qbs120_ps3_correction_gibran

Gibran Erlangga

10/10/2021

Question 1

a) my original solution was correct.
b) The values remain the same. $Var(Z + X) = Var(Z - X)$ since $Var(bX) = b^2 Var(X)$
c) my original solution was correct.
d) my original solution was correct.
e) my original solution was correct.

Question 2

a) my original solution was correct.
b) When calling prcomp() with scale=T and center=T, we are standardizing the input data (though standardizing with estimates of the mean and SD rather than the population values as in this problem). The result is that the eigenvalue decomposition is performed on the sample correlation matrix rather than the sample covariance matrix (i.e., the sample covariance matrix computed on standardized data is, per this problem, the sample correlation matrix). This can be beneficial when we want to capture the pattern of correlation between variables rather than the pattern of covariance which, by definition, will be dominated by the variables with high variance.

Question 3

a) Let the number of offspring in the second generation be represented by the random variable N. From the problem statement we know that:

$$E[N] = \mu$$
$$Var(N) = \sigma^2$$

Let the number of offspring of each second generation organism be represented by the random variable $N_{2_1}, ..., N_{2_N}$. From the problem statement we know that:

$$E[N_{2_i}] = \mu$$

$$Var(N_{2_i}) = \sigma^2$$

The total number of offspring in the third generation is given by a third random variable T_3 defined as:

$$T_3 = \sum_{i=1}^{N} N_{2_i}$$

The goal is to find the expected number of offspring in the third generation or $E[T_3]$, which takes the following value as per the Law of Total Expectation:

$$E[T_3] = E_N[E_{T3}[T3|N]]$$

If the initial number of offspring were fixed, $N = n, E[T3|N = n] = nE[N_{2_i}]$. Therefore, for a variable number of initial offspring, $E[T3|N] = NE[N_{2_i}]$.

$$\begin{aligned} E[T3] &= E[NE[N_{2_i}]] \\ &= E[N]E[N_{2_i}] \\ &= \mu\ \mu \\ &= \mu^2 \end{aligned}$$

b) To find the variance of T, we can use the formula:

$$Var(Y) = Var(E[Y|X]) + E[Var(Y|X)]$$

for this specific problem, the formula becomes:

$$Var(T3) = Var(E[T3|N]) + E[Var(T3|N)]$$

As specified above, $E[T3|N] = NE[N_{2_i}]$. Because $Var(T3|N = n) = Var(\sum_{i=1}^{n} N_{2_i}) = nVar(N_{2_i}), Var(T3|N) = NVar(N_{2_i})$. Given these formulas, the variance of T3 can be defined as:

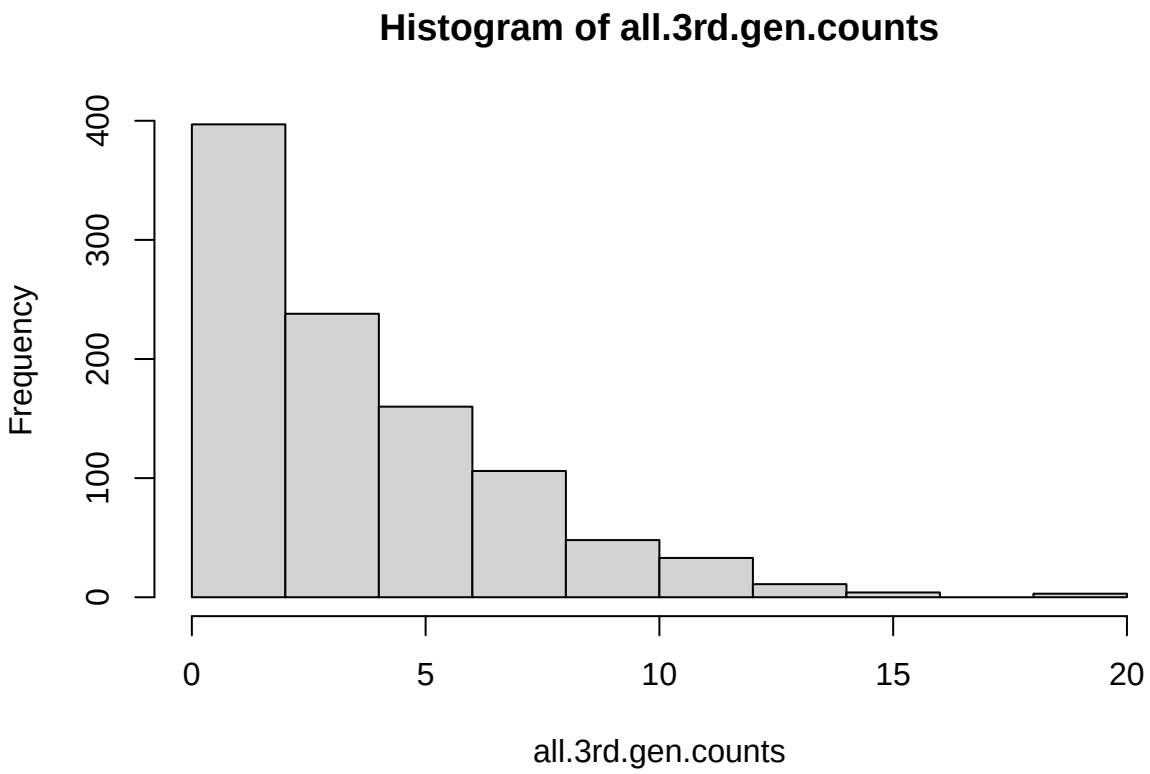
$$\begin{aligned} Var(T3) &= Var(E[T3|N]) + E[Var(T3|N)] \\ &= Var(NE[N_{2_i}]) + E[NVar(N_{2_i})] \\ &= E[N_{2_i}]^2 Var(N) + Var[N_{2_i}]E[N] \\ &= \mu^2\sigma^2 + \sigma^2\mu \\ &= \mu\sigma^2(\mu + 1) \end{aligned}$$

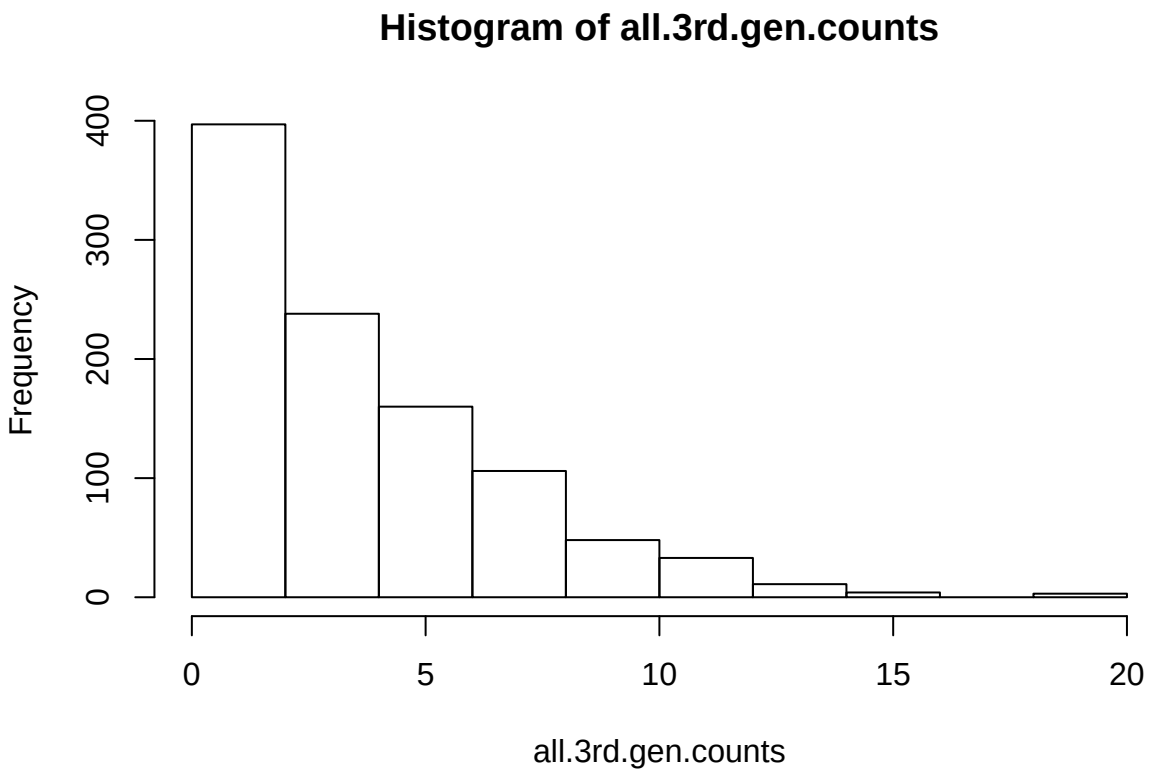
c) Code:

```
 simThirdGen = function() {
   # simulate 2 generation
   num.2 = rpois(n=1,lambda=2)
   # For each member of the second, simulate children
   num.3=0
 if (num.2 == 0) {
 return (0)
}
 for (i in 1:num.2) {
   num.3 = num.3 + rpois(n=1,lambda=2)
 }
 return (num.3)
 }

# simulate for 1000 separate populations:
num.sims=1000
all.3rd.gen.counts = rep(0, num.sims)
for (i in 1:num.sims) {
  all.3rd.gen.counts[i] = simThirdGen()
}

plot(hist(all.3rd.gen.counts))
```


