Coding in advance of the April 1, 2019 class

For next time, I would like you to write codes to compute the Bayes factor between two models of the probabilities of different numbers that appear when a die is rolled. These models are the same as in the second case we treat in Lecture 9. In each case, please normalize your model so that the expected total number of rolls is the actual total number of rolls.

In Model 1 the probability is equal to 1/6 for every number. Given our assumed normalization, this model has no parameters.

In Model 2 the probability of a 1 is $1-p$, and the probability of a 2, 3, 4, 5, or 6 is $p/5$. Given our assumed normalization, this model has a single parameter, p, and our prior will be that p can with equal probability be anywhere from 0 to 1.

Your task is to write a code that will output the Bayes factor $\mathcal{B}_{12}$ for any set of rolls, and then to apply that to find the Bayes factors for the four data sets (which are the same as they were for coding task 6, but which are reproduced here for convenience). Note that although you could in principle expand the polynomial as we did in the notes, when there are many 1's in the rolls, the expansion of $(1-p)^n$ becomes unwieldy. Thus you will need to find a better way.

To help with that, let's take the fourth data set as an example:

```
1 104
2 165
3 180
4 196
5 173
6 182
```

We'll call this data set "data.dat". In C (my preferred language), we would read this in as follows (in C, loops start with 0 rather than 1):

```
data=fopen("data.dat","r");
Ntot=0; /* Total number of rolls */
for (i=0; i<Nspots; i++)
{
   fscanf(data,"%d %d",&j,&counts[i]);
   Ntot+=counts[i];
}
fclose(data);
```

You can figure out how to read in the file in your preferred language.

The Bayes factor between two models is the ratio of the evidences between those two models, i.e., (from Lecture 9):

$$\mathcal{B}_{12} = \frac{\int \mathcal{L}_1(a_1,\ a_2, \ldots,\ a_n) p_1(a_1,\ a_2, \ldots,\ a_n) da_1\, da_2 \ldots da_n}{\int \mathcal{L}_2(b_1,\ b_2, \ldots,\ b_m) p_2(b_1,\ b_2, \ldots,\ b_m) db_1\, db_2 \ldots db_m} \tag{1}$$

where the integration in each case is over the entire model parameter space.

For Model 1, we have no parameters. Thus it's a simple matter of calculating the Poisson likelihood of the data given the model that for Ntot rolls, there should be Ntot/6 1s, Ntot/6 2s, and so on. But for data with as many total rolls as we have here, the *likelihood* will be either huge or tiny. Thus we should compute the *natural log* of the likelihood, to get the log evidence.

For Model 2 we *do* have a parameter: p. The idea is that the probability of a 1 is $(1-p)$, whereas the probability of a 2, 3, 4, 5, or 6 is $p/5$. Thus *for a given* p the likelihood is

$$\mathcal{L} = \frac{(\mathrm{Ntot}(1-p))^{d_1}}{d_1!} \prod_{i=2}^{6} \left[\frac{(\mathrm{Ntot}(p/5))^{d_i}}{d_i!} \right] e^{-\mathrm{Ntot}} \ . \tag{2}$$

The product of the factorials in the denominator is in common between all models. Therefore, we can cancel them out; we don't need to include them because we are ultimately calculating a ratio. For *this particular problem*, we are normalizing so that the models assume the same number of total rolls as we have in the data. Thus Ntot is the same for all models, and therefore $e^{-\mathrm{Ntot}}$ can be cancelled out as well. As a result, the likelihood for Model 2, with a particular value of p, can be simplified to read

$$\mathcal{L} \propto (\mathrm{Ntot}(1-p))^{d_1} \prod_{i=2}^{6} \left[(\mathrm{Ntot}(p/5))^{d_i} \right] \ . \tag{3}$$

Remember to simplify the likelihood for Model 1 in the same way!

But as before, you need to work with the *log* of this quantity, because with large numbers of rolls you'll bust your computer otherwise. You are integrating $\mathcal{L}dp$, so the log of that is $\ln \mathcal{L} + \ln dp$. Let's define logterm $\equiv \ln \mathcal{L} + \ln dp$.

We are integrating the prior (which is equal to 1 here, so we don't worry about it) times the *likelihood*, not the log likelihood, so we need to use the trick I mentioned before, in which we calculate the integral of something, working only with logs. For example, in pseudocode plus C:

```
   logevidence2=-50.0;  /* Start small because we don't want to interfere
with the actual integral. */
   [Integrate over p]
      [For a given p, find logterm]
      if (logterm>logevidence2+50.0) /* logterm is huge */
        logevidence2=logterm;
      else if (logterm>logevidence2-50.0)  /* Comparable, so add this term */
        logevidence2+=log(1.0+exp(logterm-logevidence2));
      [Note that if logterm<logevidence2-50.0, we don't add it]
   [End integral over p]
```

Note that if $p = 0$ or $p = 1$, then the model probability for at least one of the spot numbers will be zero; 0 to any power is 0, and its log is -infinity, so that will bust your computer. Thus you can't have $p = 0$ or $p = 1$.

If your log evidence for Model 1 was logevidence1, it means that the log of the Bayes factor in favor of Model 1 compared with Model 2 was logevidence1-logevidence2, and the Bayes factors itself is

$$\mathcal{B}_{12} = e^{\rm logevidence1 - logevidence2}\ . \tag{4}$$