Random Number Generating using Kernel Density Estimator

Yaqiong Yao, Cheng Huang

December 15, 2018

Abstract

The purpose of this project is to improve the method of drawing samples from an unknown distribution if we have a sample dataset. We are going to introduce kernel density estimate to estimate the cumulative density function of the given dataset. Then sample data from the estimated pdf. This method can be applied to both univariate samples and multivariate samples which are constraint by copulas.

1 Introduction

Draw samples from a set of given observed data $\{x_i\}_{i=1}^n$ with unspecified distribution is widely used especially in bootstrap and is still a topic needing pay attention to. Efron (1979) came up with the idea of bootstrap and they proposed to use observations randomly selected from $\{x_i\}_{i=1}^n$ with replacement as a sample to inspect the features of the unknown distribution. whuber (2018) introduced the idea of using kernel density estimate (KDE) to draw samples from. In this project, our method is based on the idea of implementing KDE, estimating the cumulative density function of the unknown distribution to take samples and we are going to extend the idea to multivariate samples linked by copulas.

The structure of this project is presented as following. In section 2, we introduce our methodology of taking samples and present the numerical simulations in Section 3.

2 Methodology

2.1 Univariate Case

Given samples $\{x_i\}_{i=1}^n$ from an unknown distribution, the method proposed by whuber (2018) is first to randomly select values from a pre-specified kernel density and then draw samples from data we have. Adding these two parts together forms the sample we want. They also provided a proof showing that the samples they obtained follow its kernel density estimator (KDE)

$$\hat{F}_b(x|\mathcal{D}) = \sum_{i=1}^{n} \frac{1}{n} K\left(\frac{x - x_i}{b}\right) \tag{1}$$

where $K(\cdot)$ is the cumulative density function of the selected kernel and b is the bandwidth.

Our method is to obtain the cdf of the KDE explicitly and then randomly sample $u_l \sim \text{Unif}(0,1), l = 1, 2, ..., N$. Map each u_i to the cdf of KDE and the random samples can be acquired. If we select the uniform kernel, the equation 1 becomes

$$\hat{F}_b(x|\mathcal{D}) = \frac{1}{n}\left\{\sum_{i=1}^{n}\frac{2x-2x_i+b}{2b}I\left(x_i-\frac{b}{2}<x<x_i+\frac{b}{2}\right)+\sum_{i=1}^{n}I\left(x\geq x_i+\frac{b}{2}\right)\right\}. \quad (2)$$

Obviously, this is the a piecewise function with $2n+1$ intervals and in all intervals, it is a linear function with slop $\frac{n_t}{b}, t = 1, 2, ..., 2n+1$ where n_t is the number of points affecting tth interval. Denote x_t^0 and x_t^1 as the starting point and ending point of tth interval. Thus $n_t = \sum_{i=1}^{n} I(x_i - \frac{b}{2} \leq x_t^0 < x_t^1 < x_i + \frac{b}{2})$. Then we can easily draw random samples once all intervals can be obtained.

We also use the triangular kernel to obtain the kernel density estimate. In this scenario, the equation 1 is

$$\hat{F}_b(x|\mathcal{D}) = \frac{1}{n}\left\{\sum_{i=1}^{n}\int_{x_i-\frac{b}{2}}^{x}\frac{2b-4|x-x_i|}{b^2}I\left(x_i-\frac{b}{2}<x<x_i+\frac{b}{2}\right)dx+\sum_{i=1}^{n}I\left(x\geq x_i+\frac{b}{2}\right)\right\}.$$

This is a piecewise function with $3n+1$ intervals. In each interval, the KDE is in either quadratic form or linear form. The expression for tth interval, $t = 1, 2, ..., 3n+1$ is $\lambda(x - x_t^0) + \frac{2n_t^*(x^2-(x_t^0)^2)}{b^2}$, where $\lambda = \sum_{i=1}^{n}\int_{x_i-\frac{b}{2}}^{x_t^0}\frac{2b-4|x-x_i|}{b^2}I\left(x_i-\frac{b}{2}<x<x_i+\frac{b}{2}\right)dx$ and $n_t^* = \sum_{i=1}^{n} I\left(x_i - \frac{b}{2} \leq x_t^0 < x_t^1 \leq x_i\right) - \sum_{i=1}^{n} I\left(x_i \leq x_t^0 < x_t^1 \leq x_i + \frac{b}{2}\right)$.

