ECE295, Data Assimilation and Inverse Problems, Spring 2015

Peter Gerstoft, 534-7768, gerstoft@ucsd.edu

We meet Wednesday from 5 to 6:20pm in HHS 2305A

Text for first 5 classes: Parameter Estimation and Inverse Problems (2nd Edition) here under UCSD license

Grading S

Classes

1 April, Intro; Linear discrete Inverse problems (Aster Ch 1, 2)

8 April, SVD (Aster ch 2 and 3)

15 April, Regularization (ch 4) Numerical Example: **Beamforming**

22 April, Sparse methods (ch 7.2-7.3)

29 April, Sparse methods

6 May, Bayesian methods and Monte Carlo methods (ch 11) Numerical Example: **Ice–flow from GPS**

13 May, Introduction to sequential Bayesian methods, Kalman Filter

20 May, Data assimilation, EnKF

27 May, EnKF, Data assimilation

3 June, Markov Chain Monte Carlo, PF

Homework: Call the files LastName_ExXX.

Homework is due 8am on Wednesday. That way we can discuss in class.

Hw 1: Download the matlab codes for the book (cd_5.3). Run the 3 examples for chapter 2. Come to class with one question about the examples

Hw2: Based on Example 3.3. Adapt it to a complex valued beamforming example.

% Parameters

c = 1500; % speed of sound

f = 200; % frequency

lambda = c/f; % wavelength

Beamforming example

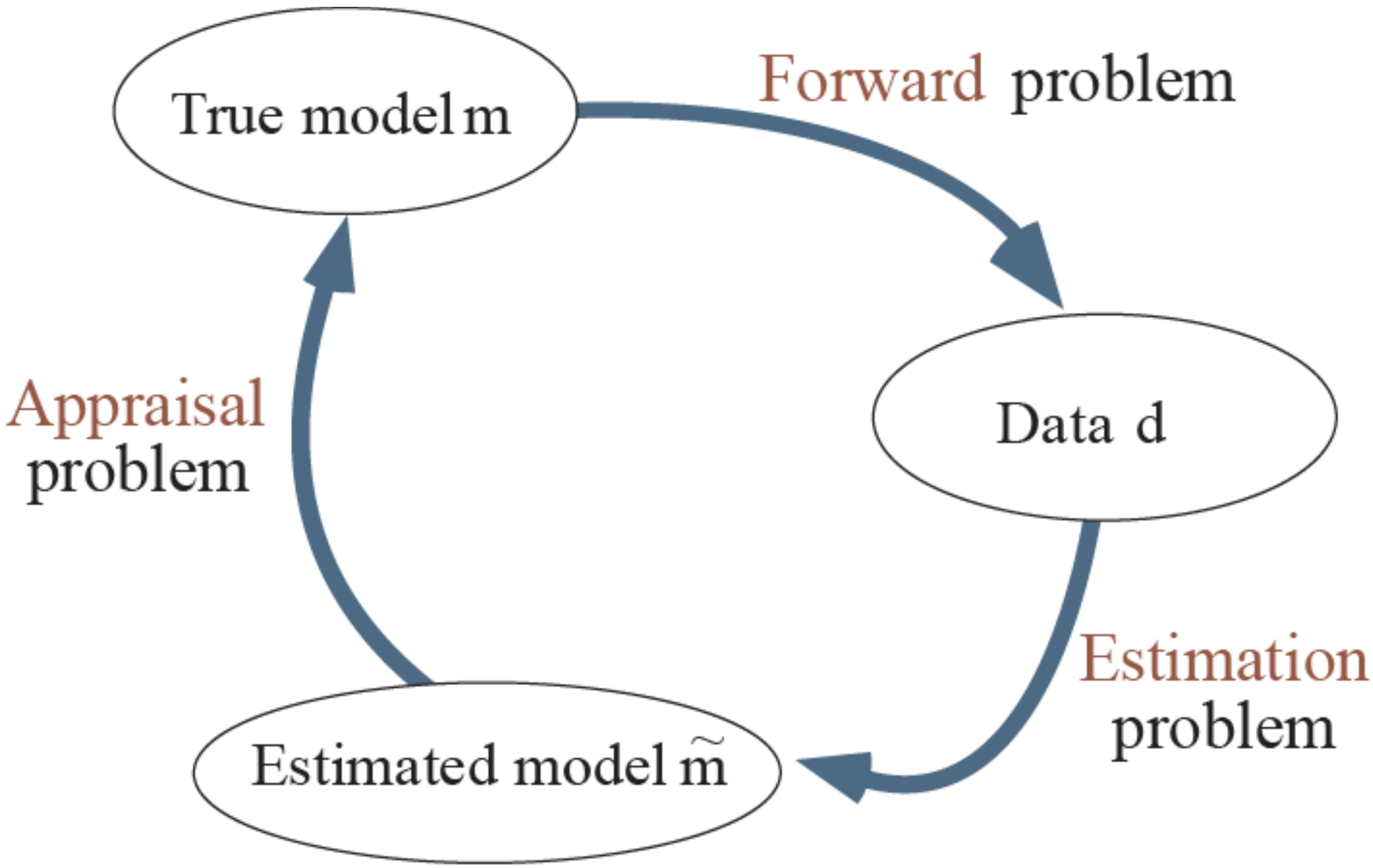
```
% Parameters
c = 1500;       % speed of sound
f = 200;        % frequency
lambda = c/f;   % wavelength
k = 2*pi/lambda;% wavenumber

% ULA-horizontal
N = 20;         % number of sensors
d = 1/2*lambda; % intersensor spacing
q = [0:1:(N-1)];% sensor numbering
xq = (q-(N-1)/2)*d; % sensor locations
% Bearing grid
theta = [-90:0.5:90];
u = sind(theta);

% Represenation matrix (steering matrix)
A = exp(-1i*2*pi/lambda*xq'*u)/sqrt(N);
```

- REVIEW

Estimation and Appraisal



Over-determined: Linear discrete inverse problem

We seek the model vector **m** which minimizes

Compare with maximum likelihood

$$\phi(\boldsymbol{m}) = \frac{1}{2}\boldsymbol{r}^T C_d^{-1}\boldsymbol{r} = \frac{1}{2}(\boldsymbol{d} - G\boldsymbol{m})^T C_d^{-1}(\boldsymbol{d} - G\boldsymbol{m})$$

Note that this is a quadratic function of the model vector.

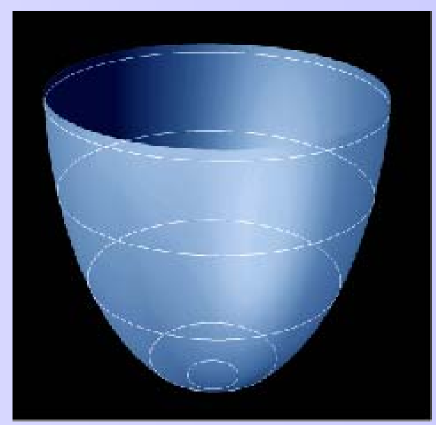
Solution: Differentiate with respect to **m** and solve for the model vector which gives a zero gradient in $\phi(\boldsymbol{m})$

This gives…

$$\nabla\phi(\boldsymbol{m}) = -G^T C_d^{-1}(\boldsymbol{d} - G\boldsymbol{m}) = 0$$

$$\Rightarrow \boldsymbol{m} = (G^T C_d^{-1} G)^{-1} G^T C_d^{-1}\boldsymbol{d}$$

This is the least-squares solution.



