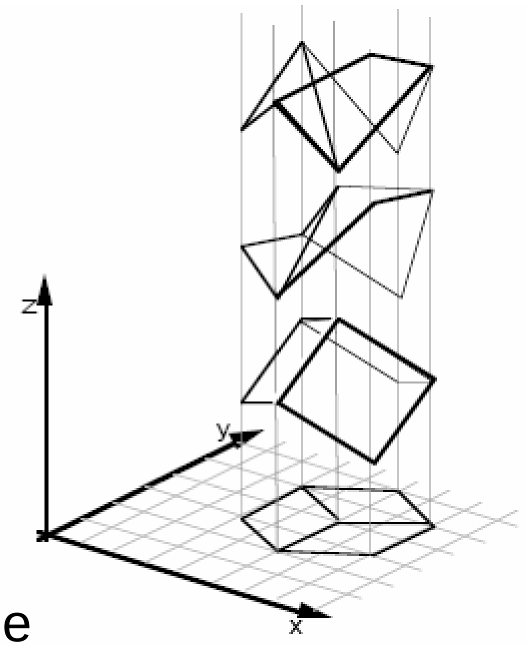
Least squares

Nuno Vasconcelos
UCSD

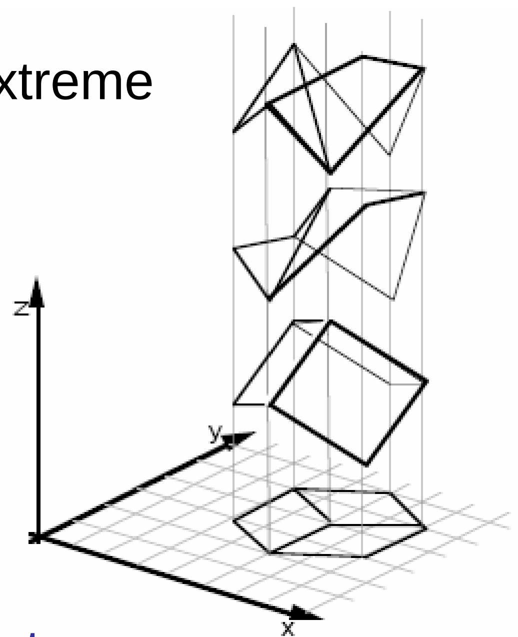
Model fitting

- one common problem in signal processing is to fit a model to a signal
  - in vision, we typically have a scene
  - it contains some "signal", which is what we are trying to understand about the scene
  - but it also contains "noise"
- typically we have a model for our signal
  - e.g. the planes the planes that we used to model the wall in PS 2
  - we saw that going from 3D to 2D is a relatively easy problem
  - vision is the opposite: I give you the image and you tell me what the 3D planes are
  - a lot harder, many scenes could fit the image



Model fitting

- we typically need to make assumptions on the scene
- these are usually in the form of a model
- while models are great help, the real world is never exactly like we modeled it, due to
  - 1) noise in the imaging process: usually not a major concern, unless the scenario is extreme (night vision, underwater, bad weather)
  - 2) deviations from the model: no model is perfect, e.g. the sun is really not a point source and is not really infinitely far away
  - this is usually the greater source of concern
  - we model what we can and assume that the rest is “noise”



- hence, we need to fit our models to the data
  - in an optimal manner, that minimizes errors due to noise

Regression

- model fitting is a regression problem
- in a regression problem we have
  - two random variables X and Y
  - a dataset of examples $\mathcal{D} = \{(x_1,y_1), \dots (x_n,y_n)\}$
  - a parametric model of the form

$$\boxed{y = f(x;\Theta) + \varepsilon}$$

  - where Θ is a parameter vector, and ε a random variable that accounts for noise
- two types of problems
  - linear regression: when f(.) is linear on Θ
  - non-linear regression: otherwise
  - note that what matters is linearity on Θ, not on *X*!

Examples

- linear regression:
  - line fitting

$$\boxed{f(x;\Theta) = \theta_1 x + \theta_0}$$

  - polynomial fitting

$$\boxed{f(x;\Theta) = \sum_{i=0}^{K} \theta_i x^i}$$

  - truncated Fourier series

$$\boxed{f(x;\Theta) = \sum_{i=0}^{K} \theta_i \sin(ix)}$$

- non-linear regression:
  - neural networks

$$\boxed{f(x;\Theta) = \frac{1}{1+e^{-\theta_1 x - \theta_0}}}$$

  - sinusoidal decompositions

$$\boxed{f(x;\Theta) = \sum_{i=0}^{K} \sin(\theta_i x)}$$

  - etc.

