ISTA 421/521
Introduction to Machine Learning

Lecture 3:
Least Mean Squares in Higher Dimensions

Clay Morrison
clayton@sista.arizona.edu
Gould-Simpson 811
Phone 621-6609

2 September 2014

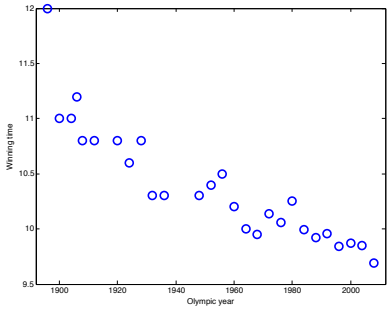
Next Topics

- Finish derivation of LMS fit (the "normal equations") for the single variable, 2 parameter **linear model**
- Moving to higher dimensions
  - Linear Algebra: matrix operators
  - Some Geometry of Linear Algebra
  - Least Mean Squares in Matrix formulation
  - The Geometry of LMS solution
- Nonlinear Response
- Generalization and Overfitting
- Regularized Least Squares

Recall: Least Mean Squares

(for single variable, 2 parameter linear model)

$$f(x; w_0, w_1) = w_0 + w_1 x$$



$$\begin{aligned}
\mathcal{L} &= \frac{1}{N}\sum_{n=1}^{N} \mathcal{L}_n(t_n, f(x_n; w_0, w_1)) \\
&= \frac{1}{N}\sum_{n=1}^{N} (t_n - f(x_n; w_0, w_1))^2 \quad \text{The specific loss fn: squared error} \\
&= \frac{1}{N}\sum_{n=1}^{N} (t_n - (w_0 + w_1 x_n))^2 \quad \text{The specific model we're working with: linear single var, 2 param (line)} \\
&= \frac{1}{N}\sum_{n=1}^{N} (w_1^2 x_n^2 + 2 w_1 x_n w_0 - 2 w_1 x_n t_n + w_0^2 - 2 w_0 t_n + t_n^2) \quad \text{Multiply out and re-arrange to put into an easier-to deal with form.} \\
&= \frac{1}{N}\sum_{n=1}^{N} (w_1^2 x_n^2 + 2 w_1 x_n (w_0 - t_n) + w_0^2 - 2 w_0 t_n + t_n^2)
\end{aligned}$$

Solving LMS: Method 1 (analytic)

(for single variable, 2 parameter linear model)

$$f(x; w_0, w_1) = w_0 + w_1 x$$ ← Our model family

$$\mathcal{L} = \frac{1}{N}\sum_{n=1}^{N} (w_1^2 x_n^2 + 2 w_1 x_n (w_0 - t_n) + w_0^2 - 2 w_0 t_n + t_n^2)$$ ← Our loss fn

Our goal: We want values for w_0 and w_1 that will *minimize* this loss function

I.e., we seek values for w_0 and w_1 that will make the loss function be the smallest when we actually sum over all the values of *x* and *t* in the dataset.

Because the loss function happens to be quadratic (in the two parameters) we can use a standard method from calculus for finding minima (maxima) directly: taking the derivative of the function and setting it to zero.

Our loss function has **two** parameters that we're trying set to minimize the loss fn, so we need to take the partial derivative (w.r.t. w_0 and w_1)

What we end up with are two functions, one for w_0 and one for w_1, and both will work with any data and give the best least mean square (LMS) fit!

Finishing…

Least Mean Squares Solution

(for single variable, 2 parameter linear model)

$$\mathcal{L} = \frac{1}{N}\sum_{n=1}^{N}(w_1^2 x_n^2 + 2w_1 x_n(w_0 - t_n) + w_0^2 - 2w_0 t_n + t_n^2)$$

(Recall: $w_0 = \bar{t} - w_1\bar{x}$)

- Partial derivative for w_1

Do the same for w_1 …

$$\frac{1}{N}\sum_{n=1}^{N}\left[w_1^2 x_n^2 + 2w_1 x_n w_0 - 2w_1 x_n t_n\right]$$ only keep terms with w_1

$$w_1^2\frac{1}{N}\left(\sum_{n=1}^{N} x_n^2\right) + 2w_1\frac{1}{N}\left(\sum_{n=1}^{N} x_n(w_0 - t_n)\right)$$ move sums inside

$$\frac{\partial\mathcal{L}}{\partial w_1} = 2w_1\frac{1}{N}\left(\sum_{n=1}^{N} x_n^2\right) + \frac{2}{N}\left(\sum_{n=1}^{N} x_n(w_0 - t_n)\right)$$ now take partial derivative

$$= w_1\frac{2}{N}\left(\sum_{n=1}^{N} x_n^2\right) + \frac{2}{N}\left(\sum_{n=1}^{N} x_n\left(\overbrace{\bar{t} - w_1\bar{x}}^{\text{solution for } w_0} - t_n\right)\right)$$ plug in solution for w_0

… and rearrange terms

$$= w_1\frac{2}{N}\left(\sum_{n=1}^{N} x_n^2\right) + \bar{t}\frac{2}{N}\left(\sum_{n=1}^{N} x_n\right) - w_1\bar{x}\frac{2}{N}\left(\sum_{n=1}^{N} x_n\right) - \frac{2}{N}\left(\sum_{n=1}^{N} x_n t_n\right)$$

- Partial derivative for w_1 continued…

$$\frac{\partial\mathcal{L}}{\partial w_1} = w_1\frac{2}{N}\left(\sum_{n=1}^{N} x_n^2\right) + \bar{t}\frac{2}{N}\left(\sum_{n=1}^{N} x_n\right) - w_1\bar{x}\frac{2}{N}\left(\sum_{n=1}^{N} x_n\right) - \frac{2}{N}\left(\sum_{n=1}^{N} x_n t_n\right)$$

$$= 2w_1\left[\left(\frac{1}{N}\sum_{n=1}^{N} x_n^2\right) - \bar{x}\,\bar{x}\right] + 2\bar{t}\bar{x} - 2\frac{1}{N}\left(\sum_{n=1}^{N} x_n t_n\right)$$ replace remaining mean x with $\bar{x}$ and group w_1 terms

$$2w_1\left[\left(\frac{1}{N}\sum_{n=1}^{N} x_n^2\right) - \bar{x}\,\bar{x}\right] + 2\bar{t}\bar{x} - 2\frac{1}{N}\left(\sum_{n=1}^{N} x_n t_n\right) = 0$$ Solve for w_1 with $\frac{\partial\mathcal{L}}{\partial w_1} = 0$ …

$$2w_1\left[\left(\frac{1}{N}\sum_{n=1}^{N} x_n^2\right) - \bar{x}\,\bar{x}\right] = 2\frac{1}{N}\left(\sum_{n=1}^{N} x_n t_n\right) - 2\bar{t}\bar{x}$$

$$w_1 = \frac{\frac{1}{N}\left(\sum_{n=1}^{N} x_n t_n\right) - \bar{t}\bar{x}}{\left(\frac{1}{N}\sum_{n=1}^{N} x_n^2\right) - \bar{x}\,\bar{x}} = \frac{\left(\frac{1}{N}\sum_{n=1}^{N} x_n t_n\right) - \left(\frac{1}{N}\sum_{m=1}^{N} t_n\right)\left(\frac{1}{N}\sum_{m=1}^{N} x_n\right)}{\left(\frac{1}{N}\sum_{n=1}^{N} x_n^2\right) - \left(\frac{1}{N}\sum_{n=1}^{N} x_n\right)^2}$$

Solving LMS: Method 1 (analytic)

(for single variable, 2 parameter linear model)

$$\mathcal{L} = \frac{1}{N}\sum_{n=1}^{N}(w_1^2 x_n^2 + 2w_1 x_n(w_0 - t_n) + w_0^2 - 2w_0 t_n + t_n^2)$$

- Partial derivative for w_0:

$$\frac{\partial \mathcal{L}}{\partial w_0} = 2w_0 + 2w_1 \frac{1}{N}\left(\sum_{n=1}^{N} x_n\right) - \frac{2}{N}\left(\sum_{n=1}^{N} t_n\right)$$

- Set $\frac{\partial \mathcal{L}}{\partial w_0} = 0$ and solve for w_0:

$$w_0 = \frac{1}{N}\left(\sum_{n=1}^{N} t_n\right) - w_1 \frac{1}{N}\left(\sum_{n=1}^{N} x_n\right) = \bar{t} - w_1 \bar{x}$$

The so-called **"normal equations"**!

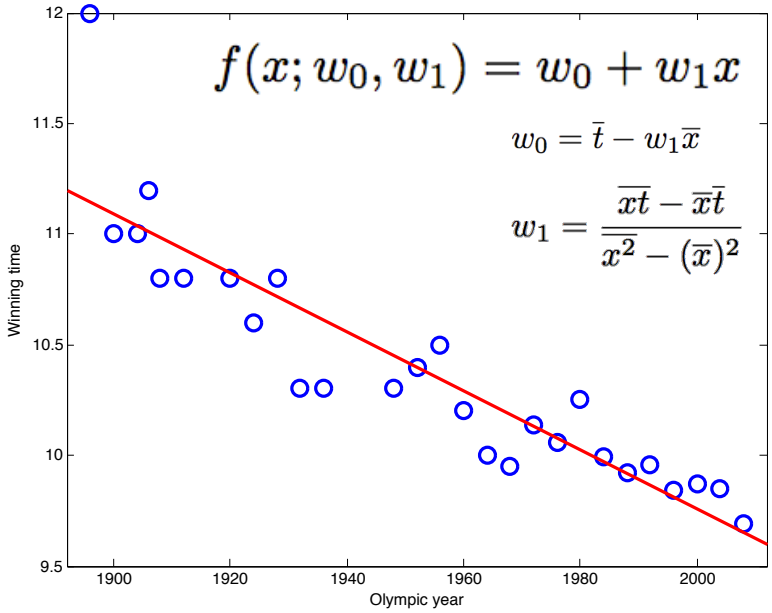
- Partial derivative for w_1:

$$\frac{\partial \mathcal{L}}{\partial w_1} = 2w_1 \frac{1}{N}\left(\sum_{n=1}^{N} x_n^2\right) + \frac{2}{N}\left(\sum_{n=1}^{N} x_n(w_0 - t_n)\right)$$

- Plug in w_0, set $\frac{\partial \mathcal{L}}{\partial w_1} = 0$ and solve for w_1:

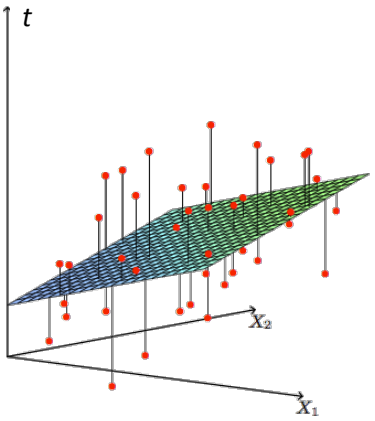
$$w_1 = \frac{\left(\frac{1}{N}\sum_{n=1}^{N} x_n t_n\right) - \left(\frac{1}{N}\sum_{m=1}^{N} t_n\right)\left(\frac{1}{N}\sum_{m=1}^{N} x_n\right)}{\left(\frac{1}{N}\sum_{n=1}^{N} x_n^2\right) - \left(\frac{1}{N}\sum_{n=1}^{N} x_n\right)^2} = \frac{\overline{xt} - \bar{x}\bar{t}}{\overline{x^2} - (\bar{x})^2}$$

Linear Least Mean Square Normal Equations



How about more than 1 input?

- Most problems will involve more than just the relationship between 1 input attribute and a target.
- Extending our linear models to higher dimensions is desirable, and at least for 2 inputs (now the "line" is a plane in 3D) it is easy to visualize the geometry.
- In general, a linear model with ***n*** input variables and ***n***+1 parameters (the ***w***'s, with their values determined) is an ***n***-dimensional "**hyperplane**" embedded in ***n***+1 dimensions.



Things quickly get messy as we increase the dimensions…

- Suppose we want a richer predictive model for the Olympic data: not only include the best overall time for the gold, but also the best times of each sprinter that raced (s_1,…,s_8)
- This is a 9 dimensional hyperplane with 10 parameters:

$$t = f(x, s_1, ..., s_8; w_0, ..., w_9) = w_0 + w_1 x + w_2 s_1 + w_3 s_2 + w_4 s_3 + w_5 s_4 + w_6 s_5 + w_7 s_6 + w_8 s_7 + w_9 s_8$$

- The math is fundamentally the same, but to derive the normal equations, we need to take 10 partial derivatives, then have 10 equations to re-arrange and substitute back in…

Matrix Product

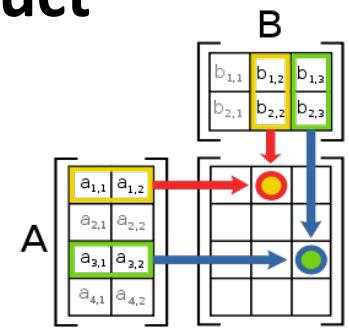
Let $\mathbf{C} = \mathbf{AB}$, where

$\mathbf{A}$ is a $N \times P$ matrix

$\mathbf{B}$ is a $P \times M$ matrix

$\mathbf{C}$ is a $N \times M$ matrix

and each entry C of $\mathbf{C}$ is: $C_{ij} = \sum_k A_{ik} B_{ky}$



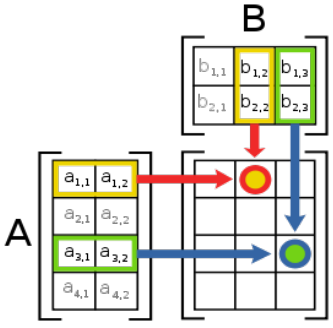
$$\begin{bmatrix} & & \\ & & \\ b_{11} & b_{12} & b_{13} \\ b_{21} & b_{22} & b_{23} \end{bmatrix}$$

$$\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \begin{bmatrix} a_{11}b_{11} + a_{12}b_{21} & a_{11}b_{12} + a_{12}b_{22} & a_{11}b_{13} + a_{12}b_{23} \\ a_{21}b_{11} + a_{22}b_{21} & a_{21}b_{12} + a_{22}b_{22} & a_{21}b_{13} + a_{22}b_{23} \end{bmatrix}$$

Book has small error on p.18: copies first line, but index for a's become a_{21}, a_{22}

Inner versus Outer Product

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} \qquad \mathbf{x}^\top \begin{bmatrix} x_1 & x_2 & \cdots & x_n \end{bmatrix}$$



Inner (dot) product

$$\mathbf{x}^\top \mathbf{x} = \sum_{n=1}^{N} x_n^2$$

Outer product

$$\mathbf{x}\mathbf{x}^\top = \begin{bmatrix} x_1^2 & x_1 x_2 & \cdots & x_1 x_n \\ x_2 x_1 & x_2^2 & \cdots & x_2 x_n \\ \vdots & \vdots & \ddots & \vdots \\ x_n x_1 & x_n x_2 & \cdots & x_n^2 \end{bmatrix}$$

Simple Linear Model in Matrix Notation

- First, express our original 1-variable, 2-param model in matrix notation:

$$\mathbf{w} = \begin{bmatrix} w_0 \\ w_1 \end{bmatrix} \qquad \mathbf{x}_n = \begin{bmatrix} 1 \\ x_n \end{bmatrix}$$

$$f(x_n; w_0, w_1) = \mathbf{w}^\top \mathbf{x}_n = w_0 + w_1 x_n$$

$$\mathcal{L} = \frac{1}{N} \sum_{n=1}^{N} (t_n - \mathbf{w}^\top \mathbf{x}_n)^2$$

Simple Linear Model in Matrix Notation

$$\mathcal{L} = \frac{1}{N} \sum_{n=1}^{N} (t_n - \mathbf{w}^\top \mathbf{x}_n)^2$$

- Next, express the operations involving all of the data (the inputs $\mathbf{x}_n$ and the targets $\mathbf{t}_n$):

$$\mathbf{w} = \begin{bmatrix} w_0 \\ w_1 \end{bmatrix}, \quad \mathbf{x}_n = \begin{bmatrix} 1 \\ x_n \end{bmatrix}$$

$$\mathbf{X} = \begin{bmatrix} \mathbf{x}_1^\top \\ \mathbf{x}_2^\top \\ \vdots \\ \mathbf{x}_N^\top \end{bmatrix} = \begin{bmatrix} 1 & x_1 \\ 1 & x_2 \\ \vdots & \vdots \\ 1 & x_N \end{bmatrix}, \quad \mathbf{t} = \begin{bmatrix} t_1 \\ t_2 \\ \vdots \\ t_N \end{bmatrix}$$

$$\mathbf{X}\mathbf{w} = \begin{bmatrix} 1 & x_1 \\ 1 & x_2 \\ \vdots & \vdots \\ 1 & x_N \end{bmatrix} \times \begin{bmatrix} w_0 \\ w_1 \end{bmatrix} = \begin{bmatrix} w_0 + w_1 x_1 \\ w_0 + w_1 x_2 \\ \vdots \\ w_0 + w_1 x_N \end{bmatrix}$$

$$\mathbf{t} - \mathbf{X}\mathbf{w} = \begin{bmatrix} t_1 - w_0 - w_1 x_1 \\ t_2 - w_0 - w_1 x_2 \\ \vdots \\ t_N - w_0 - w_1 x_N \end{bmatrix}$$

$$\begin{aligned} (\mathbf{t} - \mathbf{X}\mathbf{w})^\top (\mathbf{t} - \mathbf{X}\mathbf{w}) &= (t_1 - (w_0 + w_1 x_1))^2 + (t_2 - (w_0 + w_1 x_2))^2 + \ldots \\ &\quad + (t_N - (w_0 + w_1 x_N))^2 \\ &= \sum_{n=1}^{N} (t_n - (w_0 + w_1 x_n))^2 \\ &= \sum_{n=1}^{N} (t_n - f(x_n; w_0, w_1))^2 \end{aligned}$$

$$\mathcal{L} = \frac{1}{N} (\mathbf{t} - \mathbf{X}\mathbf{w})^\top (\mathbf{t} - \mathbf{X}\mathbf{w}) = \frac{1}{N} \sum_{n=1}^{N} (t_n - \mathbf{w}^\top \mathbf{x}_n)^2 = \frac{1}{N} \sum_{n=1}^{N} (t_n - (w_0 + w_1 x_n))^2$$

Much nicer! The $\mathbf{x}^\top\mathbf{y}$ operation allows us to drop the sums!

Simple Linear Model in Matrix Notation

- Now that we have the matrix version of the loss function, "just" take derivative…

$$\begin{aligned}\mathcal{L} &= \frac{1}{N}(\mathbf{Xw}-\mathbf{t})^\top(\mathbf{Xw}-\mathbf{t}) \\ &= \frac{1}{N}((\mathbf{Xw})^\top-\mathbf{t}^\top)(\mathbf{Xw}-\mathbf{t}) \\ &= \frac{1}{N}(\mathbf{Xw})^\top\mathbf{Xw}-\frac{1}{N}\mathbf{t}^\top\mathbf{Xw}-\frac{1}{N}(\mathbf{Xw})^\top\mathbf{t}+\frac{1}{N}\mathbf{t}^\top\mathbf{t} \\ &= \frac{1}{N}\mathbf{w}^\top\mathbf{X}^\top\mathbf{Xw}-\frac{2}{N}\mathbf{w}^\top\mathbf{X}^\top\mathbf{t}+\frac{1}{N}\mathbf{t}^\top\mathbf{t}\end{aligned}$$

note: book accidentally drops the 1/N here

$\mathbf{t}^\top\mathbf{Xw}$ and $\mathbf{w}^\top\mathbf{X}^\top\mathbf{t}$ are the transpose of one another … and both products come out to be scalars (i.e., "1x1 matrices", where transpose of one is the same as the other), so the products are the same and can be combined.

$$\frac{\partial\mathcal{L}}{\partial\mathbf{w}} = \begin{bmatrix}\frac{\partial\mathcal{L}}{\partial w_0} \\ \frac{\partial\mathcal{L}}{\partial w_1}\end{bmatrix}$$

$f(\mathbf{w})$	$\frac{\partial f}{\partial \mathbf{w}}$
$\mathbf{w}^\top\mathbf{x}$	$\mathbf{x}$
$\mathbf{x}^\top\mathbf{w}$	$\mathbf{x}$
$\mathbf{w}^\top\mathbf{w}$	$2\mathbf{w}$
$\mathbf{w}^\top\mathbf{Cw}$	$2\mathbf{Cw}$

Some useful identities

$$\frac{\partial\mathcal{L}}{\partial\mathbf{w}} = \frac{2}{N}\mathbf{X}^\top\mathbf{Xw}-\frac{2}{N}\mathbf{X}^\top\mathbf{t}=0$$
$$\mathbf{X}^\top\mathbf{Xw}=\mathbf{X}^\top\mathbf{t}.$$
$$\mathbf{Iw}=(\mathbf{X}^\top\mathbf{X})^{-1}\mathbf{X}^\top\mathbf{t}$$

The **matrix normal equation**! (guaranteed unique solution when the *n* column vectors of **X** are *linearly independent*)

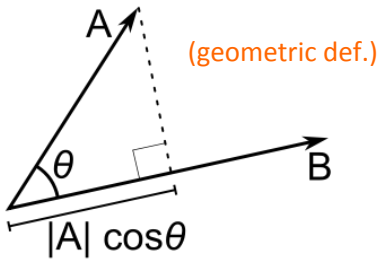
But what does it mean??

Relation of $\mathbf{a}^\top\mathbf{b}$ to Geometry

- $\mathbf{a}^\top\mathbf{b}$ is special (also $\mathbf{a}\cdot\mathbf{b}$), called the *dot product* (aka scalar product; the *inner product* for the Euclidean space)

$$\mathbf{a}\cdot\mathbf{b}=\sum_{i=1}^{n}a_ib_i=a_1b_1+a_2b_2+\cdots+a_nb_n$$ (algebraic def.)

- Plays a role in defining
  - Euclidean distance (norm)
  - Angles



(geometric def.)

$$\mathbf{a}\cdot\mathbf{a}=\|\mathbf{a}\|^2$$
$$\mathbf{a}\cdot\mathbf{b}=\|\mathbf{a}\|\,\|\mathbf{b}\|\cos\theta$$
$$\theta=\arccos\left(\frac{\mathbf{a}\cdot\mathbf{b}}{\|\mathbf{a}\|\,\|\mathbf{b}\|}\right).$$

The dot product of vectors that are 90° (or more generally, orthogonal) is = 0

Geometry of Linear Systems and their Solution

A linear equation expresses a constraint between variables

A system of linear equations – more constraints!

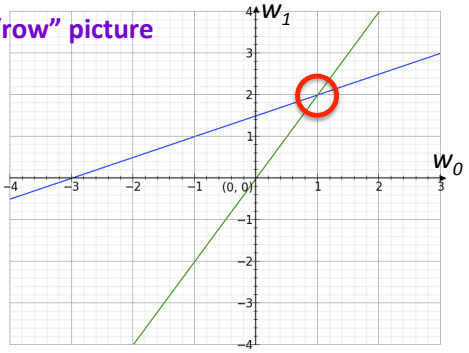
$$2w_0 - w_1 = 0$$
$$-w_0 + 2w_1 = 3$$

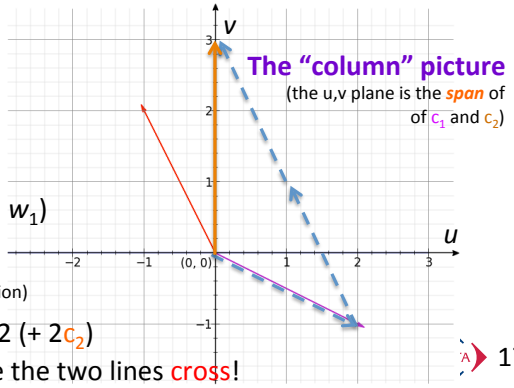
$$\begin{matrix} u \\ v \end{matrix}\begin{bmatrix} 2 & -1 \\ -1 & 2 \end{bmatrix}\begin{bmatrix} w_0 \\ w_1 \end{bmatrix} = \begin{bmatrix} 0 \\ 3 \end{bmatrix}$$

c_1 c_2

X w = t **w = X^{-1} t**

The "row" picture





The "column" picture
(the u,v plane is the ***span*** of of c_1 and c_2)

"Solving" the linear system involves finding the linear combinations (i.e., the amounts w_0 and w_1) of c_1 and c_2 that equal the column vector $(0,3)^T$.

Solve using your favorite method (e.g., Gaussian Elimination)

Here, the solution happens to be w_0=1 (1 c_1), w_1=2 (+ 2c_2)

The w_0's and w_1's corresponds to the point where the two lines cross!

Geometry of Linear Systems and their Solution

But what happens if we're **over-constrained**?

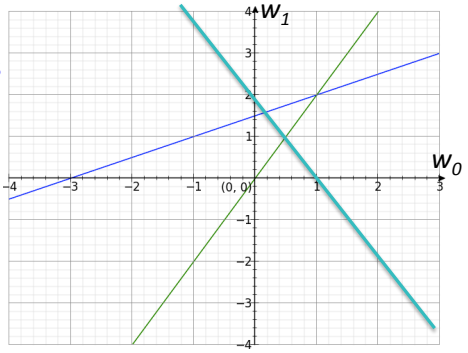
GOAL: Find a solution that is ***closest*** to (minimizes the distance between) the crossing points!

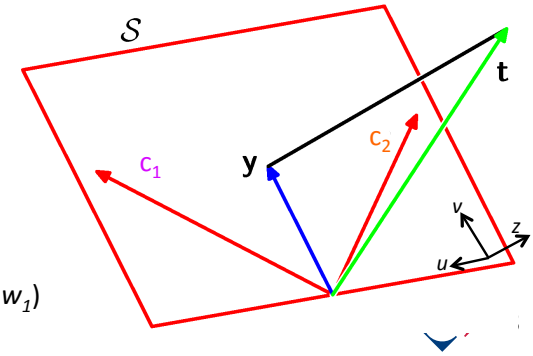
$$2w_0 - w_1 = 0$$
$$-w_0 + 2w_1 = 3$$
$$2w_0 + w_1 = 2$$

$$\begin{matrix} u \\ v \\ z \end{matrix}\begin{bmatrix} 2 & -1 \\ -1 & 2 \\ 2 & 1 \end{bmatrix}\begin{bmatrix} w_0 \\ w_1 \end{bmatrix} = \begin{bmatrix} 0 \\ 3 \\ 2 \end{bmatrix}$$

c_1 c_2 t

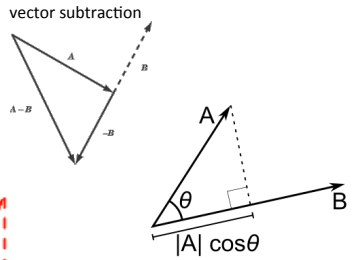
X w = t **X $\hat{w}$ = y**





"Solving" the linear system involves finding the linear combinations (i.e., the amounts w_0 and w_1) of c_1 and c_2 that equal the column vector $(0,3,2)^T$.

Finding the projection



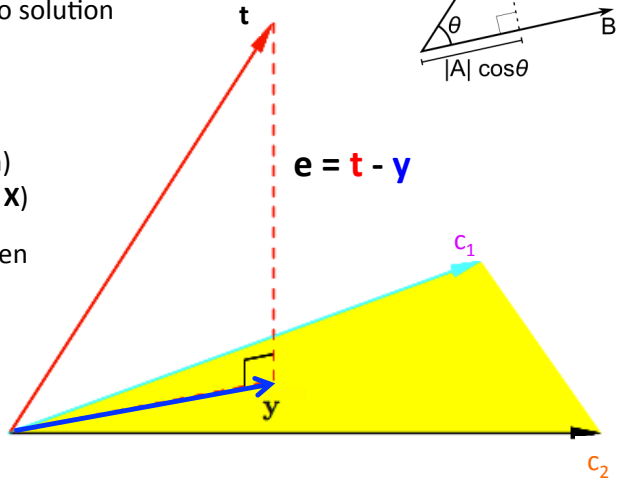
X w = **t** …what we started with, but no solution