A solution to the normal equations:

$$G^T G\boldsymbol{m} = G^T\boldsymbol{d}$$

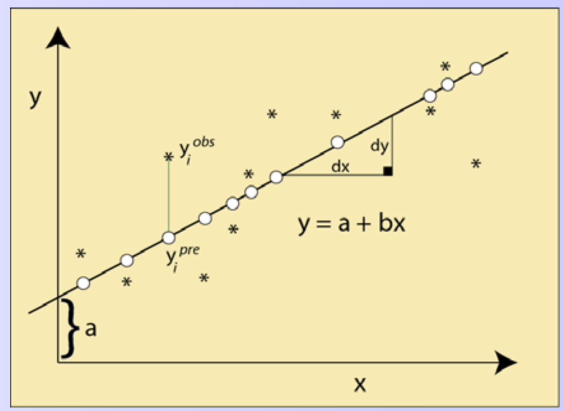
Over-determined: Linear discrete inverse problem

How does the Least-squares solution compare to the standard equations of linear regression ?

$$m = (G^T C_d^{-1} G)^{-1} G^T C_d^{-1} d$$

Given N data y_i with independent normally distributed errors and standard deviations σ_i what are the expressions for the model parameters **m** = $[a,b]^T$?

$$Gm = \begin{bmatrix} 1 & x_1 \\ 1 & x_2 \\ \vdots & \vdots \\ 1 & x_{N_d} \end{bmatrix} \begin{bmatrix} m_1 \\ m_2 \end{bmatrix} = \begin{bmatrix} d_1 \\ d_2 \\ \vdots \\ d_N \end{bmatrix} = d$$



$$m = \begin{bmatrix} N & \sum_{i=1}^N x_i \\ \sum_{i=1}^N x_i & \sum_{i=1}^N x_i^2 \end{bmatrix}^{-1} \begin{bmatrix} \sum_{i=1}^N y_i \\ \sum_{i=1}^N x_i y_i \end{bmatrix}$$

$$m = \frac{1}{N\sum_{i=1}^N x_i^2 - \left(\sum_{i=1}^N x_i\right)^2} \begin{bmatrix} \sum_{i=1}^N x_i^2 & -\sum_{i=1}^N x_i \\ -\sum_{i=1}^N x_i & N \end{bmatrix} \begin{bmatrix} \sum_{i=1}^N y_i \\ \sum_{i=1}^N x_i y_i \end{bmatrix}$$

Linear discrete inverse problem: Least squares

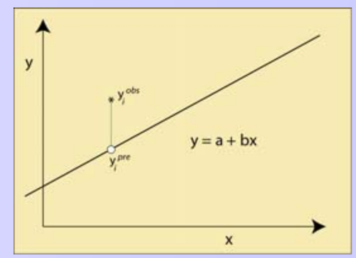
$$\boxed{\boldsymbol{m}_{LS} = (G^T C_d^{-1} G)^{-1} G^T C_d^{-1} \boldsymbol{d} = G^{-g} \boldsymbol{d}}$$

What happens in the under and even-determined cases ?

$$\boldsymbol{m} = \frac{1}{N \sum_{i=1}^N x_i^2 - \left(\sum_{i=1}^N x_i\right)^2} \begin{bmatrix} \sum_{i=1}^N x_i^2 & -\sum_{i=1}^N x_i \\ -\sum_{i=1}^N x_i & N \end{bmatrix} \begin{bmatrix} \sum_{i=1}^N y_i \\ \sum_{i=1}^N x_i y_i \end{bmatrix}$$

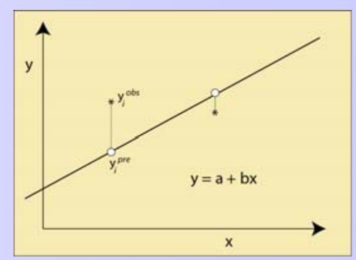
- Under-determined, N=1:

Matrix has a zero determinant
and a zero eigenvalue
an infinite number of solutions exist



- Even-determined, N=2:

$$\boldsymbol{m} = [m_1, m_2]^T, \quad m_2 = \left[\frac{y_1 - y_2}{x_1 - x_2}\right], \quad m_1 = y_1 - m_2 x_1.$$



$$\boxed{r = d - Gm = 0}$$

Prediction error is zero !

Example: Over-determined, Linear discrete inverse problem

The Ballistics example

$$y_i = m_1 + m_2 t_i - \frac{1}{2} m_3 t_i^2$$

Given data and noise

t	y
1	109:3827
2	187:5385
3	267:5319
4	331:8753
5	386:0535
6	428:4271
7	452:1644
8	498:1461
9	512:3499
10	512:9753

$$C_d^{-1} = \frac{1}{\sigma^2} I$$

$$\sigma = 8m$$

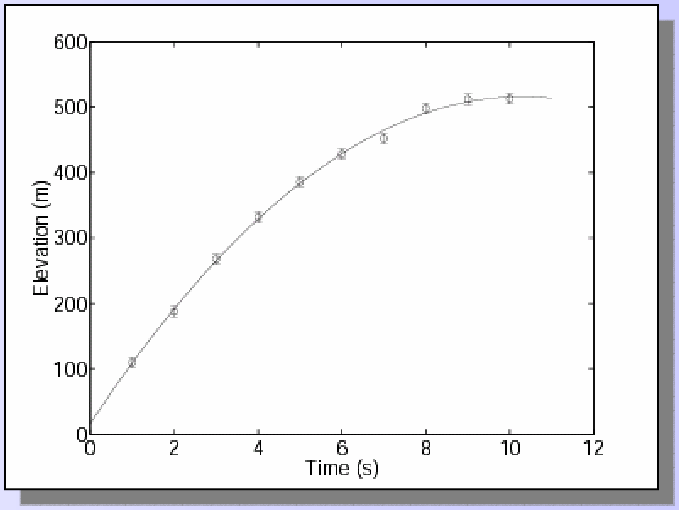
Calculate G

$$G = \begin{pmatrix} 1 & t_1 & -1/2t_1^2 \\ 1 & t_2 & -1/2t_2^2 \\ \vdots & \vdots & \vdots \\ 1 & t_M & -1/2t_M^2 \end{pmatrix}$$

$$m_{LS} = (G^T C_d^{-1} G)^{-1} G^T C_d^{-1} \mathbf{d}$$

$$m_{LS} = [16.4m, 97.0m/s, 9.4m/s^2]^T$$

$$m_{true} = [10m, 100m/s, 9.8m/s^2]^T$$

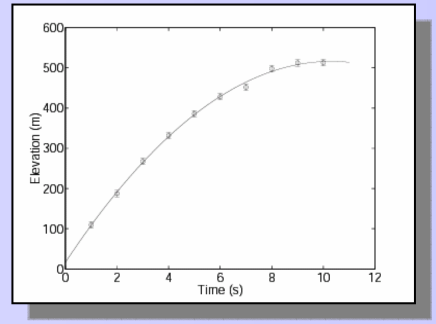


Is the data fit good enough ?

And how to errors in data propagate into the solution ?

The two questions in parameter estimation

We have our fitted model parameters



…but we are far from finished !

We need to:

- Assess the quality of the data fit.

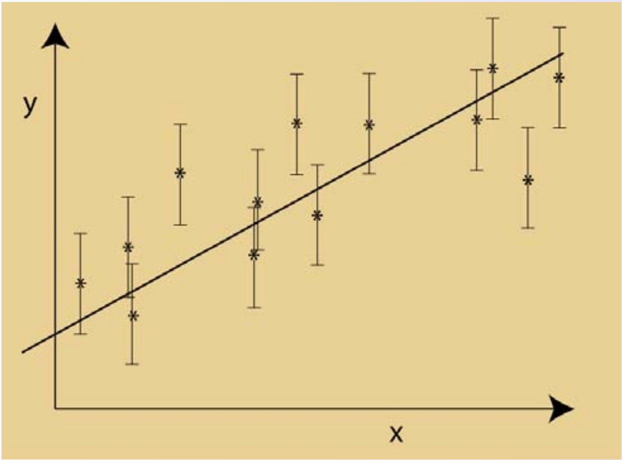
  Goodness of fit: Does the model fit the data to within the statistical uncertainty of the noise ?

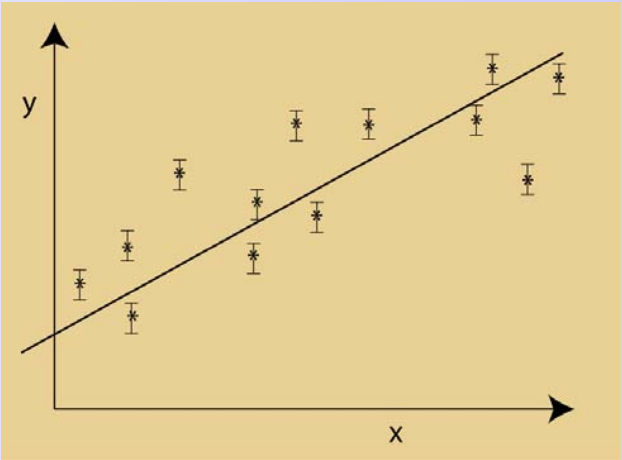
- Estimate how errors in the data propagate into the model

  What are the errors on the model parameters ?

