Lecture 1

Describing Inverse Problems

Syllabus

Lecture 01	**Describing Inverse Problems**
Lecture 02	Probability and Measurement Error, Part 1
Lecture 03	Probability and Measurement Error, Part 2
Lecture 04	The L_2 Norm and Simple Least Squares
Lecture 05	A Priori Information and Weighted Least Squared
Lecture 06	Resolution and Generalized Inverses
Lecture 07	Backus-Gilbert Inverse and the Trade Off of Resolution and Variance
Lecture 08	The Principle of Maximum Likelihood
Lecture 09	Inexact Theories
Lecture 10	Nonuniqueness and Localized Averages
Lecture 11	Vector Spaces and Singular Value Decomposition
Lecture 12	Equality and Inequality Constraints
Lecture 13	L_1 , L_∞ Norm Problems and Linear Programming
Lecture 14	Nonlinear Problems: Grid and Monte Carlo Searches
Lecture 15	Nonlinear Problems: Newton's Method
Lecture 16	Nonlinear Problems: Simulated Annealing and Bootstrap Confidence Intervals
Lecture 17	Factor Analysis
Lecture 18	Varimax Factors, Empircal Orthogonal Functions
Lecture 19	Backus-Gilbert Theory for Continuous Problems; Radon's Problem
Lecture 20	Linear Operators and Their Adjoints
Lecture 21	Fréchet Derivatives
Lecture 22	Exemplary Inverse Problems, incl. Filter Design
Lecture 23	Exemplary Inverse Problems, incl. Earthquake Location
Lecture 24	Exemplary Inverse Problems, incl. Vibrational Problems

Purpose of the Lecture

distinguish forward and inverse problems

categorize inverse problems

examine a few examples

enumerate different kinds of solutions to inverse problems

Part 1

Lingo for discussing the relationship between observations and the things that we want to learn from them

three important definitions

data, $\mathbf{d} = [d_1, d_2, \ldots d_N]^{\mathrm{T}}$

things that are measured in an experiment or observed in nature…

model parameters, $\mathbf{m} = [m_1, m_2, \ldots m_M]^{\mathrm{T}}$

things you want to know about the world …

quantitative model (or *theory*)

relationship between data and model parameters data, $\mathbf{d} = [d_1, d_2, \dots d_N]^{\mathrm{T}}$

gravitational accelerations
travel time of seismic waves

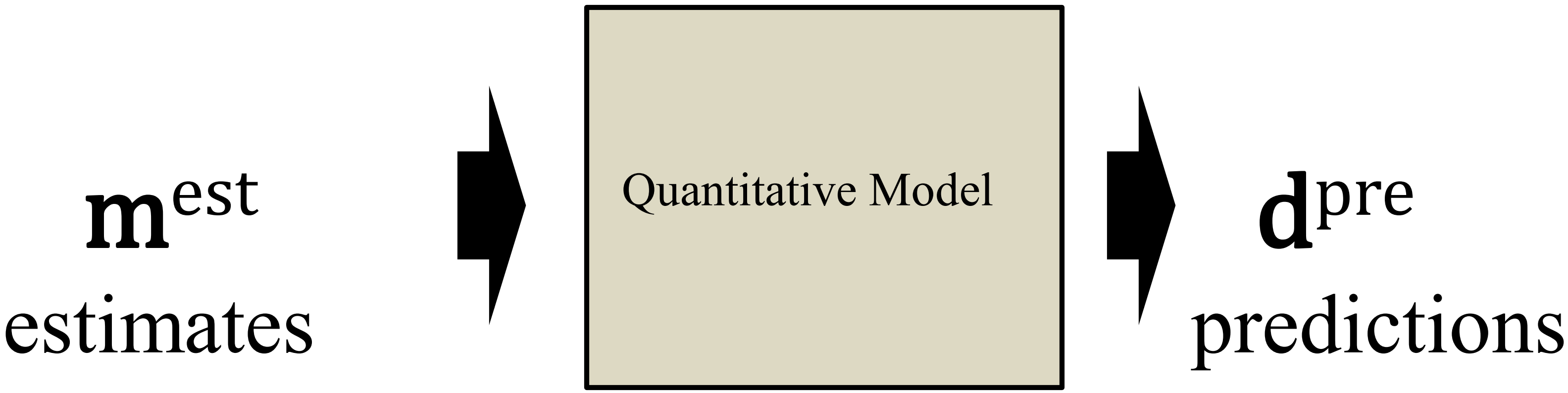
model parameters, $\mathbf{m} = [m_1, m_2, \dots m_M]^{\mathrm{T}}$

density
seismic velocity

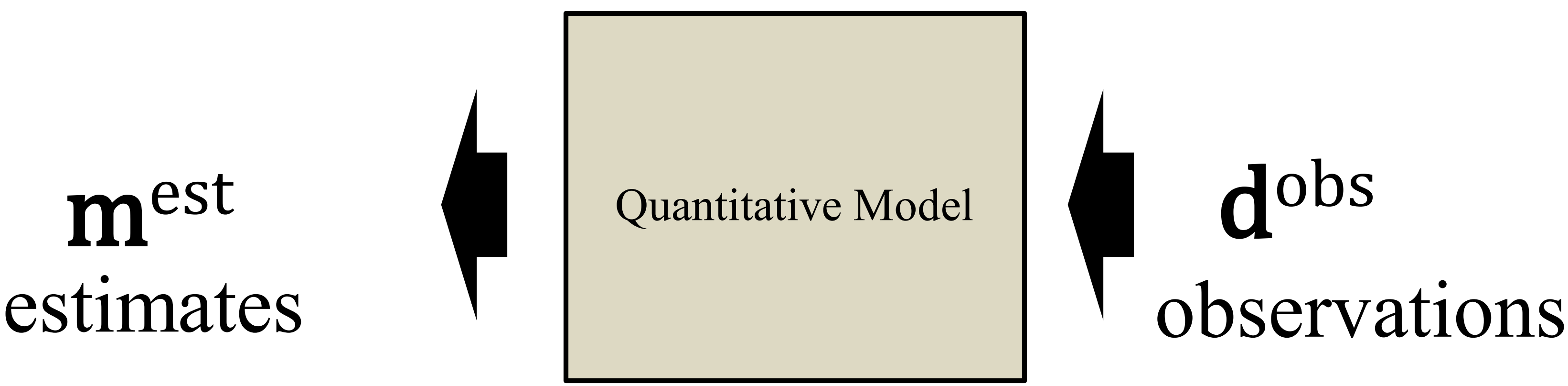
quantitative model (or *theory*)