2.2 Multivariate Case

In multivariate case, the dependence of random variables are defined by a known copula. Given the observation $\{\mathbf{x}_i\}_{i=1}^n$, where $\mathbf{x}_i = (x_{1i}, x_{2i}, ..., x_{mi})$ is the value of random variables $x_1, x_2, ..., x_m$. With the known copula C, the joint probability function is

$$C(u_1, u_2, ..., u_m) = \text{P}(x_1 \leq F_1^{-1}(u_1), x_2 \leq F_2^{-1}(u_2), ..., x_m \leq F_m^{-1}(u_m))$$

where $F_1^{-1}, F_2^{-1}, ..., F_m^{-1}$ are the cdf of every random variable and $u_1, u_2, ..., u_m$ are the marginal probability of each random variable.

The method of whuber (2018) can be extended to the multivariate case based on a selected kernel density $k(\cdot)$. The procedure is first generate $\mathbf{e}_l = (e_{1l}, e_{2l}, ..., e_{ml}), l = 1, 2, ..., N$ with each element independently selected from density $k(\cdot)$ and draw observations $\mathbf{x}_l^* = (x_{1l}^*, x_{2l}^*, ..., x_{ml}^*)$ from $\{\mathbf{x}_i\}_{i=1}^n$ with replacement. Then $\{\mathbf{x}_l^* + \mathbf{e}_l\}_{l=1}^N$ are samples we obtained and follow multivariate kernel density estimator. The proof is shown as following

$$\begin{aligned}
\text{P}(\mathbf{x}_l^* + \mathbf{e}_l \leq \mathbf{x}) &= \sum_{i=1}^{n}\text{P}\{\mathbf{x}_i + \mathbf{e}_l \leq \mathbf{x}|\mathbf{x}_l^* = \mathbf{x}_i\}\text{P}\{\mathbf{x}_l^* = \mathbf{x}_i\} \\
&= \frac{1}{n}\sum_{i=1}^{n}\text{P}\{\mathbf{e}_l \leq \mathbf{x} - \mathbf{x}_i|\mathbf{x}_l^* = \mathbf{x}_i\} \\
&= \frac{1}{n}\sum_{i=1}^{n}K_{\boldsymbol{B}}(\mathbf{x} - \mathbf{x}_i)
\end{aligned}$$

where $K_{\boldsymbol{B}}(\cdot)$ is the multivariate kernel density and $\boldsymbol{B}$ is the bandwidth matrix. Here since $(e_{1l}, e_{2l}, ..., e_{ml})$ are generated independently and identically, $\boldsymbol{B}$ is a matrix with diagonal elements being $b_1, b_2, ..., b_m$ and off-diagonal elements being 0.

Here, we suppose that copula is known, in this scenario, we can generate the marginal probabilities by copula and the dependence is constraint by the marginal probabilities. So we can calculate the CDF of KDE explicitly for each random variable respectively. Then map the marginal probabilities to its corresponding CDF to obtain the random samples.

3 Numerical Simulations

3.1 Simulation 1 : Univariate Case

The univariate example is used to illustrate the performance of our method of constructing the KDE and equating the CDF of the estimated kernel density to u, the random number generated from Unif$(0, 1)$. The original sample (figure 1) of size $n = 100$ is generated from a mixture gamma distribution $0.3\text{Gamma}(1, 5) + 0.7\text{Gamma}(5, 1)$.