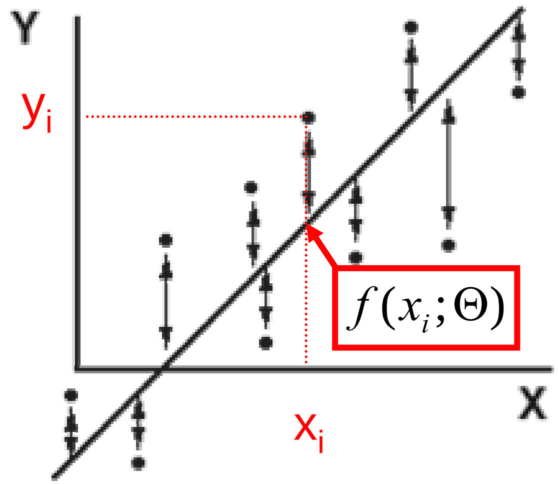
Example

- let's consider the problem of line fitting
  - the model is

$$\boxed{f(x;\Theta) = \theta_1 x + \theta_0}$$

  - we are given a set of points

$$D = \{(x_1,y_1),\ldots,(x_n,y_n)\}$$



  - the error of the fit is

$$\boxed{L = \sum_i \left(y_i - f(x_i;\Theta)\right)^2 = \sum_i \left(y_i - \theta_1 x_i - \theta_0\right)^2}$$

  - we are looking for the line that makes these distances as small as possible

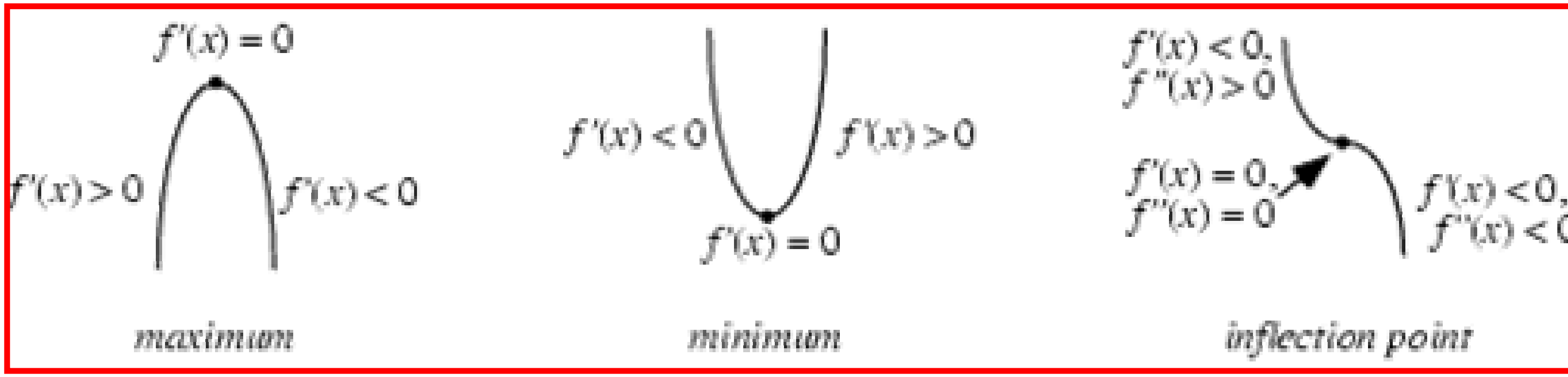
$$\boxed{L^* = \min_{\Theta} \sum_i \left(y_i - f(x_i;\Theta)\right)^2 = \min_{\theta_1,\theta_2} \sum_i \left(y_i - \theta_1 x_i - \theta_0\right)^2}$$

Optimization

- minimizing a function

$$\boxed{\min_x f(x)}$$

- when x is a scalar is high-school calculus



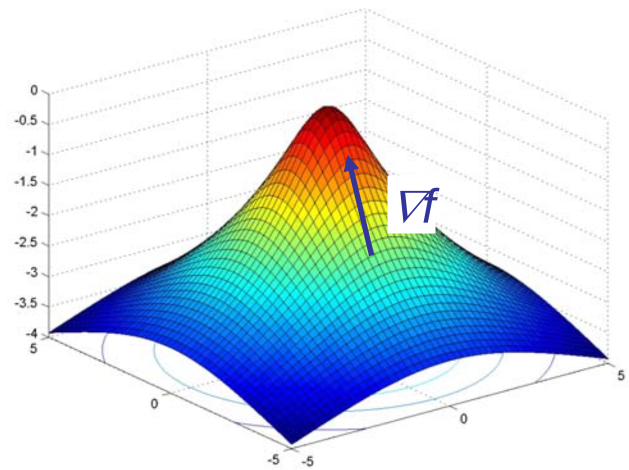
- we have a maximum when
  - first derivative is zero
  - second derivative is negative

The gradient

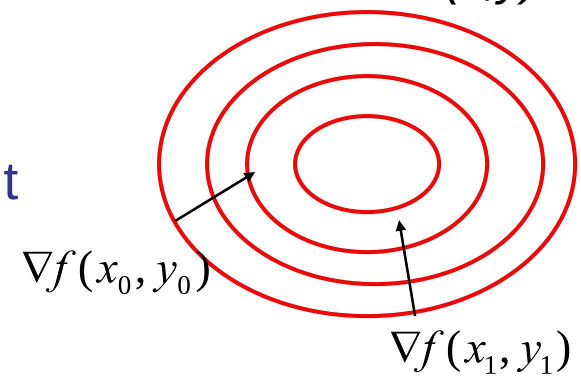
- in higher dimensions, the generalization of the derivative is the gradient
- the gradient of a function *f(w)* at *z* is

$$\nabla f(z) = \left(\frac{\partial f}{\partial w_0}(z), \cdots, \frac{\partial f}{\partial w_{n-1}}(z) \right)^T$$

- the gradient has a nice geometric interpretation
  - it points in the direction of maximum growth of the function
  - which makes it perpendicular to the contours where the function is constant

