Report

The dataset consists of 160 entries - each outcome representing a coin toss. x = 1 represents a head and x = 0, a tail. The dataset has been randomly generating using numpy libraries. The maximum likelihood estimator has been generated using numpy.random.rand() function and the corresponding dataset has been generated using numpy.random.choice() function. To ensure maximum likelihood estimator does not fall in the range [0.4, 0.6], uml is randomly generated while it falls in that range. A random dataset is generated every time the notebook is run.

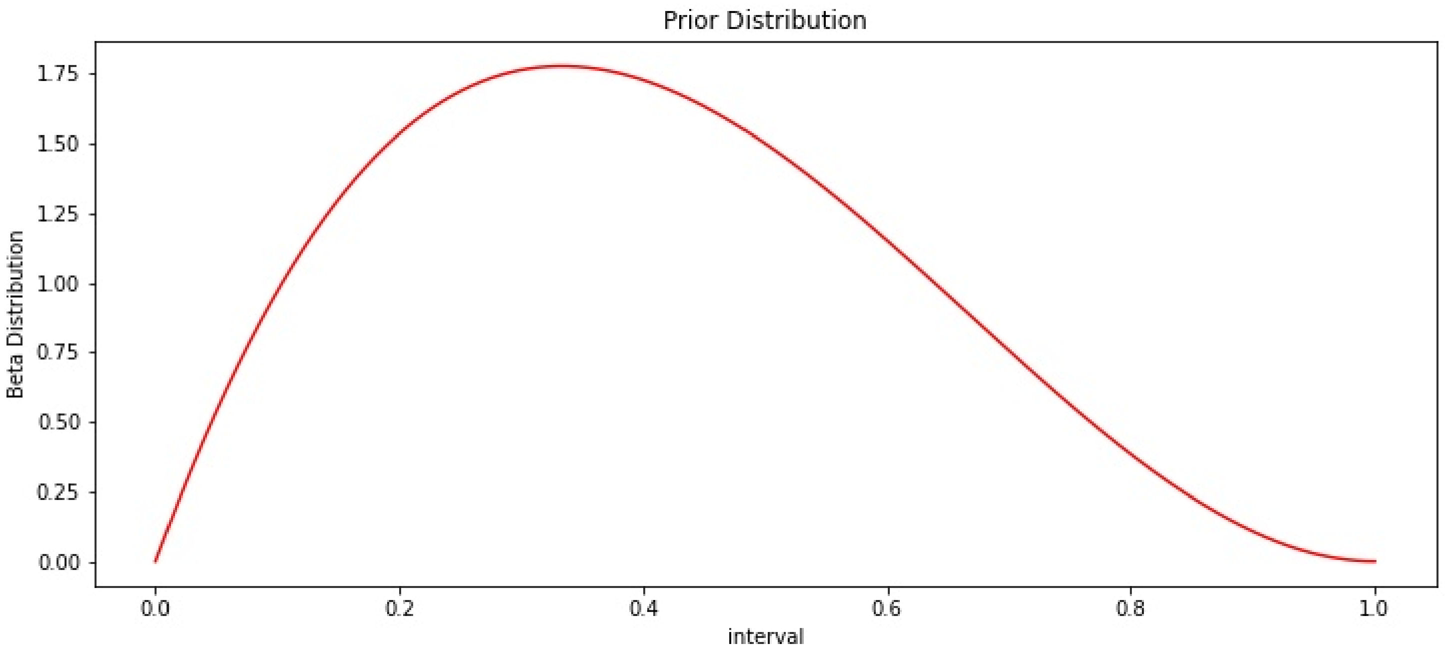
The fraction of data having x = 1, represented by Bernoulli distribution and binomial distribution give extremely over-fitted results for small datasets. In order to incorporate a Bayesian viewpoint, the domain experts knowledge of the inherent bias of the coin is introduced as the prior distribution over parameter μ. Beta distribution is chosen as the prior as it exhibits conjugacy property.

Prior Distribution

$$p(\mu|a,b) = Beta(\mu|a,b) = \frac{\mathit{\Gamma}(a+b)}{\mathit{\Gamma}(a)\mathit{\Gamma}(b)}\mu^{a-1}(1-\mu)^{b-1}$$

$$E[\mu] = \frac{a}{a+b}$$

Given, $E[\mu] = 0.4$. Therefore, $a = \frac{2}{3}b$. The values of a and b are taken as 2 and 3 respectively.



