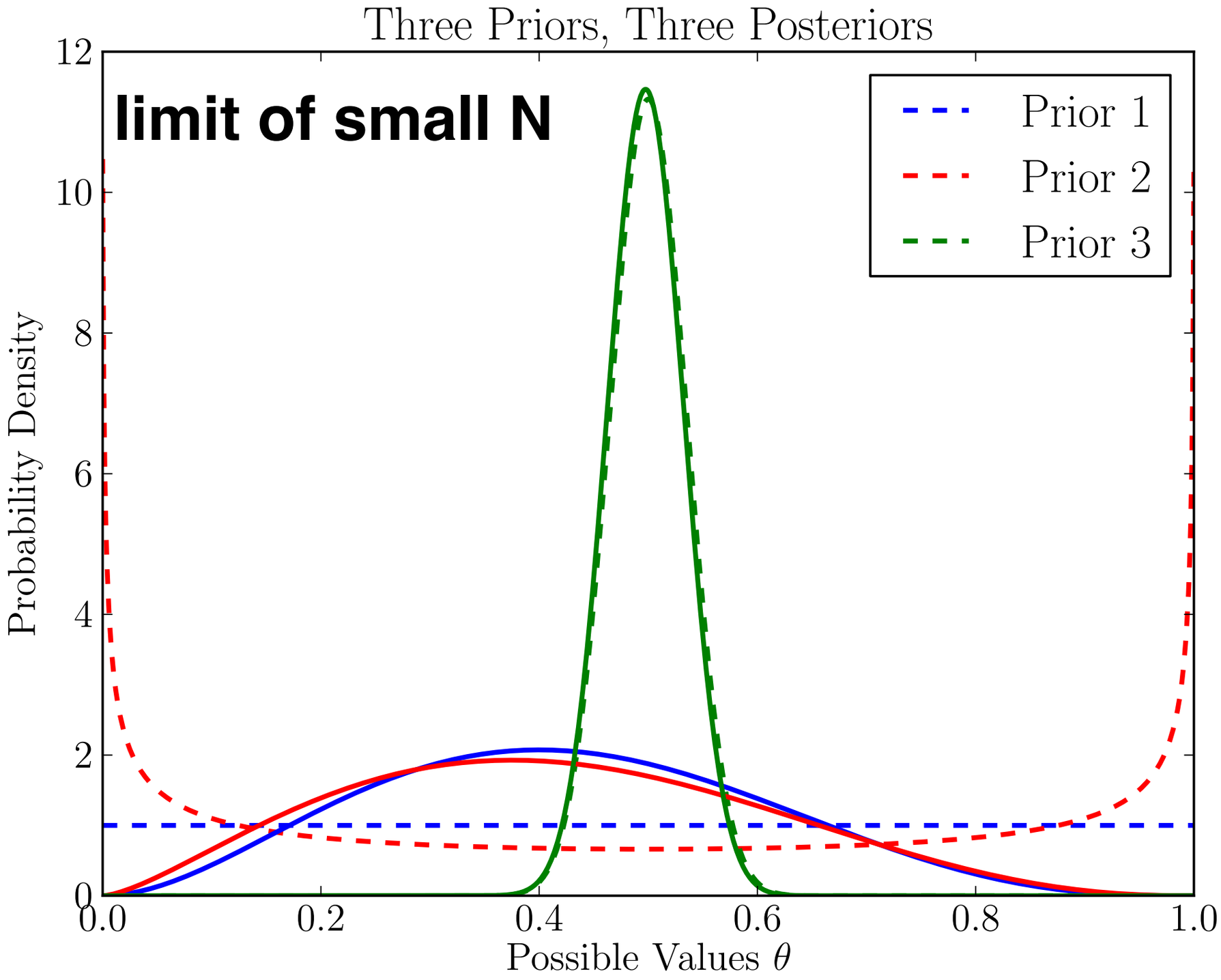
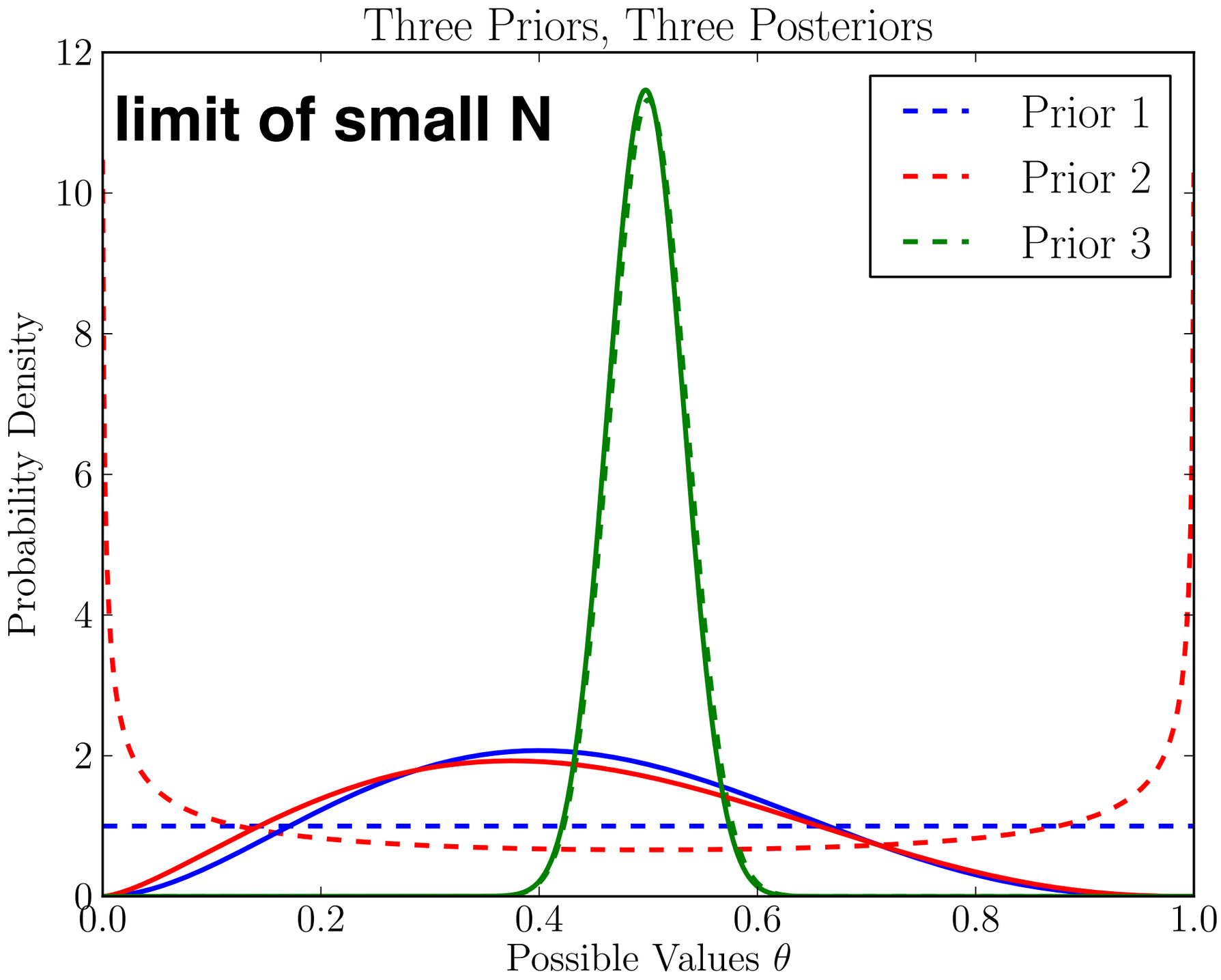
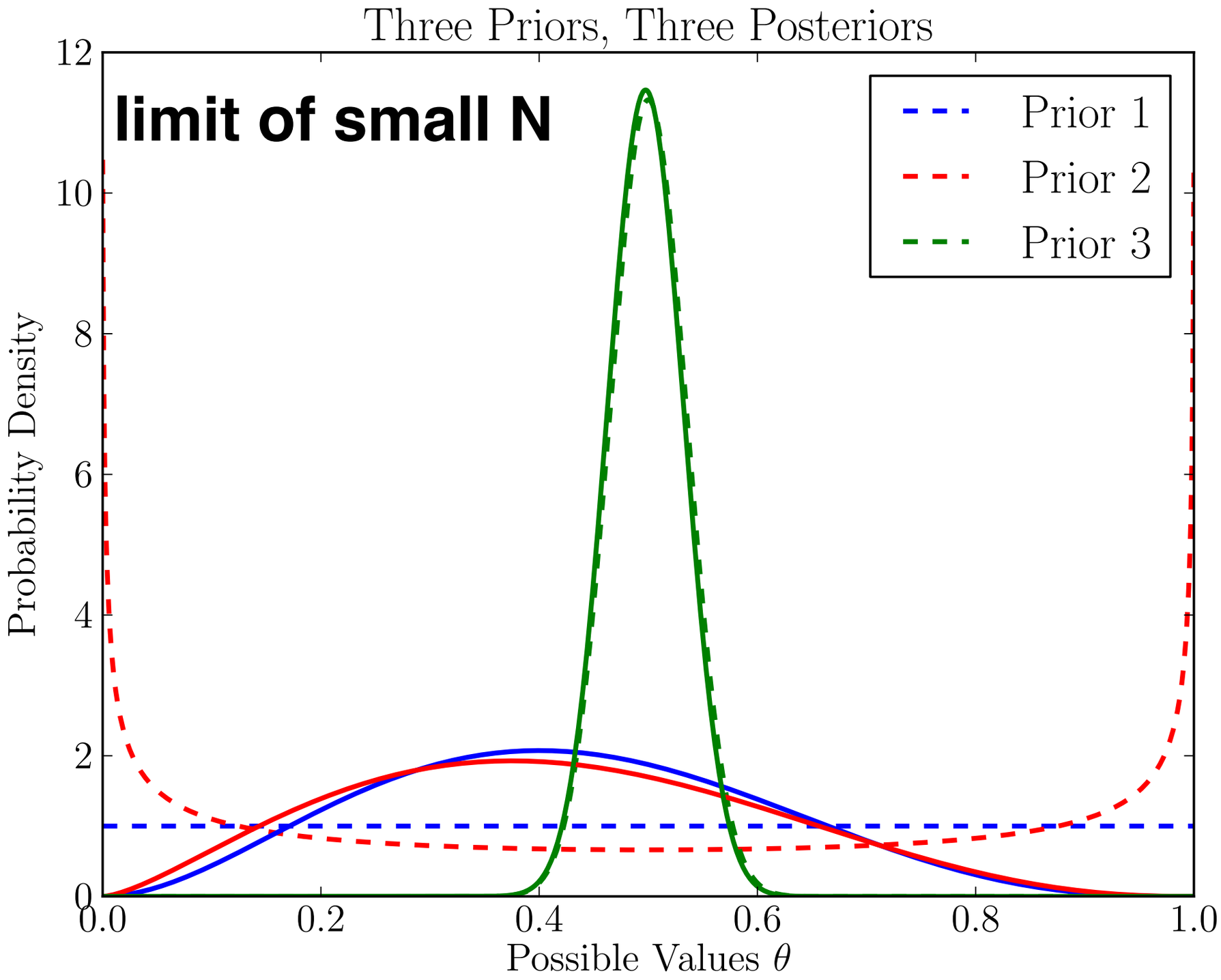

