Lectures 11: Bootstrap I.

error propagation for nonlinear functions of fit parameters

loose ends from lecture 10: (degrees of freedom)

χ^2 distribution of the fitted parameters

from lecture 10

How accurately are the fitted parameters determined?
As Bayesians, we would **instead** say, what is their posterior distribution?

Taylor series:

$$-\tfrac{1}{2}\chi^2(\mathbf{b}) \approx -\tfrac{1}{2}\chi^2_{\rm min} - \tfrac{1}{2}(\mathbf{b}-\mathbf{b}_0)^T \left[\tfrac{1}{2}\frac{\partial^2\chi^2}{\partial\mathbf{b}\partial\mathbf{b}}\right](\mathbf{b}-\mathbf{b}_0)$$

So, while exploring the χ^2 surface to find its minimum, we must also calculate the Hessian (2nd derivative) matrix at the minimum.

Then

$$P(\mathbf{b}|\{y_i\}) \propto \exp\left[-\tfrac{1}{2}(\mathbf{b}-\mathbf{b}_0)^T\boldsymbol{\Sigma}_b^{-1}(\mathbf{b}-\mathbf{b}_0)\right] P(\mathbf{b})$$

with

$$\boldsymbol{\Sigma}_b = \left[\tfrac{1}{2}\frac{\partial^2\chi^2}{\partial\mathbf{b}\partial\mathbf{b}}\right]^{-1}$$

covariance (or "standard error") matrix of the fitted parameters

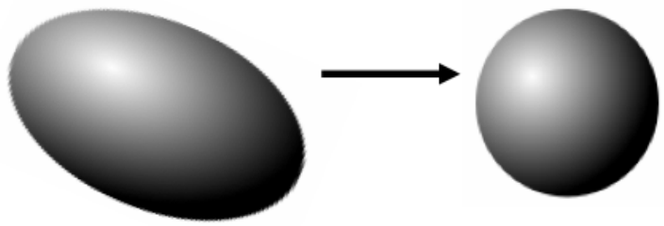
Notice that if (i) the Taylor series converges rapidly and (ii) the prior is uniform, then the posterior distribution of the $\mathbf{b}$'s is multivariate Normal

confidence intervals

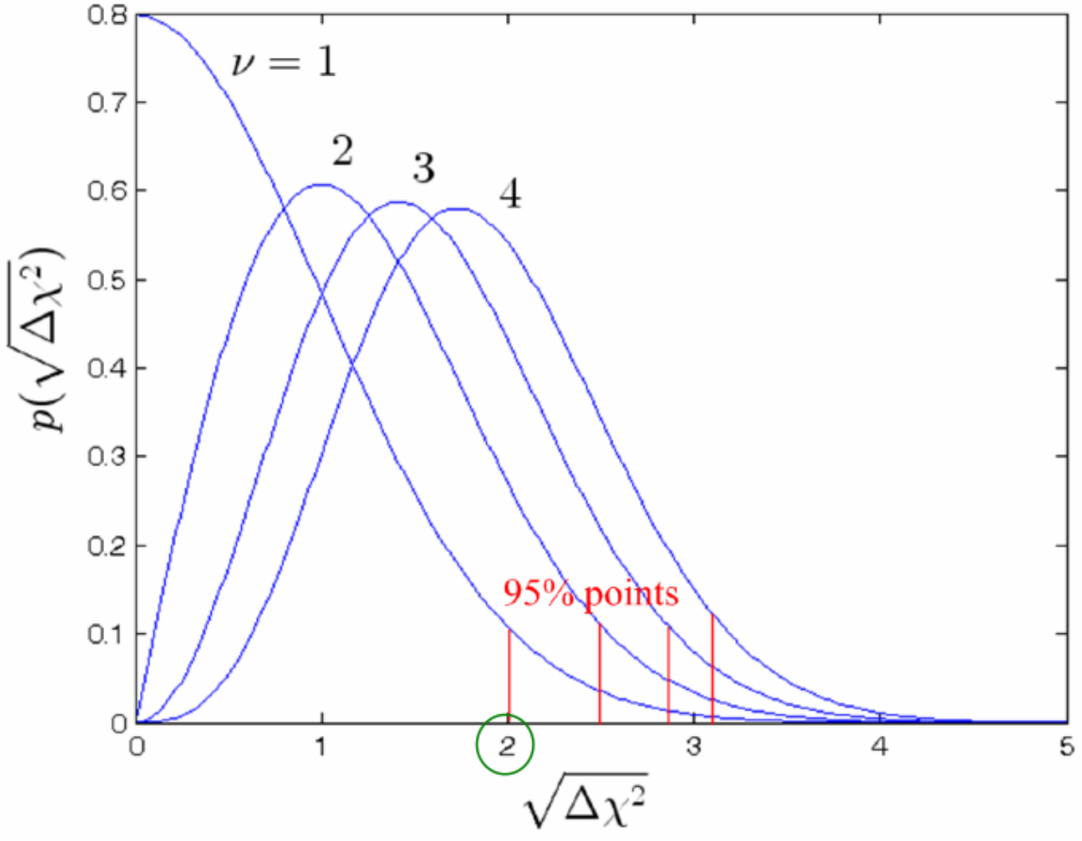
from lecture 10

What $\Delta\chi^2$ contour in ν dimensions contains some percentile probability?

