BAN 602: Quantitative Fundamentals

Spring, 2020 Online Lecture Slides – Week 2

CAL STATE EAST BAY

Agenda

- Random variable generation
- Density functions
- Cumulative distribution functions
- Quantile functions
- Sampling from a finite population
- Loops: while and if

Stochastic Simulation

Why simulate?

- We want to understand what a probability model actually does
- We want to understand how our procedure works on a test case
- We want to use a randomized algorithm
- All of these require sampling from distributions

R: Built-in Random Variable Generator Functions

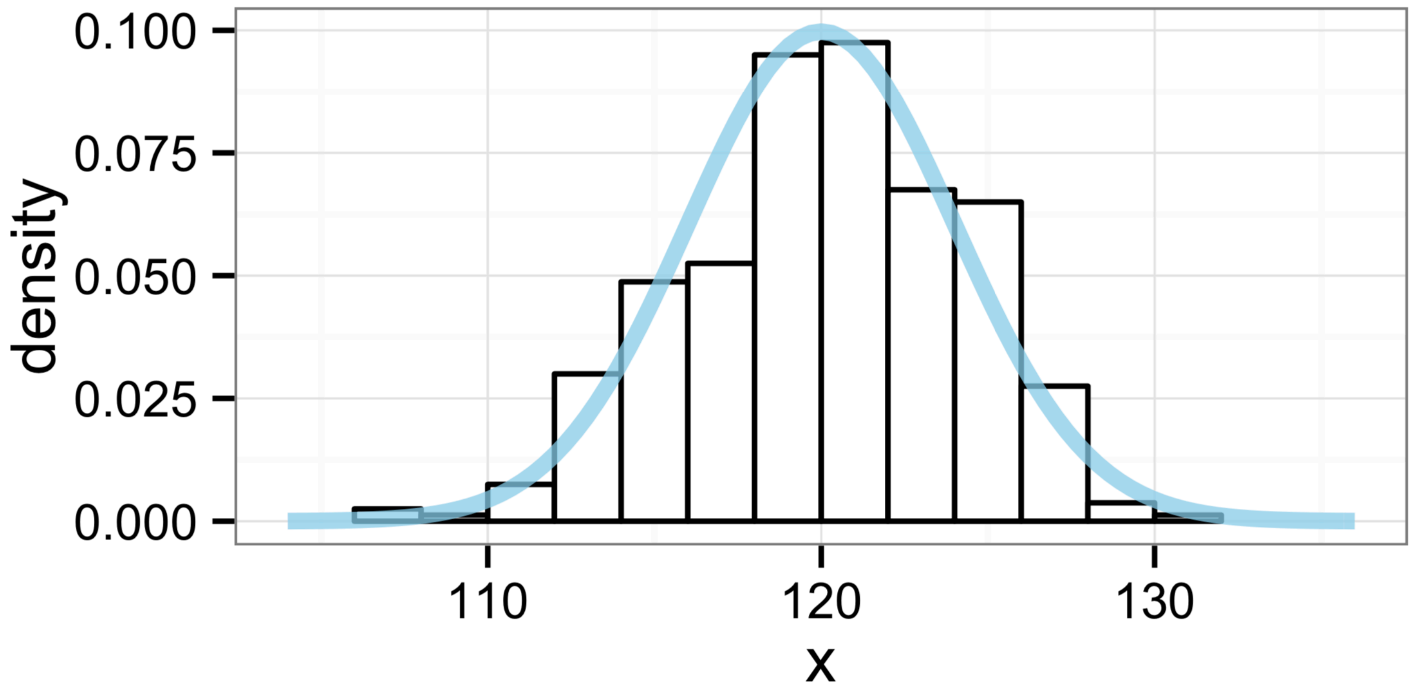
- runif(), rnorm(), rbinom(), rpois(), rexp(), rt(), etc...
- First argument in these functions is always n, number of variables to generate
- Subsequent arguments are parameters to the distribution
- Before you generate a random number, it is a good practice to set a seed using command:

  ```
  set.seed(some_number)
  ```
- If the same seed is used, the same random numbers would be generated which is helpful for replication purposes. For example, date on which the code is being written could be used as the seed.