Three Priors, Three Posteriors
limit of small N
Prior 1
Prior 2
Prior 3
Probability Density
Possible Values θ
0
2
4
6
8
10
12
0.0
0.2
0.4
0.6
0.8
1.0

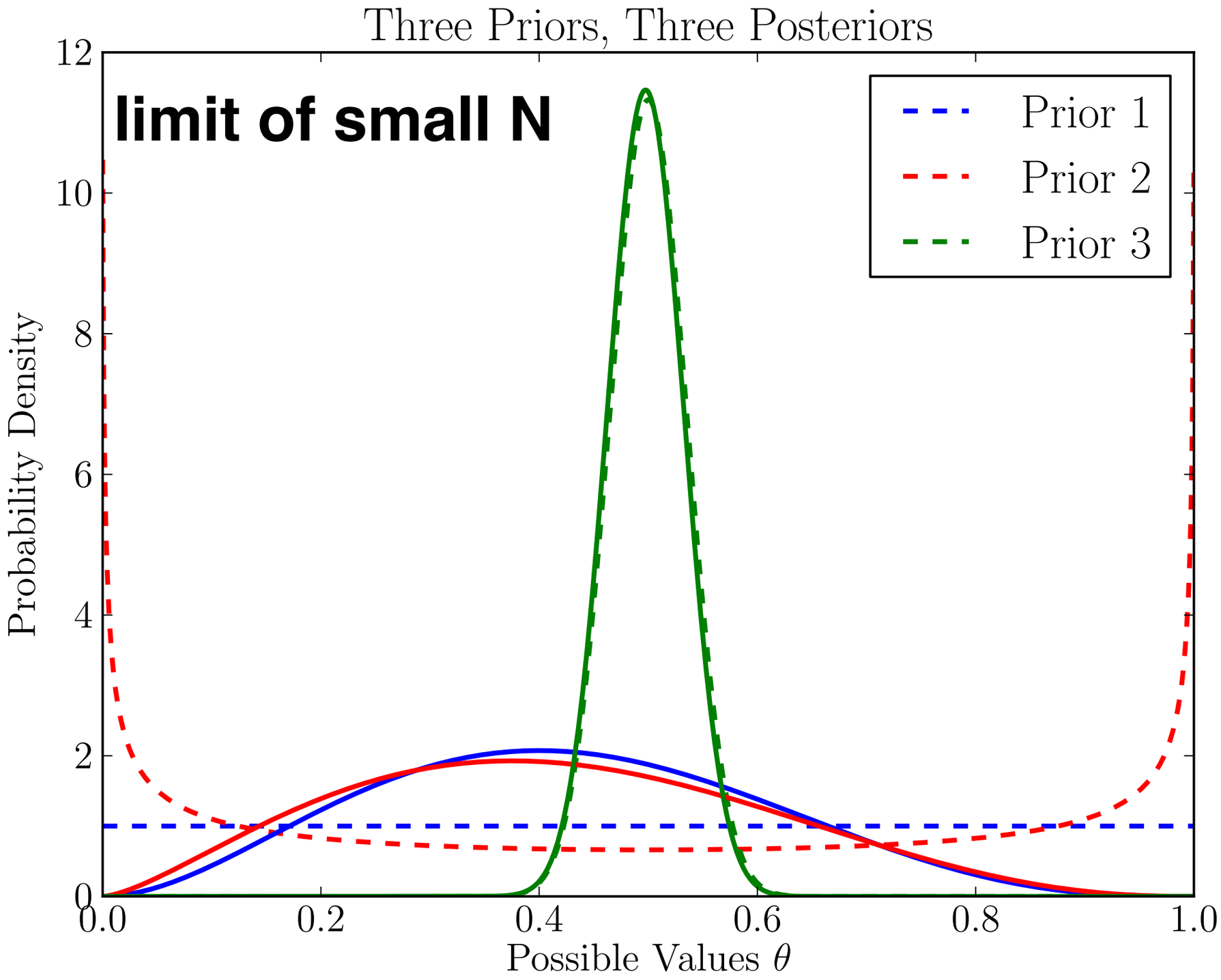


prior 1: “flat prior”, “prior ignorance”, “uninformative prior” (“improper”)



prior 1: “flat prior”, “prior ignorance”, “uninformative prior” (“improper”)