Histogram of all.3rd.gen.counts
Frequency
0
100
200
300
400
0
5
10
15
20
all.3rd.gen.counts



Expectation and variance of number of members in the 3^rd generation:

```
(est.exp = mean(all.3rd.gen.counts))
```

```
## [1] 3.935
```

```
(est.var = var(all.3rd.gen.counts))
```

```
## [1] 11.43821
```

To compare with the results from parts a) and b) we note that the variance and expected value for a Poisson RV are both $\lambda = 2$.

$$\begin{aligned} E[T3] &= \mu^2 \\ &= \lambda^2 \\ &= 4 \end{aligned}$$

$$\begin{aligned} Var[T3] &= \mu\sigma^2(\mu + 1) \\ &= \lambda\lambda(\lambda + 1) \\ &= 2 * 2(2 + 1) \\ &= 12 \end{aligned}$$

We can observe that the estimates are fairly close to the true values. Since the estimates are themselves RVs, we expect some variance around their expected values (which equal the true values of 4 and 12). This type of validation via simulation can be a useful tool for checking complex analytical calculations.

Question 4

Optional

Question 5

my original solution was correct.

Question 6

Optional

Question 7

a) Based on CLT, the sum of independent random variables, S_n defined as:

$$\lim_{n\to\infty} P(\frac{S_n - n\mu}{\sigma\sqrt{n}} \leq x) = \Phi(x)$$

In this case, we are given the pdf of the X_i. The expectation of each X can be found as:

$$\begin{aligned} E[X_n] &= \int_0^1 xf(x)\ dx \\ &= \int_0^1 xf(x)\ dx \\ &= [3x^4/4]_0^1 \\ &= 3/4 \end{aligned}$$

The variance of each X can be defined as:

$$\begin{aligned} Var(X_n) &= E[X_n^2] - E[X_n]^2 \\ &= \int_0^1 3x^4\ dx - E[X_n]^2 \\ &= 3/5 - 9/16 \\ &= 0.0375 \end{aligned}$$

The standard deviation of each X can be defined as:

$$\begin{aligned} \sigma_{X_n} &= \sqrt{Var(X_n)} \\ &= \sqrt{0.0375} \\ &= 0.1937 \end{aligned}$$

Given these, $P(S \leq 14)$ can be computed as:

$$\begin{aligned} P(S \leq 14) &= P((S - 20 * 3/4)/(0.1937 * \sqrt{20}) \leq (14 - 20 * 3/4)/(0.1937 * \sqrt{20})) \\ &\approx \Phi(-1.154) \\ &= 0.1242 \end{aligned}$$

In other words, if the expectation on each trial is 3/4, there is only small chance that the sum of 20 such trials will be less than 14.

b) For $P(S \leq 15)$, the numerator of the CLT-based probability equation becomes 0, i.e., $15 - 3/4 * 20 = 0$, irrespective of the variance of X_n. So, the probability becomes $\Phi(0)$. Since the normal distribution is symmetric, we know that this probability must be 0.5, i.e., we don't need the implementation of pnorm() in R. So, two simplifications: 1) don't need to compute $Var(X_n)$ and 2) don't need to call pnorm().