Goodness of fit

Once we have our least squares solution $\mathbf{m}_{LS}$ how do we know whether the fit is good enough given the errors in the data ?





Use the prediction error at the least squares solution !

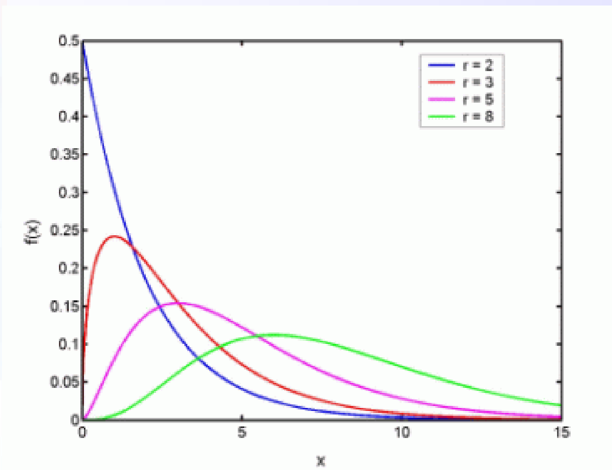
$$\phi(m_{LS}) = \frac{1}{2}(d - Gm_{LS})^T C_d^{-1}(d - Gm_{LS}) = \sum_{i=1}^{N} \left(\frac{d_i - \sum_{j=1}^{M} G_{i,j} m_j}{\sigma_i} \right)^2$$

If data errors are Gaussian this as a chi-square statistic χ^2_{obs}

Goodness of fit

For Gaussian data errors the data prediction error is the square of a Gaussian random variable hence it has a chi-square probability density function with *N-M* degrees of freedom.

$$\chi^2_{obs} = \sum_{i=1}^{N} \left(\frac{d_i - \sum_{j=1}^{M} G_{i,j} m_j}{\sigma_i} \right)^2$$



p = 0.95 p = 0.05

ndf	$\chi^2(5\%)$	$\chi^2(50\%)$	$\chi^2(95\%)$
5	1.15	4.35	11.07
10	3.94	9.34	18.31
20	10.85	19.34	31.41
50	34.76	49.33	67.50
100	77.93	99.33	124.34

$$f_{\chi^2}(x) = \frac{1}{2^{\nu/2}\Gamma(\nu/2)} x^{\nu/2-1} e^{-x/2}$$

$$p = Pr(\chi^2 \geq \chi^2_{obs})$$

$$p = \int_{\chi^2_{obs}}^{\infty} f_{\chi^2}(x)dx$$

The χ^2 test provides a means to testing the assumptions that went into producing the least squares solution. It gives the likelihood that the fit actually achieved is reasonable.

Example: Goodness of fit

The Ballistics problem

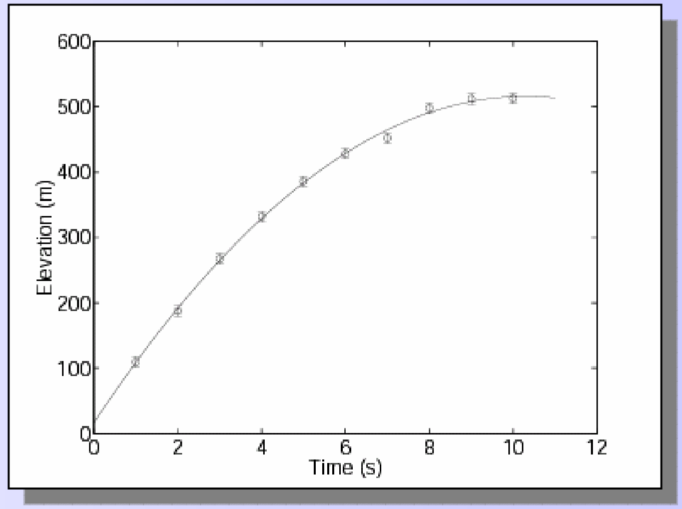
$$y_i = m_1 + m_2 t_i - \frac{1}{2} m_3 t_i^2$$

Given data and noise

t	y
1	109:3827
2	187:5385
3	267:5319
4	331:8753
5	386:0535
6	428:4271
7	452:1644
8	498:1461
9	512:3499
10	512:9753

$$C_D^{-1} = \frac{1}{\sigma^2} I$$

$$\sigma = 8m$$



$$m_{LS} = [16.4m, 97.0m/s, 9.4m/s^2]^T$$

$$\chi^2_{obs} = \sum_{i=1}^{N} \left(\frac{d_i - \sum_{j=1}^{M} G_{i,j} m_j}{\sigma_i} \right)^2 = 4.2$$

How many degrees of freedom ? ν =N-M = 10 -3=7

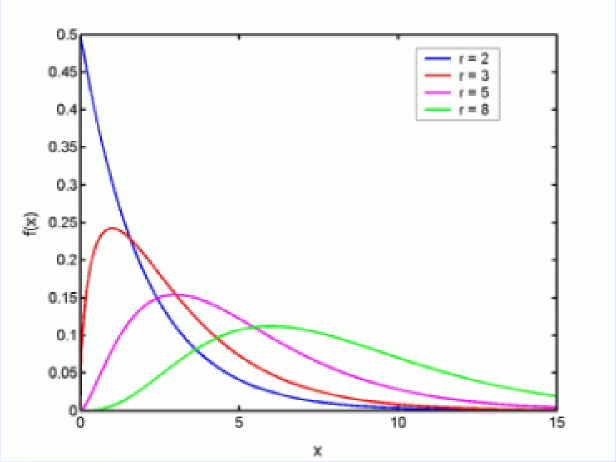
$$p = Pr(\chi^2 \geq \chi^2_{obs}) = 0.76$$

In practice values between 0.1 and 0.9 are plausible

Goodness of fit

For Gaussian data errors the chi-square statistic has a chi-square distribution with υ = *N-M* degrees of freedom.

ndf	$\chi^2(5\%)$	$\chi^2(50\%)$	$\chi^2(95\%)$
5	1.15	4.35	11.07
10	3.94	9.34	18.31
20	10.85	19.34	31.41
50	34.76	49.33	67.50
100	77.93	99.33	124.34



Exercise:

- If I fit 7 data points with a straight line and get $\chi^2 = 10^{-2}$ what would you conclude ?
- If I fit 102 data points with a straight line and get $\chi^2 = 1034.15$ what would you conclude ?
- If I fit 52 data points with a straight line and get $\chi^2 = 50$ what would you conclude ?

Goodness of fit

For Gaussian data errors the chi-square statistic has a chi-square distribution with $\upsilon = N\text{-}M$ degrees of freedom.

ndf	$\chi^2(5\%)$	$\chi^2(50\%)$	$\chi^2(95\%)$
5	1.15	4.35	11.07
10	3.94	9.34	18.31
20	10.85	19.34	31.41
50	34.76	49.33	67.50
100	77.93	99.33	124.34

What could be the cause if:

- the prediction error is much too large ? (poor data fit)
  - Truly unlikely data errors
  - Errors in forward theory
  - Under-estimated data errors
- the prediction error is too small ? (too good data fit)
  - Truly unlikely data errors
  - Over-estimated the data errors
  - Fraud !

