Sun, 30.3.25

Week 4 Lab

Question 1

N=100 N1=20 n=5

Suppose in a lot of 100 fuses there are 20 defective ones. A sample of 5 fuses are randomly selected from the lot without replacement. Let X be the number of defective fuses found in the sample.

(a)

It is easy to see that the distribution of X is $Hyper(N_1 = 20, N_2 = 80, n = 5)$.

Find the pmf of X, using the given values for N_1, N_2, and n.

Note that the pmf also requires a range — the x values for which the pmf is defined.

Either handwrite your answer, or type below.

(b)

dhyper(0,20,80,5)

Show that the probability $P(X = 0) = 19513/61110 = 0.3193$ using the `dhyper(...)` function.

Enter `?dhyper` in the Console for a reference on how to use this function.

(c)

The tabulated pmf of X can be computed using the command:

```
dhyper(0:5,20,80,5)
```

```
## [1] 0.3193094420 0.4201440026 0.2073437935 0.0478485677 0.0051482636 0.0002059305
```

Use this to fill out the probabilities in the following pmf table:

x	0	1	2	3	4	5
f(x)						

We can also verify that the above probabilities sum to 1 using the R function `sum(...)`. Confirm that this is true.

```
> dhyper(0:5,20,80,5)
[1] 0.3193094420 0.4201440026 0.2073437935 0.0478485677 0.0051482636 0.0002059305
> sum(dhyper(0:5,20,80,5))
[1] 1
```

(d)

Show that the cumulative probability $P(X \leq 3) = 0.9946458$.

Enter `?phyper` in the Console for help.

phyper(3,20,80,5)
[1] 0.9946458

(e)

Show that $E(X) = 1$, using the R functions `sum(...)` and `dhyper(...)`.

(f)

Find $E(X^2)$.

Reasoned about hypergeometric distribution syntax for a few seconds

In R, the hypergeometric distribution functions are available as:

- `dhyper(x, m, n, k)` for the probability mass function (PMF)
- `phyper(q, m, n, k)` for the cumulative distribution function (CDF)
- `qhyper(p, m, n, k)` for the quantile function
- `rhyper(nn, m, n, k)` for generating random deviates

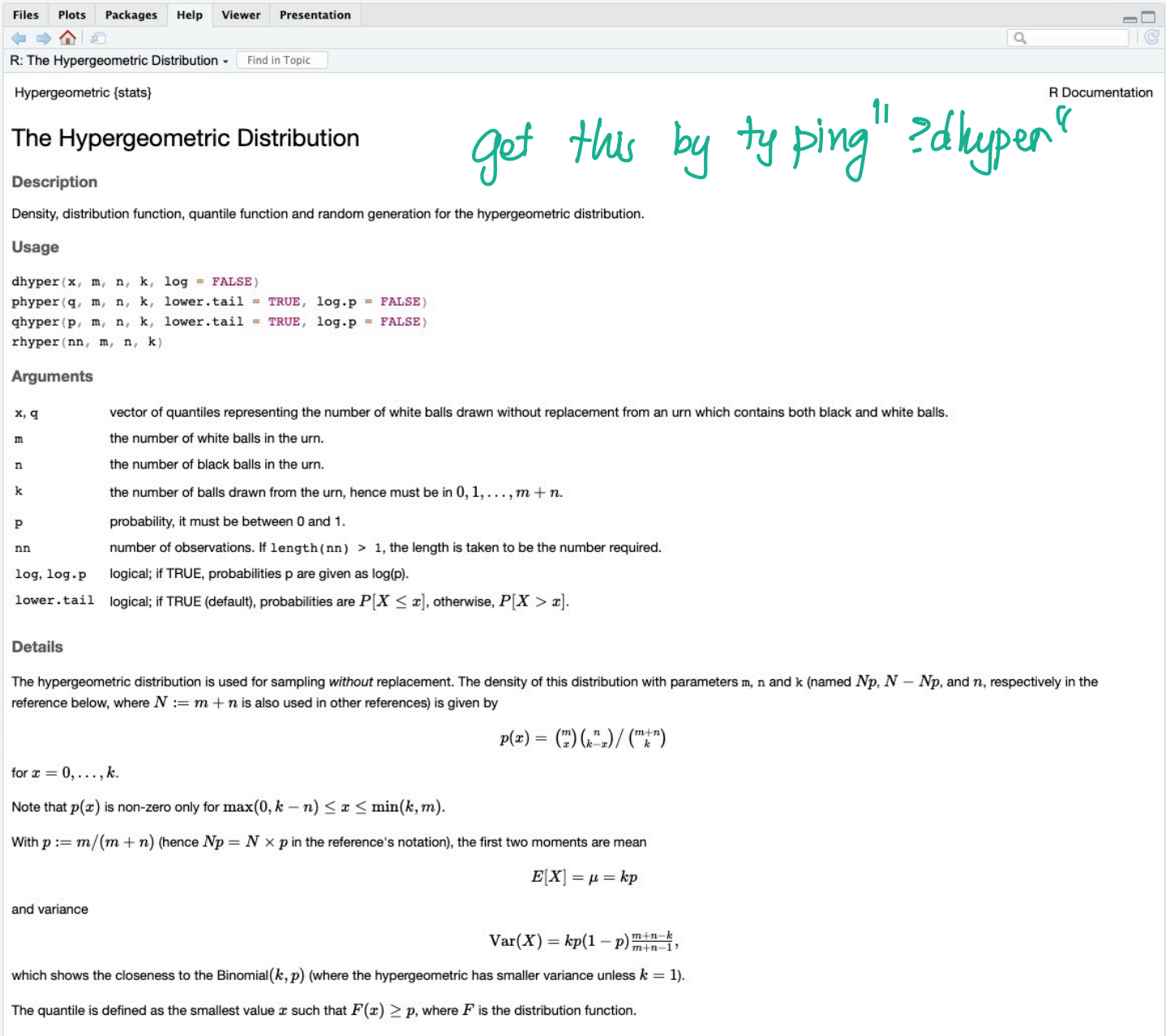
The correct syntax for the PMF is:

r Copy

```r
dhyper(x, m, n, k)
```

where:

- **x**: Number of white balls drawn (successes)
- **m**: Total number of white balls in the population
- Total number of black balls in the population



Files | Plots | Packages | Help | Viewer | Presentation

R: The Hypergeometric Distribution — Find in Topic

Hypergeometric {stats} R Documentation

get this by typing "?dhyper"

The Hypergeometric Distribution

Description

Density, distribution function, quantile function and random generation for the hypergeometric distribution.

Usage

```
dhyper(x, m, n, k, log = FALSE)
phyper(q, m, n, k, lower.tail = TRUE, log.p = FALSE)
qhyper(p, m, n, k, lower.tail = TRUE, log.p = FALSE)
rhyper(nn, m, n, k)
```

Arguments

x, q	vector of quantiles representing the number of white balls drawn without replacement from an urn which contains both black and white balls.
m	the number of white balls in the urn.
n	the number of black balls in the urn.
k	the number of balls drawn from the urn, hence must be in $0, 1, \ldots, m + n$.
p	probability, it must be between 0 and 1.
nn	number of observations. If `length(nn) > 1`, the length is taken to be the number required.
log, log.p	logical; if TRUE, probabilities p are given as log(p).
lower.tail	logical; if TRUE (default), probabilities are $P[X \le x]$, otherwise, $P[X > x]$.

Details

The hypergeometric distribution is used for sampling *without* replacement. The density of this distribution with parameters m, n and k (named Np, $N - Np$, and n, respectively in the reference below, where $N := m + n$ is also used in other references) is given by

$$p(x) = \binom{m}{x}\binom{n}{k-x} / \binom{m+n}{k}$$

for $x = 0, \ldots, k$.

Note that $p(x)$ is non-zero only for $\max(0, k - n) \le x \le \min(k, m)$.

With $p := m/(m + n)$ (hence $Np = N \times p$ in the reference's notation), the first two moments are mean

$$E[X] = \mu = kp$$

and variance

$$\mathrm{Var}(X) = kp(1 - p)\frac{m+n-k}{m+n-1},$$

which shows the closeness to the Binomial(k, p) (where the hypergeometric has smaller variance unless $k = 1$).

The quantile is defined as the smallest value x such that $F(x) \ge p$, where F is the distribution function.

1e

Reasoned about hypergeometric distribution usage for 12 seconds

From the problem setup, we have:

- $N = 100$ total fuses
- $n = 20$ defective fuses
- $N - n = 80$ good fuses
- $k = 5$ fuses drawn

Theoretical Formula for $E(X)$

For a hypergeometric random variable X (the number of "successes" in the sample), the expected value is:

$$E(X) = k \frac{n}{N}.$$

Plugging in the numbers:

$$E(X) = 5 \times \frac{20}{100} = 1.$$

Verifying in R

You can verify this via a direct summation in R:

r Copy

```r
# Define parameters
x_values <- 0:5
m <- 20    # number of defective (successes)
n <- 80    # number of good (failures)
k <- 5     # sample size

# Compute E(X) by summation: sum of x * P(X = x)
expected_value <- sum(x_values * dhyper(x_values, m, n, k))
expected_value
```

This should also return `1`. Hence, the solution to part (e) is:

$$\boxed{E(X) = 1.}$$

(g)

Find $Var(X)$.

(h)

Plot the pmf of X.