Rotate and scale the covariance to make it spherical.
(Linear, so contours still contain same probability.)



Now, each dimension is an independent Normal, and contours are labeled by radius squared (sum of ν individual t^2 values), so $\Delta\chi^2$ ~ Chisquare(ν)



$\Delta\chi^2$ as a Function of Confidence Level p and Number of Parameters of Interest ν						
	ν					
p	1	2	3	4	5	6
68.27%	1.00	2.30	3.53	4.72	5.89	7.04
90%	2.71	4.61	6.25	7.78	9.24	10.6
95.45%	4.00	6.18	8.02	9.72	11.3	12.8
99%	6.63	9.21	11.3	13.3	15.1	16.8
99.73%	9.00	11.8	14.2	16.3	18.2	20.1
99.99%	15.1	18.4	21.1	23.5	25.7	27.9

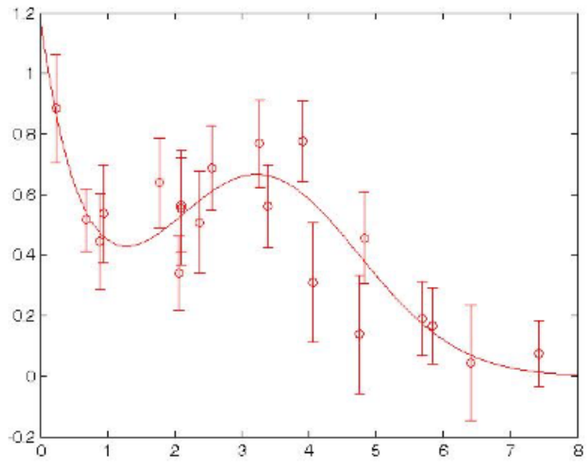
You sometimes learn "facts" like: "delta chi-square of 1 is the 68% confidence level". We now see that this is true only for one parameter at a time.

what is the Degree of Freedom?

from lecture 10

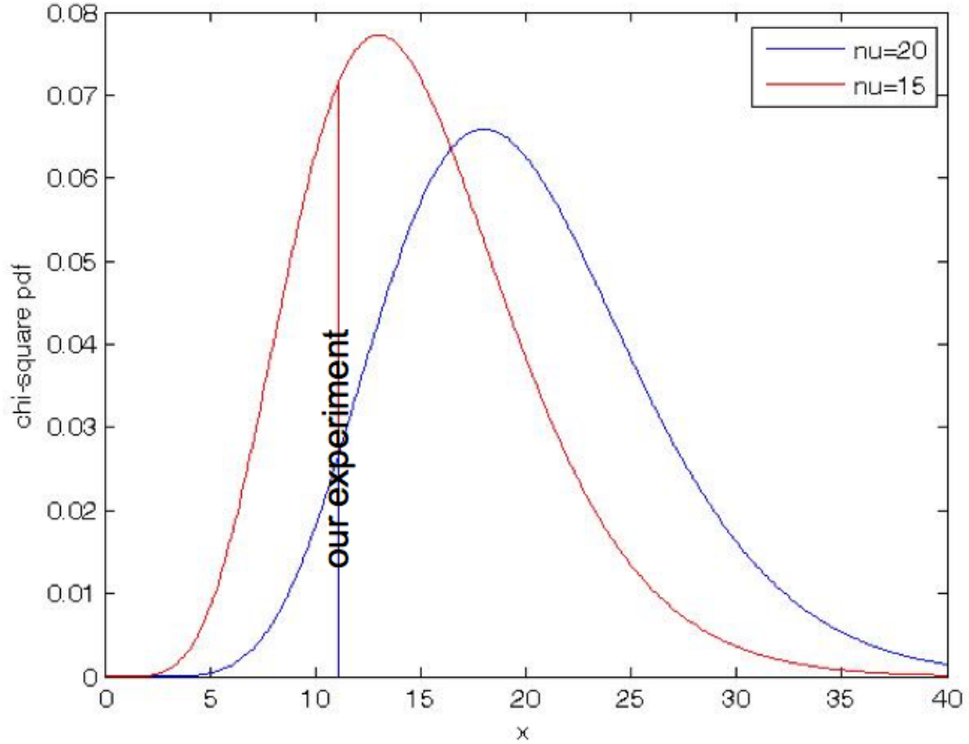
How is our fit by this test?

In our example, $\chi^2(\mathbf{b}_0) = 11.13$



This is a bit unlikely in Chisquare(20),
with (left tail) p=0.0569.

In fact, if you had many repetitions of the experiment, you would find that their χ^2 is <u>not</u> distributed as Chisquare(20), but rather as Chisquare(15)!
Why?



the magic word is:
“degrees of freedom” or DOF

what is the Degree of Freedom?

from lecture10

Degrees of Freedom: Why is χ^2 with N data points "not quite" the sum of N t^2-values? Because DOFs are reduced by constraints.

