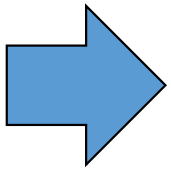
Introduction to Bayesian Statistics - 6

Edoardo Milotti

Università di Trieste and INFN-Sezione di Trieste

Bayesian estimates often require the evaluation of complex integrals. Usually these integrals can only be evaluated with numerical methods.



enter the Monte Carlo methods!

1. acceptance-rejection sampling
2. importance sampling
3. statistical bootstrap
4. Bayesian methods in a sampling-resampling perspective
5. introduction to Markov chains and to the Metropolis algorithm
6. Markov Chain Monte Carlo (MCMC)

5. ***Very*** *short introduction to Markov chains*

Consider a system such that

- the system can occupy a finite or countably infinite set of states S_n;
- the system changes state randomly at discrete times $t = 1, 2, \ldots$;
- if the system is in state S_i, then the probability that the system goes into state S_j is

$$p_{ij} = P[S(n+1) = S_j | S(n) = S_i] \qquad i, j = 1, 2, \ldots$$

  i.e., this probability depends only on the previous state, and is independent o all previous states (this is the *Markov property*);

- the *transition probabilities* p_{ij} do not depend on time n.

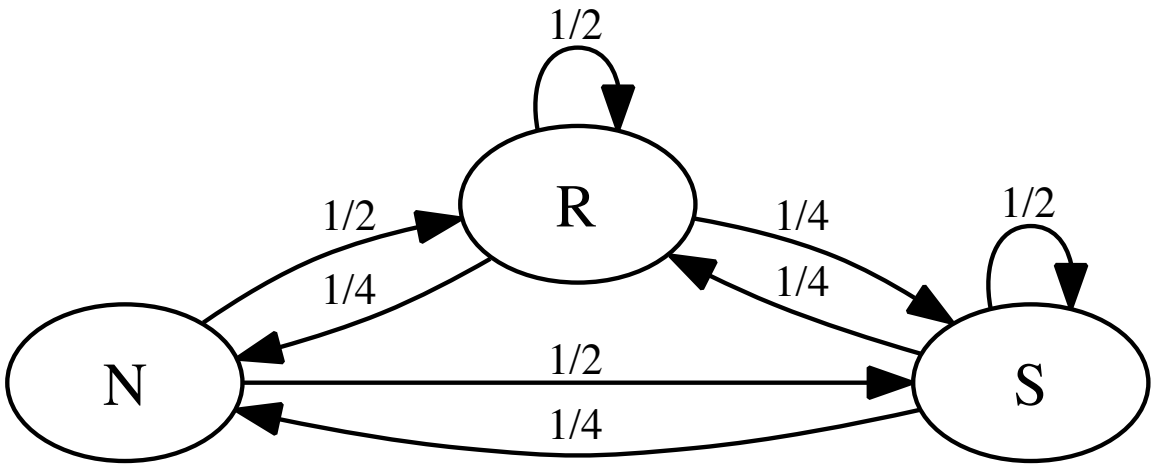
Such a system is a special type of discrete time stochastic process, which is called *Markov chain*.

Example:

in the Land of Oz they never have two nice days in a row, rather, after a sunny day it either rains or snows.

If they have a nice day, they are just as likely to have snow as rain the next day. If they have snow or rain, they have an even chance of having the same the next day. If there is change from snow or rain, only half of the time is this a change to a nice day. When we denote the three states with the symbols N (Nice), R (Rain), or S (Snow), the transition probabilities are:

$$\begin{array}{lll} p_{NN} = 0; & p_{NR} = 1/2; & p_{NS} = 1/2 \\ p_{RN} = 1/4; & p_{RR} = 1/2; & p_{RS} = 1/4 \\ p_{SN} = 1/4; & p_{SR} = 1/4; & p_{SS} = 1/2 \end{array}$$



(representation as a directed graph)

Matrix of transition probabilities (also called *transition kernel*)

$$\mathbf{P} = \begin{pmatrix} p_{NN} & p_{NR} & p_{NS} \\ p_{RN} & p_{RR} & p_{RS} \\ p_{SN} & p_{SR} & p_{SS} \end{pmatrix} = \begin{pmatrix} 0 & 1/2 & 1/2 \\ 1/4 & 1/2 & 1/4 \\ 1/4 & 1/4 & 1/2 \end{pmatrix}$$

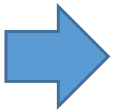
This is a *row stochastic matrix*, where all rows are such that

$$\sum_j p_{ij} = 1$$

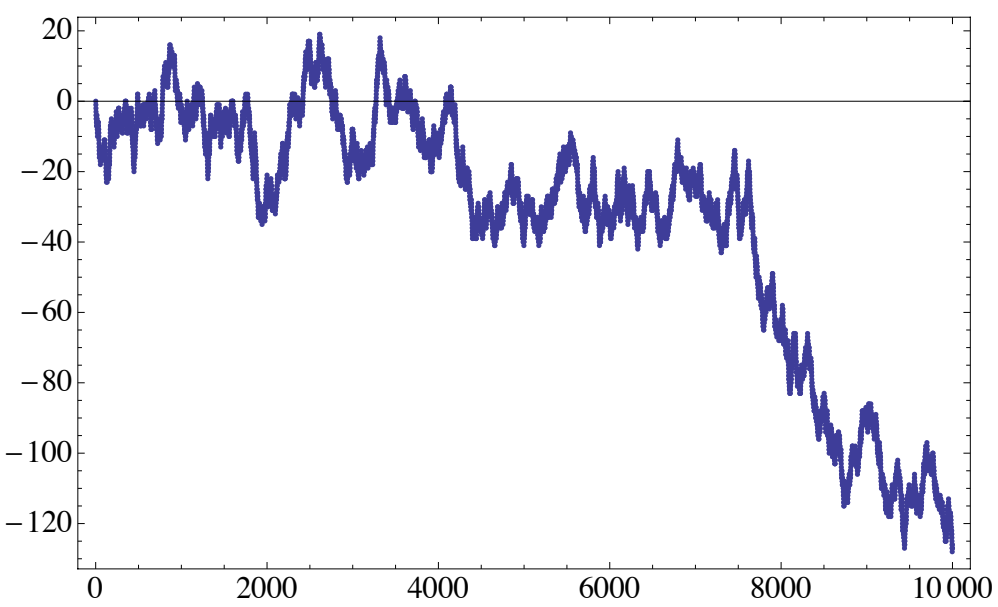
There are also column stochastic matrices, and doubly stochastic matrices that are necessarily square:

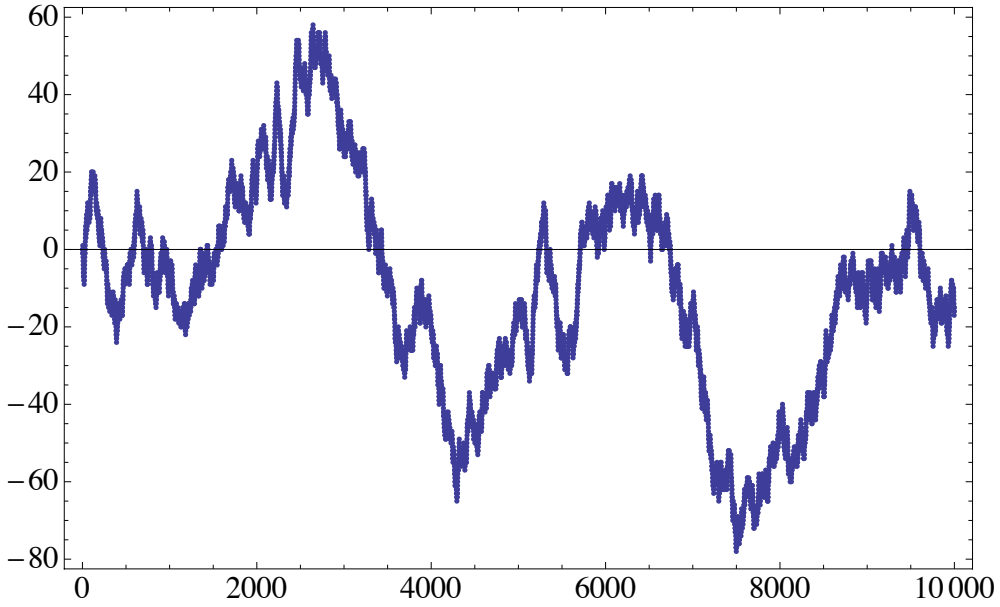
$$\sum_{i=1}^{n}\sum_{j=1}^{m} p_{ij} = \sum_{i=1}^{n} 1 = n$$

$$\sum_{j=1}^{m}\sum_{i=1}^{n} p_{ij} = \sum_{j=1}^{m} 1 = m$$



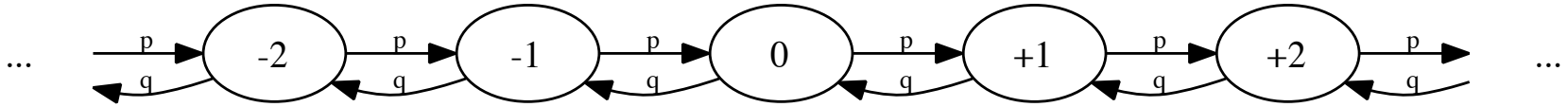
$$m = n$$





Discrete-time discrete-space random walks are an example of Markov chains with infinite states.

$$p_{i,i+1} = p; \quad p_{i,i-1} = q$$



Now let

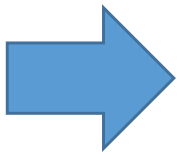
$$\pi_i^{(n)} = P[S(n) = S_i]$$

be the probability that at time *n* the system is in state S_i , then:

$$\pi_j^{(n+1)} = \sum_i P[S(n+1) = S_j | S(n) = S_i] P[S(n) = S_i] = \sum_i p_{ij} \pi_i^{(n)}$$

When we define the vector $\boldsymbol{\pi}^{(n)} = \{\pi_j^{(n)}\}$ and the matrix $\mathbf{P} = \{p_{ij}\}$ we see that the equation becomes

$$\boldsymbol{\pi}^{(n+1)} = \boldsymbol{\pi}^{(n)} \mathbf{P}$$



$$\boldsymbol{\pi}^{(n)} = \boldsymbol{\pi}^{(0)} \mathbf{P}^n$$

n-step transition kernel

For example, the transition kernels for the weather in the Land of Oz are

$$\mathbf{P} = \begin{pmatrix} 0 & 0.5 & 0.5 \\ 0.25 & 0.5 & 0.25 \\ 0.25 & 0.25 & 0.5 \end{pmatrix}$$

$$\mathbf{P}^2 = \begin{pmatrix} 0.25 & 0.375 & 0.375 \\ 0.1875 & 0.4375 & 0.375 \\ 0.1875 & 0.375 & 0.4375 \end{pmatrix}$$

$$\mathbf{P}^5 = \begin{pmatrix} 0.199219 & 0.400391 & 0.400391 \\ 0.200195 & 0.400391 & 0.399414 \\ 0.200195 & 0.399414 & 0.400391 \end{pmatrix}$$

$$\mathbf{P}^{10} = \begin{pmatrix} 0.200001 & 0.4 & 0.4 \\ 0.2 & 0.400001 & 0.4 \\ 0.2 & 0.4 & 0.400001 \end{pmatrix}$$

the transition kernels seem to converge to a fixed matrix ...

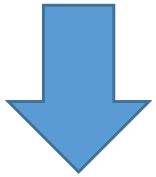
$$\mathbf{P}^{20} = \begin{pmatrix} 0.2 & 0.4 & 0.4 \\ 0.2 & 0.4 & 0.4 \\ 0.2 & 0.4 & 0.4 \end{pmatrix}$$

$$\mathbf{P}^{100} = \begin{pmatrix} 0.2 & 0.4 & 0.4 \\ 0.2 & 0.4 & 0.4 \\ 0.2 & 0.4 & 0.4 \end{pmatrix}$$

Notice that if the transition kernel converges to a fixed matrix where all rows are equal, then the distribution of states also converges to a fixed distribution which does not depend on the initial distribution:

$$\mathbf{P}^n \xrightarrow[n\to\infty]{} \mathbf{P}_\infty \qquad (\mathbf{P}_\infty)_{i,j} = f_j$$

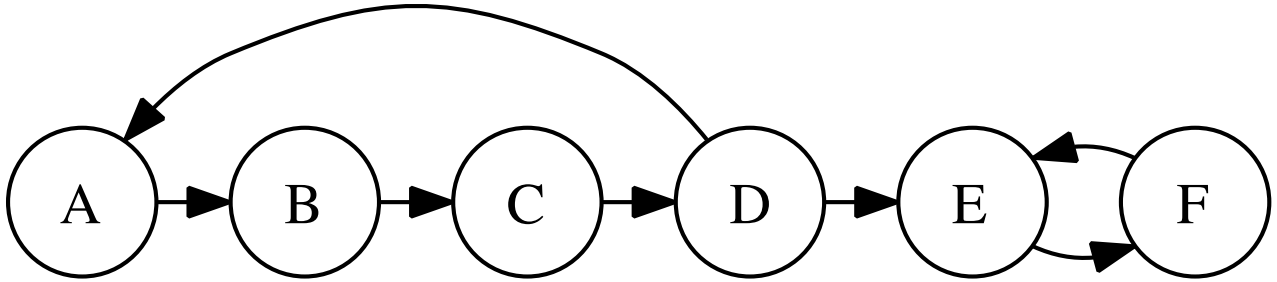
all rows equal



$$\pi_j^{(\infty)} = \sum_i \pi_i^{(0)} (\mathbf{P}_\infty)_{i,j} = \sum_i \pi_i^{(0)} f_j = f_j$$

Persistent and transient states ...

Type of state	Definition of state (assuming, where applicable, that the state is initially occupied)
Periodic	Return to state possible only at times t, $2t$, $3t$, ..., where $t > 1$
Aperiodic	Not periodic
Recurrent/Persistent	Eventual return to state certain
Transient	Eventual return to state uncertain
Ephemeral	Is a state j such that $p_{ij} = 0$ for every i
Positive-recurrent	Recurrent/persistent, finite mean recurrence time
Null-recurrent	Recurrent, infinite mean recurrence time
Ergodic	Aperiodic, positive-recurrent



This graph represents the states and the transition probabilities of a finite Markov chain with 6 states.

The arrows correspond to nonzero transition probabilities. If the chain starts with any one of states A, B, C or D, it can loop around these four states until a transition D to E occurs, then the system is locked in the E-F loop.

States A, B, C, and D are transient, while states E and F are persistent (and periodic, with period 2). A Markov chain with just one class, such that all states communicate, is said to be irreducible. This Markov chain is not irreducible.

VERY INTERESTING MATH ON PERSISTENT STATES, HOWEVER WE DO NOT PURSUE IT FURTHER, WE DO NOT NEED IT NOW.

Limiting probabilities and stationary distributions

Here we prove that the convergence that we saw in the Land of Oz example is a general feature of Markov chains, under the assumption that the chain is irreducible, and that for some N we have

$$\min_{i,j} p_{ij}^{(N)} = \delta > 0$$

Now let

$$r_j^{(n)} = \min_i p_{ij}^{(n)}; \quad R_j^{(n)} = \max_i p_{ij}^{(n)}$$

be the min and max of the j-the column vector in the n-step transition matrix.

Recall the example:

$$\mathbf{P}^2 = \begin{pmatrix} 0.25 & 0.375 & 0.375 \\ 0.1875 & 0.4375 & 0.375 \\ 0.1875 & 0.375 & 0.4375 \end{pmatrix}$$

$$\mathbf{P}^5 = \begin{pmatrix} 0.199219 & 0.400391 & 0.400391 \\ 0.200195 & 0.400391 & 0.399414 \\ 0.200195 & 0.399414 & 0.400391 \end{pmatrix}$$

$$\mathbf{P}^{10} = \begin{pmatrix} 0.200001 & 0.4 & 0.4 \\ 0.2 & 0.400001 & 0.4 \\ 0.2 & 0.4 & 0.400001 \end{pmatrix}$$

$$\mathbf{P}^{20} = \begin{pmatrix} 0.2 & 0.4 & 0.4 \\ 0.2 & 0.4 & 0.4 \\ 0.2 & 0.4 & 0.4 \end{pmatrix}$$

$$\mathbf{P}^{100} = \begin{pmatrix} 0.2 & 0.4 & 0.4 \\ 0.2 & 0.4 & 0.4 \\ 0.2 & 0.4 & 0.4 \end{pmatrix}$$

we shall show that, in each column, the min and the max become closer and closer as n grows and bracket a value that is the asymptotic matrix element (the same for all rows in a given column)

Then we find

$$r_j^{(n+1)} = \min_i p_{ij}^{(n+1)} = \min_i \mathbf{P}_{ij}^{n+1} = \min_i (\mathbf{P}\mathbf{P}^n)_{ij} = \min_i \sum_k p_{ik} p_{kj}^{(n)}$$
$$\geq \min_i \sum_k p_{ik} r_j^{(n)} = r_j^{(n)}$$

and

$$R_j^{(n+1)} = \max_i p_{ij}^{(n+1)} = \max_i \mathbf{P}_{ij}^{n+1} = \max_i (\mathbf{P}\mathbf{P}^n)_{ij} = \max_i \sum_k p_{ik} p_{kj}^{(n)}$$
$$\leq \max_i \sum_k p_{ik} R_j^{(n)} = R_j^{(n)}$$

This means that, as n grows, the minimum and the maximum values in a column vector get closer and closer (the components of the column vector get closer and closer). *But do they converge to the same value ???*

We must consider the difference

$$R_j^{(n)} - r_j^{(n)} = \max_i p_{ij}^{(n)} - \min_k p_{kj}^{(n)} = \max_{i,k} \left[p_{ij}^{(n)} - p_{kj}^{(n)} \right]$$

Then, shifting the difference by N, we find

$$R_j^{(n+N)} - r_j^{(n+N)} = \max_{i,k} \left[p_{ij}^{(n+N)} - p_{kj}^{(n+N)} \right] = \max_{i,k} \left\{ \sum_l \left[p_{il}^{(N)} - p_{kl}^{(N)} \right] p_{lj}^{(n)} \right\}$$

Next we split the difference enclosed in braces into sums of negative and positive contributions

$$\begin{aligned} \sum_l \left[p_{il}^{(N)} - p_{kl}^{(N)} \right] p_{lj}^{(n)} &= \sum_l^+ [p_{il}^{(N)} - p_{kl}^{(N)}] p_{lj}^{(n)} + \sum_l^- [p_{il}^{(N)} - p_{kl}^{(N)}] p_{lj}^{(n)} \\ &\leq \sum_l^+ [p_{il}^{(N)} - p_{kl}^{(N)}] R_j^{(n)} + \sum_l^- [p_{il}^{(N)} - p_{kl}^{(N)}] r_j^{(n)} \end{aligned}$$

Now consider the structure of the positive sum, it must contain at least one term where one subtracts the smallest element in the column, so that

$$\sum_l^+ [p_{il}^{(N)} - p_{kl}^{(N)}] = \sum_l^+ p_{il}^{(N)} - \sum_l^+ p_{kl}^{(N)} \leq \sum_l p_{il}^{(N)} - \delta = 1 - \delta$$

Similarly, for the negative sum we find

$$\sum_l^- [p_{il}^{(N)} - p_{kl}^{(N)}] = \sum_l^- p_{il}^{(N)} - \sum_l^- p_{kl}^{(N)} \geq \delta - \sum_l p_{kl}^{(N)} = -(1 - \delta)$$

and therefore

$$\sum_l \left[p_{il}^{(N)} - p_{kl}^{(N)} \right] p_{lj}^{(n)} \leq \sum_l^+ [p_{il}^{(N)} - p_{kl}^{(N)}] R_j^{(n)} + \sum_l^- [p_{il}^{(N)} - p_{kl}^{(N)}] r_j^{(n)}$$
$$\leq (1 - \delta) R_j^{(n)} - (1 - \delta) r_j^{(n)} = (1 - \delta)(R_j^{(n)} - r_j^{(n)})$$

so that taking strides of N steps at a time, and recalling that $0 < 1 - \delta < 1$

$$R_j^{(kN)} - r_j^{(kN)} < (1 - \delta)^k \left[R_j^{(N)} - r_j^{(N)} \right] \xrightarrow[k \to \infty]{} 0$$

Since

$$R_j^{(kN)} - r_j^{(kN)} < (1-\delta)^k \left[R_j^{(N)} - r_j^{(N)} \right] \xrightarrow[k\to\infty]{} 0$$

the matrix elements in the column converge to a single value p_j^*, i.e.,

$$p_{ij}^* = \lim_{n\to\infty} [\mathbf{P}^n]_{ij} = p_j^*$$

and

$$\pi_j^* = \sum_k \pi_k^{(0)} p_{kj}^* = \sum_k \pi_k^{(0)} p_j^* = p_j^*$$

This asymptotic distribution is stable, indeed from

$$\pi_j^{(n)} = \sum_k \pi_k^{(n-1)} p_{kj}$$

we find

$$[\pi^* \mathbf{P}]_j = \sum_k \pi_k^* p_{kj} = \sum_k p_k^* p_{kj} = \sum_k p_{ik}^* p_{kj} = p_{ij}^* = p_j^* = \pi_j^*$$

or, in matrix form

$$\pi^* = \pi^* \mathbf{P}$$

i.e., the asymptotic probability vector is the left eigenvector with eigenvalue 1 of the transition probability matrix. The distribution expressed by the probability vector π^* is called *invariant distribution* or *stationary distribution*.

