

Assignment 2

Joshua Petersen: PTRJOS017

24/04/2022

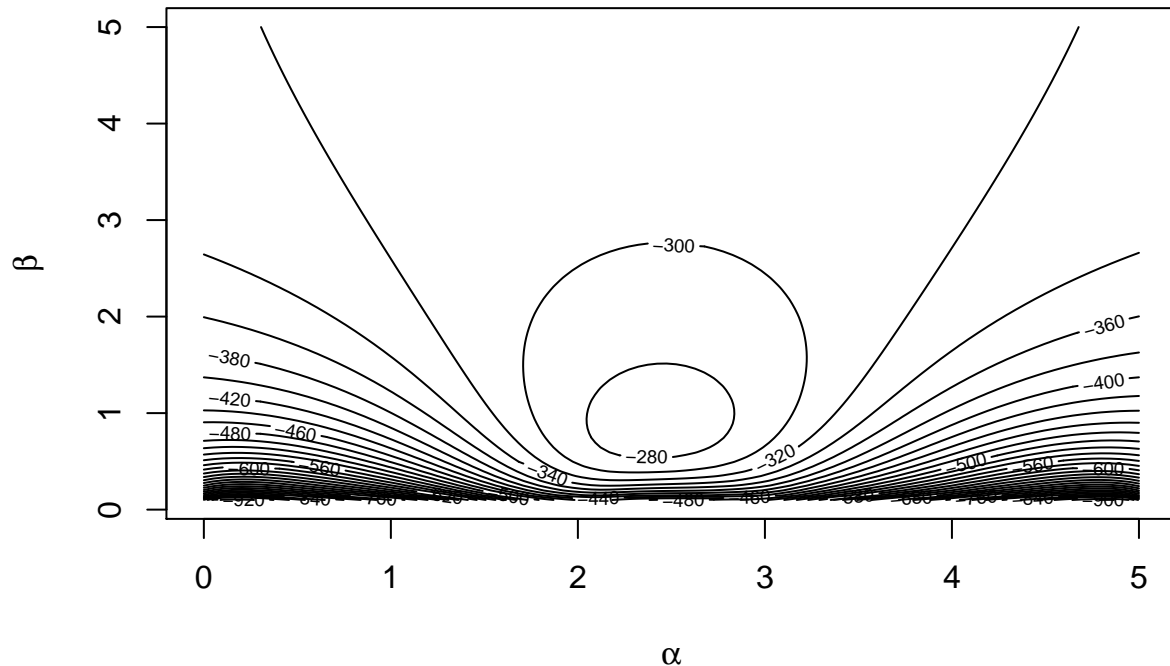
Introduction

In this report we will be investigating the problem of estimating the location of a ship out to sea based on some simulated data of flashes of light from the ship recorded on the coast line. The boat is described as being some position α along a coastline and some distance β out to sea. α is assumed to be uniformly distributed between 0 and 5 and β is assumed to be gamma distributed with shape parameter 4 and scale parameter 8. We will be exploring the likelihood function for the position of the ship, finding its maximum, applying an acceptance-rejection algorithm to sample from the posterior distribution, and performing monte-carlo integration to estimate probabilities based on the derived posterior distribution.

