CS215 Assignment 3 Report

Arpon Basu
Shashwat Garg

Autumn 2021

Contents

1 Introduction

Welcome to our report on CS215 Assignment 3. We have tried to make this report comprehensive and self-contained. We hope reading this would give you a proper flowing description of our work, methods used and the results obtained. Feel free to keep our code scripts alongside to know the exact implementation of our tasks. The pictures included in the graphs folder are a part of this report as well.

We have referred to some sites on the web for finding the MATLAB implementations (generic documentation pages) and general statistical knowledge needed for various parts of the assignment.

In many places, to better give context to the place from which the questions could have arisen, some theoretical discussions have been engaged in.

Hope you enjoy reading the report. Here we go!

2 Some Logistic Issues

Kindly install the `Statistics and Machine Learning Toolbox (v12.2 for us)` in Matlab. The `boxplot` utility is available only in the `Statistics and Machine Learning Toolbox`, not in the ordinary one.
Also note that due to some MATLAB peculiarities, when you run our code for the *very first time* in a foreign directory, the `saveas` functions in our code *may* throw an error (because you're trying to run our code in a "foreign" directory, it *may* throw a `handle not found error`). However, when you run it the second time, it will run smoothly without any glitches. Thus if your machine throws an error for the **saveas** function the first time, please press the run button again.

3 Problem 1

3.1 Introduction

This is a really good problem for understanding several critical aspects of Bayesian Estimation and points of difference with the plain Likelihood estimator. The question also sheds light on the importance of choosing a correct prior and how overconfidence in one's prior can actually result in adverse consequences.

Let us go over the algorithms used to solve the problem.

3.2 Code Flow

We start with specifying the sizes of our samples, and defining the standard variables given as data in the question. Next, we initialise arrays for holding the data to be entered into the boxplot. Next we start with the main function.

We call a function `error_arr()` which calculates and returns the error arrays for various value of sample set sizes. The remaining part of the code apart from the functions is just

decorating, arranging and labelling the box-plot to make it presentable.

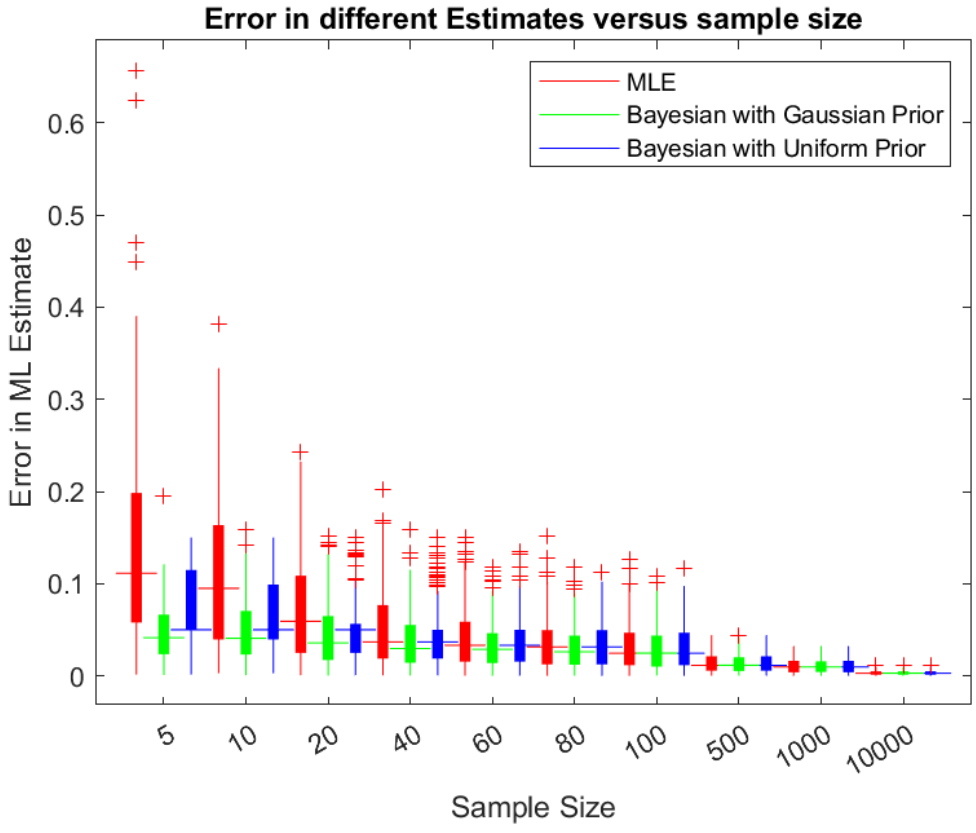
The `error_arr()` function further calls a `dataset()` function, which simply generates a required sample of the given distribution. Next we call the Max Likelihood Estimation functions and the Bayesian Estimation functions. These functions simply implement the required tasks.

