

# Discrete Random Variables

Young W. Lim

2018-02-02 Fri

- 1 Discrete Random Variables
  - Based on
  - Discrete Random Variables

## "Probability with R: An Introduction with Computer Science Applications" Jane Horgan

I, the copyright holder of this work, hereby publish it under the following licenses: GNU head Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.2 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts. A copy of the license is included in the section entitled GNU Free Documentation License.

CC BY SA This file is licensed under the Creative Commons Attribution ShareAlike 3.0 Unported License. In short: you are free to share and make derivative works of the file under the conditions that you appropriately attribute it, and that you distribute it only under a license compatible with this one.

# Hardware Failure Example

```
fail <- c(9, 20, 39, 20, 11)
sum(fail)
prob <- fail/sum(fail)
prob
round(prob, 2)

num_fail <- 0:4
plot(num_fail, prob, xlab="Number of hardware failures in a week",
     ylab="Probability", type="h")
```

# Cumulative Distribution Function

```
cprob = c(0, 0, 0, 0, 0)
cprob = prob

cprob[1] = sum(prob[1])
cprob[2] = sum(prob[1:2])
cprob[3] = sum(prob[1:3])
cprob[4] = sum(prob[1:4])
cprob[5] = sum(prob[1:5])

for (i in 1:5) {
  cprob[i] = sum(prob[1:i])
}

cprob
plot(num_fail, cprob, xlab="failures",
     ylab="probability", type="S")
```

# Discrete Uniform Distribution

```
x <- c(1:6)
prob <- rep(1/6, 6)
plot(x, prob, type="h", xlab="rolling a die",
      ylab="prob")

plot(x, (1:6)/6, type="s", xlab="rolling a die")
```

# Benford's Law

```
x <- 1:9
prob <- c(.306, .185, .124, .094, .080, .064, .051, .049, .047)
plot(x, prob, type="h", xlab="first digit", ylab="probability")
```

```
x <- 1:9
prob <- log10(1+ 1/x)
round(prob, 3)
plot(x, prob, type="h", xlab="first digit", ylab="probability")
```

# Expected Values

```
x <- c(1, 2, 3, 4, 5, 6)
mean(x)
avg <- sum(x)/length(x)
```

```
x<-1:5
p<-c(1/16, 3/16, 7/16, 3/16, 2/16)
sum(x*p)
```



# Measuring the Spread

```
x<-1:5
p<-c(1/16, 3/16, 7/16, 3/16, 2/16)
avg<-sum(x*p)
var<-sum((x-avg)^2 * p)
var
```

```
roll = sample(c(1:6), 60, replace=T)
mean(roll)
sum((roll-mean(roll))^2)/60
```

```
roll = sample(c(1:6), 6000, replace=T)
table(roll)
roll
round(table(roll)/6000, 2)
plot(table(roll)/6000, xlab="die face",
      ylab="probability")
```

# Simulating Transformation

```
fail <- sample(c(0:6), 1000, replace=T,  
              prob=c(0.17, 0.27, 0.25, 0.17, 0.09, 0.05, 0.02))  
hist(fail, breaks=6, main="")  
mean(fail)  
var(fail)
```

```
fail10 <- fail*10  
mean(fail10)  
var(fail10)
```

```
fail10p200 <- 10*fail + 200  
mean(fail10p200)  
var(fail10p200)
```