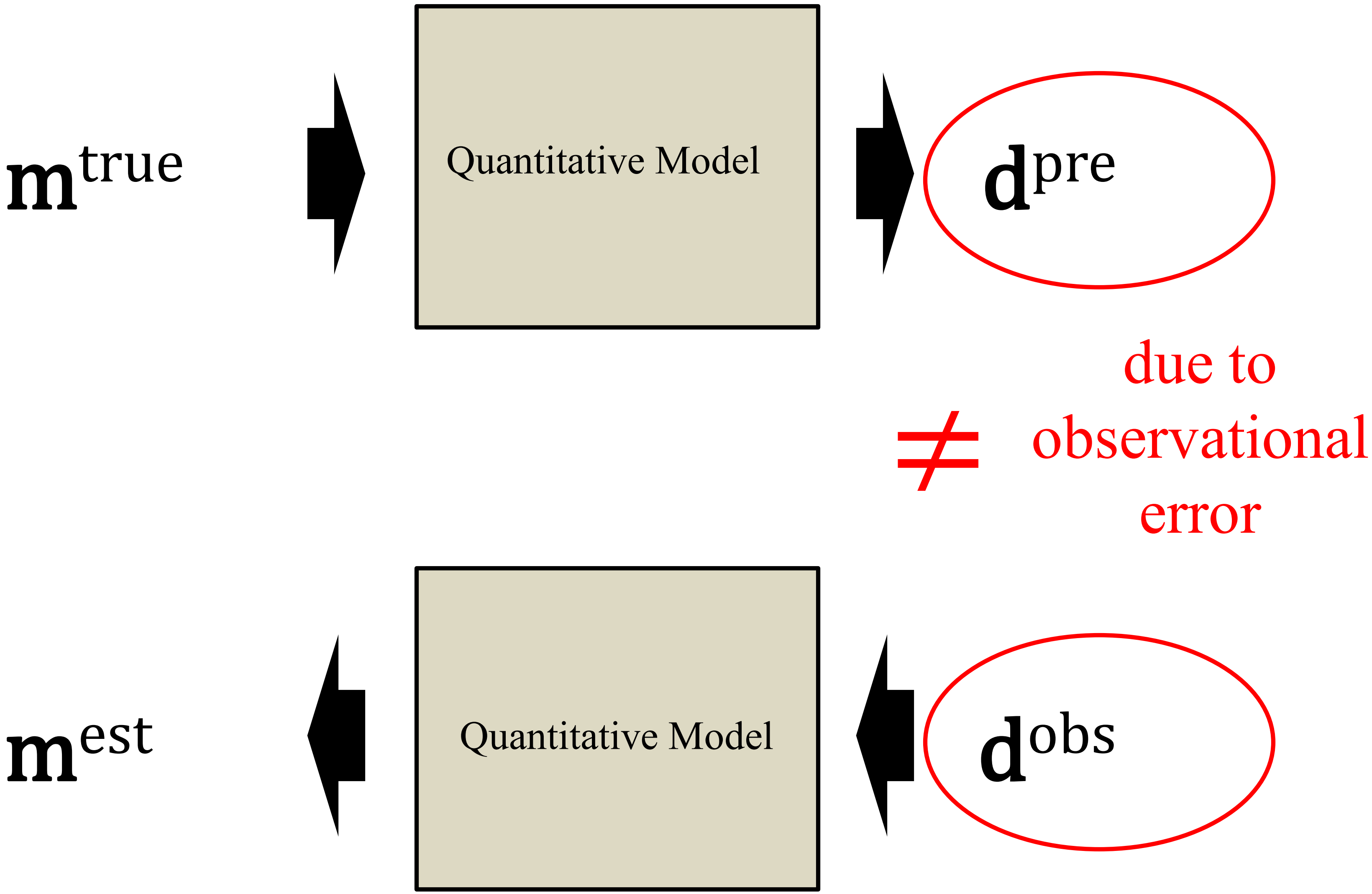
Newton's law of gravity
seismic wave equation

Forward Theory

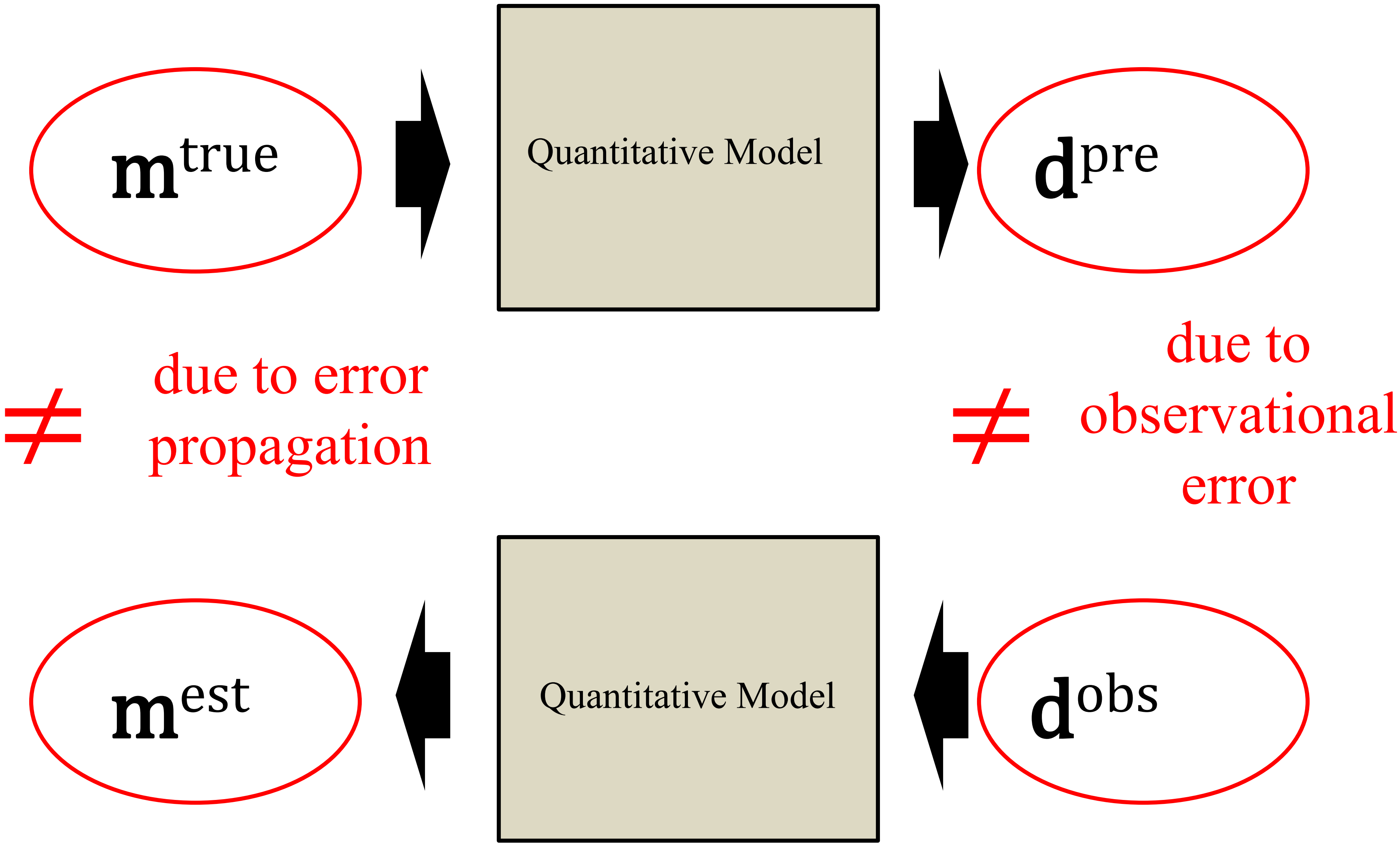


Inverse Theory





$\mathbf{m}^{true}$
Quantitative Model
$\mathbf{d}^{pre}$
due to
$\neq$
observational
error
$\mathbf{m}^{est}$
Quantitative Model
$\mathbf{d}^{obs}$



$\mathbf{m}^{\text{true}}$
Quantitative Model
$\mathbf{d}^{\text{pre}}$
≠
due to error propagation
≠
due to observational error
$\mathbf{m}^{\text{est}}$
Quantitative Model
$\mathbf{d}^{\text{obs}}$

Understanding the effects of
observational error
is central to Inverse Theory

Part 2

types of quantitative models (or *theories*)

A. Implicit Theory

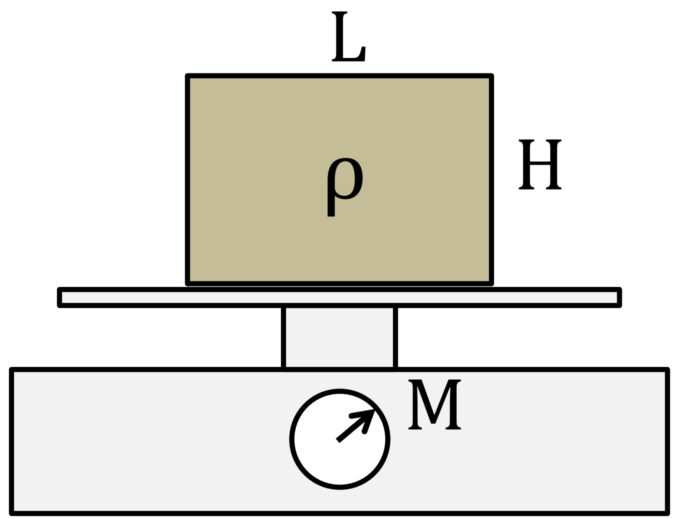
L relationships between the data and the model are known

$$\begin{aligned} f_1(\mathbf{d}, \mathbf{m}) &= 0 \\ f_2(\mathbf{d}, \mathbf{m}) &= 0 \\ &\vdots \\ f_L(\mathbf{d}, \mathbf{m}) &= 0 \end{aligned} \qquad \text{or} \qquad \mathbf{f}(\mathbf{d}, \mathbf{m}) = 0$$