Detailed balance

From the definition of conditional probabilities we find

$$P[S(n) = S_i \text{ and } S(n+1) = S_j] = P[S(n) = S_i | S(n+1) = S_j] P[S(n+1) = S_j]$$
$$= P[S(n+1) = S_j | S(n) = S_i] P[S(n) = S_i]$$

therefore, when a Markov chain is time reversed we find

$$P[S(n) = S_i | S(n+1) = S_j]$$
$$= P[S(n+1) = S_j | S(n) = S_i] \frac{P[S(n) = S_i]}{P[S(n+1) = S_j]}$$

i.e.,

$$P[S(n) = S_i | S(n+1) = S_j] = p_{ij} \frac{\pi_i^{(n)}}{\pi_j^{(n+1)}}$$

which shows that the reversed chain is time-dependent.

However if states are distributed according to the invariant distribution, we have

$$P[S(n) = S_i | S(n+1) = S_j] = p_{ij} \frac{\pi_i^*}{\pi_j^*}$$

which means that the backward transition probabilities are again time-independent, and in particular they must coincide with the forward transition probabilities, i.e.,

$$p_{ji}\pi_j^* = p_{ij}\pi_i^*$$

a condition which is called *detailed balance*.

So *if* stationary distribution *then* detailed balance ... however the reverse also holds

$$\pi_j^{(n+1)} = \sum_i \pi_i^{(n)} p_{ij} = \sum_i \pi_j^{(n)} p_{ji} = \pi_j^{(n)} \sum_i p_{ji} = \pi_j^{(n)}$$

i.e., *a distribution is stationary if and only if it satisfies the condition of detailed balance*

Physical aside: continuous-time Markov processes

The time-dependence of the reversed chain is a manifestation of the dissipative character of the chain. Another important related result is the validity of the H-theorem for Markov processes.

In the case of continuous-time processes we can write

$$P\left(S_{i_k}, t_k; S_{i_{k-1}}, t_{k-1}; \ldots; S_{i_0}, t_0\right) = \\ = P\left(S_{i_k}, t_k | S_{i_{k-1}}, t_{k-1}; \ldots; S_{i_0}, t_0\right) P\left(S_{i_{k-1}}, t_{k-1}; \ldots; S_{i_0}, t_0\right)$$

Memoryless processes

$$P\left(S_{i_k}, t_k; S_{i_{k-1}}, t_{k-1}; \ldots; S_{i_0}, t_0\right) = P\left(S_{i_k}, t_k\right)$$

Markov processes

$$P\left(S_{i_k}, t_k; S_{i_{k-1}}, t_{k-1}; \ldots; S_{i_0}, t_0\right) = P\left(S_{i_k}, t_k | S_{i_{k-1}}, t_{k-1}\right) P\left(S_{i_{k-1}}, t_{k-1}\right)$$

For Markov processes the following equation also holds

$$P(S_n, t + \Delta t) = P(S_n, t) + \\ + \sum_j \left[P\left(S_n, t + \Delta t | S_j, t\right) P(S_j, t) - P\left(S_j, t + \Delta t | S_n, t\right) P(S_n, t) \right]$$

(*master equation*).

When we assume that the transition probabilities are time-invariant and we define the transition rates T

$$P\left(S_n, t + \Delta t | S_j, t\right) = T_{n,j} \Delta t$$

we find the differential form of the master equation

$$\frac{d}{dt} P(S_n, t) = \sum_j \left[T_{n,j} P(S_j, t) - T_{j,n} P(S_n, t) \right]$$

Using the previous notation for the probability distribution on states, we can rewrite the master equation as follows

$$\frac{d\pi_n}{dt} = \sum_j \left[T_{n,j}\pi_j(t) - T_{j,n}\pi_n(t)\right]$$

Next, we assume that transition probabilities are "reversible"

$$T_{n,j} = T_{j,n}$$

so that

$$\frac{d\pi_n}{dt} = \sum_j T_{n,j}\left[\pi_j(t) - \pi_n(t)\right]$$

and therefore, at equilibrium

$$\sum_j T_{n,j}\left(\pi_j^* - \pi_n^*\right) = 0 \quad \Rightarrow \quad \pi_j^* = \pi_n^*$$

all states are equally likely at equilibrium

Now consider the following sum

$$H = \sum_n \pi_n \ln \pi_n$$

Using the master equation we find a differential equation for H

$$\frac{dH}{dt} = \sum_n \frac{d}{dt}(\pi_n \ln \pi_n) = \sum_n \frac{d\pi_n}{dt}(\ln \pi_n + 1)$$

$$= \sum_{n,j} T_{n,j} \left(\pi_j - \pi_n\right) \left(\ln \pi_n + 1\right)$$

Exchanging indexes ...

$$\frac{dH}{dt} = \sum_{n,j} T_{n,j} \left(\pi_n - \pi_j\right) \left(\ln \pi_j + 1\right)$$

Adding the two differential equations we find

$$\frac{dH}{dt} = \frac{1}{2}\sum_{n,j} T_{n,j}\left(\pi_n - \pi_j\right)\left(\ln \pi_j - \ln \pi_n\right)$$

Since

$$\left(\pi_n - \pi_j\right)\left(\ln \pi_j - \ln \pi_n\right) \leq 0$$

we find

$$\boxed{\frac{dH}{dt} \leq 0}$$

Boltzmann's H-theorem

The derivative vanishes at equilibrium, and we find that it is a stable point for *H*. Since *H* is essentially the negative of Gibbs' entropy, the theorem states that the entropy of a Markov chain increases up to a maximum which is reached at equilibrium.

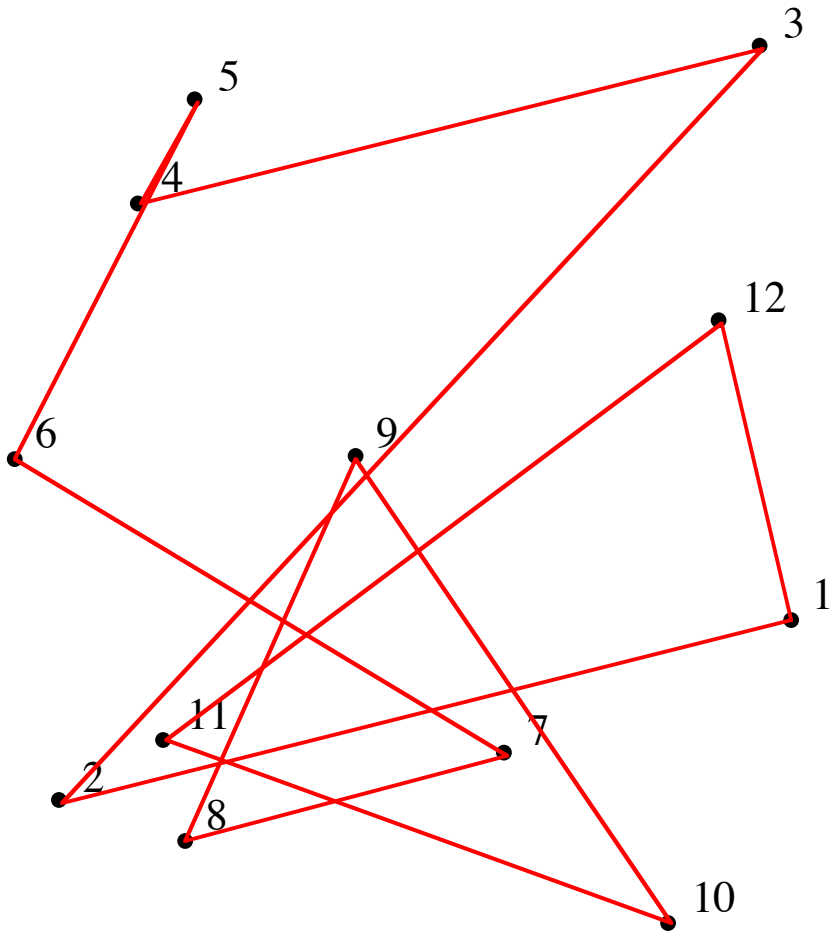
5.1 From Markov Chains to Markov Chain Monte Carlo programs

To introduce the method, we consider the *Traveling Salesman Problem* (TSP), where we want to find the shortest closed path that connects N cities.

The problem was first stated by the Viennese mathematician Karl Menger and in 1930 is one of the most studied problems in combinatorial optimization.

For many up to date links, see
http://www.math.uwaterloo.ca/tsp/index.html

In particular see the history page
http://www.math.uwaterloo.ca/tsp/history/index.html



12 “cities” randomly distributed in the (0,1) square: the path corresponds to a random permutation of the sequence of cities.

(path length L=1.93834)

Paths are enumerated by permutations of “city names”, e.g., {9, 2, 7, 8, 1, 12, 4, 5, 3, 10, 11, 6} (start at 9, step to 2, and so on until you reach 6 and then return to 9).

The total number of configurations (undirected paths) is

$$\frac{1}{2}(n-1)!$$

The problem belongs to the class of NP-complete problems (Non-Polynomial complexity, a class of particulary hard problems)

In such cases there is only one known exact solution: the full enumeration of all paths.

13 May 1983, Volume 220, Number 4598

SCIENCE

Optimization by Simulated Annealing

S. Kirkpatrick, C. D. Gelatt, Jr., M. P. Vecchi

Summary. There is a deep and useful connection between statistical mechanics (the behavior of systems with many degrees of freedom in thermal equilibrium at a finite temperature) and multivariate or combinatorial optimization (finding the minimum of a given function depending on many parameters). A detailed analogy with annealing in solids provides a framework for optimization of the properties of very large and complex systems. This connection to statistical mechanics exposes new information and provides an unfamiliar perspective on traditional optimization problems and methods.

Approximate solution of the TSP with the Simulated Annealing algorithm

path length → **energy of the system**

exploration of the configuration space with the *Metropolis algorithm*
(Metropolis, Rosenbluth Rosenbluth ,Teller and Teller, 1953)

THE JOURNAL OF CHEMICAL PHYSICS VOLUME 21, NUMBER 6 JUNE, 1953

Equation of State Calculations by Fast Computing Machines

NICHOLAS METROPOLIS, ARIANNA W. ROSENBLUTH, MARSHALL N. ROSENBLUTH, AND AUGUSTA H. TELLER,
Los Alamos Scientific Laboratory, Los Alamos, New Mexico

AND

EDWARD TELLER,* *Department of Physics, University of Chicago, Chicago, Illinois*

(Received March 6, 1953)

A general method, suitable for fast computing machines, for investigating such properties as equations of state for substances consisting of interacting individual molecules is described. The method consists of a modified Monte Carlo integration over configuration space. Results for the two-dimensional rigid-sphere system have been obtained on the Los Alamos MANIAC and are presented here. These results are compared to the free volume equation of state and to a four-term virial coefficient expansion.

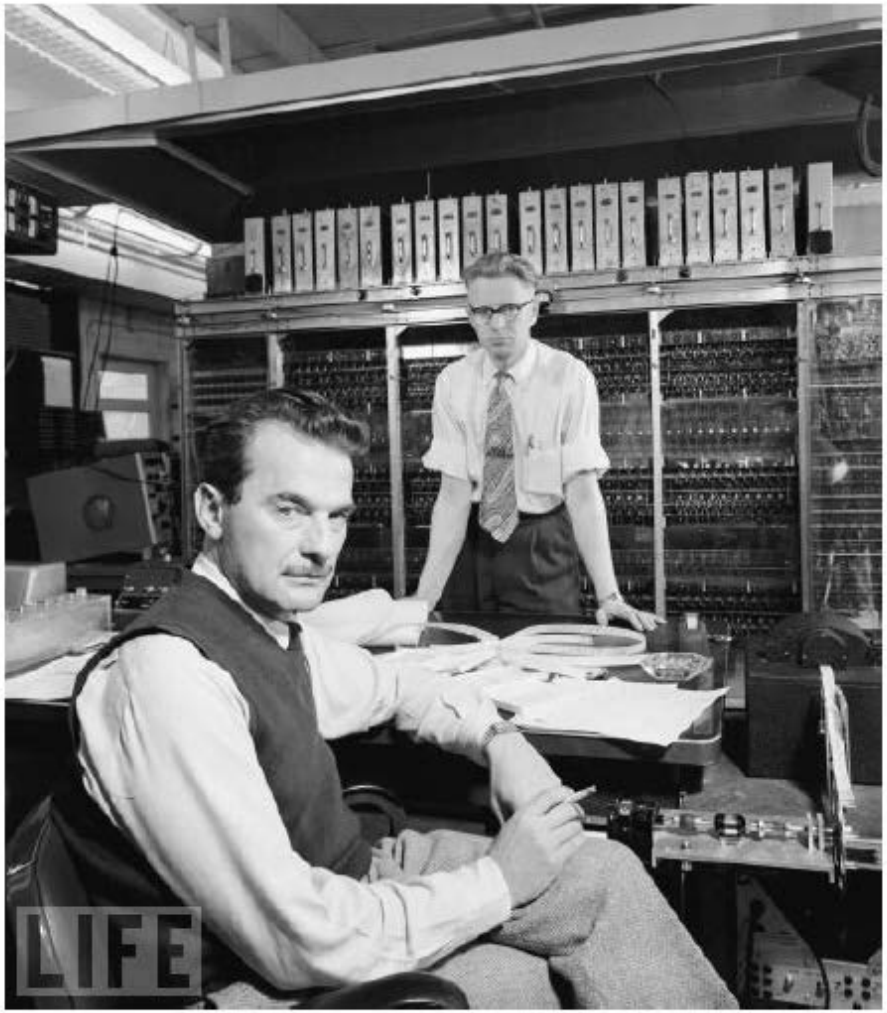


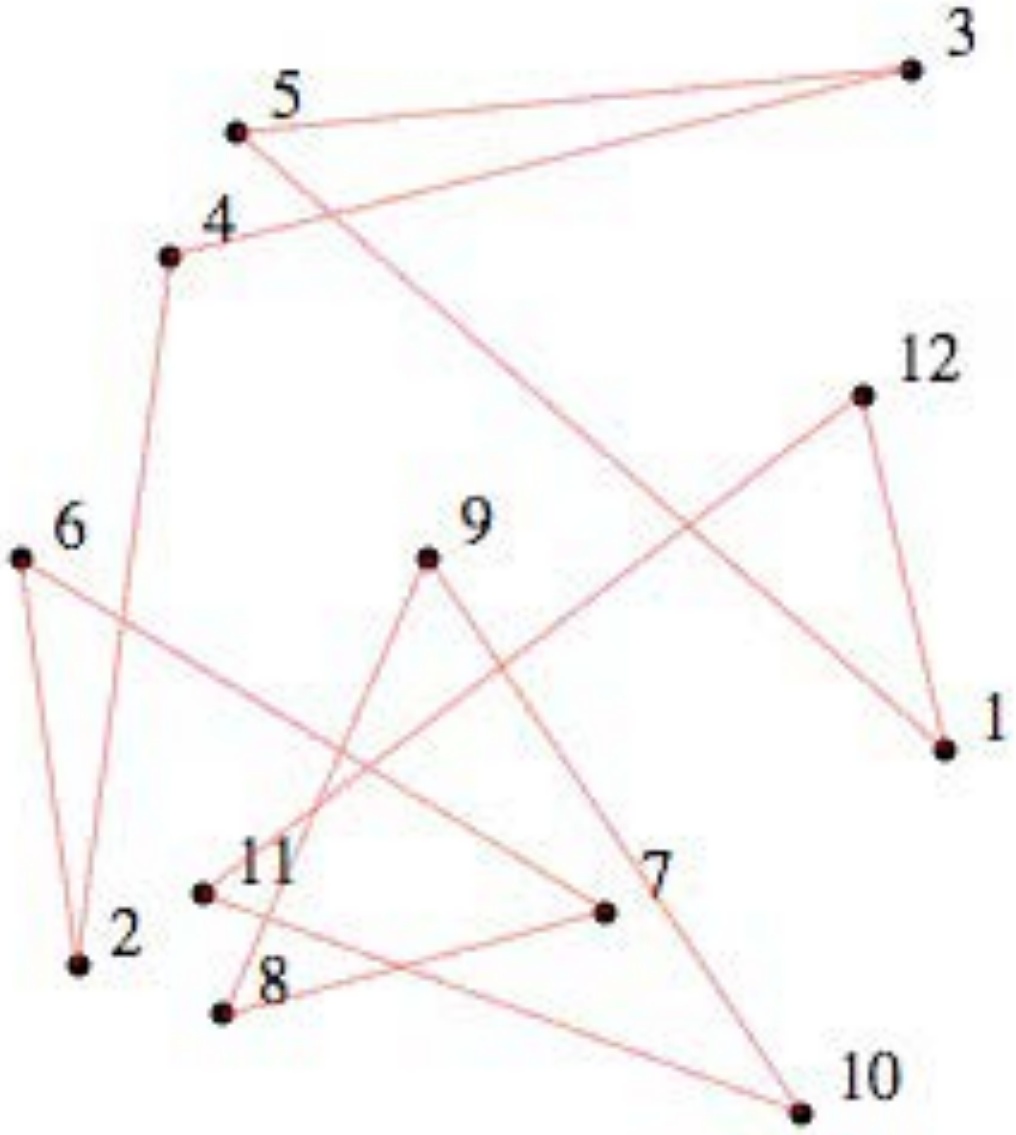
Figure 8.14: Portrait of American computer scientists Nicholas Metropolis (1915 - 1999) (seated) and James Henry Richardson (1918 - 1996) at Los Alamos National Laboratory, Los Alamos, New Mexico, November 1953 (from `http://www.life.com`).

1. We generate a new configuration C′ from the present configuration C
2. We compute the energy of the new configuration, E'
3. We compute the energy difference $\Delta E = E' - E$
4. The new configuration is accepted with probability p

$$\begin{cases} p = 1 & \Delta E < 0 \\ p = \exp\left(-\dfrac{\Delta E}{kT}\right) & \Delta E \geq 0 \end{cases}$$

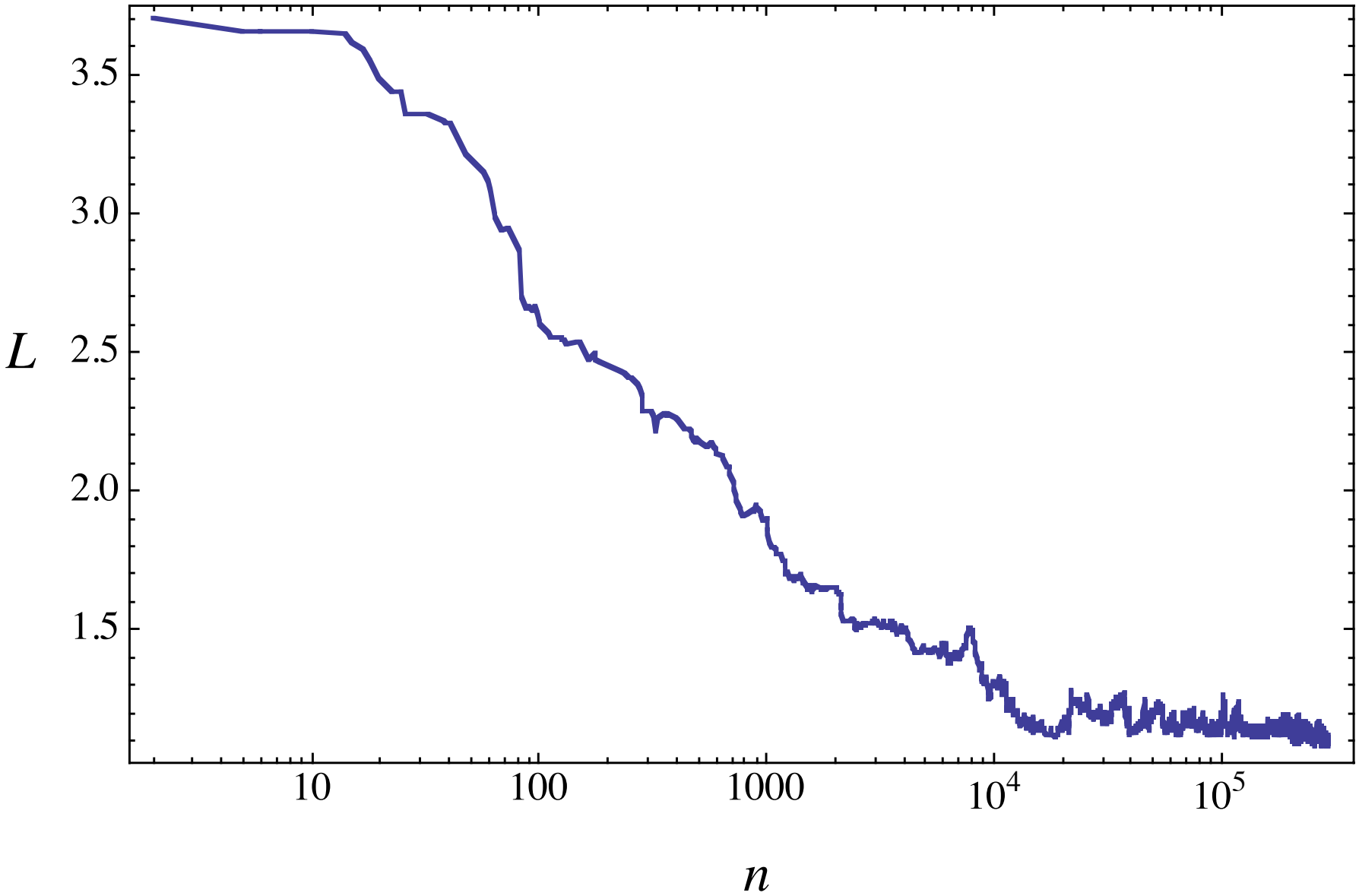
Additional details

- the algorithm needs a slow cooling (it is common to choose an exponential cooling schedule)
- if cooling is not gradual, the system can get stuck into a local minimum
- simple exchanges of pairs of cities are the individual moves in the SA solution of the TSP
- the individual steps from one configuration to the next can be described by a Markov chain



k = 1
T = 0.05
L = 1.84655

Decrease of total path length in a realization of the SA solution of a 50-cities problem



Here we note that the transition probability can be written as follows

$$T\left(C \to C'\right) = \min\left[1, \exp\left(-\frac{\left(E' - E\right)}{kT}\right)\right]$$

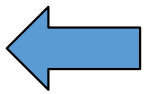
Moreover, it is easy to show that the algorithm preserves detailed balance

$$P(C)T\left(C \to C'\right) = P(C')T\left(C' \to C\right)$$

where P(C) is the stationary probability of configuration C. Indeed at equilibrium we find that, if E' > E,

$$P(C)\exp\left(-\frac{\left(E' - E\right)}{kT}\right) = P(C')$$

$$\frac{P(C')}{P(C)} = \exp\left(-\frac{\left(E' - E\right)}{kT}\right)$$



Boltzmann's distribution

Finally, we can write:

$$T\left(C \to C'\right) = \min\left[1, \frac{P\left(C'\right)}{P\left(C\right)}\right]$$

This definition of the transition probability is the starting point for an important further step, the Metropolis-Hastings algorithm.