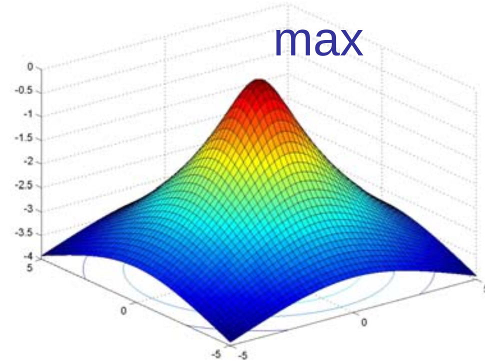


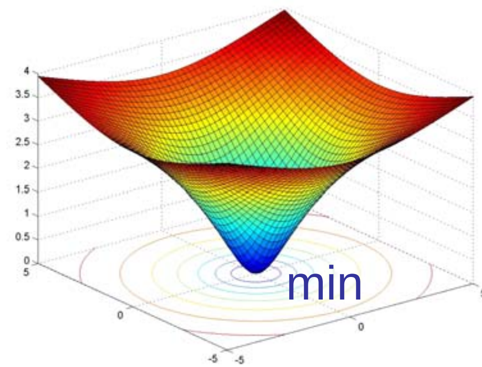
f(x,y)

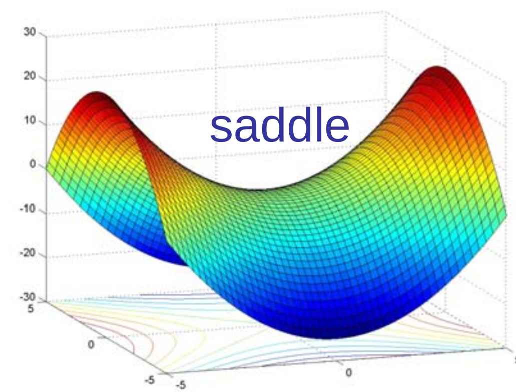


The gradient

- note that if $\nabla f = 0$
  - there is no direction of growth
  - also $-\nabla f = 0$, and there is no direction of decrease
  - we are either at a local minimum or maximum or “saddle” point
- conversely, at local min or max or saddle point
  - no direction of growth or decrease
  - $\nabla f = 0$
- this shows that we have a critical point if and only if $\nabla f = 0$
- to determine which type we need second order conditions







The Hessian

- the extension of the second-order derivative is the Hessian matrix

$$\boxed{\nabla^2 f(x) = \begin{bmatrix} \dfrac{\partial^2 f}{\partial x_0^2}(x) & \cdots & \dfrac{\partial^2 f}{\partial x_0 \partial x_{n-1}}(x) \\ & \vdots & \\ \dfrac{\partial^2 f}{\partial x_{n-1} \partial x_0}(x) & \cdots & \dfrac{\partial^2 f}{\partial x_{n-1}^2}(x) \end{bmatrix}}$$

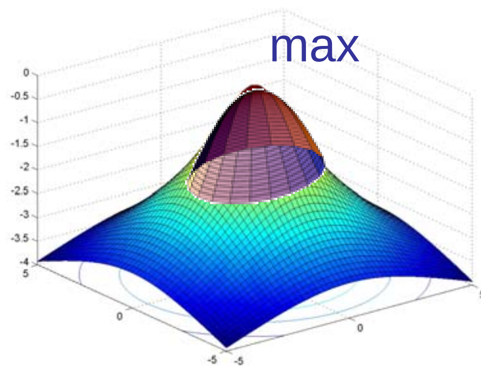
  – at each point x, gives us the quadratic function

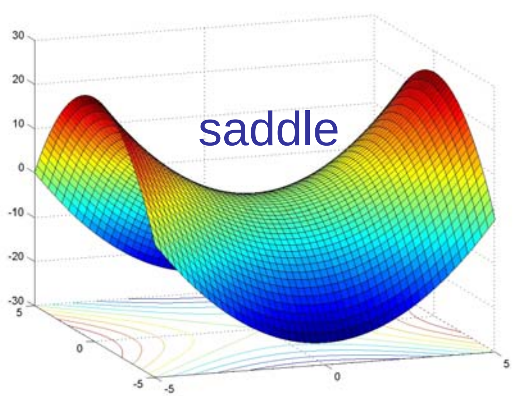
$$\boxed{x^t \nabla^2 f(x) x}$$

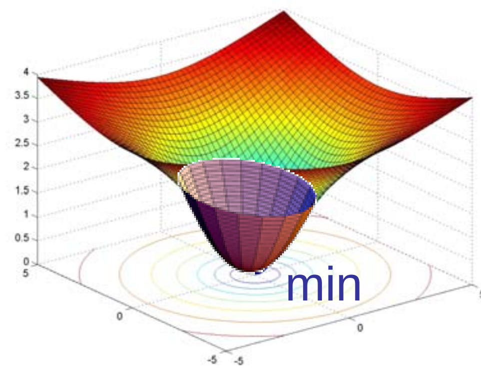
  that best approximates f(x)

The Hessian

- this means that, when gradient is zero at x, we have
  - a maximum when function can be approximated by an "upwards-facing" quadratic
  - a minimum when function can be approximated by a "downwards-facing" quadratic
  - a saddle point otherwise