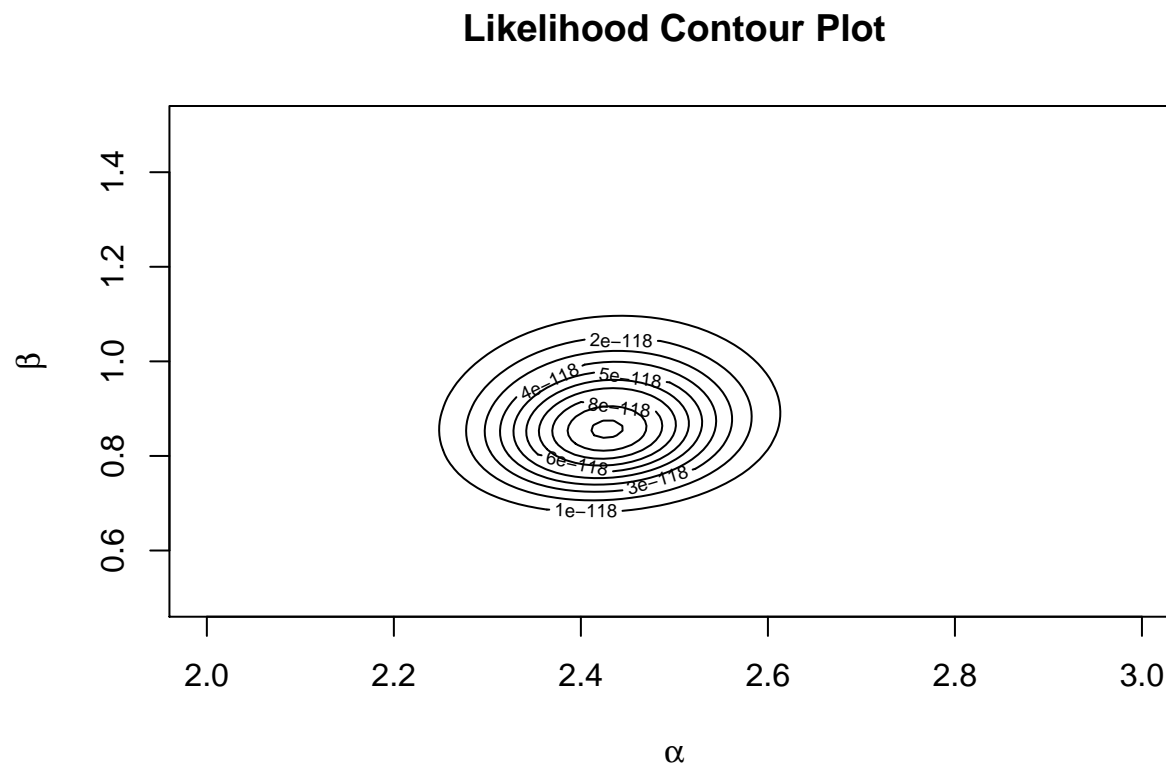
The Likelihood Function

In this section we will describe the given likelihood function derived from 200 points of simulated data and generate relevant plots thereof.

Log-likelihood Contour Plot



Here we generate a contour plot of log-likelihood function. We can see based on this plot that the maximum of the log likelihood function is likely to be in the region where alpha is between 2 and 3 and beta is between 0.5 and 1.5. For interest, we construct a contour plot of the (original) likelihood function in this region:



Here we can see that the maximum of the likelihood function (within the search range) is around $\alpha = 2.4$ and $\beta = 0.8$. We will verify this by finding the MLE of the log-likelihood function. It should be noted that this is not equivalent to the best estimate of the ship's location, as we still have not incorporated any information about the prior distribution of alpha and beta. It should also be noted that the likelihood function is taking on extremely small values. We will later scale it for ease of use and interpretation.

Finding the MLE

In this section we will transform alpha and beta to unconstrained variables, use these transformed variables in a Newton algorithm to find the maximum of the likelihood function, and verify our result with R's "optim" function.

Alpha and beta are constrained random variables: alpha is bound to the interval $[0, 5]$ and beta is bound to the interval $[0, \infty)$. To ensure we search the entire solution space, we must first transform alpha and beta to unconstrained random variables. The logit function maps the interval $[0, 1]$ to the interval $(-\infty, \infty)$. We first divide alpha by 5 and then apply the logit function, transforming it to the unconstrained random variable α^* . The natural logarithm function maps the interval $[0, \infty)$ to the interval $(-\infty, \infty)$. We take the natural logarithm of beta, transforming it to the unconstrained random variable β^* .

After applying the transformation, we apply a Newton-Raphson algorithm to the log likelihood function to find its maximum. From our investigation of the contour plot, we know the MLE is likely to be around $\alpha = 2.4$, $\beta = 0.8$. We'll take $\alpha = 2$ and $\beta = 1$ as a starting point to ensure we do not miss a solution. Applying our transformation to $(2, 1)$, we begin our search at $\alpha^* = -0.4$ and $\beta^* = 0$. The Newton Algorithm returns the following solution:

	MLE
alpha	2.4278650
beta	0.8563737

We perform the same optimisation using R's "optim" function, returning the following result:

	MLE
alpha	2.4278600
beta	0.8563567

As we can see, we get a nearly identical result, suggesting our Newton algorithm has worked as intended. Minor discrepancies can be explained by the tolerance level in our Newton algorithm, which was arbitrarily set to perform optimisation up to 5 decimal places.

