

Approximating the Sum of Independent Non-Identical Binomial Random Variables with `sinib`

by Boxiang Liu

Abstract The distribution of sum of independent non-identical binomial random variables is frequently encountered in areas such as genomics, healthcare, and operations research. Analytical solutions to the density and distribution are usually cumbersome to find and difficult to compute. Several studies have proposed approximations to the distribution, and among these is the saddlepoint approximation. In this paper, I implemented the saddlepoint approximation in the `sinib` package. I provide two examples to illustrate its usage. One example uses simulated data while the other uses real-world healthcare data. The `sinib` package addresses the gap between the theory and the implementation of approximating the sum of independent non-identical binomials.

1 Introduction

Convolution of independent non-identical binomial random variables appears in a variety of applications, such as analysis of variant-region overlap in genomics (Schmidt et al. (2015)), calculation of bundle compliance statistics in health care organizations (Benneyan and Taşeli (2010)), and reliability analysis in operations research (Kotz and Johnson (1984)).

Computing the exact density and distribution of the convolution of non-identical independent binomial random variables requires enumeration of all possible combinations of binomial outcome that satisfies the totality constraint. However, analytical solutions are often difficult to find for sum of greater than two binomial random variables. Several studies have proposed approximate solutions (Johnson et al. (2005); Jolayemi (1992)). In particular, Eisinga et al. examined the saddlepoint approximation, and compared them to exact solutions (Eisinga et al. (2013)). They note that, in practice, these approximations are often as good as the exact solution and are easy to implement in most statistical software.

Despite the theoretical development of aforementioned approximate solutions, a software implementation in R is still lacking. The `stats` package includes functions for the most frequently used distribution such as `dbinom` and `dnorm`. In addition, it also includes less frequent distributions such as `pbirthday`. However, it does not contain functions for distribution of sum of independent non-identical binomial. In this paper, I address this deficiency by implementing a saddlepoint approximation in the open source package `sinib` (Sum of independent non-identical binomial random variables). The package provides the standard suite of probability (`psinib`), distribution (`dsinib`), quantile (`qsinib`), and random variable (`rsinib`) functions. The package is accompanied by a detailed documentation, and can be easily integrated into existing applications.

The remainder of this paper is organized as follows, section 2 formulate the distribution of sum of independent non-identical binomial variables. Section 3 discuss saddlepoint approximation. Section 4 describes the design and implementation of the package. Section 5 uses two examples to illustrate the usage of `sinib`. Section 6 draws final conclusion and discusses possible future development of the package.

2 Overview of the distribution

Suppose X_1, \dots, X_m are independent non-identical binomial random variables, and $S_m = \sum_{i=1}^m X_i$. We are interested in finding the distribution of S_m .

$$P(S_m = s) = P(X_1 + X_2 + \dots + X_m = s) \quad (1)$$

In the special case of $m = 2$, the probability simplifies to

$$P(S_2 = s) = P(X_1 + X_2 = s) = \sum_{i=0}^n P(X_1 = i)P(X_2 = s - i) \quad (2)$$

Computation of the exact distribution often involves enumerating all possible combinations of each variable that sums to a given value, which becomes infeasible when n is large. A fast recursion method

to compute the exact distribution has been proposed (Butler and Stephens (2016); Arthur Woodward and Palmer (1997)). The algorithm is as follows:

1. Compute the exact distribution of each X_i .
2. Calculate the distribution of $S_2 = X_1 + X_2$ using equation 2 and cache the result.
3. Calculate $S_r = S_{r-1} + X_i$ for $r = 3, 4, \dots, m$.

Although the recursion speeds up the calculation, studies has shown that it may be numerically unstable due to round-off error in computing $P(S_r = 0)$ if r is large (Yili; Eisinga et al. (2013)). Therefore, approximation methods are still widely used.

3 Saddlepoint approximation

The saddlepoint approximation, first proposed by Daniels (1954) and later extended by Lugannani and Rice (1980), provides highly accurate approximations for the probability and cumulative density function of any distribution based on its moment generating function. In brief, let $M(s)$ be the moment generating function, and $K(s) = \log(M(s))$ be the cumulant generating function. The saddlepoint approximation to the PDF of the distribution is given as:

$$\hat{f}(s) = \frac{\exp(K(\hat{u}) - \hat{u}s)}{\sqrt{2\pi K''(\hat{u})}} \quad (3)$$

where \hat{u} is the unique value that satisfies $K(\hat{u}) = s$.