Solution Appraisal

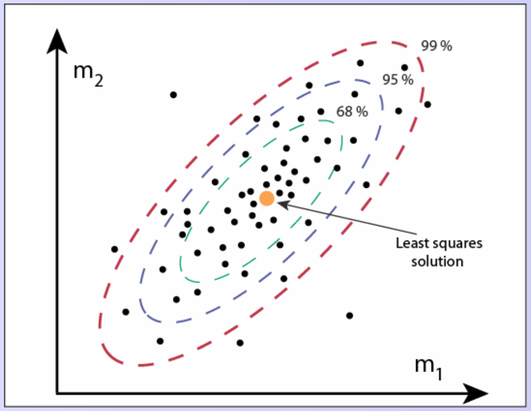
Solution error

Once we have our least squares solution $\mathbf{m}_{LS}$ and we know that the data fit is acceptable, how do we find the likely errors in the model parameters arising from errors in the data ?

$$\boldsymbol{m}_{LS} = G^{-g}\boldsymbol{d}$$

The data set we actually observed is only one realization of the many that could have been observed

$$\boldsymbol{d}' \rightarrow \boldsymbol{d} + \boldsymbol{\epsilon}$$
$$\boldsymbol{m}'_{LS} \rightarrow \boldsymbol{m}_{LS} + \boldsymbol{\epsilon}_m$$
$$\boldsymbol{m}'_{LS} = G^{-g}\boldsymbol{d}'$$
$$\boldsymbol{m}_{LS} + \boldsymbol{\epsilon}_m = G^{-g}(\boldsymbol{d} + \boldsymbol{\epsilon})$$



The effect of adding noise to the data is to add noise to the solution

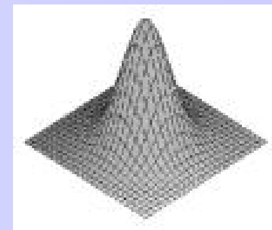
$$\boldsymbol{\epsilon}_m = G^{-g}\boldsymbol{\epsilon}$$

The model noise is a linear combination of the data noise !

Solution error: Model Covariance

Multivariate Gaussian data error distribution

$$p(\epsilon) = \frac{1}{(2\pi)^{N_d/2}|C_d|^{N_d/2}} \exp\left\{-\frac{1}{2}\epsilon^T C_d^{-1}\epsilon\right\}$$



How to turn this into a probability distribution for the model errors ?

We know that the solution error is a linear combination of the data error

$$\epsilon_m = G^{-g}\epsilon$$

The covariance of any linear combination A**d** of Gaussian distributed random variables **d** is	$\mathrm{Cov}(A\boldsymbol{d}) = A\mathrm{Cov}(\boldsymbol{d})A^T$

So we have the covariance of the model parameters

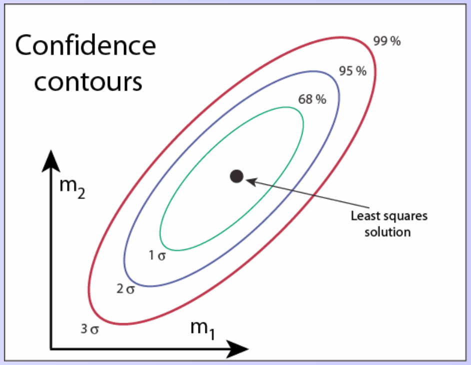
$$C_M = (G^{-g})C_d(G^{-g})^T$$

Solution error: Model Covariance

$$C_M = (G^{-g}) C_d (G^{-g})^T$$

$$G^{-g} = (G^T C_d^{-1} G)^{-1} G^T C_d^{-1}$$

$$\Rightarrow \boxed{C_M = (G^T C_d^{-1} G)^{-1}}$$

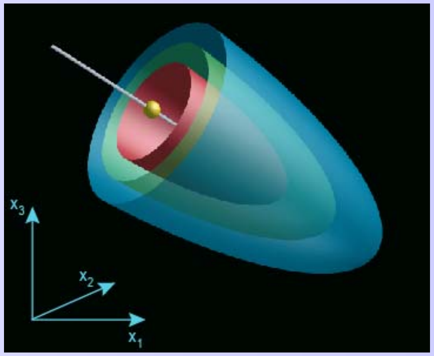


The model covariance for a least squares problem depends on data errors and not the data itself ! G is controlled by the design of the experiment.

$$p(\epsilon_m) = k' \exp\left\{ -\frac{1}{2} (m - m_{LS})^T C_M^{-1} (m - m_{LS}) \right\}$$

m_{LS} is the least squares solution

The data error distribution gives a model error distribution !

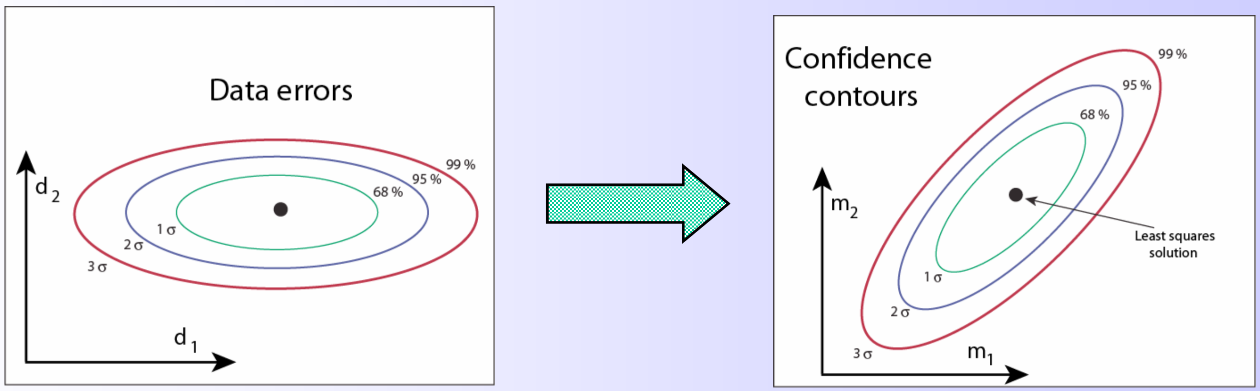


Solution error: Model Covariance

$$\boxed{C_M = (G^T C_d^{-1} G)^{-1}}$$

For the special case of independent data errors $C_d = \sigma^2 I$

$$C_M = \sigma^2 (G^T G)^{-1}$$



Independent data errors Correlated model errors

For linear regression problem

$$C_M = \frac{\sigma^2}{N\sum_{i=1}^N x_i^2 - \left(\sum_{i=1}^N x_i\right)^2} \begin{bmatrix} \sum_{i=1}^N x_i^2 & -\sum_{i=1}^N x_i \\ -\sum_{i=1}^N x_i & N \end{bmatrix}$$

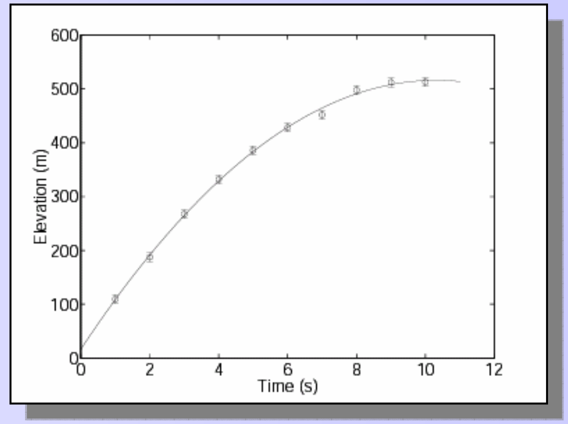
Example: Model Covariance and confidence intervals

For Ballistics problem

$$y_i = m_1 + m_2 t_i - \frac{1}{2} m_3 t_i^2$$

$$C_M = (G^T C_d^{-1} G)^{-1}$$

$$C_D^{-1} = \frac{1}{\sigma^2} I$$



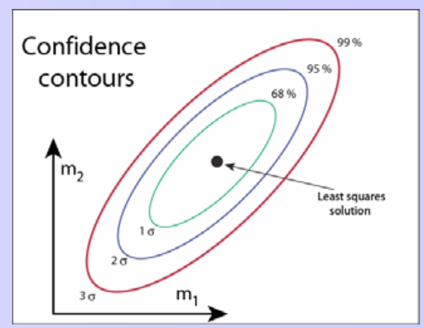
$$C_M = \begin{bmatrix} 88.53 & -33.60 & -5.33 \\ -33.60 & 15.44 & 2.67 \\ -5.33 & 2.67 & 0.48 \end{bmatrix}$$

95% confidence interval for parameter i

$$= 1.96 \times (C_M)_{i,i}^{1/2}$$

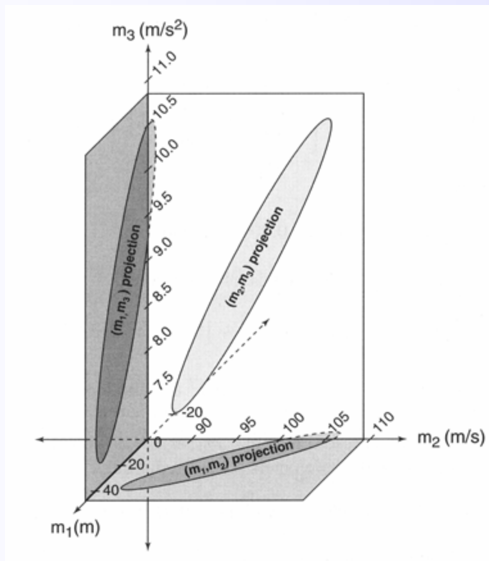
$$m_{true} = [10m, 100m/s, 9.8m/s^2]^T$$

$$m_{LS} = [16.4 \pm 18.4m, 97.0 \pm 7.7m/s, 9.4 \pm 1.4m/s^2]^T$$



Confidence intervals by projection

The M-dimensional confidence ellipsoid can be projected onto any subset (or combination) of Δ parameters to obtain the corresponding confidence ellipsoid.