Example

$$\text{mass} = \text{density} \times \text{length} \times \text{width} \times \text{height}$$

$$M = \rho \times L \times W \times H$$



$$\text{weight} = \text{density} \times \text{volume}$$

measure

mass, d_1

size, d_2, d_3, d_4,

$\mathbf{d}=[d_1, d_2, d_3, d_4]^T$ and $N=4$

want to know

density, m_1

$\mathbf{m}=[m_1]^T$ and $M=1$

$$d_1 = m_1 d_2 d_3 d_4 \quad \text{or} \quad d_1 - m_1 d_2 d_3 d_4 = 0$$

$f_1(\mathbf{d},\mathbf{m})=0$ and $L=1$

note

No guarantee that

$\mathbf{f}(\mathbf{d},\mathbf{m})=0$

contains enough information

for *unique* estimate **m**

determining whether or not there is enough

is part of the inverse problem

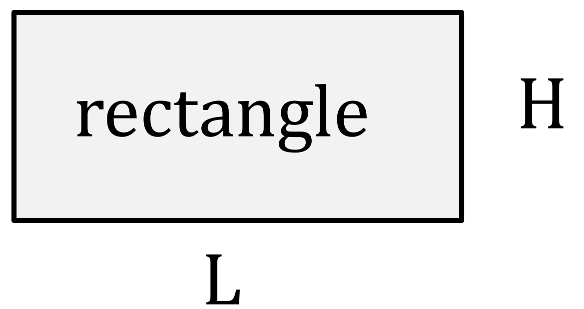
B. Explicit Theory

the equation can be arranged so that **d** is a function of **m**

$$\mathbf{d} = \mathbf{g}(\mathbf{m}) \quad \textbf{or} \quad \mathbf{d} - \mathbf{g}(\mathbf{m}) = 0$$

$L = N$ one equation per datum

Example



$$\text{Circumference} = 2 \times \text{length} + 2 \times \text{height}$$

$$\text{Area} = \text{length} \times \text{height}$$

$$\text{Circumference} = 2 \times \text{ length} + 2 \times \text{height}$$

$$C = 2L+2H$$

$$\text{Area} = \text{length} \times \text{height}$$

$$A=LH$$

measure

$C=d_1$

$A=d_2$

$\mathbf{d}=[d_1, d_2]^T$ and $N=2$

want to know

$L=m_1$

$H=m_2$

$\mathbf{m}=[m_1, m_2]^T$ and $M=2$

$$\begin{pmatrix} d_1 \\ d_2 \end{pmatrix} = \begin{pmatrix} 2m_1 + 2m_2 \\ m_1 m_2 \end{pmatrix}$$

d=g(m)

C. Linear Explicit Theory

the function $\mathbf{g}(\mathbf{m})$ is a matrix $\mathbf{G}$ times $\mathbf{m}$

$$\mathbf{d} = \mathbf{G}\mathbf{m}$$

$\mathbf{G}$ has N rows and M columns

C. Linear Explicit Theory

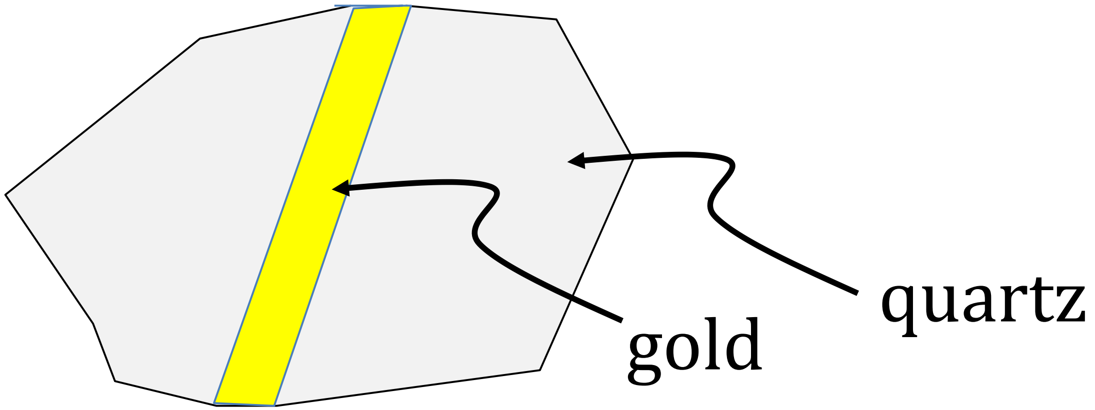
the function $\mathbf{g}(\mathbf{m})$ is a matrix $\mathbf{G}$ times $\mathbf{m}$

$$\mathbf{d} = \mathbf{G}\mathbf{m}$$

"data kernel"

$\mathbf{G}$ has N rows and M columns

Example



total volume = volume of gold + volume of quartz

$$V = V_g + V_q$$

total mass = density of gold × volume of gold
+ density of quartz × volume of quartz

$$M = \rho_g \times V_g + \rho_q \times V_q$$

$$V = V_g + V_q$$

$$M = \rho_g \times V_g + \rho_q \times V_q$$

measure

$V = d_1$

$M = d_2$

want to know

$V_g = m_1$

$V_q = m_2$

assume

$$\left.\begin{matrix}\rho_g \\ \rho_g\end{matrix}\right\}\text{ known}$$

$\mathbf{d} = [d_1, d_2]^T$ and $N=2$

$\mathbf{m} = [m_1, m_2]^T$ and $M=2$

$$\mathbf{d} = \begin{pmatrix} 1 & 1 \\ \rho_g & \rho_q \end{pmatrix} \mathbf{m}$$

D. Linear Implicit Theory

The L relationships between the data are linear

$$\mathbf{f}(\mathbf{d},\mathbf{m}) = 0 = \mathbf{F}\begin{bmatrix}\mathbf{d}\\ \mathbf{m}\end{bmatrix} = \mathbf{F}\mathbf{x}$$