6. MCMC – definition of the Metropolis-Hastings (M-H) algorithm (1970)

- we define the transition probability

$$P(\mathbf{x} \to \mathbf{y}) = q(\mathbf{x}, \mathbf{y})\alpha(\mathbf{x}, \mathbf{y})$$

and the target density $\pi(\mathbf{x})$

- we take state $\mathbf{x} = \mathbf{x}_n$

- we choose randomly another state $\mathbf{y}$ and we accept it $\left(\mathbf{y} \to \mathbf{x}_{n+1}\right)$ with probability

$$\alpha(\mathbf{x},\mathbf{y}) = \min\left\{1, \frac{\pi(\mathbf{y})q(\mathbf{y},\mathbf{x})}{\pi(\mathbf{x})q(\mathbf{x},\mathbf{y})}\right\}$$

Note that if the proposal function q is symmetrical, then the acceptance probability takes on the simpler form

$$\alpha(\mathbf{x},\mathbf{y}) = \min\left\{1, \frac{\pi(\mathbf{y})q(\mathbf{y},\mathbf{x})}{\pi(\mathbf{x})q(\mathbf{x},\mathbf{y})}\right\} \rightarrow \min\left\{1, \frac{\pi(\mathbf{y})}{\pi(\mathbf{x})}\right\}$$

and it depends on the target density only.

The M-H algorithm defines a Markov chain and it is easy to show that detailed balance holds. The transition probability is

$$P(\mathbf{x} \to \mathbf{y}) = q(\mathbf{x},\mathbf{y})\alpha(\mathbf{x},\mathbf{y}) = q(\mathbf{x},\mathbf{y})\min\left\{1, \frac{\pi(\mathbf{y})q(\mathbf{y},\mathbf{x})}{\pi(\mathbf{x})q(\mathbf{x},\mathbf{y})}\right\}$$

- case $\dfrac{\pi(\mathbf{y})q(\mathbf{y},\mathbf{x})}{\pi(\mathbf{x})q(\mathbf{x},\mathbf{y})} \geq 1$

$$\Rightarrow \quad \alpha(\mathbf{x},\mathbf{y}) = 1; \quad \alpha(\mathbf{y},\mathbf{x}) = \frac{\pi(\mathbf{x})q(\mathbf{x},\mathbf{y})}{\pi(\mathbf{y})q(\mathbf{y},\mathbf{x})} \quad \Rightarrow \quad \begin{aligned} P(\mathbf{x} \to \mathbf{y}) &= q(\mathbf{x},\mathbf{y}) \\ P(\mathbf{y} \to \mathbf{x}) &= q(\mathbf{y},\mathbf{x})\frac{\pi(\mathbf{x})q(\mathbf{x},\mathbf{y})}{\pi(\mathbf{y})q(\mathbf{y},\mathbf{x})} \end{aligned}$$

$$\Rightarrow \quad \begin{aligned} \pi(\mathbf{x})P(\mathbf{x} \to \mathbf{y}) &= \pi(\mathbf{x})q(\mathbf{x},\mathbf{y}) \\ \pi(\mathbf{y})P(\mathbf{y} \to \mathbf{x}) &= \pi(\mathbf{y})q(\mathbf{y},\mathbf{x})\frac{\pi(\mathbf{x})q(\mathbf{x},\mathbf{y})}{\pi(\mathbf{y})q(\mathbf{y},\mathbf{x})} = \pi(\mathbf{x})q(\mathbf{x},\mathbf{y}) \end{aligned}$$

- case $\dfrac{\pi(\mathbf{y})q(\mathbf{y},\mathbf{x})}{\pi(\mathbf{x})q(\mathbf{x},\mathbf{y})} < 1$

$$\Rightarrow \alpha(\mathbf{x},\mathbf{y}) = \frac{\pi(\mathbf{y})q(\mathbf{y},\mathbf{x})}{\pi(\mathbf{x})q(\mathbf{x},\mathbf{y})}; \quad \alpha(\mathbf{y},\mathbf{x}) = 1 \Rightarrow \begin{aligned} P(\mathbf{x}\to\mathbf{y}) &= q(\mathbf{x},\mathbf{y})\frac{\pi(\mathbf{y})q(\mathbf{y},\mathbf{x})}{\pi(\mathbf{x})q(\mathbf{x},\mathbf{y})} \\ P(\mathbf{y}\to\mathbf{x}) &= q(\mathbf{y},\mathbf{x}) \end{aligned}$$

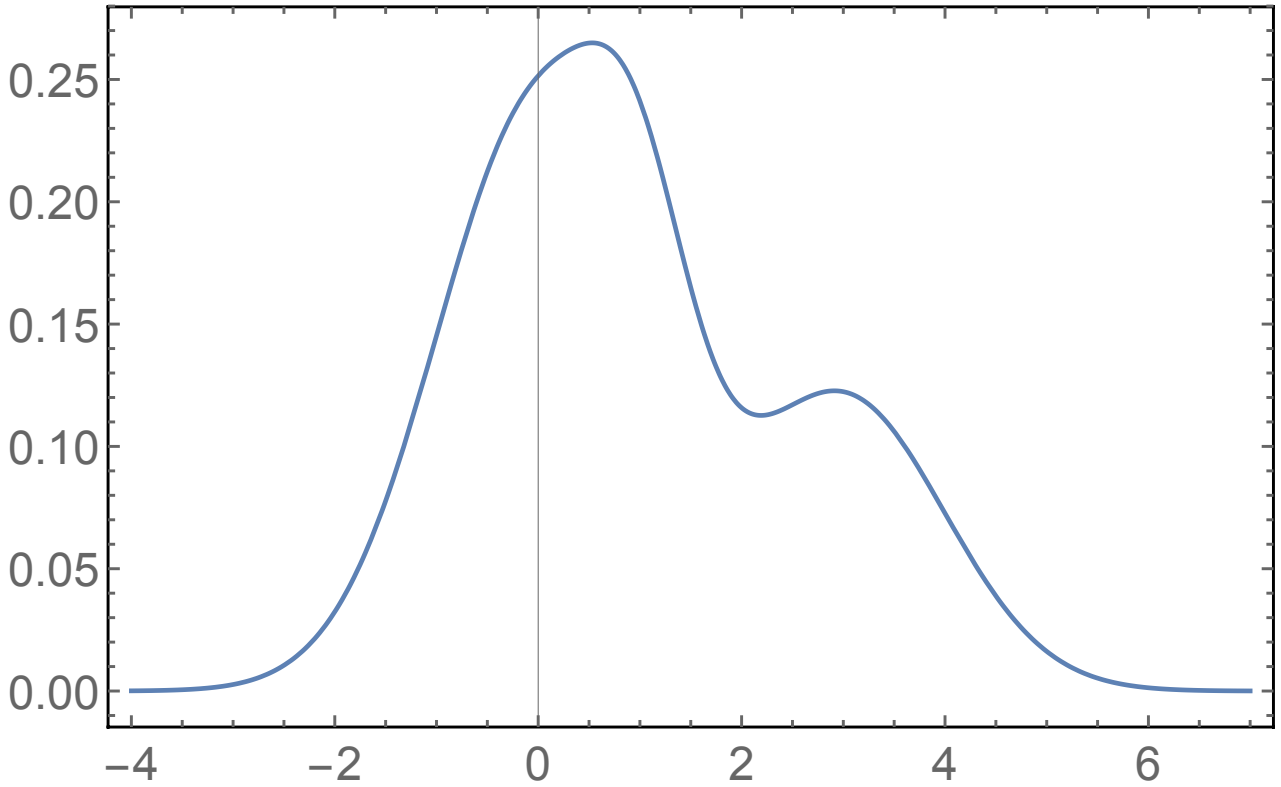
$$\Rightarrow \begin{aligned} \pi(\mathbf{x})P(\mathbf{x}\to\mathbf{y}) &= \pi(\mathbf{x})q(\mathbf{x},\mathbf{y})\frac{\pi(\mathbf{y})q(\mathbf{y},\mathbf{x})}{\pi(\mathbf{x})q(\mathbf{x},\mathbf{y})} = \pi(\mathbf{y})q(\mathbf{y},\mathbf{x}) \\ \pi(\mathbf{y})P(\mathbf{y}\to\mathbf{x}) &= \pi(\mathbf{y})q(\mathbf{y},\mathbf{x}) \end{aligned}$$

Detailed balance holds in both cases and therefore $\pi(\mathbf{x})$ is stationary

The following figure shows a simulation with the MCMC algorithm and the distribution

$$p(x) = \frac{0.6}{\sqrt{2\pi}} \exp\left(-\frac{x^2}{2}\right) + \frac{0.3}{\sqrt{2\pi}} \exp\left(-\frac{(x-3)^2}{2}\right) + \frac{0.1}{\sqrt{0.5\pi}} \exp\left(-\frac{(x-1)^2}{0.5}\right)$$

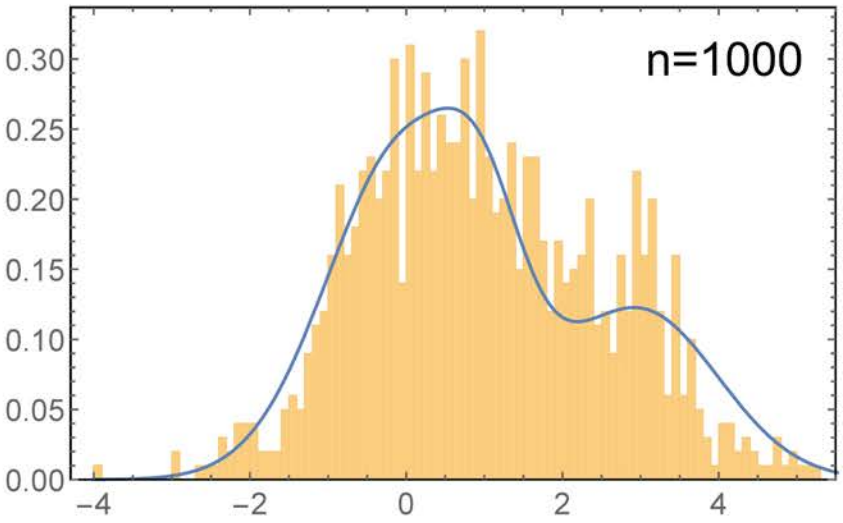
(a three-component mixture model)

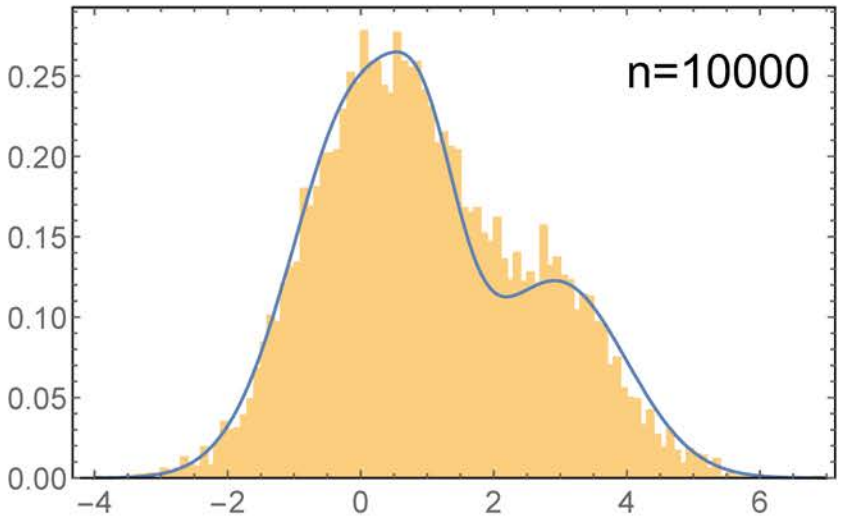


```
nrmax = 40 000;

xr = Table[0, {nrmax}];
xr[[1]] = -4;

nr = 1;
While[nr < nrmax,
 xtry = xr[[nr]] + RandomReal[NormalDistribution[0, 1]];
 If[ pdf[xtry] / pdf[xr[[nr]]] > RandomReal[] , nr++; xr[[nr]] = xtry];
]
```

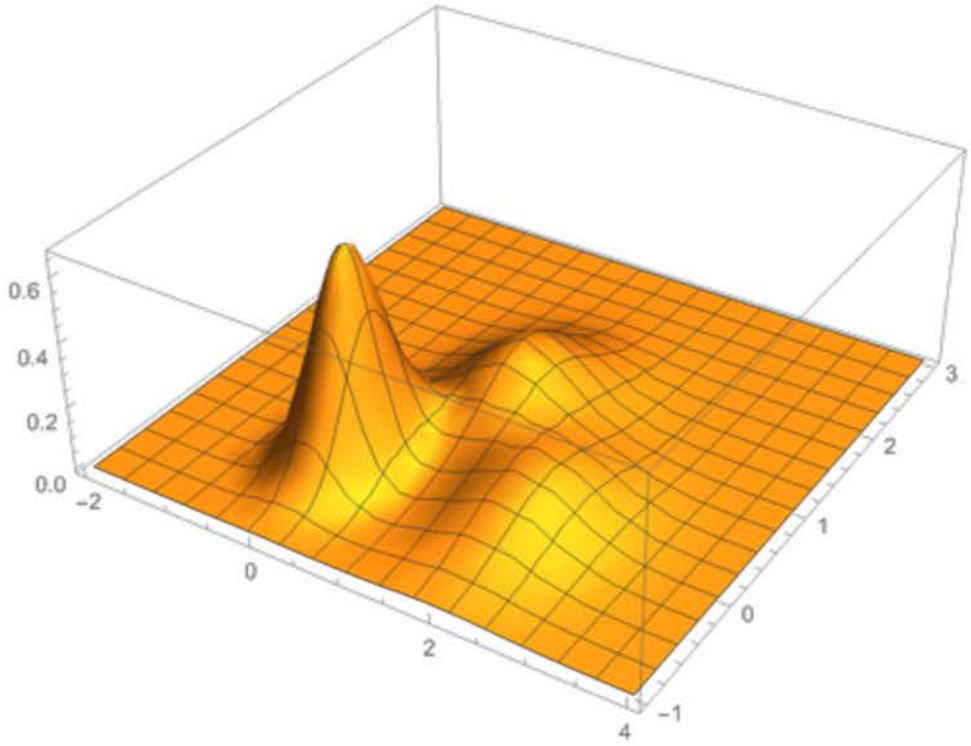




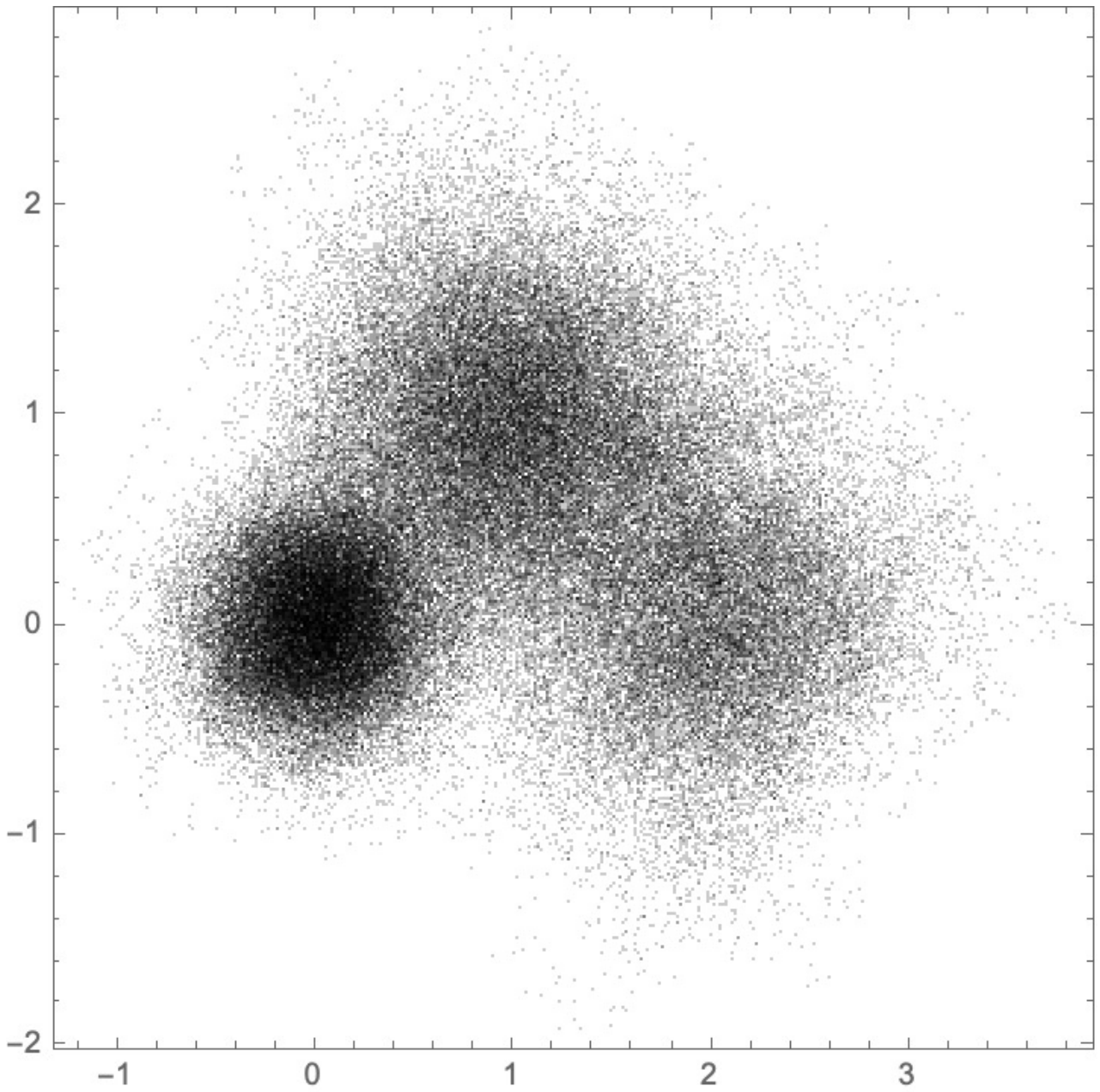
MCMC simulation of a 2D three-component mixture model

$$p(x,y) = \sum_{i=1}^{3} \frac{\alpha_i}{\sqrt{2\sigma_i^2}} \exp\left[-\frac{(x-\mu_{x,i})^2 + (y-\mu_{y,i})^2}{2\sigma_i}\right]$$

