|  |  |
| --- | --- |
| Methods to calculate Confidence intervals  With a numeric-oriented approach |  |

Daniel Abutbul

This report was written as a self-learning exercise

Contents

[Motivation 2](#_Toc108284274)

[Working Example 2](#_Toc108284275)

[Extrapolation and Confidence Intervals 3](#_Toc108284276)

[Other Literature 5](#_Toc108284277)

[Exact Solution 6](#_Toc108284278)

[∆*χ*2 Approach of Acceptable Models 9](#_Toc108284279)

[*χ*2 Reasoning and Systematic Noise 9](#_Toc108284280)

[∆*χ*2 Distribution, Acceptable Models Ellipse, and CI Bounds 9](#_Toc108284281)

[Monte Carlo Simulation of Prediction’s PDF 10](#_Toc108284282)

[Minimal Δ*χ*2 Best Fit 10](#_Toc108284283)

[With Robust Fitting 10](#_Toc108284284)

[Boot Strapping with Studentized Residual Resampling 11](#_Toc108284285)

[Bayesian Statistical modeling 12](#_Toc108284286)

[A Series of Coin Flips 12](#_Toc108284287)

[A Straight Line 16](#_Toc108284288)

[Metropolis Markov Chain Monte Carlo 17](#_Toc108284289)

[Implementation for the Bayesian Regression Problem 19](#_Toc108284290)

[Does Different Ways to update the prior matter? 19](#_Toc108284291)

[Appendix 21](#_Toc108284292)

[Analytic attempt to prove Bayesian CI 21](#_Toc108284293)

[Sources 24](#_Toc108284294)

# Motivation

For many applications, we encounter the problem of creating predictions based on observed data. While a very general case can be formulated, if one wishes to practice, understand, implement and finally use such predictions, it is best first to solve the simplest imaginable problem, and try and solve it in as many methods as possible, while pointing out their advantages and disadvantages.

In this report, we will review and implement various methods for extrapolating the value of a straight line calibrated from observed data.

We will assume certain knowledge in probability theory and linear regression.

## Working Example

Throughout this report, we will use the same simple example of the straight line:

The straight line formula tells us exactly the value of at any value of . For many real life situation, though, the formula and its parameters are unknown to the researcher. Instead, the researcher may have a sample of points which lie near the straight line, and originated from the straight line, but with added noise. This can represent for example a scientific experiment of measuring the value of at different ’s, with added measurement noise.

The sample is represented by a set of points , at each one we have the value . For our simplest working example, we assume the noise comes from a normal distribution, and we model it using a random variable , such that is sampled normally around the true value :

It is very common, and still useful, to restrict ourselves to the case that are i.i.d. variables:

Finally, our very specific working example will be of the data set:

The straight line, a possible sample points with noise from the straight line, and theirs best fit ( minimized) are given in Figure 1:

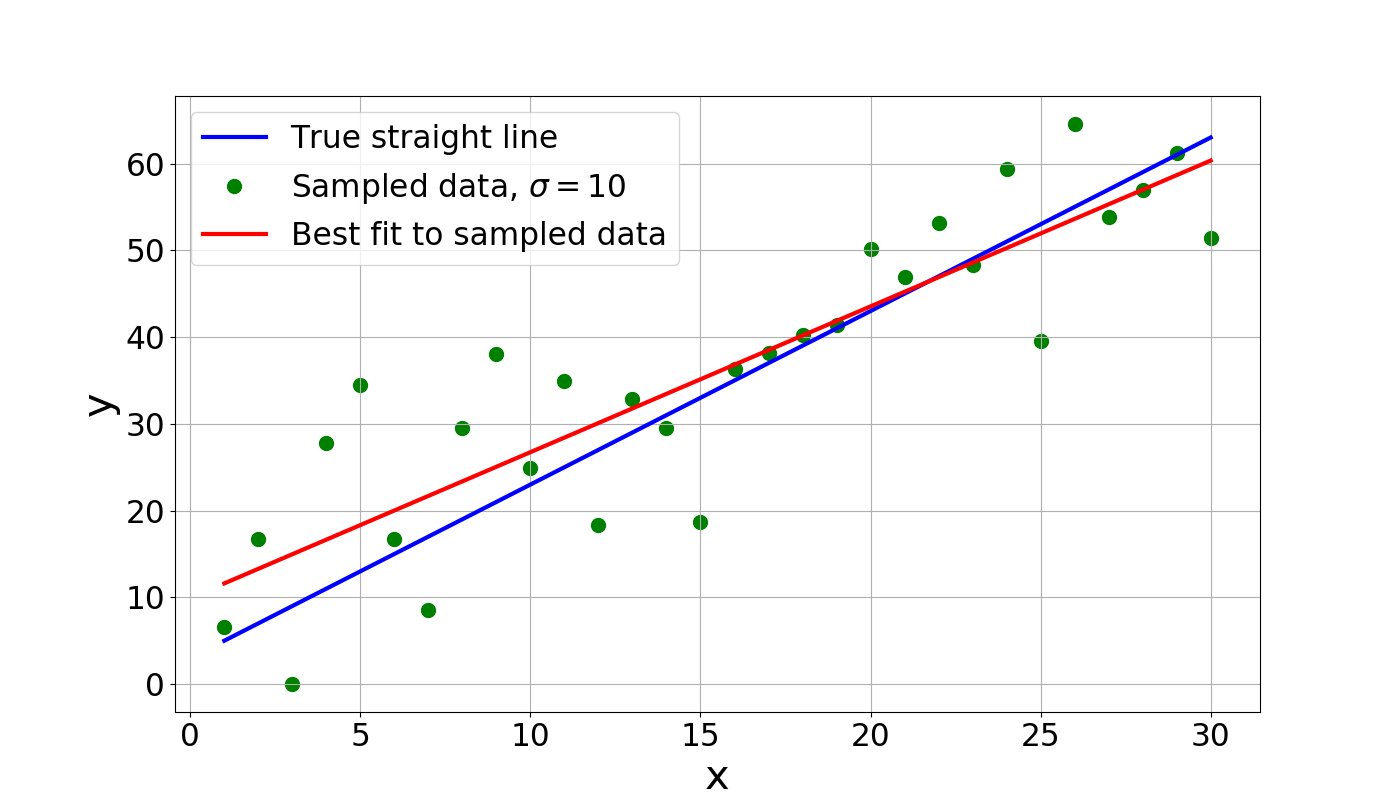


Figure 1: The straight line (blue), a sample from the straight line with normal noise ( i.i.d. random variables) (green dots) and a straight line fit (red line).

## Extrapolation and Confidence Intervals

Our task is to construct an extrapolation toward a value that is not in our data. In this report we will try to extrapolate the value of the straight line toward:

We will not only predict the value of at (that would most likely simply be the best fit to the observed data). We will decide on some interval around the extrapolated value, such that in most noise realization the true value of the straight line will be within our predicted interval, as demonstrated in Figure 2:

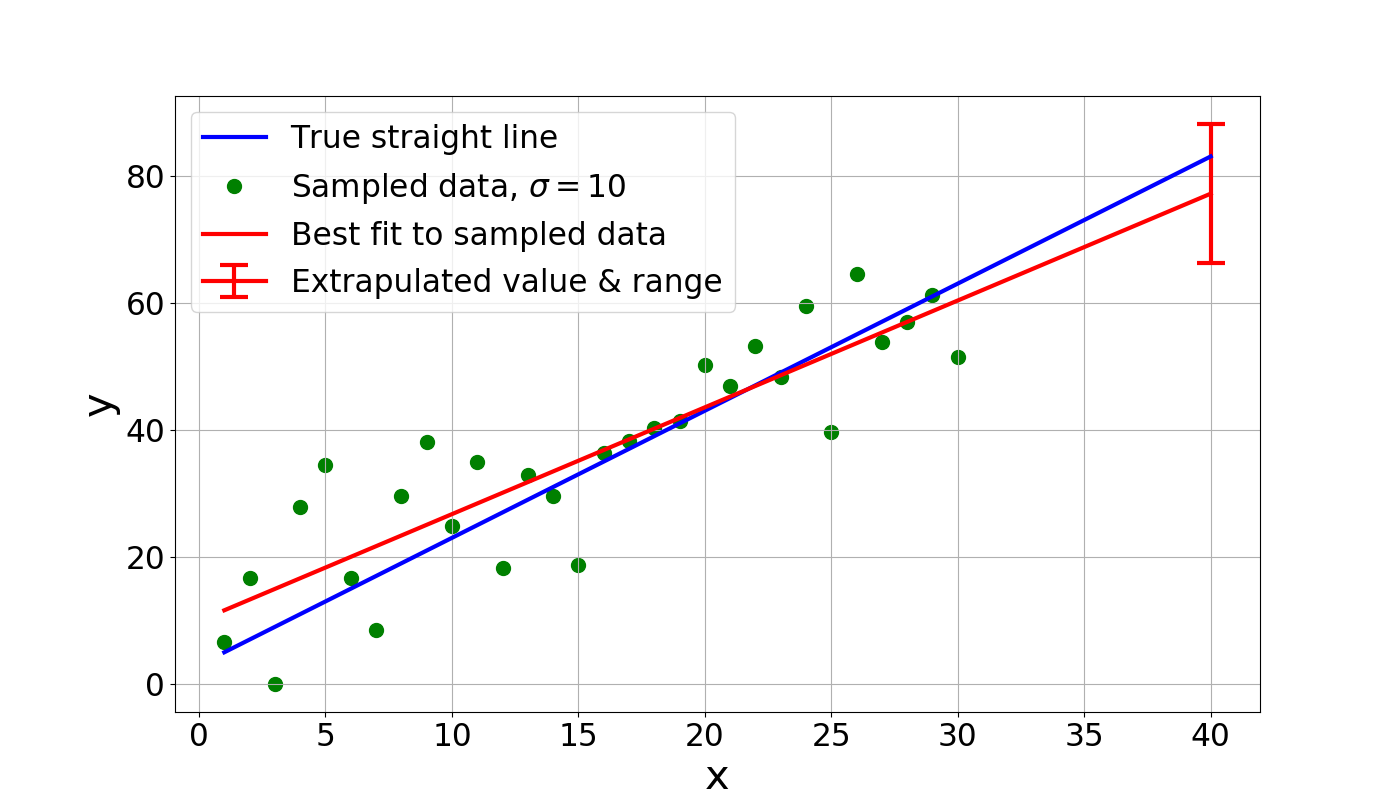


Figure 2: Extrapolation from sample x=1…30 toward and some interval around it. We choose the interval such that for most noise realizations, the true value of the straight line will fall within our interval.

The Confidence Interval (CI) is an interval around that extrapolated point, **constructed based on the sampled data**. We will say an interval is a 95%CI if the true value falls within the interval with confidence level (CL) of 0.95. Notice a 95%CI changes every noise realization; for each realization, we construct an interval based on the observed sampled data. For each realization, the true value will either be inside or outside that interval.

When we know the true line, which rarely happens in real life but does happen in this report, we can test our method for constructing 95%CI by running it for different noise realization. For each 95%CI construction method, we test if indeed in 95% of noise realizations the true line falls within that interval. That is, if indeed the CL is 95%.

In general, though, since we have only the sampled data and we construct some interval around it, we can never know for sure if the true line indeed falls within the constructed interval. When we call an interval a 95%CI in such a case, we actually **model** the noise, and then construct an interval such that we believe the CL is 95%, that is the probability the true line is inside that interval is 0.95.

## Other Literature

There is a lot of literature on the subject. The goal of this report is also to implement and practice the different methods, and so it does not claim novelty over existing literature (and definitely does not put as a goal to resemble a scientific peer-reviewed paper). Some main sources used in the preparation of this report are ‎[1]-[TODO].

# Exact Solution

Conveniently, the problem of calculating the 95%CI for the case of extrapolating a straight line has an exact solution. Since our goal is to test methods for CI construction in the simplest environment we can, having an exact solution is a great benefit.

Why then should we follow up in next chapters with all the other methods? Because the exact solution we will construct works only for the extrapolation of the straight line. For example, if we want to predict the value of some target function, say ( are the straight line parameters) the exact solution does not apply.

To construct the exact solution, we will use the linear regression language, for which a general model that is linear in its parameters is given by:



( is an vector and is vector. The result is a scalar)

represent the sample point at which we calculate (notice for a straight line: ), and are the parameters of the linear model. For the straight line we have and so . As mentioned above, we wish to investigate the case when we sample from a linear model with added noise, modeled by the normal i.i.d. random variables . In vector notion, we rewrite equation ‎(2) in the form:

is called the design matrix. For samples of a straight line at points we have . The best fit as function of the noise is given by ‎[2]:

( is an matrix, and is an square matrix which has inverse. is matrix, and is matrix, and so the result is an matrix)

Our best fit prediction to any point is given using equation ‎(6) with the replacement of the true parameters by the fitted ones :

Since our goal is to determine an interval around our prediction, it is natural to calculate the variance of . Also notice since are linear combination of the normal random variables , they are themselves normally distributed.

To calculate the variance, another useful result from linear regression models is that ‎[3], and so:

Since one can see that , and so to summarize is a normal random variable with mean and variance given by:



In normal distributions an interval of around the mean covers 95.4% of the area under the pdf, and so it will be most convenient to calculate the 95.4%CI throughout the report. Formula ‎(10) gives us then the 95.4%CI around any prediction constructed symmetrically around , since the probability for to fall within an interval of from the true is 0.954. We can plot the 95.4%CI interval as function of (for the same sample data from Figure 2), as shown in Figure 3:

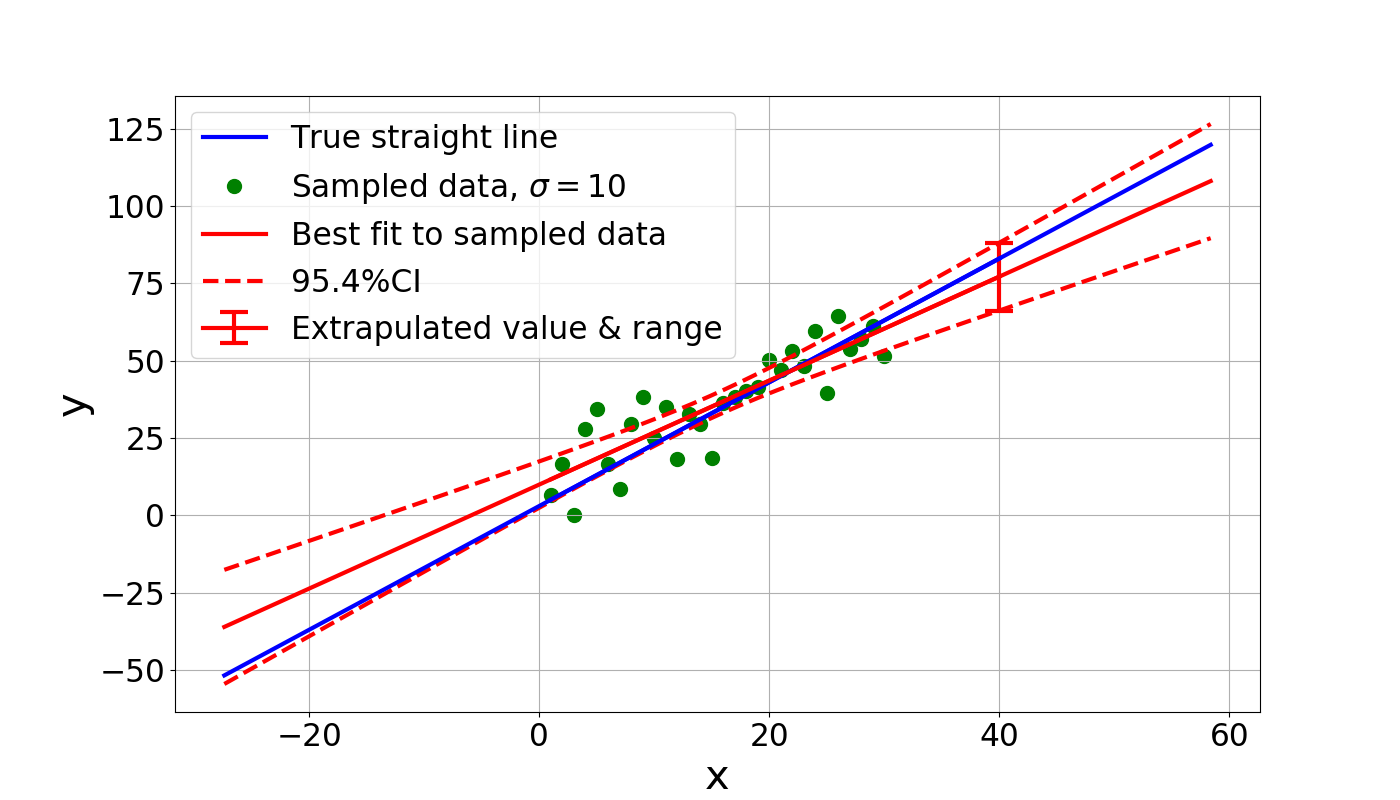


Figure 3: CI (95.4%CI) around the best fit. It can be seen that the prediction is more accurate around the middle of the sample, but as the extrapolation deviate from the sample the uncertainty rises.

The error bar at is of size 21.9, that is . The standard deviation is about half the size of our original noise’s standard deviation .

For any CI calculation method for which the interval magnitude is independent of the noise realization, like in the case of the exact solution, we should then get the same CI that was calculated exactly here, that is an interval of size 21.9.

We can still run our test and see that indeed the true value of falls within the constructed interval in 95.4% of cases. Using 100,000 monte carlo realizations, in 95.3% the true value falls within the CI interval, which means that: a. our exact solution is correct and b. from 100,000 Monte Carlo realizations we can expect an error of about 0.1% in the CL[[1]](#footnote-1).

# ∆*χ*2 Approach of Acceptable Models

## *χ*2 Reasoning and Systematic Noise

## ∆*χ*2 Distribution, Acceptable Models Ellipse, and CI Bounds

# Monte Carlo Simulation of Prediction’s PDF

## Minimal Δ*χ*2 Best Fit

TODO: It is completely pointless seen we have proven above that the CI interval is independent of the nosie realization

## With Robust Fitting

TODO: running MC here will be heavy, as for each noise realization one need to perform other MC realizations to calculate the CI interval. Maybe the CI interval is independent of the noise realization? It is possible… In such case calculating the CI interval once will be quite enough?

TODO: Choose robust fitting – median? Probably there is something very common in the literature to use here…

# Boot Strapping with Studentized Residual Resampling

TODO: See ‎[6]. Use Studentized as they are equally distributed, while the residuals are not (have different variance…)

# Bayesian Statistical modeling

All the methods mentioned above falls into the class of frequentist interpretation of probability. Following ‎[7], we will define the interpretation as considering probability as a marginal frequency after enough realizations were sampled.

In contrast, Bayesian statistical modeling starts by stating a prior probability distribution over the parameters to be calibrated. This distribution represents a belief the researched holds prior to measuring a sample. This belief is updated based on Bayes theorem: Given two subgroups , let the conditional probability of A given B be defined by:

Then we can update our prior beliefs on the parameters A, , according to observed sample B by:

Which is known as Bayes theorem. Is the belief of the event A posterior to measuring the event B, and are the prior beliefs.

## A Series of Coin Flips

Let a biased coin be tossed times, with a probability for heads (and for tails). From a series of tosses, a frequentist researcher would construct an estimator based on the sample ():

Where is the number of times the coin yelled heads. After enough tosses, it is guaranteed that the difference of and the real will vanish.

Let us now try to apply a Bayesian approach. In the Bayesian approach we have a model with free parameter q which we need to define a prior belief on its value, and we wish to update this belief according to the observed sample.

Let us start with a prior belief that is equally distributed in the interval : . Notice we defined a probability on the parameter, and so constructing a probability on an event such as the coin falling on heads is not immediate – one needs to sum over all parameter values:

For prediction a general sample of length , with , is binomially distributed ‎[8], and so needs to be constructed with respect to [[2]](#footnote-2):



(The integral was done using WolframAlpha). We see a constant prior gives equal distribution to all . For a general B the calculation can be done by asking how many different events B correspond to the same k, which is , and then . Notice it is quite different then estimating some for the prediction – then for any B we would have , and . In contrast, here we got quite a different structure for our predictions ( is constant!).

Now, our sample space is not just the results of a coin toss anymore, but rather we define a belief over the parameter space . Therefore, to update our belief over the parameter space, let A be the reality that the probability of heads is q, so it is updated by the observation of k by:



A demonstration of for () is given in Figure 4:

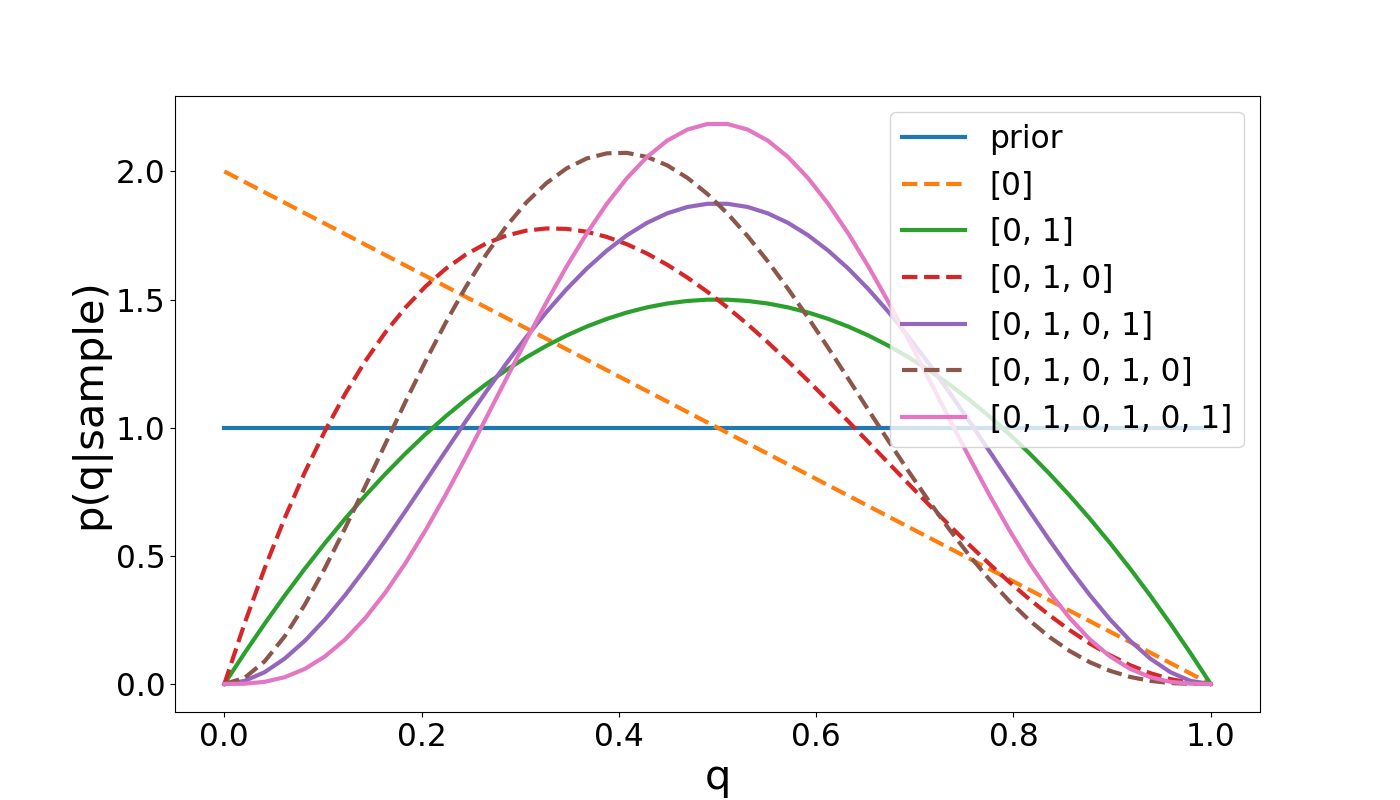


Figure 4: Evolution of the Bayesian belief of the parameter q of a binomial process, as we perform more tosses. Events with more 0’s (dashed lines) are biased toward q=0, and events with equal number of 0’s and 1’s form a symmetric distribution that get sharpened as we increase the number of tosses.

A prediction on a future event with heads from tosses is then given by:



Where in the last transition we calculated the integral via equation ‎(16) with the replacement .

The formula becomes quite tedious. We can simplify things by asking a simpler question, what is the probability after observing B to measure heads in the next toss, that is calculate equation ‎(18) for :



We can see the prediction is very similar to the frequentist prediction, and as (keeping the percentage of heads tosses, where is the true probability to get heads), the difference between the frequentist and the Bayesian predictions become negligible.

Finally, we need to go back to the question in hand throughout this report, which is: do CI (Credible Intervals) calculated using the method gives the correct intervals. Equation ‎(17) gives us a pdf on the parameter given a sample k, from it we can construct an interval which covers 95% of the area beneath the distribution. This interval is called the credible interval, and is the “Bayesian analog” of the frequentist confidence interval. We will denote it as well as . All is left is to answer the question: given a sample k drawn with a parameter , what is the probability that ? [[3]](#footnote-3)

To answer that, for each value of and true value of the coin we may generate a random sample , calculate and answer weather or not.

First, let us denote , then they satisfy the equation:



Now, using the special incomplete beta function:

We can solve the integrals:

We will choose a CI which is symmetric in the sense that . This choice of credible interval in Bayesian statistics is a highest density interval (HDI), which is a choice of minimal interval that covers 95% of the posterior (that is, interval which covers largest value of the posterior). This choice HDI will give us the solutions:

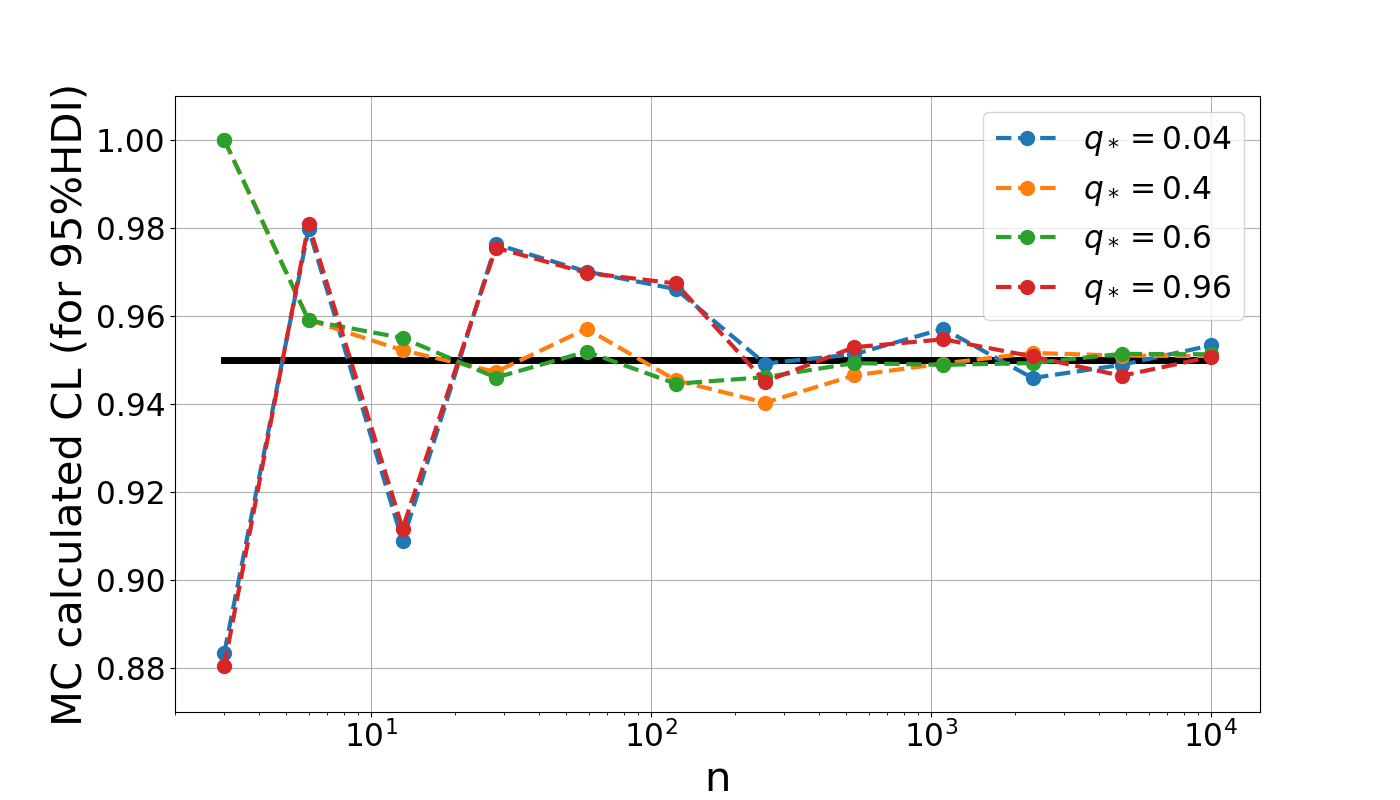


Figure 5: The numerically evaluated CL of , calculated from generating a realization of k a binomial sample with parameter , multiple times, and for each realization calculating and asking whether or not was inside that interval. We plot it as function of n the length of the sample, to show how our prior may introduce bias and how increasing n allows us to forget the prior. The bias introduced by our prior, comes also from our choice of structure of to be symmetrically two-tailed. For low n and close to we are biased toward high CL..

Notice for extreme values of the convergence is quite tricky. Indeed, by the construction of our CI, we can immediately see that for or the CL will be 0, because thanks to our prior and the HDI choice we always leave at least a small neighborhood of out of our HDI.

It is somewhat tricky and unclear if we could prove that after infinite samples we would get the correct CL, at least for values. It seems that for some cases, such proof that the Bayesian Credible Interval and the frequentist Confidence Interval coincide, exists ‎[12].

## A Straight Line

Going back to our working example of predicting the value of a fitted straight line, we need now to define a prior (). Given a sample we update our belief by:



Where:

First let us write , and then let see if we can choose a reasonable prior that will both allow us a wide interval of values for , and so applicable for as many cases as possible, and also will simplify our calculations.

Using equation ‎(6), we know that , where are i.i.d. with . Therefor, given & is also normally i.i.d., around with variance . We may write , and so:

That is, . If we choose a prior normally distributed in with mean and variance , then:

And so:

The integral can be carried out using tools for Gaussian integrals, but it will be quite inconvenient. Instead, we will use this problem as an exercise/motivation to employ a method which does not use the term at all: the Markov Chain Monte Carlo method to sample (that is, without explicitly calculating the normalization ).

## Metropolis Markov Chain Monte Carlo

A Markov chain, or Markov process, is a random process at which the probability to advance to the next state in the chain depends only on the current state. Denoting the current state by and the next state by , since the probability to advance from to depends only on the states , we may write:



Where is the transition probability. After infinite steps, we may assume that we arrive to a stationary distribution . This assumption, while given without a proof or evidence, can be very useful to many different systems. Applying with equation ‎(27) yields the condition of global balance:

That is, distribution does not change by the propagation of the Markov process.

We will use this process for the following quite common case: say we want to sample from a distribution , but we know it only up to a constant: , and C is unknown. This case also happens in statistical mechanics (where ), and it happens here for Bayesian process for which we do not know , but do know . We will design a Markov process, that is choose , in such a way that will satisfy equation ‎(28). Therefore, after enough steps of the Markov process, approximately a node will come from the distribution. Farther more, in the design of we will make use of only, without the unknown constant C.

The specific method we will do is implement a Metropolis algorithm, which is a type of a choice of using the function . In the Metropolis algorithm, given that we are at state , we generate a candidate from a distribution . Then we accept () or reject () that candidate with probability .

If we assume for simplicity , that is the new candidate , then we can calculate :



That is the probability to transit to given we are at is the probability to choose as the next candidate, times the probability to accept as the next node.

Next, a class of possible choices for which are sad to satisfy detailed balance are functions such that:

Which gives using equation ‎(29):

Where we have used the fact that . Such a choice of is called detailed balance. Given detailed balance we can show by showing satisfy equation ‎(28):

We have not specify yet how to choose , only that if it satisfy detailed balance (equation ‎(30)), then . Notice as represent a probability we must have . A common choice for , the Metropolis choice, is:



A few subtleties are we must have as is defined only when we are at a node and the next node is chosen as a candidate. For such that , and so , a good restriction on , otherwise will be illdefined. We can think of values for which as “illegal” by the distribution, and so it is reasonable to have for any illegal values of . The choice ‎(32) can be tested to satisfy equation ‎(30):

The last transition is since if then we get 1 in the numerator and in the denominator, and if then the numerator is and the denominator is 1.

Notice we almost not limited ourselves in the choice of , only that for such that we should choose also.