Full M-dimensional ellipsoid

$$\Delta^2 = (m - m_{LS})^T C_M^{-1} (m - m_{LS})$$

Projected ν dimension ellipsoid

$$\Delta^2 = \delta m'^T [C_{M,proj}]^{-1} \delta m'$$

$\delta m' =$ Projected model vector

$C_{M,proj} =$ Projected covariance matrix

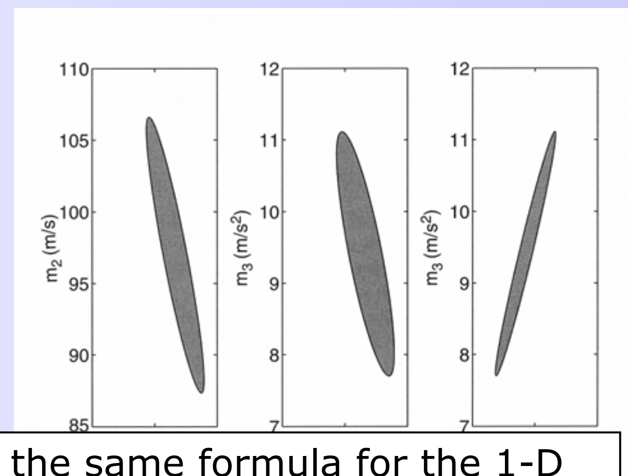
$\Delta^2 =$ Chosen percentage point of the χ^2_ν distribution

To find the 90% confidence ellipse for (x,y) from a 3-D (x,y,z) ellipsoid

$$\Delta^2 = 4.61 \quad [= \chi^2_2(90\%)]$$

$$C_{M,proj} = \begin{bmatrix} \sigma^2_{1,1} & \sigma^2_{1,2} \\ \sigma^2_{1,2} & \sigma^2_{2,2} \end{bmatrix}$$

$$\delta m' = [m_1 - m_{LS,1}, m_2 - m_{LS,2}]^T$$



Can you see that this procedure gives the same formula for the 1-D case obtained previously ?

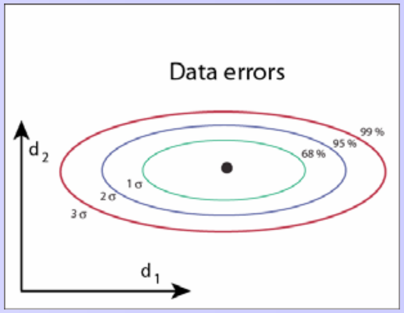
What if we do not know the errors on the data ?

Both Chi-square goodness of fit tests and model covariance Calculations require knowledge of the variance of the data.

What can we do if we do not know σ ?

Consider the case of

$$C_D = \sigma^2 I$$



Independent data errors

$$\chi^2_{N-M} = \frac{1}{\sigma^2}\sum_{i=1}^{N}\left(d_i - \sum_{j=1}^{M} G_{i,j}m_j\right)^2 \Rightarrow \sigma^2 = \frac{1}{(N-M)}\sum_{i=1}^{N}\left(d_i - \sum_{j=1}^{M} G_{i,j}m_j\right)^2$$

Calculated from least squares solution

$$C_M = \sigma^2 (G^T G)^{-1}$$

So we can still estimate model errors using the calculated data errors but we can no long claim anything about goodness of fit.

Model Resolution matrix

If we obtain a solution to an inverse problem we can ask what its relationship is to the true solution

$$\boldsymbol{m}_{est} = G^{-g}\boldsymbol{d}$$

But we know

$$\boldsymbol{d} = G\boldsymbol{m}_{true}$$

and hence

$$\boldsymbol{m}_{est} = G^{-g}G\boldsymbol{m}_{true} = R\boldsymbol{m}_{true}$$

The matrix R measures how `good an inverse' G^{-g} is.

The matrix R shows how the elements of m_{est} are built from linear combination of the true model, m_{true}. Hence matrix R measures the amount of blurring produced by the inverse operator.

For the least squares solution we have

$$G^{-g} = (G^T C_D^{-1} G)^{-1} G^T C_D^{-1} \qquad \Rightarrow R = I$$

Data Resolution matrix

If we obtain a solution to an inverse problem we can ask what how it compares to the data

$$\boldsymbol{d}_{pre} = G\boldsymbol{m}_{est}$$

But we know

$$\boldsymbol{m}_{est} = G^{-g}\boldsymbol{d}_{obs}$$

and hence

$$\boldsymbol{d}_{pre} = GG^{-g}\boldsymbol{d}_{obs} = D\boldsymbol{d}_{obs}$$

The matrix D is analogous to the model resolution matrix R but measures how independently the model produced by G^{-g} can reproduce the data. If D = I then the data is fit exactly and the prediction error **d**-G**m** is zero.

Recap: Goodness of fit and model covariance

- Once a best fit solution has been obtained we test goodness of fit with a chi-square test (assuming Gaussian statistics)

$$\left[\chi^2_{obs} = \sum_{i=1}^{N} \frac{(d_i - \sum_{j=1}^{M} G_{i,j} m_j)^2}{\sigma_i^2}\right]$$

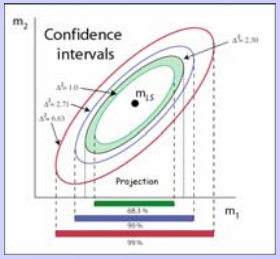
- If the model passes the goodness of fit test we may proceed to evaluating model covariance (if not then your data errors are probably too small)

- Evaluate model covariance matrix

$$C_M = (G^T C_d^{-1} G)^{-1}$$

- Plot model or projections of it onto chosen subsets of parameters

$$\Delta'^2 = (m - m_{LS})'^T C'^{-1}_M (m - m_{LS})'$$



- Calculate confidence intervals using projected equation

$$\sigma_{M,i} = \Delta \times \sigma_{i,i}$$

Where Δ^2 follows a χ^2_1 distribution

Recap: Linear discrete inverse problems

- The Least squares solution minimizes the prediction error.

$$\boxed{\phi(\mathbf{m}_{LS}) = (\mathbf{d} - G\mathbf{m}_{LS})^T C_d^{-1} (\mathbf{d} - G\mathbf{m}_{LS})}$$

$$\boxed{\mathbf{m}_{LS} = (G^T C_d^{-1} G)^{-1} G^T C_d^{-1} \mathbf{d} = G^{-g}\mathbf{d}}$$

- Goodness of fit criteria tells us whether the least squares model adequately fits the data, given the level of noise.

$$\phi(\mathbf{m}_{LS}) \rightarrow \chi^2_{N-M}$$

Chi-square with *N-M* degrees of freedom

- The covariance matrix describes how noise propagates from the data to the estimated model

$$\boxed{C_M = (G^T C_d^{-1} G)^{-1}}$$

$$\Delta^2 \rightarrow \chi^2_M$$

Chi-square with *M* degrees of freedom

$$\boxed{(\mathbf{m} - \mathbf{m}_{LS})^T C_M^{-1} (\mathbf{m} - \mathbf{m}_{LS}) < \Delta^2}$$

Gives confidence intervals

- The resolution matrix describes how the estimated model relates to the true model