CAL STATE EAST BAY

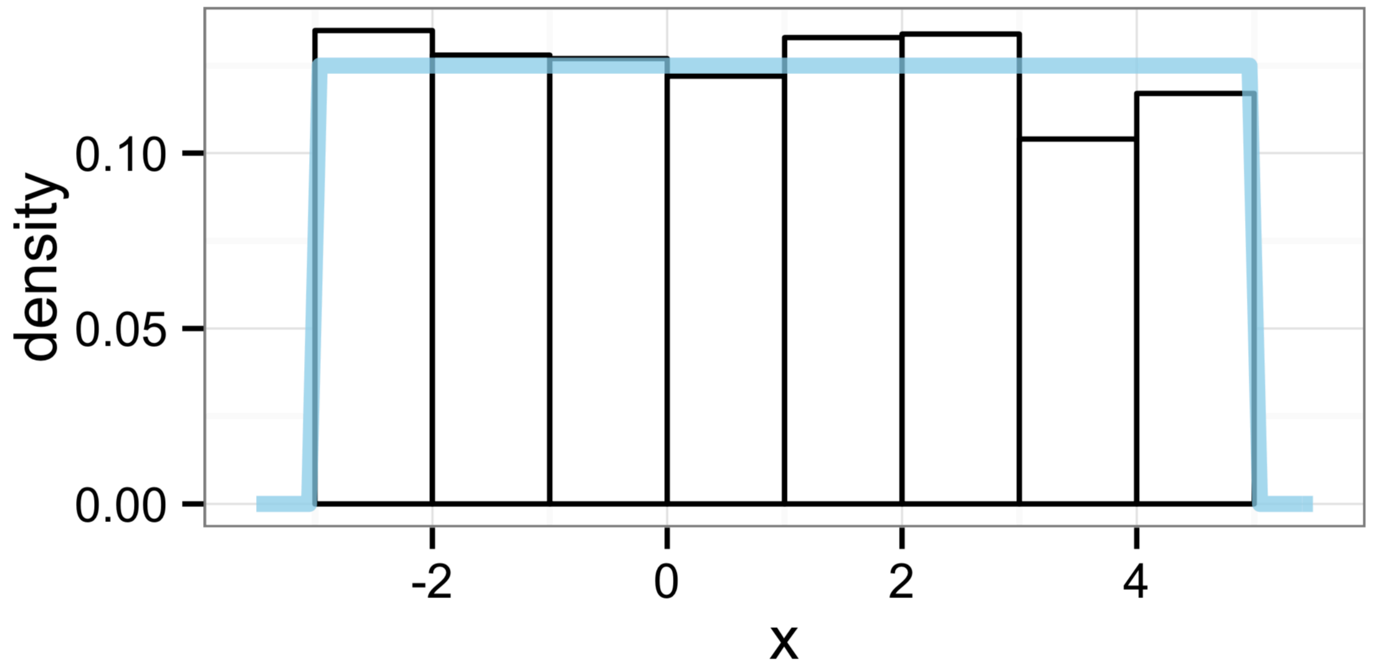
Normal Distribution

```
# Generate 400 random variables from a
# Normal distribution with mean 120 and variance 16

x <- rnorm(n=400, mean=120, sd=4)
```



Uniform Distribution

```
# Generate 1000 random variables from an
# Uniform distribution on [-3, 5]

x <- runif(n=1000, min=-3, max=5)
```



Some more examples

- Generate a random sample of size 100 from a normal distribution of mean 70 and standard deviation 5:

```
observations <- rnorm(100, mean=70, sd=5)
```

- These random numbers generated can be plotted in histogram:

```
hist(observations, breaks = 20)
```

- Random samples (10) can be generated from a Binomial distribution with 20 trials and probability of success 0.5:

```
rbinom(10, size=20, prob=.5)
```

- To generate 5 random numbers from a Poisson distribution with average rate of occurrence 10 per period:

```
rpois(5, lambda=10)
```

R: Random Variable Functions

- Probability density/mass function (PDF) – dunif(), dnorm(), dbinom(), etc...
- Cumulative distribution function (CDF) – punif(), pnorm(), pbinom(), etc...
- Quantile function (inverse CDF) – qunif(), qnorm(), qbinom(), etc...