I now apply saddlepoint approximation to sum of indepedent non-identical binomial random variables (a detailed review is written by Eisinga et al. (2013)). Suppose that $X_i \sim \text{Binomial}(n_i, p_i)$ for $i = 1, 2, \dots, m$. The cumulant generating function of $S_m = \sum X_i$ is:

$$K(u) = \sum_{i=1}^m n_i \ln(1 - p_i + p_i \exp(u)) \quad (4)$$

The first- and second-order derivative of $K(u)$ are:

$$K'(u) = \sum_{i=1}^m n_i q_i \quad (5)$$

$$K''(u) = \sum_{i=1}^m n_i q_i (1 - q_i) \quad (6)$$

where $q_i = p_i \exp(u) / (1 - p_i + p_i \exp(u))$.

The saddlepoint of \hat{u} can be obtained by solving $K'(\hat{u}) = x$. A unique root can always be found because $K(u)$ is strictly convex and therefore $K'(u)$ is monotonically increasing on the real line.

The above shows the first-order approximation of the distribution. I can increase the accuracy of the approximation by adding a second-order correction term (Daniels (1987)).

$$\hat{P}_2(S = s) = \hat{P}_1(S = s) \left\{ 1 + \frac{K'''(\hat{u})}{8K''(\hat{u})} - \frac{5[K'''(\hat{u})]^2}{24[K''(\hat{u})]^3} \right\} \quad (7)$$

where

$$K'''(\hat{u}) = \sum_{i=1}^m n_i q_i (1 - q_i) (1 - 2q_i)$$

and

$$K''''(\hat{u}) = \sum_{i=1}^m n_i q_i (1 - q_i) [1 - 6q_i (1 - q_i)]$$

Although the saddlepoint equation cannot be solved at boundaries $s = 0$ and $s = \sum_{i=1}^m n_i$, I can compute the exact probability at boundary with ease:

$$P(S = 0) = \prod_{i=1}^m (1 - p_i)^{n_i} \quad (8)$$

$$P(S = n_1 + n_2 + \dots + n_m) = \prod_{i=1}^m p_i^{n_i} \quad (9)$$

Incorporation of boundary solutions into the approximation gives:

$$\bar{P}(S = s) = \begin{cases} P(S = 0), & s = 0 \\ [1 - P(S = 0) - P(S = \sum n_i)] \frac{\hat{P}_2(S=s)}{\sum_{i=1}^m n_i \hat{P}_2(S=i)}, & 0 < s < \sum n_i \\ P(S = \sum n_i), & s = \sum n_i \end{cases} \quad (10)$$

I implemented equation 10 as the final approximation of the probability density function.

For the cumulative density, Daniels (1987) gave the following approximator:

$$\hat{P}_3(x) = \begin{cases} \Phi(\hat{w}) + \phi(\hat{w})\left(\frac{1}{\hat{w}} - \frac{1}{u_1}\right), & \text{if } x \neq \mu \\ \frac{1}{2} - \frac{1}{\sqrt{2\pi}} \left[\frac{K'''(0)}{6K''(0)^{3/2}} - \frac{1}{2\sqrt{K''(0)}} \right], & \text{if } x = \mu \end{cases} \quad (11)$$

where $\hat{w} = \text{sign}(\hat{u})[2\hat{u}K'(\hat{u}) - 2K(\hat{u})]^{1/2}$ and $u_1 = [1 - \exp(-\hat{u})][K''(\hat{u})]^{1/2}$. The letters Φ and ϕ denotes the probability and density of the standard normal distribution.

The accuracy can be improved by adding a second-order continuity correction given as follows:

$$\hat{P}_4(S \geq s) = \hat{P}_3(S \geq s) - \phi(\hat{w}) \left[\frac{1}{\hat{u}_2} \left(\frac{\hat{\kappa}_4}{8} - \frac{5\hat{\kappa}_3^2}{24} \right) - \frac{1}{\hat{u}_2^3} - \frac{\hat{\kappa}_3}{2\hat{u}_2^2} + \frac{1}{\hat{w}^3} \right] \quad (12)$$

where $\hat{u}_2 = \hat{u}[K''(\hat{u})]^{1/2}$, $\hat{\kappa}_3 = K'''(\hat{u})[K''(\hat{u})]^{-3/2}$, and $\hat{\kappa}_4 = K''''(\hat{u})[K''(\hat{u})]^{-2}$.

I implement equation 12 in the package to approximate the cumulative distribution.

