by an amount of 0.0134 from the average which can be subtracted from event 3. Thus, in step 2 the updated distributions become

\[
\{p/i\} = \{0/1, 0/2, 0.2667/3, 0.12/4, 0.12/5, 0.16/6\}
\]

with alias representation,

\[
\{\Pi/\Lambda\} = \{0.92/3, 0.48/1, -, -, -, -\}
\]

Like before, the alias probability is \(N\) times 0.1533 with alias outcome 3. w

Next consider events 3 and 4. The updated distributions in step 3 now becomes

\[
\{p/i\} = \{0/1, 0/2, 0.22/3, 0/4, 0.12/5, 0.16/6\}
\]

with alias representation,

\[
\{\Pi/\Lambda\} = \{0.92/3, 0.48/1, -, 0.72/3, -, -\}
\]

Next consider events 3 and 5. The updated distributions in step 4 now becomes

\[
\{p/i\} = \{0/1, 0/2, 0.1733/3, 0/4, 0.12/5, 0.16/6\}
\]

with alias representation,

\[
\{\Pi/\Lambda\} = \{0.92/3, 0.48/1, -, 0.72/3, 0.72/3, -\}
\]

Next consider events 3 and 6. The updated distributions in step 5 now becomes

\[
\{p/i\} = \{0/1, 0/2, 0.1667/3, 0/4, 0/5, 0/6\}
\]

with alias representation,
\{\Pi/\Lambda\} = \{0.92/3, 0.48/1, -1, 0.72/3, 0.72/3, 0.96/3\}

Finally, in the 6th step, the updated distribution becomes

\{p/i\} = \{0/1, 0/2, 0/3, 0/4, 0/5, 0/6\}

with alias representation,

\{\Pi/\Lambda\} = \{0.92/3, 0.48/1, 1.0/3, 0.72/3, 0.72/3, 0.96/3\}

Note that the above representation is not unique, there can be different non-alias/alias representation \Pi/\Lambda depending on how the iteration is carried out.

References
[1] Jolliffe I T 2002 Principal Component Analysis (Berlin, Springer-Verlag)
[2] Nagahara Yuichi 2003 Computational Statistics and Data Analysis vol 47 p 1-29
[3] Walker A J 1977 ACM Transactions on Mathematical Software vol 3 p 253
[4] Edwards A L, Rathkopf J A, and Smidt R K 1991 Report UCRL-JC-104791
[5] Popescu L M 2000 Journal of Computational Physics vol 160 p 612
[6] Eadie W T, Drijard D, James F E, Roos M and Sadoulet B 1971 Statistical Methods in Experimental Physics (Amsterdam, North-Holland)