CS F320 - Foundations of Data Science

Assignment 3

BY

Name of the Student	ID Number
Rikil Gajarla	***2017A7PS0202H***
Kasuba Badri Vishal	***2017A7PS0270H***

Foundations of Data Science Report

BIRLA INSTITUTE OF TECHNOLOGY AND SCIENCE, PILANI

I Semester, 2019-20

Abstract

This focus of this assignment is on getting the final distribution of probability of an event happening by applying the observed events to the initial assumed distribution. This approach adjust the prior (predicted from domain expertise) distribution based on data seen in real observations to get a final distribution.

Theory

Let the probability of getting head on a biased coin is μ. Let us generate a dataset D by randomly tossing the coin. Let 1 denote occurrence of head and 0 denote occurrence of tail. Let the total number of observations be N. Then, μ_{ML} , the maximum likelihood estimator is equal to (m/N) where m is the number of heads observed (no of 1's in dataset) .

We Know that Posterior distribution ∝ Likelihood distribution x Prior distribution
Let the prior distribution be a Beta distribution with parameters a and b. Through Bayes theorem,

$$P(\mu|D, a, b) \propto P(D|\mu) \times P(\mu|a, b)$$

As we know, coin tossing follows a Bernoulli distribution. It's probability density function is given by:

$$p(D|\mu) = \prod_{n=1}^{N} p(x_n|\mu) = \prod_{n=1}^{N} \mu^{x_n}(1-\mu)^{1-x_n}$$

[Likelihood Function]

x_n is 1 if head and 0 if tail

Let PDF of Beta distribution be (prior distribution):

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

[Beta Distribution]

Let us assume $E[\mu]$ for Beta distribution be = 0.4 (this will be changed later based on observations). For Beta Distribution,

$$E[\mu] = \frac{a}{a+b} \text{ so for } E[\mu] = 0.4 \Rightarrow \frac{a}{b} = \frac{4}{6}$$

So, we will start with parameters (a, b) = (4, 6) to get $E[\mu] = 0.4$

Then, the Posterior distribution is:

$$P(\mu \mid D, a, b) \propto \mu^{x_n+a-1}(1-\mu)^{b-x_n}$$

[Posterior Distribution]

So, we see that a increases by no of heads observed and b increases by no of tails observed. The new μ_{ML} in posterior now changes according to actual observations. For sufficiently large amount of data, it reaches the μ_{ML} of the dataset.

Implementation

Size of dataset taken is 160, which are random values of {0,1} such that $\mu_{ML} \notin (0.4,0.6)$. μ_{ML} is taken to be approximately 0.6625

Prior and Posterior distributions are implemented as Beta distributions and values of a,b are taken as 4,6 respectively so as to make mean of prior to be 0.4.

Model-1

We have taken individual prior distribution resulting datapoint and evaluated the posterior for that point and updated the prior distribution as equal to resultant posterior and did the same until all data points are evaluated

Model-2

By The Known Dataset , The posterior distribution is evaluated considering entire dataset in which prior distribution and likelihood function are evaluated for entire dataset.

Comparison

As the dataset is unchanging for the entire work flow , both the models result into the same result. I.e probability of getting head. I

Remarks :

