WMS 7th Ed, Chapter 3 Problem 120

Oliver de Pug

Capture, Recapture

If we have a population of size N of which k are tagged, then there are $N-k$ untagged animals. The recapture sample is of size 3 taken from the population of size N. The second sample can be obtained in $\binom{N}{3}$ ways. There are $\binom{N-k}{2}\binom{k}{1}$ ways of recapturing exactly one of the tagged animals. So, the probability of getting exactly one of the $k=4$ tagged animals is

$$\mathrm{P}(X=1) = \frac{\binom{N-4}{2}\binom{4}{1}}{\binom{N}{3}} \tag{1}$$

$$= \frac{12(N-4)(N-5)}{N(N-1)(N-2)} \tag{2}$$

To find the most probable value we can look at the probability for various N.

```
hyper.pmf <- function(N=4){
  (12*(N-4)*(N-5))/(N*(N-1)*(N-2))
}
N <- 4:20
(hyperprobs <- data.frame(N, p=hyper.pmf(N)))
```

```
##     N         p
## 1   4 0.0000000
## 2   5 0.0000000
## 3   6 0.2000000
## 4   7 0.3428571
## 5   8 0.4285714
## 6   9 0.4761905
## 7  10 0.5000000
## 8  11 0.5090909
## 9  12 0.5090909
## 10 13 0.5034965
## 11 14 0.4945055
## 12 15 0.4835165
## 13 16 0.4714286
## 14 17 0.4588235
## 15 18 0.4460784
## 16 19 0.4334365
## 17 20 0.4210526
```

Alternatively, we can use the internal **R** function, ***dhyper***, to determine the probabilities. The trick here is to be sure to interpret the parameters properly.

Looking at the help file for ***dhyper(x, m, n, k)*** we see that:

- ***x*** is the number of Type I items selected w/o replacement
- ***m*** is the number of Type I items in the population
- ***n*** is the number of Type II items in the population
- ***k*** is the number of items drawn from the population

In our problem we have that $N = m + n$. This means that for $m = 4$, $n = N - 4$. We also know that $k = 3$. Since the problem asks for $P(X = 1)$, we have $x = 1$. The trick is to look at various n to get $N = n + 4$.

```r
m <- 4
n <- N-4
k <- 3
x <- 1
hyperprobs$p.r <- dhyper(x, m, n, k)
hyperprobs
```

```
##     N         p       p.r
## 1   4 0.0000000 0.0000000
## 2   5 0.0000000 0.0000000
## 3   6 0.2000000 0.2000000
## 4   7 0.3428571 0.3428571
## 5   8 0.4285714 0.4285714
## 6   9 0.4761905 0.4761905
## 7  10 0.5000000 0.5000000
## 8  11 0.5090909 0.5090909
## 9  12 0.5090909 0.5090909
## 10 13 0.5034965 0.5034965
## 11 14 0.4945055 0.4945055
## 12 15 0.4835165 0.4835165
## 13 16 0.4714286 0.4714286
## 14 17 0.4588235 0.4588235
## 15 18 0.4460784 0.4460784
## 16 19 0.4334365 0.4334365
## 17 20 0.4210526 0.4210526
```

We can look at the ratio of successive terms to maximize the likelihood.

$$\frac{f(N+1)}{f(N)} = \frac{(N-3)(N-2)}{(N+1)(N-5)}$$

Setting this ratio equal to one we find:

$$(N-3)(N-2) = (N+1)(N-5) \tag{3}$$
$$N^2 - 5N + 6 = N^2 - 4N - 5 \tag{4}$$
$$N = 11 \tag{5}$$
$$N + 1 = 12 \tag{6}$$

A look at values of the ratio confirm that these values are maxima.

```r
N <- 6:20
data.frame(N, hyper.pmf(N+1)/hyper.pmf(N))
```