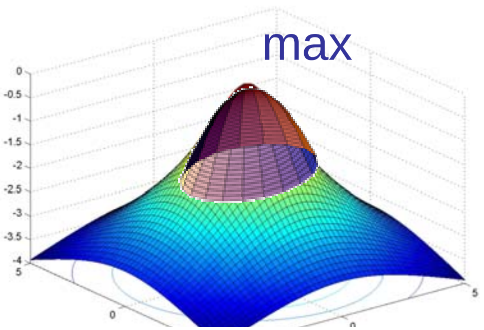


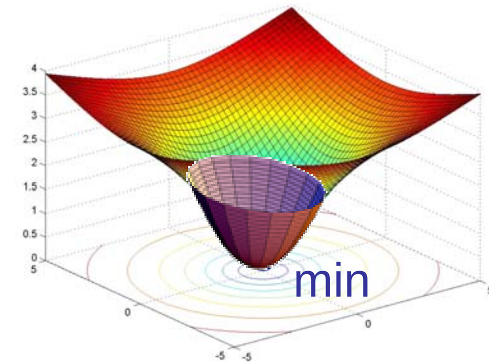
The Hessian

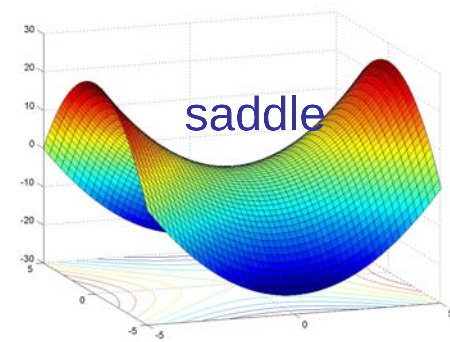
- for any matrix M, the function

$$\boxed{f(x) = x^t M x}$$

- is
  - upwards facing quadratic when M is negative definite
  - downwards facing quadratic when M is positive definite
  - saddle otherwise
- hence, all that matters is the positive definiteness of the Hessian
- we have a minimum when the Hessian is positive definite







Optimality conditions

- **Definition:** each of the following is a necessary and sufficient condition for a real symmetric matrix A to be (semi) positive definite:
    i) $x^T A x \geq 0, \ \forall x \neq 0$
    ii) all eigenvalues of A satisfy $\lambda_i \geq 0$
    iii) all upper-left submatrices A_k have non-negative determinant
    iv) there is a matrix R with independent rows such that $A = R^T R$

- upper left submatrices:

$$A_1 = a_{1,1} \qquad A_2 = \begin{bmatrix} a_{1,1} & a_{1,2} \\ a_{2,1} & a_{2,2} \end{bmatrix} \qquad A_3 = \begin{bmatrix} a_{1,1} & a_{1,2} & a_{1,3} \\ a_{2,1} & a_{2,2} & a_{2,3} \\ a_{3,1} & a_{3,2} & a_{3,3} \end{bmatrix} \qquad \cdots$$

Optimality conditions

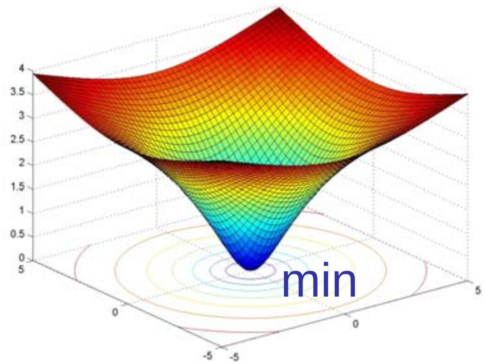
- in summary
- *w** is a local minimum of *f(w)* if and only if
  - *f* has zero gradient at *w**

$$\boxed{\nabla f(w^*) = 0}$$

  - and the Hessian of *f* at *w** is positive definite

$$\boxed{d^t \nabla^2 f(w^*) d \geq 0, \quad \forall d \in \Re^n}$$

  - where



$$\nabla^2 f(x) = \begin{bmatrix} \frac{\partial^2 f}{\partial x_0^2}(x) & \cdots & \frac{\partial^2 f}{\partial x_0 \partial x_{n-1}}(x) \\ & \vdots & \\ \frac{\partial^2 f}{\partial x_{n-1} \partial x_0}(x) & \cdots & \frac{\partial^2 f}{\partial x_{n-1}^2}(x) \end{bmatrix}$$

Example

- to solve

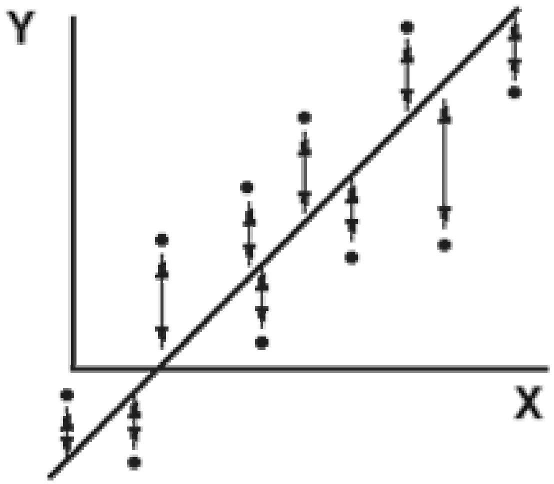
$$\boxed{L^{*} = \min_{\theta_1,\theta_2} \sum_i \left(y_i - \theta_1 x_i - \theta_0\right)^2}$$