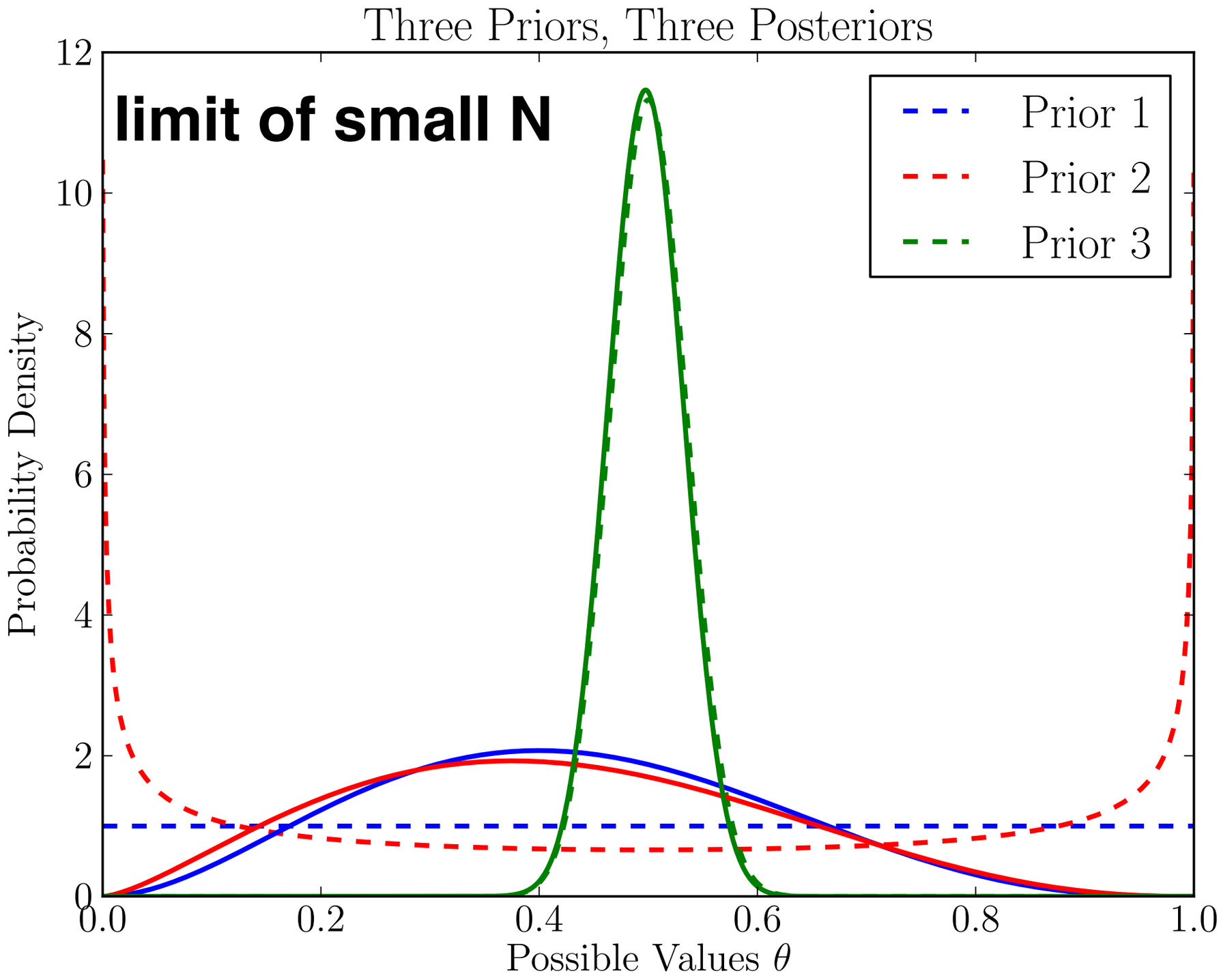
prior 2: emphasizing the extremes (“informative prior”)



prior 1: “flat prior”, “prior ignorance”, “uninformative prior” (“improper”)

prior 2: emphasizing the extremes (“informative prior”)

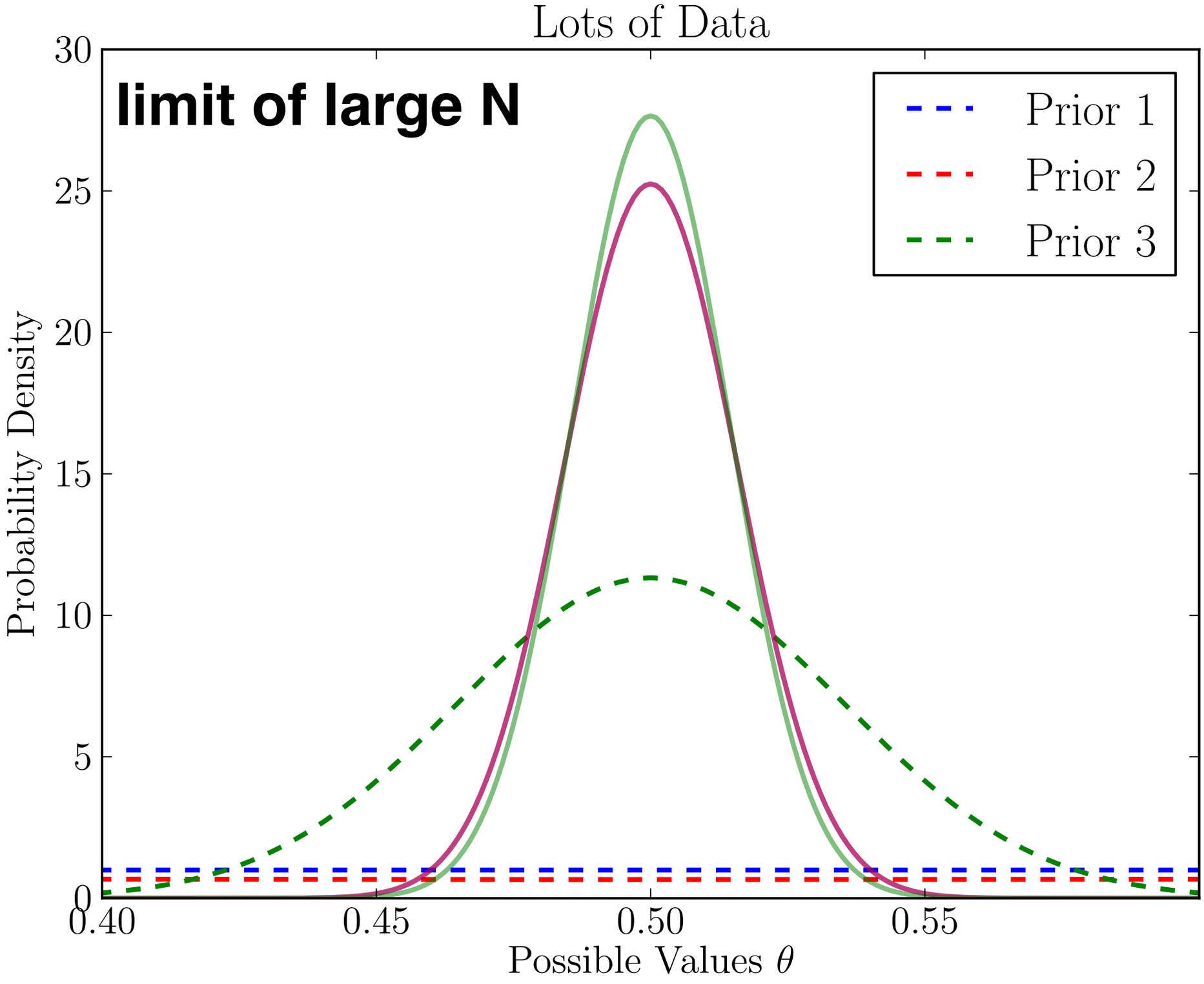
prior 3: good prior knowledge (“informative prior”)