First consider a hypothetical situation where the data has linear constraints:

$$t_i = \frac{y_i - \mu_i}{\sigma_i} \sim \mathrm{N}(0,1)$$

joint distribution on all the t's, if they are independent

$$p(\mathbf{t}) = \prod_i p(t_i) \propto \exp\left(-\tfrac{1}{2}\sum_i t_i^2\right)$$

χ^2 is squared distance from origin $\sum t_i^2$

Linear constraint:

$$\sum_i \alpha_i y_i = C \Rightarrow \langle C \rangle = \sum_i \alpha_i \mu_i$$

$$C = \sum_i \alpha_i (\sigma_i t_i + \mu_i)$$

$$= \sum_i \alpha_i \sigma_i t_i + C$$

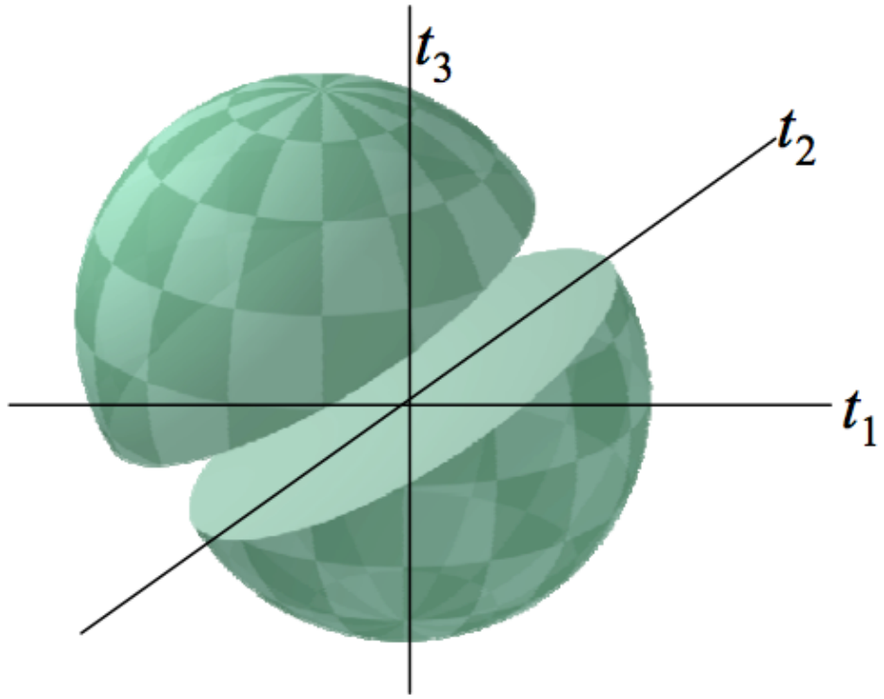
$$\text{So, } \sum_i \alpha_i \sigma_i t_i = 0$$

a hyper plane through the origin in t space!

what is the Degree of Freedom?

from lecture 10

Constraint is a plane cut through the origin. Any cut through the origin of a sphere is a circle.



So the distribution of distance from origin is the same as a multivariate normal "ball" in the lower number of dimensions. Thus, each linear constraint reduces ν by exactly 1.

We don't have explicit constraints on the y_i's. But as the y_i's wiggle around (within their errors) we do have the constraint that we want to keep the MLE estimate $\mathbf{b_0}$ fixed. (E.g., we have 20 wiggling y_i's and only 5 b_i's to keep fixed.)

So by the implicit function theorem, there are M (number of parameters) approximately linear constraints on the y_i 's. So $\nu = N - M$, the so-called number of degrees of freedom (d.o.f.).

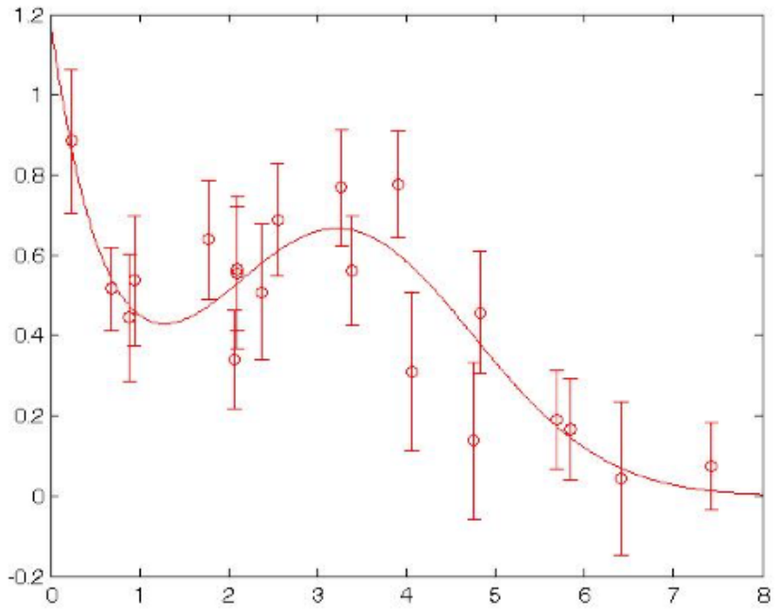
what is the Degree of Freedom?