  – we set the gradient to zero

$$\boxed{\begin{cases} \dfrac{\partial L}{\partial \theta_0} = -2\sum_i \left(y_i - \theta_1 x_i - \theta_0\right) = 0 \\ \dfrac{\partial L}{\partial \theta_1} = -2\sum_i \left(y_i - \theta_1 x_i - \theta_0\right) x_i = 0 \end{cases}}$$

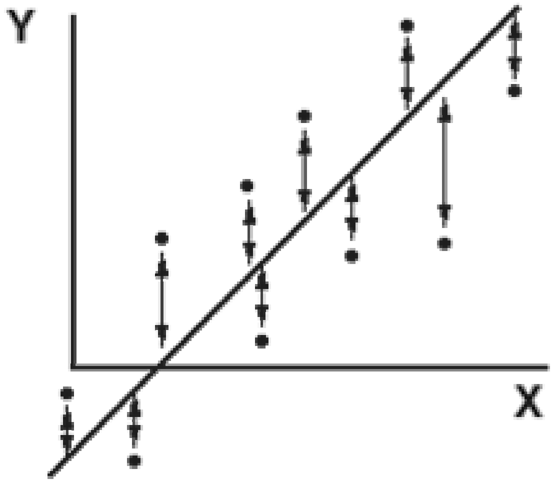
$$\downarrow$$

$$\boxed{\begin{cases} \sum_i y_i = \theta_1 \sum_i x_i + n\theta_0 \\ \sum_i y_i x_i = \theta_1 \sum_i x_i^2 + \theta_0 \sum_i x_i \end{cases}} \rightarrow \boxed{\begin{bmatrix} \frac{1}{n}\sum_i y_i \\ \frac{1}{n}\sum_i y_i x_i \end{bmatrix} = \begin{bmatrix} 1 & \frac{1}{n}\sum_i x_i \\ \frac{1}{n}\sum_i x_i & \frac{1}{n}\sum_i x_i^2 \end{bmatrix} \begin{bmatrix} \theta_0 \\ \theta_1 \end{bmatrix}}$$



Example

- $$\begin{bmatrix} \frac{1}{n}\sum_i y_i \\ \frac{1}{n}\sum_i y_i x_i \end{bmatrix} = \begin{bmatrix} 1 & \frac{1}{n}\sum_i x_i \\ \frac{1}{n}\sum_i x_i & \frac{1}{n}\sum_i x_i^2 \end{bmatrix} \begin{bmatrix} \theta_0 \\ \theta_1 \end{bmatrix}$$



- and, denoting

$$\boxed{\langle y \rangle = \frac{1}{n}\sum_i y_i, \quad \langle x^k \rangle = \frac{1}{n}\sum_i x_i^k, \quad \langle yx \rangle = \frac{1}{n}\sum_i y_i x_i}$$

- we get

$$\boxed{\begin{bmatrix} \langle y \rangle \\ \langle xy \rangle \end{bmatrix} = \begin{bmatrix} 1 & \langle x \rangle \\ \langle x \rangle & \langle x^2 \rangle \end{bmatrix} \begin{bmatrix} \theta_0 \\ \theta_1 \end{bmatrix}}$$

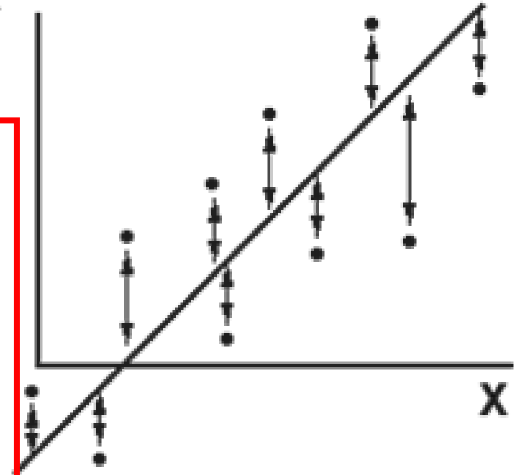
Example

- the solution is

$$\begin{bmatrix}\theta_0\\ \theta_1\end{bmatrix} = \begin{bmatrix}1 & \langle x\rangle\\ \langle x\rangle & \langle x^2\rangle\end{bmatrix}^{-1}\begin{bmatrix}\langle y\rangle\\ \langle xy\rangle\end{bmatrix}$$

$$= \frac{1}{\langle x^2\rangle - \langle x\rangle^2}\begin{bmatrix}\langle x^2\rangle & -\langle x\rangle\\ -\langle x\rangle & 1\end{bmatrix}\begin{bmatrix}\langle y\rangle\\ \langle xy\rangle\end{bmatrix}$$

$$= \frac{1}{\langle x^2\rangle - \langle x\rangle^2}\begin{bmatrix}\langle x^2\rangle\langle y\rangle - \langle x\rangle\langle xy\rangle\\ \langle xy\rangle - \langle x\rangle\langle y\rangle\end{bmatrix}$$



Least squares

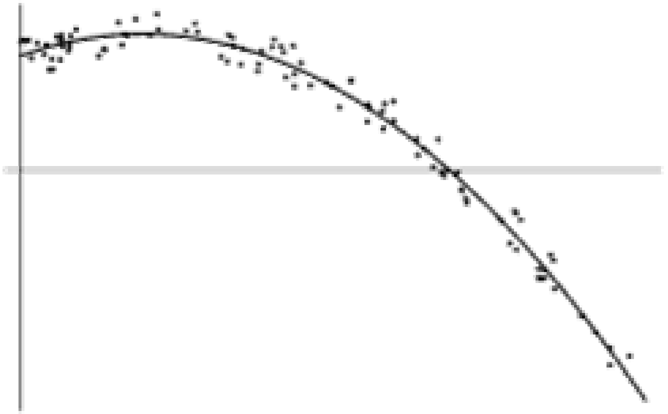
- what if I have other models?
- can we write this more generally?
  - we can write the model

$$f(x;\Theta) = \theta_1 x + \theta_0$$

  - as

$$\boxed{f(x;\Theta) = \gamma(x)^T \Theta}$$

$$\gamma(x) = \begin{bmatrix} 1 \\ x \end{bmatrix} \qquad \Theta = \begin{bmatrix} \theta_0 \\ \theta_1 \end{bmatrix}$$



- this can be generalized to any model if we make

$$\boxed{\gamma(x) = \begin{bmatrix} \gamma_0(x) \\ \vdots \\ \gamma_k(x) \end{bmatrix} \qquad \Theta = \begin{bmatrix} \theta_0 \\ \vdots \\ \theta_k \end{bmatrix}}$$

Examples

- note that the $\gamma(x)$ can be arbitrary non-linear functions of x
  - line fitting

$$\boxed{f(x;\Theta) = \theta_1 x + \theta_0}$$

$$\boxed{\gamma(x)^T = [1 \quad x]}$$

  - polynomial fitting

$$\boxed{f(x;\Theta) = \sum_{i=0}^{K} \theta_i x^i}$$

$$\boxed{\gamma(x)^T = \begin{bmatrix} 1 & \ldots & x^K \end{bmatrix}}$$

  - truncated Fourier series

$$\boxed{f(x;\Theta) = \sum_{i=0}^{K} \theta_i \sin(ix)}$$

$$\boxed{\gamma(x)^T = \begin{bmatrix} 0 & \ldots & \sin(Kx) \end{bmatrix}}$$

Least squares

- we can write the error

$$L = \sum_i \left(y_i - \theta_1 x_i - \theta_0\right)^2$$

- as

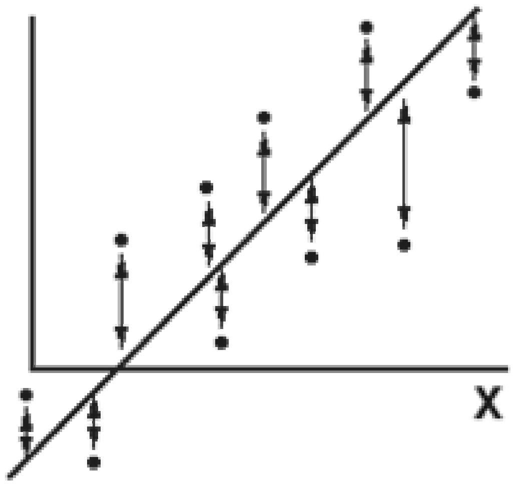
$$L = \sum_i \left(y_i - \gamma(x_i)^T \Theta\right)^2$$

- or

$$\boxed{L = \left\| y - \Gamma(x)\Theta \right\|^2}$$

- where

$$\boxed{y = \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix} \qquad \Gamma(x) = \begin{bmatrix} \gamma(x_1)^T \\ \vdots \\ \gamma(x_n)^T \end{bmatrix} \qquad \Theta = \begin{bmatrix} \theta_0 \\ \vdots \\ \theta_k \end{bmatrix}}$$



Examples

- the most important component is the matrix *Γ(x)*
  - line fitting
    polynomial fitting

$$\boxed{\Gamma(x) = \begin{bmatrix} 1 \, x_1 \\ \vdots \;\; \vdots \\ 1 \, x_n \end{bmatrix}}$$

$$\boxed{\Gamma(x) = \begin{bmatrix} 1 & \dots & x_1^K \\ & \vdots & \\ 1 & \dots & x_n^K \end{bmatrix}}$$

  - truncated Fourier series

$$\boxed{\Gamma(x) = \begin{bmatrix} 0 & \dots & \sin(Kx_1) \\ & \vdots & \\ 0 & \dots & \sin(Kx_n) \end{bmatrix}}$$

Matrix derivatives

- to compute the gradient and Hessian it is useful to rely on matrix derivatives
- some examples that we will use

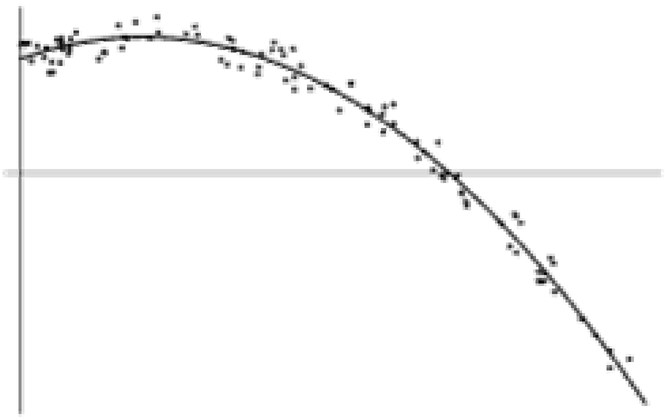
$$\boxed{\nabla_{\Theta}\left(A\Theta\right) = A^T}$$

$$\boxed{\nabla_{\Theta}\left(\Theta^T A\Theta\right) = (A + A^T)\Theta}$$

$$\boxed{\nabla_{\Theta}\left\|b - A\Theta\right\|^2 = -2A^T(b - A\Theta)}$$

- there are various lists of the most popular formulas
- one example is http://www.ee.ic.ac.uk/hp/staff/dmb/matrix/calculus.html