prior 1: “flat prior”, “prior ignorance”, “uninformative prior” (“improper”)

prior 2: emphasizing the extremes (“informative prior”)

prior 3: good prior knowledge (“informative prior”)


Lots of Data
limit of large N
Prior 1
Prior 2
Prior 3
Probability Density
30
25
20
15
10
5
0
0.40
0.45
0.50
0.55
Possible Values θ

Data analysis recipes: Fitting a model to data*

David W. Hogg
Center for Cosmology and Particle Physics, Department of Physics, New York University
Max-Planck-Institut für Astronomie, Heidelberg

Jo Bovy
Center for Cosmology and Particle Physics, Department of Physics, New York University

Dustin Lang
Department of Computer Science, University of Toronto
Princeton University Observatory

Abstract

We go through the many considerations involved in fitting a model to data, using as an example the fit of a straight line to a set of points in a two-dimensional plane. Standard weighted least-squares fitting is only appropriate when there is a dimension along which the data points have negligible uncertainties, and another along which all the uncertainties can be described by Gaussians of known variance; these conditions are rarely met in practice. We consider cases of general, heterogeneous, and arbitrarily covariant two-dimensional uncertainties, and situations in which there are bad data (large outliers), unknown uncertainties, and unknown but expected intrinsic scatter in the linear relationship being fit. Above all we emphasize the importance of having a "generative model" for the data, even an approximate one. Once there is a generative model, the subsequent fitting is non-arbitrary because the model permits direct computation of the likelihood of the parameters or the posterior probability distribution. Construction of a posterior probability distribution is indispensible if there are "nuisance parameters" to marginalize away.

It is conventional to begin any scientific document with an introduction that explains why the subject matter is important. Let us break with tradition and observe that in almost all cases in which scientists fit a straight line to their data, they are doing something that is simultaneously *wrong* and *unnecessary*. It is wrong because circumstances in which a set of two

*The notes begin on page 39, including the license[1] and the acknowledgements[2].

Data analysis recipes: Fitting a model to data*

David W. Hogg
Center for Cosmology and Particle Physics, Department of Physics, New York University
Max-Planck-Institut für Astronomie, Heidelberg

Jo Bovy
Center for Cosmology and Particle Physics, Department of Physics, New York University

Dustin Lang
Department of Computer Science, University of Toronto
Princeton University Observatory

Abstract

We go through the many considerations involved in fitting a model to data, using as an example the fit of a straight line to a set of points in a two-dimensional plane. Standard weighted least-squares fitting is only appropriate when there is a dimension along which the data points have negligible uncertainties, and another along which all the uncertainties can be described by Gaussians of known variance; these conditions are rarely met in practice. We consider cases of general, heterogeneous, and arbitrarily covariant two-dimensional uncertainties, and situations in which there are bad data (large outliers), unknown uncertainties, and unknown but expected intrinsic scatter in the linear relationship being fit. Above all we emphasize the importance of having a "generative model" for the data, even an approximate one. Once there is a generative model, the subsequent fitting is non-arbitrary because the model permits direct computation of the likelihood of the parameters or the posterior probability distribution. Construction of a posterior probability distribution is indispensible if there are "nuisance parameters" to marginalize away.

It is conventional to begin any scientific document with an introduction that explains why the subject matter is important. Let us break with tradition and observe that in almost all cases in which scientists fit a straight line to their data, they are doing something that is simultaneously *wrong* and *unnecessary*. It is wrong because circumstances in which a set of two

*The notes begin on page 39, including the license[1] and the acknowledgements[2].

Data analysis recipes: Fitting a model to data*

David W. Hogg
Center for Cosmology and Particle Physics, Department of Physics, New York University
Max-Planck-Institut für Astronomie, Heidelberg

Jo Bovy
Center for Cosmology and Particle Physics, Department of Physics, New York University

Dustin Lang
Department of Computer Science, University of Toronto
Princeton University Observatory

Abstract

We go through the many considerations involved in fitting a model to data, using as an example the fit of a straight line to a set of points in a two-dimensional plane. Standard weighted least-squares fitting is only appropriate when there is a dimension along which the data points have negligible uncertainties, and another along which all the uncertainties can be described by Gaussians of known variance; these conditions are rarely met in practice. We consider cases of general, heterogeneous, and arbitrarily covariant two-dimensional uncertainties, and situations in which there are bad data (large outliers), unknown uncertainties, and unknown but expected intrinsic scatter in the linear relationship being fit. Above all we emphasize the importance of having a "generative model" for the data, even an approximate one. Once there is a generative model, the subsequent fitting is non-arbitrary because the model permits direct computation of the likelihood of the parameters or the posterior probability distribution. Construction of a posterior probability distribution is indispensible if there are "nuisance parameters" to marginalize away.

It is conventional to begin any scientific document with an introduction that explains why the subject matter is important. Let us break with tradition and observe that in almost all cases in which scientists fit a straight line to their data, they are doing something that is simultaneously *wrong* and *unnecessary*. It is wrong because circumstances in which a set of two

*The notes begin on page 39, including the license[1] and the acknowledgements[2].

It is wrong because circumstances in which a set of two dimensional measurements—outputs from an observation, experiment, or calculation—are truly drawn from a narrow, linear relationship is exceedingly rare. Indeed, almost any transformation of coordinates renders a truly linear relationship non-linear. Furthermore, even if a relationship *looks* linear, unless there is a confidently held theoretical reason to believe that the data are generated from a linear relationship, it probably isn't linear in detail; in these cases fitting with a linear model can introduce substantial systematic error, or generate apparent inconsistencies among experiments that are intrinsically consistent.