Acceptance-Rejection Sampling

In this section, we will use the acceptance rejection algorithm (AR) to sample from $\pi(\alpha, \beta|x)$ (the scaled posterior distribution function). We will be using $h(\alpha, \beta) = \pi(\alpha)\pi(\beta)$ as a candidate function.

From our prior assumptions, we know that alpha is uniformly distributed over the interval (0,5). $\pi(\alpha)$ is therefore given as $1/(5-0) = 0.2$. We have assumed that $\beta \sim \Gamma(4, 8)$. We can therefore find $\pi(\beta)$ using R's dgamma function (i.e. $\pi(\beta) = \text{dgamma}(\beta, 4, 8)$). Therefore, $h(\alpha, \beta) = \pi(\alpha)\pi(\beta) = 0.2 * \text{dgamma}(\beta, 4, 8)$

Assuming α is independent of β , $\pi(\alpha, \beta) = \pi(\alpha)\pi(\beta)$. We can therefore find $\pi(\alpha, \beta)$ in R as $\pi(\alpha, \beta) = 0.2 * \text{dgamma}(\beta, 4, 8)$. $\pi(\alpha, \beta)$ is therefore the same function as the candidate function, $h(\alpha, \beta)$. It should be noted here that $\pi(\alpha, \beta)$ is actually only a function of β , as $\pi(\alpha)$ is constant over its domain. Graphs of $\pi(\alpha)$, $\pi(\beta)$, and $\pi(\alpha, \beta)$ have been included as an appendix.

To begin, we must first determine the range of values our candidate function can take on. Because our candidate function $\sim 0.2 * \Gamma(4, 8)$, we know its mode (i.e. maximum) is equal to $(4-1)/8 = 3/8$, which means the function ranges between 0 and $0.2 * \text{dgamma}(3/8, 4, 8) = 0.35847$. We have already assumed that the domain of the function in terms of alpha is between 0 and 5. In order to perform acceptance rejection sampling, we will need to restrict the domain in terms of beta to a manageable interval. From our inspection of the likelihood function and from what we know about the gamma distribution, it is highly unlikely even for a very large simulation that our algorithm will accept many points past $\beta = 2$. Even this is generous - my preliminary application of the algorithm to 10 million points accepted none greater than $\beta = 1.2$

We now have a region to sample points from - drawing a box in 3-space with:

$$\begin{aligned} 0 < \alpha < 5 \\ 0 < \beta < 2 \\ 0 < h(\alpha, \beta) < 0.35847 \end{aligned}$$

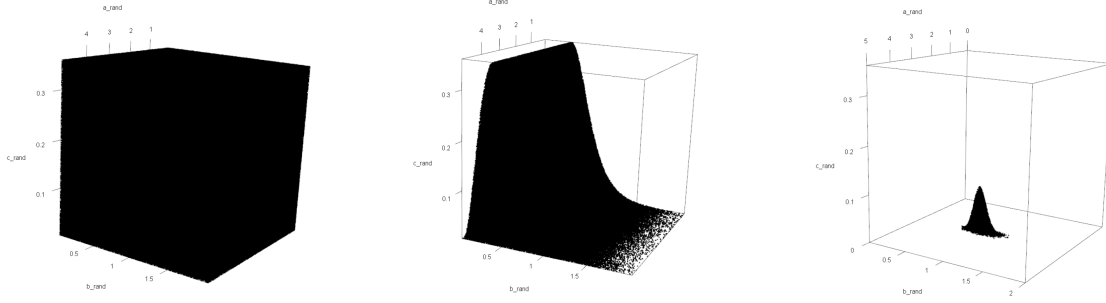
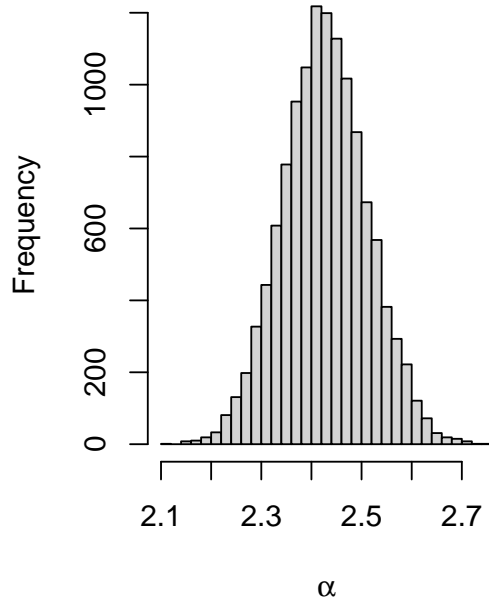
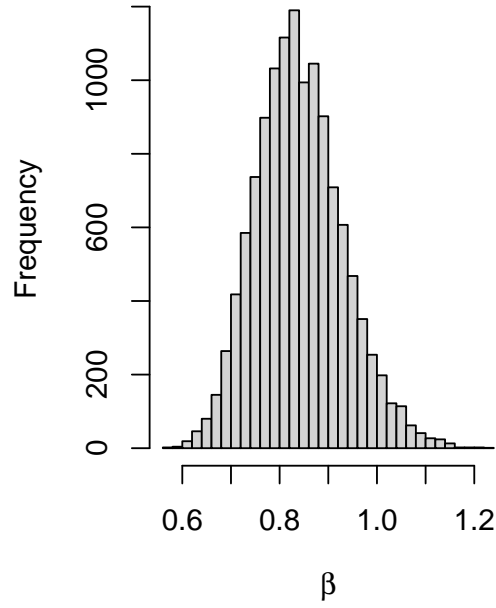