from lecture10

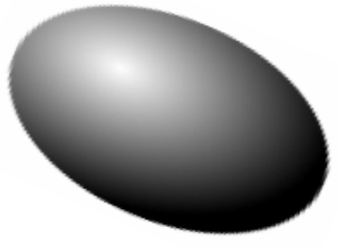
Review:

1. Fit for parameters by minimizing

$$\chi^2 = \sum_{i=1}^{N} \left(\frac{y_i - y(\mathbf{x}_i|\mathbf{b})}{\sigma_i} \right)^2$$



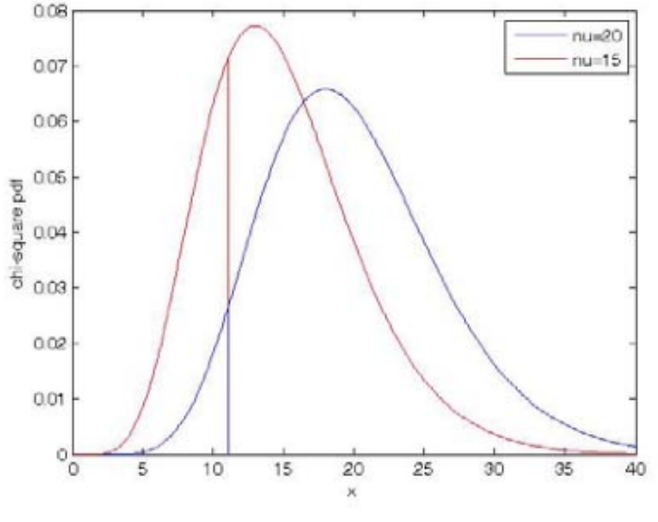
2. (Co)variances of parameters, or confidence regions, by the change in χ^2 (i.e., $\Delta\chi^2$) from its minimum value χ^2_{min}.



$\Delta\chi^2$ as a Function of Confidence Level p and Number of Parameters of Interest ν

	ν					
p	1	2	3	4	5	6
68.27%	1.00	2.30	3.53	4.72	5.89	7.04
90%	2.71	4.61	6.25	7.78	9.24	10.6
95.45%	4.00	6.18	8.02	9.72	11.3	12.8
99%	6.63	9.21	11.3	13.3	15.1	16.8
99.73%	9.00	11.8	14.2	16.3	18.2	20.1
99.99%	15.1	18.4	21.1	23.5	25.7	27.9

3. Goodness-of-fit (accept or reject model) by the p-value of χ^2_{min} using the correct number of DOF.



error on nonlinear functions of fitted parameters?

like error on b3*b5 ?

error on nonlinear function of fitted parameters?

What is the uncertainty in quantities other than the fitted coefficients:

I. Linearized error propagation

$\mathbf{b}_0$ is the MLE parameters estimate

$\mathbf{b}_1 \equiv \mathbf{b} - \mathbf{b}_0$ is the RV as the parameters fluctuate

$$f \equiv f(\mathbf{b}) = f(\mathbf{b}_0) + \nabla f\ \mathbf{b}_1 + \cdots$$

$$\langle f \rangle \approx \langle f(\mathbf{b}_0) \rangle + \nabla f\ \cancel{\langle \mathbf{b}_1 \rangle} = f(\mathbf{b}_0)$$

$$\left\langle f^2 \right\rangle - \langle f \rangle^2 \approx 2 f(\mathbf{b}_0)(\nabla f\ \cancel{\langle \mathbf{b}_1 \rangle}) + \left\langle (\nabla f\ \mathbf{b}_1)^2 \right\rangle$$

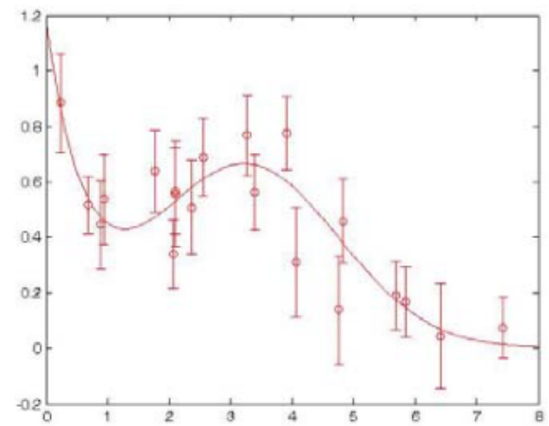
$$= \nabla f\ \left\langle \mathbf{b}_1 \mathbf{b}_1^T \right\rangle \nabla f\ ^T$$