4 The sinib package

The package implementation used only functions in base R and the stats package to minimize compatibility issues. The arguments for the functions in the **sinib** package are designed to have similar meaning to those in the **stats** package, thereby minimizing the learning required. To illustrate, I compare the arguments of the ***binom** and the ***sinib** functions.

From the help page of the binomial distribution:

- x, q: vector of quantiles.
- p: vector of probabilities.
- n: number of observations.
- size: number of trials.
- prob: probability of success on each trial.
- log, log.p: logical; if TRUE, probabilities p are given as log(p).
- lower.tail: logical; if TRUE (default), probabilities are $P[X \leq x]$, otherwise, $P[X > x]$.

Since the distribution of sum of independent non-identical binomials is defined by a series of trial and probability pairs (each pair for one constituent binomial), it was necessary to redefine these arguments in the ***sinib** functions. Therefore, two arguments are different,

- size: integer vector of number of trials.
- prob: numeric vector of success probabilities.

All other arguments remain the same. It is worth noting that when size and prob arguments are given as vectors of length 1, the ***sinib** functions reduces to ***binom** functions:

```
# Binomial:
dbinom(x = 1, size = 2, prob = 0.5)
[1] 0.5
```

```
# Sum of binomials:
library(sinib)
dsinib(x = 1, size = 2, prob = 0.5)
[1] 0.5
```

With that in mind, the next section shows a few examples to illustrate the usage of **sinib**.

5 Two examples

Sum of two binomials

I use two examples to illustrate the use of this package, starting from the simplest case of two binomial random variables with the same mean but different sizes, $X \sim \text{Bin}(n, p)$ and $Y \sim \text{Bin}(m, p)$. The distribution of $S = X + Y$ has an analytical solution, $S \sim \text{Bin}(m + n, p)$. We can therefore use different combinations of (m, n, p) to assess the accuracy of the saddlepoint approximation of the CDF. I use $m, n = \{10, 100, 1000\}$ and $p = \{0.1, 0.5, 0.9\}$ to assess the approximation. The ranges of m and n are chosen to be large and the value of p are chosen to represent both boundaries.

```
library(foreach)
library(data.table)
library(cowplot)
library(sinib)

# Comparison of CDF between truth and approximation:
data=foreach(m=c(10,100,1000),.combine='rbind')%do%{
  foreach(n=c(10,100,1000),.combine='rbind')%do%{
    foreach(p=c(0.1, 0.5, 0.9),.combine='rbind')%do%{
      a=pbinom(q=0:(m+n),size=(m+n),prob = p)
      b=psinib(q=0:(m+n),size=as.integer(c(m,n)),prob=c(p,p))
      data.table(s=seq_along(a),truth=a,approx=b,m=m,n=n,p=p)
    }
  }
}

ggplot(data,aes(x=truth,y=approx,color=as.character(p)))+
  geom_point(alpha=0.5)+
  facet_grid(m~n)+
  theme_bw()+
  scale_color_discrete(name='prob')+
  xlab('Truth')+ylab('Approximation')
```

Figure 1 shows that the approximations are close to the grouths across a range of parameters. We can further examine the accuracy by looking at the differences between the approximations and the grounth truths.

```
p2=ggplot(data[m==100&n==100],
  aes(x=s,y=truth-approx,color=as.character(p)))+
  geom_point(alpha=0.5)+theme_bw()+
  scale_color_discrete(name='prob')+
  xlab('Quantile')+ylab('Truth-Approximation')+
  geom_vline(xintercept=200*0.5,color='green',linetype='longdash')+
  geom_vline(xintercept=200*0.1,color='red',linetype='longdash')+
  geom_vline(xintercept=200*0.9,color='blue',linetype='longdash')
```

Figure 2 shows the difference between the truth and the approximation for $m = n = 100$. The dashed line indicate the mean of each random variable. The approximations perform well expect around the mean, but the large deviation is less than $5e-4$.

