R TUTORIAL, #9:
PROBABILITY DISTRIBUTIONS AND SAMPLING

The (>) symbol indicates something that you will type in.
A bullet (•) indicates what the R program should output (and other comments).

PROBABILITY DISTRIBUTION FOR ONE DIE

> Type: face = seq(1:6)
- In fact, face = 1:6 will also work.

> Type: face
- You will see the six die faces.

> Type: p = c(1, 1, 1, 1, 1, 1) / 6
- This assigns a uniform distribution to the die faces; each gets a probability of $\frac{1}{6}$.

> Type: data.frame(face, p)
- We see a tabular description of the distribution.

PROBABILITY SPIKE PLOT FOR ONE DIE

> Type: plot(face, p, type="h", xlim=c(0,6), ylim=c(0,1))
> Type: points(face, p, pch=16, cex=2)
- We see a probability spike plot of the distribution.
- This resembles a probability histogram with very thin bars.
- In the 'points' command, 'pch' may be thought of as "point character."
- 'pch=16' gives us filled-in circular disks at the points.
- If we do 'pch=15', for example, we get boxes at the points.
- 'cex' controls the size of the disks, boxes, etc.

> Type: points(face, p, pch=15, cex=3.5, col="red")

SAMPLING FROM A UNIFORM DISTRIBUTION

• Let's roll our die.
> Type: sample(face, size=1)
    • It is assumed that we have a uniform distribution. As we will see later, we can type 'sample(face, size=1, prob=p)', where p is our probability distribution, uniform or otherwise.

> (Repeat the above command a few times. You should get a variety of results.)

• Let's roll a pair of dice and get their sum.
> Type: sum(sample(face, size=2, replace=T))
    • We want to sample with replacement, because we want doubles to be possible.
    • The distribution of totals is not uniform!

> (Repeat the above command a few times. You should get a variety of results.)

> Type: sample(face, size=100, replace=T)
    • Our die is rolled 100 times.
    • How do the numbers of '1's, '2's, etc. compare?
    • 'replace=T' means that we sample with replacement. It is OK for us to get the same number more than once.

DESCRIBING SAMPLE RESULTS

• Let's store some sample results and do a frequency table, a relative frequency table, a histogram, and a pie chart.

> Type: results = sample(face, size=100, replace=T)
> Type: results

> Type: sort(results)

> Type: table(results)
    • We obtain a frequency table of our results.

> Type: table(results) / length(results)
    • We obtain a relative frequency table of our results.
    • Note: length(results) is 100 here.

> Type: hist(results, prob=T, breaks=seq(-0.5, 6.5, by=1), ylim=c(0,1))
- We see a relative frequency histogram of our results.
- For example, the interval from 3.5 to 4.5 corresponds to "4."
- This should resemble the (theoretical) probability distribution for one die.

> Type: lines(c(0.5,6.5), c(1/6,1/6), col="blue")
- This adds a blue line from the point $\left(0.5, \frac{1}{6}\right)$ to $\left(6.5, \frac{1}{6}\right)$.
- The line corresponds to the (theoretical) probability distribution.

> Type: pie(table(results), face)
- We see a pie chart.

A NON-UNIFORM DISTRIBUTION

> Type: x = c(2, 4, 8, 16)
> Type: p2 = c(1/8, 1/8, 1/4, 1/2)
- This assigns a non-uniform distribution to the *x* values.

> Type: data.frame(x, p2)
- We see a tabular description of the distribution.

> Type: plot(x, p2, type="h", xlim=c(0,20), ylim=c(0,1))
> Type: points(x, p2, pch=16, cex=2)
- We see a probability spike plot of the distribution.

> Type: results2 = sample(x, size=100, replace=T, prob=p2)
- 'prob=p2' indicates that p2 contains our probabilities for the *x* values.

> Type: results2
- What do you notice?

> Type: sort(results2)
- What do you notice?

> Type: table(results2)
- We obtain a frequency table of our results.
- What do you notice?

> Type: table(results2) / length(results2)
- We obtain a relative frequency table of our results.
- Note: length(results2) is 100 here.

> Type: hist(results2, prob=T, breaks=seq(-0.5, 16.5, by=1), ylim=c(0,1))
- We see a relative frequency histogram of our results.
- For example, the interval from 3.5 to 4.5 corresponds to "4."

MEAN (OR EXPECTED VALUE) OF A DISTRIBUTION

> Type: data.frame(x, p2)

- We see a tabular description of the distribution.

• Let's find the mean (or expected value) of this distribution. It's a "zigzag."

> Type: the.mean = sum(x*p2)

- The formula is: μ, or $E(X) = \sum x \cdot P(x)$.
- We use 'the.mean' to avoid conflict with the 'mean' command.

> Type: the.mean

VARIANCE AND STANDARD DEVIATION OF A DISTRIBUTION

• Let's find the variance of this distribution.

> Type: the.var = sum(x^2 * p2) - the.mean^2

- The formula is:

$$\sigma^2 = E\left(X^2\right) - \mu^2 = E\left(X^2\right) - \left[E(X)\right]^2 = \left[\sum x^2 \cdot P(x)\right] - \mu^2.$$

- 'p2' is the name of the probability vector; there is no squaring here!
- Also, $\sigma^2 = E\left[(X-\mu)^2\right] = \sum (x-\mu)^2 \cdot P(x)$, so we could do:

  the.var = sum(((x - the.mean)^2) * p2)

> Type: the.var

• Let's find the standard deviation of this distribution.

> Type: the.sd = sqrt(the.var)

> Type: the.sd

COMPARING SAMPLE RESULTS WITH THEORETICAL RESULTS

• How do our sample mean, variance, and standard deviation compare with the [theoretical] mean, variance, and standard deviation for the distribution?

> Type: results2

- This was our sample of size 100 from the distribution.

> Type: mean(results2)

- Compare this sample mean with the.mean, the theoretical mean of the distribution.

> Type: var(results2)

- We do want the sample variance here.
- Compare this sample variance with the.var, the theoretical variance of the distribution.

> Type: sd(results2)

- Compare this sample standard deviation with the.sd, the theoretical standard deviation of the distribution.