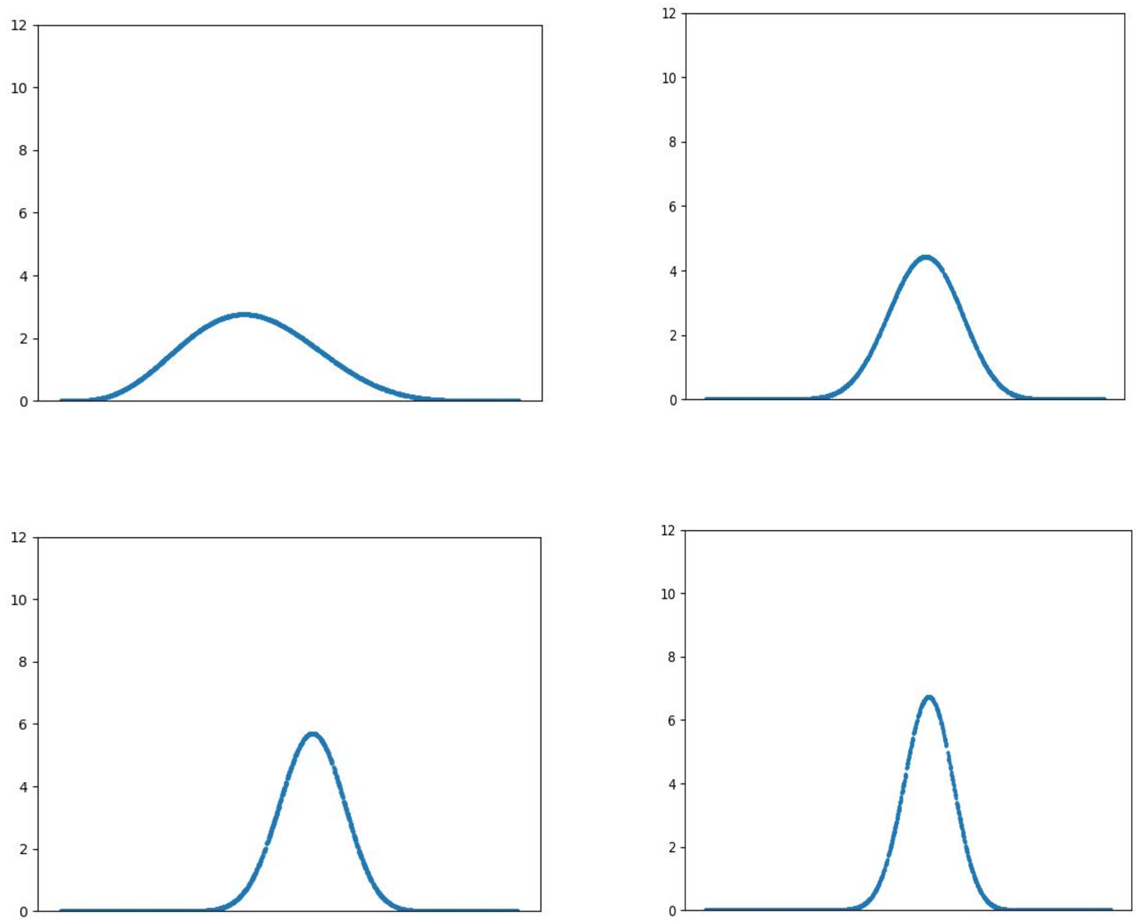
If more points are added to the dataset, then more better probability of getting head can be known precisely as more points result in more critical analysis and behaviour of biased event could be understood better.

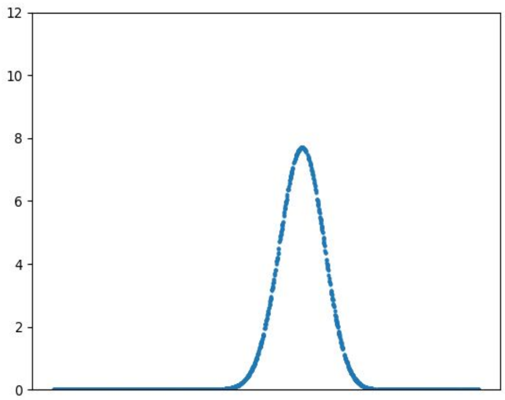
If μ_{ML} =0.5 then, posterior distribution would look like normal distribution as the mean approaches 0.5 , the coin behaves more as unbiased coin and distribution approaches normal distribution