Examples of Density Function

- Compute the probability of getting x=0 from a normal distribution with mean = 0 and standard deviation =1 (i.e. a standard normal distribution)

```
dnorm(0, mean = 0, sd = 1)
```

- To create a normal distribution and plot it for a vector containing values in a certain range:

```
curve(dnorm(x), from=-4, to=4)
```

- To create a binomial distribution and plot it for number of successes being between 0 and 50 from 50 trials:

```
x <- 0:50
plot(x, dbinom(x, size=50, prob=.33), type="h")
```

- To create a Poisson distribution and plot it for number of occurrences being between 0 and 15 with average number of occurrences being 10 in a given period:

```
x <- 0:15
plot(x, dpois(x, lambda = 10), type="h")
```

Examples of Cumulative Distribution Function

- Describes the probability of getting a certain value or less in given distribution
- Probability of getting a value of 160 or more from a normal distribution with mean 132 and sd 13:

```
pnorm(160, mean=132, sd=13, lower.tail = F)
```

- lower.tail = F returns the area under the bell curve on the right side of x=160. So the above command is same as:

```
1-pnorm(160, mean=132, sd=13)
```

- Probability of getting 18 or fewer number of occurrences from a Poisson distribution with average 12 occurrences in a given interval is:

```
ppois(18, lambda=12)
```

- Probability of getting 16 or more successes from 20 trials of a Binomial distribution with probability of success being 0.5 is:

```
1-pbinom(15, size=20, prob=.5)
```

- Probability of time between two successive arrivals is 2 units of time or less when the mean time between arrivals is 3 units of time:

```
1-pexp(2, rate=1/3)
```

CAL STATE EAST BAY

Examples of Quantile Function

- Quantile function is the inverse of the cumulative distribution function
- The z-score of the pth percentile of the standard normal distribution can be found using qnorm():

  qnorm(.9987)

- This returns the z-value of the 99.87th percentile of standard normal distribution.
- The function qnorm() is the inverse of pnorm().

  qnorm(pnorm(3)) will return 3

- qnorm(0.95, mean=100, sd=10) would return the 95th percentile of a normal distribution with mean=100 and standard deviation=10
- Similarly, to compute the 20th percentile of a Binomial distribution with number of trials=10, and probability of success =1/3 would be:

  qbinom(0.2, 10, 1/3)

- The 20th percentile of a Poisson distribution with average 12 occurrences:

  qpois(0.8, 12, lower.tail=F)

- This is equivalent to: qpois(0.2, 12, lower.tail=T)

R: Sample from a finite population

sample(x, size, replace=FALSE, prob=NULL)

- Built-in function to draw a random sample of given size from x, optionally with replacement and/or weights
- sample(x) – random permutation of x if x is a vector

Examples

```
# Draw a random sample from (1,2,3,4,5)
# of size 10 with replacement


x <- sample(1:5, size=10, replace=TRUE)

print(x) [1] 3 3 5 1 5 5 1 5 2 3

# Draw a random sample from the 50 states # of size 3 without replacement

data(state)
x <- sample(state.name, size=3, replace=FALSE)
print(x)

[1] "Arkansas" "Illinois" "Rhode Island"
```

Caution

```
# When x is a single number, sample()
# behaves differently. Treats the population as
# 1:x for convenience

sample(100, size=3)

[1] 100 77 26

# equivalently...

sample(1:100, size=3)
```

Be careful when x is a single number – this “convenient” behavior of sample() can cause headaches

R: while loops

- Repeatedly do something while condition is TRUE

```
while(condition) {
        # DO SOMETHING
}
```

- **Example:**

```
cointoss <- function() {
                x <- sample(c('H', 'T'), size = 1)
                return(x)
            }

# Wait for heads
while(cointoss() != 'H') {
        print("Darn, tails")
}
print("Yay, heads!")
```

R: if-else statement

- ***Do something IF condition is TRUE:***

```
if(condition) {
        # DO SOMETHING
}
```

- ***Do something IF condition is TRUE, otherwise do something ELSE:***

```
if(condition) {
        # DO SOMETHING
} else {
        # DO SOMETHING ELSE
}
```

- **Example:**

```
if(cointoss() == 'H') {
  print("Heads, you wash the dishes")
} else {
        print("Tails, I wash the dishes")
}
```

CAL STATE EAST BAY