L rows
$N+M$ columns

in all these examples **m** is discrete

$$d_i = \sum_{j=1}^{M} G_{ij} m_j$$

discrete inverse theory

one could have a continuous *m(x)* instead

$$d_i = \int G_i(x)\, m(x)\ \mathrm{d}x$$

continuous inverse theory

in this course we will usually approximate a continuous $m(x)$

as a discrete vector $\mathbf{m}$

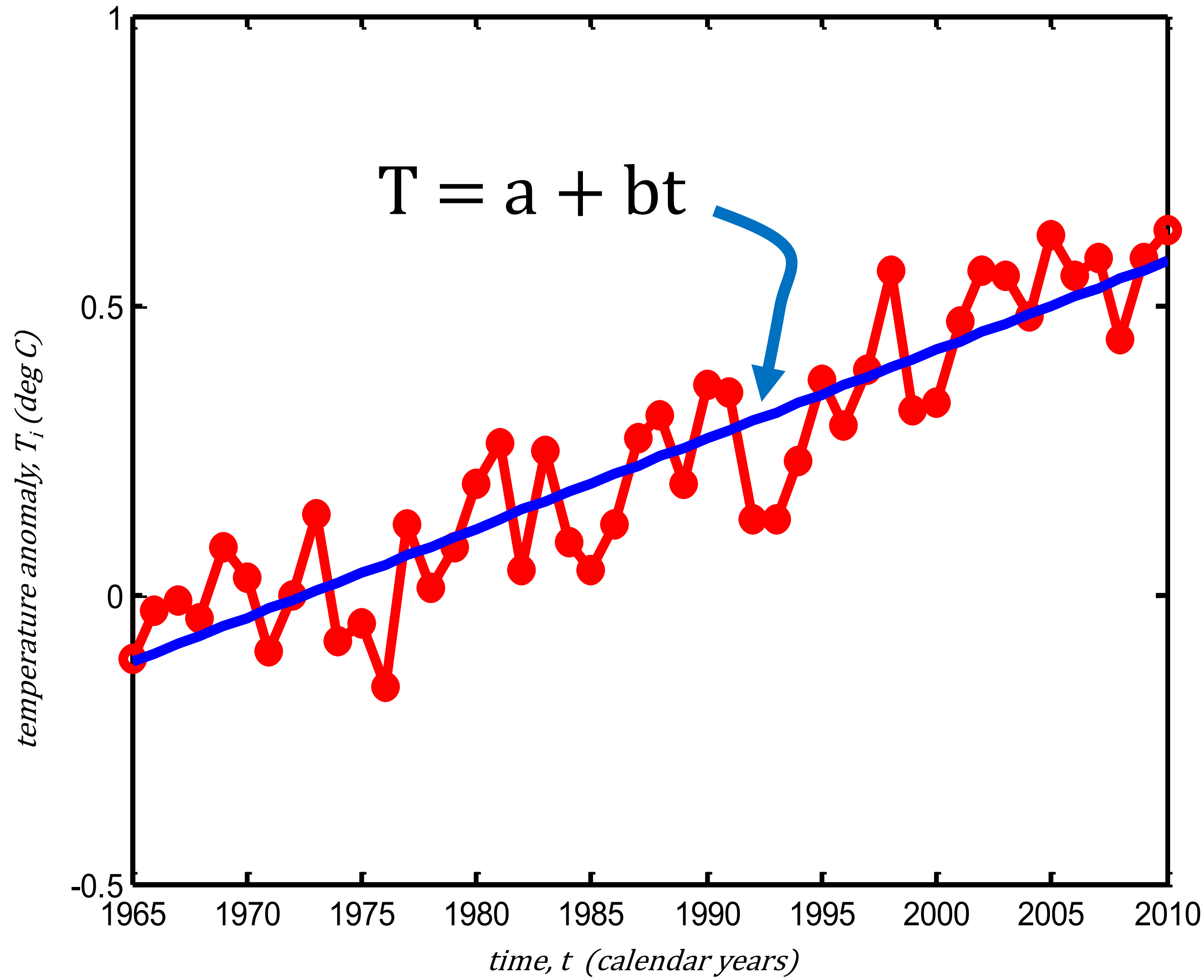
$$\mathbf{m} = [m(\Delta x), m(2\Delta x), m(3\Delta x) \ \ldots \ m(M\Delta x)]^T$$

but we will spend some time later in the course dealing with the continuous problem directly

Part 3

Some Examples

A. Fitting a straight line to data



$$
\begin{aligned}
T_1 &= a + bt_1 \\
T_2 &= a + bt_2 \\
&\vdots \\
T_N &= a + bt_N
\end{aligned}
$$

each data point

is predicted by a
straight line

matrix formulation

$$\begin{bmatrix} T_1 \\ T_2 \\ \vdots \\ T_N \end{bmatrix} = \begin{bmatrix} 1 & t_1 \\ 1 & t_2 \\ \vdots & \vdots \\ 1 & t_N \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix}$$

$$\mathbf{d} = \mathbf{G} \underset{M=2}{\mathbf{m}}$$

B. Fitting a parabola

$$T = a + bt + ct^2$$

$$
\begin{aligned}
T_1 &= a + bt_1 + ct_1^2 \\
T_2 &= a + bt_2 + ct_2^2 \\
&\vdots \\
T_N &= a + bt_N + ct_N^2
\end{aligned}
$$

each data point

is predicted by a strquadratic curve

matrix formulation

$$\begin{bmatrix} T_1 \\ T_2 \\ \vdots \\ T_N \end{bmatrix} = \begin{bmatrix} 1 & t_1 & t_1^2 \\ 1 & t_2 & t_2^2 \\ \vdots & \vdots & \vdots \\ 1 & t_N & t_N^2 \end{bmatrix} \begin{bmatrix} a \\ b \\ c \end{bmatrix}$$

$$\mathbf{d} = \mathbf{G} \quad \mathbf{m}$$

$M=3$

straight line

$$\begin{bmatrix} T_1 \\ T_2 \\ \vdots \\ T_N \end{bmatrix} = \begin{bmatrix} 1 & t_1 \\ 1 & t_2 \\ \vdots & \vdots \\ 1 & t_N \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix}$$

parabola

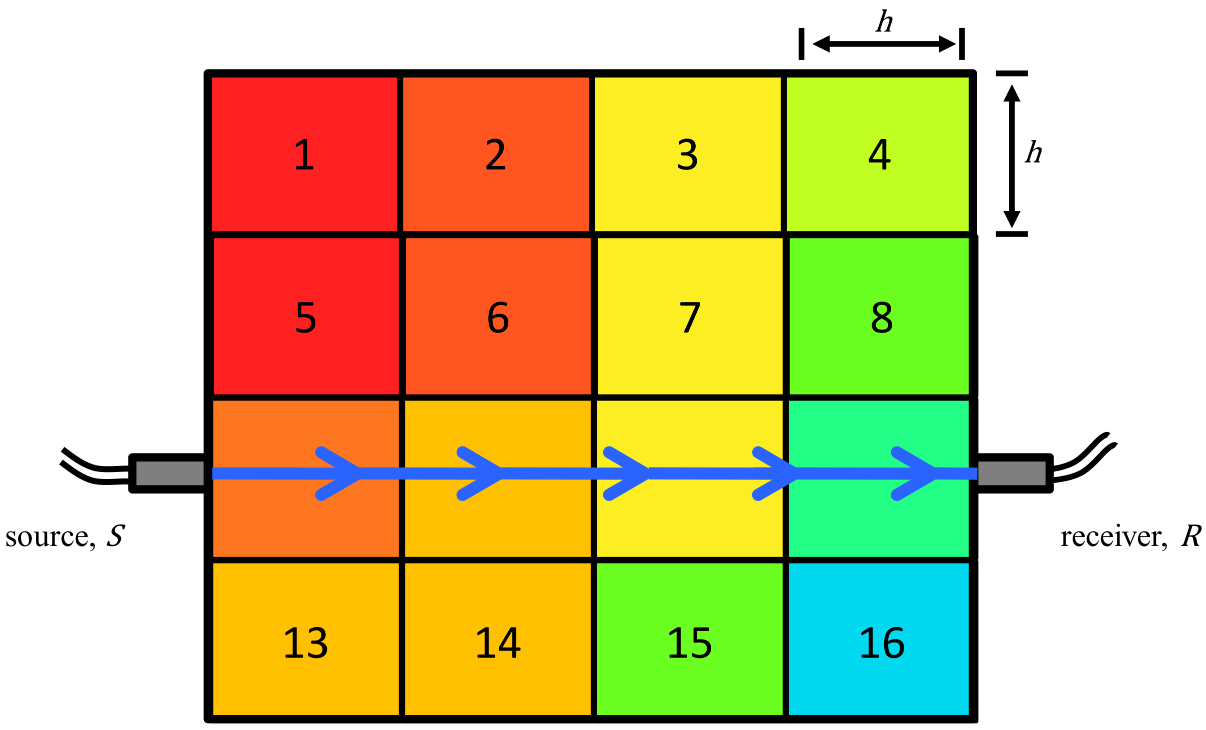
$$\begin{bmatrix} T_1 \\ T_2 \\ \vdots \\ T_N \end{bmatrix} = \begin{bmatrix} 1 & t_1 & t_1^2 \\ 1 & t_2 & t_2^2 \\ \vdots & \vdots & \vdots \\ 1 & t_N & t_N^2 \end{bmatrix} \begin{bmatrix} a \\ b \\ c \end{bmatrix}$$

note similarity

in *MatLab*

```
G=[ones(N,1), t, t.^2];
```

C. Acoustic Tomography



travel time = length × slowness

collect data along rows and columns

row 1: $T_1 = hs_1 + hs_2 + hs_3 + hs_4$

row 2: $T_2 = hs_5 + hs_6 + hs_7 + hs_8$

$\vdots$ $\vdots$

column 4: $T_8 = hs_4 + hs_8 + hs_{12} + hs_{16}$

matrix formulation

$$\begin{bmatrix} T_1 \\ T_2 \\ \vdots \\ T_8 \end{bmatrix} = h \begin{bmatrix} 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} s_1 \\ s_2 \\ \vdots \\ s_{16} \end{bmatrix}$$

