Approximate Normal Random Variable with Binomial Discrete Random Variable

Fan Wang

2021-06-13

Contents

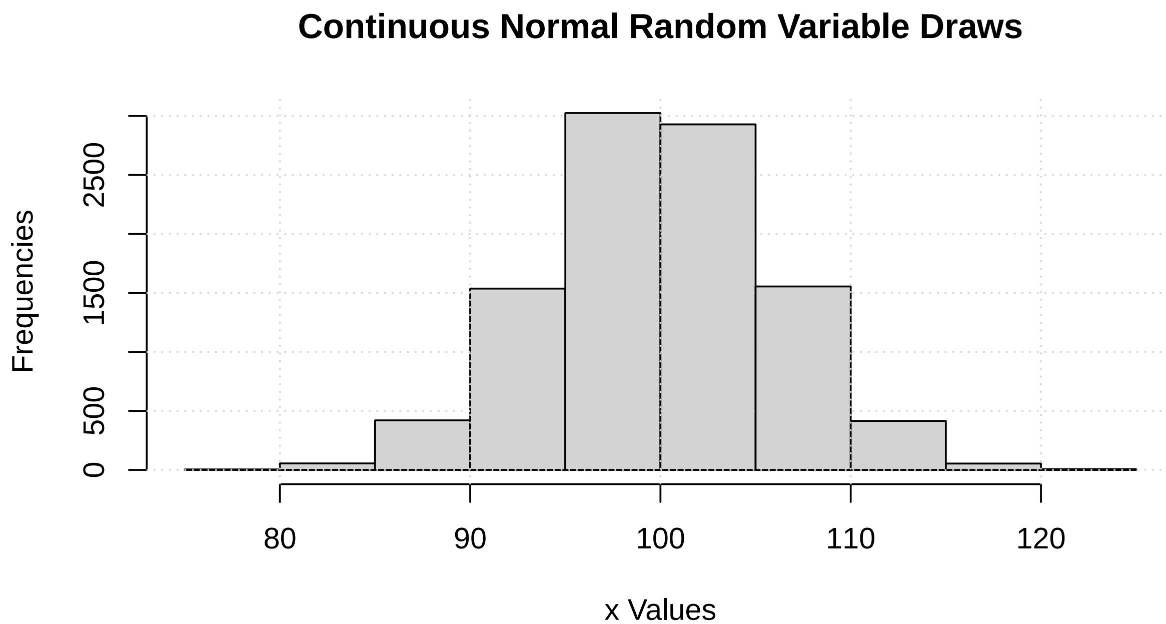
1 Discrete Approximation of Continuous Random Variables

Go to the **RMD**, **R**, **PDF**, or **HTML** version of this file. Go back to fan's REconTools Package, R Code Examples Repository (bookdown site), or Intro Stats with R Repository (bookdown site).

1.1 Use Binomial Discrete Random Variable to Approximate Continuous Normal

First, draw from a Continuous Random Variable. Sample N draws from a normal random variable.

```
# Random normal Data Vector (not equal outcomes)
set.seed(123)
it_sample_N <- 10000
fl_rnorm_mean <- 100
fl_rnorm_sd <- 6
ar_data_rnorm <- rnorm(it_sample_N, mean=fl_rnorm_mean, sd=fl_rnorm_sd)
# Visualize
par(new=FALSE)
hist(ar_data_rnorm, xlab = 'x Values', ylab = 'Frequencies', main = '')
title(main = 'Continuous Normal Random Variable Draws')
grid()
```



We use the binomial to approximate the normal distribution, we know that these relationships are approximately true, where μ and σ are the mean and standard deviations of the normal random variable, and n and p and the number of "trials" and the "probability-of-success" for the binomial distribution:

$$\mu = n \cdot p$$
$$n = \frac{\mu}{p}$$
$$\sigma^2 = n \cdot p \cdot (1-p) = \mu \cdot (1-p)$$

Given these, we have can translate between the normal random variable's parameters and the binomial discrete random variable's parameters:

$$p = 1 - \frac{\sigma^2}{\mu}$$
$$n = \frac{\mu}{1 - \frac{\sigma^2}{\mu}} = \frac{\mu}{\frac{\mu - \sigma^2}{\mu}} = \frac{\mu^2}{\mu - \sigma^2}$$

There are two important aspects to note here:

1. Since p must be positive, this means $\frac{\sigma^2}{\mu} < 1$ and $\sigma^2 < \mu$, which is the condition for the above transformation to work.
2. The binomial discrete random variable will have non-zero mass for very small probability events at the left-tail. These very low outcome events are highly unlikely to be observed or drawn from sampling the continuous random variable. The presence of these left-tail values might impact the computation of certain statistics, for example the Atkinson Index for highly inequality averse planners.

```
# Use binomial to approximate normal
fl_p_binom <- 1 - fl_rnorm_sd^2/fl_rnorm_mean
fl_n_binom <- round(fl_rnorm_mean^2/(fl_rnorm_mean - fl_rnorm_sd^2))
fl_binom_mean <- fl_n_binom*fl_p_binom
```

```
fl_binom_sd <- sqrt(fl_n_binom*fl_p_binom*(1-fl_p_binom))
# Mean and sd
print(paste0('BINOMI mean=', fl_binom_mean))
```

```
## [1] "BINOMI mean=99.84"
```

```
print(paste0('BINOMI mean=', fl_binom_sd))
```

```
## [1] "BINOMI mean=5.99519807846246"
```

```
# drv = discrete random variable
ar_drv_rbinom_xval <- seq(1, fl_n_binom)
ar_drv_rbinom_prob <- dbinom(ar_drv_rbinom_xval, size=fl_n_binom, prob=fl_p_binom)
# ignore weight at x=0
ar_drv_rbinom_prob <- ar_drv_rbinom_prob/sum(ar_drv_rbinom_prob)
```

Visualize the binomial discrete random variable:

```
# graph
par(new=FALSE)
ar_ylim = c(0,1)
ffi_atkinson_ineq(ar_data_rnorm, -1)
```

```
## [1] 0.003615375
```

```
ar_atkinson <- apply(mt_rho, 1, function(row){ffi_atkinson_ineq(ar_data_rnorm, row[1])})
plot(ar_drv_rbinom_xval, ar_drv_rbinom_prob, xlab = 'Binomial x Values', ylab = 'Discrete Probabilitys,
title(main = paste0('Binomial Approximate of Normal Random Variable'),
      sub = paste0('binop=', round(fl_p_binom, 2), ';binon=', round(fl_n_binom, 2), ';binomean=', round
grid()
```