Bayesian Data Analysis

Bayesian Inference with applications to SPTs

Sabarno Saha

2025-03-31

IISERK

Outline

1. The Mighty Bayes theorem

The mighty Bayes Theorem

Definition 1.1 (Bayes Theorem)

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

Let us slightly modify the above equation to suit our needs.

Problem Description :
Suppose we get some data D and we want to infer the model parameters θ and the model M_i that generated the data.

Problem Description :
Suppose we get some data D and we want to infer the model parameters θ and the model M_i that generated the data.

Aloha! Bayes
We will try to use Bayes Theorem to infer the model parameters.[1]

Let M be the model and D be the data we obtained. Then Bayes Theorem can be written as

$$\overbrace{P(M|D)}^{\text{Posterior Distribution}} = \frac{\overbrace{P(D|M)}^{\text{Likelihood}}\overbrace{P(M)}^{\text{Prior}}}{\underbrace{P(D)}_{\text{Evidence}}}$$

- **Posterior Distribution** $[P(M|D)]$: The probability of the model given the data.
- **Likelihood** $[P(D|M)]$: The probability of the data given the model.
- **Prior** $[P(M)]$: The probability of the model.
- **Evidence** $[P(D)]$: The probability of the data.

Let $M_i \in \mathbb{M}$ where $\mathbb{M}$ is the set of models considered for the analysis. Let $\boldsymbol{\theta}$ be a vector consisting of the parameters of the model. The *practical* Bayes Theorem as

$$P(\boldsymbol{\theta}|\boldsymbol{D}, M_i) = \frac{P(\boldsymbol{D}|\boldsymbol{\theta}, M_i)P(\boldsymbol{\theta}|M_i)}{P(\boldsymbol{D}|M_i)}$$

- **Posterior Distribution** $[P(\theta|D, M_i)]$: The probability of the parameters given the data and the model.
- **Likelihood** $[P(D|\theta, M_i)]$: The probability of the data given the parameters and the model.
- **Prior** $[P(\theta|M_i)]$: The probability of the parameters given the model.
- **Evidence** $[P(D|M_i)]$: The probability of the data given the model.

2. Toy Example 1 : Coin Toss

Problem:
Suppose we have a coin and we want to infer the probability of getting a head, say p. Let us take n, say $n = 20$ tosses out of which r are heads. We want to infer the value of p given the data

$$\boldsymbol{\theta} = \{p\}$$

Here r is our data and p is the parameter we want to infer.

Now we know that the random variable representing one coin toss $X_c \sim \text{Bernoulli } (p)$. Then after getting r heads in n tosses, the likelihood function is a Binomial distribution. The likelihood function is given as

$$P(r|\boldsymbol{\theta}, n) = P(r|p, n) = \binom{n}{r} p^r (1-p)^{n-r}$$

The Beta Prior is a conjugate prior for the Binomial likelihood function. The posterior distribution for the Uniform Prior is given as

We will use two priors here :

- **Uniform Prior** :

$$P(p) = 1$$

  for $0 \leq p \leq 1$

- **Beta Prior** :

$$P(p) = \beta(p|a,b) = \left(\frac{\Gamma(a+b)}{\Gamma(a)\Gamma(b)}\right) p^{a-1}(1-p)^{b-1}$$

  for $0 \leq p \leq 1$

The Uniform Prior is used when we have no prior information about the parameter.

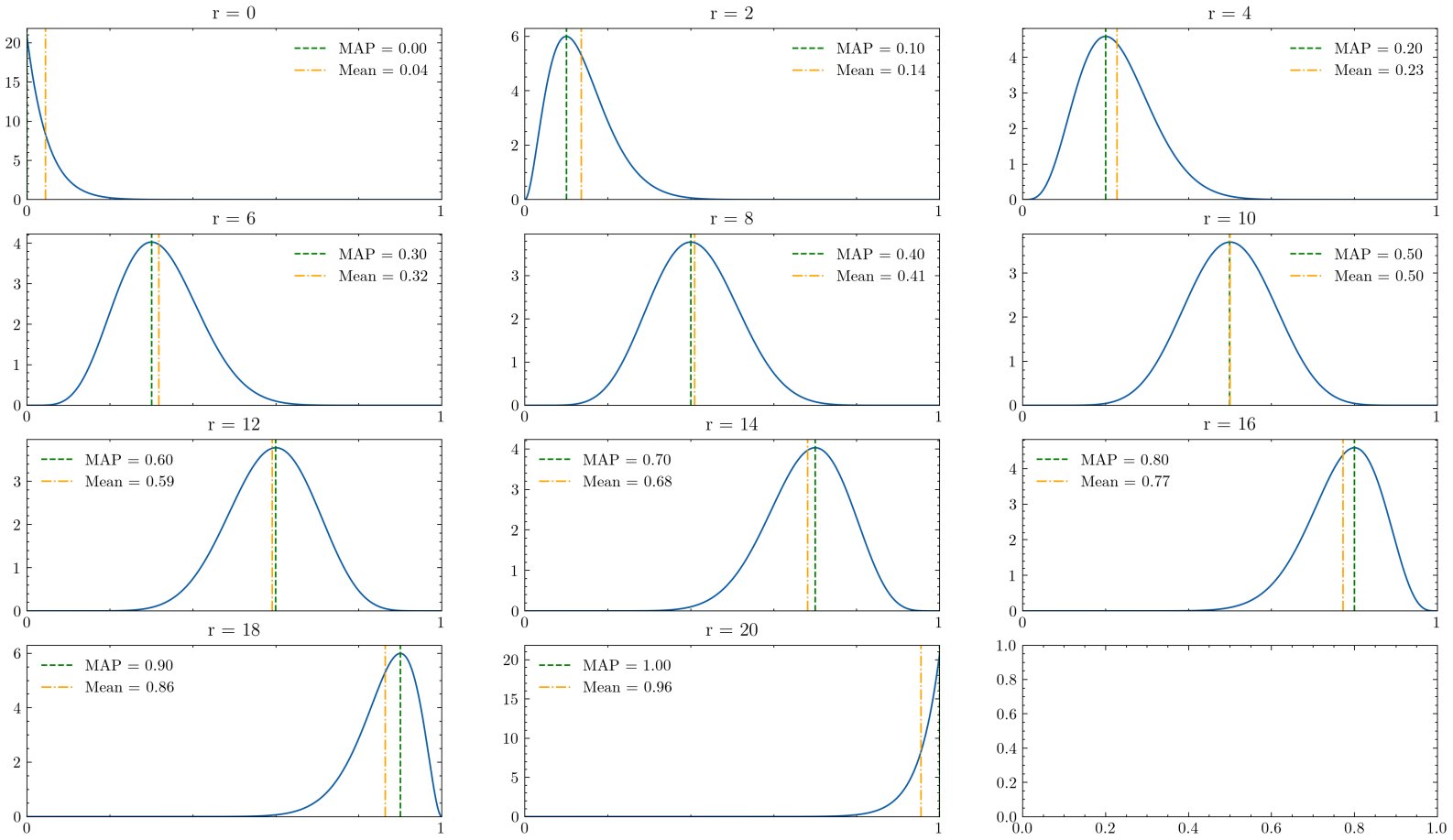
The Beta Prior is used when we have some prior information about the parameter. We assume that the coin is somewhat fair and choose $a = b = \frac{n}{2} = 10$.