$$\mathbf{d} = \mathbf{G} \mathbf{m}$$

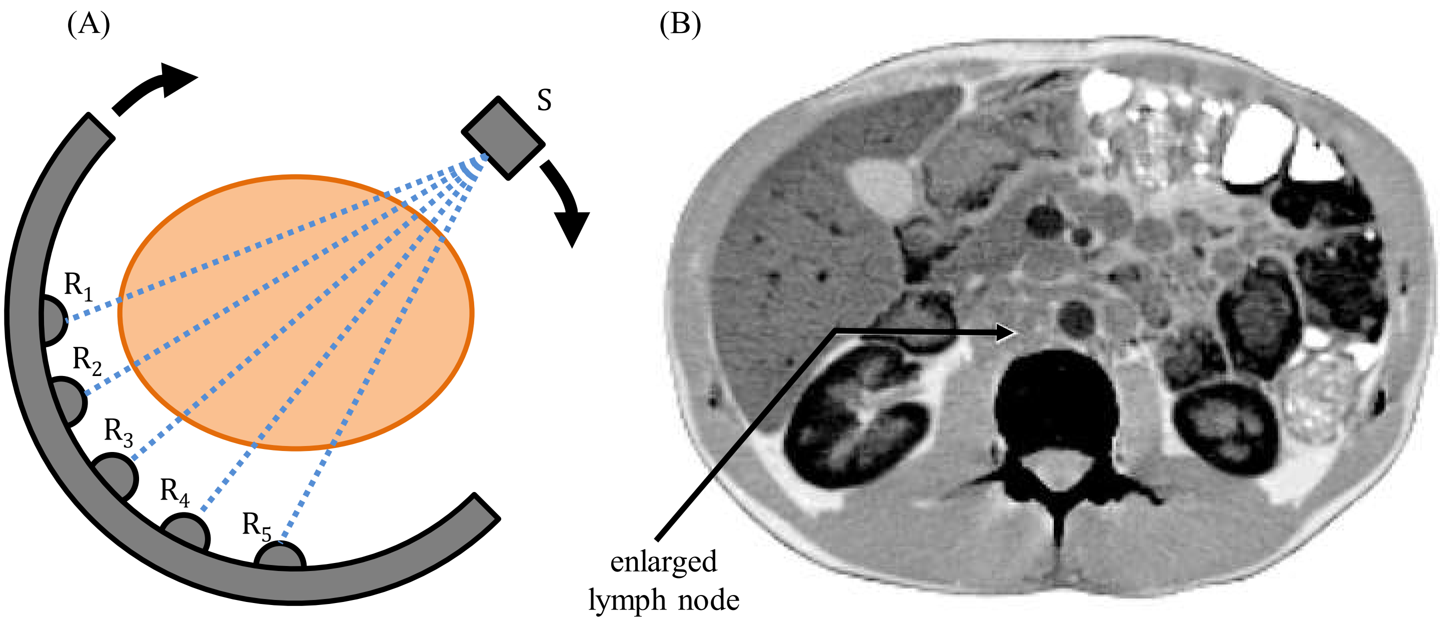
$N=8$

$M=16$

In *MatLab*

```
G=zeros(N,M);
for i = [1:4]
for j = [1:4]
     % measurements over rows
     k = (i-1)*4 + j;
     G(i,k)=1;
     % measurements over columns
     k = (j-1)*4 + i;
     G(i+4,k)=1;
end
end
```

D. X-ray Imaging



theory

$$\frac{\mathrm{d}I}{\mathrm{d}s} = -c(x,y)\, I$$

I = Intensity of x-rays (data)
s = distance
c = absorption coefficient (model parameters)

$$I_i = I_0 \exp\left\{-\int_{\text{beam } i} c(x,y)\,\mathrm{d}s\right\}$$

$$\ln I_0 - \ln I_i = \int_{\text{beam } i} c(x,y)\,\mathrm{d}s$$

Taylor Series approximation

$$\Delta I_i = \frac{I_0 - I_i}{I_0} = \sum_{j=1}^{M} \Delta s_{ij}\, c_j$$

$$I_i = I_0 \exp\left\{-\int_{\mathrm{beam}\ i} c(x,y)\,\mathrm{d}s\right\}$$

$$\ln I_0 - \ln I_i = \int_{\mathrm{beam}\ i} c(x,y)\,\mathrm{d}s$$

Taylor Series approximation

discrete pixel approximation

$$\Delta I_i = \frac{I_0 - I_i}{I_0} = \sum_{j=1}^{M} \Delta s_{ij}\, c_j$$

$$I_i = I_0 \exp\left\{ -\int_{\text{beam } i} c(x,y)\,\mathrm{d}s \right\}$$

$$\ln I_0 - \ln I_i = \int_{\text{beam } i} c(x,y)\,\mathrm{d}s$$

Taylor Series approximation

discrete pixel approximation

$$\Delta I_i = \frac{I_0 - I_i}{I_0} = \sum_{j=1}^{M} \Delta s_{ij}\, c_j$$

length of beam i in pixel j

$$\mathbf{d} \qquad = \qquad \mathbf{G}\,\mathbf{m}$$

matrix formulation

$$\begin{bmatrix} \Delta I_1 \\ \Delta I_2 \\ \cdots \\ \Delta I_N \end{bmatrix} = \begin{bmatrix} \Delta s_{11} & \Delta s_{12} & \cdots & \Delta s_{1M} \\ \Delta s_{21} & \Delta s_{22} & \cdots & \Delta s_{2M} \\ \cdots & \cdots & \cdots & \cdots \\ \Delta s_{N1} & \Delta s_{N2} & \cdots & \Delta s_{NM} \end{bmatrix} \begin{bmatrix} c_1 \\ c_2 \\ \cdots \\ c_M \end{bmatrix}$$

