Random Variables and the Binomial Distribution

(Highlights from Chapters 15 and 16)

Random Variables

- A **random variable** is a variable whose possible values are numerical outcomes of a random phenomenon. Use capital letters like X, Y, Z to denote random variables. Use lower case letters x, y, z to denote specific observed values.
- **Example:** X = number of times Paul the Octopus correctly predicts the winner of a World Cup Soccer/Football game in 8 attempts. He got $x = 8$ correct!

(image credit: Wolfgang Rattay/Reuters)

- **Example:** X = number of M&M's in a sample of size 100 that are blue. Maybe in a particular sample I observe $x = 22$.
- **Example:** X = number of infants in a sample of size 16 who choose a helpful toy. In our sample we observed $x = 14$.

Binomial Distribution

The **Binomial** distribution represents the sampling distribution of a count of the number of observational units in our sample with a certain characteristic.

- We're thinking about a categorical variable
  - Paul's prediction correct? Yes/No
  - M&M color: Brown, Red, Orange, Yellow, Green, Blue
- We are really interested in the proportion of the "population" that are in one of the categories (p)
  - What proportion of all predictions that Paul might ever make would be correct?
  - What proportion of all M&M's are blue?
- In a sample of size n, how many observational units are in that category? (x)
  - $n = 8$ predictions by Paul, $x = 8$ correct
  - $n = 100$ M&M's, $x = 22$ blue
- To use a binomial distribution, the results for different observational units in our sample must be **independent**:
  - Knowing the outcome for one observational unit in our sample doesn't change what we know about the outcome for other observational units in our sample (other than that they came from the same population with the same p).
  - Knowing one of Paul's predictions was correct doesn't change the probability that another will be correct
  - Knowing one M&M was blue doesn't change the probability that another will be blue

We use a short-hand notation to describe this situation:

- $X \sim \text{Binomial}(n, p)$
  - "X follows a Binomial distribution with sample size n and probability of success p"

There is a formula for calculating the probability that $X = x$, for each possible value of x from 0 to n:

$P(X = x) = \frac{n!}{x!(n-x)!} p^x (1-p)^{(n-x)}$

Let's have R do these calculations for us instead.

Example in Detail: Paul the Octopus

Define X = the number of successful predictions in 8 attempts.

Suppose $p = 0.8$ (Paul's predictions are pretty good!)

We could use the model $X \sim \text{Binomial}(8, 0.8)$

`dbinom`: Calculate the probability that X is exactly equal to some value

What's the probability that Paul gets 8 out of 8 predictions correct?

```
dbinom(x = 8, size = 8, prob = 0.8)
```

```
## [1] 0.1677722
```

What's the probability that Paul gets 7 out of 8 predictions correct?

```
dbinom(x = 7, size = 8, prob = 0.8)
```

```
## [1] 0.3355443
```

Displaying the full Binomial distribution

```
Paul_success_probabilities <- data.frame(
    num_successes = seq(from = 0, to = 8)
  )

Paul_success_probabilities
```

