NC STATE UNIVERSITY

The statistical underpinnings of maximum likelihood and Bayesian inference

Molecular Epidemiology of Infectious Diseases

Lecture 2

January 21st, 2026

A word on likelihoods

A likelihood is the probability of **data *X*** given some **model *M*** and its **parameter values *θ***

$$P(X|M,\theta)$$

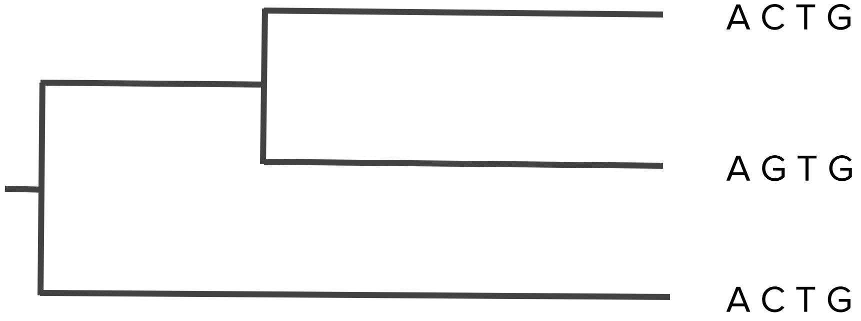
Likelihood based phylogenetic methods seek to find the tree that maximizes the likelihood of the sequence data under some model of molecular evolution

$$P(Seq|Tree,\theta)$$

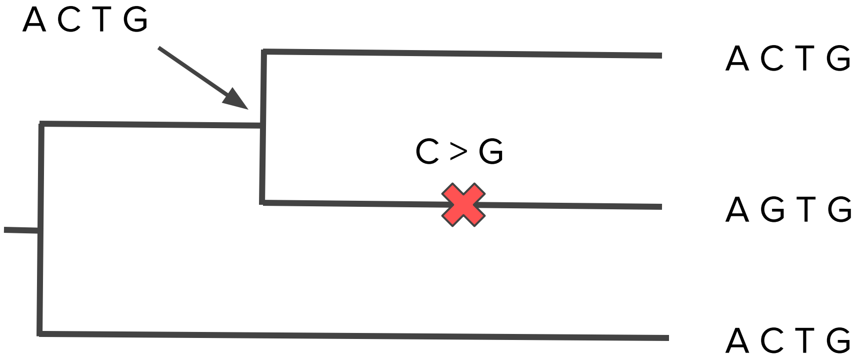
We therefore need to compute the likelihood of sequence data given a tree

Let’s start by assuming we have a phylogeny with aligned sequences at the tips

Likelihood of sequence data on trees

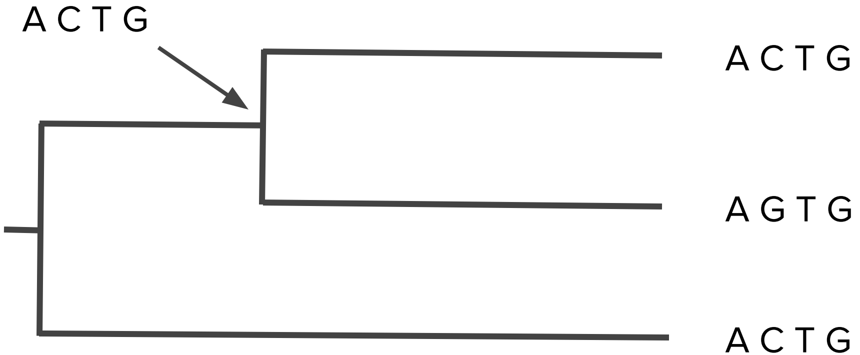


Likelihood of sequence data on trees



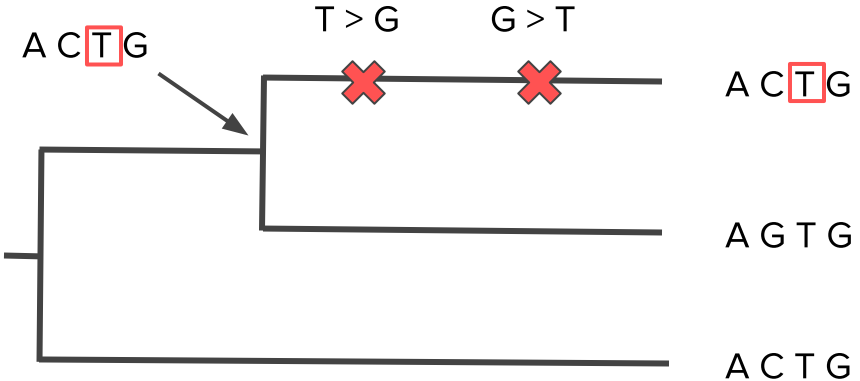
If we could directly observe sequence evolution on the tree, computing the likelihood of the sequence data would be easy. We could just compute the probability of every mutation event and multiply those probabilities together.

Likelihood of sequence data on trees



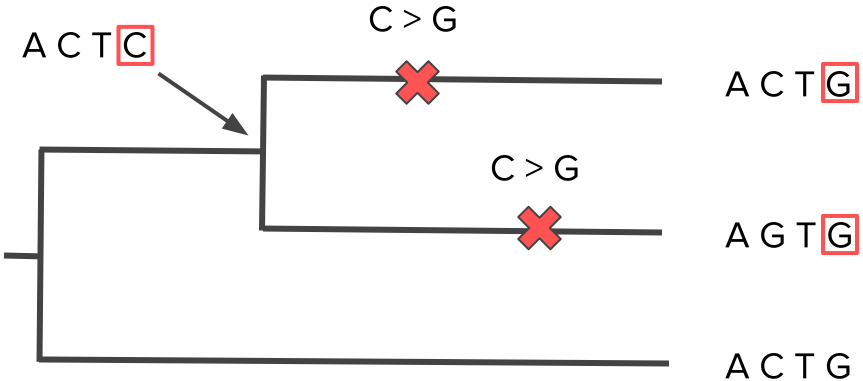
The problem is that we observe sequences at the tips but not their evolutionary history. Thus we have to take all possible evolutionary trajectories into account.

Likelihood of sequence data on trees



This includes the possibility of **multiple substitutions** occurring at a particular site.

Likelihood of sequence data on trees



And **convergent substitutions** occurring on different branches.

Modeling molecular evolution

We normally model sequence evolution as a **Markov process**.

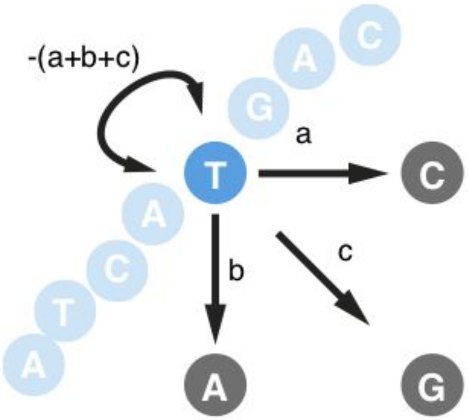
A Markov process is a type of **memoryless stochastic process**, i.e. a series of random events through time where the probability of jumping to a new state depends only the current state.

Example: the probability of a nucleotide base mutating to another base depends only on the current state, not previous states.

There are discrete and continuous time Markov processes. We generally model sequence evolution in continuous time.

Markovian models of sequence evolution

At a given site, the rate at which transitions between different bases occur is given by a **substitution rate matrix**:

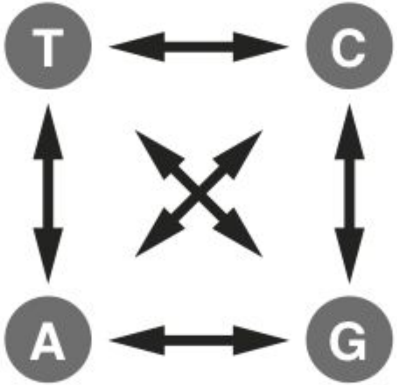


$$\begin{array}{c} \\ T \\ C \\ A \\ G \end{array}\begin{array}{c} \begin{array}{cccc} T & C & A & G \end{array} \\ \begin{pmatrix} -(a+b+c) & a & b & c \\ d & -(d+e+f) & e & f \\ g & h & -(g+h+i) & i \\ j & k & l & -(j+k+l) \end{pmatrix} \end{array}$$

Some common substitution models for DNA sequence evolution

The Jukes-Cantor model

The Jukes-Cantor model is the most basic substitution model for nucleotide sequences. All substitutions have the same rate **λ**:

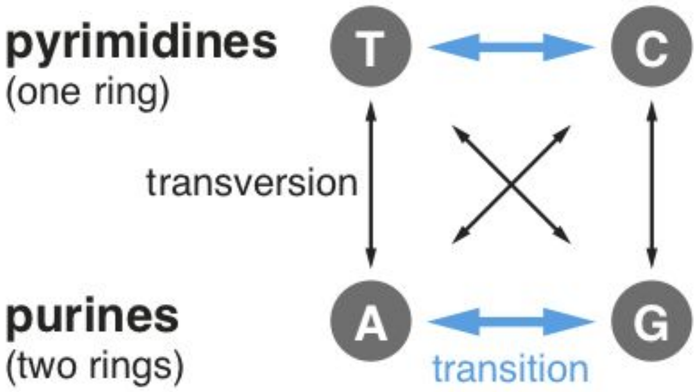


$$\begin{array}{c|cccc} & T & C & A & G \\ \hline T & \cdot & \lambda & \lambda & \lambda \\ C & \lambda & \cdot & \lambda & \lambda \\ A & \lambda & \lambda & \cdot & \lambda \\ G & \lambda & \lambda & \lambda & \cdot \end{array}$$

Jukes and Cantor (1969)

The K80 model

The K80 model allows for two substitution rates, one for **transitions (α)** and one for **transversions (β)**:

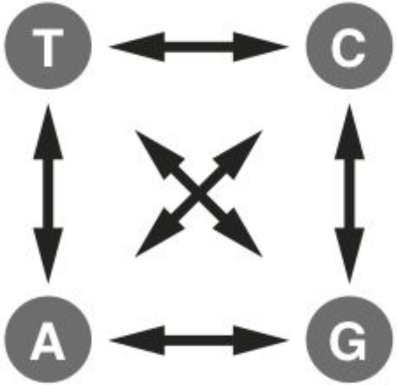


$$
\begin{array}{c|cccc}
 & T & C & A & G \\
\hline
T & \cdot & \alpha & \beta & \beta \\
C & \alpha & \cdot & \beta & \beta \\
A & \beta & \beta & \cdot & \alpha \\
G & \beta & \beta & \alpha & \cdot
\end{array}
$$

Kimura (1980)

The GTR model

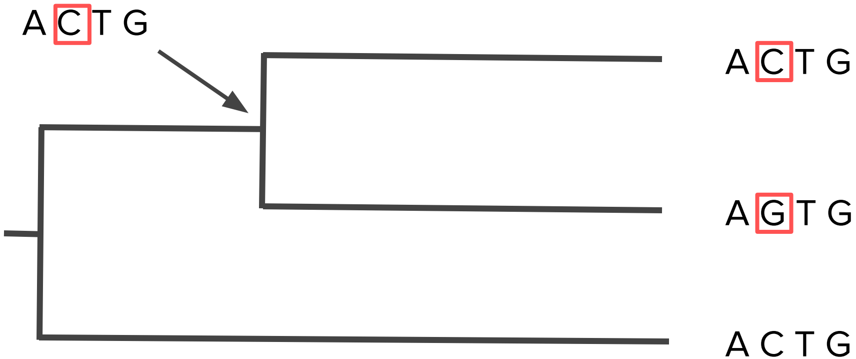
The generalized time reversible model (GTR) allows for six different substitution rates for each pair of nucleotides but assumes rates are symmetric.



$$\begin{array}{c|cccc} & T & C & A & G \\ T & \cdot & a\pi_C & b\pi_A & c\pi_G \\ C & a\pi_T & \cdot & d\pi_A & e\pi_G \\ A & b\pi_T & d\pi_C & \cdot & f\pi_G \\ G & c\pi_T & e\pi_C & f\pi_A & \cdot \end{array}$$

Yang (1994); Zharkikh (1994)

Likelihood of sequence data on trees



So far we have rates of nucleotide substitutions, but we need to find **transition probabilities** to compute the likelihood.

Modeling molecular evolution

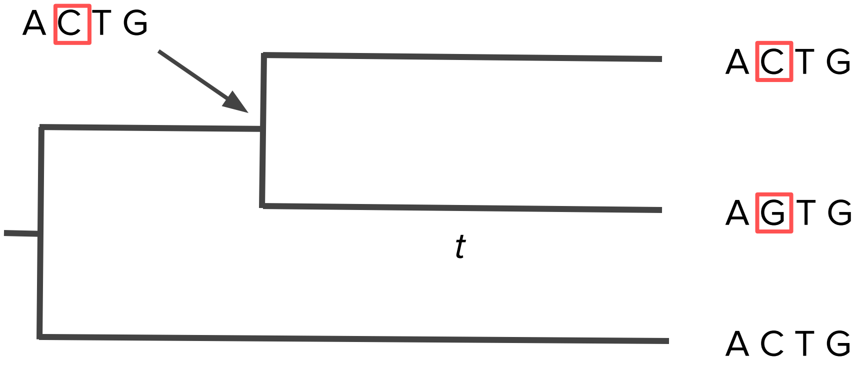
We can compute transition probabilities under a continuous-time Markov model given our substitution matrix ***Q*** and the time elapsed along a branch ***t***.

$$P(t) = e^{Qt}$$

The elements of *P(t)* give us the probability of every possible transition. Importantly, these **transition probabilities take into account every possible substitution path**.

$$P(t) = \begin{bmatrix} P_{T,T} & P_{T,C} & P_{T,A} & P_{T,G} \\ P_{C,T} & P_{C,C} & P_{C,A} & P_{C,G} \\ P_{A,T} & P_{A,C} & P_{A,A} & P_{A,G} \\ P_{G,T} & P_{G,C} & P_{G,A} & P_{G,G} \end{bmatrix}$$

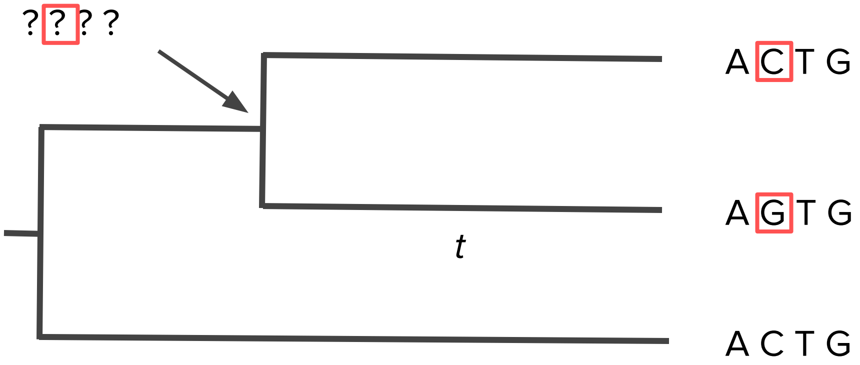
Computing likelihoods at one site



Given the ancestral sequence of the parent, we can compute the likelihood at a single site:

$$L(Seq|Tree) = P_{C,C}(t) * P_{C,G}(t)$$

Computing likelihoods at one site



If the ancestral sequences are not observed, we must integrate or sum over all possible ancestral states:

$$L(Seq|Tree) = \sum_{X \in \{A,C,T,G\}} \left(P_{X,C}(t) * P_{X,G}(t)\right)$$

Computing the total likelihood

Felsenstein’s pruning algorithm (J. Mol. Evol., 1981) uses dynamic programming to compute likelihoods on larger trees. The algorithm traverses the tree from tips to root, combining the partial likelihoods of two subtrees at each internal node.

We generally assume sites evolve independently, so we can multiple the likelihood of each site to compute the total likelihood of the sequence data at all sites.

$$L(Seq|Tree) = \prod_{i=1}^{i=N} L(Seq_i|Tree)$$

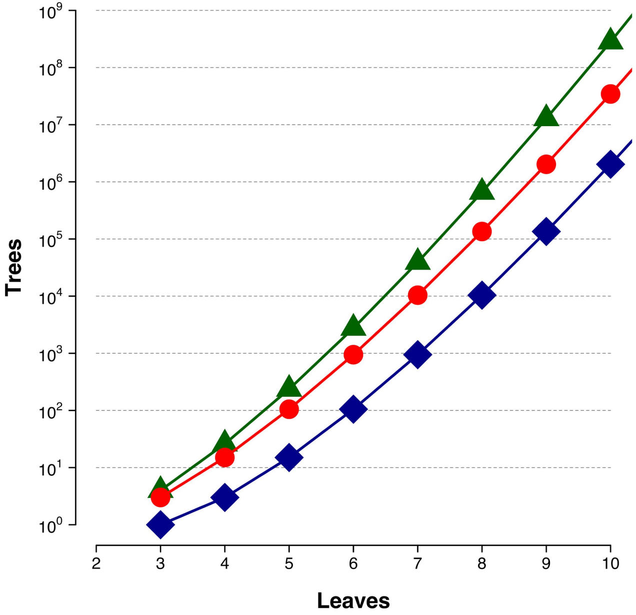
Maximum likelihood tree reconstruction

Likelihood-based tree reconstruction methods **search tree space** to find the tree that maximizes the likelihood of the sequence data.

The number of potential trees grows rapidly with the number of tips. There are *(2n-3)!!* rooted binary trees for *n* tips.

Most ML methods like RAxML employ a heuristic rather than exhaustive tree searches.

***Also need to estimate evolutionary parameters like substitution rates



Red shows rooted binary trees.

Towards a Bayesian worldview

Adopting a Bayesian worldview

Bayesian inference is really all about combining information in a rational way while dealing with uncertainty

Basic model: Prior beliefs → New data → Updated beliefs

The way we combine information follows directly from basic probability theory (i.e. Bayes theorem)

Bayesian reasoning: An example

Let’s say your doctor just diagnosed you with a very rare disease found in only one out of every 1,000 people (0.1% prevalence)

We know that the true positive rate of the diagnostic test is 95% and the false positive rate is 5%

What is the probability that you are actually sick?

Bayes theorem

Bayes theorem tells us how to correctly compute **conditional probabilities** of the form *P(A|B)*.

That is, what is the probability of observing outcome A given that we observed outcome B?

Bayes theorem tells us that:

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

Bayes theorem: an example

In our example, we want to compute the conditional probability *P(sick | +)*.

Applying Bayes theorem, we see that:

$$P(sick|+) = \frac{P(+|sick)}{P(+)}P(sick)$$

Bayes theorem: an example

We already know two pieces of information needed:

We know that the *prior probability P(sick)* is 1 in 1000 = 0.001

We know the true positive rate is: P(+ | sick) = 0.95.

Bayes theorem:

$$P(sick|+) = \frac{P(+|sick)}{P(+)} P(sick)$$

Bayes theorem: an example

But how do we compute the total probability of testing positive *P(+)*?

We need to sum all the ways we could have been diagnosed as positive whether healthy or sick. So the total probability of being positive is:

$$P(+) = P(+|sick)P(sick) + P(+|healthy)P(healthy) = 0.05$$

The true positive rate is 95%, so P(+ | sick) = 0.95.

The false positive rate is 5%, so *P(+ | healthy)* = 0.05.

P(sick) = 0.001

P(healthy) = 1 - P(sick) = 0.999.

Bayes theorem: an example

Putting everything back into Bayes theorem:

$$P(sick|+) = \frac{P(+|sick)}{P(+)}P(sick) = \frac{0.95}{0.05}0.001 = 0.0187$$

Bayes theorem: an example

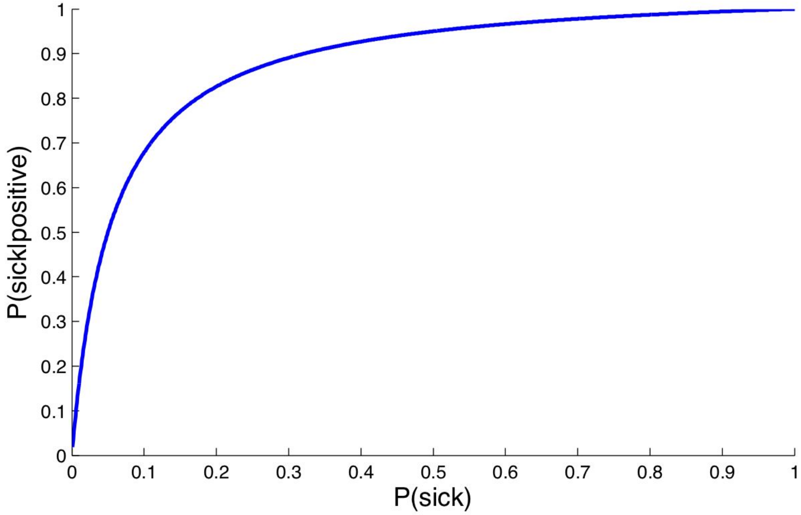
Putting everything back into Bayes theorem:

$$P(sick|+) = \frac{P(+|sick)}{P(+)}P(sick) = \frac{0.95}{0.05}0.001 = 0.0187$$

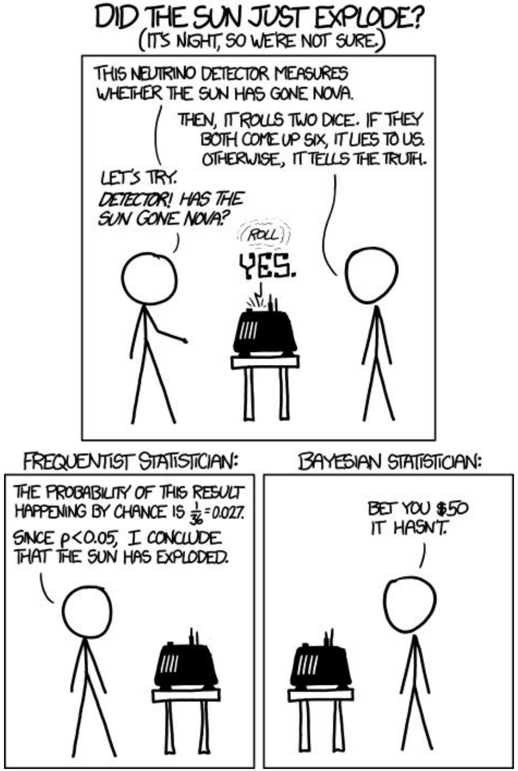
Interpretation: the relative low prior probability of being sick combined with the relatively high probability of a false positive means the actual probability of being sick is low (>2%).

Dependence on prior beliefs

The actual probability of being sick given a positive test depends very strongly on the background rate or prevalence *P(sick)*.



The problem with ignoring prior info


DID THE SUN JUST EXPLODE?
(IT'S NIGHT, SO WE'RE NOT SURE.)
THIS NEUTRINO DETECTOR MEASURES WHETHER THE SUN HAS GONE NOVA.
THEN, IT ROLLS TWO DICE. IF THEY BOTH COME UP SIX, IT LIES TO US. OTHERWISE, IT TELLS THE TRUTH.
LET'S TRY.
DETECTOR! HAS THE SUN GONE NOVA?
ROLL
YES.
FREQUENTIST STATISTICIAN:
THE PROBABILITY OF THIS RESULT HAPPENING BY CHANCE IS 1/36 = 0.027. SINCE p<0.05, I CONCLUDE THAT THE SUN HAS EXPLODED.
BAYESIAN STATISTICIAN:
BET YOU $50 IT HASN'T.

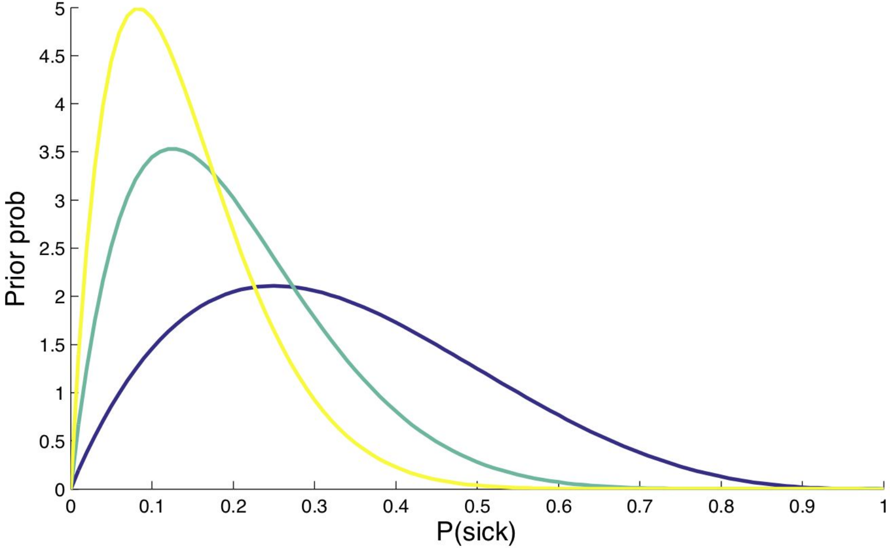
https://xkcd.com/1132/

But how do we quantify our prior beliefs?

Bayesian priors

In Bayesian inference, **we summarize our *prior* beliefs using a prior distribution**

The prior is a probability distribution over all possible values of an unknown (random) variable.



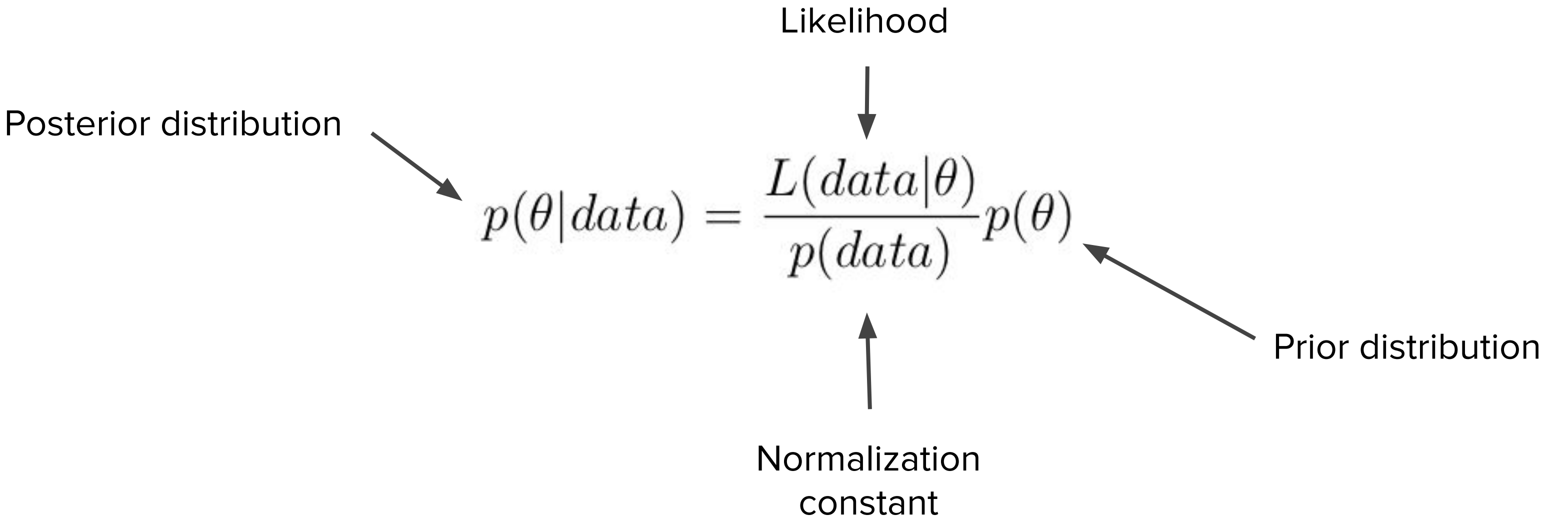
Bayesian inference

Our prior beliefs get updated when we observe new information or data using Bayes theorem:

$$p(\theta|data) = \frac{L(data|\theta)}{p(data)} p(\theta)$$

Bayesian inference

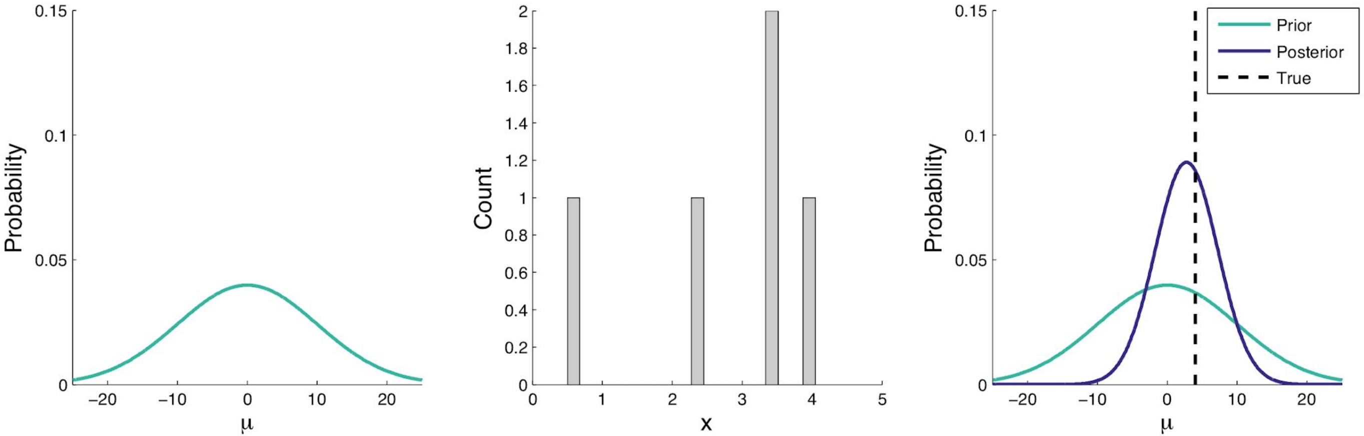
Our prior beliefs get updated when we observe new information or data using Bayes theorem:



$$p(\theta|data) = \frac{L(data|\theta)}{p(data)}p(\theta)$$

Bayesian inference: another example

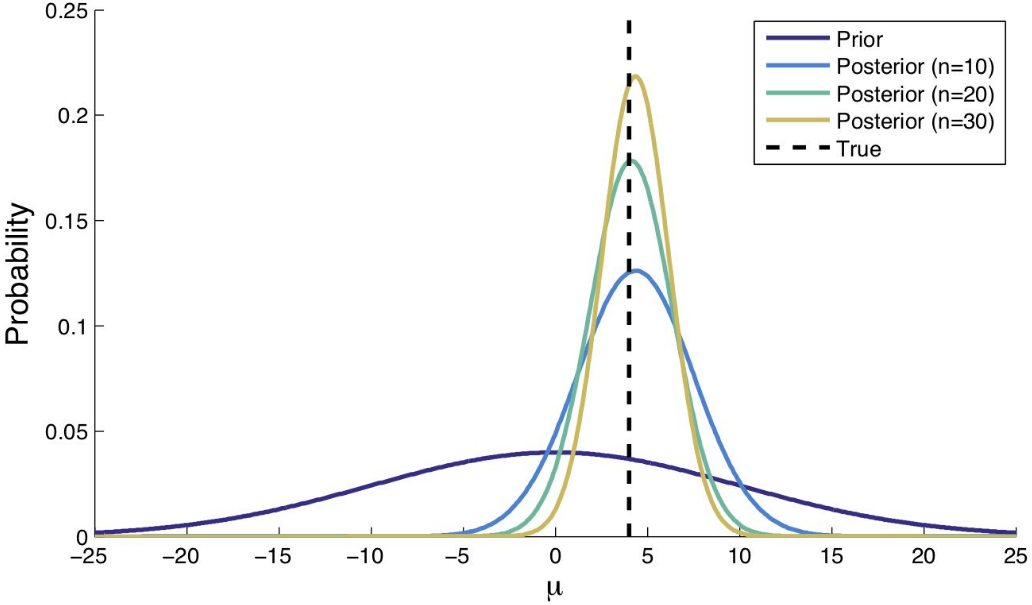
Inferring the mean value of a normally distributed population given a limited sample



The true mean μ = 4, so observing some data shifts the prior distribution away from zero towards the 4.

Bayesian inference: another example

The relative contribution of the prior to the posterior decreases with more data



Computing the posterior

We can generally compute the unnormalized posterior probability of a given parameter value:

$$p(\theta = x|data) \propto L(data|\theta = x)p(\theta = x)$$

However, it is generally very difficult to compute the normalization constant:

$$p(data) = \sum_{\theta} L(data|\theta)p(\theta)$$

$$p(data) = \int_{\theta} L(data|\theta)p(\theta)d\theta$$

Computing the posterior

If we cannot analytically compute the posterior, we can sample values from the posterior distribution and then use these samples to construct an approximation to the posterior distribution.

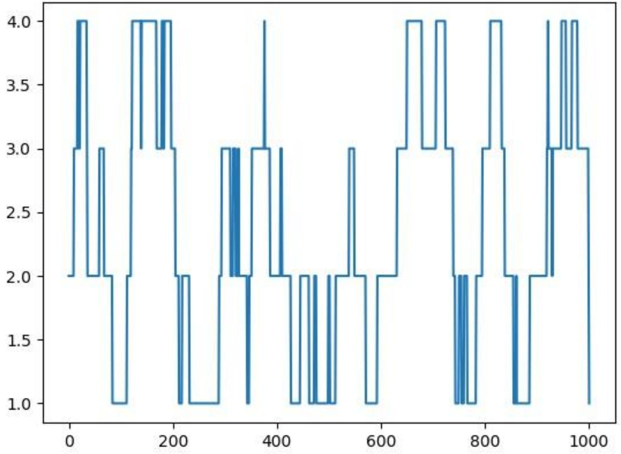
in Bayesian inference, **Markov chain Monte Carlo (MCMC)** is the most commonly used method to sample from a desired distribution.

What is a Markov chain?

A Markov chain is a Markov process that randomly jumps between different states over time. The state of the process at time t_n depends only on the previous state at time t_{n-1}.

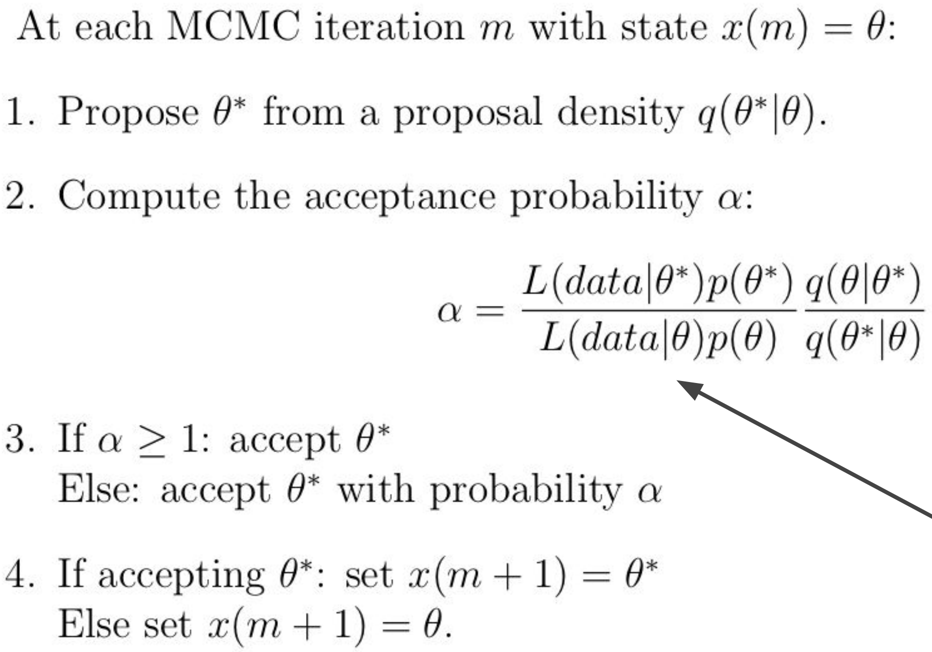
MCMC is an example of discrete-time process.

Example: a one dimensional random walk



The Metropolis-Hastings algorithm

The Metropolis-Hastings (MH) algorithm is one commonly used variant of MCMC:



At each MCMC iteration m with state $x(m) = \theta$:

1. Propose θ^* from a proposal density $q(\theta^*|\theta)$.
2. Compute the acceptance probability α:

$$\alpha = \frac{L(data|\theta^*)p(\theta^*)}{L(data|\theta)p(\theta)} \frac{q(\theta|\theta^*)}{q(\theta^*|\theta)}$$

3. If $\alpha \geq 1$: accept θ^*
   Else: accept θ^* with probability α
4. If accepting θ^*: set $x(m+1) = \theta^*$
   Else set $x(m+1) = \theta$.

Hastings term

Ratio of posterior probabilities

The Metropolis-Hastings algorithm

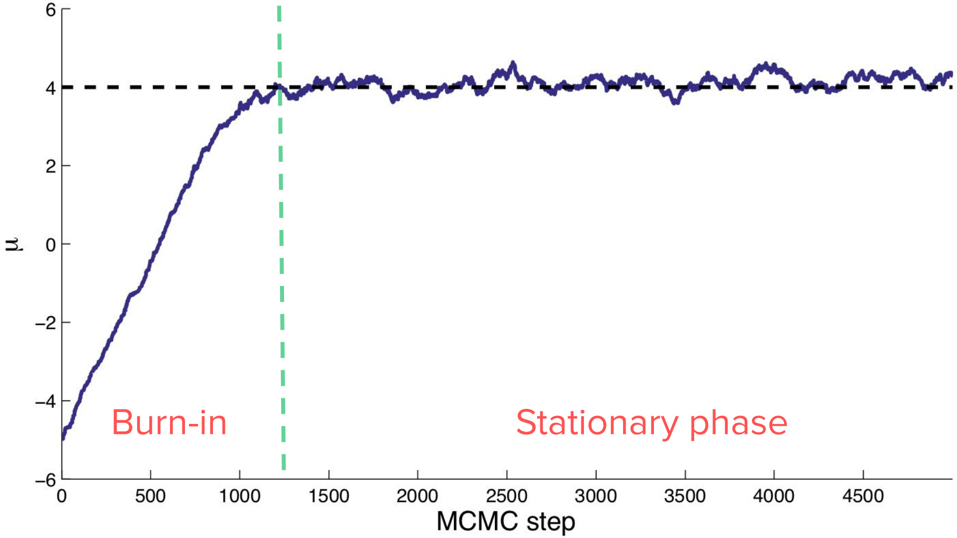
The **main idea behind the MH algorithm** is that we accept parameters with a probability proportional to their posterior probability.

This means that the amount of time the chain spends in state *x* will be proportional to the posterior probability of *x*.

However, for this to be true, the chain needs to have reached its **stationary phase or distribution** (i.e. equilibrium).

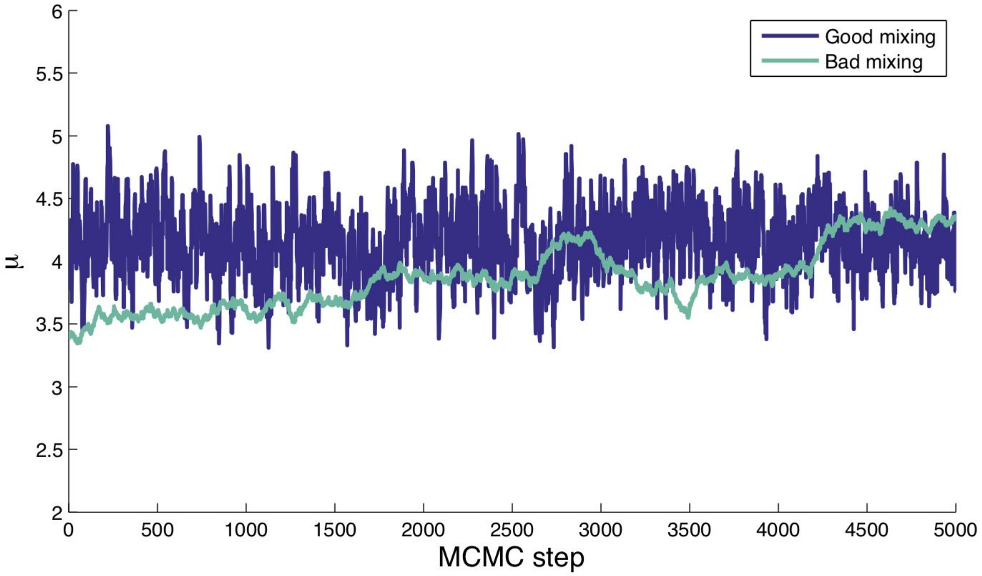
MCMC: Convergence

Samples from a MCMC are only valid once the chain has **converged** on its stationary distribution



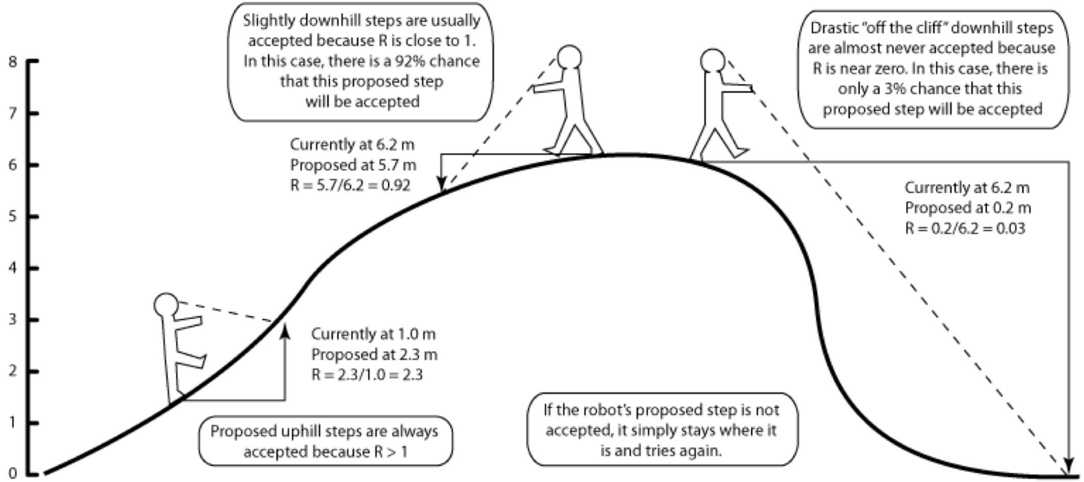
MCMC: Mixing

Mixing refers to how efficiently the chain explores the posterior distribution. Since we want pseudo-independent samples from the posterior, we want good mixing = low autocorrelation between successive samples.



MCMC: The blind robot analogy

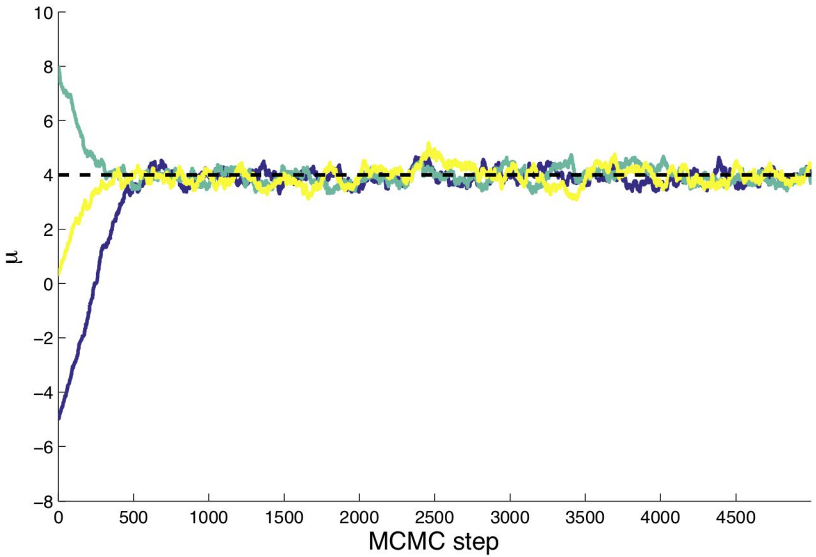
Achieving good mixing requires a good proposal distribution.



Slightly downhill steps are usually accepted because R is close to 1. In this case, there is a 92% chance that this proposed step will be accepted
Drastic "off the cliff" downhill steps are almost never accepted because R is near zero. In this case, there is only a 3% chance that this proposed step will be accepted
Currently at 6.2 m
Proposed at 5.7 m
R = 5.7/6.2 = 0.92
Currently at 6.2 m
Proposed at 0.2 m
R = 0.2/6.2 = 0.03
Currently at 1.0 m
Proposed at 2.3 m
R = 2.3/1.0 = 2.3
Proposed uphill steps are always accepted because R > 1
If the robot's proposed step is not accepted, it simply stays where it is and tries again.
8
7
6
5
4
3
2
1
0

MCMC: Checking convergence

Because of issues with mixing and convergence, it is always a good idea to run multiple chains starting from different initial values.



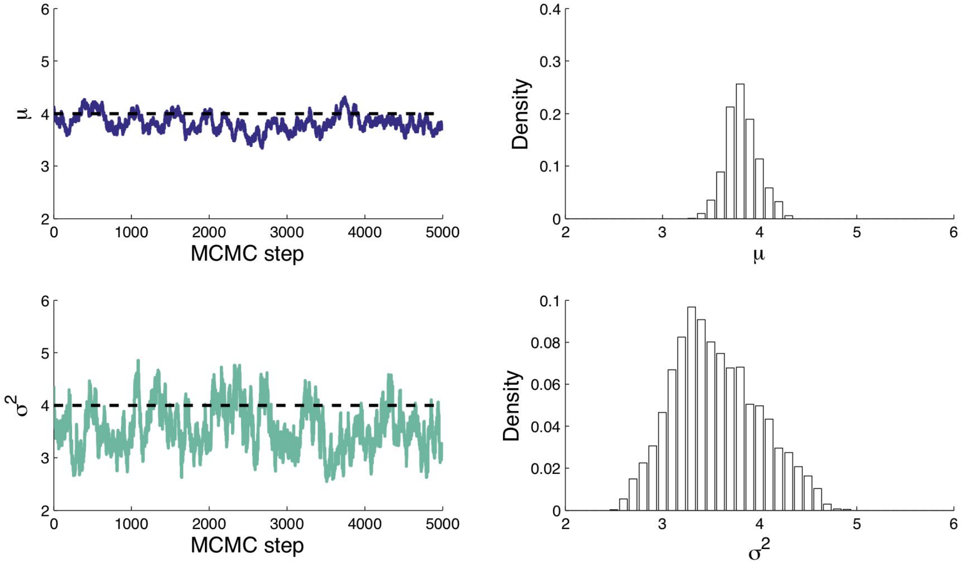
MCMC in higher dimensions

MCMC is often used to infer the **joint posterior distribution** of two or more variables e.g. $p(X,Y|Z)$

For many high-dimensional problems, MCMC is the only practical approach to Bayesian inference.

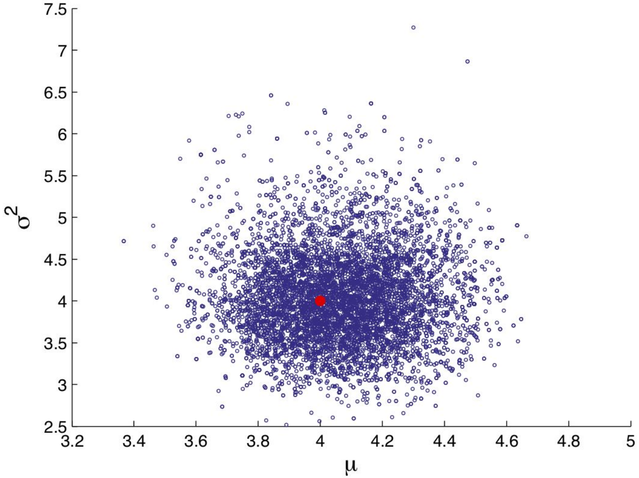
MCMC: 2D example

Let's use the MH algorithm to estimate both the mean and variance of a normal distribution:



MCMC: 2D example

In 2D, the amount of time the chain spends at a particular combination of parameters is proportional to their joint posterior probability.



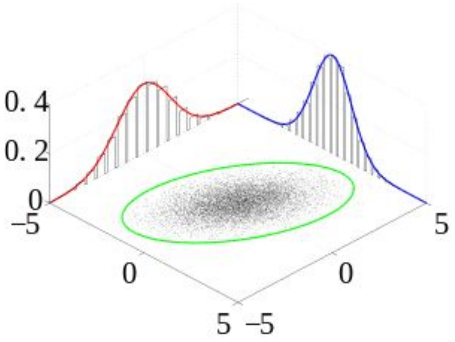
Joint probability here means probability of a μ value **and** a σ^2 value together

Joint vs. marginal distributions

The **joint posterior** is the probability distribution over all unknown variables or parameters.

The **marginal posterior** is the probability distribution over a given parameter integrated (i.e. averaged) over all possible values of the other parameters.

Computing the marginal distribution allows us to take into account uncertainty in other estimated parameters.



Summary of Bayesian inference

Bayes theorem tells us how to compute conditional probabilities of the form *P(A|B)* given we have information about *P(A)*. *P(A)* represents our prior beliefs about *A*.

Bayes theorem lets us compute the posterior distribution of a variable by combining prior information with new information coming from the data through the likelihood function.

Both the posterior and the prior are probability distributions over an unobserved (random) variable.

For many problems, we cannot directly compute the posterior but we can approximate it using MCMC.