$$\mathbf{d} = \mathbf{G} \quad \mathbf{m}$$

$N \approx 10^6$

$M \approx 10^6$

note that $\mathbf{G}$ is huge

$10^6 \times 10^6$

but it is sparse

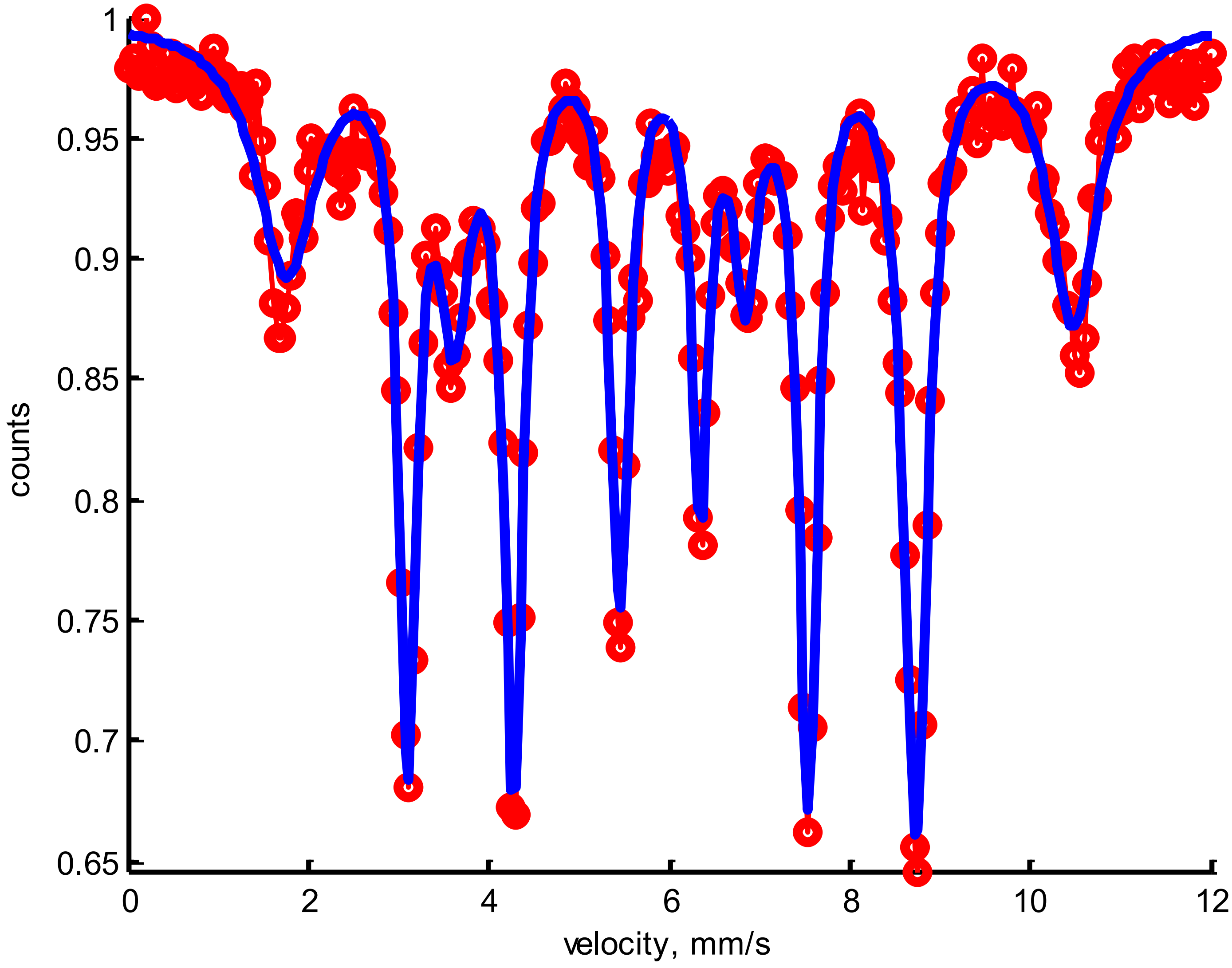
(mostly zero)

since a beam passes through only a tiny fraction of the total number of pixels

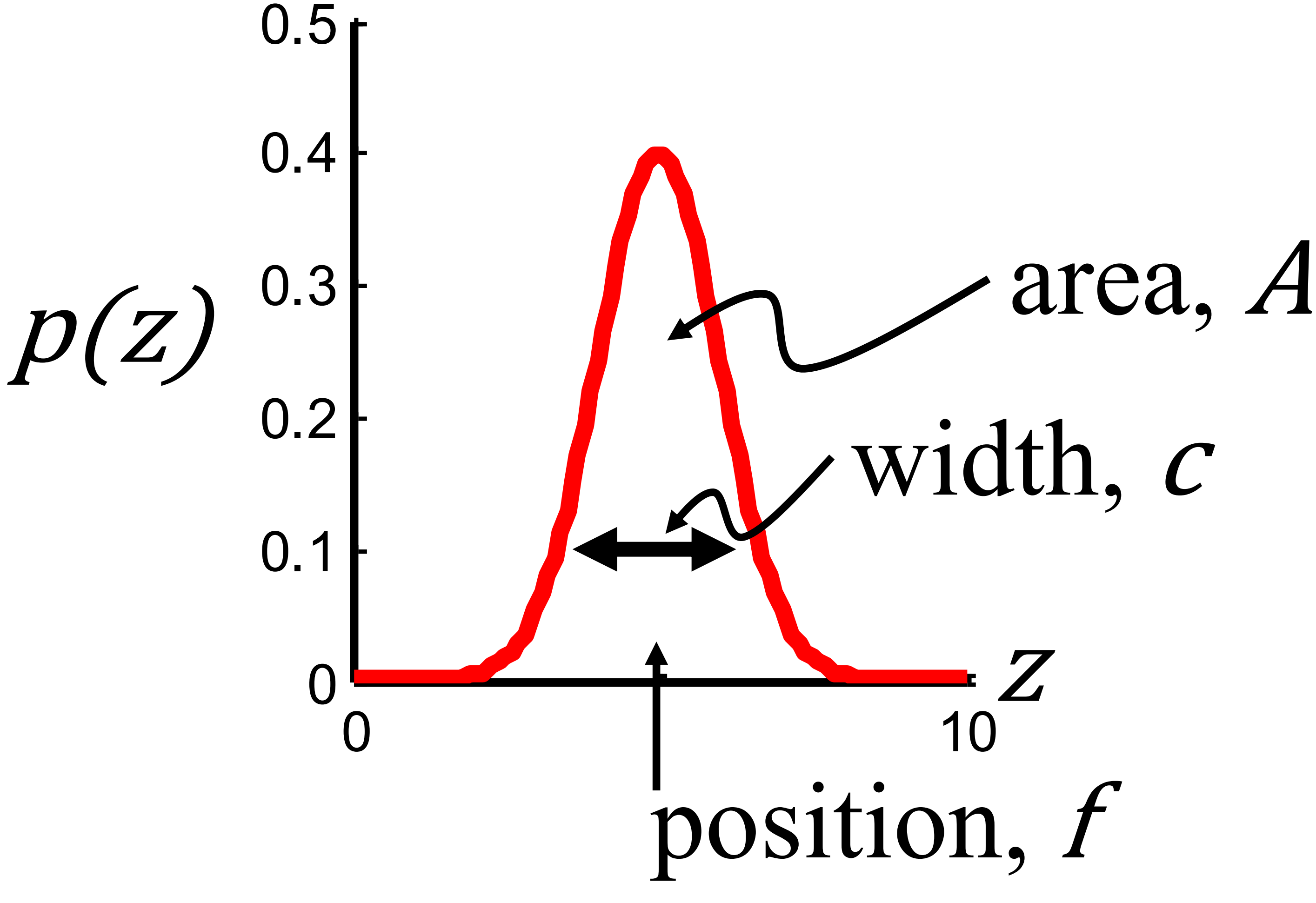
in *MatLab*

```
G = spalloc( N, M, MAXNONZEROELEMENTS);
```

E. Spectral Curve Fitting



single spectral peak



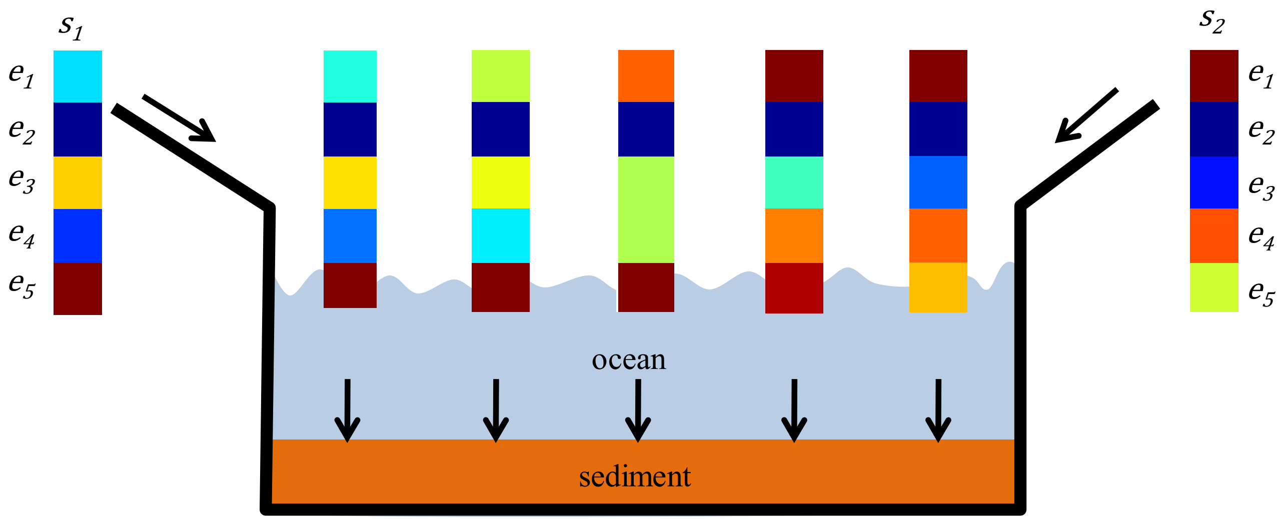
q spectral peaks

"Lorentzian"

$$d_i = \sum_{j=1}^{q} p(z_i, f_j, A_j, c_j) = \sum_{j=1}^{q} \frac{A_j c_j^2}{(z_i - f_j)^2 + c_j^2}$$

$$\mathbf{d} = \mathbf{g}(\mathrm{m})$$

F. Factor Analysis



$$\begin{bmatrix}\text{sample}\\ \text{composition}\end{bmatrix} = \sum_{\substack{\text{end}\\ \text{members}}} \begin{bmatrix}\text{amount of}\\ \text{end member}\end{bmatrix} \begin{bmatrix}\text{end member}\\ \text{composition}\end{bmatrix}$$

$$\mathbf{d} = \mathbf{g}(\mathbf{m})$$

Part 4

What kind of solution are we looking for ?