$$= \nabla f\ \mathbf{\Sigma}\ \nabla f\ ^T$$

Linearized error propagation

In our example, if we are interested in the area of the “hump”,

```
bfit =
    1.1235    1.5210    0.6582    3.2654    1.4832
covar =
    0.1349    0.2224    0.0068   -0.0309    0.0135
    0.2224    0.6918    0.0052   -0.1598    0.1585
    0.0068    0.0052    0.0049    0.0016   -0.0094
   -0.0309   -0.1598    0.0016    0.0746   -0.0444
    0.0135    0.1585   -0.0094   -0.0444    0.0948
```



$$f = b_3 b_5$$

$$\nabla f = (0, 0, b_5, 0, b_3)$$

$$\nabla f \, \mathbf{\Sigma} \, \nabla f^T = b_5^2 \Sigma_{33} + 2 b_3 b_5 \Sigma_{35} + b_3^2 \Sigma_{55} = 0.0336$$

$$\sqrt{0.0336} = 0.18$$

So $b_3 b_5 = 0.98 \pm 0.18$ ← the one standard deviation (1-σ) error bar

Is it normally distributed?

Absolutely not! A function of normals is not normal (although, if they are all narrow, it might be close).

Sampling the posterior histogram

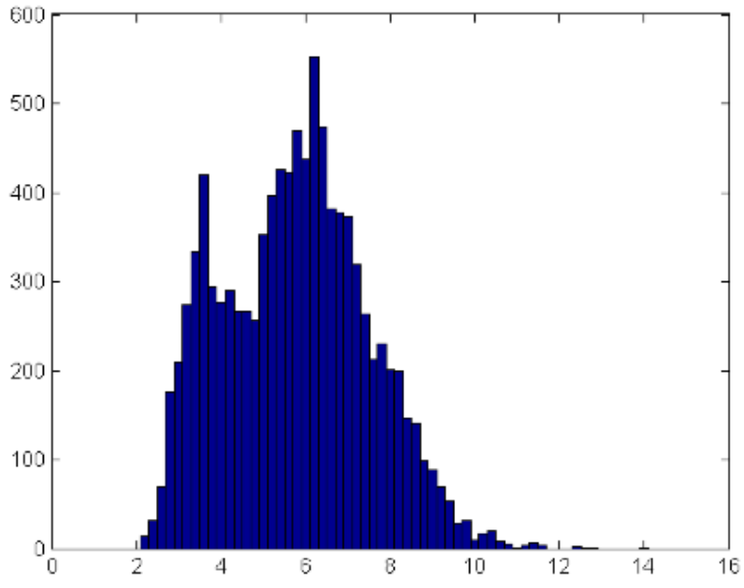
Method 2: Sample from the posterior distribution

1. Generate a large number of (vector) **b**'s

$$\mathbf{b} \sim \mathrm{MVNormal}(\mathbf{b}_0, \boldsymbol{\Sigma}_b)$$

2. Compute your $f(\mathbf{b})$ separately for each **b**

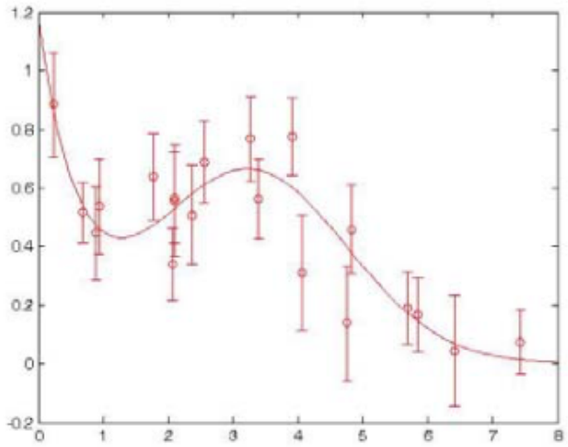
3. Histogram



Note again that **b** is typically (close to) m.v. normal because of the CLT, but your (nonlinear) f may not, in general, be anything even close to normal!