Next we examine the approximation for the PDF.

```
# Comparison of PDF between truth and approximation:
data=foreach(m=c(10,100,1000),.combine='rbind')%do%{
  foreach(n=c(10,100,1000),.combine='rbind')%do%{
    foreach(p=c(0.1, 0.5, 0.9),.combine='rbind')%do%{
      a=dbinom(x=0:(m+n),size=(m+n),prob = p)
      b=dsinib(x=0:(m+n),size=as.integer(c(m,n)),prob=c(p,p))
      data.table(s=seq_along(a),truth=a,approx=b,m=m,n=n,p=p)
    }
  }
}
```

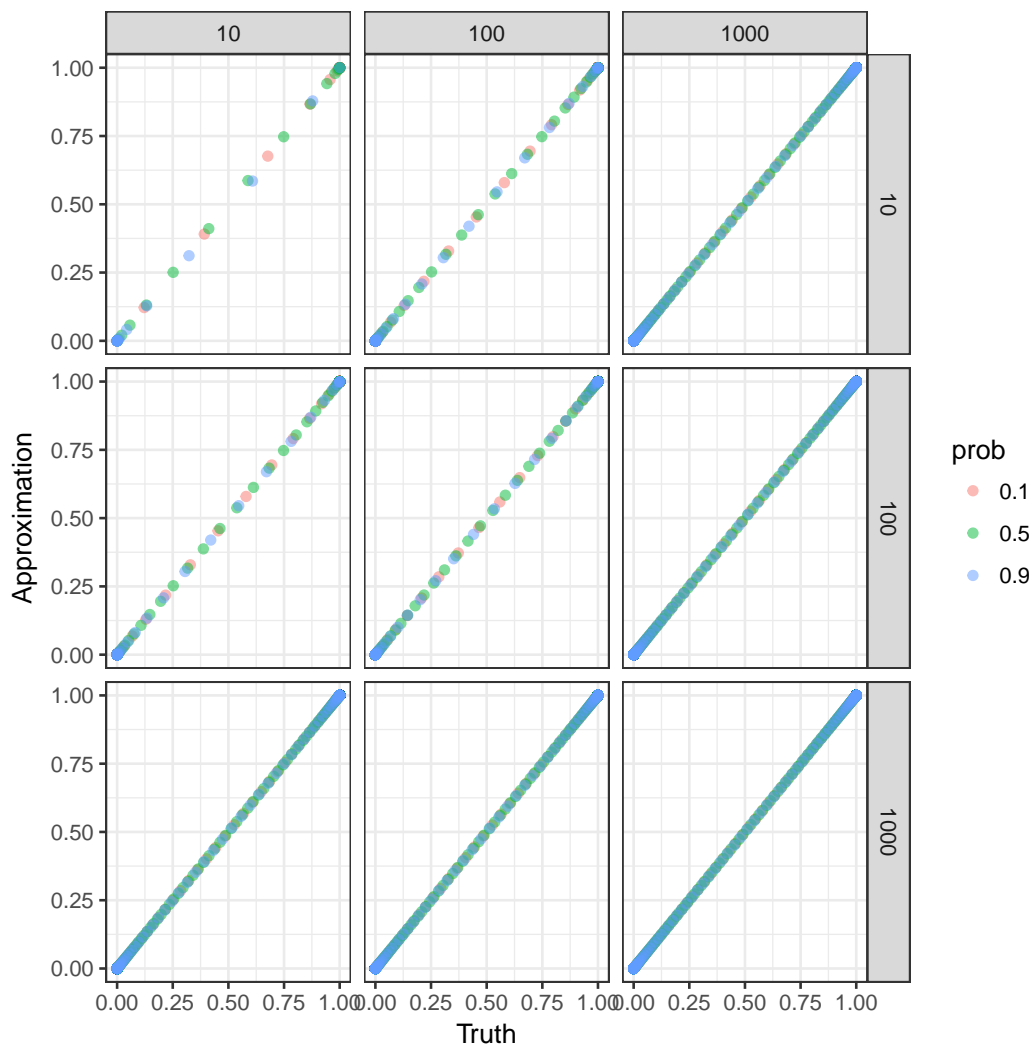


Figure 1: Comparison of CDF between truth and approximation

```
ggplot(data, aes(x=truth, y=approx, color=as.character(p)))+
  geom_point(alpha=0.5)+facet_grid(m~n)+
  theme_bw()+scale_color_discrete(name='prob')+
  xlab('Truth')+ylab('Approximation')
```

Figure 3 shows that the approximations and the ground truths are similar. Once again, we further examine the difference between the truth and the approximation. One example for $m = n = 100$ is shown in figure 4. As expected, the approximation degrades around the mean, but the largest deviation is less than $4e-7$.

```
p4=ggplot(data[m==100&n==100],
  aes(x=s, y=truth-approx, color=as.character(p)))+
  geom_point(alpha=0.5)+theme_bw()+
  scale_color_discrete(name='prob')+
  xlab('Quantile')+ylab('Truth-Approximation')+
  geom_vline(xintercept=200*0.5, color='green', linetype='longdash')+
  geom_vline(xintercept=200*0.1, color='red', linetype='longdash')+
  geom_vline(xintercept=200*0.9, color='blue', linetype='longdash')
```

Healthcare monitoring

In the second example, I use a health system monitoring dataset by [Benneyan and Taşeli \(2010\)](#). Suppose the following values for n_i and p_i .