A: Estimate of model parameters

meaning numerical values

$$m_1 = 10.5$$
$$m_2 = 7.2$$

But we really need confidence limits, too

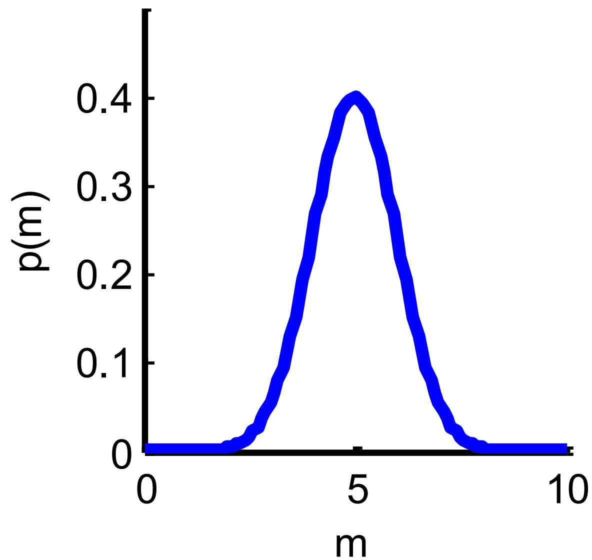
$$m_1 = 10.5 \pm 0.2$$
$$m_2 = 7.2 \pm 0.1$$

or

$$m_1 = 10.5 \pm 22.3$$
$$m_2 = 7.2 \pm 9.1$$

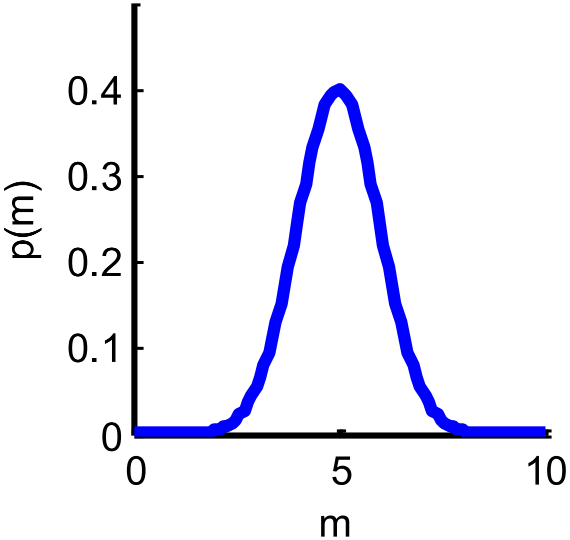
completely different implications!

B: probability density functions

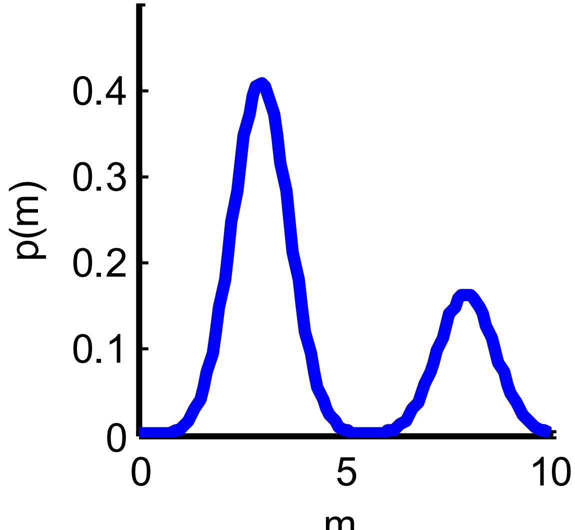


if $p(m_1)$ simple

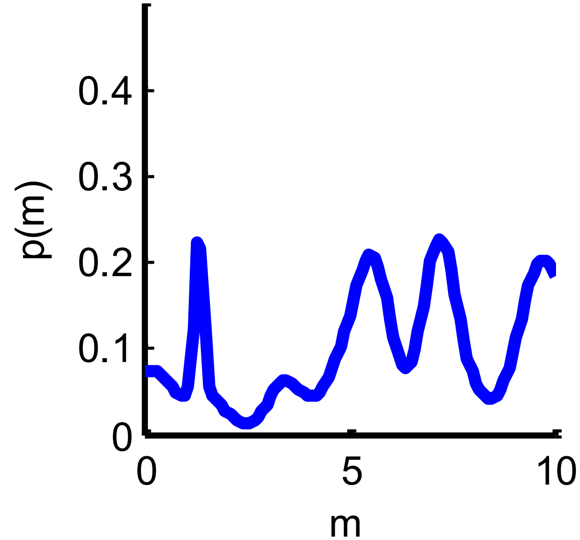
not so different than confidence intervals



m is about 5 plus or minus 1.5



m is either about 3 plus of minus 1 or about 8 plus or minus 1 but that's less likely



we don't really know anything useful about m

C: localized averages

$$A = 0.2m_9 + 0.6m_{10} + 0.2m_{11}$$

might be better determined than either m_9 or m_{10} or m_{11} individually

Is this useful?

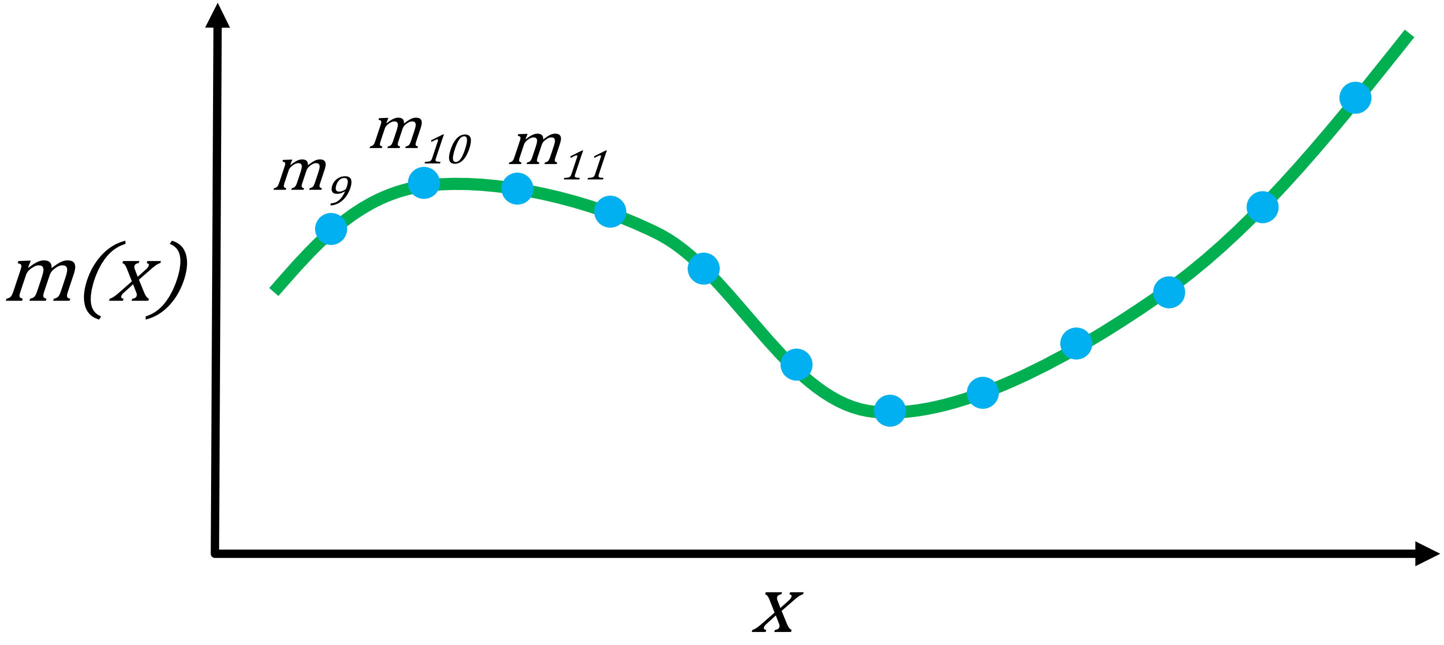
Do we care about

$$A = 0.2m_9 + 0.6m_{10} + 0.2m_{11}$$

?