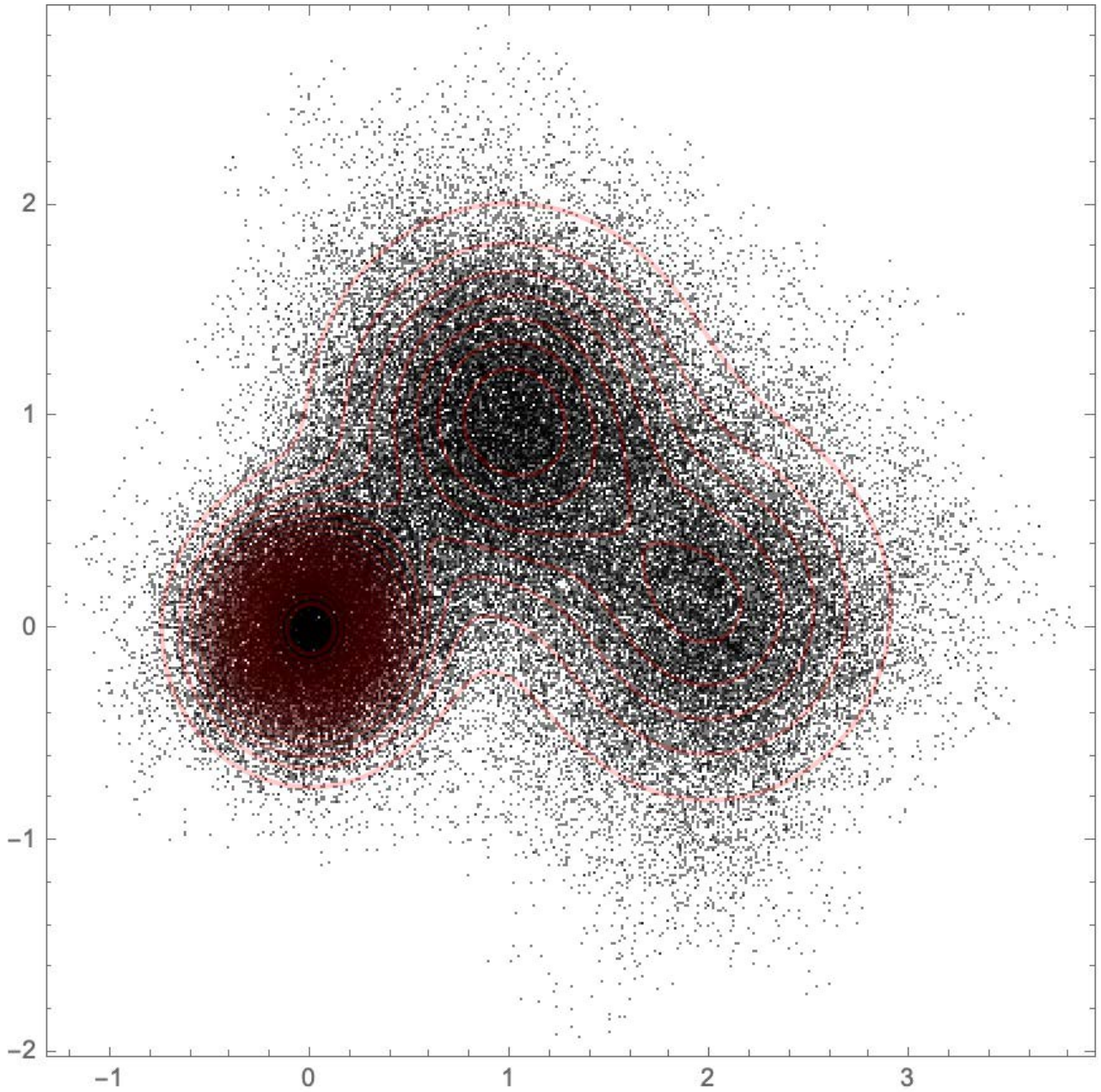
$$\begin{array}{llll} \alpha_1 = 0.5; & \mu_{x,1} = 0; & \mu_{y,1} = 0; & \sigma_1 = 0.3; \\ \alpha_2 = 0.3; & \mu_{x,2} = 1; & \mu_{y,2} = 1.; & \sigma_2 = 0.5; \\ \alpha_3 = 0.2; & \mu_{x,3} = 2; & \mu_{y,3} = 0.1; & \sigma_3 = 0.5; \end{array}$$