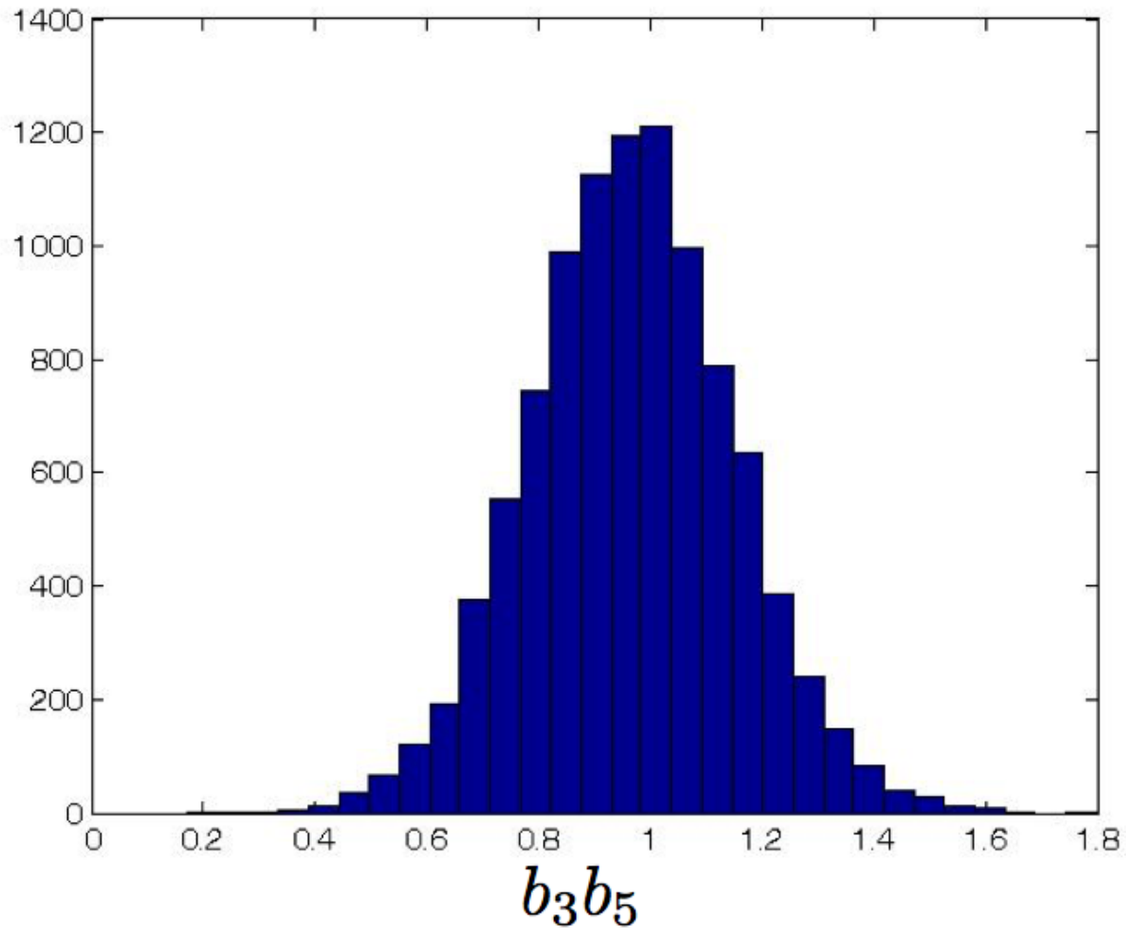
Sampling the posterior histogram

Our example:



```
bees = mvnrnd(bfit,covar,10000);
humps = bees(:,3).*bees(:,5);
hist(humps,30);
std(humps)
```

std = 0.1833



Does it matter that I use the full covar, not just the 2x2 piece for parameters 3 and 5?

comparison of linear propagation and posterior sampling:

Compare linear propagation of errors to sampling the posterior

- Note that even with lots of data, so that the distribution of the b's really → multivariate normal, a derived quantity might be very non-Normal.
  - In this case, sampling the posterior is a good idea!
- For example, the ratio of two normals of zero mean is Cauchy
  - which is very non-Normal!.
- So, sampling the posterior is a more powerful method than linear propagation of errors.
  - even when optimistically (or in ignorance) assuming multivariate Gaussian for the fitted parameters
- In fact, sampling the posterior distribution of large Bayesian models whose parameters are not at all Gaussian is, under the name MCMC, the most powerful technique in modern computational statistics.

bootstrap sampling

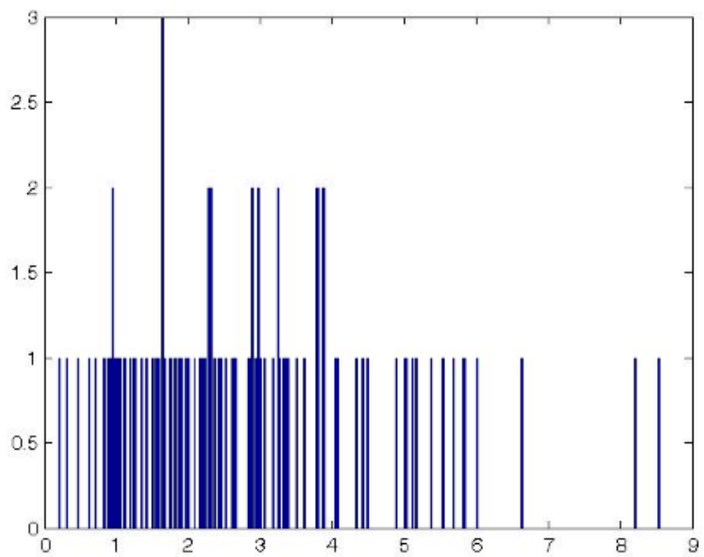
Method 3: Bootstrap resampling of the data

- We applied some end-to-end process to a data set and got a number f out
- The data set was drawn from a population of repetitions of the identical experiment
  - which we don't get to see, unfortunately
  - we see only a sample of the population
- We'd like to draw new data sets from the population, reapply the process, and see the distribution of answers
  - this would tell us how accurate the original answer, on average, was
  - but we can't: we don't have access to the population
- However, the data set itself is an estimate of the population pdf!
  - in fact, it's the only estimate we've got!
- So we draw from the data set – with replacement – many "fake" data sets of equal size, and carry out the proposed program
  - does this sound crazy? for a long time many people thought so!
  - Bootstrap theorem [glossing over technical assumptions]: The distribution of any resampled quantity around its full-data-set value estimates (naively: "asymptotically has the same histogram as") the distribution of the data set value around the population value.