Figure 1: Acceptance-Rejection visualisation (3 panels)

We generate 10 million points within this box. These points have been plotted in three dimensions in the first panel of in figure 3. The next step in our algorithm is to keep only the points that fall beneath the candidate function. These points have been plotted in the second panel of figure 3. The third step is to keep of those points that fall under the posterior distribution. These points have been plotted in the third panel of figure 3. It should be noted that the posterior distribution function has been scaled here. As previously mentioned, the likelihood function returns very small values. The likelihood function has been scaled here such that its peak is 1 unit high. This ensures that the posterior distribution fits perfectly under the candidate function. Traditionally, we scale the candidate function and not the function we are sampling from, but in this case it gives the exact same results in terms of alpha and beta, and the sampled points can be scaled however we want after the fact. If we wanted a true reflection of the posterior function, we should scale them such that the volume under the surface of the function integrates to 1, but for our purposes this is unnecessary as the results for alpha and beta hold regardless of the scale of the posterior function.

A closer look at the scaled posterior has been included in the appendix. We can now use the sampled points to answer questions about the location of the boat. Of 10 million initial sample points, 12474 were ultimately accepted (the acceptance rate could be improved by refining the interval ranges or by specifying a more neatly fitting candidate function). The alpha and beta values of these points have been plotted on the following histograms:

Histogram of sampled alphas**Histogram of sampled betas**

From these histograms we observe that the modal value for alpha was around 2.42 and the modal value for beta was around 0.82. Using optim to estimate the mode of the posterior distribution yields the following results:

	MLE
alpha	2.4248851
beta	0.8210894

This verifies the result of our acceptance rejection sampling algorithm, which found similar values for the mode of the function in terms of alpha and beta

Of our 12474 accepted points, 10568 (84.7%) had beta values greater than 0.75 - suggesting that $\Pr(\text{beta} > 0.75)$ is approximately 0.847

The correlation coefficient between the alpha and beta values of our sample points was found to be 0.071, suggesting weak or no correlation. This is exactly what we'd expect given our assumption that alpha and beta are independently distributed.

Monte Carlo Integration

In this section, we use Monte Carlo methods to estimate $\Pr(2.4 < \alpha < 2.6)$, for a fixed value of $\beta = 0.8$. We begin by finding the area under the scaled posterior function (univariate in terms of alpha). The curve integrates to 0.01965. We scale the posterior function by the inverse of this value so that it will integrate to one and therefore be a true density function. We plot the curve of this univariate posterior distribution and shade the area which we mean to find the area of in red.