We generated $N = 1000$ samples from the proposed method using both uniform kernel and triangular kernel. Figure 2 shows the histograms of samples generated by these two kernels. We can see that both kernels give us samples close to the true density, stating that our method performs well in this situation.

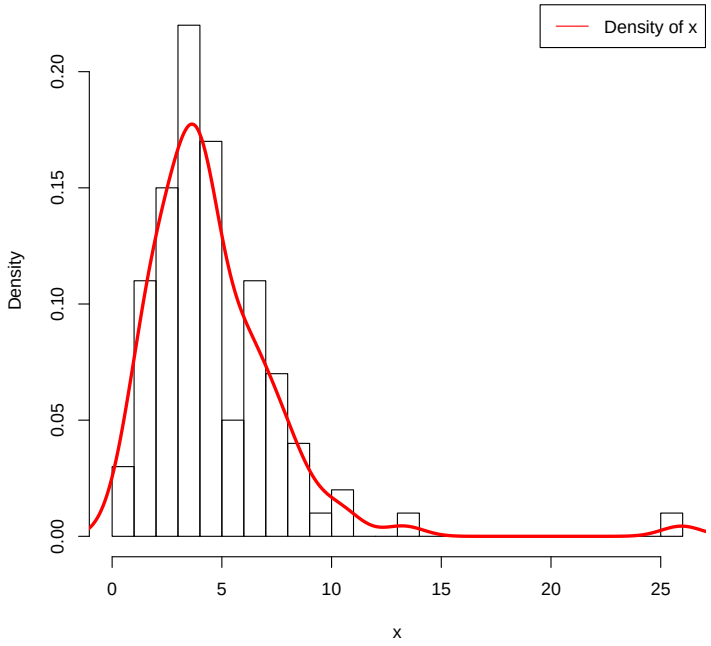
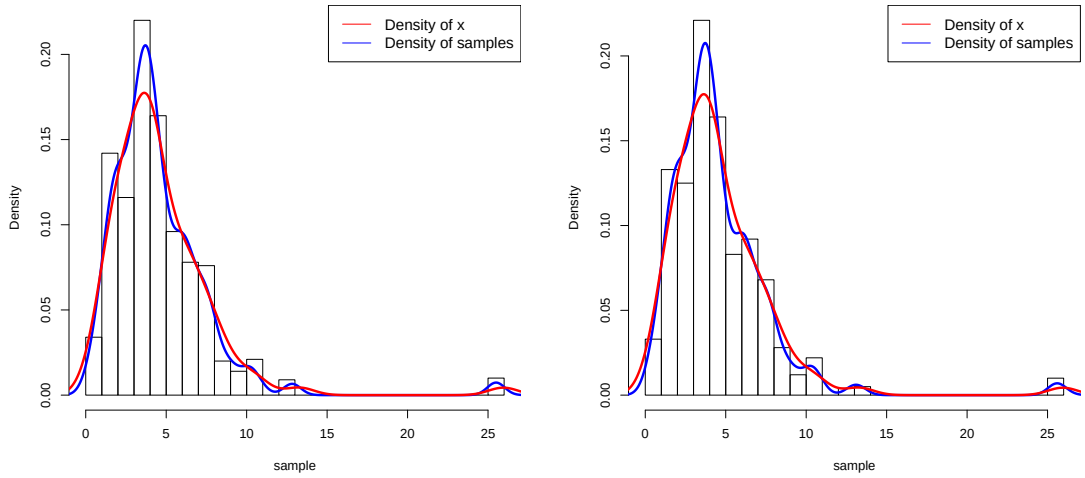


Figure 1: Sample from mixture gamma



(a) Uniform kernel (b) Triangular kernel

Figure 2: Histogram of samples generated by different kernels.

