Stat 110: R Section

Credit to Joe Blitzstein, Kenneth Baclawski, & Matt DosSantos DiSorbo

Justin Zhu

Good packages

```
#install.packages('Rlab')
library(Rlab)
```

```
## Rlab 2.15.1 attached.
```

```
##
## Attaching package: 'Rlab'
```

```
## The following objects are masked from 'package:stats':
##
##     dexp, dgamma, dweibull, pexp, pgamma, pweibull, qexp, qgamma,
##     qweibull, rexp, rgamma, rweibull
```

```
## The following object is masked from 'package:datasets':
##
##     precip
```

Bernoulli

```
# Parameters (n, p for binomial and geometric)
set.seed(110)
n = 10
p.bern = 5/10
Y = rbern(n, p.bern)
Y
```

```
##  [1] 1 0 1 1 1 0 1 1 1 1
```

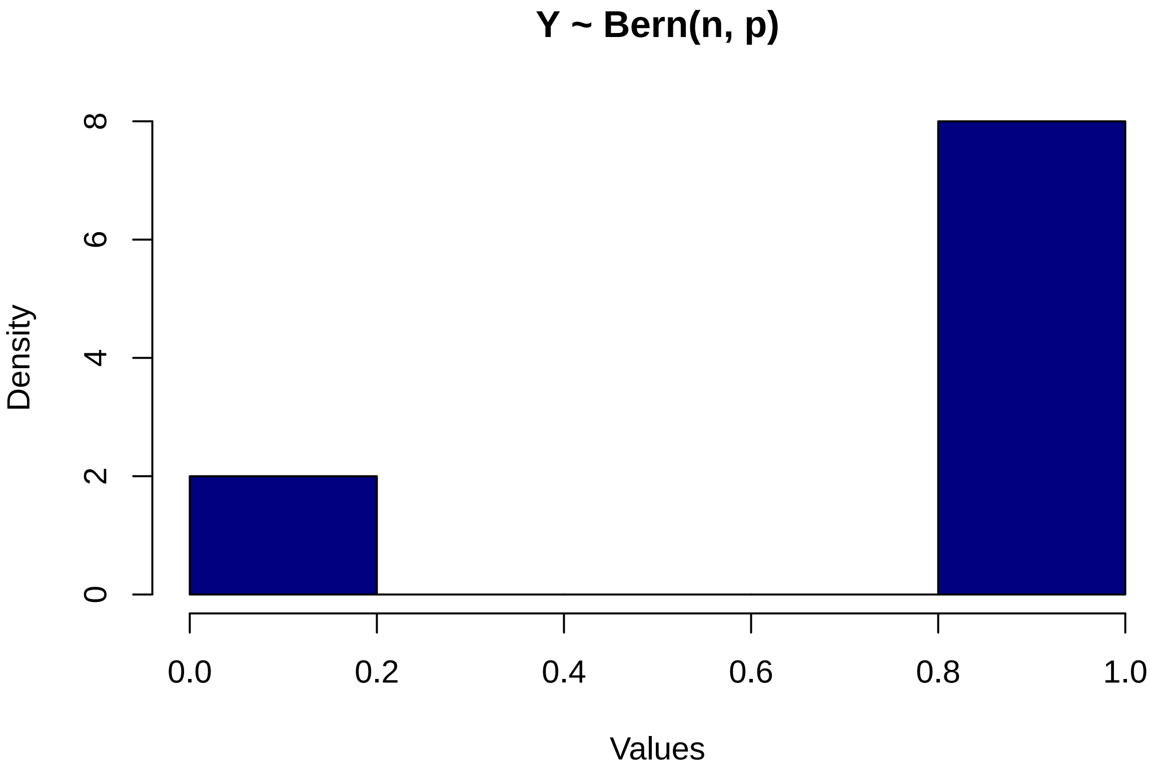
```
# Finding mean and variance
mean(Y)
```

```
## [1] 0.8
```

```
var(Y)
```

```
## [1] 0.1777778
```

```r
# Graphics
# Overall histogram
hist(Y, main = "Y ~ Bern(n, p)", ylab = "Density", xlab = "Values",
     col = rgb(0, 0, 0.5))
```



Binomial

```r
# Parameters (n, p for binomial and geometric)
set.seed(110)
sims = 1000
n = 10
p.binom = 5/10
Y_binom = rbinom(sims, n, p.binom)

# Finding mean and variance
mean(Y_binom)
```

```
## [1] 4.889
```

```r
var(Y_binom)
```

```
## [1] 2.403082
```

```r
# Graphics
# Overall histogram
#hist(Y_binom, main = "Y ~ Bern(n, p)", ylab = "Density", xlab = "Values",
     #col = rgb(0.5, 0, 1, 1))
```