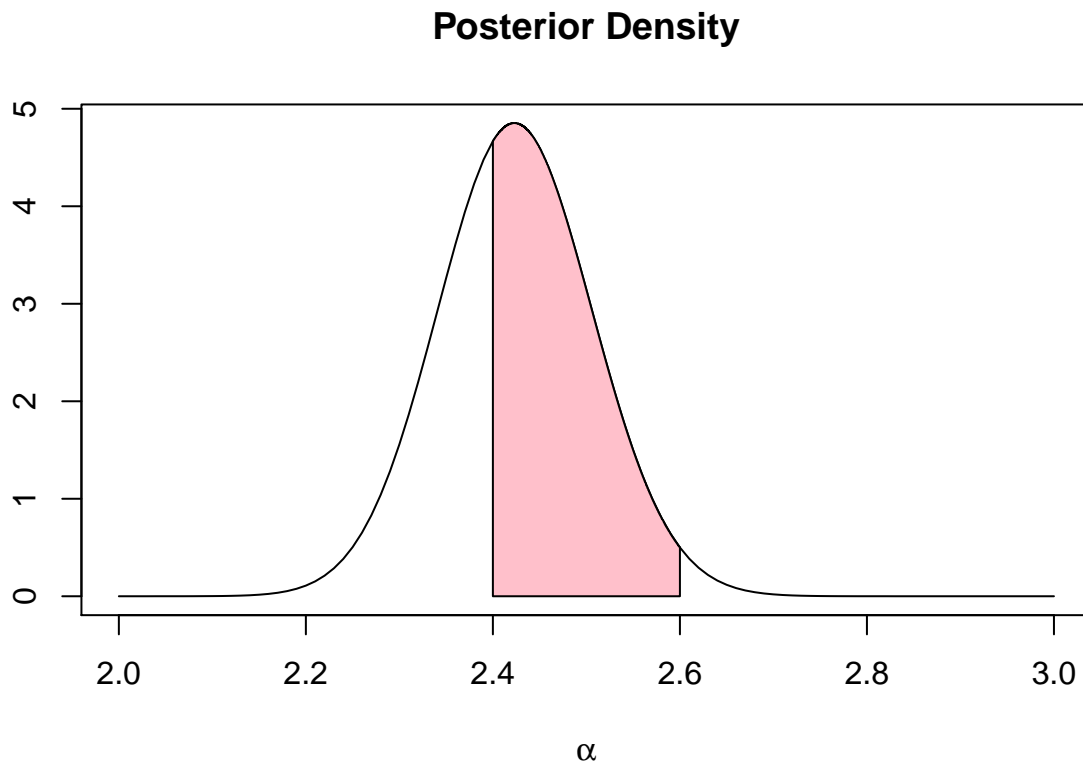


Figure 2: Probability density function of correctly scaled posterior distribution for beta = 0.8

We then randomly generate 10000 values uniformly distributed between 2.4 and 2.6 and parse them to the posterior function. We store the results and take their mean. We then multiply this value by the length of the interval to approximate the integral of the function from 2.4 to 2.6. This returns the value 0.5973183, implying that, for $\beta = 0.8$, $\Pr(2.4 < \alpha < 2.6)$ is approximately 59.73%

We verify this answer by performing numerical integration using R's `integrate` function. This returns the value 0.5973263, which matches our answer to 4 decimal places. A greater sample size would improve the accuracy of this result.