## Hamiltonian Monte Carlo

This algorithm ‎[11] is a type of Metropolis Markov chain algorithm, used in applications of Bayesian models to sample from target distributions . The approach uses a result of statistical mechanics that given a Hamiltonian:

A physical system evolves according to:

And when the system interacts with a thermal bath, the probability for each configuration is proportional to the Boltzmann factor:

Therefore, if we wish to sample from a target distribution , we may choose:

Now, the Hamiltonian Monte Carlo is a specific choice of . In the algorithm, which we will not describe in many details, for a configuration of the system , we suggest a configuration by generating momentums , and propagating positions in time from to , according to the Hamilton equation (use finite difference method such as leapfrog integration). Then, the new proposed configuration is the time propagated system , which we accept or reject as the new state in the Markov Chain. We generate the momentums in such a way that , which is quite natural here because the Hamiltonian dynamics is reversible, so if we arrived from to as the result of getting a momentum , then using we should go back from to . So if we generate the momentums symmetrically around zero, say normally with mean zero for example, then .

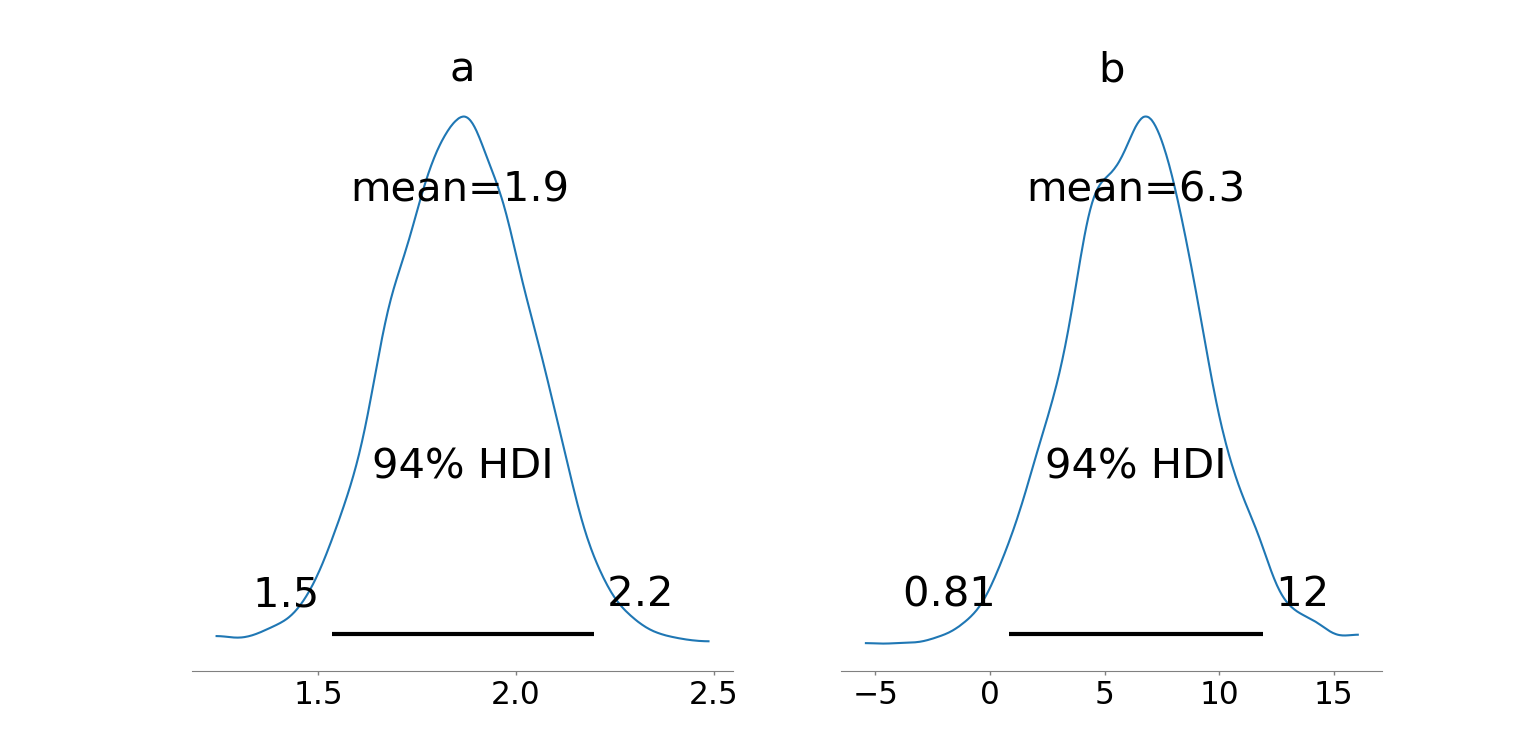
As it turns out, while the dynamics should be preserving energy, that is , in practice error enter via the ode integration. Therefore, we choose . That is, we acknowledge the error introduce from the integration as a deviation from conservation of energy, and instead of propagating the error as an error to the measure we sample, we allow the error by adding a Metropolis rejection probability. Therefore, we sample from the correct measure even when non negligible error in the integration is presence.

Notice the measure we sample for is rather then the target distribution. There are specific choices for the kinetic energy such as , for which the traces out in the integration and it is evident that . For choices where one needs to make sure such a trace out occurs.

## Implementation for the Bayesian Regression Problem

An implementation of the Hamiltonian Monte Carlo (HMC) method exists in pymc package ‎[13]. While it is somewhat an overkill for the task in hand, as we have an analytic form of the prior up to a constant, and it has only two parameters[[4]](#footnote-4), we will use it as an important exercise, and will use a tool which can be easily generalized to harder problems.

First, we look at our working example’s data, and calculate how the posterior looks like after updating the priors with the observed sample:



To construct these graphs, we used the priors given in ‎(23), with and . We see converges very quickly, but has a very large variance.

## Generalization for Unknown Variance

One of the advantages of the Bayesian approach, is for the case we do not know the variance of the noise . In the frequentist approach, one result to estimating it from the sample [[5]](#footnote-5), which in turn can lead to some complicated distribution. In the Bayesian approach, all we need to do is add to the parameters a parameter , choose the priors for and independent of and choose a prior for .