What is the probability of getting at most 2?

```r
#Sum up the probability mass function!
dbinom(0, n, p.binom) + dbinom(1, n, p.binom) + dbinom(2, n, p.binom)
```

```
## [1] 0.0546875
```

```r
# Or alternatively, use the cumulative function:
pbinom(2, n, p.binom)
```

```
## [1] 0.0546875
```

#Bridging the distributions

```r
set.seed(110)
sims = 1000
n = 10
p.binom = 5/10
Y_sum = rep(0, sims)
for (i in 1:sims){
  Y = rbern(n, p.bern)
  Y_sum[i] = sum(Y)
}

#hist(Y_sum, main = "Y_sum ~ n * Bern(p)", ylab = "Density", xlab = "Values",
     #col = rgb(0.5, 0, 1, 1))

# We can compare the mean and variance between these two distributions
mean(Y_sum)
```

```
## [1] 5.105
```

```r
mean(Y_binom)
```

```
## [1] 4.889
```

```r
var(Y_sum)
```

```
## [1] 2.256231
```

```r
var(Y_binom)
```

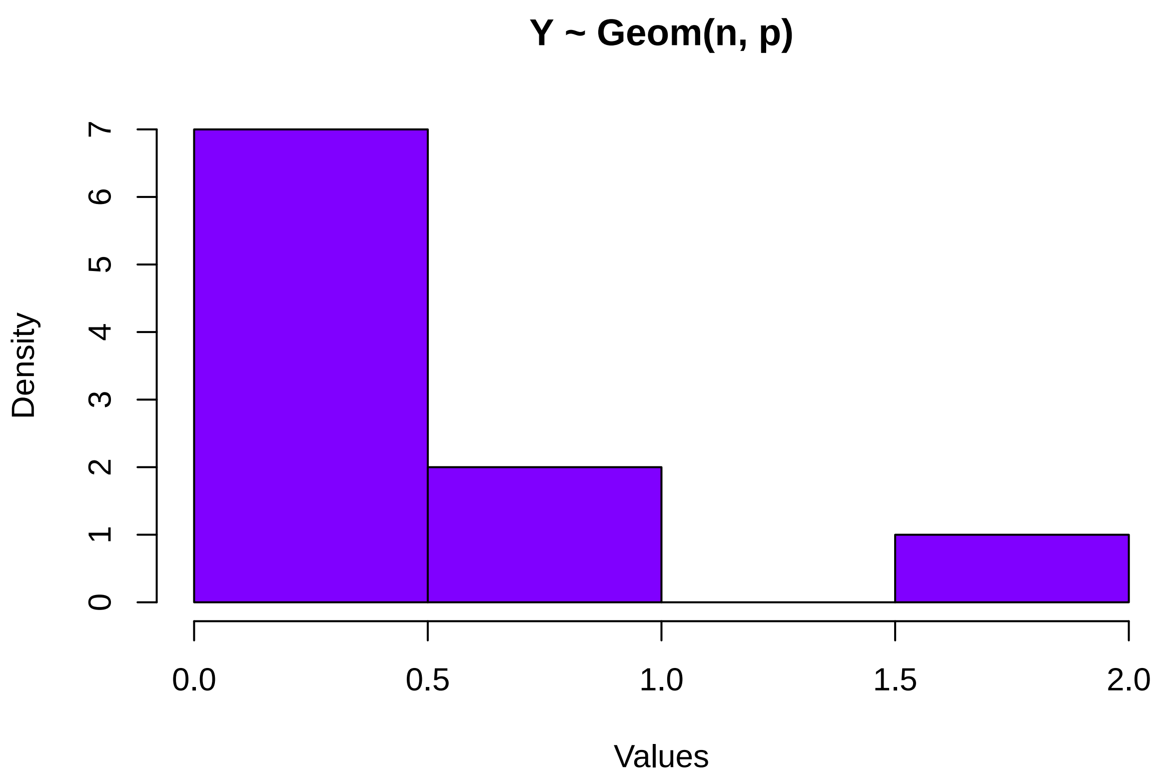
```
## [1] 2.403082
```

Geometric

```r
set.seed(110)
# Same old, Same old: Parameters first!
n = 10
p.geom = 5/10

# Find the distribution
Y_geom = rgeom(n, p.geom)

# We can now plot
hist(Y_geom, main = "Y ~ Geom(n, p)", ylab = "Density", xlab = "Values",
     col = rgb(0.5, 0, 1, 1))
```



Finding probabilities

```r
seven_heads = dgeom(7, p.geom)
seven_heads
```

```
## [1] 0.00390625
```

Hypergeometric

Think of a jar with b blue balls and w white balls. If you are drawing n balls total and are hoping to pick the white balls and not pick the blue balls and you then X be the number of white balls that you pick, then X has a Hypergeometric distribution. HGeom(w,b,n).

```r
# white balls
m = w = 8
n = b = 5
k = 3
x = 2

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

```
## [1] 0.4895105
```