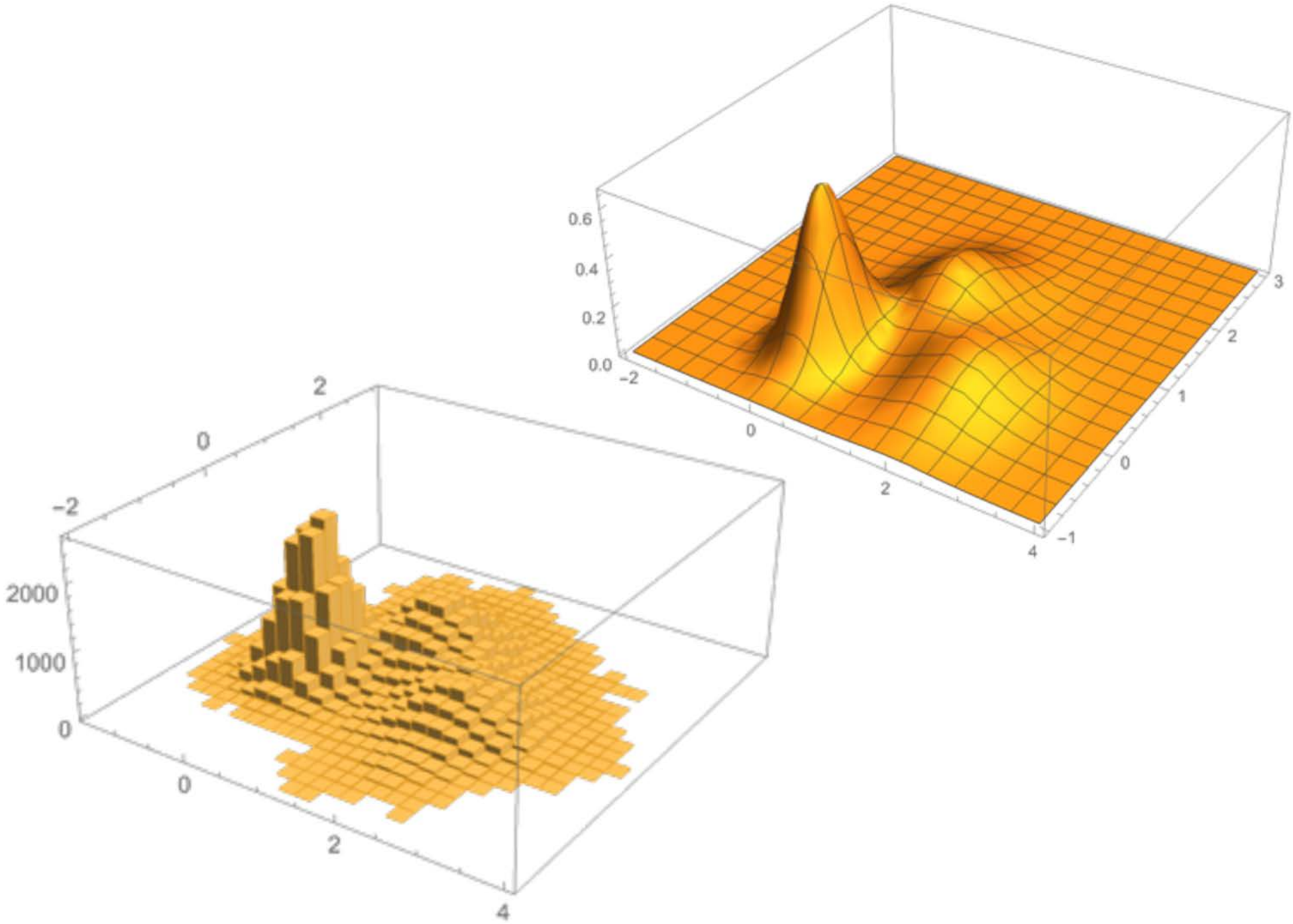


100000 steps

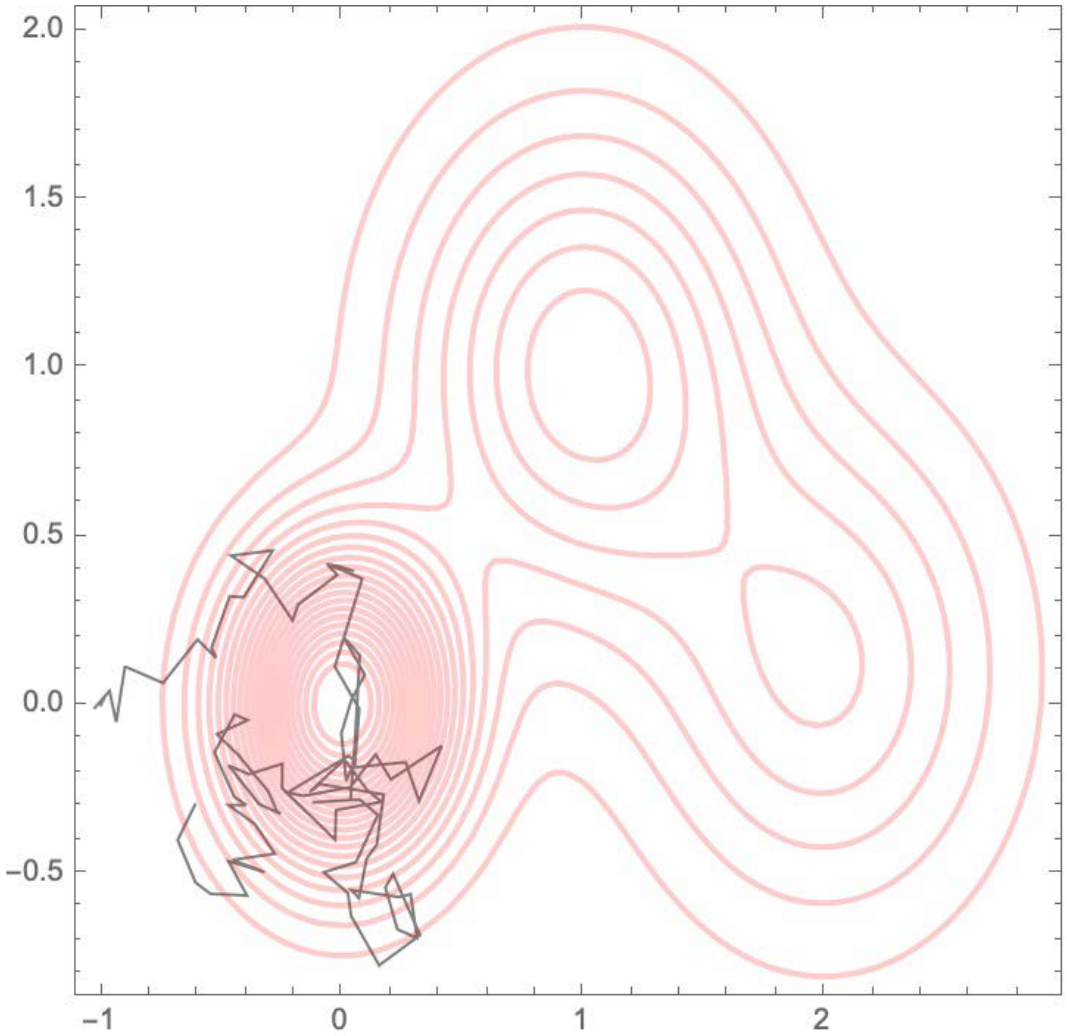


100000 steps

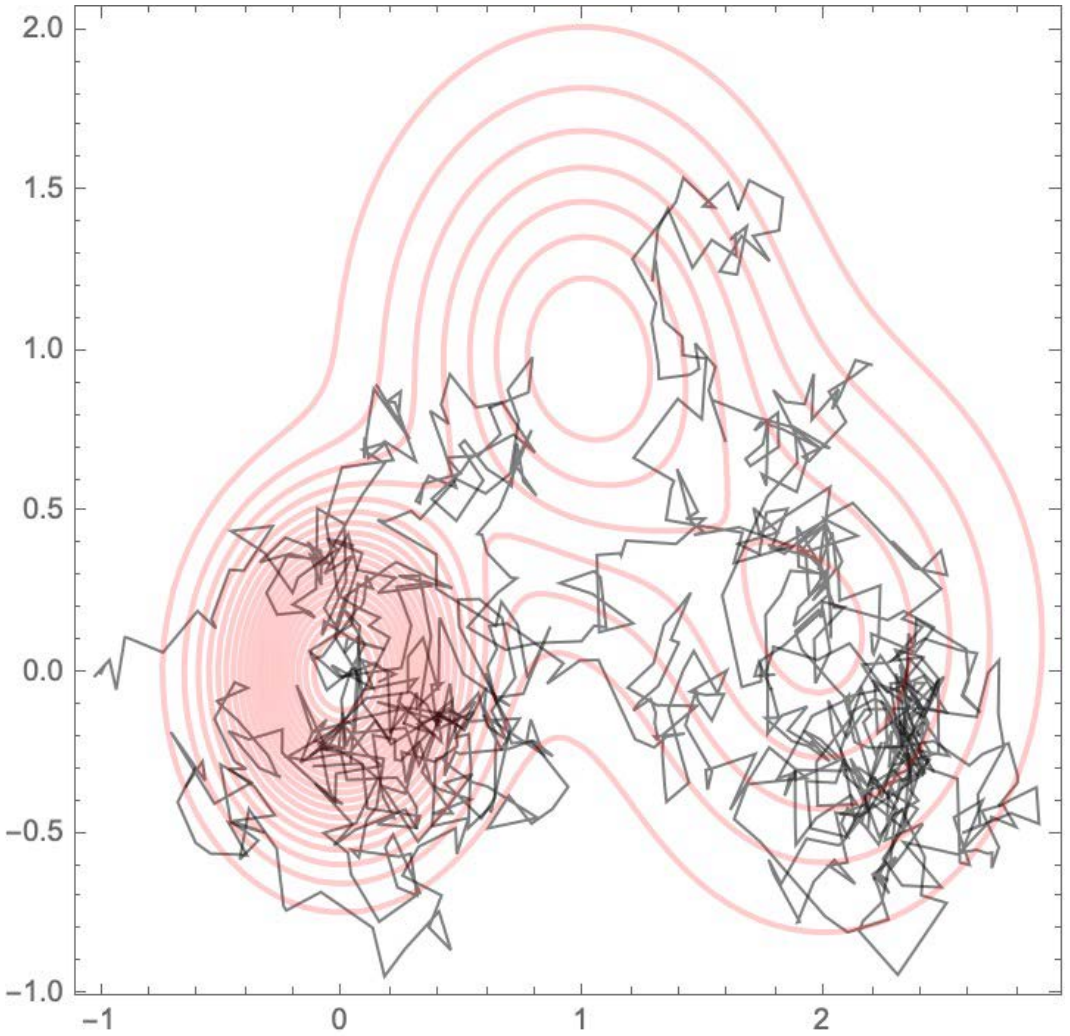



0.6
0.4
0.2
0.0
−2
0
2
4
3
2
1
0
−1
2000
1000
0
−2
0
2
4

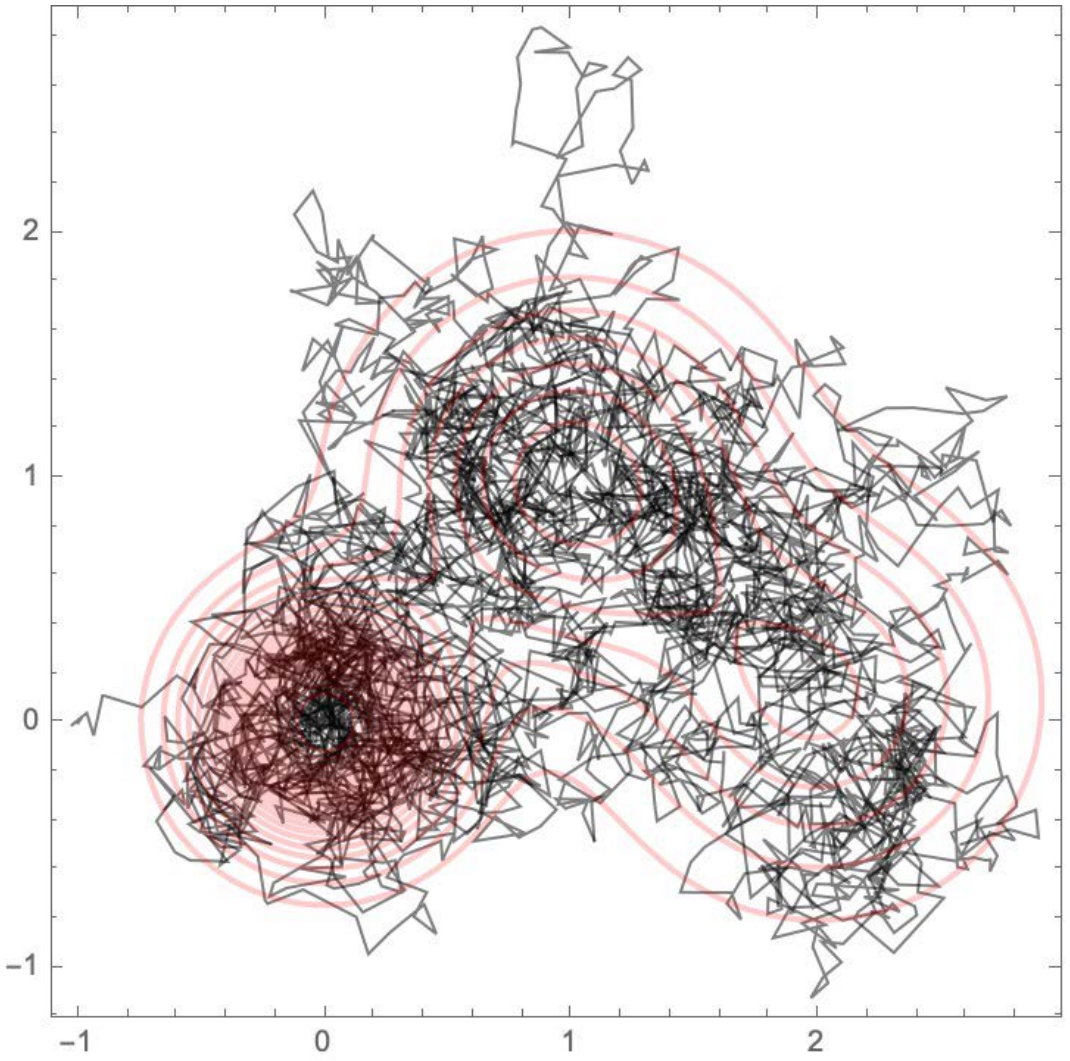
100 steps



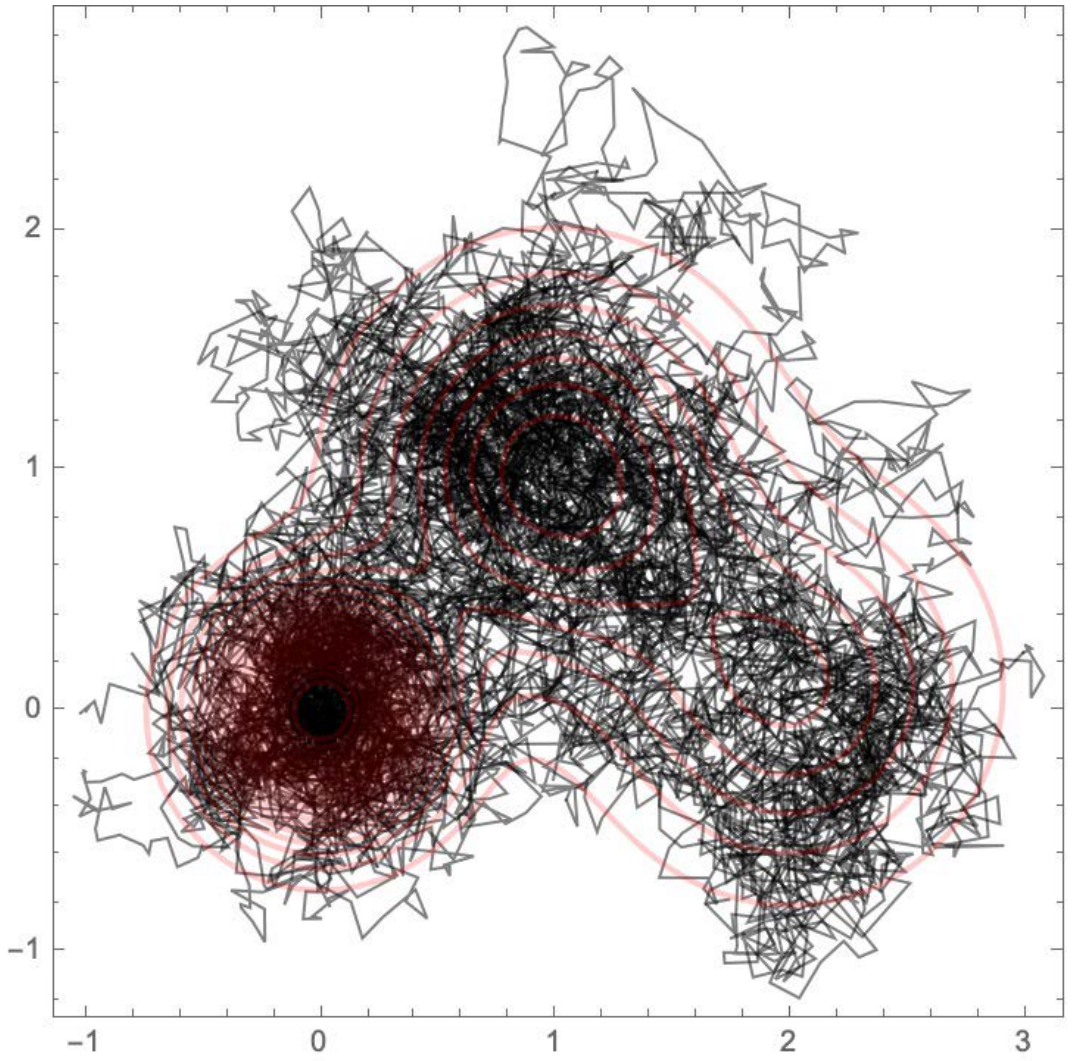
1000 steps



4000 steps



10000 steps

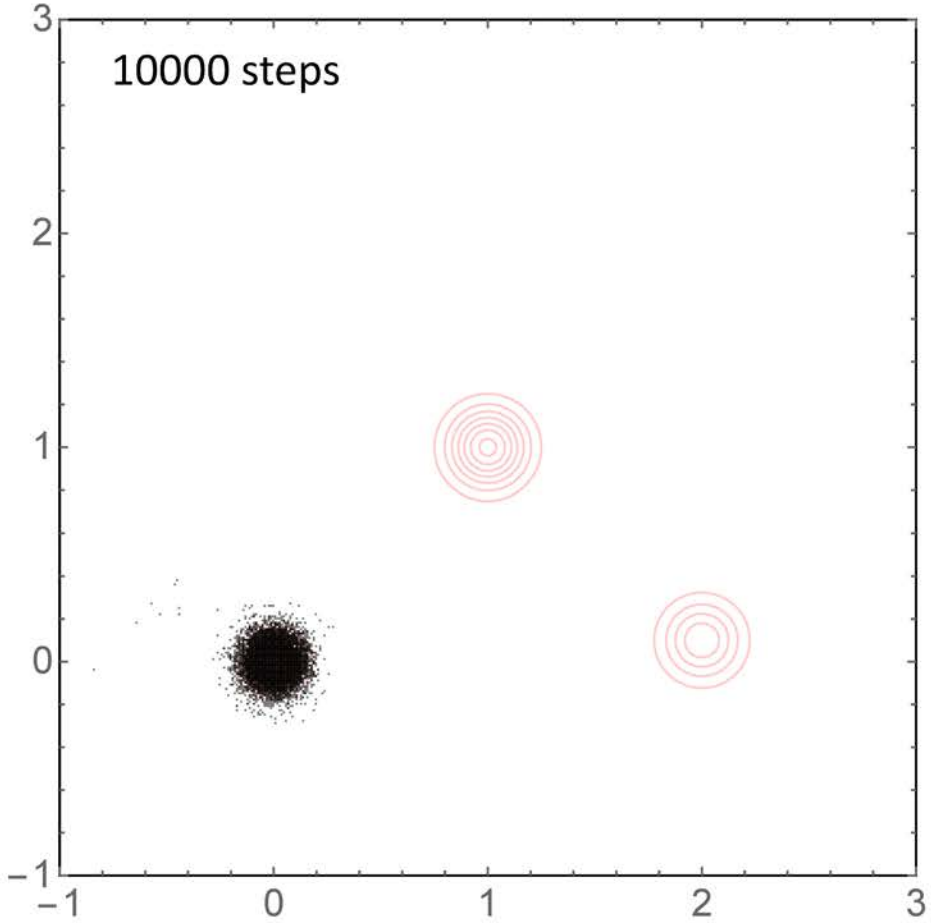


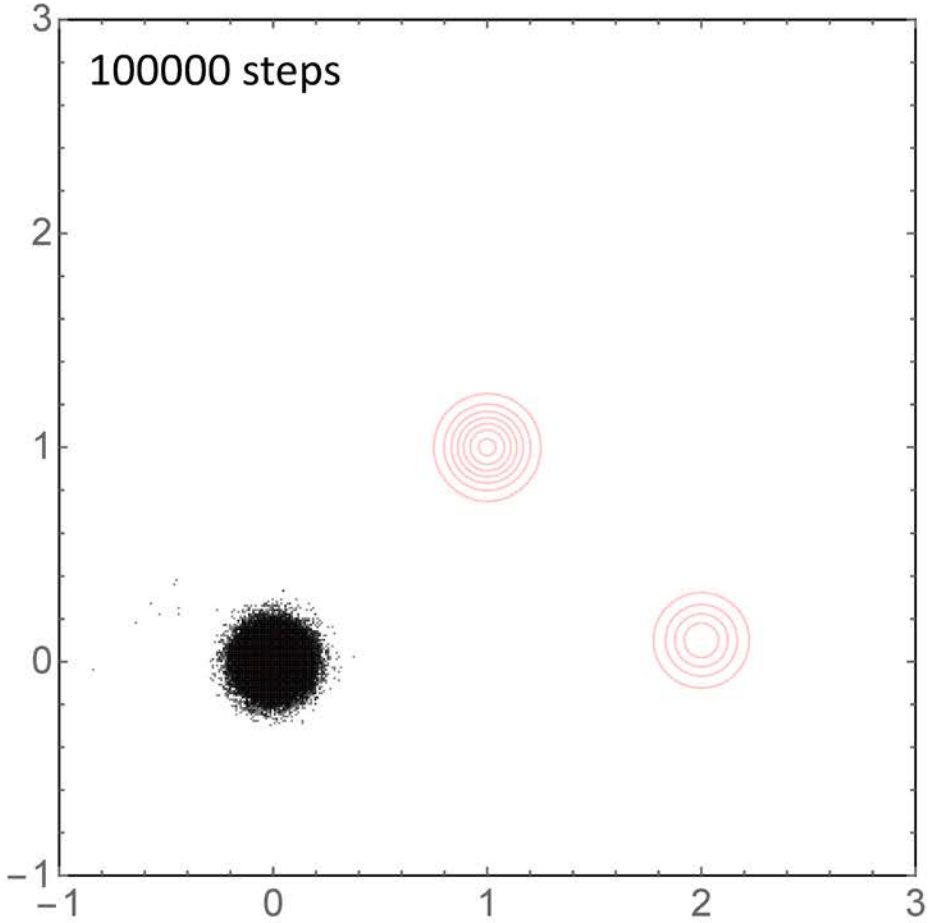
Notice that when the peaks are very narrow, the random walker may have problems visiting all of the peaks

$$p(x,y) = \sum_{i=1}^{3} \frac{\alpha_i}{\sqrt{2\sigma_i^2}} \exp\left[-\frac{(x-\mu_{x,i})^2 + (y-\mu_{y,i})^2}{2\sigma_i}\right]$$

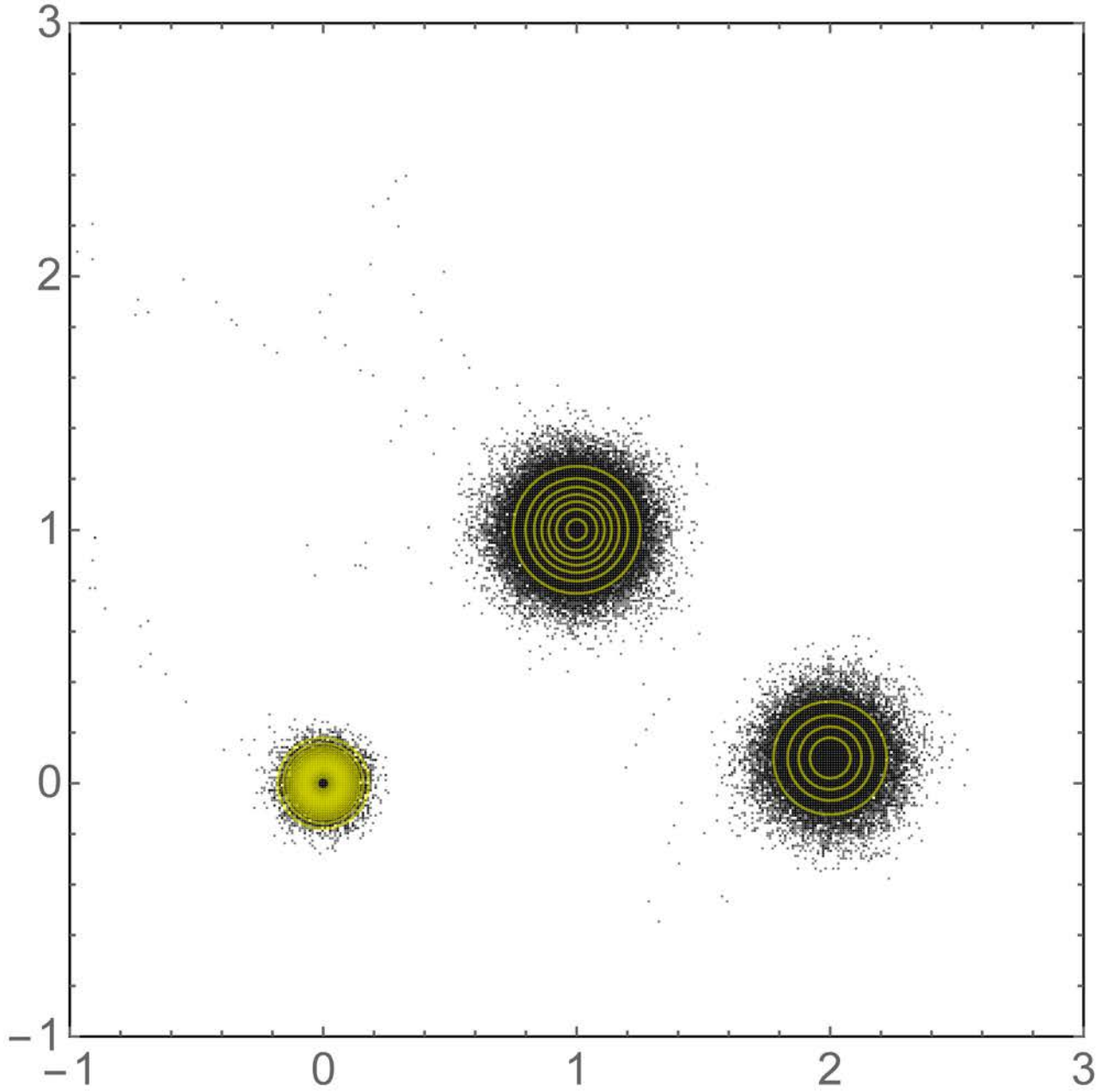
$$\begin{array}{llll} \alpha_1 = 0.5; & \mu_{x,1} = 0; & \mu_{y,1} = 0; & \sigma_1 = 0.0725; \\ \alpha_2 = 0.3; & \mu_{x,2} = 1; & \mu_{y,2} = 1.; & \sigma_2 = 0.125; \\ \alpha_3 = 0.2; & \mu_{x,3} = 2; & \mu_{y,3} = 0.1; & \sigma_3 = 0.125; \end{array}$$

With isolated, narrow peaks, increasing the number of steps may not suffice



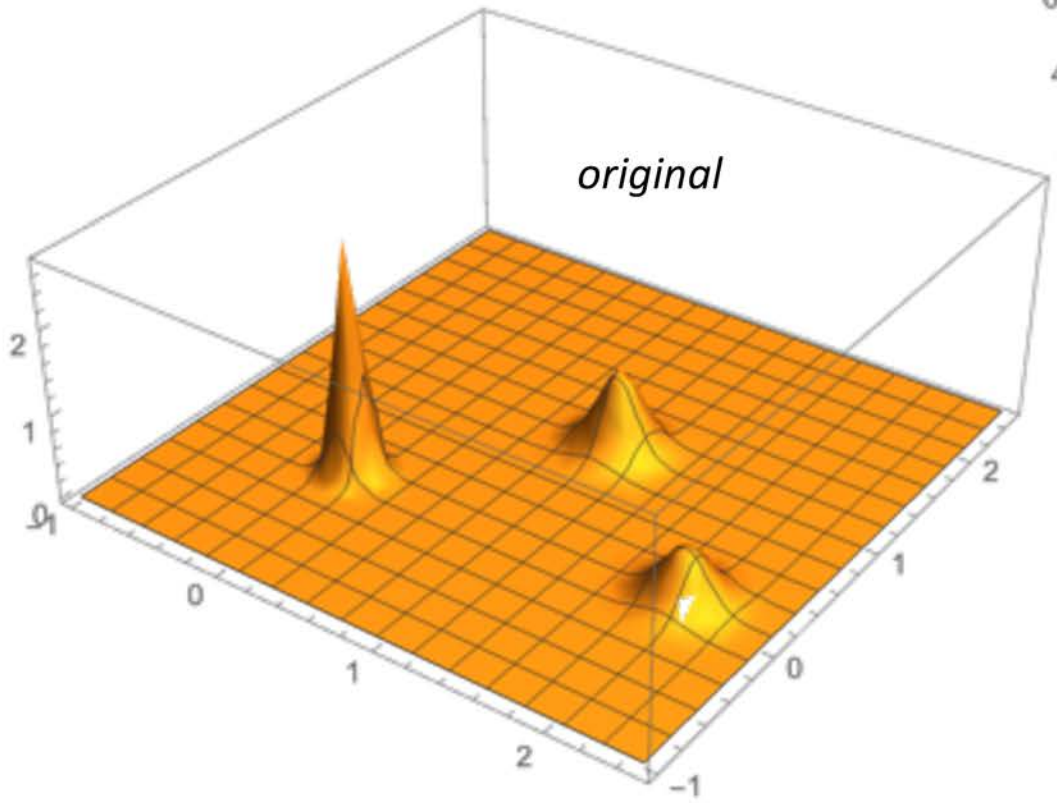


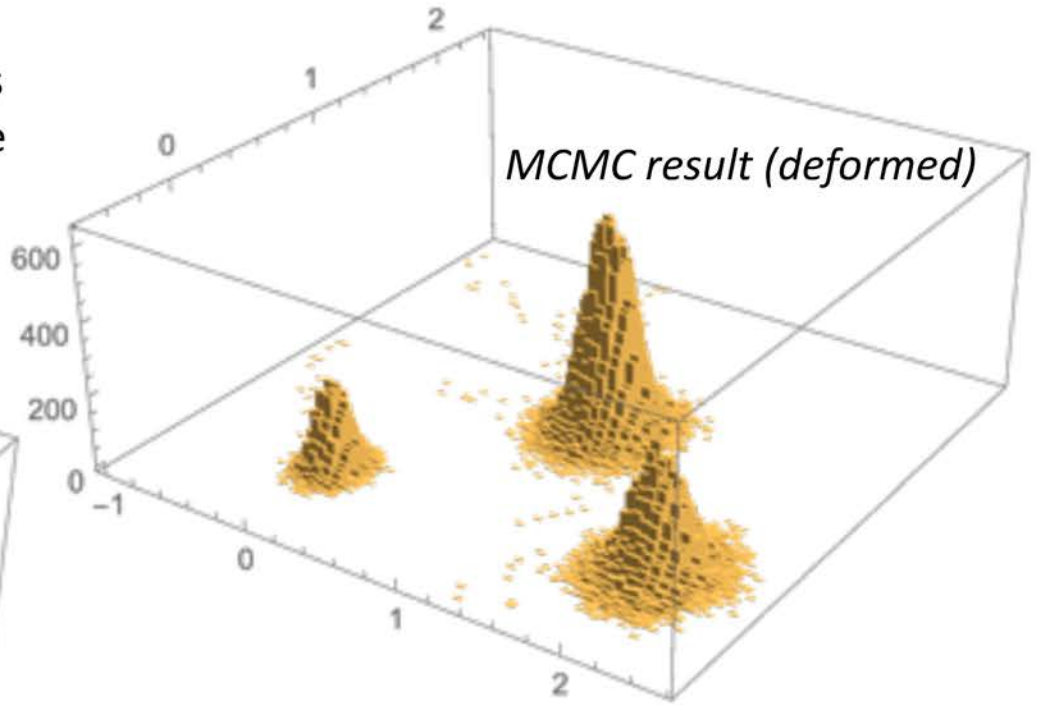
100000 steps, subdivided into 10 parallel chains with random starting points



The starting points of the chains are uniformly distributed in the plot region, however the "regions of influence" of each peak vary considerably.

This leads to more chains being attracted into the lower peaks, with the result that the distribution is somewhat deformed (wrong alpha's in the mixture model)





Many techniques have been developed to avoid these pitfalls

Convergence of the MCMC sequence to the asymptotic distribution

Statistical Science
1992, Vol. 7, No. 4, 457–511

Inference from Iterative Simulation Using Multiple Sequences

Andrew Gelman and Donald B. Rubin

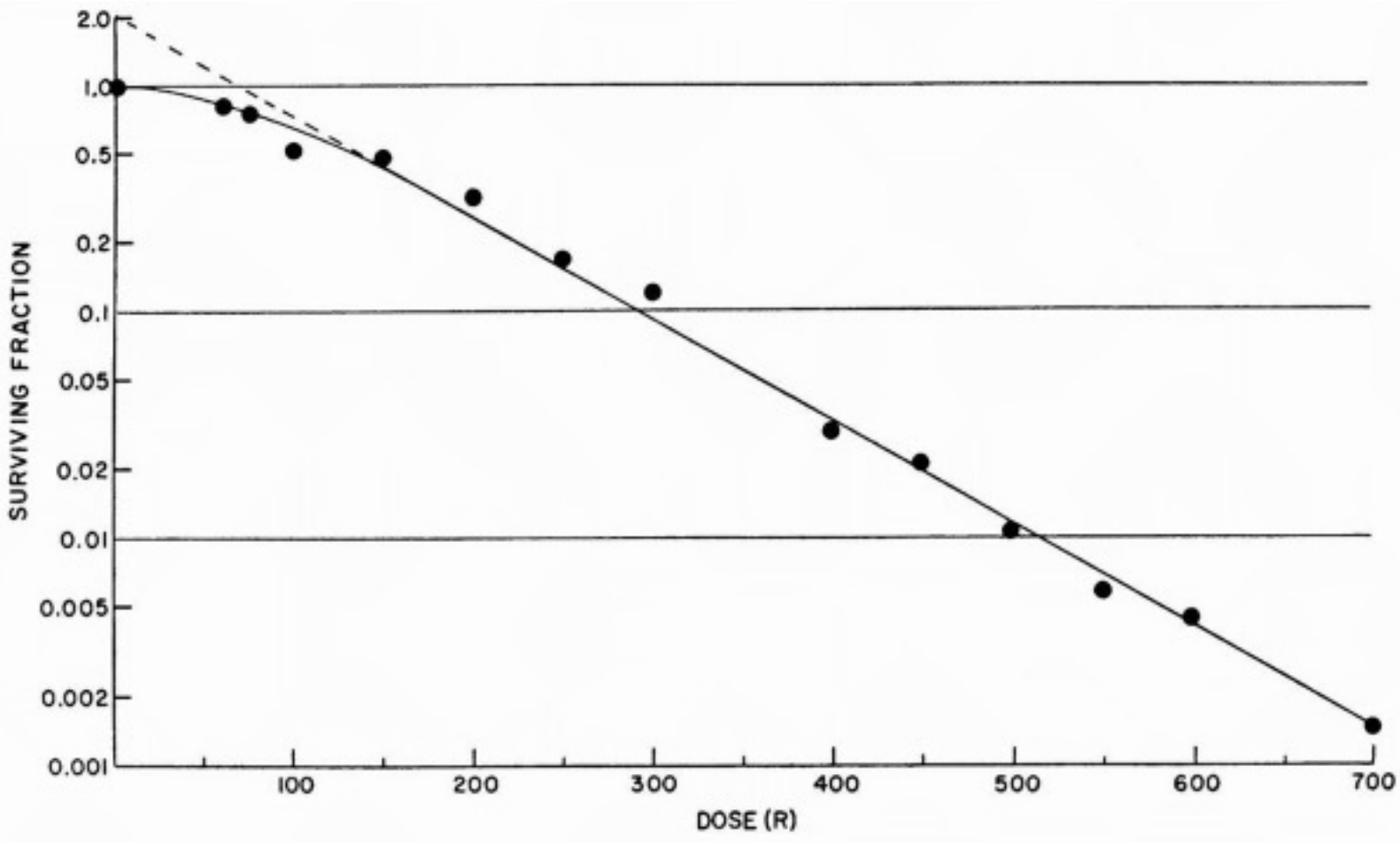
Abstract. The Gibbs sampler, the algorithm of Metropolis and similar iterative simulation methods are potentially very helpful for summarizing multivariate distributions. Used naively, however, iterative simulation can give misleading answers. Our methods are simple and generally applicable to the output of any iterative simulation; they are designed for researchers primarily interested in the science underlying the data and models they are analyzing, rather than for researchers interested in the probability theory underlying the iterative simulations themselves. Our recommended strategy is to use several independent sequences, with starting points sampled from an overdispersed distribution. At each step of the iterative simulation, we obtain, for each univariate estimand of interest, a distributional estimate and an estimate of how much sharper the distributional estimate might become if the simulations were continued indefinitely. Because our focus is on applied inference for Bayesian posterior distributions in real problems, which often tend toward normality after transformations and marginalization, we derive our results as normal-theory approximations to exact Bayesian inference, conditional on the observed simulations. The methods are illustrated on a random-effects mixture model applied to experimental measurements of reaction times of normal and schizophrenic patients.

