Topic 11a: Probability: Discrete Cases

The following table illustrates the case where we have discrete values, in this case languages, and a probability associated with each. Given the data that we have we must assume that there is no case where someone is studying two or more languages.

Language	Mandarin	Spanish	Xhosa	French	German	Japanese
Probability of studying	0.1150	0.3009	0.1681	0.2035	0.0619	0.1504

We could use this information to get:

P(French) = 0.2035

P(Spanish') = P(Spanish^c) =1 - P(Spanish) = 1 - 0.3009 = 0.6991

This is not as interesting as is the case where we have discrete values that are numeric (and hopefully interval or ratio measures). Here is such a case:

measure	14	17	18	22	23	25
Probability that measure	0.1769	0.2615	0.1462	0.1923	0.0923	0.1308
Probability of that measure as a fraction	23 / 130	34 / 130	19 / 130	25 / 130	12 / 130	17 / 130

P(X = 18) = P(18) =0.1462 = 19 / 130

P(X ≠17) = P(17^c) = 1 - P(17) = 1 - 0.2615 = 0.7385 = 1 - 34/130 = 130/130 - 34/130 = 96/130

We can also get

P(X ≤ 18) = P(14) + P(17) + P(18) = 23/130 + 34/130 + 16/130 = (23+34+19)/130 = 76/130

P(X > 18) = 1 - P(X≤18) = 1 - 76/130 = 130/130 - 76/130 = (130 - 76)/130 = 54/130

Even more than that, we can talk about the mean and the standard deviation of any population that conforms to the probabilities given in that table. For example, a population of 130 marbles, each with a number on it where we have 23 marbles with the number 14 on them, 34 marbles with 17 on them, 19 with 18 on them, 25 with 22, 12 with 23, and 17 with 25 would have the probabilities shown for randomly selecting a single marble. We can do this in R as:

```
2      # create our table
3      # first get the individual measures
4  vals  <-  c( 14, 17, 18, 22, 23, 25 )
5      # then get the number of times each measure
6      # needs to happen
7  marble <- c( 23, 34, 19, 25, 12, 17 )
8      # then make the population of marbles
9  all_marbles <- rep( vals, marble)
```

```
>      # create our table
>      # first get the individual measures
> vals  <-  c( 14, 17, 18, 22, 23, 25 )
>      # then get the number of times each measure
>      # needs to happen
> marble <- c( 23, 34, 19, 25, 12, 17 )
>      # then make the population of marbles
> all_marbles <- rep( vals, marble)
```

```
all_marbles <- rep( vals, marble)
    # and look at them
all_marbles
```

```
> all_marbles
  [1] 14 14 14 14 14 14 14 14 14 14 14 14 14 14 14 14 14 14 14 14 14 14
 [21] 14 14 14 17 17 17 17 17 17 17 17 17 17 17 17 17 17 17 17 17 17 17
 [41] 17 17 17 17 17 17 17 17 17 17 17 17 17 17 17 17 17 18 18 18 18 18
 [61] 18 18 18 18 18 18 18 18 18 18 18 18 18 18 18 18 18 22 22 22 22 22
 [81] 22 22 22 22 22 22 22 22 22 22 22 22 22 22 22 22 22 22 22 22 22 22
[101] 22 23 23 23 23 23 23 23 23 23 23 23 23 25 25 25 25 25 25 25 25 25
[121] 25 25 25 25 25 25 25 25 25 25
```

```
# just to be sure we conform, make a
# frequency table
source("../make_freq_table.R")
make_freq_table(all_marbles )
```

```
> # just to be sure we conform, make a
> # frequency table
> source("../make_freq_table.R")
> make_freq_table(all_marbles )
  Items Freq   rel_freq cumul_freq rel_cumul_freq  pie
1    14   23 0.17692308         23      0.1769231 63.7
2    17   34 0.26153846         57      0.4384615 94.2
3    18   19 0.14615385         76      0.5846154 52.6
4    22   25 0.19230769        101      0.7769231 69.2
5    23   12 0.09230769        113      0.8692308 33.2
6    25   17 0.13076923        130      1.0000000 47.1
```

```
    # then we can find the mean and standard
    # deviation of our population of marbles
mean( all_marbles )
    #  remember this is a population
source( "../pop_sd.R")
pop_sd( all_marbles )
```

```
>     # then we can find the mean and standard
>     # deviation of our population of marbles
> mean( all_marbles )
[1] 19.17692
>     #  remember this is a population
> source( "../pop_sd.R")
> pop_sd( all_marbles )
[1] 3.674162
```

```
    # But what if we had a different population
    # that still conformed to the probabilities
    # that we were given
big_pop <- rep( all_marbles, 7 )
    # just to be sure we conform, make a
    # frequency table
make_freq_table( big_pop )
     # find the mean and standard deviation of this
mean( big_pop )
pop_sd( big_pop )
```

```
>     # But what if we had a different population
>     # that still conformed to the probabilities
>     # that we were given
> big_pop <- rep( all_marbles, 7 )
>     # just to be sure we conform, make a
>     # frequency table
> make_freq_table( big_pop )
  Items Freq   rel_freq cumul_freq rel_cumul_freq  pie
1    14  161 0.17692308        161      0.1769231 63.7
2    17  238 0.26153846        399      0.4384615 94.2
3    18  133 0.14615385        532      0.5846154 52.6
4    22  175 0.19230769        707      0.7769231 69.2
5    23   84 0.09230769        791      0.8692308 33.2
6    25  119 0.13076923        910      1.0000000 47.1
>      # find the mean and standard deviation of this
> mean( big_pop )
[1] 19.17692
> pop_sd( big_pop )
[1] 3.674162
```

Any population that conforms to our probability table will produce those same mean and standard deviation results.

This example gives us an opportunity to introduce a new measurement, the **expected value**. The **expected value** of a discrete distribution is the **sum** of the **product** of the values with their associated probabilities. So, for the values $v_1, v_2, v_3, \ldots, v_m$ and the associated probabilities $P(v_1), P(v_2), P(v_3), \ldots, P(v_m)$, we have the formula

$$E(X) = \Sigma_{i=1}^{m} v_i \cdot P(v_i)$$

However, $P(v_i) = \frac{t_i}{n}$ where t_i is the number of times we have each value and n is the total number of items. But that means that the formula becomes

$$E(X) = \Sigma_{i=1}^{m} v_i \cdot \left(\frac{t_i}{n}\right)$$

In that formual the n is a constant so we can move it outside of the summation symbol to get

$$E(X) = \frac{1}{n} \Sigma_{i=1}^{m} v_i \cdot t_i$$

This is good, but $v_i \cdot t_i$ is just telling us to take t_i copies of v_i. That is the same as adding up all of the copies of all the values so this becomes

$$E(X) = \frac{1}{n} \Sigma_{i=1}^{n} x_i = \mu$$

which is our definition of the mean, μ.

How do we find the standard deviation of the discrete distribution? We just find the population standard deviation for any population that meets the values and probabilities of the distribution. There is, however, a formula that we could use. In fact that formula has two versions.

$$\sigma = \sqrt{\Sigma (v_i - \mu)^2 \cdot P(v_i)}$$

and

$$\sigma = \sqrt{\Sigma \left(v_i^2 \cdot P(v_i)\right) - \mu^2}$$

See the web page for an example of expected value in terms of "betting" and game theory.