Even if the investigator doesn't care that the fit is wrong, it is likely to be unnecessary. Why? Because it is rare that, given a complicated observation, experiment, or calculation, the important *result* of that work to be communicated forward in the literature and seminars is the *slope and intercept* of a best-fit line! Usually the full distribution of data is much more rich, informative, and important than any simple metrics made by fitting an overly simple model.

That said, it must be admitted that one of the most effective ways to communicate scientific results is with catchy punchlines and compact, approximate representations, even when those are unjustified and unnecessary. It can also sometimes be useful to fit a simple model to predict new data, given existing data, even in the absence of a physical justification for the fit.[3] For these reasons—and the reason that in rare cases the fit *is* justifiable and essential—the problem of fitting a line to data comes up very frequently in the life of a scientist.

It is a miracle with which we hope everyone reading this is familiar that *if* you have a set of two-dimensional points (x, y) that depart from a perfect, narrow, straight line $y = m\,x + b$ only by the addition of Gaussian-distributed noise of known amplitudes in the y direction only, *then* the maximum-likelihood or best-fit line for the points has a slope m and intercept b that can be obtained justifiably by a perfectly linear matrix-algebra operation known as "weighted linear least-square fitting". This miracle deserves contemplation.

“All models are wrong, but some are useful.”

~ George Box, 1978

Well-known example

$$PV = RT$$

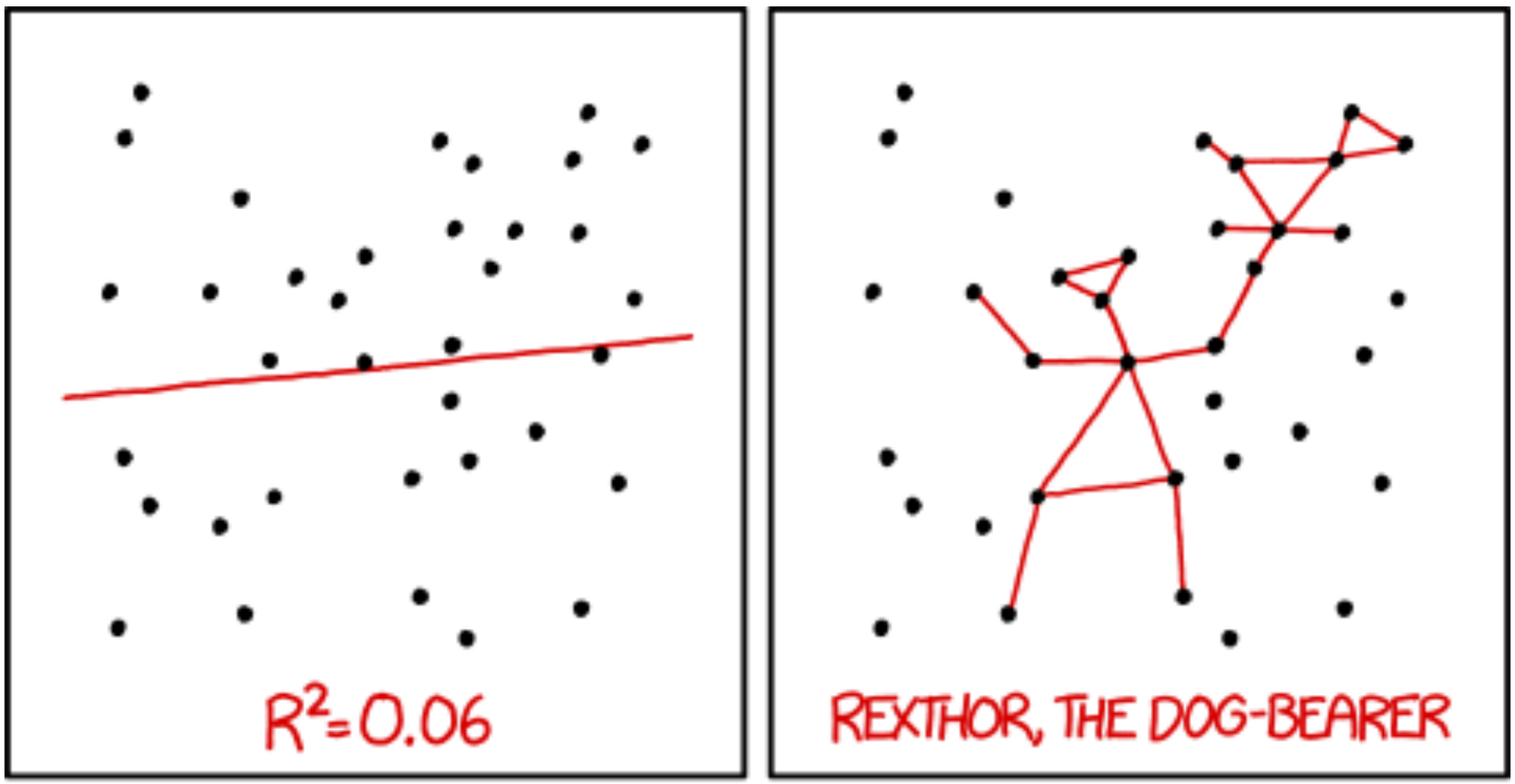
Well-known example

$$y = mx + b$$

“A model is a simplification or approximation of reality and hence will not reflect all of reality. …

Box noted that “all models are wrong, but some are useful.” While a model can never be “truth,” a model might be ranked from very useful, to useful, to somewhat useful to, finally, essentially useless.”

~ Burnham & Anderson, 1998



I DON'T TRUST LINEAR REGRESSIONS WHEN IT'S HARDER TO GUESS THE DIRECTION OF THE CORRELATION FROM THE SCATTER PLOT THAN TO FIND NEW CONSTELLATIONS ON IT.

http://xkcd.com/1725

“Parameter Estimation”

Given a model, what can we learn about its parameters

e.g., **y = mx + b**

“Model Selection”

Given multiple models, can we decide which one better represents the data

e.g.,

y = mx + b

VS.

y = b + mx + nx^2 + px^3

“Parameter Estimation”

We can compute the posterior up to a normalization constant:

$$P(\theta|D) = \frac{1}{Z} P(D|\theta)\, P(\theta)$$

“Parameter Estimation”

We can compute the posterior up to a normalization constant:

$$P(\theta|D) = \frac{1}{Z} P(D|\theta)\, P(\theta)$$

This constant, **Z**, is the evidence:

$$Z = P(D) = \int P(D|\theta)\, P(\theta) d\theta$$

“Parameter Estimation”

Usually done in (natural) log-probability space.

$$P(\theta|D) = \frac{1}{Z} P(D|\theta)\, P(\theta) \longrightarrow$$

“Parameter Estimation”

Usually done in (natural) log-probability space.

$$P(\theta|D) = \frac{1}{Z} P(D|\theta)\, P(\theta) \longrightarrow$$

$$\mathrm{lnP}(\theta|\mathrm{D}) = \mathrm{lnP}(\mathrm{D}|\theta) + \mathrm{lnP}(\theta) + \ln(\frac{1}{\mathrm{Z}})$$