3.2 Simulation 2 : Bivariate Case Generate sample from bivariate copula data by kernel density estimator (KDE)

3.2.1 Generate data from gaussian copula

The original sample is generated from Copulas because we want the sample to be correlated. Here we use bivariate random variables, x, which marginally follows Gamma$(1, 2)$ and y, which marginally follows $\chi^2(10)$. Therefore first we use a multivariate Gaussian to generate two correlated random variables, a and b, with mean $\begin{pmatrix} 0 \\ 0 \end{pmatrix}$ and covariance matrix $\begin{pmatrix} 1 & 0.5 \\ 0.5 & 1 \end{pmatrix}$.Then we transform them using the cumulative Gaussian distribution into $u = \Phi(a)$ and $v = \Phi(b)$. Now u and v have marginal uniform distributions and they are still correlated. Finally transform $x = F^{-1}(u)$ and $y = G^{-1}(v)$ where $F(x)$ is the cumulative distribution function of Gamma$(1, 2)$ and $G(y)$ is the cumulative distribution function of $\chi^2(10)$. By this way, we generate data of size $n = 200$ and the plot is shown in figure 3.

3.2.2 Method 1: Draw sample from Kernel density estimation(KDE)

The first method is generate random variables inspired by the method of whuber (2018). As proved in Section 2, we first resample from the original data x and y with replacement, and then draw a value from the kernel density, uniform kernel density in our example. The newly generated sample is their summation. Figure 4 in below shows generated sample using this method for $N = 1000$.

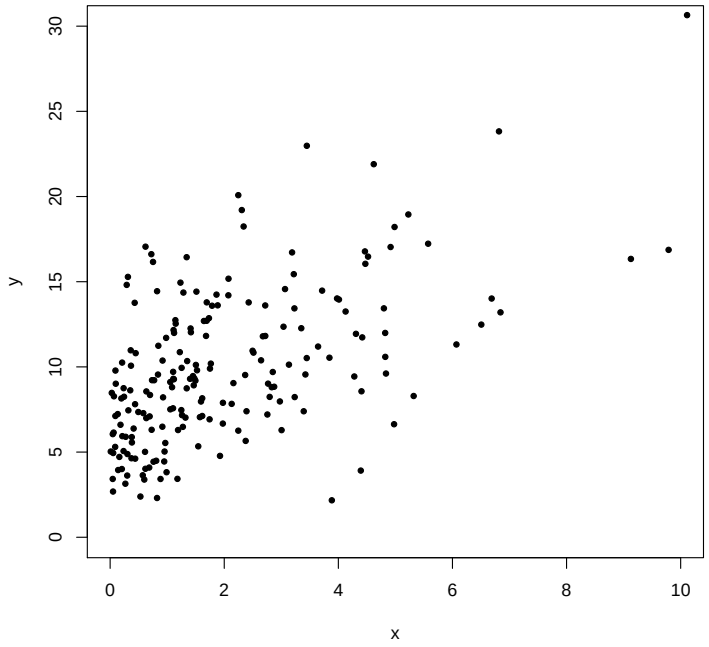
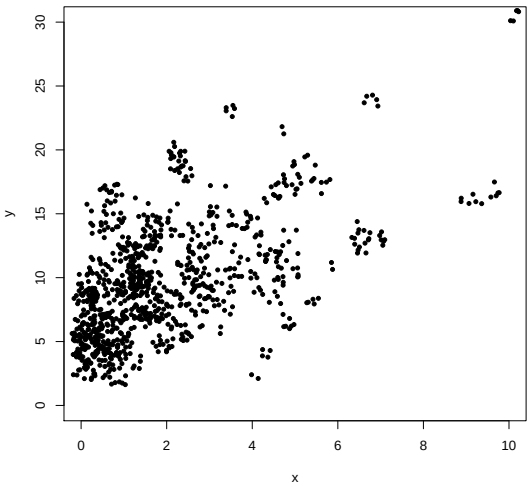


Figure 3: Data generated from copula.