Key words and phrases: Bayesian inference, convergence of stochastic processes, EM, ECM, Gibbs sampler, importance sampling, Metropolis algorithm, multiple imputation, random-effects model, SIR.

1.3 Our Approach

Our method is composed of two major steps. First, an estimate of the target distribution is created, centered about its mode (or modes, which are typically found by an optimization algorithm) and "overdispersed" in the sense of being more variable than the target distribution. The approximate distribution is then used to start several independent sequences of the iterative simulation. The second major step is to analyze the multiple sequences to form a distributional estimate of what is known about the target random variable, given the simulations thus far. This distributional estimate, which is in the form of a Student's t distribution for each scalar estimand, is somewhere between its starting and target distributions and provides the basis for an estimate of how close the simulation process is to convergence – that is, how much sharper the distributional estimate might become if the simulations were run longer.

Example of application of the MCMC technique in radiobiology



Survival curve for HeLa cells in culture exposed to x-rays. (From Puck TT, Markus Pl: Action of x-rays on mammalian cells. *J Exp Med* 103:653-666, 1956)

Phenomenology: the linear-quadratic law

$$S(D) \approx e^{-\alpha D - \beta D^2}$$

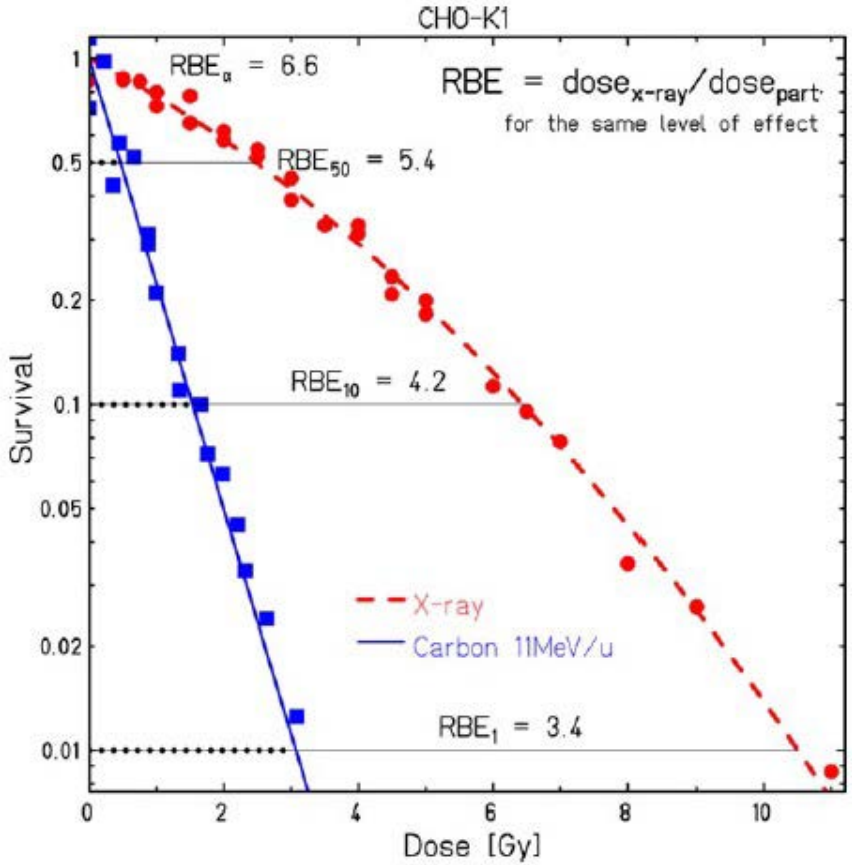


Fig. 1. Clonogenic survival curves illustrating the higher efficiency of the carbon ions compared with X-rays [10] (courtesy of the author, dr. Wilma K. Weyrather).

Target theory

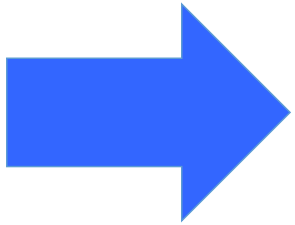
Simple Poisson model:

Probability of hitting *n* times a given target, when the average number of good hits is *a*:

$$P(n) = \frac{a^n}{n!} e^{-a}$$

Probability missing the target: $P(0) = e^{-a}$

Average number of hits: $a = D/D_0$



$$S(D) = P(0, D) = e^{-D/D_0}$$

Multitarget model, asymptotic behavior and threshold effect.

If there are multiple targets, say *n* targets, all of which must be hit to kill a cell, then the probability of missing at least one of them – i.e., the survival probability – is

$$S(D) = 1 - (1 - e^{-D/D_0})^n$$

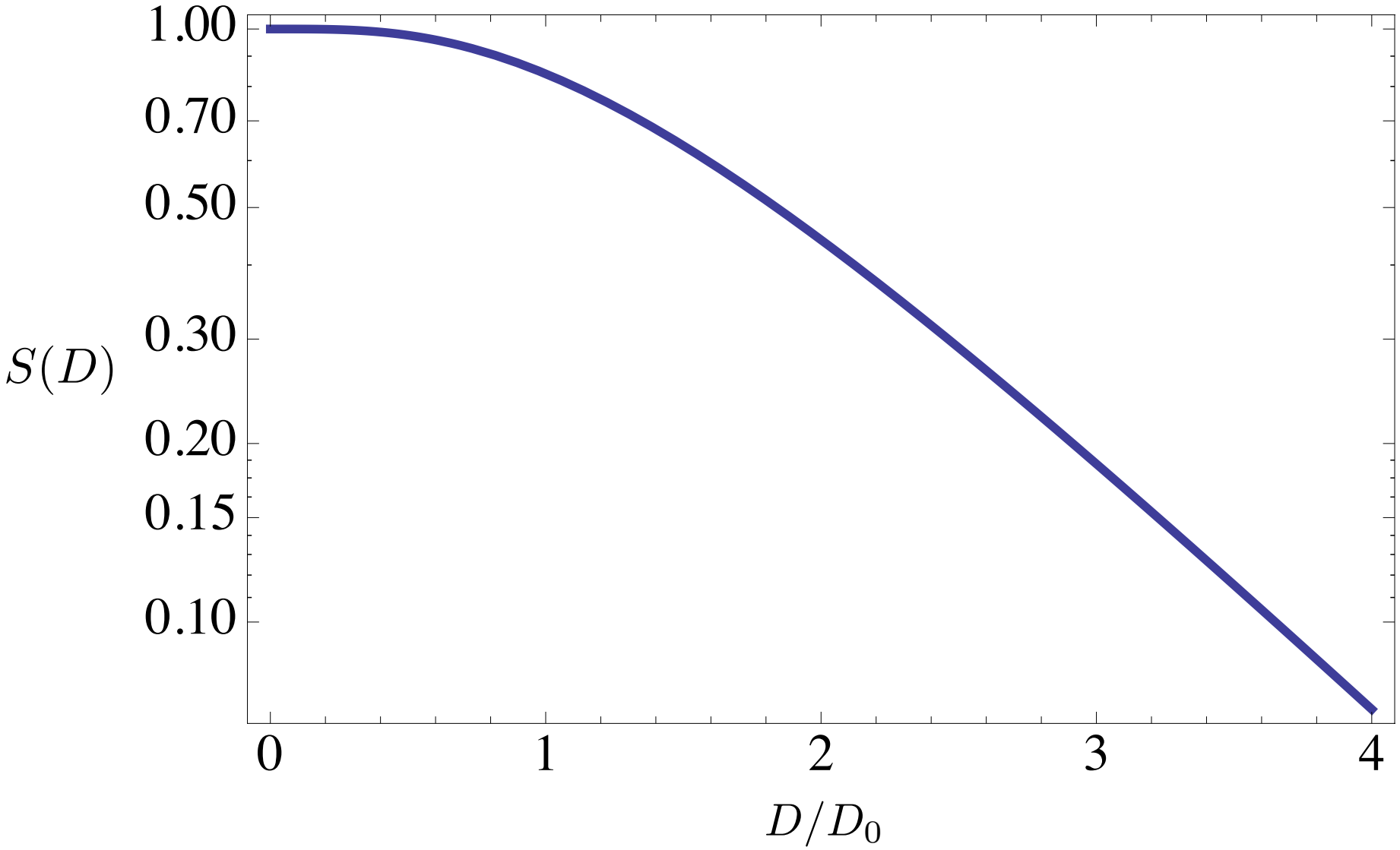
then, for large dose

$$S(D) \approx n e^{-D/D_0}$$

i.e.,

$$\ln S(D) \approx \ln n - D/D_0$$

which is a linear relation with intercept ln *n*, and slope -1/D_0.


1.00
0.70
0.50
0.30
0.20
0.15
0.10
$S(D)$
0
1
2
3
4
D/D_0

Notice that

$$\left[\frac{d}{dD}e^{-\alpha D-\beta D^2}\right]_{D=0} = (-\alpha - 2\beta D)e^{-\alpha D-\beta D^2}\Big|_{D=0} = -\alpha$$

and that

$$\frac{d}{dD}\left[1-(1-e^{-D/D_0})^n\right]_{D=0} = -n\,\frac{e^{-D/D_0}}{D_0}(1-e^{-D/D_0})^{n-1}\Big|_{D=0} = 0$$

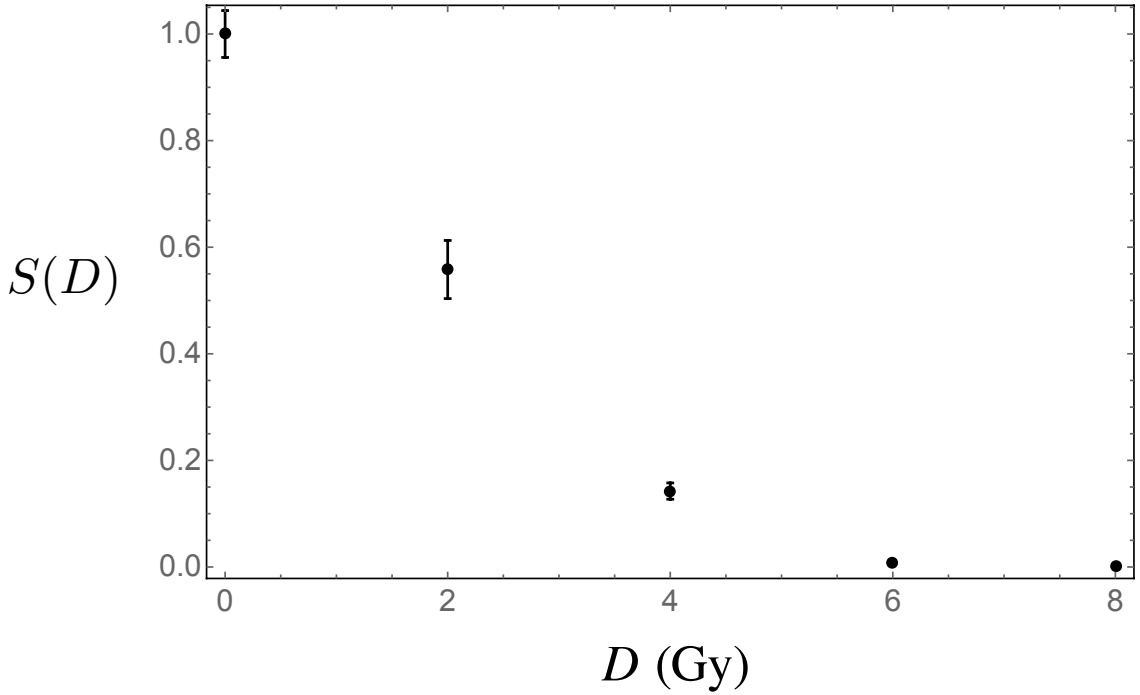
The derivatives differ in the origin, and the multitarget model fails to reproduce the observed linear-quadratic law.

The RCR (Repairable-Conditionally Repairable Damage) model

In this case the surviving fraction is

$$S = \exp(-aD) + bD\exp(-cD)$$

This is a 3-parameter expression, which is not easy to fit to data when the data set is small.



1a. Simple Gaussian likelihood for the LQ model

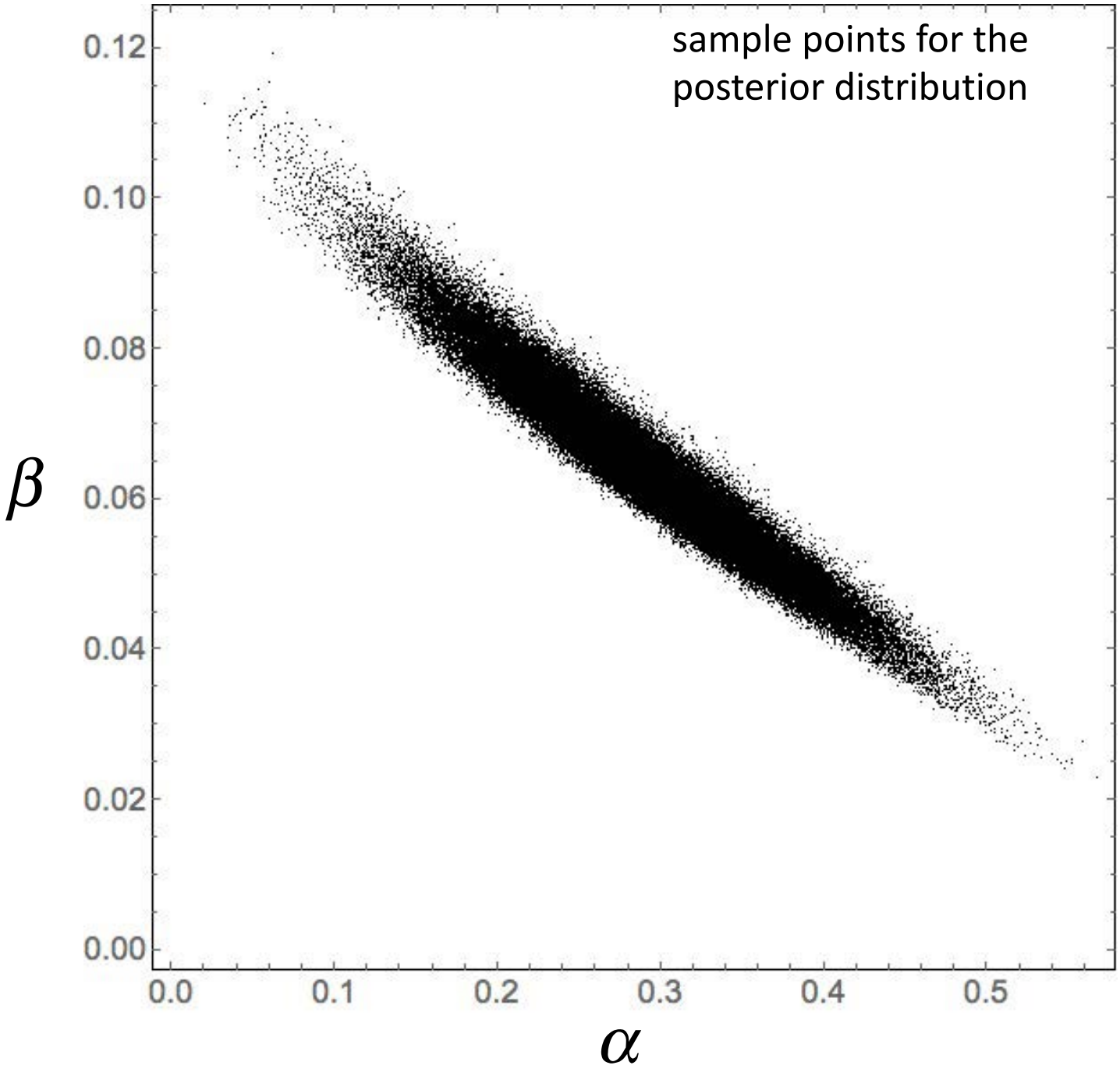
$$L(\alpha,\beta)=\prod_k \exp\left(-\frac{\left(S_k-S(\alpha,\beta)\right)^2}{2\sigma_k^2}\right)$$

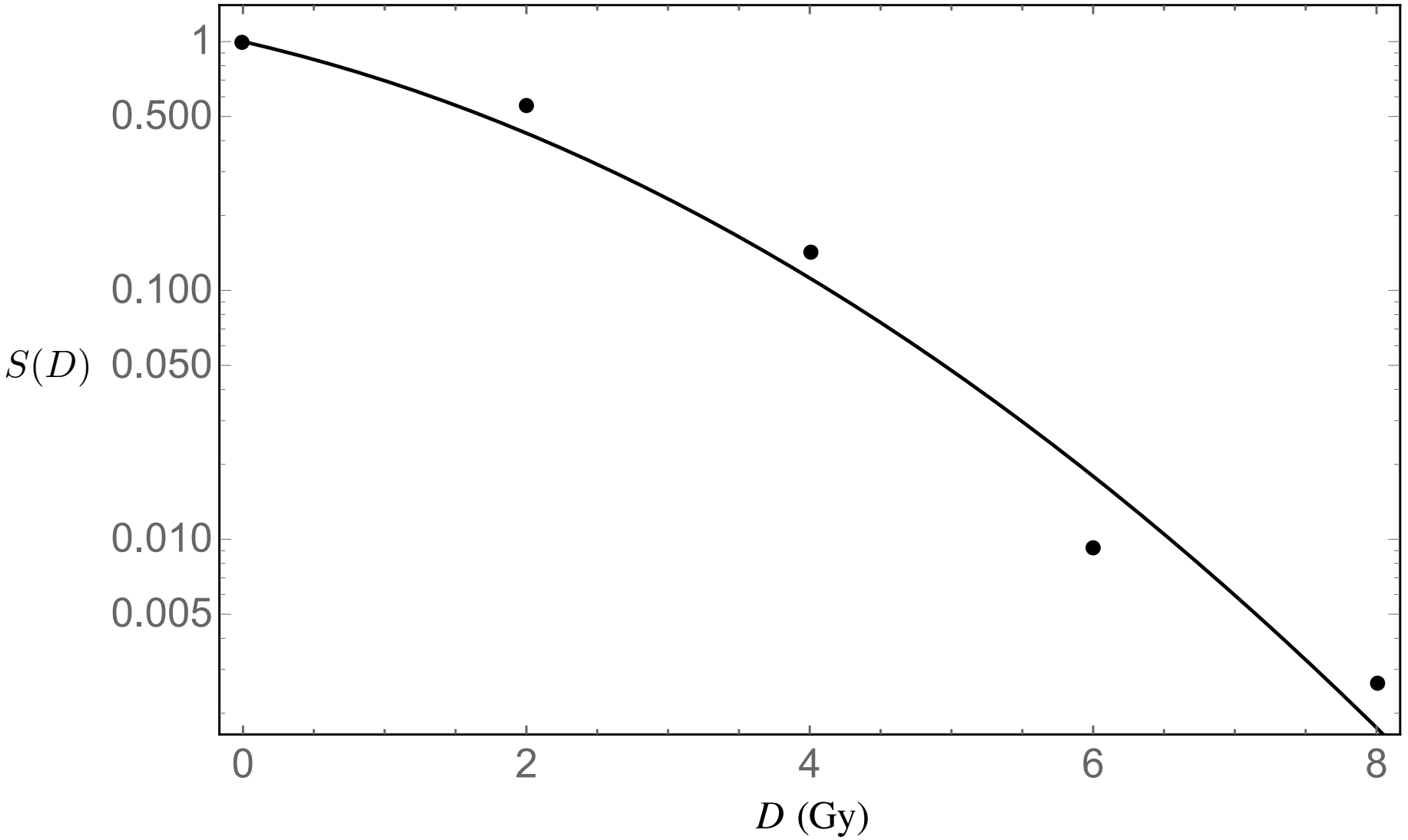
1b. Chose exponential priors for the parameters

1c. Complete posterior pdf

$$p\left(\alpha,\beta\middle|\{S_k\},I\right)=\left[\prod_k \exp\left(-\frac{\left(S_k-S(\alpha,\beta)\right)^2}{2\sigma_k^2}\right)\right]\exp(-0.1\alpha)\exp(-0.1\beta)$$

1d. Use MCMC to find the MAP estimate (and any moment of the pdf)


sample points for the
posterior distribution
0.12
0.10
0.08
0.06
0.04
0.02
0.00
β
0.0
0.1
0.2
0.3
0.4
0.5
α


1
0.500
0.100
0.050
0.010
0.005
S(D)
0
2
4
6
8
D (Gy)

2a. Simple Gaussian likelihood for the RCR model

$$L(a,b,c)=\prod_k \exp\left(-\frac{\left(S_k-S(a,b,c)\right)^2}{2\sigma_k^2}\right)$$