Posterior Distribution

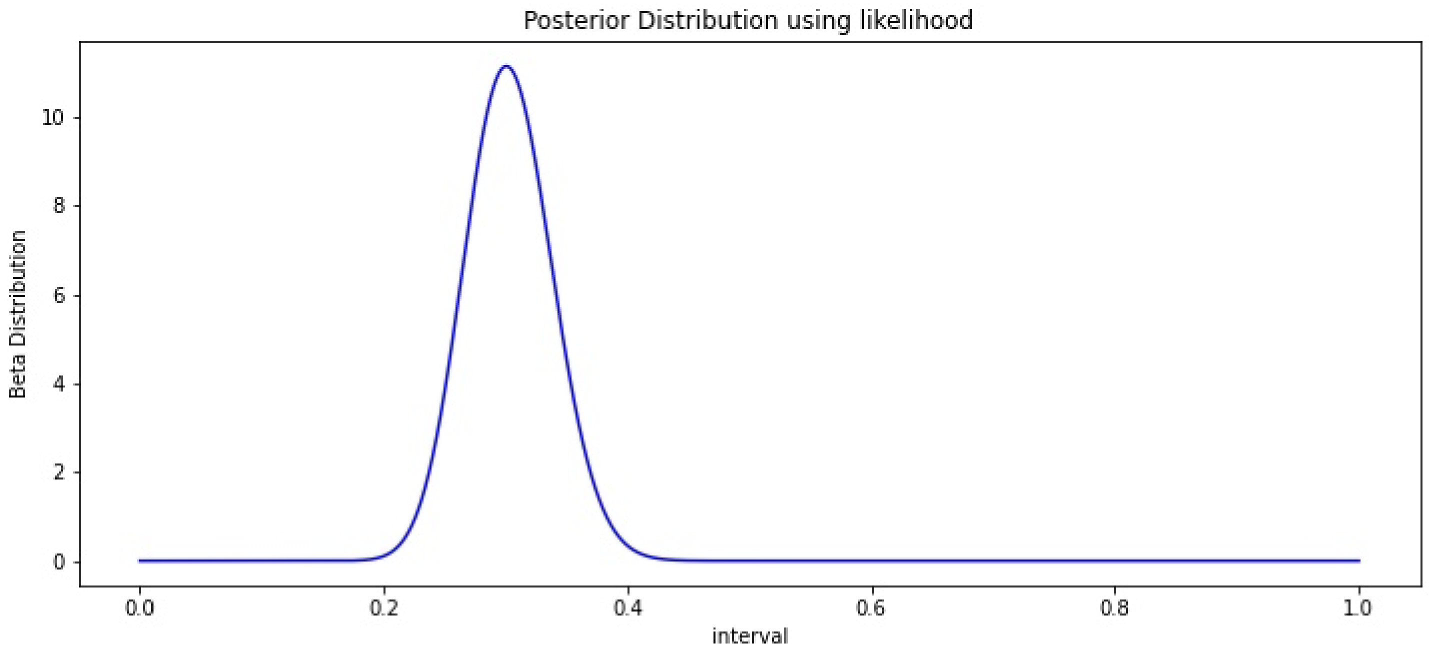
Posterior distribution of μ is obtained by multiplying the beta prior by the binomial likelihood function and normalizing.

$$p(\mu|m,l,a,b) \propto \mu^{m+a-1}(1-\mu)^{l+b-1}$$

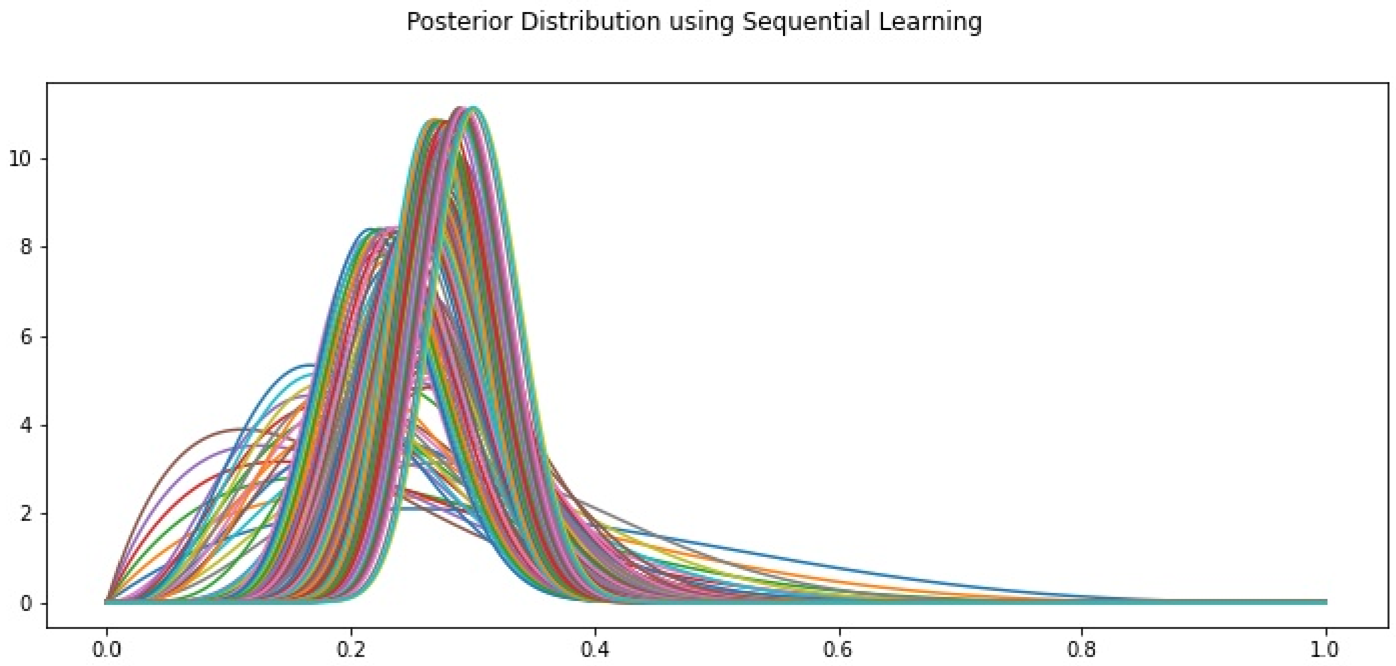
$$p(\mu|m,l,a,b) = \frac{\Gamma(m+a+l+b)}{\Gamma(m+a)\Gamma(l+b)}\mu^{m+a-1}(1-\mu)^{l+b-1}$$