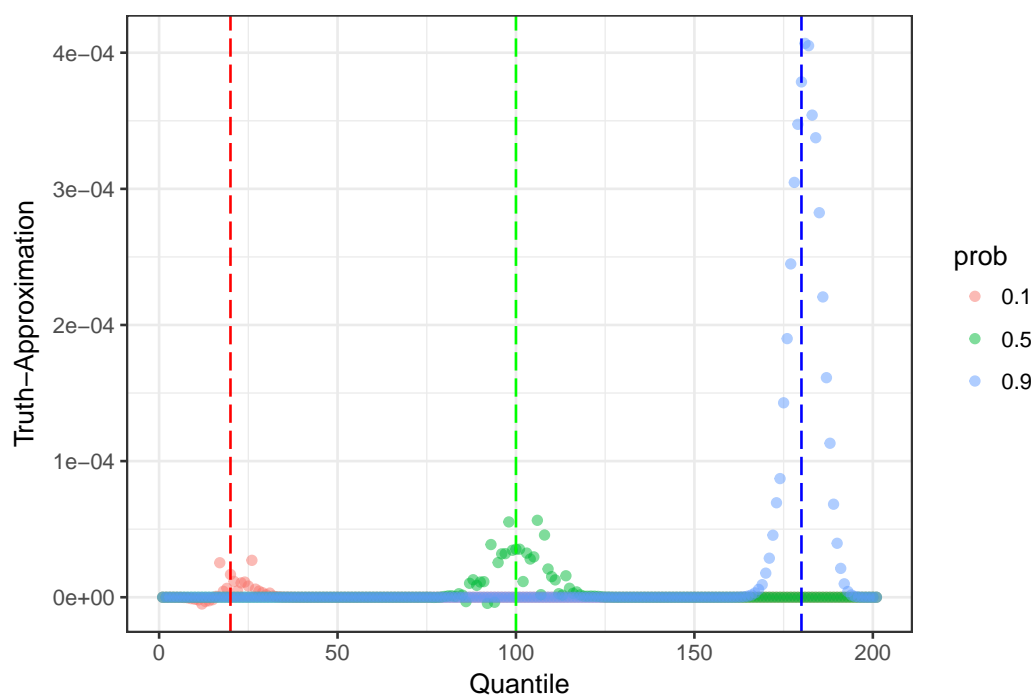


Figure 2: Difference in CDF between truth and approximation

```
size=as.integer(c(12, 14, 4, 2, 20, 17, 11, 1, 8, 11))
prob=c(0.074, 0.039, 0.095, 0.039, 0.053, 0.043, 0.067, 0.018, 0.099, 0.045)
```

Since it is difficult to find an analytical solution to the density, I estimated the density based on $1e8$ trials and use it as the ground truth. I then compare simulations with $1e3$, $1e4$, $1e5$, and $1e6$ trials, as well as the saddlepoint approximation to the ground truth.

```
# Sinib:
approx=dsinib(0:sum(size),size,prob)
approx=data.frame(s=0:sum(size),pdf=approx,type='saddlepoint')

# Simulation:
data=foreach(n_sim=10^c(3:6,8),.combine='rbind')%do%{
  n_binom=length(prob)
  set.seed(42)
  mat=matrix(rbinom(n_sim*n_binom,size,prob),nrow=n_binom,ncol=n_sim)

  S=colSums(mat)
  sim=sapply(X = 0:sum(size), FUN = function(x) {sum(S==x)/length(S)})
  data.table(s=0:sum(size),pdf=sim,type=n_sim)
}

data=rbind(data,approx)
truth=data[type=='1e+08',]

merged=merge(truth[,list(s,pdf)],data,by='s',suffixes=c('_truth','_approx'))
merged=merged[type!='1e+08',]

ggplot(merged,aes(pdf_truth,pdf_approx))+
  geom_point()+
  facet_grid(~type)+
  geom_abline(intercept=0,slope=1)+
  theme_bw()+
  xlab('Truth')+ylab('Approx')
```

Figure 5 shows that $1e6$ simulations and saddlepoint approximation are visually indistinguishable from the ground truth, while smaller sizes shows clear deviations from the truth. To be precise, I