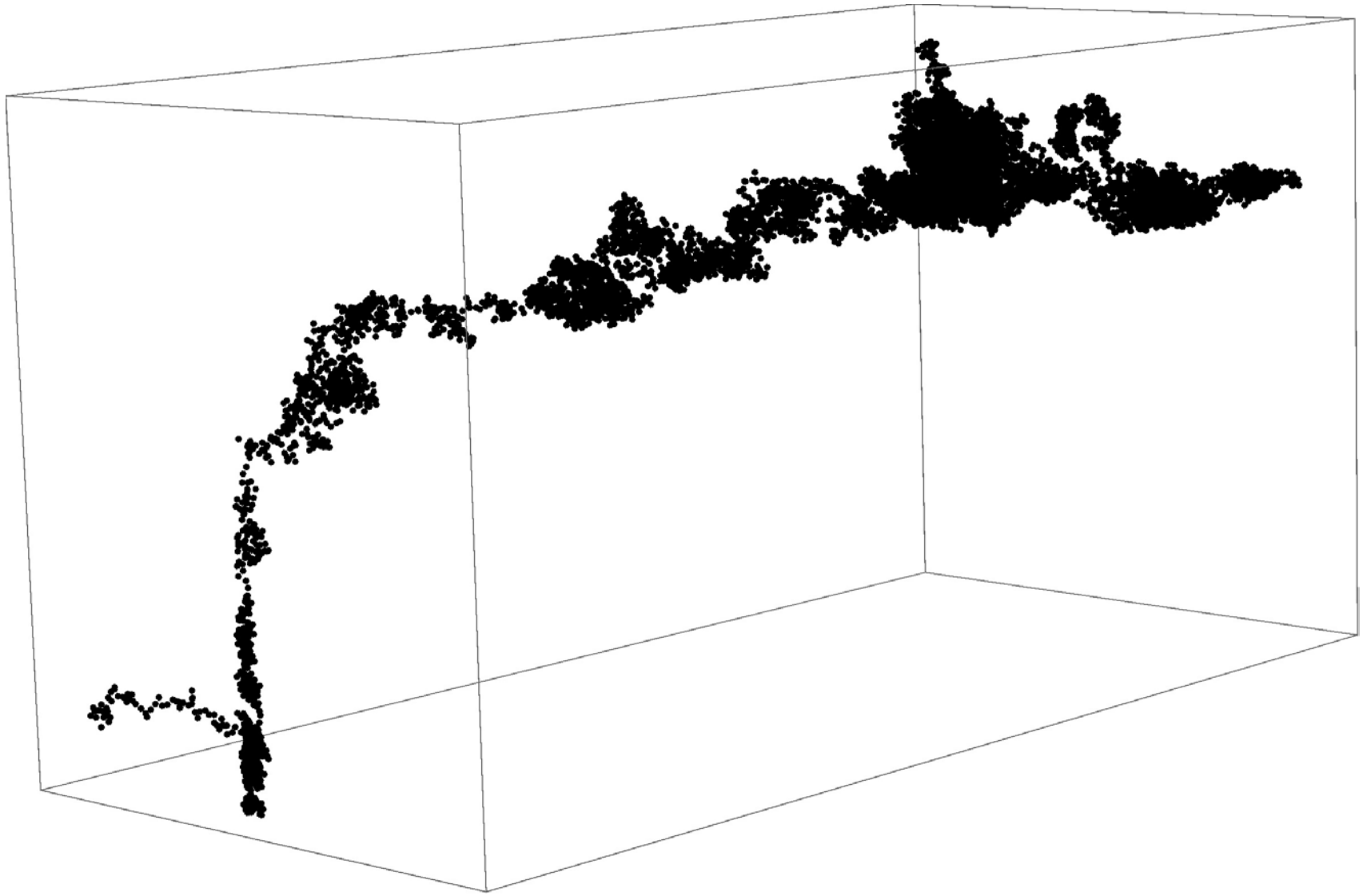
2b. Chose exponential priors for the parameters

2c. Complete posterior pdf

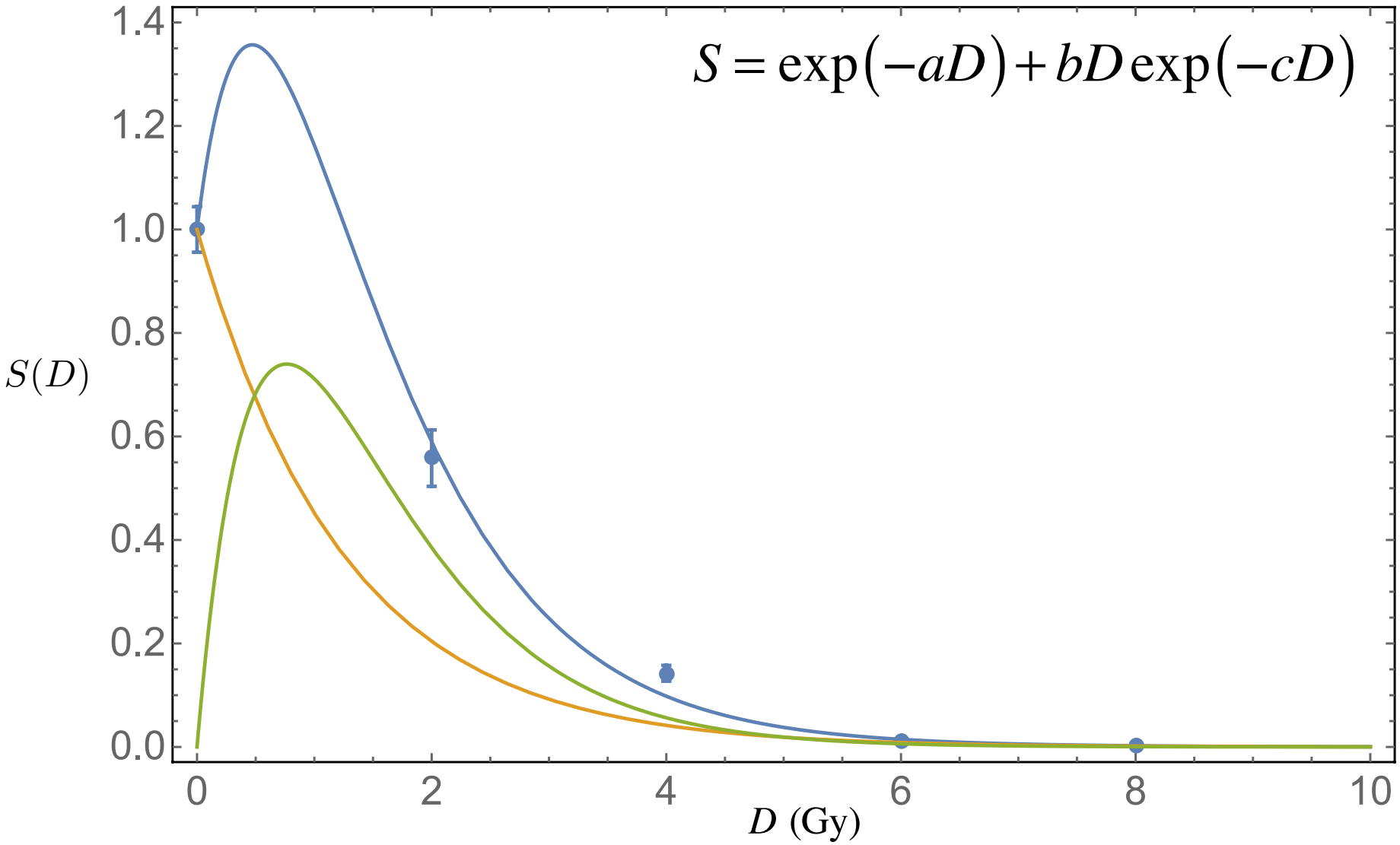
$$p\left(a,b,c\middle|\{S_k\},I\right)=\left[\prod_k \exp\left(-\frac{\left(S_k-S(a,b,c)\right)^2}{2\sigma_k^2}\right)\right]e^{-0.2a}e^{-0.2b}e^{-0.2c}$$

2d. Use MCMC to find the MAP estimate (and any moment of the pdf)

Path in (a,b,c) space

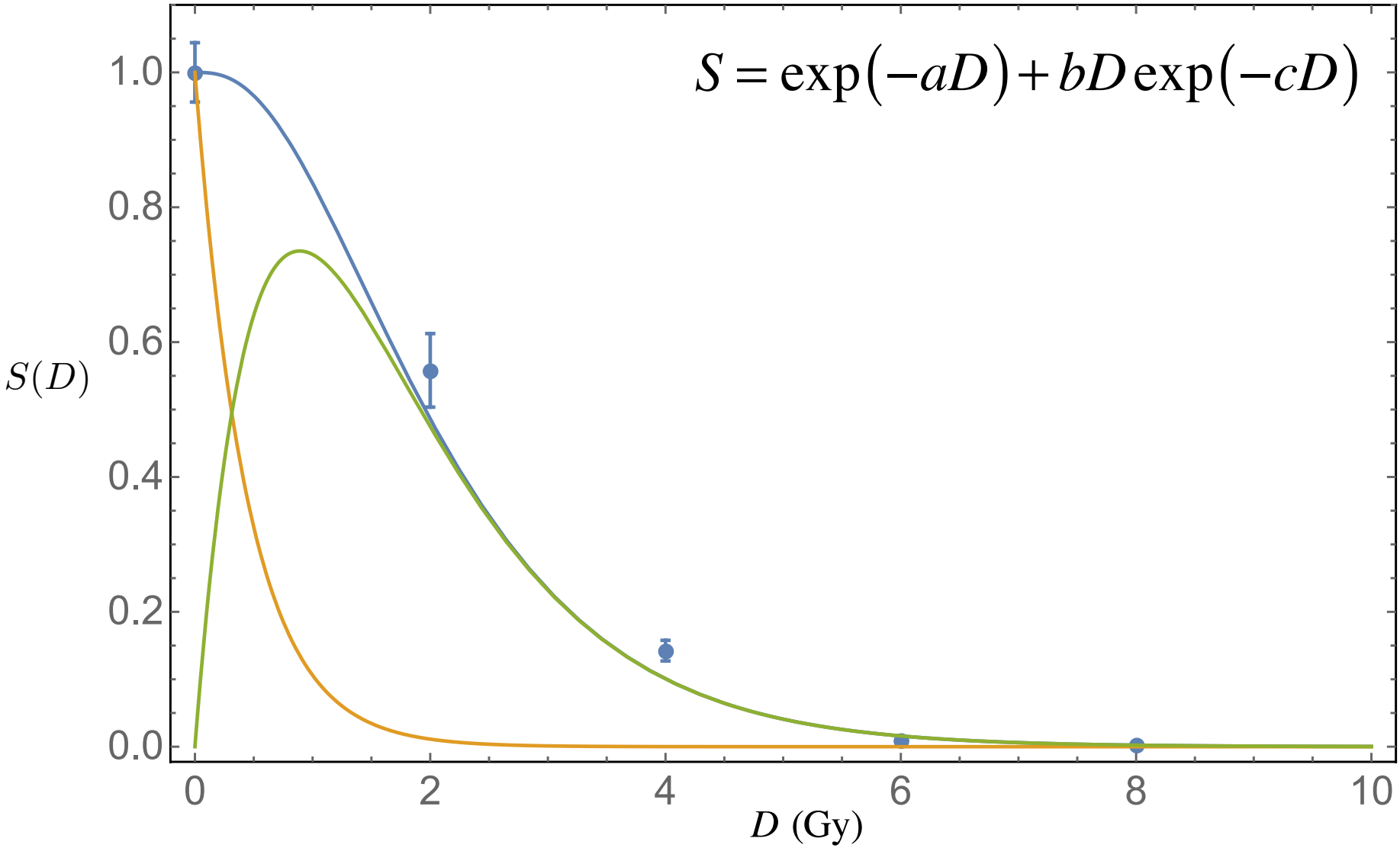


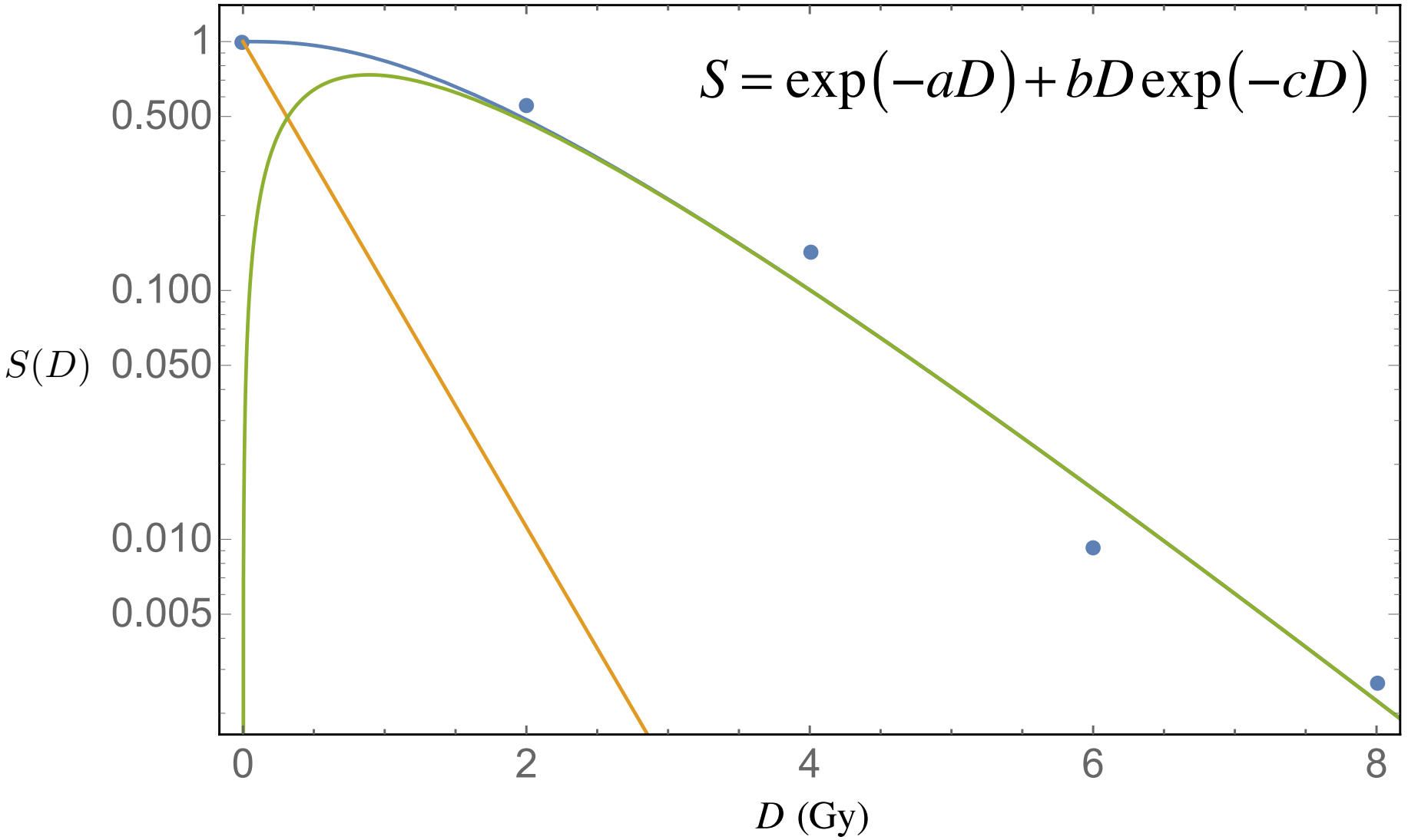
Fit showing individual components: unsatisfactory result



Revise priors to include constraint on derivative

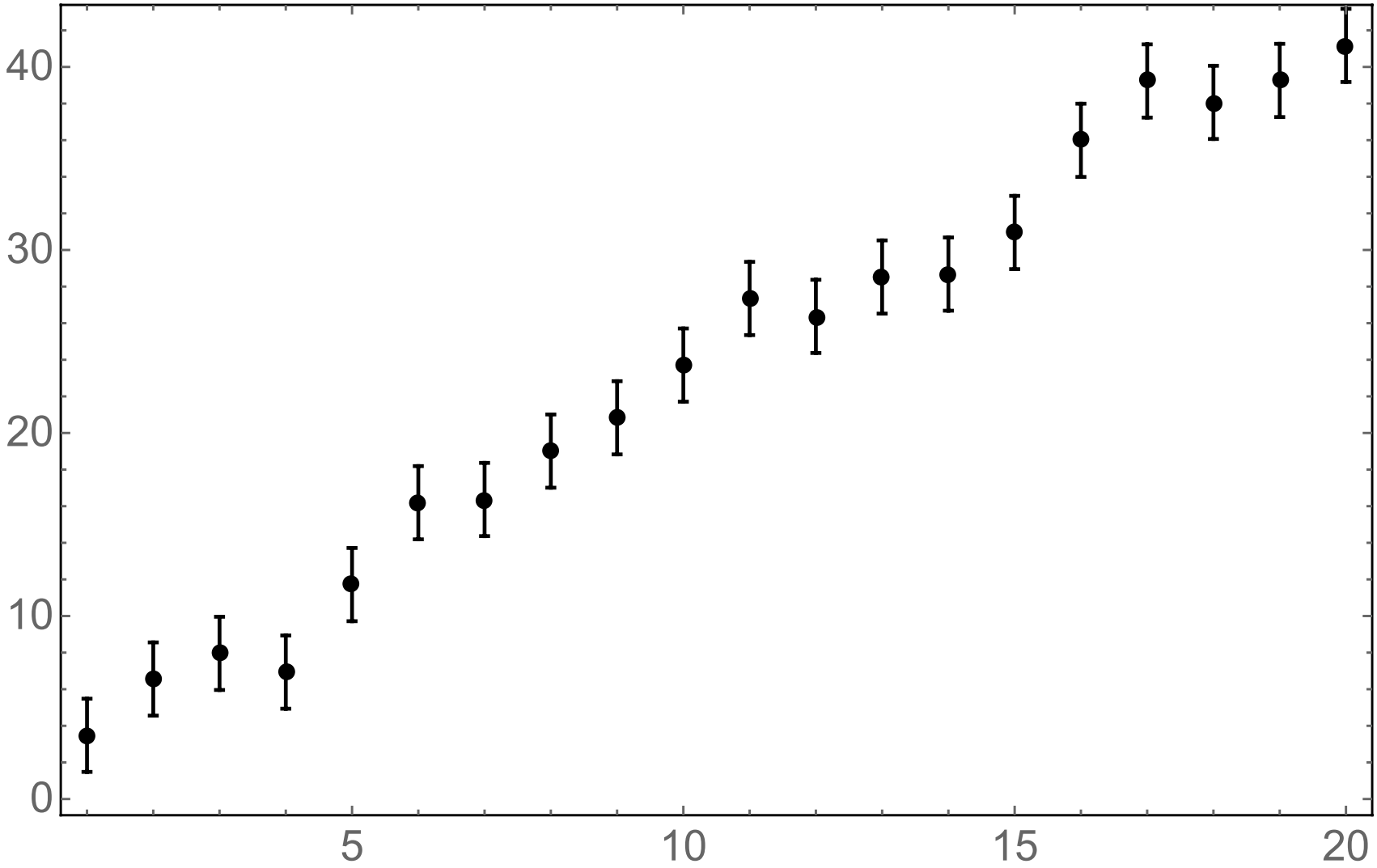
(priors vanish where derivative in the origin is positive)