“Parameter Estimation”

Usually done in (natural) log-probability space.

$$P(\theta|D) = \frac{1}{Z} P(D|\theta)\, P(\theta) \longrightarrow$$

$$\mathrm{lnP}(\theta|\mathrm{D}) = \mathrm{lnP}(\mathrm{D}|\theta) + \mathrm{lnP}(\theta) + \ln(\frac{1}{\mathrm{Z}})$$

“ln P”

“Parameter Estimation”

Usually done in (natural) log-probability space.

$$P(\theta|D) = \frac{1}{Z} P(D|\theta)\, P(\theta) \longrightarrow$$

$$\ln P(\theta|D) = \ln P(D|\theta) + \ln P(\theta) + \ln\left(\frac{1}{Z}\right)$$

“ln P”

“ln like”
“log-likelihood”

“Parameter Estimation”

Usually done in (natural) log-probability space.

$$P(\theta|D) = \frac{1}{Z} P(D|\theta)\, P(\theta) \longrightarrow$$

$$\ln P(\theta|D) = \ln P(D|\theta) + \ln P(\theta) + \ln\left(\frac{1}{Z}\right)$$

“ln P”

“ln like”
“log-likelihood”

“ln Prior”
“Prior”

“Parameter Estimation”

Usually done in (natural) log-probability space.

$$P(\theta|D) = \frac{1}{Z} P(D|\theta)\, P(\theta) \longrightarrow$$

$$\ln P(\theta|D) = \ln P(D|\theta) + \ln P(\theta) + \ln\left(\frac{1}{Z}\right)$$

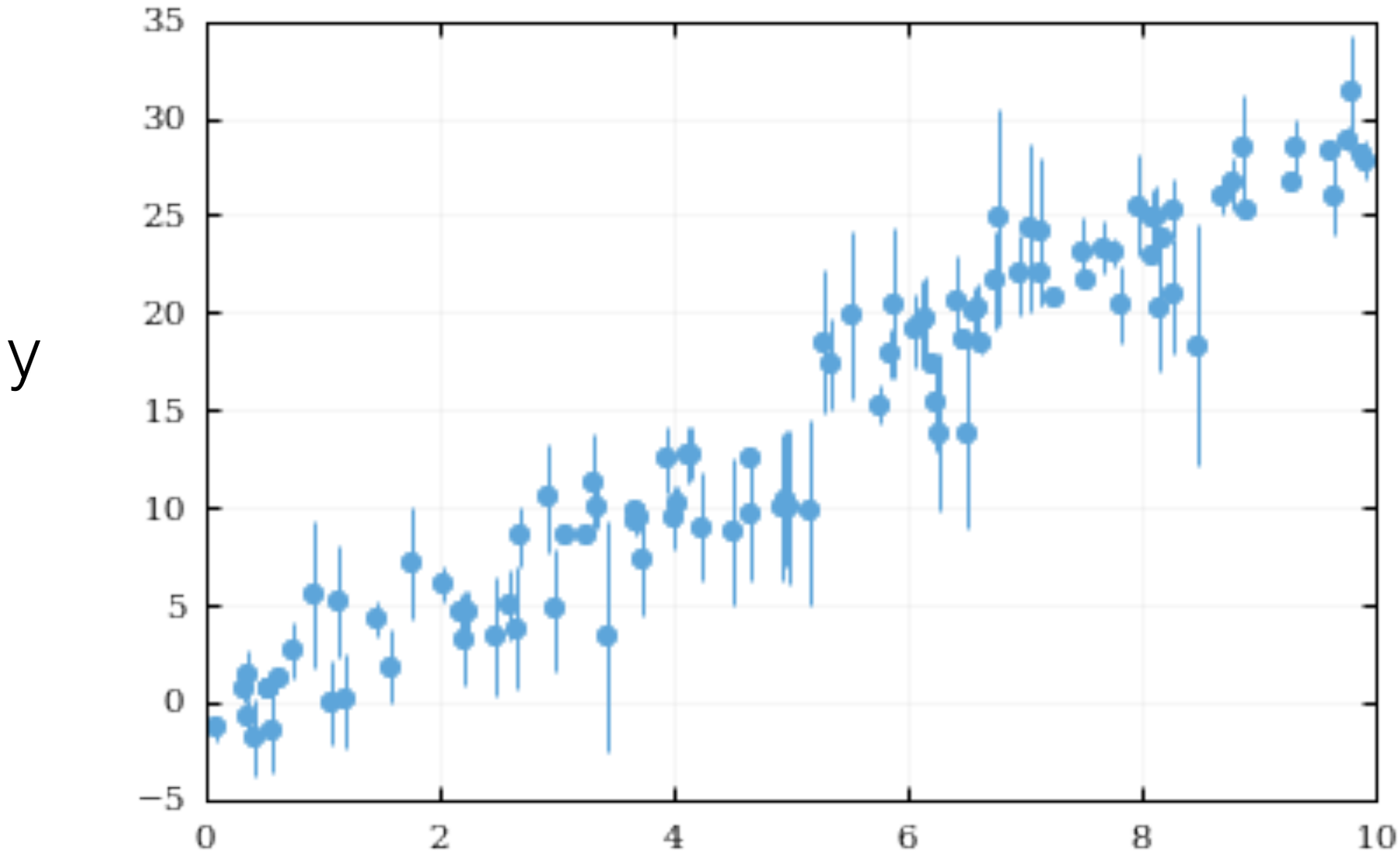
“ln P”

“ln like”
“log-likelihood”

“ln Prior”
“Prior”