Using bayes theorem, we can calculate the posterior distribution for the Uniform Prior to be just proportional to the likelihood function. The posterior distribution is given as

$$P(p|r,n) = \frac{1}{Z}\binom{n}{r}p^r(1-p)^{n-r}$$

where Z is the evidence or the normalizing constant.

2.2 Uniform Prior for 20 coin tosses.



The current beta prior we assume is when we expect the coin to be fair. So we take the beta prior to be

$$P(p) = \beta(p|10, 10) \propto p^9(1-p)^9$$

Note that it is harder to budge the beta prior than the uniform prior, because we might have prior information that the coin is fair. So if we get skewed data the posterior distribution still has a preconceived notion that the coin is fair.

We will see this when we compare the two priors.

Prior :

$$P(p) = \beta(p|a,b) = \frac{1}{B(a,b)} p^a (1-p)^b$$

where $0 \leq p \leq 1$

Prior :

$$P(p) = \beta(p|a,b) = \frac{1}{B(a,b)} p^a (1-p)^b$$

where $0 \leq p \leq 1$

Likelihood :

$$P(r|p,n) = \binom{n}{r} p^r (1-p)^{n-r}$$

Prior :

$$P(p) = \beta(p|a,b) = \frac{1}{B(a,b)} p^a (1-p)^b$$

where $0 \leq p \leq 1$

Likelihood :

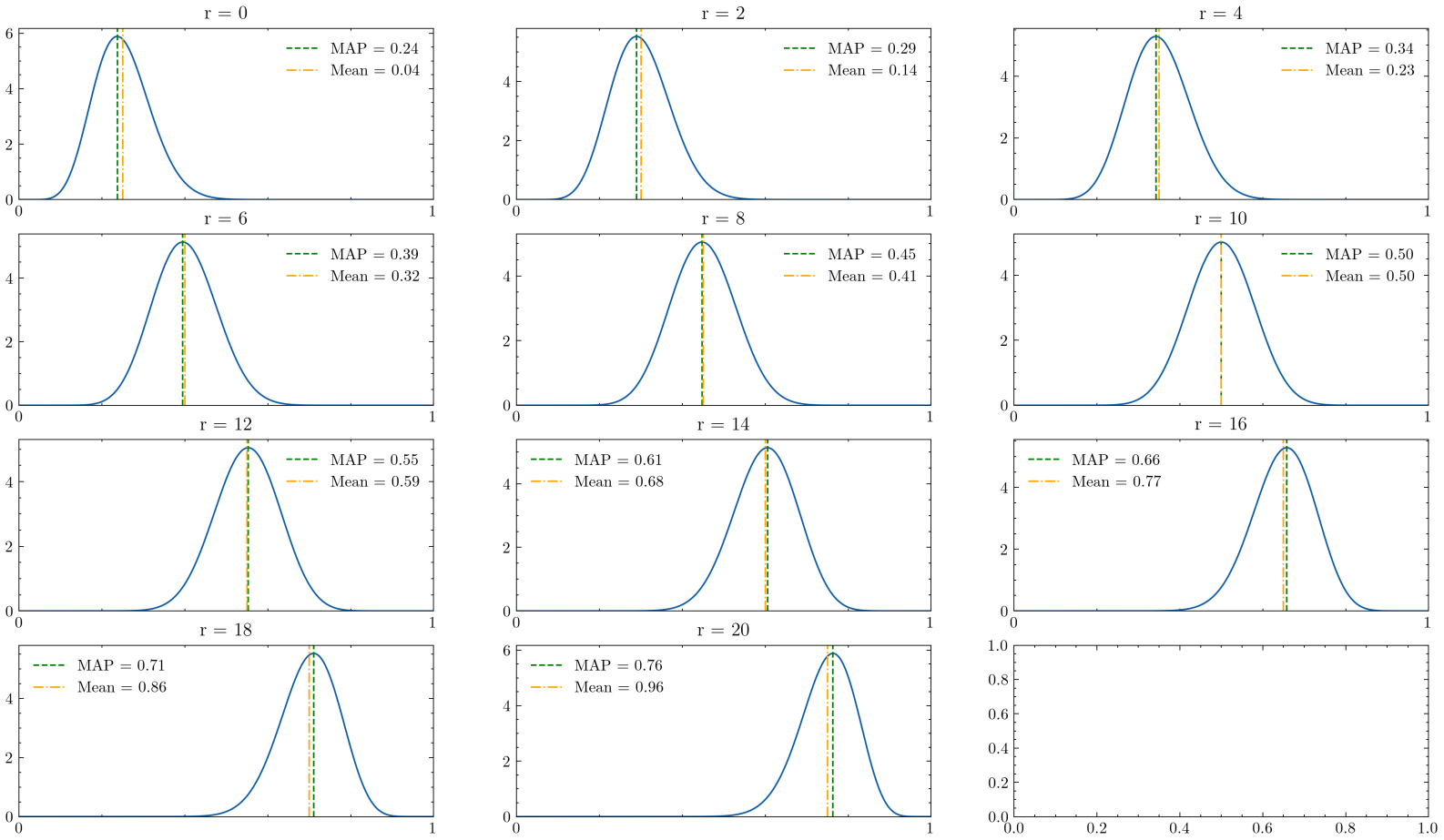
$$P(r|p,n) = \binom{n}{r} p^r (1-p)^{n-r}$$

Posterior :

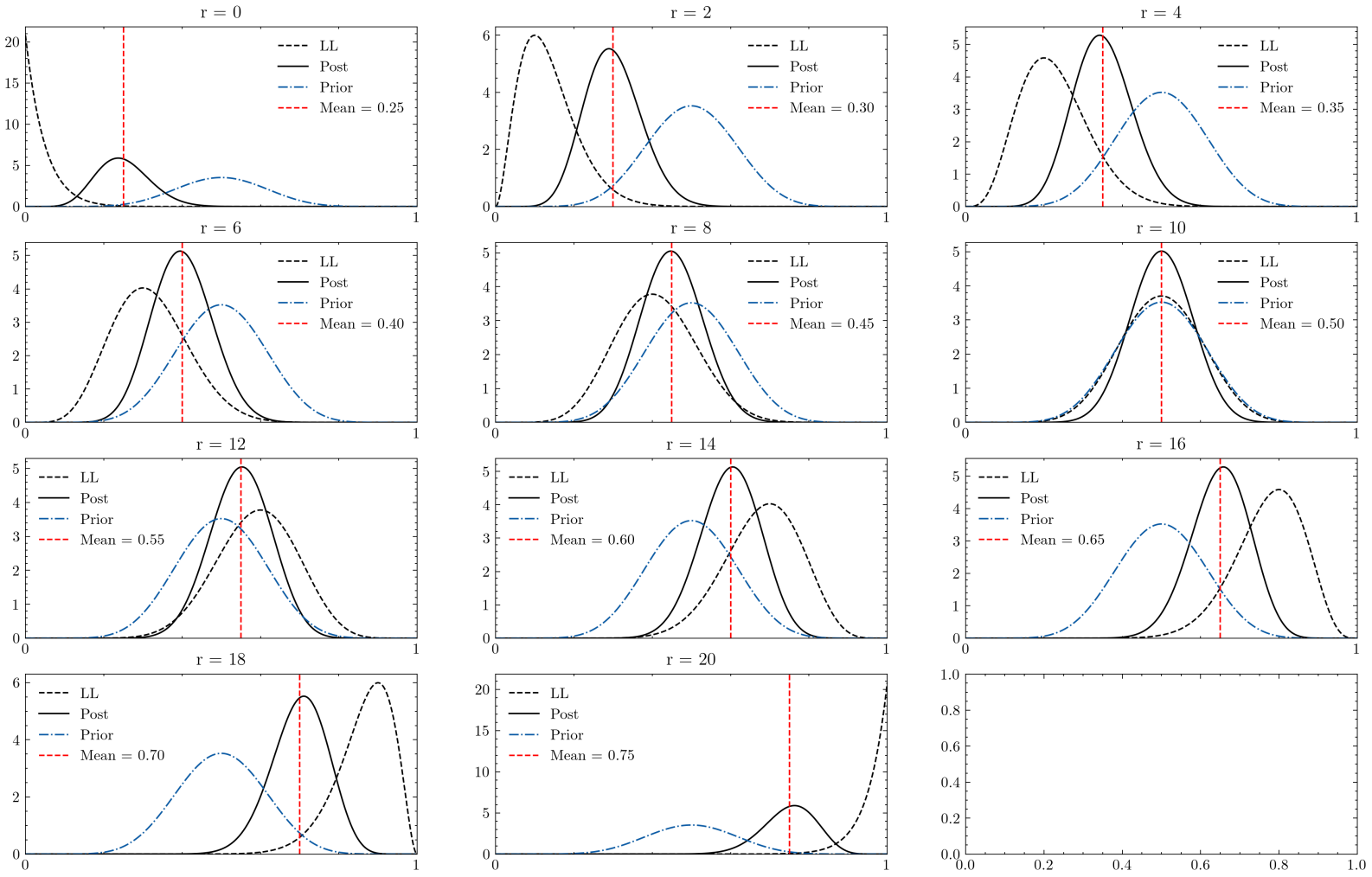
$$P(p|r,n) = \frac{\beta(p|a+r, b+n-r)}{Z} = \frac{1}{Z} p^{a+r-1} (1-p)^{b+n-r-1}$$

absorbing the terms not containing p in Z.

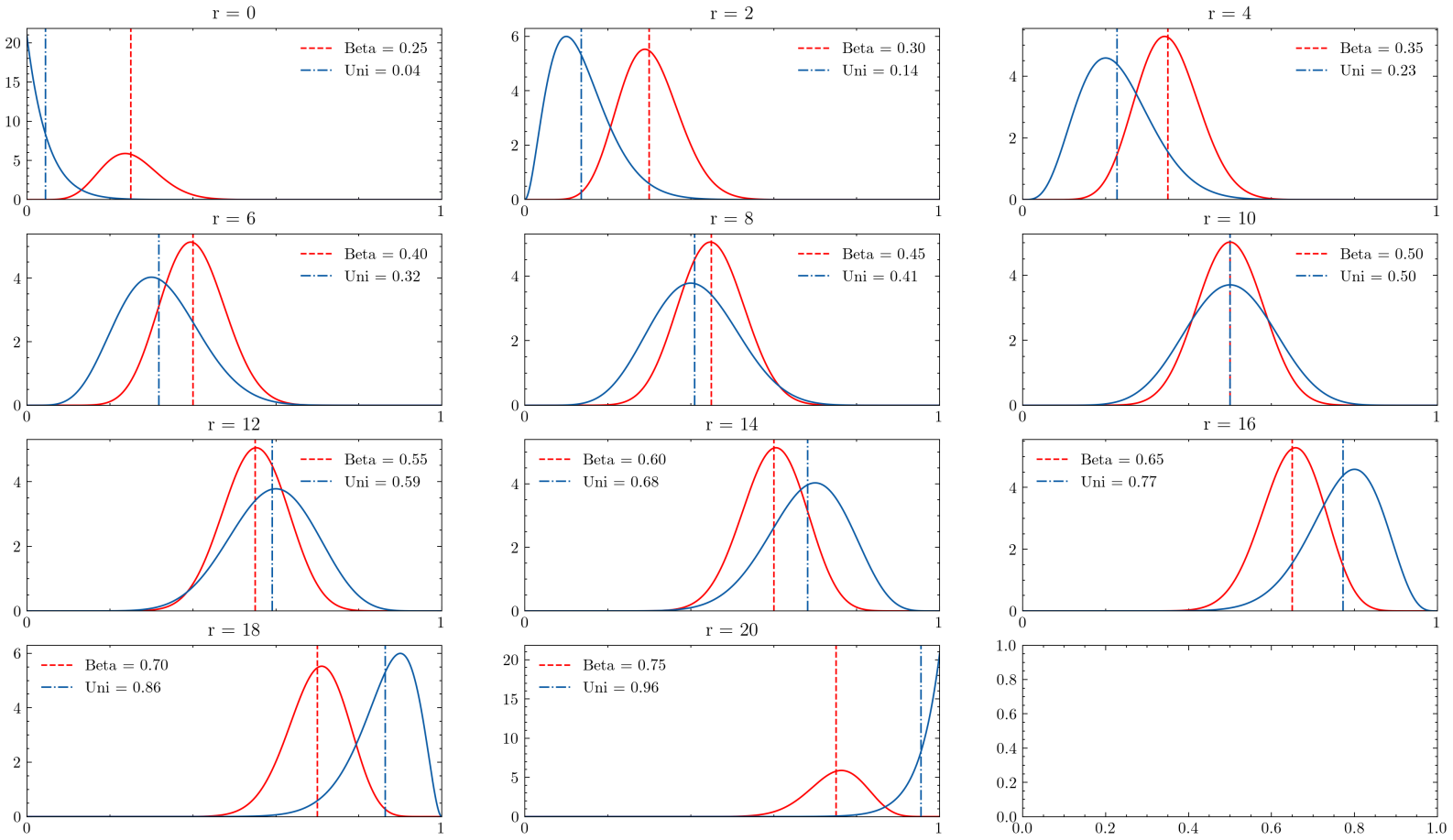
2.4 Beta Prior for 20 coin tosses.



2.5 Priors, LL, and Posterior for Beta prior

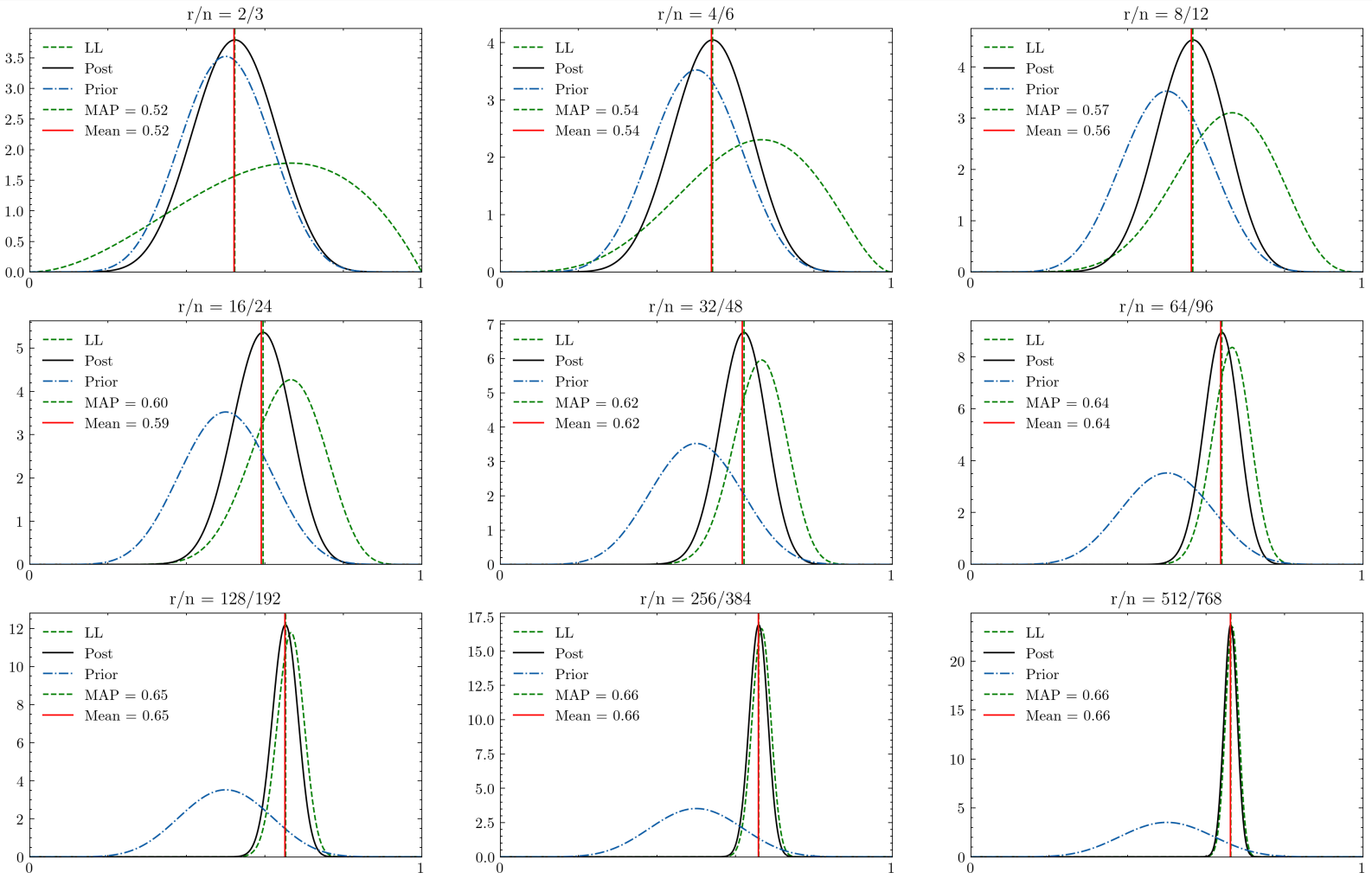


2.6 Comparison Between Uniform and Beta Priors



- **Bayesian** : We have a prior distribution for the parameter and we update it using the data to get the posterior distribution.
- **Frequentist** : We , theoretically, have an infinite number of data points and we use the data to estimate the parameter.

2.7 Bayesian to Frequentist thinking



3. Toy Example 2: Finding Evidence (Z)

Generally we are mostly instered in the posterior distribution which can then be normalized to find the normalization constant, which in this case is the evidence of the Model, Z. Integrating over small parameters spaces are possible to find the evidence of the model, but it is computationally expensive for higher dimensional parameter vector $\boldsymbol{\theta}$. We will see how to calculate the evidence of the model using an algorithm called Nested Sampling.

Originally proposed by John Skilling in 2004, Nested Sampling is a method to calculate the evidence of the model. The idea is to transform the multi-dimensional integral to a one-dimensional integral. The algorithm is as follows:

- Initialize the algorithm with a set of live points in the parameter space.
- Find the point with the lowest likelihood and replace it with a new point with a higher likelihood.
- The point with the lowest likelihood is called the “dead point”.
- The algorithm stops when the evidence is calculated to a desired accuracy.

Some math here ;)

The evidence $P(D|M) = Z(M)$ is given as

$$Z(M) = \int_{\Omega(\theta)} L(\boldsymbol{\theta})\pi(\boldsymbol{\theta})d\boldsymbol{\theta}$$

where $\Omega(\theta)$ is the prior volume and $L(\boldsymbol{\theta})$ is the likelihood function.

The evidence can also be interpreted as the normalization constant for the posterior distribution. Instead of computing overall parameter space to find the integral of the posterior. What we do is take the prior landscape deformed by the prior and try to define iso-likelihood contours and integrate over the contours to generate an integral in 2-D space which can be performed by any quadrature rule like rectangular, trapezoid, or Simpsons. So what we do now is define something called a prior mass namely $X(\lambda)$,

$$X(\lambda) = \int_{L(\boldsymbol{\theta})>\lambda} \pi(\boldsymbol{\theta}) d\boldsymbol{\theta}$$

Now we define $L(X(\lambda)) = \lambda$. Note that the integrand in evidence integration has a vector parameter while this has a scalar parameter. We can see that the lambda ranges from 0 to 1, as the prior mass decreases as lambda goes from 0 to 1, as likelihood is a probability and thus it is less than 1.

$$Z(M) = \int_0^1 L(X) dX$$

This can be transformed into a summation. Let us define the lowest likelihood at the j^{th} iteration, suppose which belongs to the p^{th} walker and we denote the lowest likelihood at the j^{th} we have $L(X_p) = L_j$ and $w_j = \Delta X_j = X_j - X_{j-1}$ so that we have

$$Z(M) = \sum_{i=1}^{N} L_j w_j \longrightarrow \int_0^1 L(X) dX$$

The posterior distribution can be given as

$$P_j = \frac{L_j w_j}{Z}$$

The implementation follows from Thapa et. al. [2]. We start by uniformly sampling N points which are our walkers, $\theta_1, ..., \theta_N$ from the prior surface in $(N+1)$ dimensions. We start by calulating the likelihoods of the walkers $L(\theta_i) = L_i$. Now we denote the lowest likelihood as $L_j = L(\theta_p)$ in the jth iteration, i.e. the likelihood of the pth walker. Now we store L_j and then assign the weights for our evidence calculation, $w_{j=1} = \frac{1}{N+1}$ for the 1st iteration and $w_{j\neq 1} = w_{j-1}\frac{K}{K+1}$. Now we add the $L_j w_j$ to the evidence Z according. So at the jth iteration we have the evidence to be

$$Z = \sum_{i=1}^{j} L_j w_j$$

Now we reject the point with the lowest likelihood i.e., we reject the pth point. We then sample another point at the $(j+1)^{th}$ iteration, using an MCMC algorithm constrained to $L(\theta_{new}) > L_j$, that is the new point point must have likelihood greater than the lowest

likelihood of the previous iteration. Now we come to the the stopping criteria, that is when we stop sampling of new points and return the complete evidence. Let us denote

$$Z_i^{rem} = w_j \sum_{i=1}^{N} L(\boldsymbol{\theta}_{j.m})$$

where $\boldsymbol{\theta}_{j,m}$ denotes the m^{th} walker in the j^{th} iteration. We now define the stopping ratio R_j at the jth iteration and we define it to be,

$$R_j = \frac{Z_j^{rem}}{Z}$$

We stop the nested sampling when $R_j < \varepsilon(\sim 10^{-4})$ where ε is the tolerance. We denote the last iteration to be $j = j_{\max}$. The final value of our Evidence Z is given as

$$Z = Z_{j_{\max}-1} + Z_{j_{\max}}^{rem}$$

We calculate $\ln Z$ instead of Z to prevent floating point errors. We also calulate log likelihood and log weight functions. So our calculation comes out to be,

$$\ln(Z_j) = \ln\left[\exp\left(\ln(Z_{j-1})\right) + \exp\left(\ln(L_j) + \ln(w_j)\right)\right]$$

In machine learning algorithms, an approximation is used called the logsumexp approximation

$$\begin{aligned}\log(\exp(a) + \exp(b)) &= \begin{cases} a + \log(1 + \exp(b - a)) & a \geq b \\ b + \log(1 + \exp(a - b)) & a < b \end{cases} \\ &= \max\{a, b\} + \ln(1 + \exp(-|a - b|))\end{aligned}$$

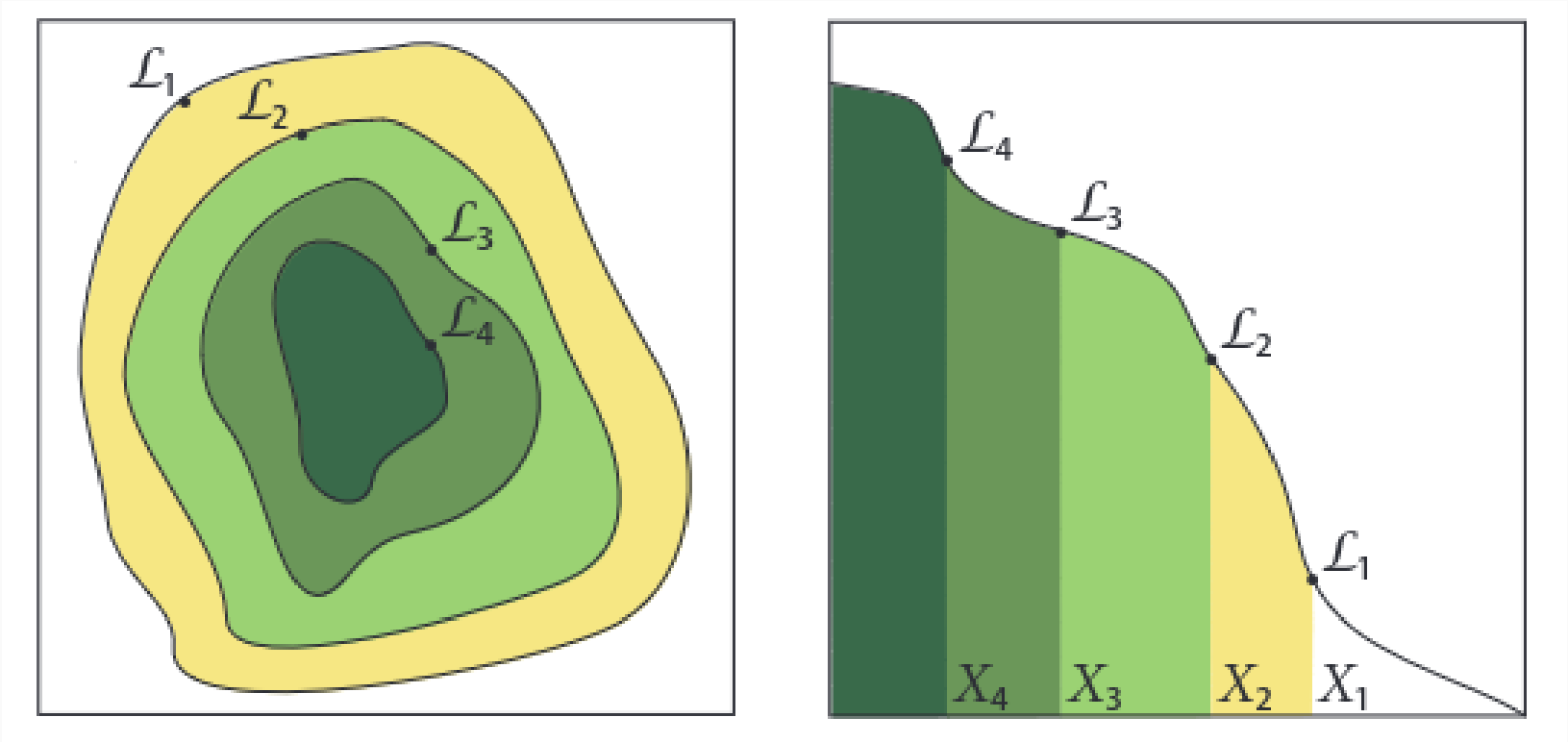
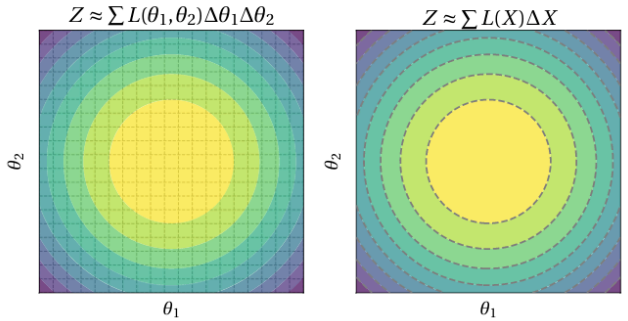
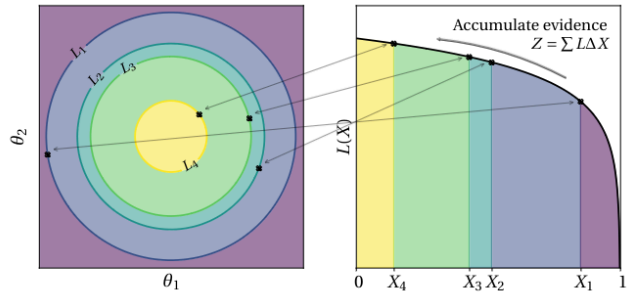


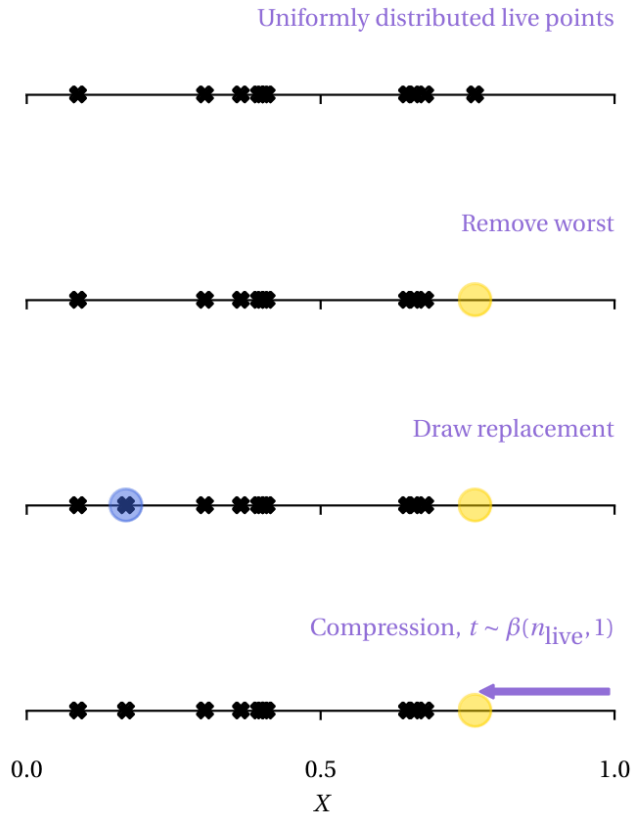
Figure 1: NS algorithm by Feroz et. al.



a | **The NS evidence identity.** The colours represent contours of a two-dimensional likelihood function. Rather than summing over little cubes (left), we combine cubes of similar likelihood together and sum over them (right).



b | **NS on a two dimensional problem.** We show the dead points and their iso-likelihood contours (left) and the corresponding contributions to the evidence integral (right). The volumes X_i are estimated statistically in NS.



c | **Compression in one iterate of NS.**

Figure 1 | **Illustrations of NS algorithm.**

Figure 2: NS algorithm illustration

Problem:
Suppose we take a particle and let the particle exhibit random motion. We want to infer the parameters of the model that generated the data. The model that we will consider is the Kramer's Equation generalized to a high friction limit.

The Langevin Equation is given by [3]

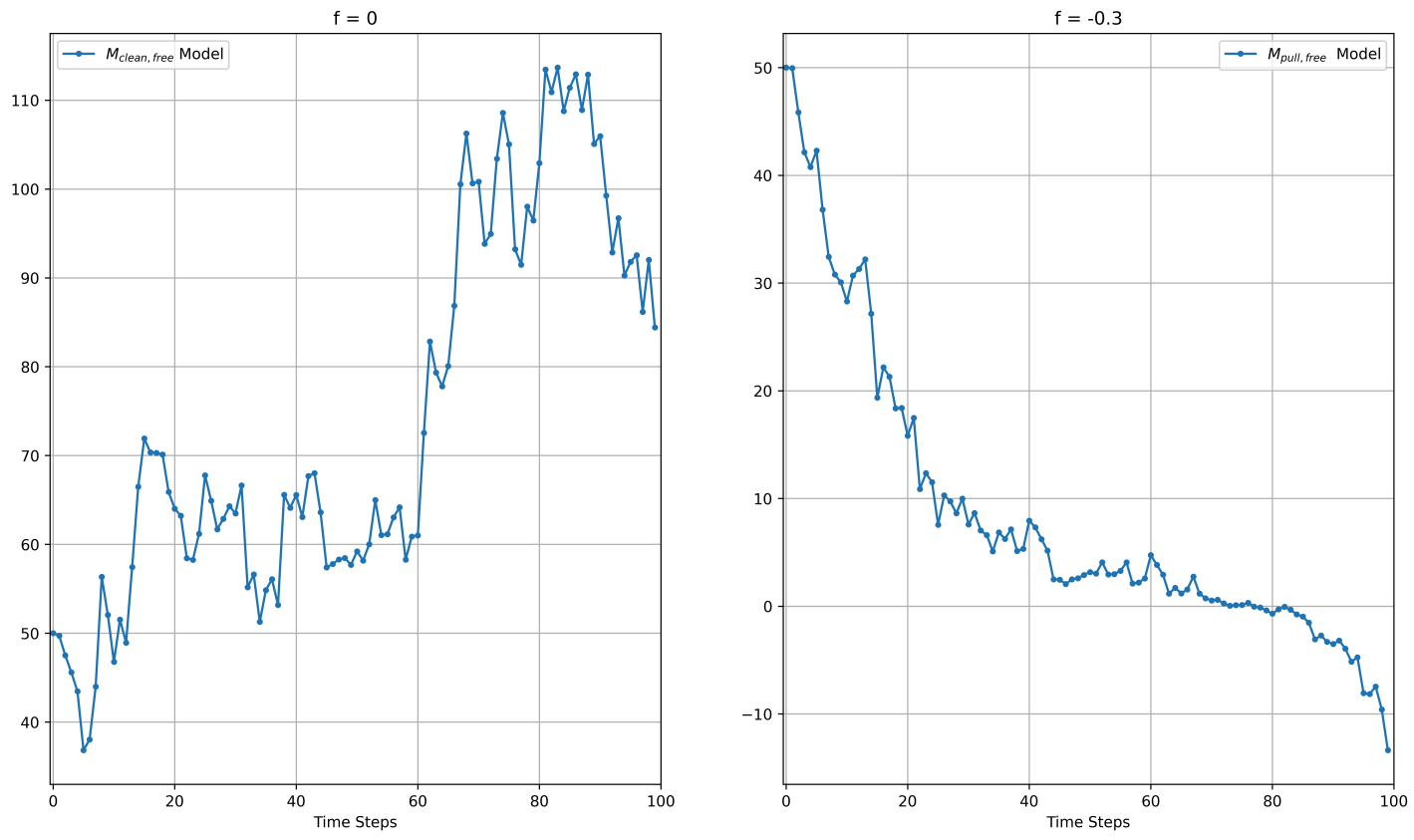
$$\frac{dx}{dt} = \mu(x)f + k_B T \partial_x \mu(x) + \sqrt{2k_B T \mu(x)}\hat{\xi}(t)$$

The test model uses the Fokker-Planck Equation which is

$$\partial_t P(x,t) = -\partial_x [\mu(x) f P(x,t) - k_B T \mu(x) \partial_x P(x,t)]$$

where $P(x,t)$ is the probability distribution at time t. $\mu(x)$ is the function that describes motility of the particle and f is the applied force on the particle. The paper [4] does not assume f as a function of coordinates but assumes it to be a constant.

