Austin Alexander
STA 601
Lab Assignment 1
January 19, 2013

Results:

For n=30:

Average Frequentist 95% Interval Length, Total In-Interval Count:
0.2780123, .9477

Average Uniform Prior 95% Interval Length, Total In-Interval Count:
0.2725699, .9621

Average Beta(8,2) Prior 95% Interval Length, Total In-Interval Count:
0.2384857, .9798

For n=5:

Average Frequentist 95% Interval Length, Total In-Interval Count:
0.5137233, .6685

Average Uniform Prior 95% Interval Length, Total In-Interval Count:
0.5671859, .9408

Average Beta(8,2) Prior 95% Interval Length, Total In-Interval Count:
0.3749377, .9995

Comments:

It is obvious from the above results that using a beta(8,2) prior provides both tighter bounds and greater accuracy than the other two methods. This is not surprising since it distributes most of the prior probability mass near the actual value of p. The poor showing of the frequentist approach relative to the uniform prior is a little more interesting. For n=30, they are nearly the same, but for n=5, the frequentist approach is not very good. This is due in part to my use of the normal approximation for the confidence intervals (as I understood to be the intended interpretation of the "frequentist method"; the exact binomial quantile function would give wider bounds at the lower n), and it is in part due to the complete reliance on the MLE instead of any prior assumptions. Using instead the uniform assumption results in a slightly wider interval length, but much higher accuracy rates as it helps pull low y values up towards the true value of p.

R Code (n=30):

```
#prior_uni = beta(1,1)
#prior_beta = beta(8,2)
#using the normal approximation of the binomial confidence interval

#initializing target variables
sim_num = 10000 #number of simulations
n = 30
p = 0.8
frequentist_count = 0
total_frequentist_length = 0
uni_count = 0
total_uni_length = 0
beta_count = 0
total_beta_length = 0

for (i in 1:sim_num){
  y = rbinom(1, n, p) #number of successes in n trials
  p_hat = y/n
  #using the normal approximation of the binomial confidence interval
  frequentist_lower = p_hat-1.96*(p_hat*(1-p_hat)/n)^(0.5)
  frequentist_upper = p_hat+1.96*(p_hat*(1-p_hat)/n)^(0.5)
  #increment the count if it is in the interval
  frequentist_count = frequentist_count + ((frequentist_lower<=p)&&(frequentist_upper>=p))
  frequentist_length = frequentist_upper - frequentist_lower
  total_frequentist_length = total_frequentist_length + frequentist_length

  posterior_uni_lower = qbeta(0.025, 1+y, 1+n-y)
  posterior_uni_upper = qbeta(0.975, 1+y, 1+n-y)
  uni_count = uni_count + ((posterior_uni_lower <= p)&&(posterior_uni_upper>=p))
  posterior_uni_length = posterior_uni_upper - posterior_uni_lower
  total_uni_length = total_uni_length + posterior_uni_length

  posterior_beta_lower = qbeta(0.025, 8+y, 2+n-y)
  posterior_beta_upper = qbeta(0.975, 8+y, 2+n-y)
  beta_count = beta_count + ((posterior_beta_lower<=p)&&(posterior_beta_upper>=p))
  posterior_beta_length = posterior_beta_upper - posterior_beta_lower
  total_beta_length = total_beta_length + posterior_beta_length

}

print('Average Frequentist 95% Interval Length, Coverage %:')
print(c(total_frequentist_length, frequentist_count)/sim_num)
print('Average Uniform Prior 95% Interval Length, Coverage %:')
print(c(total_uni_length, uni_count)/sim_num)
print('Average Beta(8,2) Prior 95% Interval Length, Coverage %:')
print(c(total_beta_length, beta_count)/sim_num)
```