just a constant, don’t need to compute

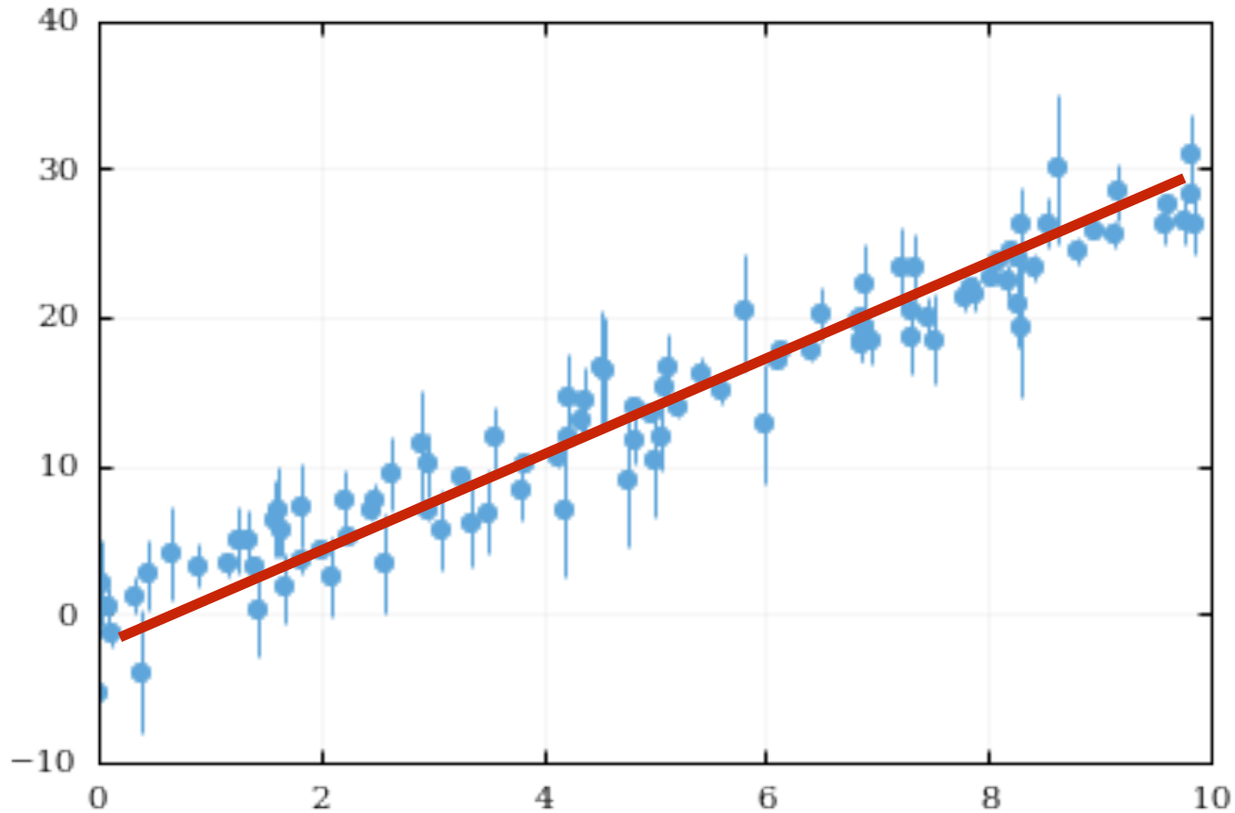
“Parameter Estimation”:
fitting a line to noisy data



“Parameter Estimation”:
Graphically

The “scene”

y = mx + b



y

x

Observations

y_i

sigma_i

find values of **m** and **b**, given data (i.e., **y** and **sigma**)

“Parameter Estimation”:

model

$$y = mx + b$$

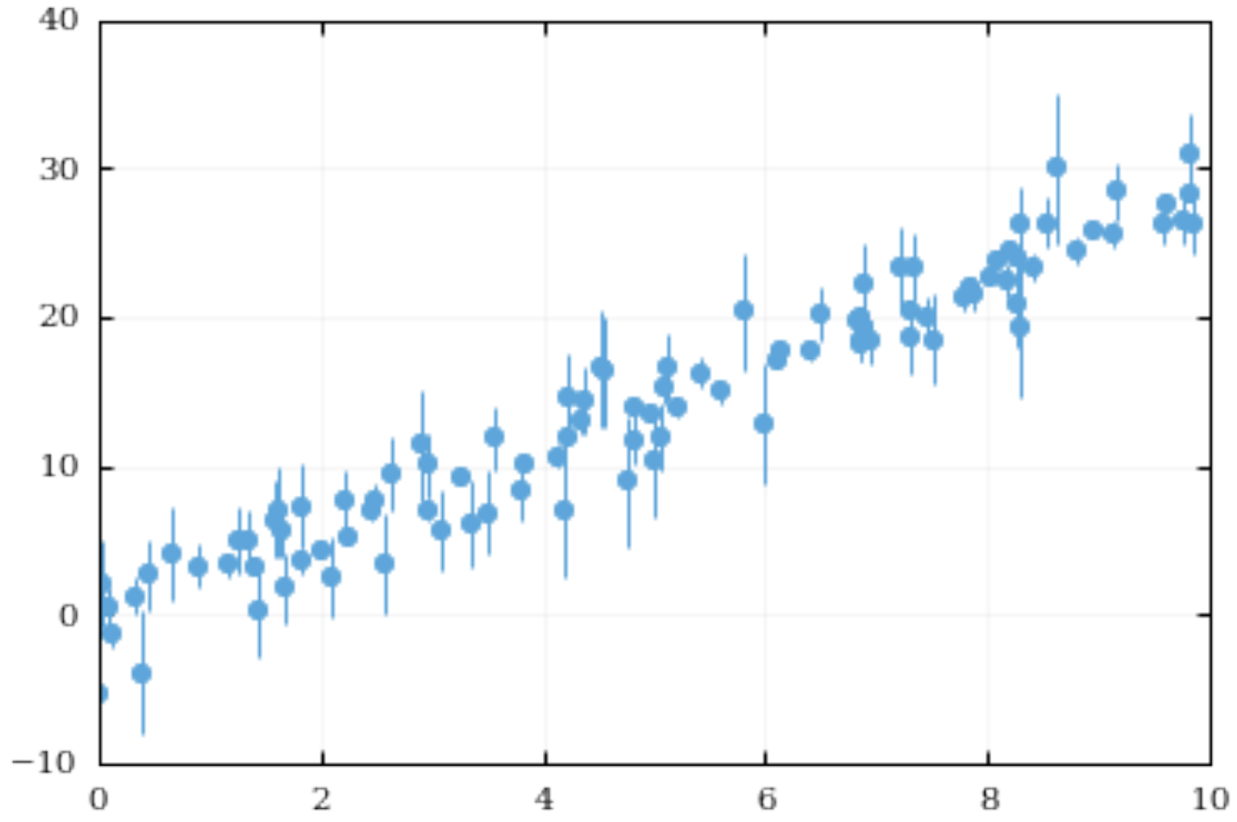
In [7]:
```
def model(x,m,b):
    y = m * x + b
    return y
```

“Parameter Estimation”:
generate fake data (model + Gaussian noise)

In [11]:
```
# generate fake data and add some Gaussian noise
def fake_data(x,m,b,dy):
    y = model(x,m,b)
    error =  dy * np.random.randn(len(y))
    y += error
    return y, error
```

“Parameter Estimation”:
generate fake data (model + Gaussian noise)

In [16]:
```
# generate x coordinates and sort them
x = np.sort(np.random.uniform(0,10, 100))
m, b = 3., -1. # slope and intercept
data, error = fake_data(x, m, b, 2.5)
print(data, error)
plt.errorbar(x,data, yerr=error, marker='o', ls='None')
```



“Parameter Estimation”:

Define Likelihood Function

Assume Gaussian error distribution for data, yields: χ^2

$$P(D|\theta) = \frac{1}{\sqrt{(2\pi\sigma^2)}}\, e^{\frac{-(y-\bar{y})^2}{2\sigma^2}}$$

<u>the data (or observations)</u>

$y = set\, of\, data\, points\, (\text{“}D\text{”})$

$\sigma = uncertainties$

<u>the model</u>

$\bar{y} = mx + b$

$\theta = (m, b)$

“Parameter Estimation”:

Define Likelihood Function

Assume Gaussian error distribution for data, yields: χ^2

$$P(D|\theta) = \prod_i^N \frac{1}{\sqrt{(2\pi\sigma_i^2)}}\, e^{\frac{-(y_i-\bar{y_i})^2}{2\sigma_i^2}}$$