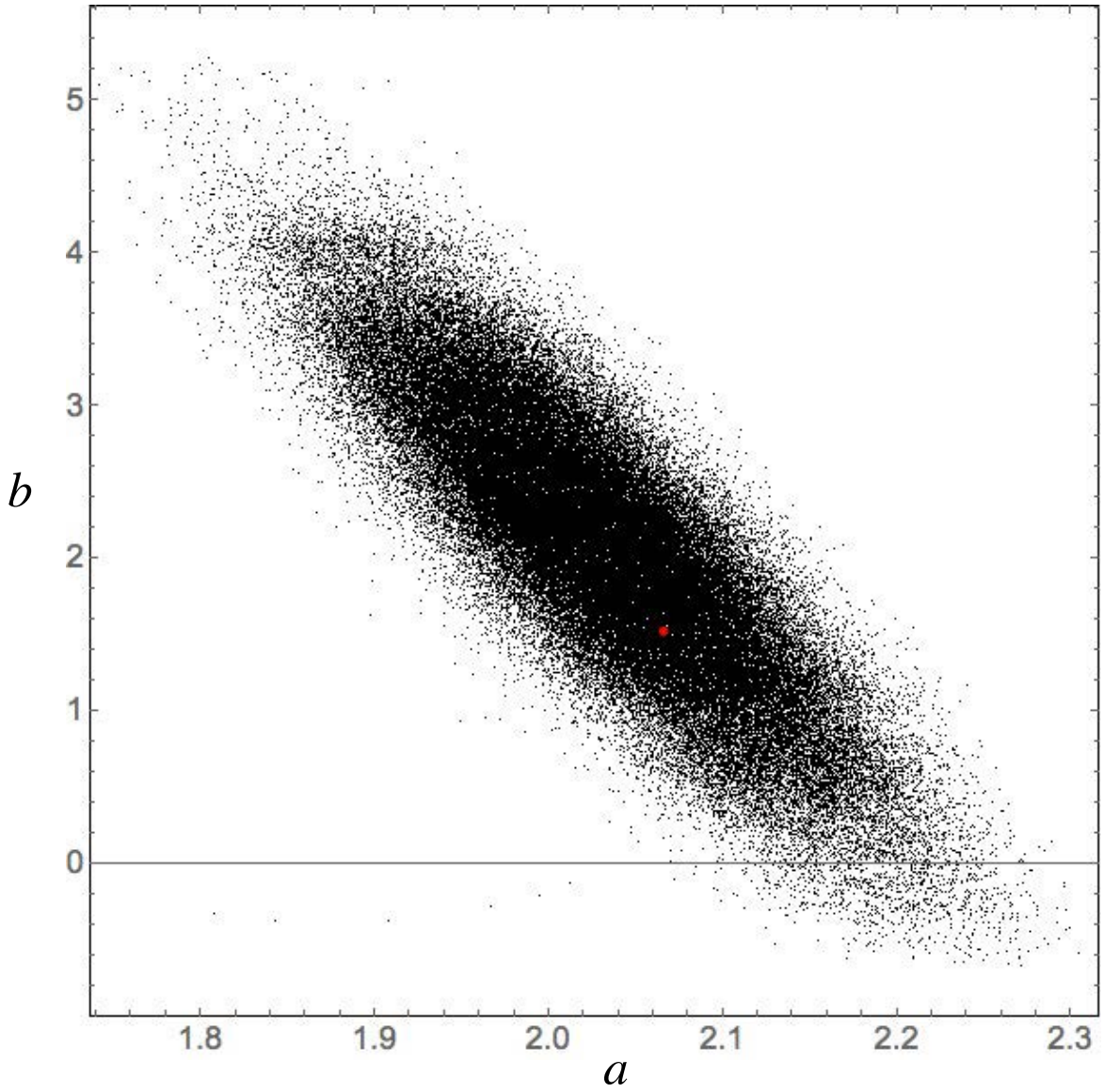

$S = \exp(-aD) + bD\exp(-cD)$
1
0.500
0.100
0.050
0.010
0.005
$S(D)$
0
2
4
6
8
D (Gy)

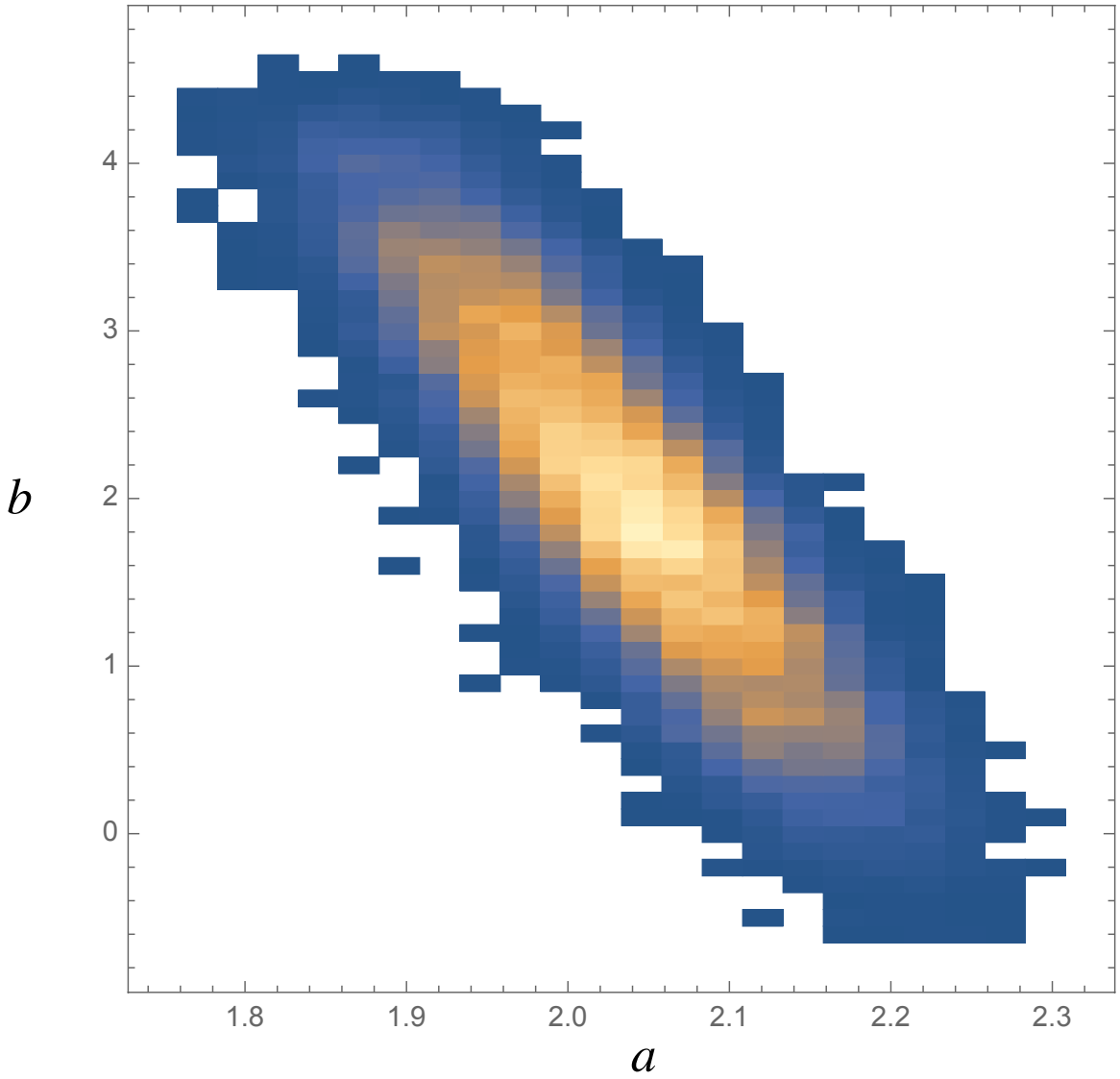
“Straight line fit” with the MCMC

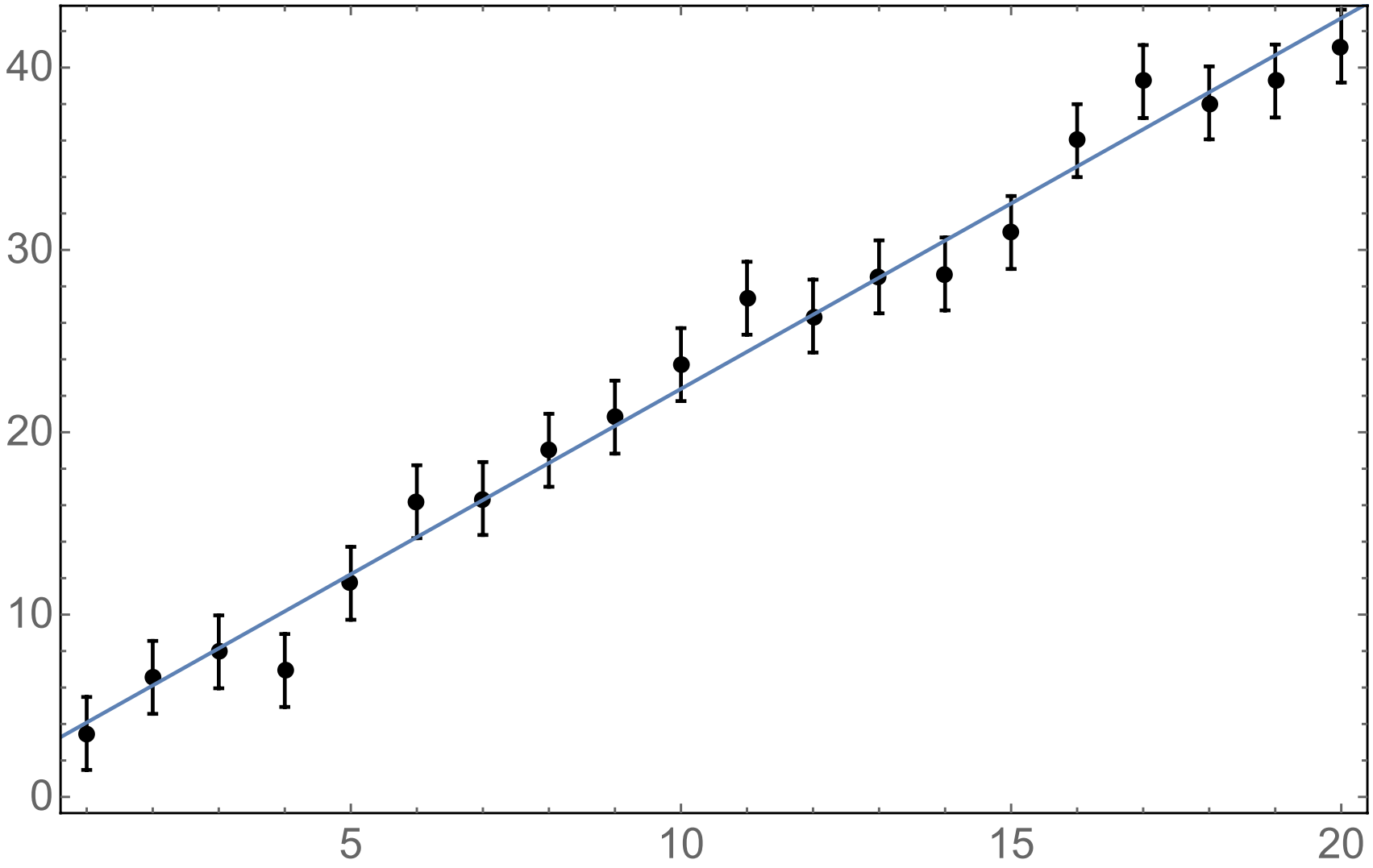
An example with Gaussian errors and exponential priors.



model $y = ax + b$




b
4
3
2
1
0
1.8
1.9
2.0
2.1
2.2
2.3
a


40
30
20
10
0
5
10
15
20

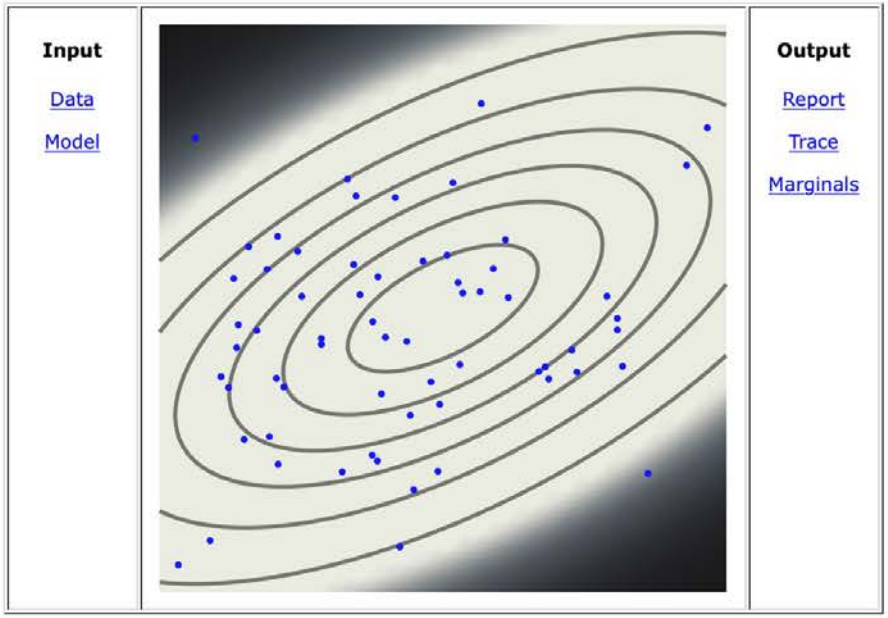
MacMCMC (v1.5)

State-of-the-art Data Analysis for Mac OS X™

MacMCMC is a free and extremely powerful application for the analysis of data of any kind. It is one half of a two-part project. The other half is a free ebook—a strongly recommended preliminary—available here.

To see *MacMCMC* in action, consider this famous example from the literature (Arnold and Libby, 1949):

Carbon-14 Dating



Given the *MacMCMC* report, any graphing software may be used to prepare a plot showing model versus data.
Note: The blue line in this plot uses mean estimates; the red line shows the prior uncertainty for parameter A.

Principal Features

General

- Complete, standalone Mac application
- 100% Bayesian inference
- 100% ensemble MCMC
- Access to low-level options
- Parallelized for maximum speed

https://causascientia.org/software/MacMCMC/MacMCMC.html

Age Determinations by Radiocarbon Content: Checks with Samples of Known Age

J. R. Arnold and W. F. Libby

Institute for Nuclear Studies, University of Chicago, Chicago, Illinois

FURTHER TESTS of the radiocarbon method of age determination (*1–3, 6, 8, 10*) for archaeological and geological samples have been completed. All the samples used were wood dated quite accurately by accepted methods. The measurement technique consisted in the combustion of about 1 ounce of wood, the collection of the carbon dioxide, its reduction to elementary carbon with hot magnesium metal, and the measurement of 8 grams of this carbon spread uniformly over the 400-square-centimeter surface of the sample cylinder in a screen wall counter (*7, 9*). The background count was reduced during the latter part of the work to 7.5 counts per minute (cpm), which is some 2 percent of the unshielded background, by the use of 4 inches of iron inside 2 inches of lead shielding, plus 11 anticoincidence counters 2 inches in diameter and 18 inches long, placed symmetrically around the working screen wall counter inside the shielding. The screen wall counter had a sensitive portion 8 inches in length, so the long anticoincidence shielding counters afforded considerable protection on the ends. No end counters were used. The data obtained are presented in Table 1 and Fig. 1.

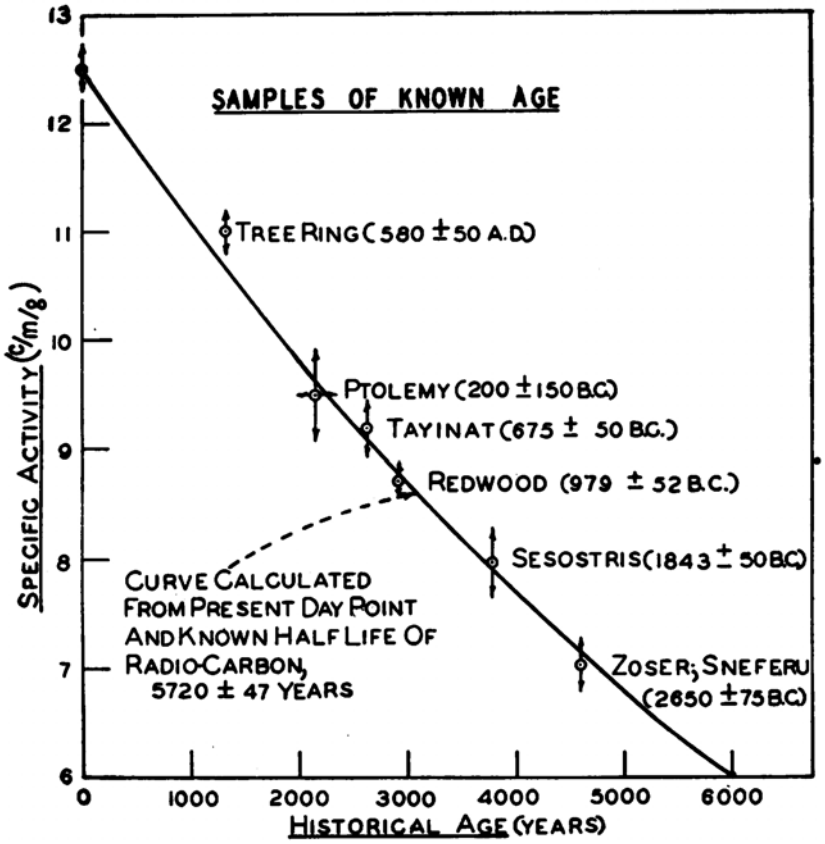
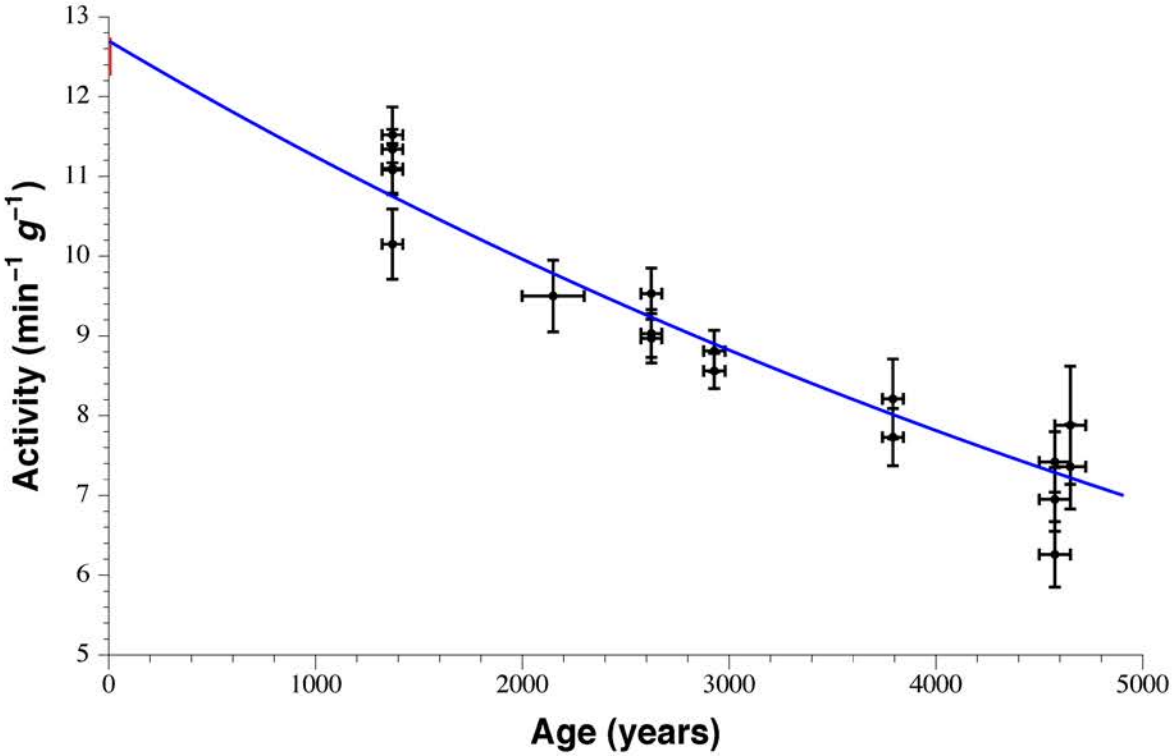


FIG. 1. Specific activities for samples of known age.

Test run with MacMCMC




```
Data: C14.dat   Model: C14.mcmc   3 May 2020 at 19:48:53

# chains x sample/chain: 300 x 3334 = 1000200 (thinning = 10)

log(marginal likelihood): -183.208

A
MAP, Mean, Median, Mode, G-R stat: 12.6865 12.6952 12.6966 12.6817 1.003
Credible Intervals: 12.4079 12.4877 12.5239 12.8679 12.9015 12.978

h
MAP, Mean, Median, Mode, G-R stat: 5710.08 5708.23 5708.35 5707.39 1.013
Credible Intervals: 5587.2 5616.54 5630.98 5785.18 5800.11 5828.53
```

https://docs.pymc.io

Friendly modelling API

PyMC3 allows you to write down models using an intuitive syntax to describe a data generating process.

Cutting edge algorithms and model building blocks

Fit your model using gradient-based MCMC algorithms like NUTS, using ADVI for fast approximate inference — including minibatch-ADVI for scaling to large datasets — or using Gaussian processes to build Bayesian nonparametric models.

```
import pymc3 as pm

X, y = linear_training_data()
with pm.Model() as linear_model:
    weights = pm.Normal("weights", mu=0, sigma=1)
    noise = pm.Gamma("noise", alpha=2, beta=1)
    y_observed = pm.Normal(
        "y_observed",
        mu=X @ weights,
        sigma=noise,
        observed=y,
    )

    prior = pm.sample_prior_predictive()
    posterior = pm.sample()
    posterior_pred = pm.sample_posterior_predictive(posterior)
```

•*References:*

• H. Varian, "Bootstrap Tutorial", The Mathematica Journal **9** (2005) 768

•A. F. M. Smith and A. E. Gelfand: "Bayesian Statistics Without Tears: A Sampling-Resampling Perspective", Am. Stat. **46** (1992) 84

• B. Walsh: "Markov Chain Monte Carlo and Gibbs Sampling", http://web.mit.edu/~wingated/www/introductions/mcmc-gibbs-intro.pdf

• S. P. Brooks: "Markov Chain Monte Carlo and Its Application", The Statistician **47** (1998) 69

• D. Hogg and D. Foreman-Mackey: "Data Analysis Recipes: Using Markov Chain Monte Carlo", ApJ Suppl. Ser. **236** (2018) 1 https://iopscience.iop.org/article/10.3847/1538-4365/aab76e