bootstrap sampling

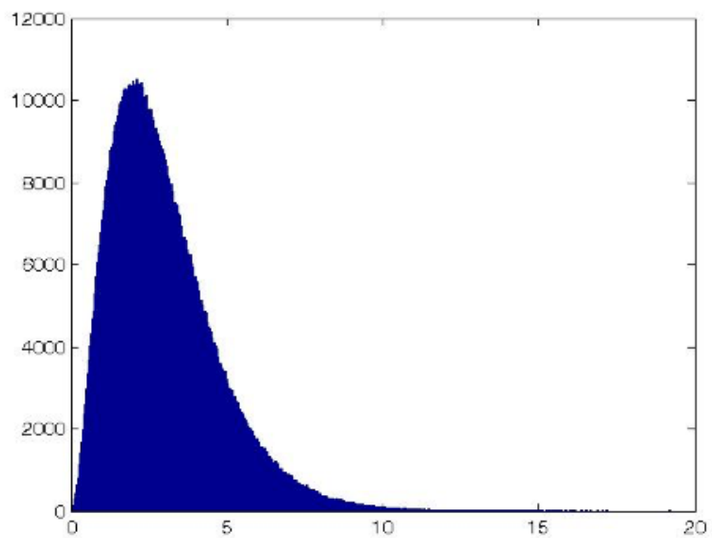
Let's try a simple example where we can see the "hidden" side of things, too.

Visible side (sample):



These happen to be drawn from a Gamma distribution.

Hidden side (population):



Statistic we are interested in happens to be (it could be anything):

$$\frac{\text{mean of distribution}}{\text{median of distribution}}$$

```
sammedian = median(sample)
sammean = mean(sample)
samstatistic = sammean/sammedian
sammedian =
    2.6505
sammean =
    2.9112
samstatistic =
    1.0984
```

How accurate is this?

```
themedian = median(bigsample)
themean = mean(bigsample)
thestatistic = themean/themedian
themedian =
    2.6730
themean =
    2.9997
thestatistics =
    1.1222
```

bootstrap sampling

Gamma distribution:

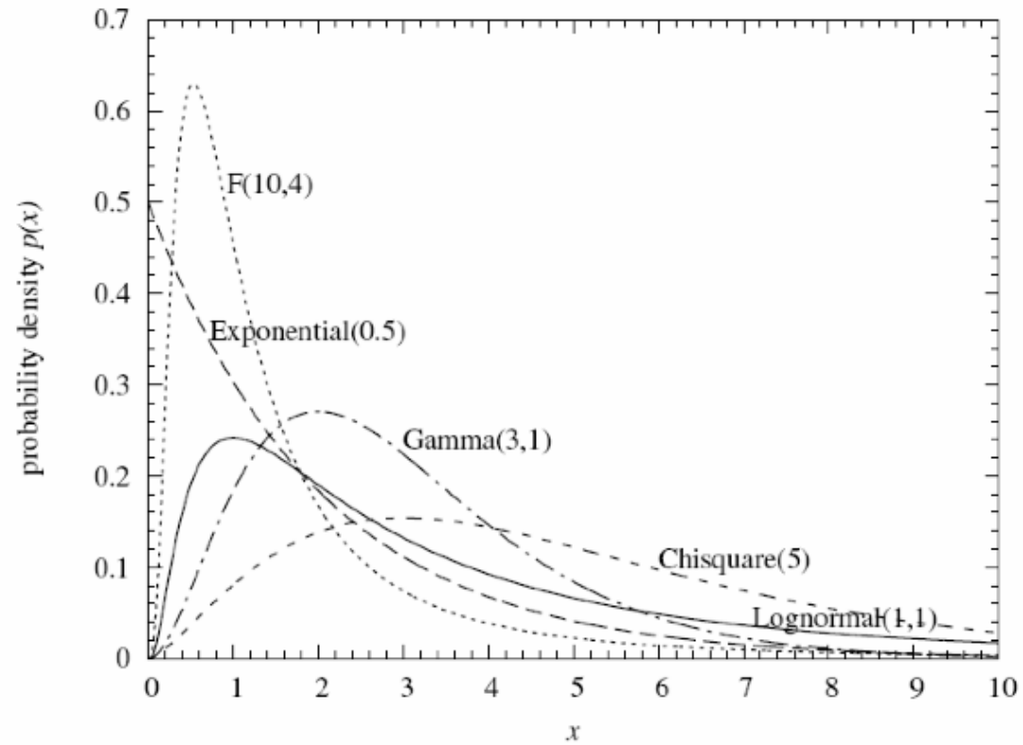
$$x \sim \text{Gamma}(\alpha, \beta), \qquad \alpha > 0,\ \beta > 0$$

$$p(x) = \frac{\beta^{\alpha}}{\Gamma(\alpha)} x^{\alpha-1} e^{-\beta x}, \qquad x > 0$$