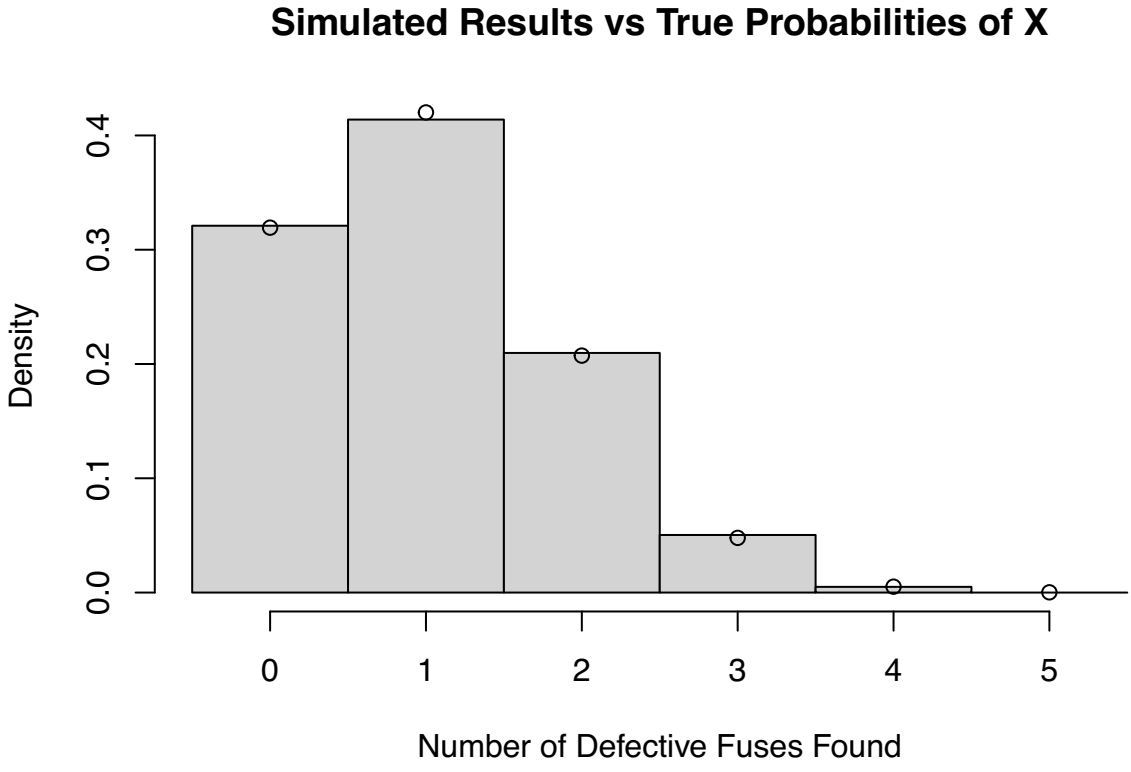
(i)

Much like in the Week 3 lab, we can compare the above pmf with simulated observations from this hypergeometric distribution. To generate 10000 independent observations from X, use the function `rhyper(...)`:

```
r <- rhyper(10000, 20, 80, 5)
```

We can compare the results of this simulation with the true probabilities using a histogram. The histogram, generated using the function `hist(...)` with argument `probability = TRUE`, shows the proportion of time we observe each possible outcome of X in our sample. We can then superimpose the true probabilities over this histogram:

```
breaks <- (0:6) - 0.5
hist(r, breaks, probability = TRUE,
     xlab = "Number of Defective Fuses Found",
     main = "Simulated Results vs True Probabilities of X")
points(0:5, dhyper(0:5,20,80,5))
```



Are the simulation's results approximately equal to the true probabilities? Why might they differ slightly?

My answer:

Question 2

A tetrahedron (four-sided die with outcomes 1,2,3,4) is rolled twice. Let X equal the sum of the two outcomes. The pmf of X can be derived and is given in the following table:

x	2	3	4	5	6	7	8
$P(X = x)$	1/16	2/16	3/16	4/16	3/16	2/16	1/16
rel.freq1 $m = 10000$							
rel.freq2 $m = 10000$							

(a)

Plot the pmf of X.

(b)

In rolling the tetrahedron twice as a random experiment, we use R to simulate 10000 trials of the experiment and then calculate the relative frequency table of the generated outcomes of X. This can be done using the following commands:

```
x1 <- sample(1:4, size=10000, replace=T)
x2 <- sample(1:4, size=10000, replace=T)
x.sum <- x1 + x2
table(x.sum)/10000
```

```
## x.sum
##      2      3      4      5      6      7      8
## 0.0610 0.1273 0.1909 0.2462 0.1859 0.1255 0.0632
```

Complete the 'rel.freq1' row in the table using the results returned from R.

(c)

The 10000 trials of the experiment in (b) can be done alternatively using only one instance of `sample(...)`. In the below code chunk, replace the `#?` with the appropriate vectors (and remove the argument `eval=FALSE` before running the code or knitting the file).

```
x <- #?
pmf <- #?
x.sum1 <- sample(x, size = 10000, replace = T, prob = pmf)
table(x.sum1)/10000
```

Complete the 'rel.freq2' row in the table using the results returned from R.

(d)

Find $E(X)$ and $Var(X)$ (using the true pmf of X).

(e)

Using `x.sum` and `x.sum1`, find the sample mean and sample variance of the generated numbers.

Compare the sample means and the sample variances with $E(X)$ and $Var(X)$ obtained in (d). Do their values differ? Explain why.

My answer: