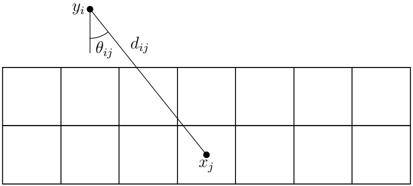
Gravimetric prospecting

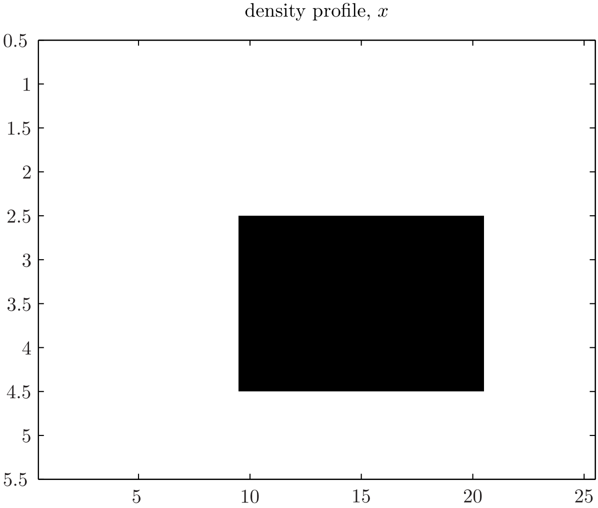


- x_j is excess density of voxel j
- y_i is gravity anomaly at location i
- θ_{ij} is angle from location i to voxel j
- d_{ij} is distance from location i to voxel j
- Newton's law of gravitation:

$$y_i = \sum_{j=1}^{n} \frac{G \cos(\theta_{ij})}{d_{ij}^2} x_j, \qquad i = 1, \ldots, m$$

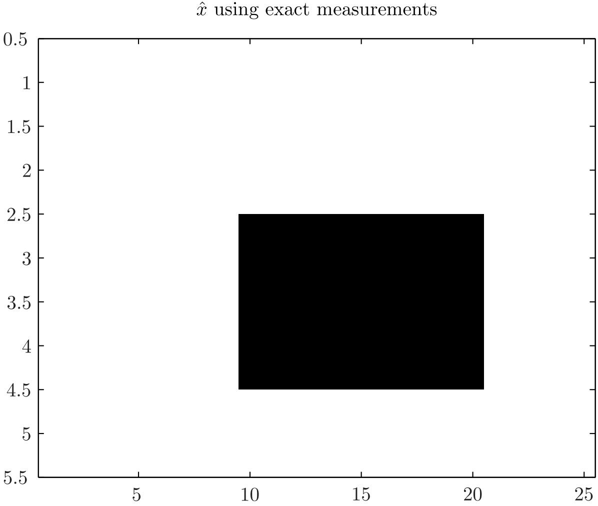
Gravimetric prospecting

an example



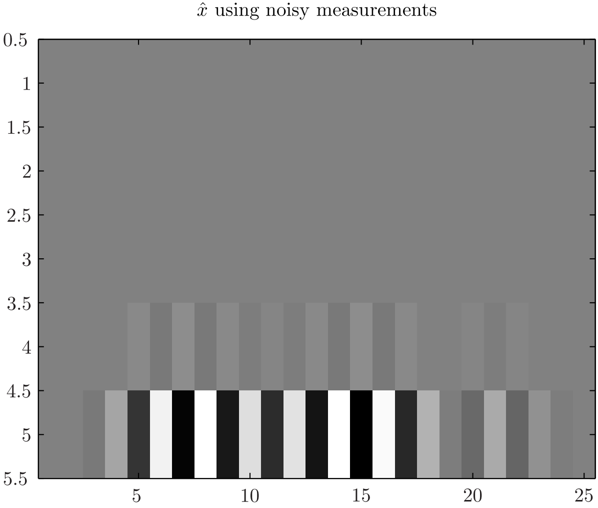
Gravimetric prospecting

estimated density with exact measurements



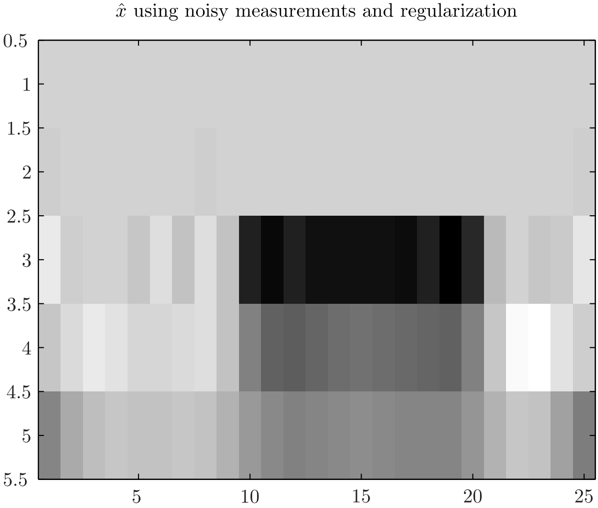
Gravimetric prospecting

estimated density with noisy measurements ($\pm 0.01\%$)

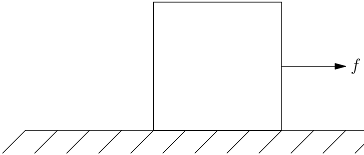


Gravimetric prospecting

estimated density with noisy measurements and regularization



Forces applied to a unit mass



- unit mass initially at rest at the origin
- force $f(t)$ applied for $0 \leq t \leq n$
- $f(t) = x_j$ for $j - 1 \leq t < j$, $j = 1, \ldots, n$
- final position:

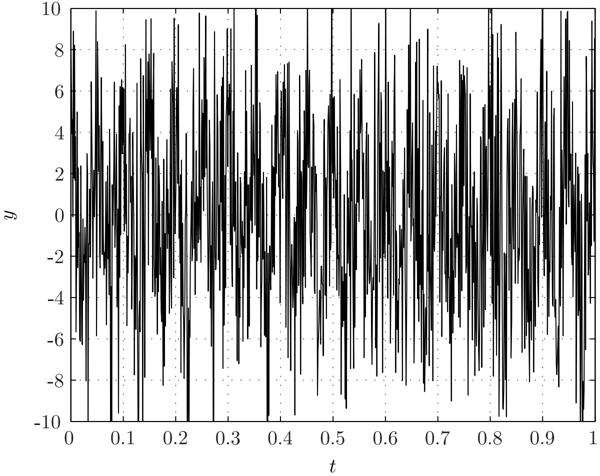
$$y_1 = (n - \tfrac{1}{2})x_1 + (n - \tfrac{3}{2})x_2 + \cdots + \tfrac{1}{2}x_n$$

- final velocity:

$$y_2 = x_1 + x_2 + \cdots + x_n$$

A noisy signal

a sum of two sinusoids corrupted by noise



Approximate Fourier transform

$$
\begin{aligned}
X(f) &= \int_{-\infty}^{+\infty} x(t)e^{-2\pi i f t}\,dt \\
&\approx \int_{0}^{1} x(t)e^{-2\pi i f t}\,dt \\
&\approx \frac{1}{N}\sum_{n=0}^{N-1} x\left(\frac{n}{N}\right)e^{-2\pi i f n/N}
\end{aligned}
$$

The discrete Fourier transform

define the discrete Fourier transform of a signal $x_0, \ldots, x_{N-1}$:

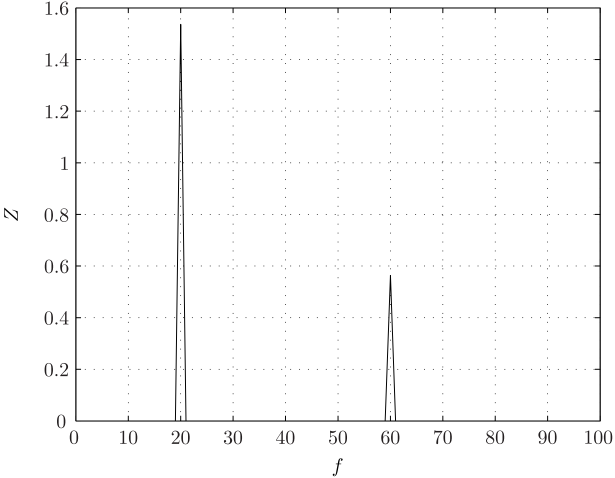
$$X_k = \sum_{n=0}^{N-1} x_n e^{-2\pi i k n/N}, \qquad k = 0, \ldots, N-1$$

we can also express these equations as

$$\begin{aligned} X_0 &= e^{-2\pi i (0)(0)/N} x_0 + \cdots + e^{-2\pi i (0)(N-1)/N} x_{N-1} \\ &\vdots \\ X_k &= e^{-2\pi i (N-1)(0)/N} x_0 + \cdots + e^{-2\pi i (N-1)(N-1)/N} x_{N-1} \end{aligned}$$

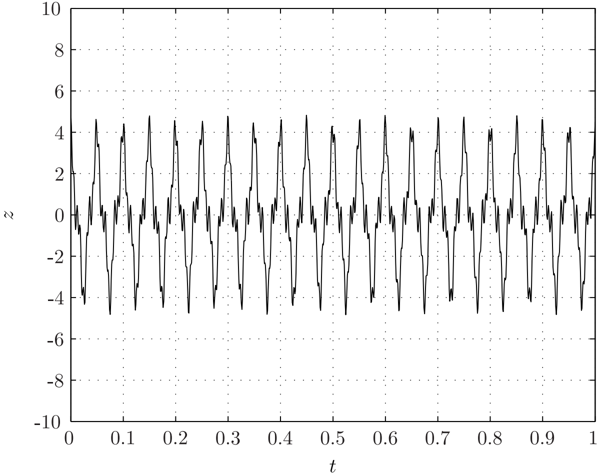
Filtering in the frequency domain

apply threshold filter



The denoised signal

applying the inverse DFT gives



Systems of linear equations

system of linear equations:

$$\begin{array}{ccccccccc} y_1 & = & A_{11} & x_1 & + & \cdots & + & A_{1n} & x_n \\ & \vdots & & & & & & & \\ y_m & = & A_{m1} & x_1 & + & \cdots & + & A_{mn} & x_n \end{array}$$

matrix representation:

$$y = Ax$$

where

$$y = \begin{bmatrix} y_1 \\ \vdots \\ y_m \end{bmatrix}, \qquad A = \begin{bmatrix} A_{11} & \cdots & A_{1n} \\ \vdots & \ddots & \vdots \\ A_{m1} & \cdots & A_{mn} \end{bmatrix}, \qquad x = \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix}$$

right side of system *defines* matrix-vector multiplication

The "row" view

$$y_i = \sum_{j=1}^{n} A_{ij} x_j = A_{i*} x$$

- y_i is the ith measurement (known)
- x_j is the jth parameter (unknown)
- A_{ij} is the sensitivity of ith sensor to jth parameter
- ith row of A associated with ith measurement

sample problems:

- given y, find *an* x such that $y = Ax$
- given y, find *all* x such that $y = Ax$
- given y, find an x such that $y \approx Ax$

The “column” view

$$y = \sum_{j=1}^{n} x_j A_{*j}$$

- x is the vector of inputs or design parameters (to be chosen)
- y is the vector of outputs or results (given)
- A_{*j} is effect of jth input on vector of outputs
- jth column of A associated with jth input

sample problems:

- given y_{des}, find *an* x such that $Ax = y_{\text{des}}$
- given y_{des}, find *all* x such that $Ax = y_{\text{des}}$
- given y_{des}, find the “smallest” x such that $Ax = y_{\text{des}}$

Standard basis vectors

- jth standard basis vector in $\mathbb{R}^n$ is vector $e_j \in \mathbb{R}^n$ such that

$$(e_j)_i = \delta_{ij} = \begin{cases} 1 & i = j, \\ 0 & \text{otherwise} \end{cases}$$

- δ_{ij} (Kronecker delta) often useful in formulating problems
- standard basis vectors extract slices of a matrix:

$$Ae_j = A_{*j}, \qquad e_i^\mathsf{T} A = A_{i*}, \qquad e_i^\mathsf{T} A e_j = A_{ij}$$

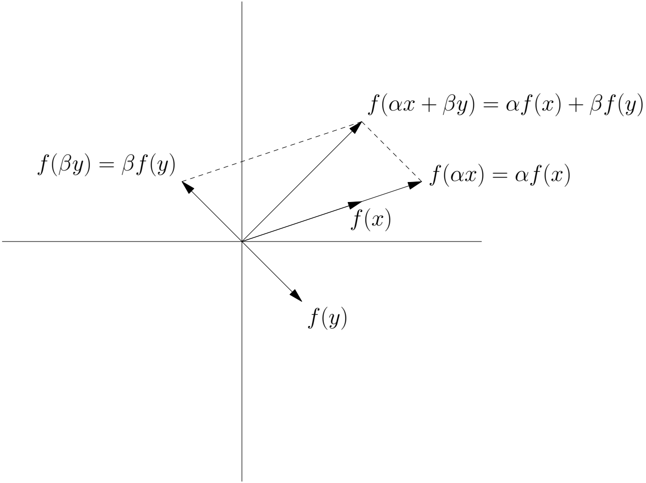
- `MATLAB`: `sparse(j,1,1,n,1)`

Linear functions

a function $f : \mathbb{R}^n \to \mathbb{R}^m$ is *linear* if it is

- *additive:* $f(x+y) = f(x) + f(y)$ for all $x, y \in \mathbb{R}^n$
- *homogeneous:* $f(\alpha x) = \alpha f(x)$ for all $\alpha \in \mathbb{R}$ and $x \in \mathbb{R}^n$

Superposition principle



Matrix multiplication function

- if $f(x) = Ax$ for some matrix $A \in \mathbb{R}^{m \times n}$,
  - then $f : \mathbb{R}^n \to \mathbb{R}^m$ is linear
- conversely: if $f : \mathbb{R}^n \to \mathbb{R}^m$ is linear,
  - then there is a matrix $A \in \mathbb{R}^{m \times n}$ such that $f(x) = Ax$
  - the matrix A is unique
- matrix is concrete representation of abstract linear function

Linearization

if $f : \mathbb{R}^n \to \mathbb{R}^m$ is differentiable at $x_0 \in \mathbb{R}^n$, then

$$f(x) \text{ is very near } f(x_0) + Df(x_0)(x - x_0)$$

whenever

$$x \text{ is near } x_0$$

where

$$\left[Df(x_0)\right]_{ij} = \left.\frac{\partial f_i}{\partial x_j}\right|_{x=x_0}$$

is the derivative (Jacobian) matrix

Linearization

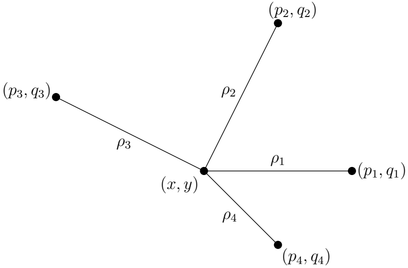
define the deviations

$$\delta x = x - x_0,$$
$$\delta y = f(x) - f(x_0)$$

small deviations are (approximately) related by a linear function:

$$\delta y \approx Df(x_0)\delta x$$

Linearized range measurements



- (x, y) is an unknown location in the plane
- (p_i, q_i) are known locations of beacons for $i = 1, \ldots, n$
- measure distance ρ_i between (x, y) and beacon i

Linearized range measurements

- $\rho \in \mathbb{R}^4$ is a nonlinear function of $(x, y) \in \mathbb{R}^2$:

$$\rho_i(x, y) = \sqrt{(x - p_i)^2 + (y - q_i)^2}$$

- linearize around (x_0, y_0):

$$\delta\rho \approx A \begin{bmatrix} \delta x \\ \delta y \end{bmatrix},$$

where

$$A_{i1} = \frac{x_0 - p_i}{\sqrt{(x_0 - p_i)^2 + (y_0 - q_i)^2}},$$
$$A_{i2} = \frac{y_0 - q_i}{\sqrt{(x_0 - p_i)^2 + (y_0 - q_i)^2}}$$

Definition of matrix multiplication

- suppose $A \in \mathbb{R}^{m \times p}$ and $B \in \mathbb{R}^{p \times n}$
- matrices represent linear functions
  - $f : \mathbb{R}^p \to \mathbb{R}^m$ such that $f(z) = Az$
  - $g : \mathbb{R}^n \to \mathbb{R}^p$ such that $g(x) = Bx$
- define matrix product AB as matrix representation of $f \circ g$

Definition of matrix multiplication

- let $z = g(x)$ and $y = f(z)$
- definition of matrix-vector multiplication:

$$y_i = \sum_{k=1}^{p} A_{ik} z_k, \qquad z_k = \sum_{j=1}^{n} B_{kj} x_j$$

- combining these expressions gives

$$y_i = \sum_{k=1}^{p} A_{ik} \left(\sum_{j=1}^{n} B_{kj} x_j \right) = \sum_{j=1}^{n} \left(\sum_{k=1}^{p} A_{ik} B_{kj} \right) x_j$$

- therefore,

$$(AB)_{ij} = \sum_{k=1}^{p} A_{ik} B_{kj}$$

Entries of matrix product

$$(AB)_{ij} = \sum_{k=1}^{p} A_{ik} B_{kj} = A_{i*} B_{*j}$$

- entries of AB are inner products of rows of A, columns of B
- (i, j) entry is inner product of ith row of A, jth column of B

Rows of matrix product

$$(AB)_{i*} = \sum_{k=1}^{p} A_{ik} B_{k*} = A_{i*} B$$

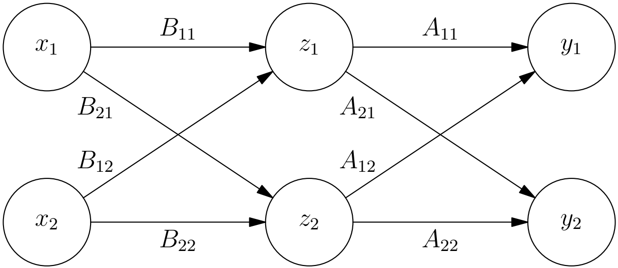
- rows of AB are linear combinations of rows of B
- ith row of A gives coefficients for ith row of AB
- ith row of AB is ith row of A times B
- blending measurements

Columns of matrix product

$$(AB)_{*j} = \sum_{k=1}^{p} A_{*k} B_{kj} = AB_{*j}$$

- columns of AB are linear combinations of columns of A
- jth column of B gives coefficients for jth column of AB
- jth column of AB is A times jth column of B
- effects of secondary inputs

Signal flow



- $A_{ik}B_{kj}$ is gain from input x_i to output y_j through z_k
- $(AB)_{ij} = \sum_{k=1}^{p} A_{ik}B_{kj}$ is total gain from input x_i to output y_j