$$\text{Mean}\{\text{Gamma}(\alpha, \beta)\} = \alpha/\beta$$

$$\text{Var}\{\text{Gamma}(\alpha, \beta)\} = \alpha/\beta^2$$

When $\alpha \geq 1$ there is a single mode at $x = (\alpha - 1)/\beta$

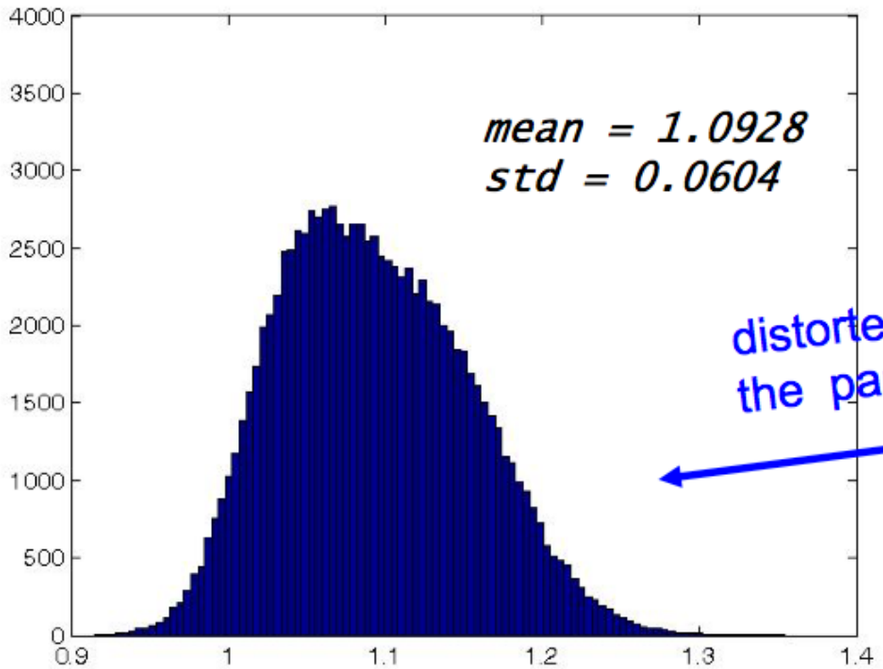


bootstrap sampling

To estimate the accuracy of our statistic, we bootstrap

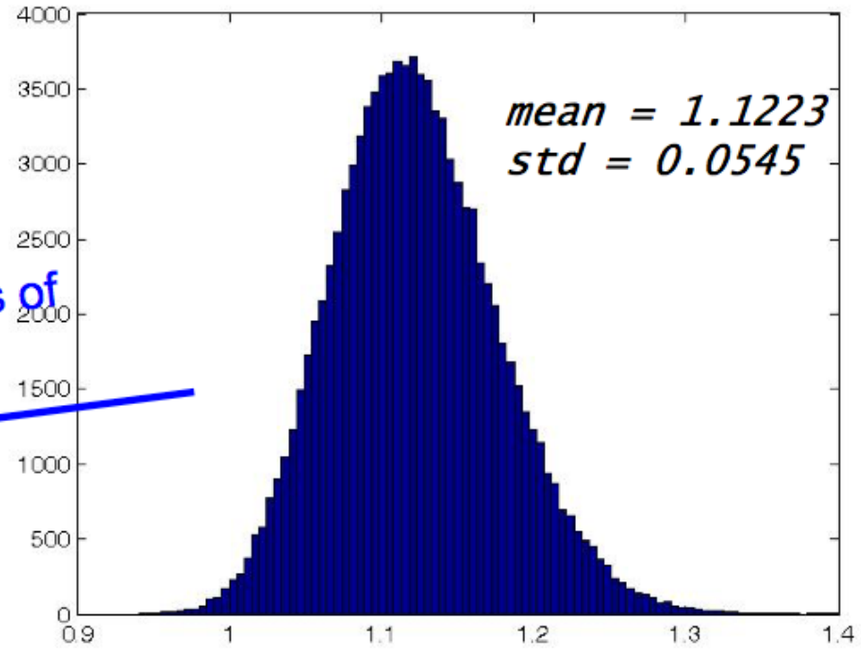
```
ndata = 100;
nboot = 100000;
vals = zeros(nboot,1);
for j=1:nboot,
   choose = randsample(ndata,ndata,true);
   vals(j) = mean(sample(choose))
          /median(sample(choose));
end
hist(vals,100)
```

new sample of integers in 1:ndata, with replacement



distorted by peculiarities of the particular data set

```
ndata = 100;
nboot = 100000;
vals = zeros(nboot,1);
for j=1:nboot,
   sam = randg(3,[ndata 1]);
   vals(j) = mean(sam)/median(sam);
end
hist(vals,100)
```



Things to notice:

The mean of resamplings does not improve the original estimate! (Same data!)

The distribution around the mean is not identical to that of the population. But it is close and would become identical asymptotically for large *ndata* (not *nboot*!).