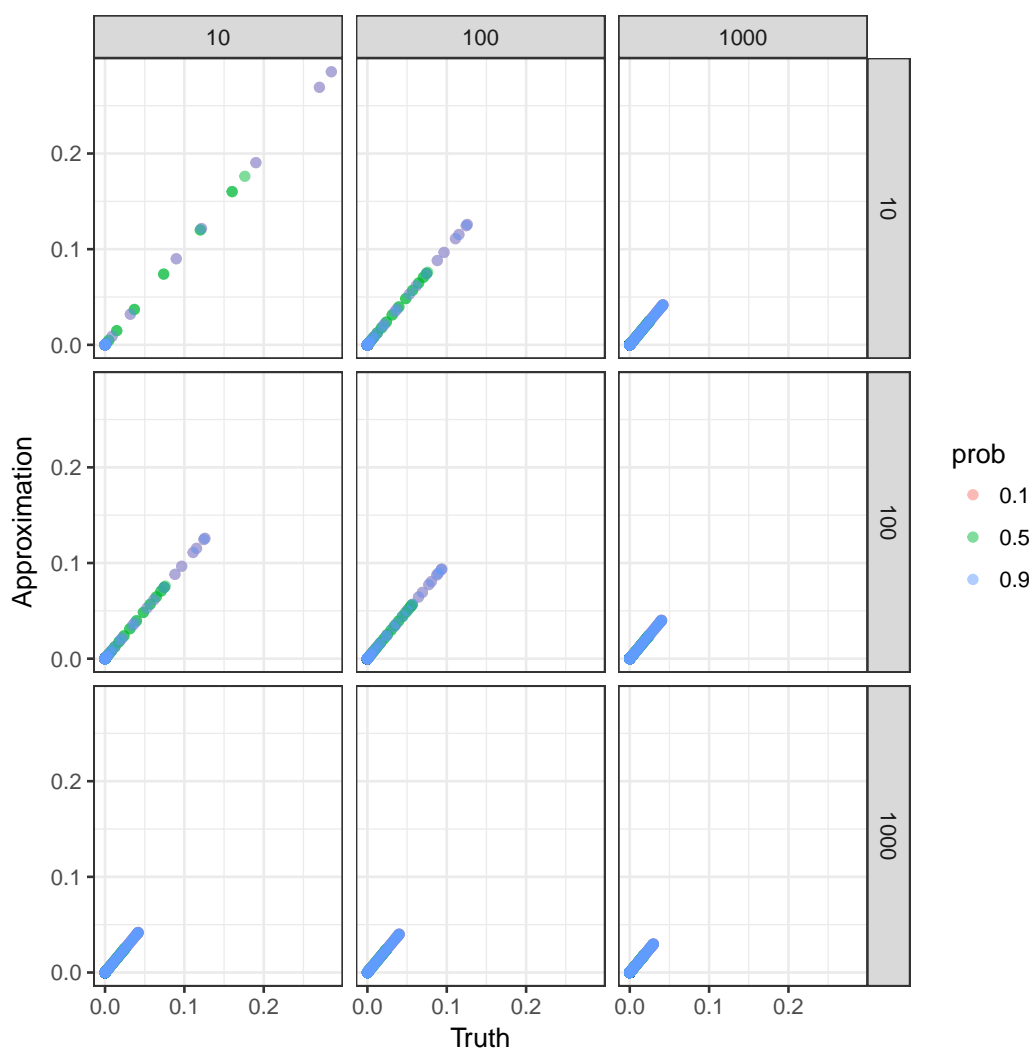


Figure 3: Comparison of PDF between truth and approximation

plotted the difference in PDF between the truth and the approximation.

```
merged[,diff:=pdf_truth-pdf_approx]
```

```
ggplot(merged,aes(s,diff))+
  geom_point()+
  facet_grid(~type)+
  theme_bw()+
  xlab('Quantile')+ylab('Truth-Approx')
```

Figure 6 shows that the saddlepoint method and simulation with $1e6$ both provide good approximations, while simulations of smaller sizes show clear deviations. On the other hand, saddlepoint approximation is 5 times faster than simulation of $1e6$ trials.

```
ptm=proc.time()
n_binom=length(prob)
mat=matrix(rbinom(n_sim*n_binom,size,prob),nrow=n_binom,ncol=n_sim)
S=colSums(mat)
sim=sapply(X = 0:size, FUN = function(x) {sum(S==x)/length(S)})
proc.time()-ptm
#   user  system elapsed
#  1.008   0.153   1.173

ptm=proc.time()
approx=dsinib(0:size,size,prob)
```