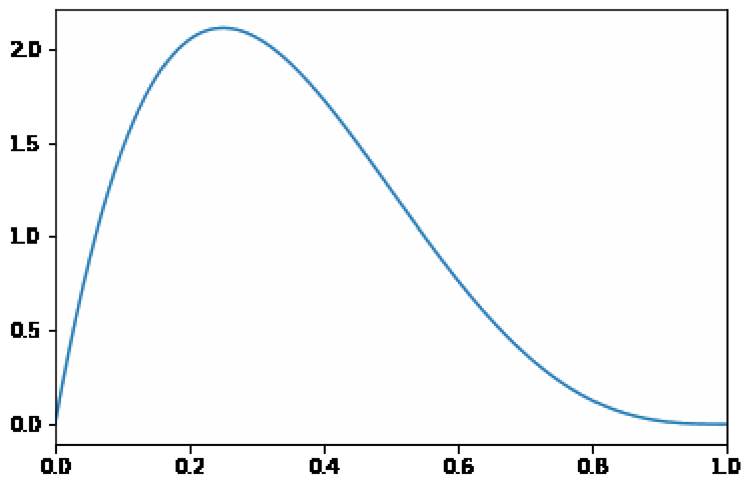
$$PosteriorMeanValue = E[\mu] = \frac{a+m}{a+m+b+l}$$

where m = number of heads in the dataset. l = number of tails in the dataset = size of dataset - m = 160 - m.



Evolution of Posterior in Sequential Learning:





Final hyperparameters of posterior distribution are a = 50, b = 115. Posterior Mean Value obtained is 0.30303030303030304.

Similarities and Differences between sequential approach and likelihood approach:

Differences

1. The sequential approach incorporates a Bayesian viewpoint. It is independent of the choice of the prior and the likelihood function. It records observations one at a time and discards them before the consideration of the next observation. At each stage, the derived posterior distribution acts as a prior distribution for the following data. Whereas when the whole dataset is available at once, the number of ones and zeros in the dataset respectively update the parameter values at once.
2. Only Sequential approach works in the case of real-time learning scenarios where a steady stream of data is arriving, and predictions must be made before all of the data is seen.

Similarities

1. Both the approaches yield the same final posterior distribution. The posterior mean for μ always lies between the prior mean and the maximum likelihood estimate for μ for a finite dataset.

As the number of observations increases, the posterior distribution becomes more sharply peaked and the variance and uncertainty represented by the posterior distribution will steadily decrease. In the limit of an infinitely large data set m, l → ∞ the result reduces to the maximum likelihood result.

If $\mu_{ML} = 0.5$, the prior distribution assumes a non-biased coin. Since the dataset is randomly generated, the posterior distribution peaks at approximately $\mu = 0.5$, i.e. the final distribution approximately becomes a Guassian with mean 0.5.

Bob's model would be more helpful and easier while working with large real time data as it makes use of observations one at a time, or in small batches, and then discard them before the next observations, therefore it does not require the whole data set to be stored or loaded into memory. It can also be employed in real-time learning scenarios where a steady stream of data is arriving, and predictions must be made before all of the data is seen.

Likelihood function takes the form of $\mu^x(1-\mu)^{1-x}$, therefore, if a distribution that is proportional to the powers of μ and $(1-\mu)$ is taken as the prior, then the posterior distribution would have the same functional dependance as the prior. If a Beta distribution is taken as prior, the posterior would indeed be another Beta function and its normalization coefficient can be obtained easily. However, since Gamma, Guassian or Pareto lack this property, known as conjugacy, the posteriors of obtained would differ functionally from their priors which will complicate computation.