Least squares



- in summary, we always have

$$\boxed{L = \left\| y - \Gamma(x)\Theta \right\|^2}$$

- to minimize this we simply have to find x such that

$$\nabla_\Theta L = -2\Gamma(x)^T \left[y - \Gamma(x)\Theta \right] = 0$$

or

$$\Gamma(x)^T \Gamma(x)\Theta = \Gamma(x)^T y$$

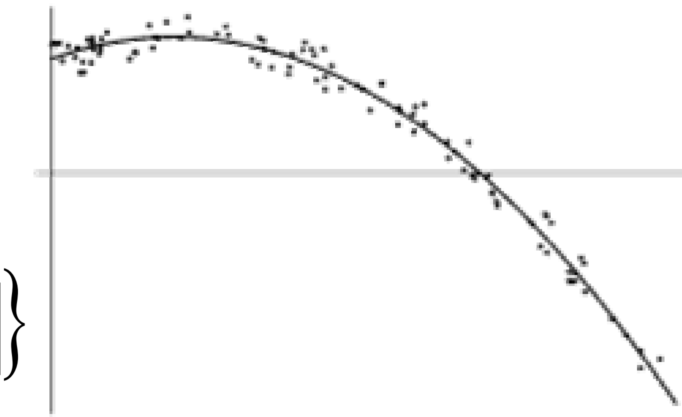
from which, as long as $\Gamma(x)^T \Gamma(X)$ is invertible,

$$\boxed{\Theta^* = \left[\Gamma(x)^T \Gamma(x) \right]^{-1} \Gamma(x)^T y}$$

Least squares

- we next check the Hessian

$$\nabla^2_\Theta L = \nabla_\Theta \left(\nabla_\Theta L\right)$$
$$= -2\nabla_\Theta \left\{\Gamma(x)^T \left[y - \Gamma(x)\Theta\right]\right\}$$
$$= 2\Gamma(x)^T \Gamma(x)$$



- this is positive definite if the rows of $\Gamma(x)$ are independent
- which turns out to be
  - the condition for $\Gamma(x)^T\Gamma(X)$ to be invertible,
  - which is the necessary condition for the solution to be feasible
- note that we design $\Gamma(x)$, so we can always make this happen
- usually we only have to make sure all the x_i are different

Least squares

- in summary
  - a problem of the type

$$\boxed{L = \left\| y - \Gamma(x)\Theta \right\|^2}$$

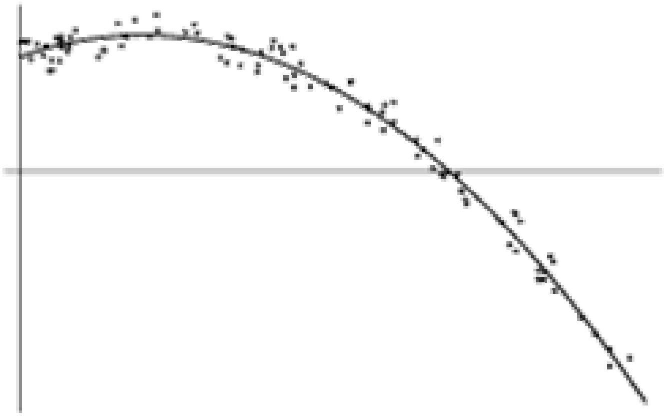
  - has least squares solution

$$\boxed{\Theta^* = \left[\Gamma(x)^T \Gamma(x)\right]^{-1} \Gamma(x)^T y}$$

  - the matrix

$$\boxed{\Gamma(x)^\Pi = \left[\Gamma(x)^T \Gamma(x)\right]^{-1} \Gamma(x)^T}$$

  - is called the pseudo-inverse of $\Gamma(x)$



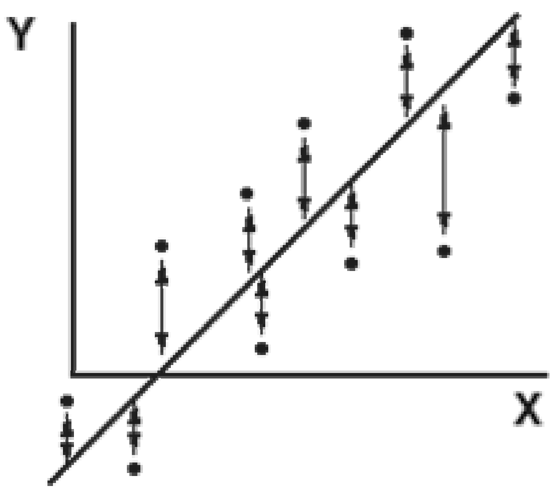
Least squares

- here is a way of thinking about this
  - we have a system of equations

$$\boxed{y = \Gamma(x)\Theta}$$

  - this cannot be solved because $\Gamma(x)$ is not invertible
  - e.g. for the line

$$\boxed{\begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix} = \begin{bmatrix} 1 & x_1 \\ \vdots & \vdots \\ 1 & x_n \end{bmatrix} \begin{bmatrix} \theta_0 \\ \theta_1 \end{bmatrix}}$$