Beamforming

Beamforming frequency

$$p(f) = \int p(t) e^{-i2\pi ft} dt \qquad \text{FFT}$$

$$p(t) = \int p(f) e^{i2\pi ft} df \qquad \text{IFFT}$$

Pressure field is a sum of plane waves

$$p(f,\mathbf{r}) = \int p(f,\mathbf{k}) e^{i(\mathbf{k}^T\mathbf{r})} d\mathbf{k}$$

$$p(f,\mathbf{k}) = \int p(f,\mathbf{r}) e^{-i(\mathbf{k}^T\mathbf{r})} d\mathbf{r}$$

Based of the observed field $p(f,\mathbf{r}_k)$ at discrete ranges $\mathbf{r}_k$ the $p(f,\mathbf{k}_j)$ is estimated

$$p(f,\mathbf{k}_j) = \sum_k p(f,\mathbf{r}_k) e^{-i(\mathbf{k}_j^T\mathbf{r}_k)} = \mathbf{w}^H \mathbf{p}$$

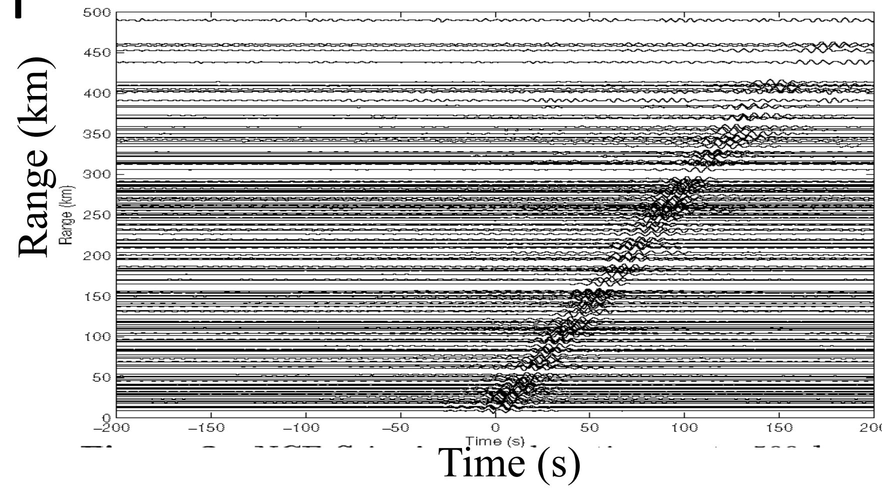
Where

$$\mathbf{w} = \begin{bmatrix} e^{i(\mathbf{k}_j^T\mathbf{r}_1)} \\ \vdots \\ e^{i(\mathbf{k}_j^T\mathbf{r}_N)} \end{bmatrix} \qquad \mathbf{p} = \begin{bmatrix} p(f,\mathbf{r}_1) \\ \vdots \\ p(f,\mathbf{r}_N) \end{bmatrix}$$

Beamforming

$$p(f) = \int p(t) e^{-i2\pi f t} dt \qquad \text{FFT}$$

$$p(t) = \int p(f) e^{i2\pi f t} df \qquad \text{IFFT}$$



Pressure field is a sum of plane waves

$$B(t,m) = \sum_k p_k(t - \tau_{km})$$

$$= \sum_k \int [\int p_k(t - \tau_{km}) e^{-i2\pi f t} dt] e^{i2\pi f t} df$$

$$= \sum_k \int e^{-i2\pi f \tau_{km}} [\int p_k(t) e^{-i2\pi f t} dt] e^{i2\pi f t} df$$

$$= \sum_k \int e^{-i2\pi f \tau_{km}} p_k(f) e^{i2\pi f t} df$$

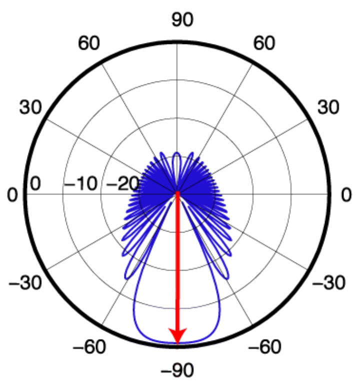
$$B(f,m) = \sum_k e^{-i2\pi f \tau_{km}} p_k(f) = \mathbf{w}^H \mathbf{p}$$

Where

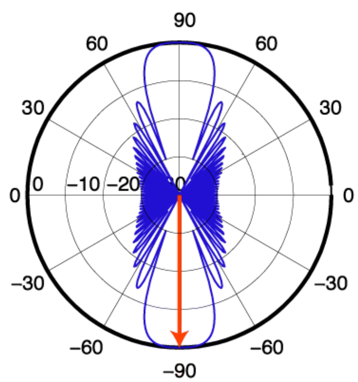
$$\mathbf{w} = \begin{bmatrix} e^{i2\pi f \tau_{km}} \\ \vdots \\ e^{i2\pi f \tau_{km}} \end{bmatrix} \qquad \mathbf{p} = \begin{bmatrix} p(f, \mathbf{r}_1) \\ \vdots \\ p(f, \mathbf{r}_N) \end{bmatrix}$$

Aliazing

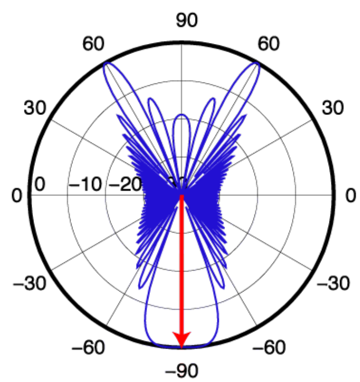
$d < \lambda / 2$



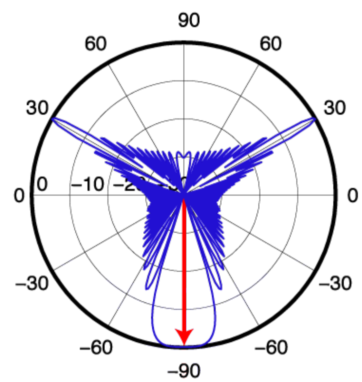
$d = \lambda / 2$



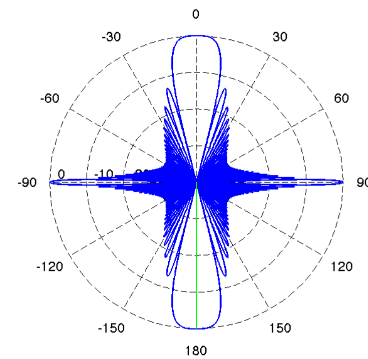
$d > \lambda / 2$



$d > \lambda / 2$



$d = \lambda$



SVD

Proof: Minimum Length solution

Constrained minimization

$$Min \quad L(\boldsymbol{m}) = \boldsymbol{m}^T\boldsymbol{m} \; : \; \boldsymbol{d} = G\boldsymbol{m}$$

Lagrange multipliers leads to unconstrained minimization of

$$\phi(\boldsymbol{m},\boldsymbol{\lambda}) = \boldsymbol{m}^T\boldsymbol{m} + \boldsymbol{\lambda}^T(\boldsymbol{d} - G\boldsymbol{m}) \qquad = \sum_{j=1}^{M} m_j^2 + \sum_{i=1}^{N} \lambda_i (d_i - G_{i,j} m_j)$$

$$\frac{\partial \phi}{\partial \boldsymbol{\lambda}} = \boldsymbol{d} - G\boldsymbol{m} = \boldsymbol{0} \longrightarrow \frac{\partial \phi}{\partial \lambda_i} = \sum_{i=1}^{N} (d_i - G_{i,j} m_j)$$

$$\frac{\partial \phi}{\partial \boldsymbol{m}} = 2\boldsymbol{m} - G^T\boldsymbol{\lambda} = \boldsymbol{0} \longrightarrow \frac{\partial \phi}{\partial m_j} = 2m_j - \sum_{i=1}^{N} \lambda_i G_{i,j}$$

$$\Rightarrow \boldsymbol{m} = \frac{1}{2}\, G^T\boldsymbol{\lambda}$$

$$\Rightarrow \boldsymbol{d} = G\boldsymbol{m} = \frac{1}{2}\, GG^T\boldsymbol{\lambda}$$

$$\Rightarrow \boldsymbol{\lambda} = 2\,(GG^T)^{-1}\boldsymbol{d}$$

$$\Rightarrow \boldsymbol{m} = G^T(GG^T)^{-1}\boldsymbol{d}$$

Minimum Length and least squares solutions

$$\boldsymbol{m}_{ML} = G^T(GG^T)^{-1}\boldsymbol{d} \qquad \boldsymbol{m}_{LS} = (G^TG)^{-1}G^T\boldsymbol{d}$$

$$\boldsymbol{m}_{est} = G^{-g}\boldsymbol{d}$$

Data resolution matrix

$$\boldsymbol{d}_{pre} = D\boldsymbol{d}_{obs}$$

$$D = GG^{-g}$$

Minimum length

$$D = GG^T(GG^T)^{-1} = I$$

$$R = G^T(GG^T)^{-1}G \neq I$$

Least squares

$$D = G(G^TG)^{-1}G^T \neq I$$

$$R = (G^TG)^{-1}G^TG = I$$

There is symmetry between the least squares and minimum length solutions. Least squares complete solves the over-determined problem and has perfect model resolution, while the minimum length solves the completely under-determined problem and has perfect data resolution. For mix-determined problems all solutions will be between these two extremes.

