POL211 TA Session

Gento Kato

November 29, 2018

```
setwd(dirname(rstudioapi::getActiveDocumentContext()$path))
rm(list = ls()) # Remove all objects from workspace.
```

Distribution Functions

Binomial Distribution

```
?rbinom
```

- size = number of trials (n in lecture)
- prob = probability of success (p in lecture)

```
# Random Draws from the distribution: n = sample size
rbinom(10, size = 1000, prob = c(0.9))
```

```
##  [1] 894 901 898 894 890 920 897 901 890 890
```

```
# Pr of specific values in the distribution i.e., f(x) = Pr(X=x)
dbinom(900, size = 1000, prob = c(0.9))
```

```
## [1] 0.04201679
```

```
# Cummulative Probability of values i.e., F(x) Pr(X<=q)
pbinom(900, size = 1000, prob = c(0.9))
```

```
## [1] 0.5154177
```

```
sum(dbinom(0:900, size = 1000, prob = c(0.9)))
```

```
## [1] 0.5154177
```

```
# The value that satisfies the specific cummulative probabilities
qbinom(0.5154177, size = 1000, prob = c(0.9))
```

```
## [1] 900
```

Poisson Distribution

```
?rpois
```

- lambda = mean

```
rpois(10, 414)
```

```
##  [1] 436 402 370 421 416 435 401 364 404 466
```

```
dpois(400, 414)
```

```
## [1] 0.01569664
```

```
ppois(400, 414)
```

```
## [1] 0.2549781
```

```
qpois(0.8, 414)
```

```
## [1] 431
```

Negative Binomial

```
?rnbinom
```

- size = number of successful trials (r in lecture)
- prob = probability of success in each trial (p in lecture)

```
rnbinom(10, size=1000, prob=0.8)
```

```
##  [1] 235 232 248 241 242 250 279 219 258 255
```

```
dnbinom(200, size=1000, prob=0.8)
```

```
## [1] 0.0003296854
```

```
pnbinom(250, size=1000, prob=0.8)
```

```
## [1] 0.5169171
```

```
qnbinom(0.8, size=1000, prob=0.8)
```

```
## [1] 265
```

Continuous Uniform

```
?runif
```

- min = lower limit of distribution (a in lecture)
- max = upper limit of distribution (b in lecture)

```
runif(10, min=10, max=20)
```

```
##  [1] 17.70673 18.01991 14.23575 11.75013 12.01273 17.30176 18.32876
##  [8] 11.73562 16.34723 19.30243
```

```
dunif(15, min=10, max=20)
```

```
## [1] 0.1
```

```
punif(17, min=10, max=20)
```

```
## [1] 0.7
```

```
qunif(0.8, min=10, max=20)
```

```
## [1] 18
```

Exponential

```
?rexp
```

- rate = lambda in lecture

```
rexp(10, rate=5)
```

```
##  [1] 0.11176158 0.05267572 0.19385470 0.06071571 0.32959488 0.08254175
##  [7] 0.11943636 0.16197520 0.13269487 0.07513755
```

```r
dexp(15, rate=5)
```

```
## [1] 1.339318e-32
```

```r
pexp(2, rate=2)
```

```
## [1] 0.9816844
```

```r
qexp(0.8, rate=5)
```

```
## [1] 0.3218876
```

Normal

```r
?rnorm
```

- mean = lower limit of distribution (a in lecture)
- sd = upper limit of distribution (b in lecture)

```r
rnorm(10, mean=5, sd=10)
```

```
##  [1]   7.935734   7.698089 -24.669090  -8.881785   2.965018  -2.746348
##  [7]  17.126606   7.779368  16.866239  23.863354
```

```r
dnorm(8, mean=5, sd=10)
```

```
## [1] 0.03813878
```

```r
pnorm(10, mean=5, sd=10)
```

```
## [1] 0.6914625
```

```r
qnorm(0.8, mean=5, sd=10)
```

```
## [1] 13.41621
```

Using Data from MASS Package (Just in case)

```r
# Install package (only once per PC)
```

```r
install.packages("MASS")
```

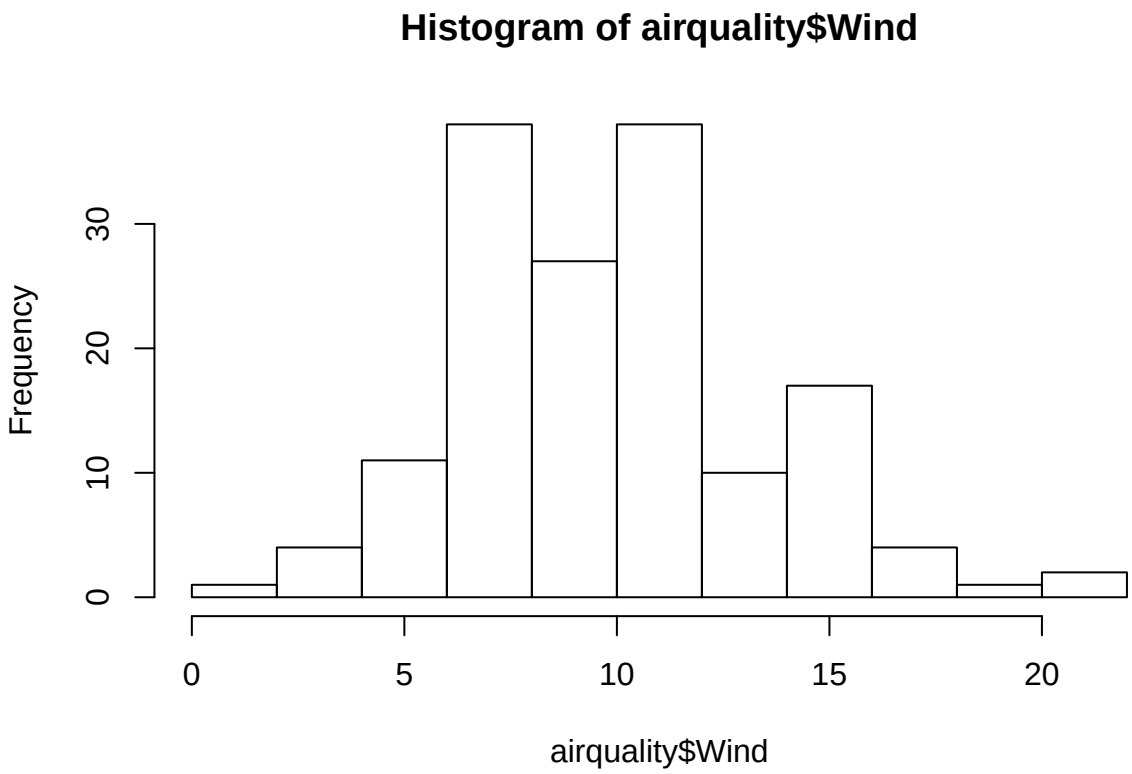
```r
library(MASS)

# Data list
data()

# Choose Data
data(airquality)

# See help file
?airquality

# Plot
hist(airquality$Wind)
```


Histogram of airquality$Wind
Frequency
0
10
20
30
0
5
10
15
20
airquality$Wind