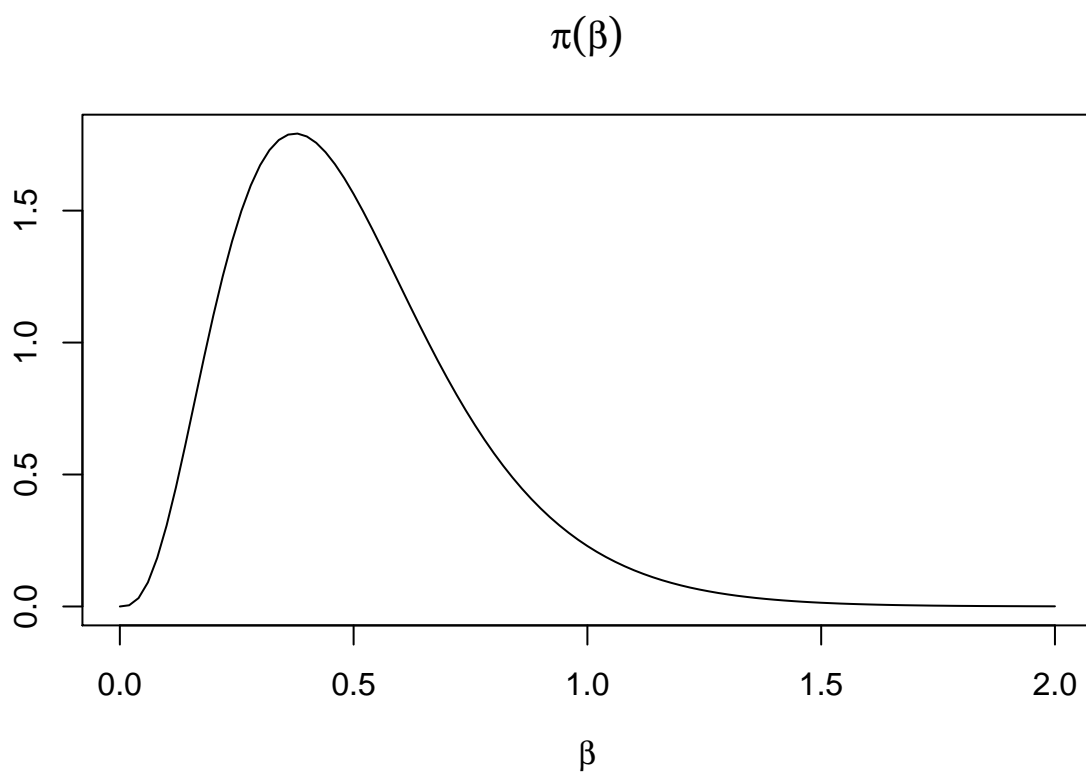
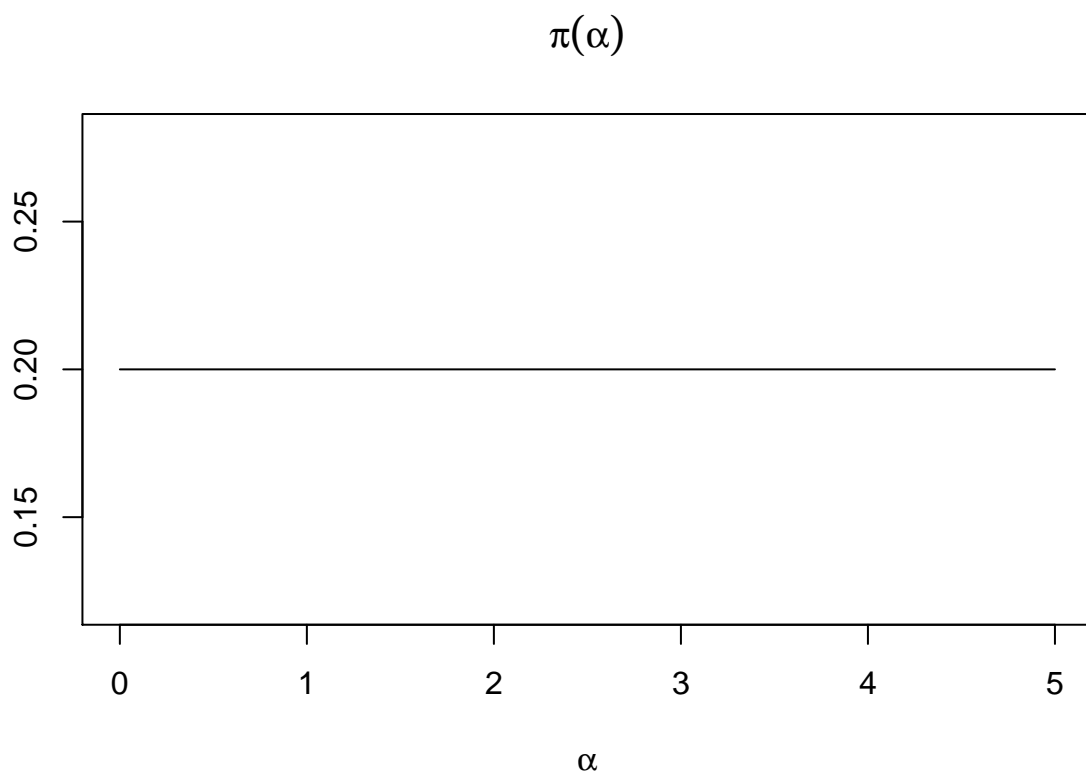
Conclusion

The most obvious takeaway from the various exercises performed here is that, based on the given data, the boat was estimated to be located at $\alpha = 2.42$ and $\beta = 0.82$. We came to this conclusion based on both numerical methods such as optimisation of the posterior function and on monte carlo methods such as the stated acceptance rejection algorithm.