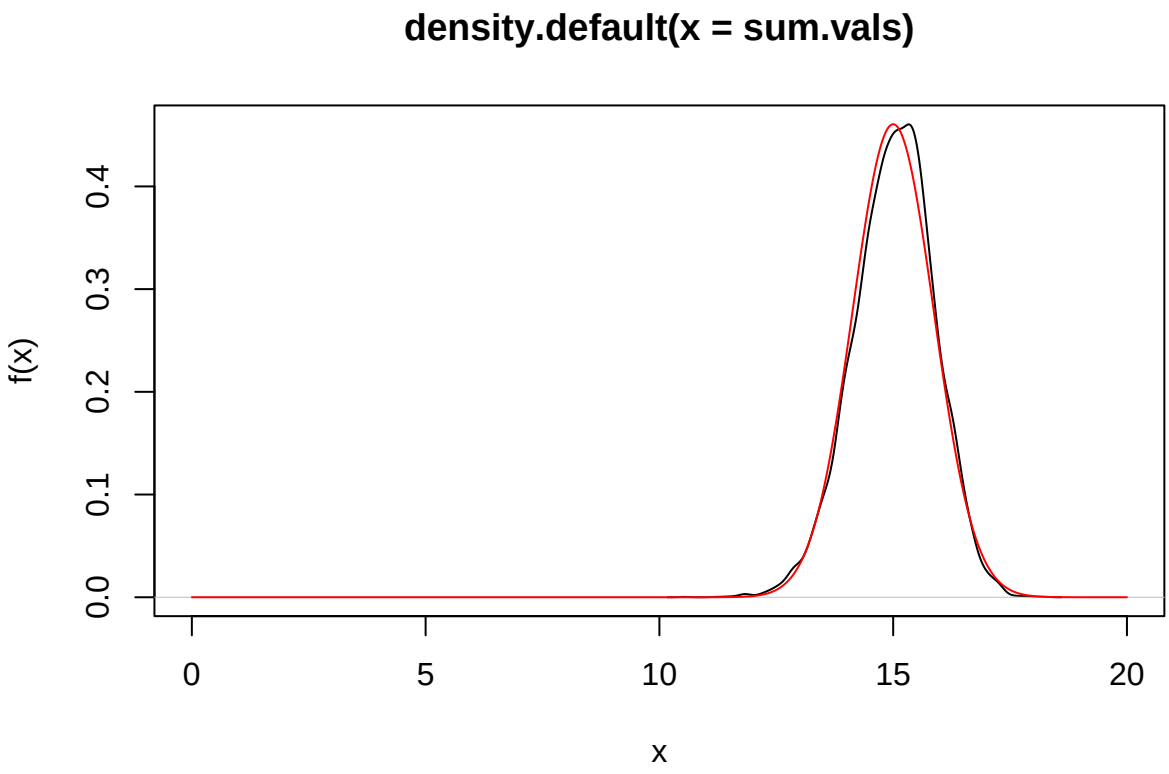
c) We are now trying to plot the simulated density and CLT-based approximation ($N(15, 0.1937^2 * 20)$) to validate the CLT approximation. To simulate that, we will use the inverse CDF method which is defined as:

$$\begin{aligned} F(x) &= \int_0^x f(u)\ du \\ &= \int_0^x 3u^2\ du \\ &= [u^3]_0^x \\ &= x^3 \end{aligned}$$

The inverse CDF is therefore $F^{-1}(x) = (p)^{1/3}$. To simulate the inverse CDF method, we will generate U(0,1) RVs and then plug into $F^{-1}(x)$:

```
n = 10000
sim.vals = matrix(runif(n*20)^(1/3), nrow=n)
sum.vals = apply(sim.vals, 1, sum)
plot(density(sum.vals), xlab="x", ylab="f(x)", xlim=c(0,20))
x.vals = seq(from=0,to=20, by=0.01)
points(x.vals, dnorm(x.vals, mean=15, sd=0.1937*sqrt(20)), type="line", col="red")
```

```
## Warning in plot.xy(xy.coords(x, y), type = type, ...): plot type 'line' will be
## truncated to first character
```



Question 8

a) my original solution was correct.
b) Answer:

```
(I_f = pnorm(1) - pnorm(0))
```

```
## [1] 0.3413447
```

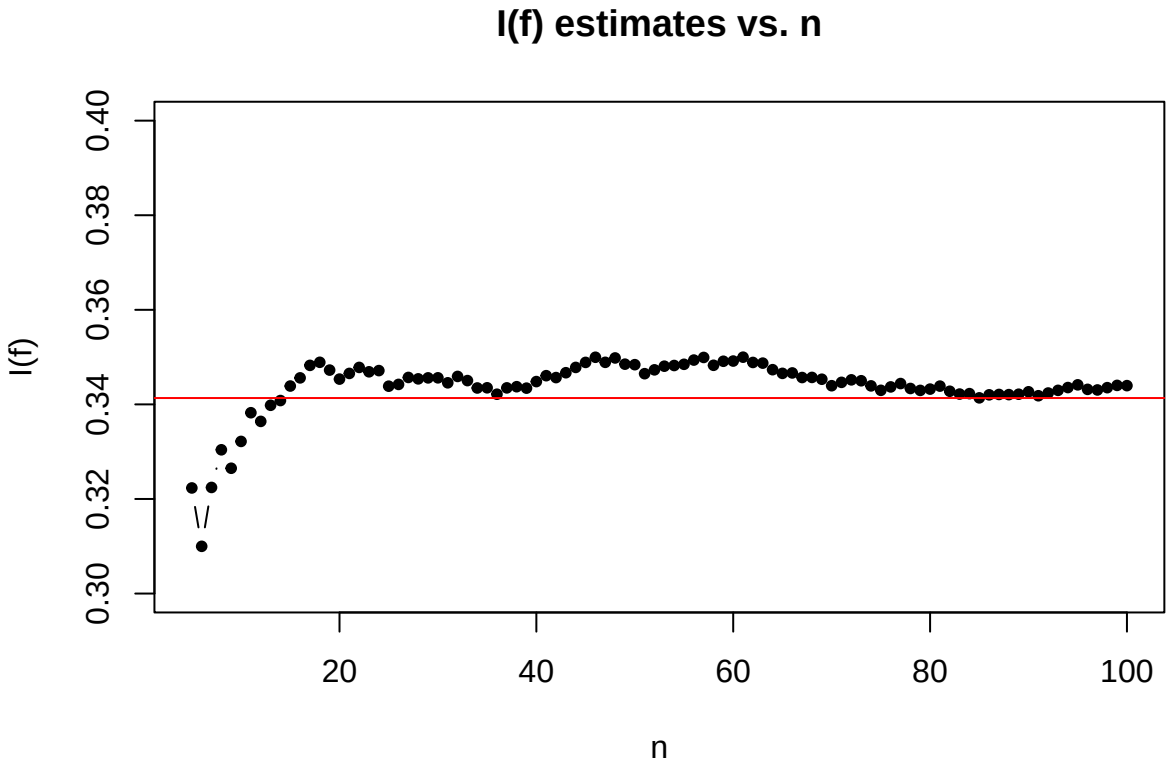
```
n.vals = 5:100
x.vals = runif(100)
g.vals = rep(1,100) # U(0,1) density is 1 in region
f.vals = dnorm(x.vals)
I.est = rep(0, length(n.vals))
for (i in 1:length(n.vals)) {
  n = n.vals[i]
  I.est[i] = mean(f.vals[1:n]/g.vals[1:n])
}

I.est[1:10]
```

```
##  [1] 0.3223274 0.3099832 0.3224342 0.3303972 0.3264976 0.3321690 0.3382382
##  [8] 0.3364066 0.3398092 0.3408127
```

plot estimates vs n with true $I(f)$ as a horizontal blue line:

```
plot(n.vals, I.est, main="I(f) estimates vs. n", xlab="n", ylab="I(f)", type="b", pch=20, ylim=c(0.3,0.4
abline(h=I_f, col="red")
```



As expected, the numerical integration values converge to the true integral (or the R approximation of the value) as n increases.

c) Optional