Ben Kahle
9/21/14
Bayesian

Progress Report 3

Exercises:

This week I worked on an exercise to determine the bias of a euro when the data is uncertain (there is a probability it is wrong). I started from the Euro class defined in Think Bayes which defined the updated likelihood of a hypothesis as the following:

```
x = hypo / 100.0
heads, tails = data
like = x**heads * (1-x)**tails
return like
```

I amended this function to add an probability that the data from the euro is incorrect, self.probWrong. This parameter is passed into the class during instantiation. After reading the heads and tails values I adjust them given the probability that the given data was incorrect.

```
notHeads = heads*self.probWrong
notTails = tails*self.probWrong
heads = heads - notHeads + notTails
tails = tails - notTails + notHeads
```

This change reduces the number of heads and tails by the probability that the coin was not actually a heads or tail, respectively. With a set of 6 euros having a 0, 0.2, 0.4, 0.6, 0.8, and 1 probability of being incorrect, I updated them all with 140 head claims and 100 tail claims. The perfect euro (0% incorrect) had a credible interval of 53-63 while the lying euro (100% incorrect) had a credible interval of 37-47; a complete reflection across 50%. The center Euros had a slightly wider credible region (11 percentage points) and had a slightly lower peak probability. As the probability of being incorrect reaches 50% the bias of the coin is reduced and as the probably goes beyond 50% the bias inverts completely. In retrospect, this result seems very reasonable as with a larger probability of being incorrect the result being biased will be reduced more dramatically than the result being biased against, equalizing the probabilities.

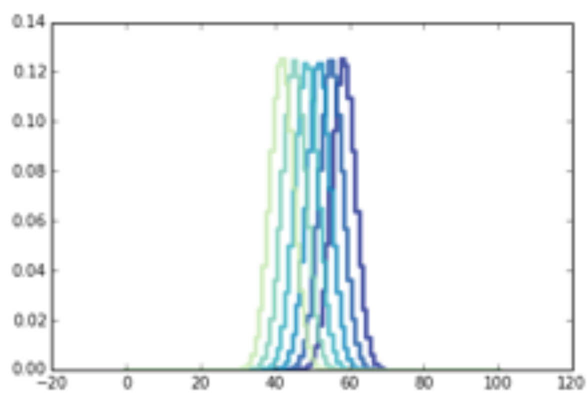

Figure 1: PMFs of the 6 Euros showing the equalization of the bias, and slight widening of the credible interval as the probability of being incorrect approaches 50%.

Reading + Reflection:

I enjoyed the in-depth analysis of the real world problems presented in chapters 6 and 7, although the addition of so many new concepts and their python implementation made the 6th chapter a much longer endeavor than previous chapters. While I appreciated seeing the pdf, kde, and cdf in application in the solving of the *Price is Right* problem, I think I need some additional examples of each concept on their own to get a better understanding of their definition and uses. Additionally, while I understand the steps in isolation to build the python implementation, putting all of the steps together in order to make the OptimalBid method still confuses me a bit. I think with a bit more in-depth inspection of each step I will see the whole picture more clearly. I found the discussion section at the end of the chapter to provide an interesting insight into a significant difference between Bayesian statistics and classical statistical analysis. During this course, I have found that seeing the posterior distribution provides more context and gives me significantly more information than a single point estimate of probability would. As a result, along with the logical progression, I have come to appreciate the Bayesian approach a good deal. I found the description of the Bruins pro*blem* in chapter 7 much more approachable than the *Price is Right* problem. Using the Poisson PMF to calculate the priors for the teams seems like a reasonable approximation and it was interesting to read about the optimization options in the discussion section. I found it particularly interesting to see a real world example where the lack of data contributes to a posterior heavily dependent on the prior. I read through the first half of the Cameron Davidson-Pilon chapter and found the contents to be a very quick and enjoyable introduction to the Bayesian approach. The examples and descriptions of simple Bayesian problems were framed in interesting ways and were easily understandable. I think the book follows closely to the Think Bayes (and the rest of the Think X series) style of more easily understandable language and more example based descriptions.

Case Study:

I have continued doing some research through r/statistics and other channels to try to find a problem and data that interests me specifically for a case study, but have not been inspired too much as of yet. I did however, skip to the end of the Davidson-Pilon chapter to his explanation of the PyMC library and looked through some of the examples. While I have not yet done too much comparison to the ThinkBayes2 library, I think working to convert some of the problems we have worked on to the PyMC library would be a very interesting problem and would also be very effective at helping my work towards one of my course goals of improving my Python knowledge and skills.