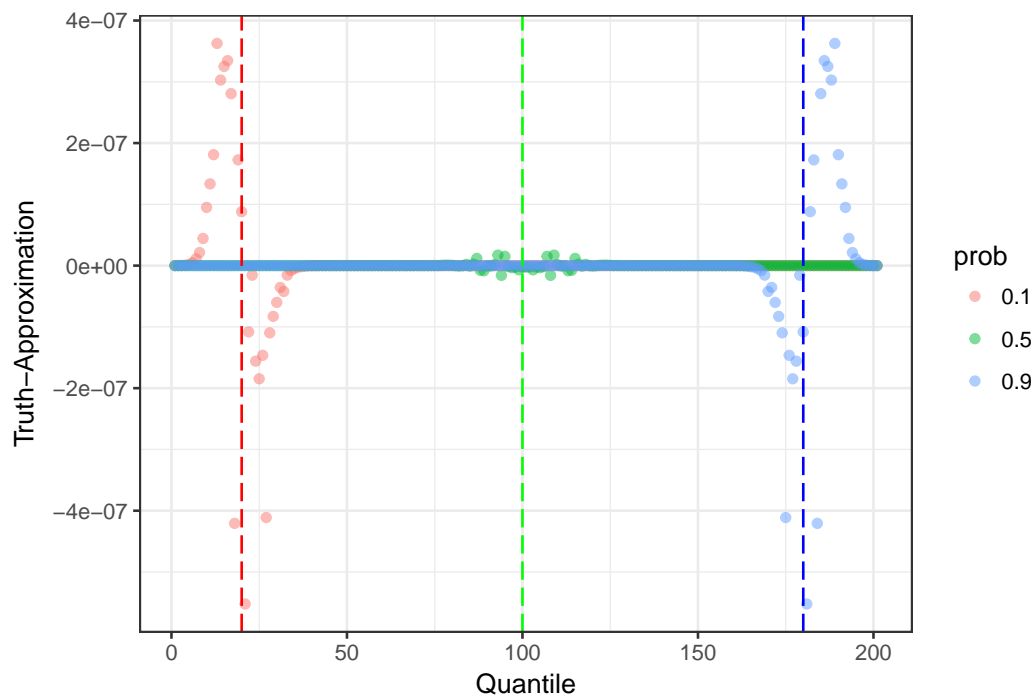


Figure 4: Difference in PDF between truth and approximation

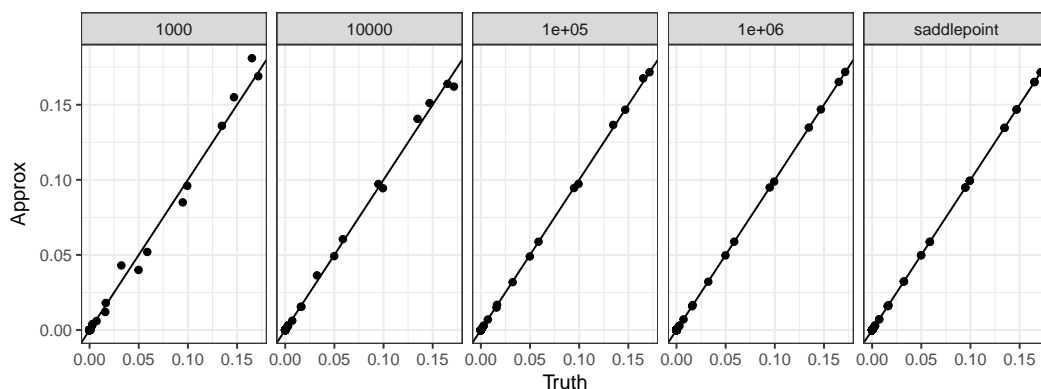


Figure 5: Comparison of PDF between truth and approximation

```
proc.time()-ptm
#   user  system elapsed
# 0.025   0.215   0.239
```

6 Conclusion and future directions

In this paper, I presented an implementation of the saddlepoint method to approximate the distribution of sum of independent and non-identical binomials. I assessed the accuracy of the method by, first, comparing it with the analytical solution on a simple case of two binomials, and second, with the simulated ground truth on a real-world dataset appearing in health care monitoring. These assessments suggest that, while saddlepoint approximation deviates from the ground truth around the means, it generally provides a approximation superior to simulation in terms of both speed and accuracy. Overall, the **sinib** package addresses the gap between the theory and implementation on the approximation of sum of indepdent non-identical binomial random variables.