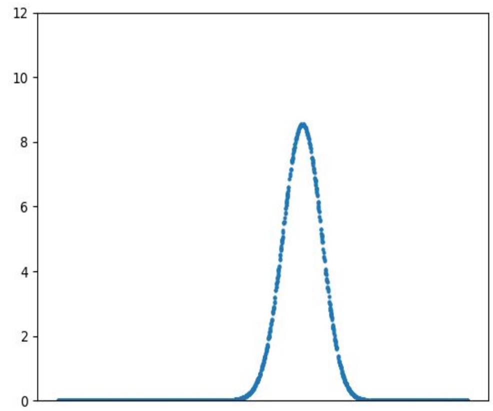
Results

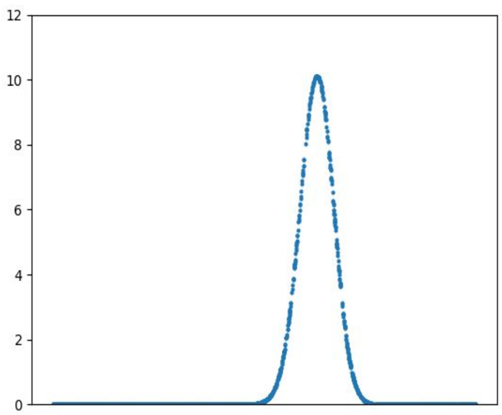
Part A

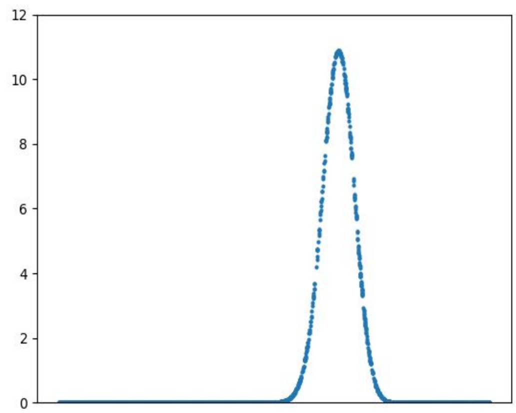
Sequence of images




12
10
8
6
4
2
0

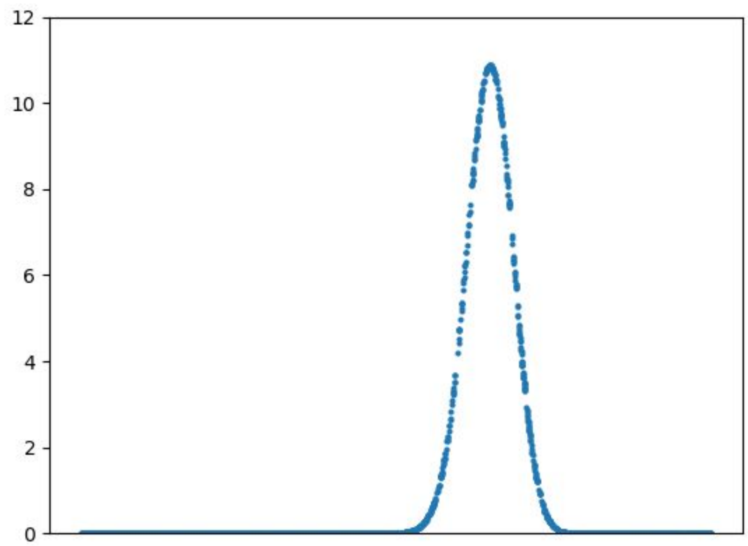

12
10
8
6
4
2
0


12
10
8
6
4
2
0


12
10
8
6
4
2
0

Part B

Final result:



Conclusion

Posterior Distribution helps in effective summarizing the values that might be obtained by uncertain values like tossing a coin, flipping a dice etc. Prior Distribution predicts outcome before the event occurs. With both methods , we can infer that both evaluates to same result when dataset is same. Generally sequential Learning is prefered when events occur and change with period of time whereas other method of finding posterior is used when we have all dataset which persist with us already.