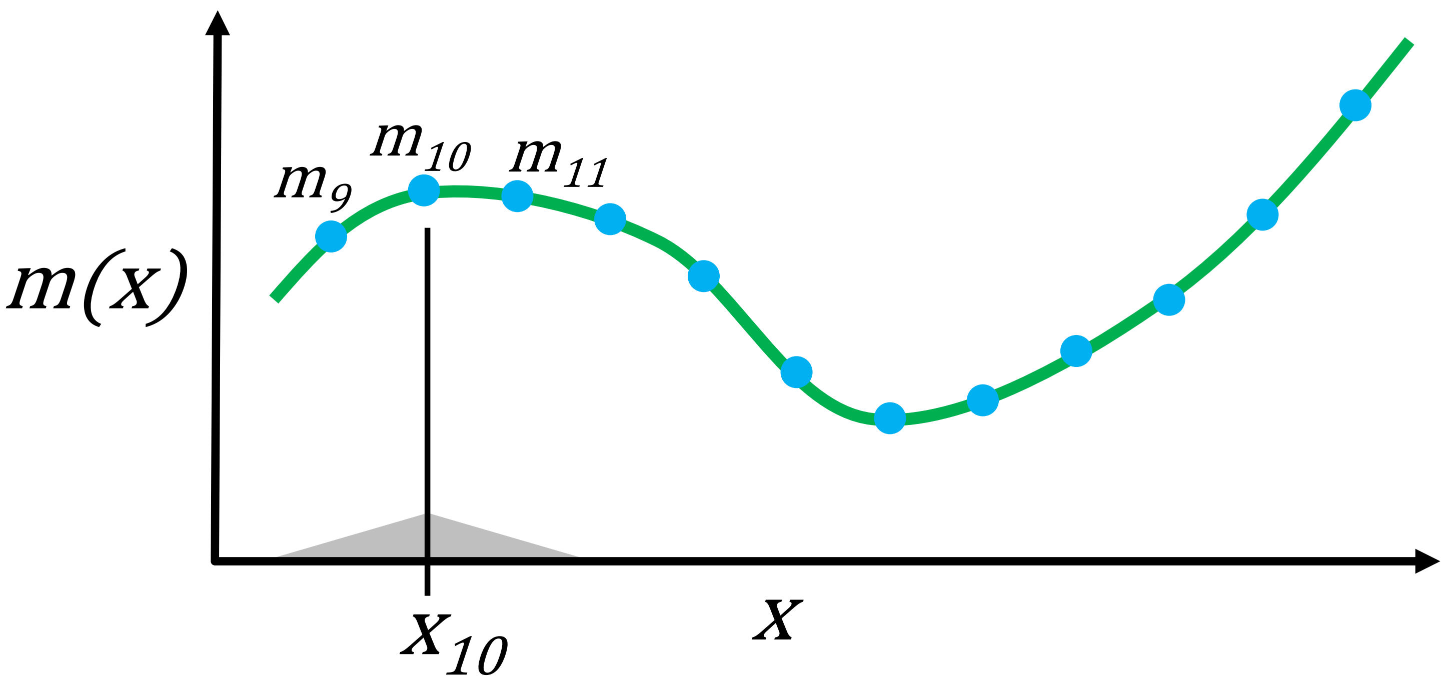
Maybe …

Suppose
if **m** is a discrete approximation of *m(x)*

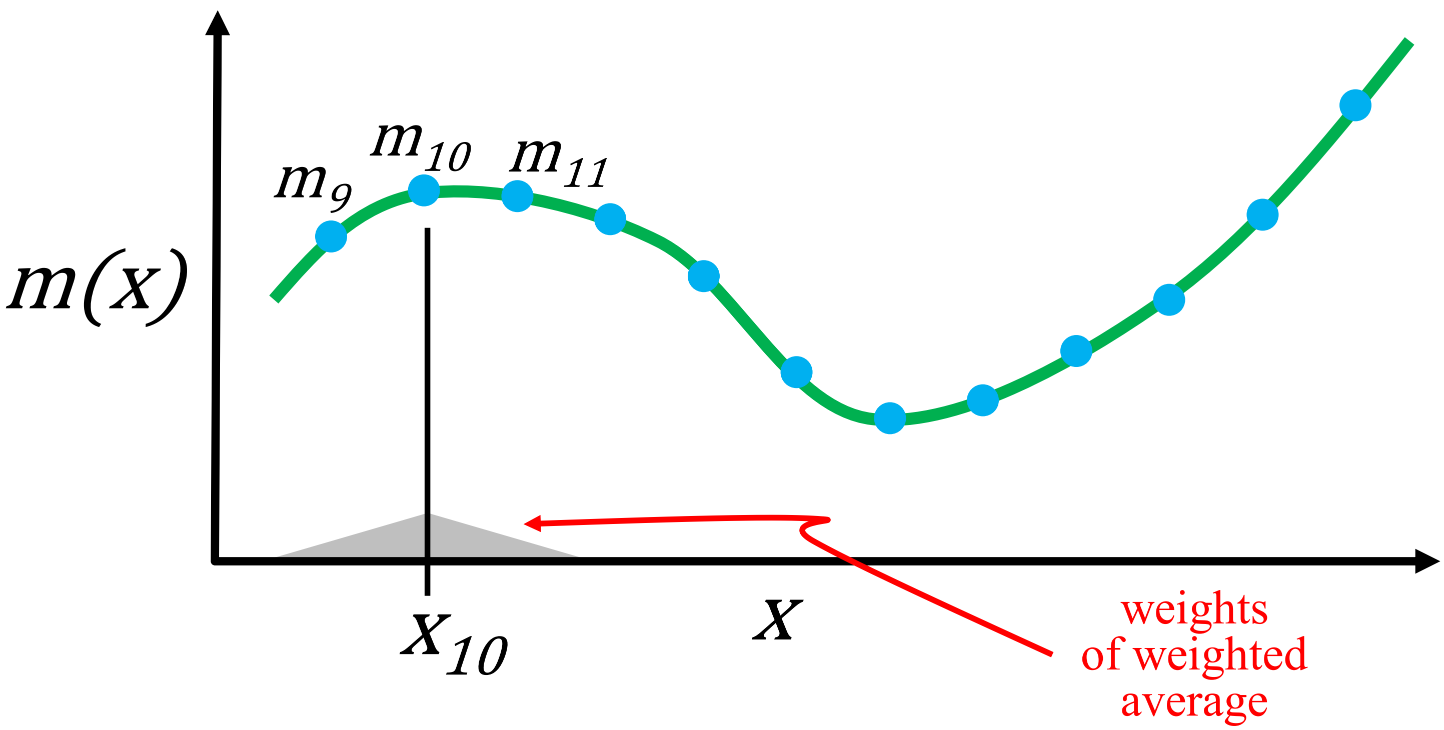


$$A = 0.2m_9 + 0.6m_{10} + 0.2m_{11}$$

weighted average of $m(x)$ in the vicinity of x_{10}



average "localized' in the vicinity of x_{10}



Localized average mean
can't determine $m(x)$ *at* $x=10$
but can determine
average value of $m(x)$ *near* $x=10$

Such a localized average might very well be useful