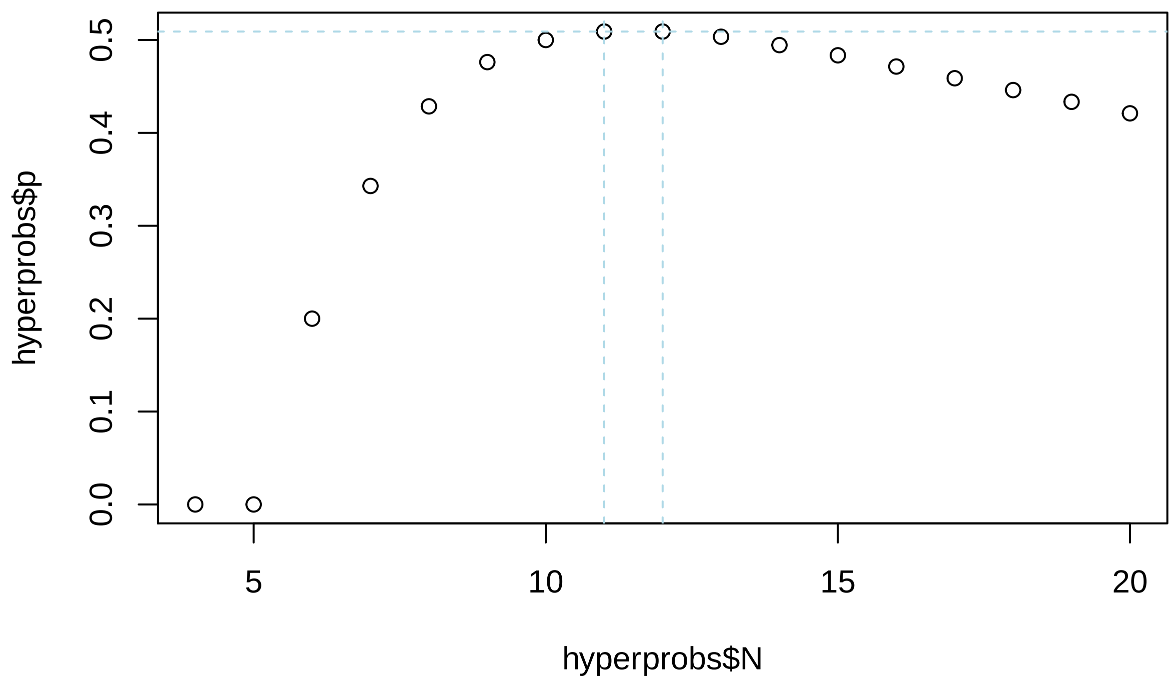
```
##     N hyper.pmf.N...1..hyper.pmf.N.
## 1   6                     1.7142857
## 2   7                     1.2500000
## 3   8                     1.1111111
## 4   9                     1.0500000
## 5  10                     1.0181818
## 6  11                     1.0000000
## 7  12                     0.9890110
## 8  13                     0.9821429
## 9  14                     0.9777778
## 10 15                     0.9750000
## 11 16                     0.9732620
```

```
## 12 17                         0.9722222
## 13 18                         0.9716599
## 14 19                         0.9714286
## 15 20                         0.9714286
```

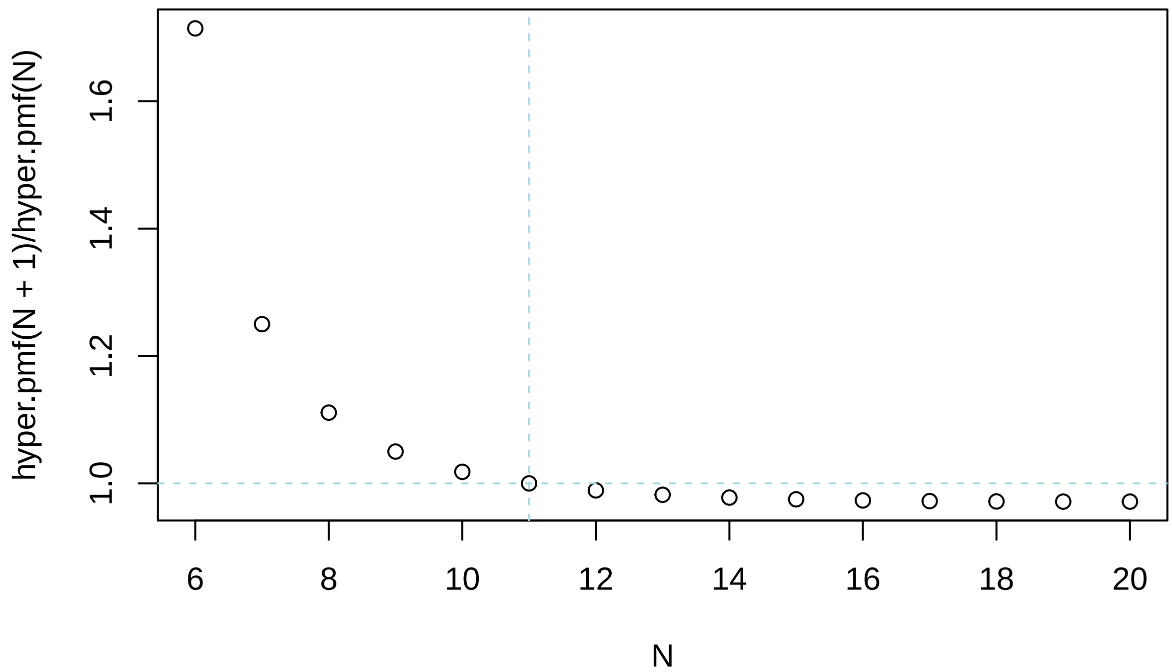
Noting that the ratio is greater than one for $N \leq 10$, we know that the pmf is increasing for these N. Similarly, since the ratio is less than one for $N \geq 12$, the pmf is decreasing for these values. The pmf is equal to one for $N = 11$, and we see that the pmf is maximized for $N = 11$ and $N + 1 = 12$.