References

- [1] A. Clark *Newton-Raphson Alogrithms* R Code, 2022
A. Clark *Simulated location values* Data, 2022
- [2] , R Core Team *R: A Language and Environment for Statistical Computing*, R Foundation for Statistical Computing, Vienna, Austria, 2021, <https://www.R-project.org/>

Appendix



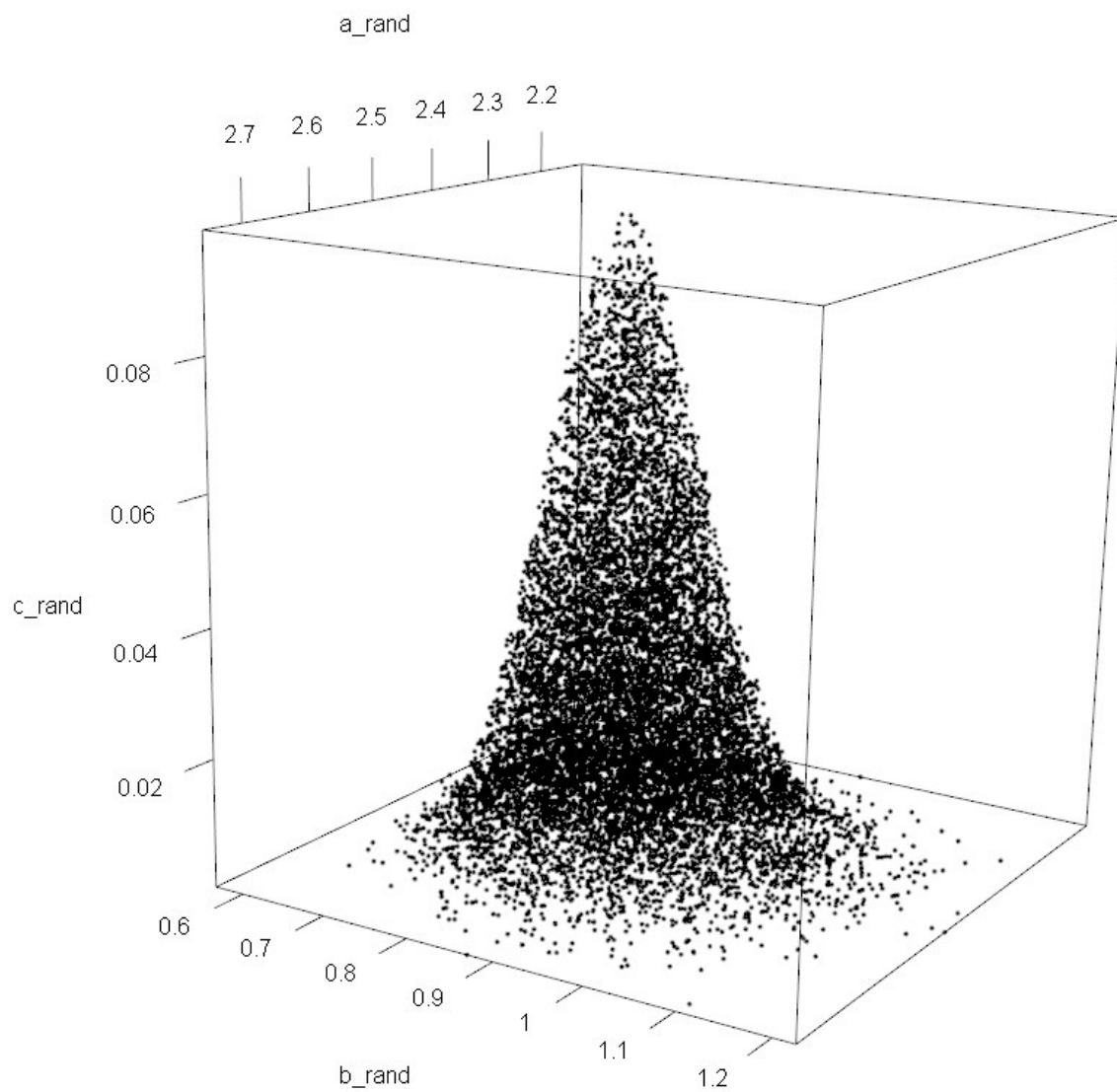


Figure 3: 12474 posterior samples (arbitrary scale)