Here is a sample track,



Let the track the particle follows be denoted by $\boldsymbol{x}$ which is a vector of data points. Let the data points $\boldsymbol{x}$ be $x_0, x_1, ..., x_N$. Note the random part of the equation is We can then get the likelihood function to be

$$\begin{aligned} P(\boldsymbol{x}|\boldsymbol{\theta}, M_i) &= P((x_0, x_1, ..., x_N)|\boldsymbol{\theta}, M_i) \\ &= P(x_N|\ x_{N-1}, ..., x_0, \boldsymbol{\theta}, M_i) P(x_{N-1}, ..., x_0, \boldsymbol{\theta}, M_i) \end{aligned}$$

Simplifying and continuing further we get(we also fix the first step at x_0)

$$
\begin{aligned}
P(\boldsymbol{x}|\boldsymbol{\theta}, M_i) &= P(x_0|\boldsymbol{\theta}, M_i) \prod_{j=1}^{N} P\big(x_j|x_{j-1}, ..., x_1, \boldsymbol{\theta}, M_i\big) \\
&= \prod_{j=1}^{N} P\big(x_j|x_{j-1}, \boldsymbol{\theta}, M_i\big) \\
&= \prod_{j=1}^{N} \frac{1}{\sqrt{2\pi\sigma_j^2}} \exp\left[-\frac{\big(x_j - \overline{x_j}\big)^2}{2\sigma_j^2}\right]
\end{aligned}
$$

where mean and variance is given as

$$
\begin{aligned}
\overline{x_j} &= x_{j-1} + \big[\mu\big(x_{j-1}\big) f + k_B T \partial_x \mu\big(x_{j-1}\big)\big] \Delta t \\
\sigma_j^2 &= 2 k_B T \mu\big(x_{j-1}\big) \Delta t
\end{aligned}
$$

We have obtained a generated track using the **pull,free** using the true parameters $D_0 = 0.2, \alpha = 1, f = -0.3$ and $\sigma^2_{mn} = 0$. All the inferred parameters are inferred using nested sampling on the

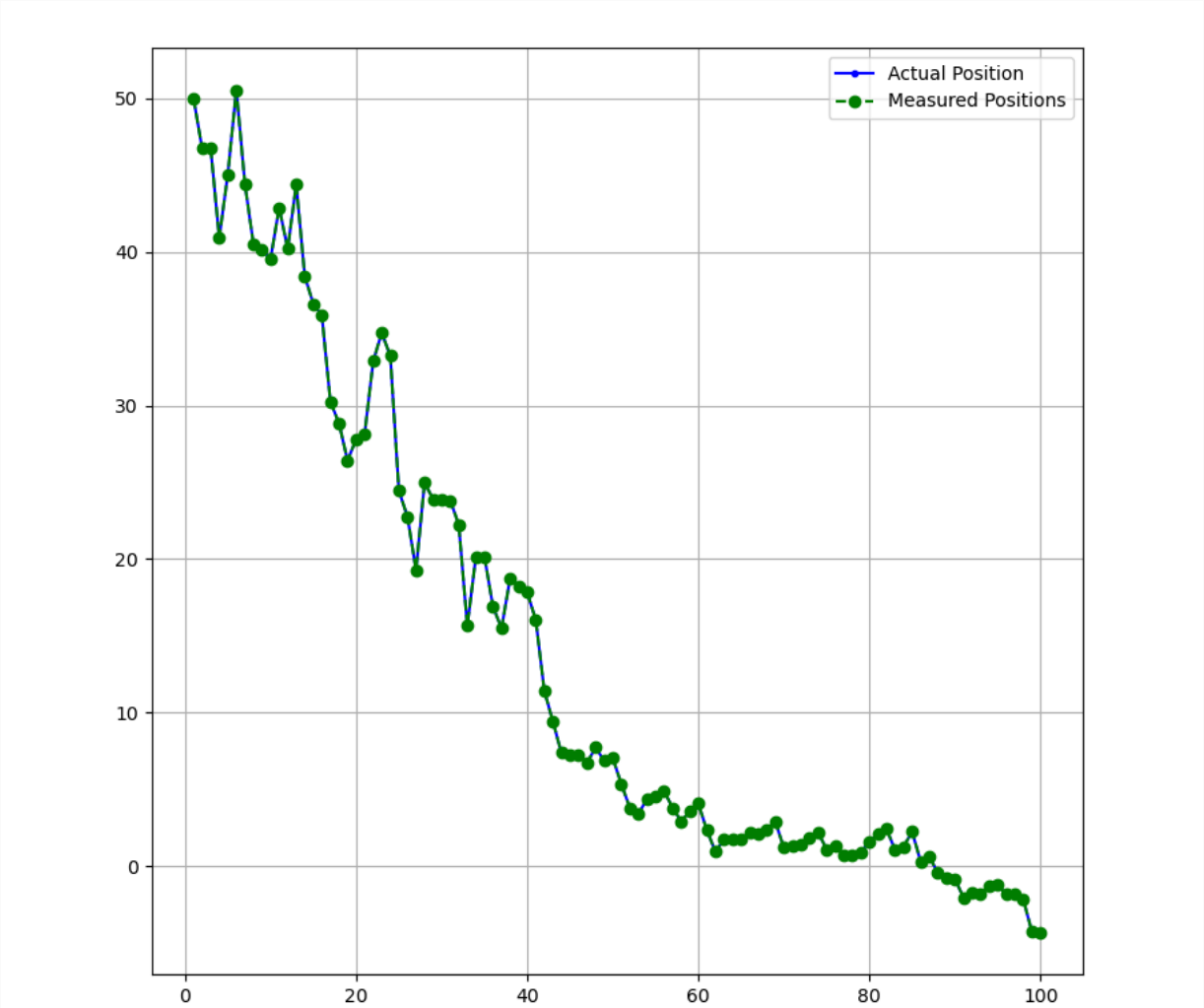
Inferred Parameters:

Names	**Inferred Value** $V \pm 3\sigma$	True Value
Diffusion Coefficient D_0	0.150 ± 0.1333	0.2
Alpha α	1.047 ± 0.3068	1
Applied Force f	-0.287 ± 0.2408	-0.3
Measurement noise σ^2_{mn}	0.050 ± 0.1562	0