A few plots may help.

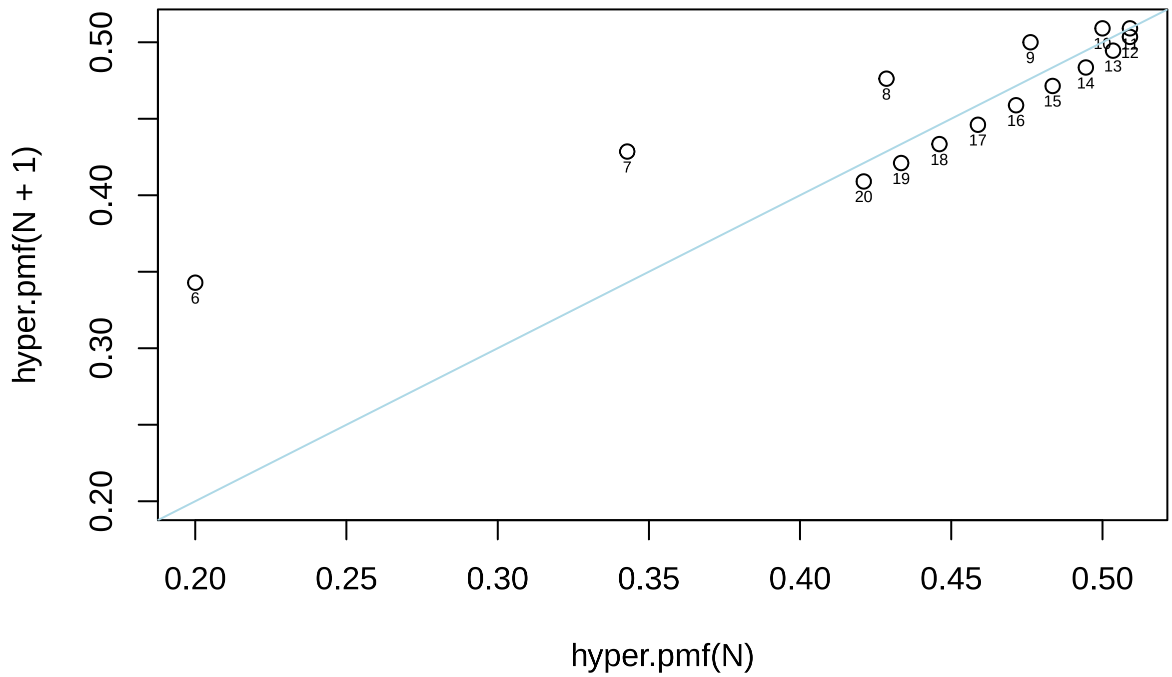
```
plot(hyperprobs$N, hyperprobs$p)
abline(h=max(hyperprobs$p), lty=2, col="lightblue")
abline(v=11:12, lty=2, col="lightblue")
```




```
plot(N, hyper.pmf(N+1)/hyper.pmf(N))
abline(h=1, lty=2, col="lightblue")
abline(v=11, lty=2, col="lightblue")
```




```
plot(hyper.pmf(N), hyper.pmf(N+1), ylim=range(hyper.pmf(N)))
text(hyper.pmf(N), hyper.pmf(N+1)-0.01, N, cex=0.5)
abline(a=0, b=1, lty=1, col="lightblue")
```


0.20
0.30
0.40
0.50
hyper.pmf(N + 1)
0.20
0.25
0.30
0.35
0.40
0.45
0.50
hyper.pmf(N)
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20