The MLE function simply calculates the sample mean. The Bayesian estimation using Bayesian Prior simply uses the function using product of Gaussian distributions, basically weighted average of the means according to inverse variance. The Bayesian estimation using uniform prior simply condenses the samples to the given range in case they lie outside the given range.

This way we get a simple algorithm to calculate the estimations. Repeating this around 200 times, we get a whole set of arrays. This is what we plot in the box-plot.

3.3 Graphs and Results obtained

Finally we observe the following result-



3.4 Observations and their Rationalization

Its clear how the error decreases as N increases for all the three estimates. This is expected since more data tends to lead to better estimates.

Also, for the Bayesian estimate with Gaussian estimate, more data helps to reduce any biases imposed due to prior. This is evident here, as the prior assumes Gaussian centered at 10.5 while in reality the μ_{real} is 10. But we see the effects of this become negligible as N increases.

All three estimates become nearly perfect at large N. But in reality we won't always have the access to such large data. **In lack of data, one should always try to define a reasonable prior on the data.** This would really help in improving the quality of the estimate.

As an example, in the simple comparison of MLE and Gaussian estimate with Uniform prior, one can see how the uniform prior does not add any information about the distribution except the region in which the real value is expected to exist. Still, the error difference is significant. Why? **This is because the sample variance is still very high at low sample sizes and the sample mean might very well be far away from the true value**. The uniform distribution helps clip such cases and maintain proper locality.

Gaussian Prior also performs a similar action on the likelihood function and we see even better results at low sample sizes. Not only that, **Gaussian Prior seems to be the best choice for all sample sizes, equating in performance to others as the sample sizes increase.**

This shows how even non perfect priors can prove to be useful. Again, this highlights the philosophy of statistics, of trying to make the best out of whatever limited information and resources we might have, to push the models to their best.

Given a choice, **we will choose Gaussian prior here**, but in general, it might be better to try to come up with several prior distributions and see which seems to fit best to the data.

4 Problem 2

4.1 Introduction

This problem has a minimal coding aspect which is unremarkable but for it's theoretical background, this problem also asks us to make 2 derivations, which we shall make in the subsections below. The coding part is exactly similar to the previous problem, thus we skip the in-depth explanation here. The difference is only in the exact formulae of λ here and Gaussian μ and σ in the previous question.

4.2 Analytic Form of y

This is an example of transformation of random variables. We know that if X and Y are two continuous random variables, $r(x)$ is a function (assume bijective for now) such that $Y = r(X)$, and the PDF of X is $p(x)$, then we have that the PDF $g(y)$ of Y should be

$$g(y) = p(r^{-1}(y))|\frac{d}{dy}r^{-1}(y)|$$

In our case we have $p(x) = \mathbf{1}_{x\in[0,1]}$ (since $X \sim \text{Unif}(0,1)$) and $r(x) = -\frac{1}{\lambda}\log(x) \implies r^{-1}(y) = \mathrm{e}^{-\lambda y}$. Thus, keeping in mind that $\lambda > 0$ we have that

$$g(y) = \mathbf{1}_{\mathrm{e}^{-\lambda y}\in[0,1]}|\frac{d}{dy}\mathrm{e}^{-\lambda y}| = \lambda\mathrm{e}^{-\lambda y}\mathbf{1}_{y\geq 0}$$

ie:-, Y has the **exponential distribution with parameter** λ.