Singular value decomposition

SVD is a method of analyzing and solving linear discrete ill-posed problems.

At its heart is the *Lanczos decomposition* of the matrix G

$$\boxed{\boldsymbol{d} = G\boldsymbol{m}}$$

$$G = USV^T$$

$$\underset{N \times M}{G} = \underset{N \times N}{[\boldsymbol{u}_1, \boldsymbol{u}_2, \ldots, \boldsymbol{u}_N]} S \underset{M \times M}{[\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_M]^T}$$

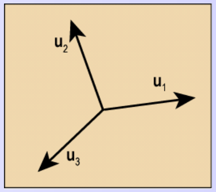
U is an N x N ortho-normal matrix with columns that span the data space

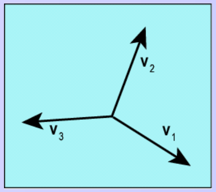
V is an M x M ortho-normal matrix with columns that span the model space

S is an N x M diagonal matrix with non-negative elements $\rightarrow$ singular values

$$UU^T = U^T U = I_N$$

$$VV^T = V^T V = I_M$$





Ill-posed problems arise when some of the singular values are zero

Lanczos (1977)

For a discussion see Ch. 4 of Aster et al. (2004)

Singular value decomposition

Given G, how do we calculate the matrices U, V and S ?

$$G = USV^T$$

$$U = [\boldsymbol{u}_1|\boldsymbol{u}_2|\dots|\boldsymbol{u}_N]$$
$$V = [\boldsymbol{v}_1|\boldsymbol{v}_2|\dots|\boldsymbol{v}_M]$$

It can be shown that the columns of *U* are the eigenvectors of the matrix GG^T

$$GG^T\boldsymbol{u}_i = s_i^2\boldsymbol{u}_i$$

Try and prove this !

It can be shown that the columns of *V* are the eigenvectors of the matrix G^TG

$$G^TG\boldsymbol{v}_i = s_i^2\boldsymbol{v}_i$$

Try and prove this !

The eigenvalues, s_i^2, are the square of the elements in diagonal of the N x M matrix S.

If N > M

$$S = \begin{bmatrix} s_1 & 0 & \cdots & 0 \\ 0 & s_2 & 0 & 0 \\ \vdots & & \ddots & 0 \\ 0 & 0 & 0 & s_M \\ \hline 0 & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

If M > N

$$S = \left[\begin{array}{cccc|ccc} s_1 & 0 & \cdots & 0 & 0 & \cdots & 0 \\ 0 & s_2 & 0 & 0 & 0 & \cdots & 0 \\ \vdots & & \ddots & 0 & 0 & \cdots & 0 \\ 0 & 0 & 0 & s_N & 0 & \cdots & 0 \end{array}\right]$$

Singular value decomposition

$$S = \begin{bmatrix} s_1 & 0 & \cdots & 0 \\ 0 & s_2 & 0 & 0 \\ \vdots & & \ddots & 0 \\ 0 & 0 & 0 & s_M \\ \hdashline 0 & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

$$S = \left[\begin{array}{cccc:ccc} s_1 & 0 & \cdots & 0 & 0 & \cdots & 0 \\ 0 & s_2 & 0 & 0 & 0 & \cdots & 0 \\ \vdots & & \ddots & 0 & 0 & \cdots & 0 \\ 0 & 0 & 0 & s_N & 0 & \cdots & 0 \end{array}\right]$$

Suppose the first p are non-zero, then N x M non square matrix S S can be written in a partitioned form

$$S = \begin{array}{cc} \text{M} & \\ \begin{bmatrix} S_p & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} & \text{N} \end{array}$$

By convention we order the singular values

$$s_1 \geq s_2 \geq \cdots \geq s_p$$

$$S_p = \begin{array}{cc} \text{p} & \\ \begin{bmatrix} s_1 & 0 & \cdots & 0 \\ 0 & s_2 & 0 & 0 \\ \vdots & & \ddots & 0 \\ 0 & 0 & 0 & s_p \end{bmatrix} & \text{p} \end{array}$$

$$U = [\boldsymbol{u}_1|\boldsymbol{u}_2|\ldots|\boldsymbol{u}_N]$$

$$V = [\boldsymbol{v}_1|\boldsymbol{v}_2|\ldots|\boldsymbol{v}_M]$$

where the submatrix s_p is a p x p diagonal matrix contains the non-zero singular values

$$p_{max} = \min(N, M)$$

Singular value decomposition

If only the first p singular values are nonzero we write

$$G = [U_p \mid U_o] \begin{bmatrix} S_p & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} [V_p \mid V_o]^T$$

U_p represents the first p columns of U

U_o represents the last N-p columns of U $\rightarrow$ A data null space is created

V_p represents the first p columns of V

V_o represents the last M-p columns of V $\rightarrow$ A model null space is created

Properties

$$U_p^T U_o = 0 \quad U_o^T U_p = 0 \quad V_p^T V_o = 0 \quad V_o^T V_p = 0$$

$$U_p^T U_p = I \quad U_o^T U_o = I \quad V_o^T V_o = I \quad V_p^T V_p = I$$

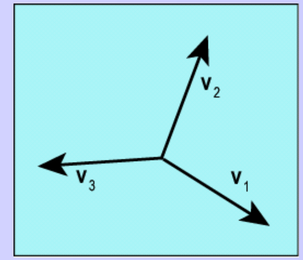
Since the columns of V_o and U_o multiply by zeros we get the *compact* form for G

$$G = U_p S_p V_p^T$$

Model null space

Consider a vector made up of a linear combination of the columns of V_o

$$\boldsymbol{m}_v = \sum_{i=p+1}^{M} \lambda_i \boldsymbol{v}_i$$



The model m lies in the space spanned by columns of V_o

$$G\boldsymbol{m}_v = \sum_{i=p+1}^{M} \lambda_i U_p S_p V_p^T \boldsymbol{v}_i = \boldsymbol{0}$$

So any model of this type has no affect on the data. It lies in the model null space !

Where have we seen this before ?

Consequence: If any solution exists to the inverse problem then an infinite number will

Assume the model m_{ls} fits the data $G\boldsymbol{m}_{ls} = \boldsymbol{d}_{obs}$

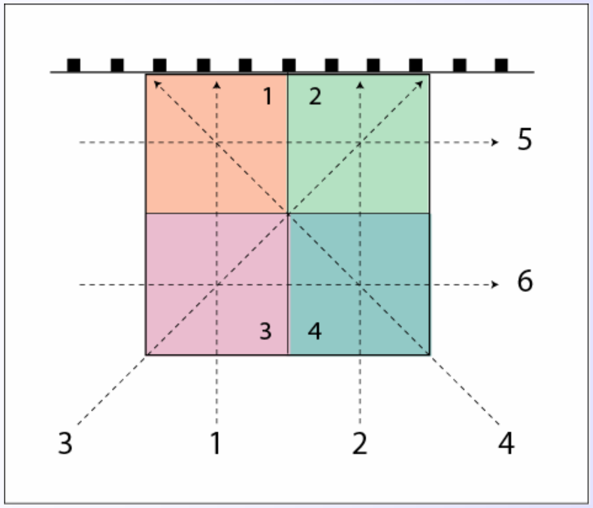
Uniqueness question of Backus and Gilbert