X ŵ = **y** …something we can solve

In particular, we want **y** to be the ***closest*** to **t** but in the column space (the span) of $\mathbf{c}_1$ and $\mathbf{c}_2$ (which are the columns of **X**)

But we know the shortest distance between the end of **t** and the span is a vector **e** that is ***orthogonal*** (right angles!) to $\mathbf{c}_1$ and $\mathbf{c}_2$ (i.e., orthogonal to **X**)

That only happens when $\mathbf{e}^T\mathbf{X} = \mathbf{X}^T\mathbf{e} = 0$



Let's solve for when $\mathbf{X}^T\mathbf{e} = 0$

$\mathbf{X}^T(\mathbf{t} - \mathbf{y}) = 0$	plug in for **e**
$\mathbf{X}^T(\mathbf{t} - \mathbf{X}\hat{\mathbf{w}}) = 0$	substitute original $\mathbf{X}\hat{\mathbf{w}}=\mathbf{y}$ (since we don't know **y**)
$\mathbf{X}^T\mathbf{t} - \mathbf{X}^T\mathbf{X}\hat{\mathbf{w}} = 0$	multiply through…
$\mathbf{X}^T\mathbf{X}\hat{\mathbf{w}} = \mathbf{X}^t\mathbf{t}$	
$\hat{\mathbf{w}} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^t\mathbf{t}$	… ah hah! $\mathbf{I}\mathbf{w} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{t}$

Finding the projection

Parameters of hyperplane

$$\begin{matrix} 2w_0 & - & w_1 & = & 0 \\ -w_0 & + & 2w_1 & = & 3 \\ 2w_0 & + & w_1 & = & 2 \end{matrix}$$

inputs | targets

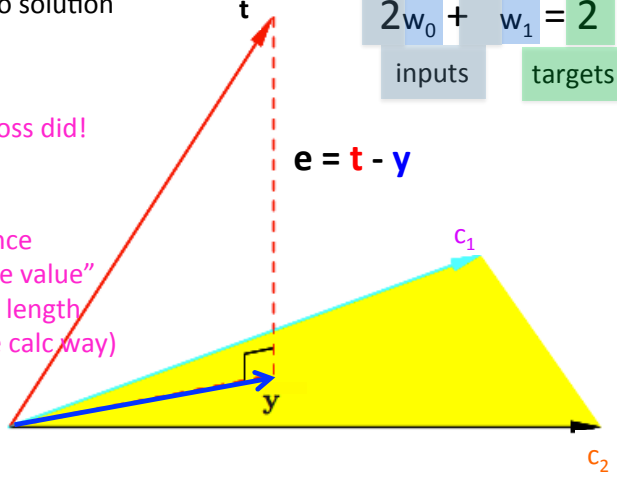
X w = **t** …what we started with, but no solution

X ŵ = **y** …something we can solve

This is just what minimizing the squared loss did!

$$\mathcal{L} = \frac{1}{N}(\mathbf{t} - \mathbf{X}\mathbf{w})^T(\mathbf{t} - \mathbf{X}\mathbf{w})$$

This "error" vector **e** is the shortest distance
(Note: we can't differentiate the "absolute value" so the squaring of the difference – of the length of **e** – was a better choice for doing it the calc way)



That only happens when $\mathbf{e}^T\mathbf{X} = \mathbf{X}^T\mathbf{e} = 0$

Let's solve for when $\mathbf{X}^T\mathbf{e} = 0$

$\mathbf{X}^T(\mathbf{t} - \mathbf{y}) = 0$	plug in for **e**
$\mathbf{X}^T(\mathbf{t} - \mathbf{X}\hat{\mathbf{w}}) = 0$	substitute original $\mathbf{X}\hat{\mathbf{w}}=\mathbf{y}$ (since we don't know **y**)
$\mathbf{X}^T\mathbf{t} - \mathbf{X}^T\mathbf{X}\hat{\mathbf{w}} = 0$	multiply through…
$\mathbf{X}^T\mathbf{X}\hat{\mathbf{w}} = \mathbf{X}^t\mathbf{t}$	
$\hat{\mathbf{w}} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^t\mathbf{t}$	… ah hah! $\mathbf{I}\mathbf{w} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{t}$