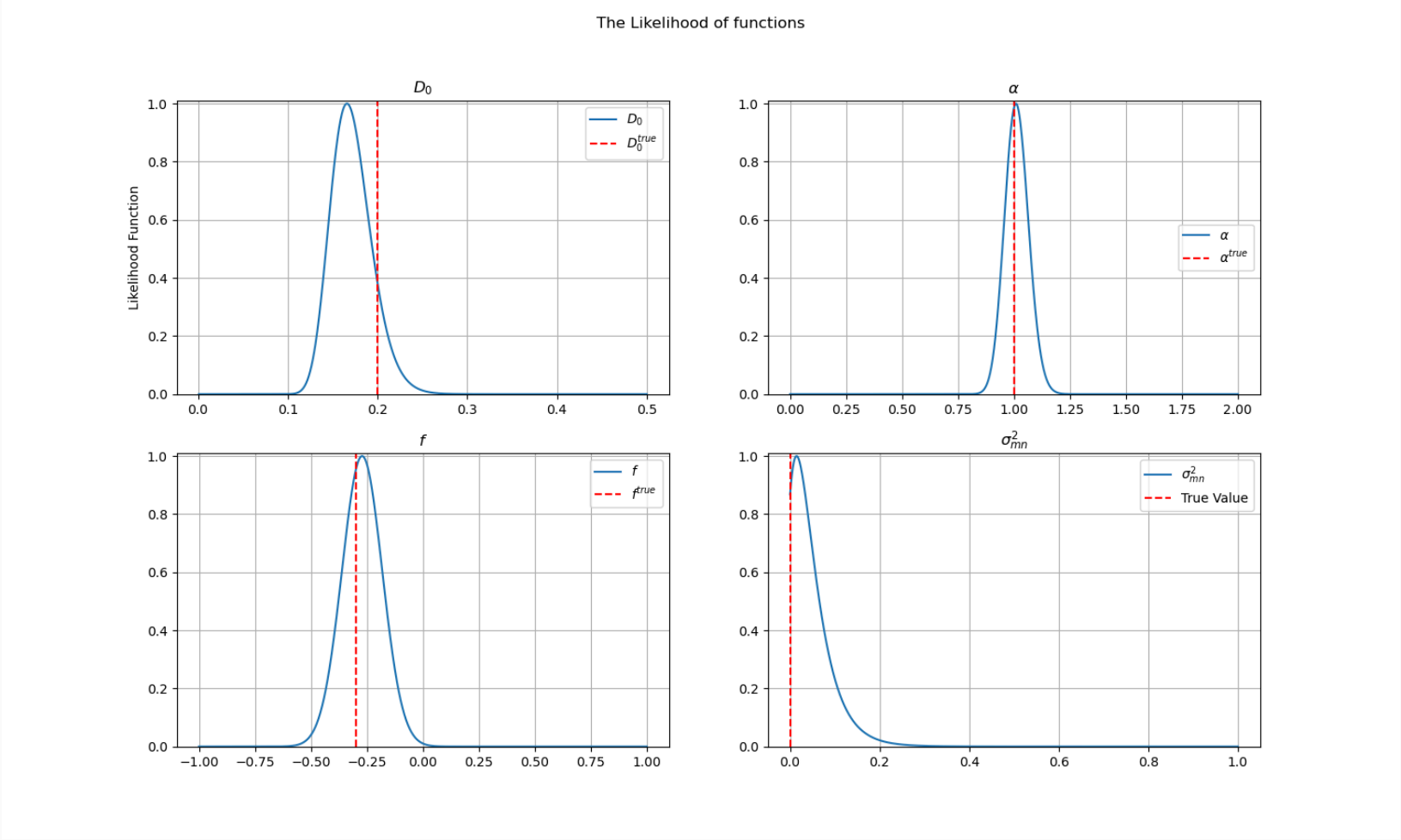
Table 1: Inferred Parameters and their errors (Model: pull,free)

The inferred parameters are given in the table above. We get the likelihood functions for each parameter. We also plot the track.

3.3 Example of Nested Sampling



3.3 Example of Nested Sampling



```
 Model 1  (free, clean)  with Model Prob = 0.009
--------------------------------------------------------------------
 ln Evidence = -185.454  +/- 0.12537
    Posterior Parameters
 1. Diffusion Coefficient D_0 = 0.167  +/-  0.0087
 2. Exponent alpha            = 1.034  +/-  0.0234
 3. Applied Force f           = 0.000  +/-  0.0000
 4. Var of Measurement noise  = 0.000  +/-  0.0000
 Maximum Likelihood = -194.276
 Information H = 1.57186
```

```
 Model 2  (pull, clean)  with Model Prob = 0.725
--------------------------------------------------------------------
 ln Evidence = -181.084  +/- 0.12760
    Posterior Parameters
 1. Diffusion Coefficient D_0 = 0.166  +/-  0.0073
 2. Exponent alpha            = 1.007  +/-  0.0212
 3. Applied Force f           = -0.269  +/-  0.0194
 4. Var of Measurement noise  = 0.000  +/-  0.0000
 Maximum Likelihood = -184.888
 Information H = 1.62827
```

```
 Model 3  (free, mn)  with Model Prob = 0.005
--------------------------------------------------------------------
 ln Evidence = -185.985  +/- 0.15212
    Posterior Parameters
 1. Diffusion Coefficient D_0 = 0.138  +/-  0.0109
 2. Exponent alpha            = 1.099  +/-  0.0345
 3. Applied Force f           = 0.000  +/-  0.0000
 4. Var of Measurement noise  = 0.028  +/-  0.0077
 Maximum Likelihood = -229.162
 Information H = 2.31412
```

```
 Model 4  (pull, mn)  with Model Prob = 0.260
--------------------------------------------------------------------
 ln Evidence = -182.109  +/- 0.16666
    Posterior Parameters
 1. Diffusion Coefficient D_0 = 0.151  +/-  0.0130
 2. Exponent alpha            = 1.025  +/-  0.0370
 3. Applied Force f           = -0.298  +/-  0.0358
 4. Var of Measurement noise  = 0.024  +/-  0.0109
 Maximum Likelihood = -249.447
 Information H = 2.77771

The Highest Model probability for the given data is for the Model (pull, clean)
```

4. References

Bibliography

[1] C. A. L. Bailer-Jones, *Practical Bayesian Inference: A Primer for Physical Scientists*. Cambridge: Cambridge University Press, 2017.

[2] S. Thapa, M. A. Lomholt, J. Krog, A. G. Cherstvy, and R. Metzler, "Bayesian analysis of single-particle tracking data using the nested-sampling algorithm: maximum-likelihood model selection applied to stochastic-diffusivity data," *Physical Chemistry Chemical Physics*, vol. 20, no. 46, pp. 29018–29037, 2018, doi: 10.1039/c8cp04043e.

[3] N. Shiraishi, *An Introduction to Stochastic Thermodynamics: From Basic to Advanced*. Springer Nature Singapore, 2023. doi: 10.1007/978-981-19-8186-9.

[4] J. Krog and M. A. Lomholt, "Bayesian inference with information content model check for Langevin equations," *Physical Review E*, vol. 96, no. 6, Dec. 2017, doi: 10.1103/physreve.96.062106.