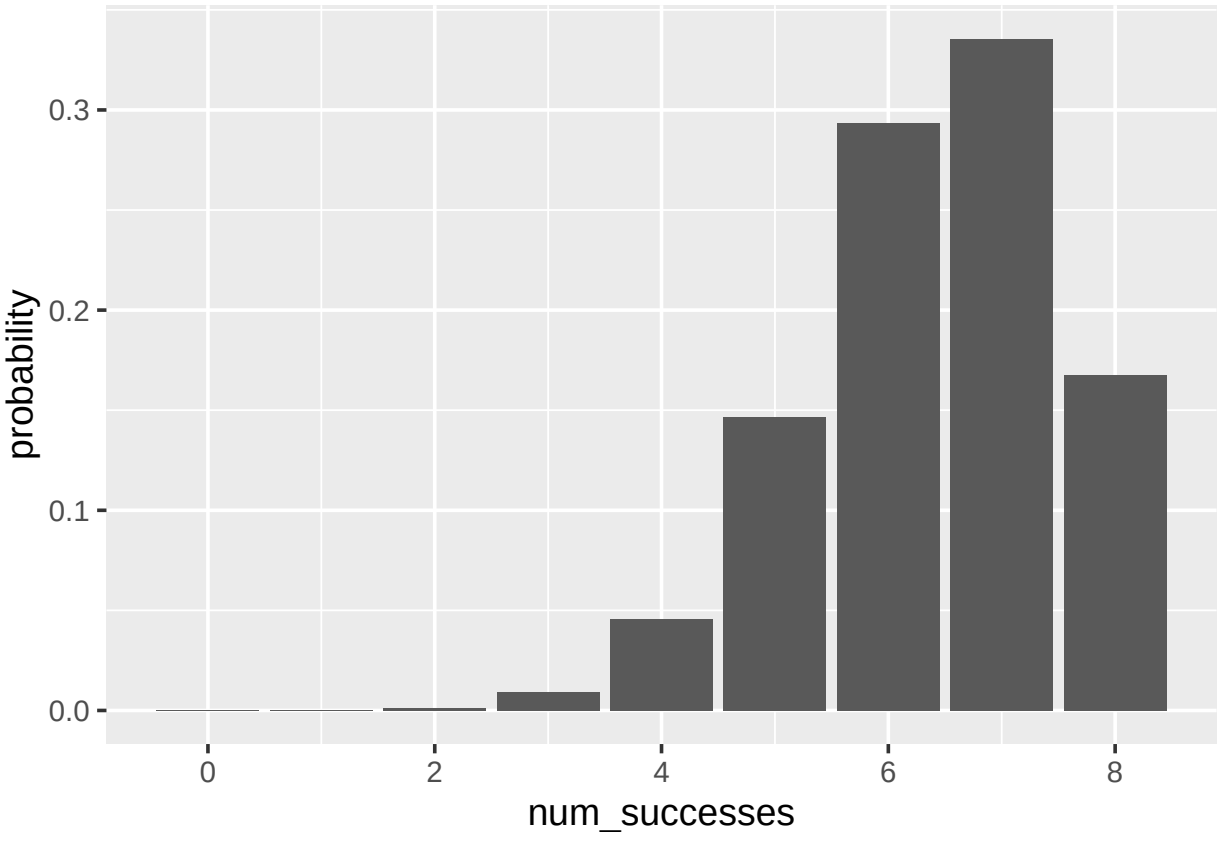
```
##   num_successes
## 1             0
## 2             1
## 3             2
## 4             3
## 5             4
## 6             5
## 7             6
## 8             7
## 9             8
```

```r
Paul_success_probabilities <- Paul_success_probabilities %>%
  mutate(
    probability = dbinom(x = num_successes, size = 8, prob = 0.8)
  )

Paul_success_probabilities
```

```
##   num_successes probability
## 1             0  0.00000256
## 2             1  0.00008192
## 3             2  0.00114688
## 4             3  0.00917504
## 5             4  0.04587520
## 6             5  0.14680064
## 7             6  0.29360128
## 8             7  0.33554432
## 9             8  0.16777216
```

```r
ggplot(data = Paul_success_probabilities,
    mapping = aes(x = num_successes, y = probability)) +
  geom_col() +
  theme_gray(base_size = 14)
```



`pbinom`: Calculate the probability that X is $\leq$ or $>$ some value

What's the probability that Paul gets 2 or fewer predictions correct?

By default, `pbinom` will calculate $P(X \leq q)$ for the specified value q.

```
pbinom(q = 2, size = 8, prob = 0.8)
```

```
## [1] 0.00123136
```

Compare to:

```
dbinom(x = 0, size = 8, prob = 0.8) +
  dbinom(x = 1, size = 8, prob = 0.8) +
  dbinom(x = 2, size = 8, prob = 0.8)
```

```
## [1] 0.00123136
```

d. What's the probability that Paul gets 6 or more predictions correct?

If we specify the argument `lower.tail = FALSE`, `pbinom` will calculate $P(X > q)$ for the specified value q.

```
pbinom(q = 5, size = 8, prob = 0.8, lower.tail = FALSE)
```

```
## [1] 0.7969178
```

Compare to:

```
dbinom(x = 6, size = 8, prob = 0.8) +
  dbinom(x = 7, size = 8, prob = 0.8) +
  dbinom(x = 8, size = 8, prob = 0.8)
```

```
## [1] 0.7969178
```

Summary of R Commands:

Calculate $P(X = 3)$	`dbinom(x = 3, size = 8, prob = 0.8)`
Calculate $P(X \leq 4)$	`pbinom(q = 4, size = 8, prob = 0.8)`
Calculate $P(X \geq 6)$	`pbinom(q = 6-1, size = 8, prob = 0.8, lower.tail = FALSE)`