<u>the data (or observations)</u>

$y_i = ith\, data\, point$

$\sigma_i = uncertainty\, on\, ith\, data\, point$

<u>the model</u>

$\bar{y_i} = mx_i + b_i$

$\theta = (m, b)$

“Parameter Estimation”:
Define Likelihood Function

Assume Gaussian error distribution for data, yields: χ^2

$$P(D|\theta) = \prod_i^N \frac{1}{\sqrt{(2\pi\sigma_i^2)}}\, e^{\frac{-(y_i-\bar{y_i})^2}{2\sigma_i^2}}$$

assume data are drawn **“identically and independently”** (iid)

the data (or observations)

$y_i = ith\, data\, point$

$\sigma_i = uncertainty\, on\, ith\, data\, point$

the model

$\bar{y_i} = mx_i + b_i$

$\theta = (m, b)$

“Parameter Estimation”:
Define Likelihood Function

$$P(D|\theta) = \prod_{i}^{N} \frac{1}{\sqrt{(2\pi\sigma_i^2)}}\, e^{\frac{-(y_i - \bar{y_i})^2}{2\sigma_i^2}}$$

write as log-likelihood ⟶

$$\ln \mathrm{P}(\mathrm{D}|\theta) = -0.5 \sum_{\mathrm{i}}^{\mathrm{N}} \left(\frac{(\mathrm{y_i} - \bar{\mathrm{y_i^2}})}{\sigma_\mathrm{i}^2} + \ln(2\pi\sigma_\mathrm{i}^2) \right)$$

“Parameter Estimation”:
Define Likelihood Function

$$\ln \mathrm{P}(\mathrm{D}|\theta) = -0.5 \sum_{\mathrm{i}}^{\mathrm{N}} \left(\frac{(\mathrm{y_i} - \bar{\mathrm{y}_\mathrm{i}^2})}{\sigma_\mathrm{i}^2} + \ln(2\pi\sigma_\mathrm{i}^2) \right)$$

In [13]:
```
# write down log-likelihood function
def lnlike(theta, x, y, sigma):
    return -0.5 * np.sum( (y - model(x, theta[0], theta[1]))**2 / sigma**2 + np.log(2. * np.pi * sigma**2))
```

“Parameter Estimation”:
Define log-Priors

Assume flat priors on **m** and **b**:

```
In [14]: def lnprior(theta):
             if 0. < theta[0] < 4. and -5. < theta[1] < 2.0:
                 return 0.
             else:
                 return -np.inf
```

If outside prior range, return -inf

“Parameter Estimation”:
Write down log-posterior

In [15]:
```
def lnprob(theta, x, y, sigma):
    return lnprior(theta) + lnlike(theta, x, y, sigma)
```

“Parameter Estimation”:
Write down log-posterior

$$\ln \mathrm{P}(\theta|\mathrm{D}) = \ln \mathrm{P}(\mathrm{D}|\theta) + \ln \mathrm{P}(\theta) + \ln(\frac{1}{\mathrm{Z}})$$

In [15]:
```
def lnprob(theta, x, y, sigma):
    return lnprior(theta) + lnlike(theta, x, y, sigma)
```

“Parameter Estimation”:

How do we actually get the values for m and b?

“Parameter Estimation”:

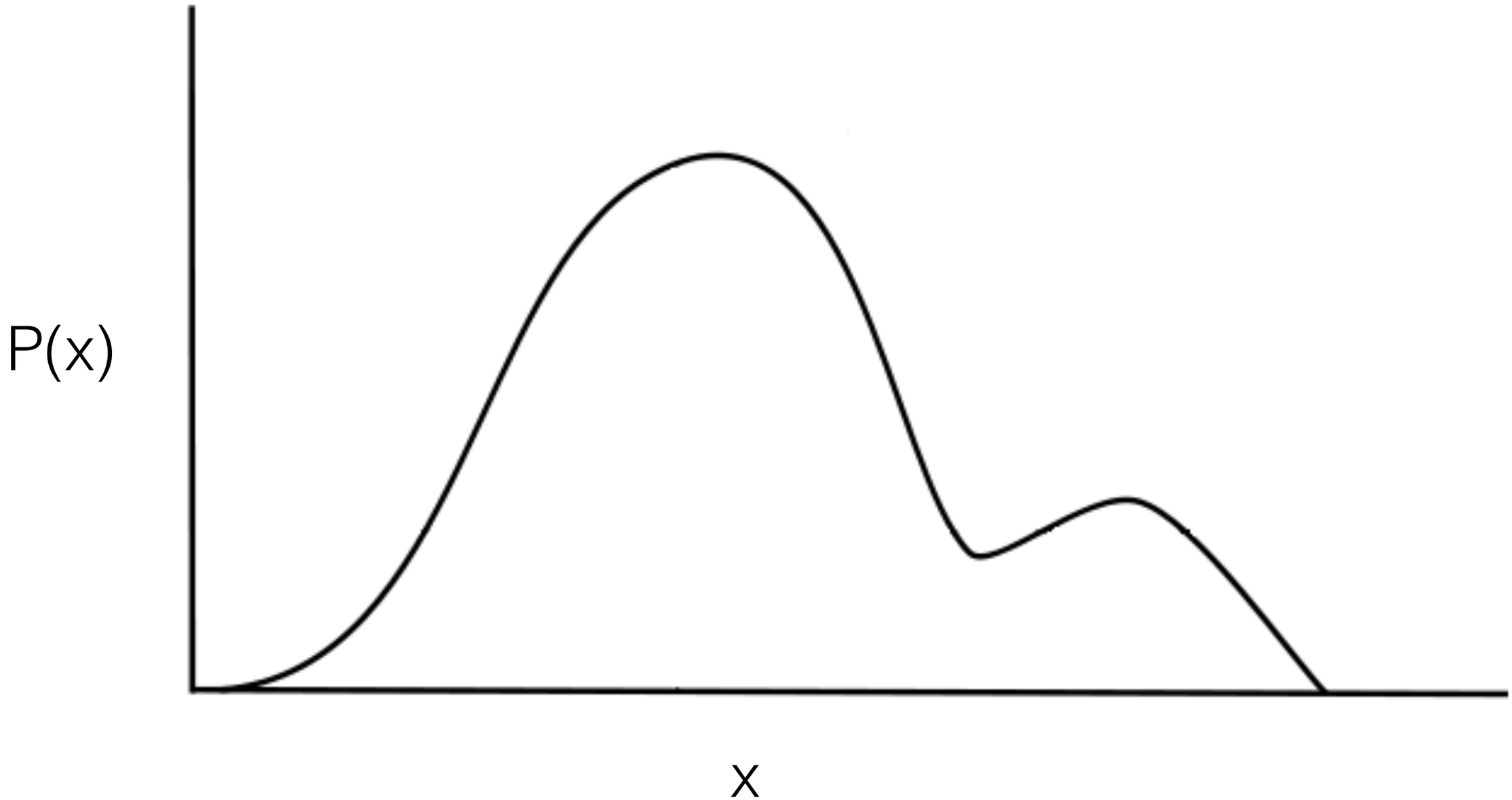
How do we actually get the values for m and b?

$$\ln P(\theta|D) = \ln P(D|\theta) + \ln P(\theta) + \ln(\frac{1}{Z})$$

“Parameter Estimation”:

How do we actually get the values for m and b?

1d case



“Parameter Estimation”:

How do we actually get the values for m and b?

1. Solve the problem exactly.
2. Compute P(x) at gridded values of x.
3. Optimize.
4. Sample.