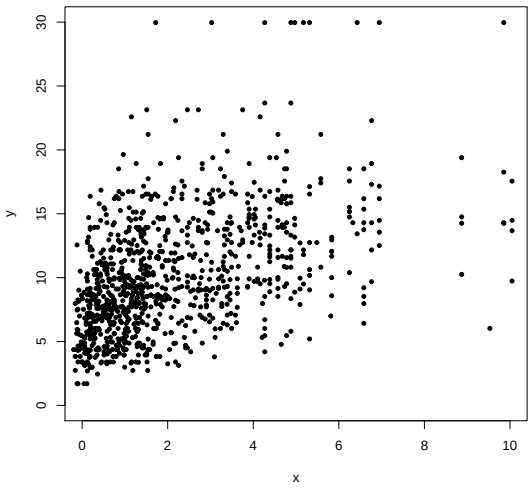
3.2.3 Method 2: Calculate the kernel density estimation and solve $x = F^{-1}(u)$

The second method is to construct the kernel density estimation and by equating the CDF of the estimation with u, the value draw from uniform (0,1). The sample is solved from the equation $x = F^{-1}(u)$, where $F(\cdot)$ is the CDF of estimated kernel density. Figure 4 in below shows generated sample using KDE for $N = 1000$ using both uniform kernel and triangular kernel.

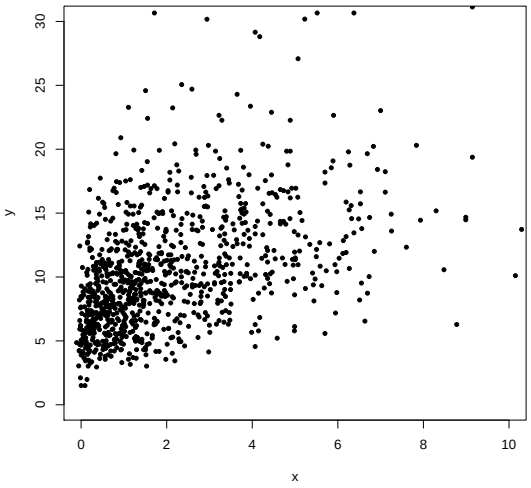
Figure 4 also gives the plot of samples of size $N = 1000$ generated from true density, which is defined in section 3.2. It shows that the samples generated from triangular kernel is the one most close to samples from true density.



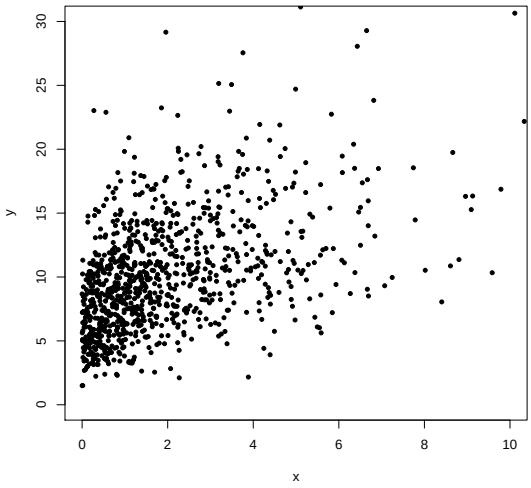
(a) Method 1



(b) Method 2 : Uniform kernel



(c) Method 2 : Triangular kernel



(d) Generate samples from true density

Figure 4: Samples generated from different methods.

References

B. Efron. Bootstrap methods: Another look at the jackknife. *Ann. Statist.*, 7(1):1–26, 01 1979. doi: 10.1214/aos/1176344552. URL `https://doi.org/10.1214/aos/1176344552`.

whuber. How can i draw a value randomly from a kernel density estimate? `https://stats.stackexchange.com/questions/321542/how-can-i-draw-a-value-randomly-from-a-kernel-density-estimate/321726`, 2018. Accessed: 2018-01-05.