# Appendix

## Analytic attempt to prove Bayesian CI

To solve the question analytically, we can remind ourselves that actually depends only on , and so . Given , becomes a random variable with pdf which depends on , and for each value of we construct a which either contains or does not contains . We constructed correctly if in 95% of realizations of .

Let be the upper and lower bounds of the CI, that is , then the area beneath the curve in that interval should be 0.95, and so:

To simplify our construction, rather than building in an intuitive way around the maximal likelihood value , we will fix and adapt the upper bound such that equation ‎(32) will hold. is now a well-defined random variable, and we are left with the question, what is the probability that ?

To investigate the question, let us first examine the case . In that case our prior and so . The main problem is that it is independent of , and so for the CI will give 100%CL, while for the CI will give 0%CL. This is quite trivial, as we cannot predict anything before we collect data. But, as we will show next, having more data does not truly remove the problem.

To see the fundamental problem, lets say for a finite that we sample for (or very close to 1). For finite and our constant prior, will take some continuous form between , and since we constructed our CL to start from 0 and end at some , a value of (and its neighborhood) will never be part of the CL, again resulting in values of for which the true CL is 0%. One can claim this is the result of the way we constructed the CI to start from 0, but if we had symmetric CI then at both neighborhoods of 0 and 1 we will have that problem, and if we would have started the CI from 1 downward until some value , then around 0 we will have the same problem. Therefore, any method to construct the CI will introduce some “bias” against some values of , for which no matter what realization we will get there is a zero chance for getting , while other values of will have a too high probability (such as in our current CL construction method, since it will always be in the CL with probability 1).

Then we see that does not include in 95% of the noise realization, as it should, and so the Bayesian method is some-what false. What else can we claim that would be reasonable and will recover our trust in the Bayesian method? We can examine the **asymptotic** behavior for large !

As , we can hope to see the error, or the interval of values of for which our CI has a very wrong CL, disappear. This is reasonable as only with infinite amount of data we have a chance to truly forget our prior, which at the end can always lead us in a wrong path.

Stirling’s approximation :

And for large n we can try the assumption:

Which means for that all realizations will have , and so:

TODO: did I just downgrade k from a random variable to a number, leading for the CL being just a number, being some exact number, resulting in a 100%CL for some and 0% to others, again?

Using WolframAlpha:

Where is the hypergeometric function **Error! Reference source not found.**. A solution for as function of k would have help, but this requires solving the equation:

Which is general and hard. But, if we are lucky, we can try and calculate and hope to get just the expression above, which by construction of will be 0.95, resulting in the proof that our CI is consistent. In order to do so, we need to express as function of , which is given by the binomial distribution:

Let , which happens with probability . Then:

# Sources

1. “Numerical Recipes, the Art of Scientific Computing”, Third Edition, <https://e-maxx.ru/bookz/files/numerical_recipes.pdf> . William H. Press, Saul A. Teukolsky, William T. Vetterling & Brian P. Flannery, Cambridge university press
2. Wikipedia linear regression, <https://en.wikipedia.org/wiki/Linear_regression>
3. “Statistics 512: Applied Linear Models; Chapter 5: Linear Regression in Matrix Form”, <https://www.stat.purdue.edu/~boli/stat512/lectures/topic3.pdf> , Bo Li, associate Professor, [Department of Statistics](http://www.stat.purdue.edu/), [Purdue University](http://www.purdue.edu/)
4. Benjamin T. Vincent . "Bayesian regression with truncated or censored data". In: *PyMC Examples*. Ed. by PyMC Team. DOI: [10.5281/zenodo.5654871](https://doi.org/10.5281/zenodo.5654871), <https://www.pymc.io/projects/examples/en/latest/generalized_linear_models/GLM-truncated-censored-regression.html>
5. <https://en.wikipedia.org/wiki/Confidence_interval>
6. <https://en.wikipedia.org/wiki/Bootstrapping_(statistics)>
7. <https://en.wikipedia.org/wiki/Bayesian_statistics>
8. <https://en.wikipedia.org/wiki/Binomial_distribution>
9. <https://en.wikipedia.org/wiki/Hypergeometric_function>
10. <https://en.wikipedia.org/wiki/Stirling%27s_approximation>
11. <https://www.youtube.com/watch?v=pHsuIaPbNbY&list=PLqdbxUnkqOw2nKn7VxYqIrKWcqRkQYOsF&index=12>
12. <https://en.wikipedia.org/wiki/Credible_interval>
13. <https://www.pymc.io/projects/docs/en/latest/installation.html>

1. In the python implementation, a 95.4% CL was given exactly as an input, and the number of standard deviation used to construct the CI was calculated to high precision, and so the number of standard deviation was corrected from 2.0 to 1.9954 [↑](#footnote-ref-1)
2. Notice although , it is incorrect to write , because we are not certain the . If our prior was than that was the case, but when we are not certain, all possibilities interefere. [↑](#footnote-ref-2)
3. Notice the question is phrased in a frequentist frame work, in the sense it does not acknowledge the prior. In this sense, we calculated the true CL of the constructed 95% **credible** interval. [↑](#footnote-ref-3)
4. HMC shines when there are many dimensions, and finding the area in the phase space with a lot of influence on the predictions () is a very hard problem. Also, for some problems, even after we find this area of the phase space, traveling through it to create samples can be tricky, and this is another advantage of the HMC algorithm. Non the less, in the regression problem clearly the phase space is very simple as the dimension is only 2. [↑](#footnote-ref-4)
5. one may construct such estimator from the residuals from the best fit , as their squared sum is distributed with and d.o.f., and so . [↑](#footnote-ref-5)