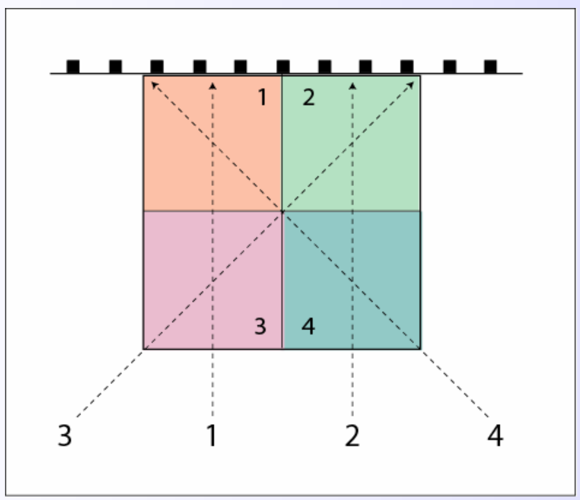
$$\begin{aligned} G(\boldsymbol{m}_{ls} + \boldsymbol{m}_v) &= G\boldsymbol{m}_{ls} + G\boldsymbol{m}_v \\ &= \boldsymbol{d}_{obs} + \boldsymbol{0} \end{aligned}$$

The data can not constrain models in the model null space

Example: tomography

Idealized tomographic experiment





$$\delta \boldsymbol{d} = G \delta \boldsymbol{m}$$

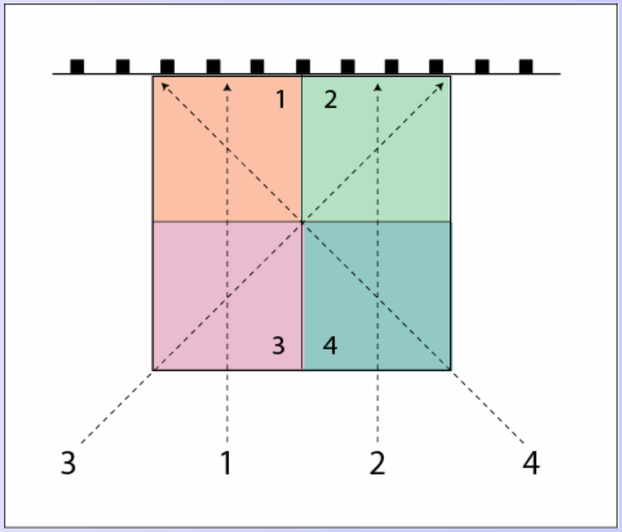
$$G = \begin{bmatrix} G_{1,1} & G_{1,2} & G_{1,3} & G_{1,4} \\ \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots \end{bmatrix}$$

What are the entries of G ?

Example: tomography

Using rays 1-4

$$\delta \boldsymbol{d} = G\delta \boldsymbol{m}$$



$$G = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & \sqrt{2} & \sqrt{2} & 0 \\ \sqrt{2} & 0 & 0 & \sqrt{2} \end{bmatrix}$$

$$G^T G = \begin{bmatrix} 3 & 0 & 1 & 2 \\ 0 & 3 & 2 & 1 \\ 1 & 2 & 3 & 0 \\ 2 & 1 & 0 & 3 \end{bmatrix}$$

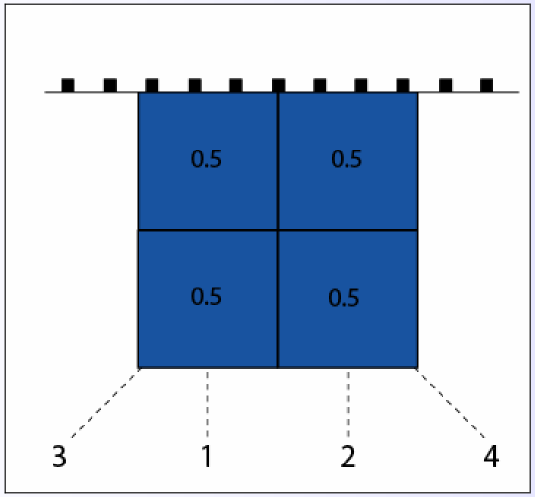
This has eigenvalues 6,4,2,0.

$$V_p = \begin{bmatrix} 0.5 & -0.5 & -0.5 \\ 0.5 & 0.5 & 0.5 \\ 0.5 & 0.5 & -0.5 \\ 0.5 & -0.5 & 0.5 \end{bmatrix} \qquad V_o = \begin{bmatrix} 0.5 \\ 0.5 \\ -0.5 \\ -0.5 \end{bmatrix} \qquad G\boldsymbol{v_o} = \boldsymbol{0}$$

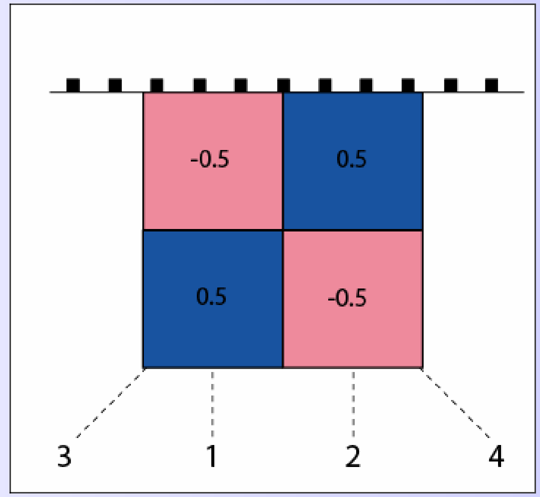
What type of change does the null space vector correspond to ?

Worked example: Eigenvectors

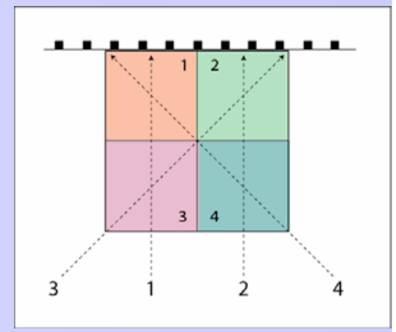
$S_1^2=6$



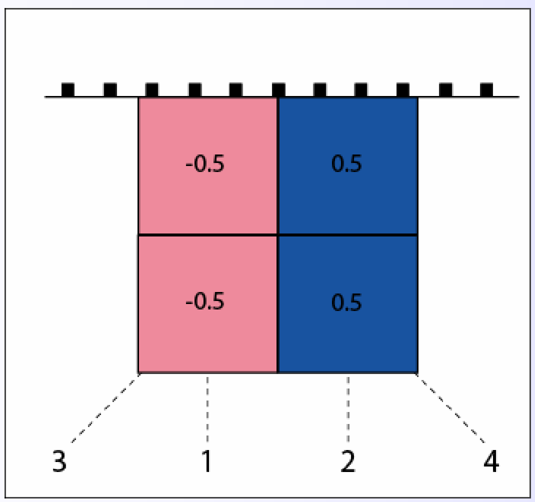
$S_2^2=4$



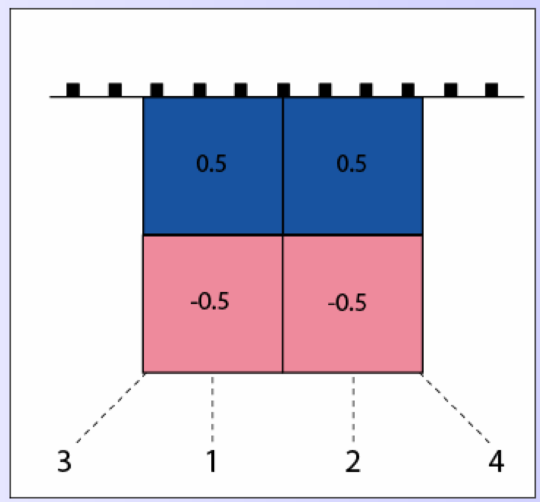
$$V_p = \begin{bmatrix} 0.5 & -0.5 & -0.5 \\ 0.5 & 0.5 & 0.5 \\ 0.5 & 0.5 & -0.5 \\ 0.5 & -0.5 & 0.5 \end{bmatrix}$$



$S_3^2=2$



$S_4^2=0$



$$V_o = \begin{bmatrix} 0.5 \\ 0.5 \\ -0.5 \\ -0.5 \end{bmatrix}$$

Data and model null spaces