In the future, I aim to explore other approximation methods such as the Kolmogorov approximation and the Pearson curve approximation described by [Butler and Stephens \(2016\)](#).

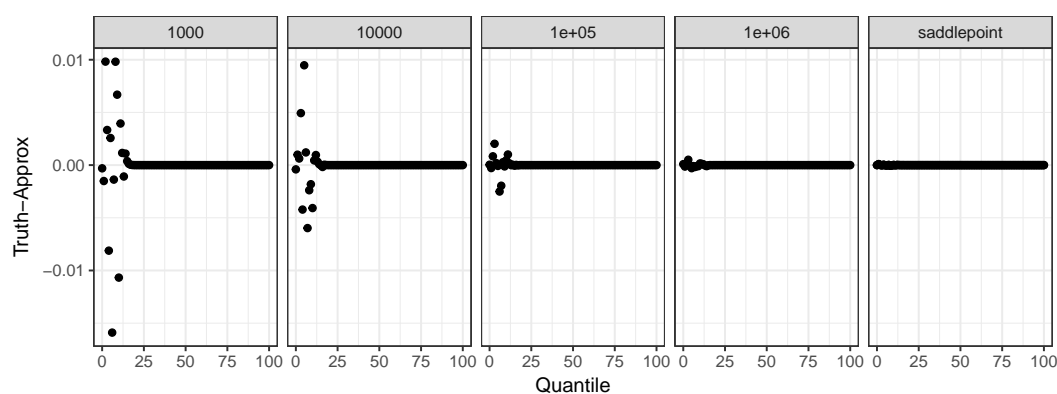


Figure 6: Comparison of PDF between truth and approximation

Bibliography

- J. Arthur Woodward and C. G. S. Palmer. On the exact convolution of discrete random variables. *Applied Mathematics and Computation*, 83(1):69–77, Apr. 1997. [p2]
- J. C. Benneyan and A. Taşeli. Exact and approximate probability distributions of evidence-based bundle composite compliance measures. *Health Care Management Science*, 13(3):193–209, Feb. 2010. [p1, 5]
- K. Butler and M. A. Stephens. The Distribution of a Sum of Independent Binomial Random Variables. *Methodology and Computing in Applied Probability*, 19(2):557–571, Dec. 2016. [p2, 8]
- H. E. Daniels. Saddlepoint Approximations in Statistics. *The Annals of Mathematical Statistics*, 25(4): 631–650, Dec. 1954. [p2]
- H. E. Daniels. Tail Probability Approximations. *International Statistical Review / Revue Internationale de Statistique*, 55(1):37–48, Apr. 1987. [p2, 3]
- R. Eisinga, M. Te Grotenhuis, and B. Pelzer. Saddlepoint approximations for the sum of independent non-identically distributed binomial random variables. *Statistica Neerlandica*, 67(2):190–201, May 2013. [p1, 2]
- N. L. Johnson, A. W. Kemp, and S. Kotz. *Univariate Discrete Distributions*. Johnson/Univariate Discrete Distributions. John Wiley & Sons, Inc., Hoboken, NJ, USA, Jan. 2005. [p1]
- J. K. Jolayemi. A unified approximation scheme for the convolution of independent binomial variables. *Applied Mathematics and Computation*, 49(2-3):269–297, June 1992. [p1]
- S. Kotz and N. L. Johnson. Effects of False and Incomplete Identification of Defective Items on the Reliability of Acceptance Sampling. *Operations Research*, 32(3):575–583, May 1984. [p1]
- R. Lugannani and S. Rice. Saddle Point Approximation for the Distribution of the Sum of Independent Random Variables. *Advances in Applied Probability*, 12(2):475, June 1980. [p2]
- E. M. Schmidt, J. Zhang, W. Zhou, J. Chen, K. L. Mohlke, Y. E. Chen, and C. J. Willer. GREGOR: evaluating global enrichment of trait-associated variants in epigenomic features using a systematic, data-driven approach. *Bioinformatics*, 31(16):2601–2606, Aug. 2015. [p1]
- H. Yili. On Computing the Distribution Function for the Sum of Independent and Non-identical Random Indicators . URL <https://pdfs.semanticscholar.org/fe97/c1358ec01c86cb8bbc4574fa064748f37e94.pdf>. [p2]

Boxiang Liu
Stanford University
300 Pasteur Drive, Stanford, CA
United States
bliu2@stanford.edu