4.3 Derivation of $\hat{\lambda}^{\text{ML}}$

Let our dataset at hand be $S := \{y_1, y_2, ..., y_n\}$. Then, considering that we have an exponential distribution with parameter λ, the likelihood function $\mathcal{L}(S;\lambda) = \prod_{y\in S}\lambda\mathrm{e}^{-\lambda y} = \lambda^n\mathrm{e}^{-\lambda\sum_{y\in S}y}$. Taking the logarithm of $\mathcal{L}(S;\lambda)$ and differentiating it w.r.t λ yields

$$\hat{\lambda}^{\text{ML}} = \frac{n}{\sum_{y\in S}y}$$

4.4 Derivation of $\hat{\lambda}^{\text{PosteriorMean}}$

We know that the posterior distribution is $\propto$ Prior $\cdot$ Likelihood.
Now, $\text{Prior}(\lambda;\alpha,\beta) = \Gamma(\lambda;\alpha,\beta) = \frac{\beta^\alpha}{\Gamma(\alpha)}\lambda^{\alpha-1}\mathrm{e}^{-\beta\lambda}$, while $\text{Likelihood}(S;\lambda) = \mathcal{L}(S;\lambda) = \lambda^n\mathrm{e}^{-\lambda\sum_{y\in S}y}$. Thus,

$$\text{Posterior}(\lambda;S) \propto \text{Prior}\cdot\text{Likelihood} \propto \lambda^{\alpha-1}\mathrm{e}^{-\beta\lambda}\cdot\lambda^n\mathrm{e}^{-\lambda\sum_{y\in S}y} \propto \lambda^{n+\alpha-1}\mathrm{e}^{-\lambda(\beta+\sum_{y\in S}y)}$$

$$\propto \Gamma(\lambda; n+\alpha, \beta+\sum_{y\in S}y)$$

Now comes a very crucial step: **Note that once we get the functional form of the posterior distribution in terms of some well known distribution, we can ignore the intermediate constants that arise in our calculations, such as the probability of the evidence, and after the desired functional form has been derived, we can simply plug in it's normalization constant at the end**.
Thus,

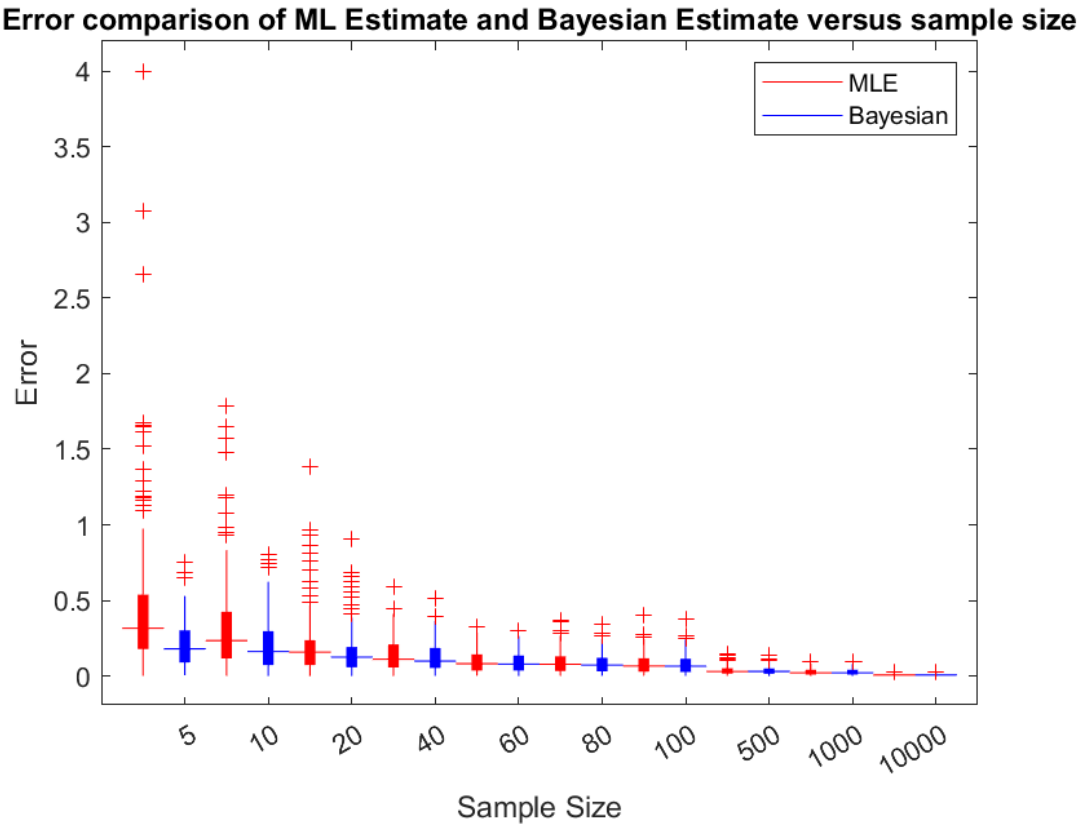
$$\text{Posterior}(\lambda;S) \propto \Gamma(\lambda; n+\alpha, \beta+\sum_{y\in S}y)$$

$$\implies \text{Posterior}(\lambda;S) = \frac{(\beta+\sum_{y\in S}y)^{n+\alpha}}{\Gamma(n+\alpha)}\lambda^{n+\alpha-1}\mathrm{e}^{-\lambda(\beta+\sum_{y\in S}y)}$$

$$\implies \hat{\lambda}^{\text{PosteriorMean}} = \mathbb{E}_{\text{Posterior}(\lambda;S)}[\lambda] = \frac{n+\alpha}{\beta+\sum_{y\in S}y}$$

4.5 Results

After the box-plots asked in the question were drawn, these were the results obtained:



As is clear from the graph, the **Bayesian estimates outperform the MLE ones** (in terms of both median error and deviation): Why could be that so? It's because since the prior on λ is "close" to the true value in some sense (especially the shape parameter $\alpha = 5.5$ being close to $\lambda_{\text{true}} = 5$), the data **reinforces** the prior, and thus converges quickly as compared to the MLE estimates.

As we have seen many times before, the **variances of both the estimates tend to zero as $N \longrightarrow \infty$, in accordance with the Law of Large Numbers.**

Thus we would **choose the Bayesian estimate (with the given prior) over the MLE one in our case**.

5 Problem 3

5.1 Introduction

This problem is entirely theoretical and has no coding aspect to it, and instead quite a few derivations have been asked for, which we shall provide one by one in the sub-sections below.

5.2 Derivation of $\hat{\theta}^{\text{ML}}$

We know that the probability distribution $g(x)$ for Unif$(0, \theta)$ is $\frac{1}{\theta}\mathbf{1}_{x\in[0,\theta]}$. Thus if our dataset is $S := \{x_1, x_2, ..., x_n\}$, then our likelihood function $\mathcal{L}(S;\theta)$ is $\prod_{x\in S} g(x;\theta) = \prod_{x\in S} \frac{1}{\theta}\mathbf{1}_{x\in[0,\theta]}$.

In order to maximize our likelihood function, we must first ensure that it's non-negative: Indeed, if the maximum entry $x_{\max}$ in S is greater than the variable θ, then $g(x_{\max};\theta) = 0$ and consequently $\mathcal{L}(S;\theta) = 0$ too. Thus, we have that $\theta \geq x_{\max}$.

Under that constraint, we lift the indicator functions to get that $\mathcal{L}(S;\theta) = \frac{1}{\theta^n}$. Since we also know that $\theta \geq x_{\max}$, we can conclude that the **maximum likelihood estimator** for θ is $x_{\max}$, ie:- **the largest element in the set** S.

Hence $\hat{\theta}^{\text{ML}} = \max(S)$, where S is the dataset we have at hand.

5.3 Derivation of $\hat{\theta}^{\text{MAP}}$

We know that $\hat{\theta}^{\text{MAP}}$ is the **mode of the posterior distribution**. Thus our goal will be to determine the posterior distribution for the given prior and likelihood functions.

Note that the prior is the **Pareto Distribution**, which has the form $P(\theta;\theta_m,\alpha) \propto (\frac{\theta_m}{\theta})^\alpha \mathbf{1}_{\theta\geq\theta_m}$, with the shape parameter $\alpha > 1$ and the scale parameter $\theta_m > 0$.

We also know that the posterior distribution (under the same constraints as $\theta \geq x_{\max}$)

$$f(\theta) \propto \text{Prior} \cdot \text{Likelihood} \propto (\frac{\theta_m}{\theta})^\alpha \mathbf{1}_{\theta\geq\theta_m} \cdot \frac{1}{\theta^n} \propto (\frac{\theta_m}{\theta})^{n+\alpha} \mathbf{1}_{\theta\geq\theta_m} \propto \text{Pareto}(\theta;\theta_m, n+\alpha)$$

where we **we have conveniently ignored the normalization factors at each step with the proportionality symbols: Indeed, once we have the functional form of the distribution of the posterior, the normalization factors can be deduced at the end without requiring the intermediate constants in the calculation**.

Thus, our posterior distribution is $f(\theta|S) \sim \text{Pareto}(\theta;\theta_m, n+\alpha)$, and $\hat{\theta}^{\text{MAP}}$, which is the mode of the distribution, can be readily seen to be θ_m (since the PDF is strictly decreasing from θ_m onwards).

Hence $\hat{\theta}^{\text{MAP}} = \theta_m$.

5.4 Derivation of $\hat{\theta}^{\text{PosteriorMean}}$

Since our posterior distribution is just Pareto$(\theta;\theta_m, n+\alpha)$, $\hat{\theta}^{\text{PosteriorMean}}$, which is the mean of the posterior distribution taken over the unknown parameter θ, is equal to $\mathbb{E}_{\text{Pareto}(\theta;\theta_m,n+\alpha)}[\theta] = \frac{n+\alpha}{n+\alpha-1}\theta_m$, where the last step was derived using a standard integration.

Hence $\hat{\theta}^{\text{PosteriorMean}} = \frac{n+\alpha}{n+\alpha-1}\theta_m$.

5.5 Does $\hat{\theta}^{\text{MAP}} \longrightarrow \hat{\theta}^{\text{ML}}$ as $n \longrightarrow \infty$?

Note that while $\hat{\theta}^{\text{MAP}}$ is a constant and doesn't vary with n, $\hat{\theta}^{\text{ML}}$ does vary with n, but it's relationship can't be expressed analytically, as $x_{\max}$ is itself a random variable. However,

as $n \longrightarrow \infty$, we **can claim that** $\hat{\theta}^{\mathrm{ML}} \longrightarrow \theta_{\mathrm{true}}$, where θ_{true} is the true value of θ.

Thus, if $\theta_{\mathrm{true}} \neq \theta_m$, then in fact $\hat{\theta}^{\mathrm{ML}}$ **will not converge towards** $\hat{\theta}^{\mathrm{MAP}}$.

This is a **very undesirable situation as the effects of a prior aren't "washed away" by the actual evidence available**, and thus a **misguided prior will continue to influence results even as** $n \longrightarrow \infty$. Only in the case when we have already guessed θ_{true} in the prior itself, ie:- $\theta_{\mathrm{true}} = \theta_m$, will the Bayesian and Maximum Likelihood estimates coincide for large n.

5.6 Does $\hat{\theta}^{\mathrm{PosteriorMean}} \longrightarrow \hat{\theta}^{\mathrm{ML}}$ as $n \longrightarrow \infty$?

Note that $\hat{\theta}^{\mathrm{PosteriorMean}} = \frac{n+\alpha}{n+\alpha-1}\theta_m \longrightarrow \theta_m$ as $n \longrightarrow \infty$, and $\hat{\theta}^{\mathrm{ML}} \longrightarrow \theta_{\mathrm{true}}$ as argued above, and as with the above case if $\theta_{\mathrm{true}} \neq \theta_m$ then $\hat{\theta}^{\mathrm{PosteriorMean}} \not\longrightarrow \hat{\theta}^{\mathrm{ML}}$.

Thus $\hat{\theta}^{\mathrm{PosteriorMean}} \longrightarrow \hat{\theta}^{\mathrm{ML}}$ **iff** $\theta_{\mathrm{true}} = \theta_m$, and for the exactly the same reasons stated above, this is not a desirable situation to be in.