  - we multiply both sides by $\Gamma(x)^T$

$$\boxed{\Gamma(x)^T y = \Gamma(x)^T \Gamma(x)\Theta}$$

Least squares

– this is now a solvable system

$$\begin{bmatrix} 1 & \dots & 1 \\ x_1 & \dots & x_n \end{bmatrix} \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix} = \begin{bmatrix} 1 & \dots & 1 \\ x_1 & \dots & x_n \end{bmatrix} \begin{bmatrix} 1 & x_1 \\ \vdots & \vdots \\ 1 & x_n \end{bmatrix} \begin{bmatrix} \theta_0 \\ \theta_1 \end{bmatrix}$$

– whose solution is given by the pseudo-inverse

$$\Theta^* = \left[\Gamma(x)^T \Gamma(x)\right]^{-1} \Gamma(x)^T y$$

– and we have just seen that this is the best solution for the original problem in the least squares sense

$$\Theta^* = \arg\min_{\Theta} \left\| y - \Gamma(x)\Theta \right\|^2$$

Least squares

- in general the least squares solution is quite easy to compute
- let's redo the line example

$$\Gamma(x)^T \Gamma(x) = \begin{bmatrix} 1 & \dots & 1 \\ x_1 & \dots & x_n \end{bmatrix} \begin{bmatrix} 1 \, x_1 \\ \vdots \; \vdots \\ 1 \, x_n \end{bmatrix} = n \begin{bmatrix} 1 & \langle x \rangle \\ \langle x \rangle & \langle x^2 \rangle \end{bmatrix}$$

$$\Gamma(x)^T y = \begin{bmatrix} 1 & \dots & 1 \\ x_1 & \dots & x_n \end{bmatrix} \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix} = n \begin{bmatrix} \langle y \rangle \\ \langle xy \rangle \end{bmatrix}$$

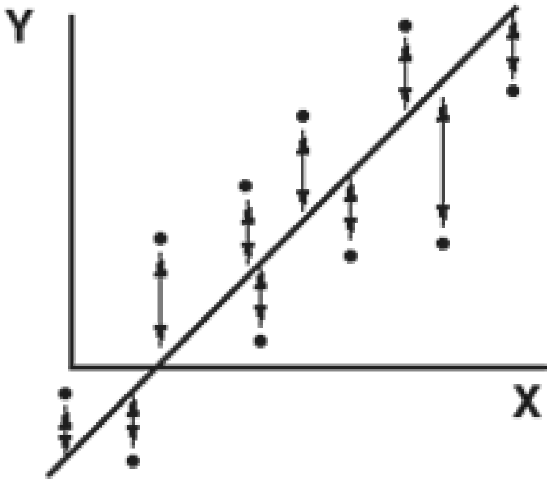
Least squares

- and

$$\Theta^* = \left[\Gamma(x)^T \Gamma(x)\right]^{-1} \Gamma(x)^T y$$

- leads to

$$\Theta^* = \begin{bmatrix} 1 & \langle x \rangle \\ \langle x \rangle & \langle x^2 \rangle \end{bmatrix}^{-1} \begin{bmatrix} \langle y \rangle \\ \langle xy \rangle \end{bmatrix}$$



- which is the solution that we had obtained before, with a lot more work

Least squares

- what about a kth order polynomial model

$$\boxed{f(x;\Theta) = \sum_{i=0}^{K} \theta_i x^i}$$

$$\boxed{\Gamma(x) = \begin{bmatrix} 1 & \dots & x_1^K \\ & \vdots & \\ 1 & \dots & x_n^K \end{bmatrix}}$$

$$\Gamma(x)^T \Gamma(x) = \begin{bmatrix} 1 & \dots & 1 \\ & \vdots & \\ x_1^K & \dots & x_n^K \end{bmatrix} \begin{bmatrix} 1 & \dots & x_1^K \\ & \vdots & \\ 1 & \dots & x_n^K \end{bmatrix}$$

$$= n \begin{bmatrix} 1 & \dots & \langle x^K \rangle \\ & \vdots & \\ \langle x^K \rangle & \dots & \langle x^{2K} \rangle \end{bmatrix}$$

Least squares

- and

$$\Gamma(x)^T y = \begin{bmatrix} 1 & \dots & 1 \\ & \vdots & \\ x_1^K & \dots & x_n^K \end{bmatrix} \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix} = n \begin{bmatrix} \langle y \rangle \\ \vdots \\ \langle x^K y \rangle \end{bmatrix}$$

- combining the two

$$\boxed{\Theta^* = \begin{bmatrix} 1 & \dots & \langle x^K \rangle \\ & \vdots & \\ \langle x^K \rangle & \dots & \langle x^{2K} \rangle \end{bmatrix}^{-1} \begin{bmatrix} \langle y \rangle \\ \vdots \\ \langle x^K y \rangle \end{bmatrix}}$$

- it can't get any easier than this!

Geometric interpretation

- there is also a nice geometric way to derive the least squares solution
- we want to minimize

$$\boxed{L = \left\| y - \Gamma(x)\Theta \right\|^2}$$

- given the known matrix

$$\boxed{\Gamma(x) = \begin{bmatrix} | & & | \\ \Gamma_1 & \dots & \Gamma_K \\ | & & | \end{bmatrix}}$$

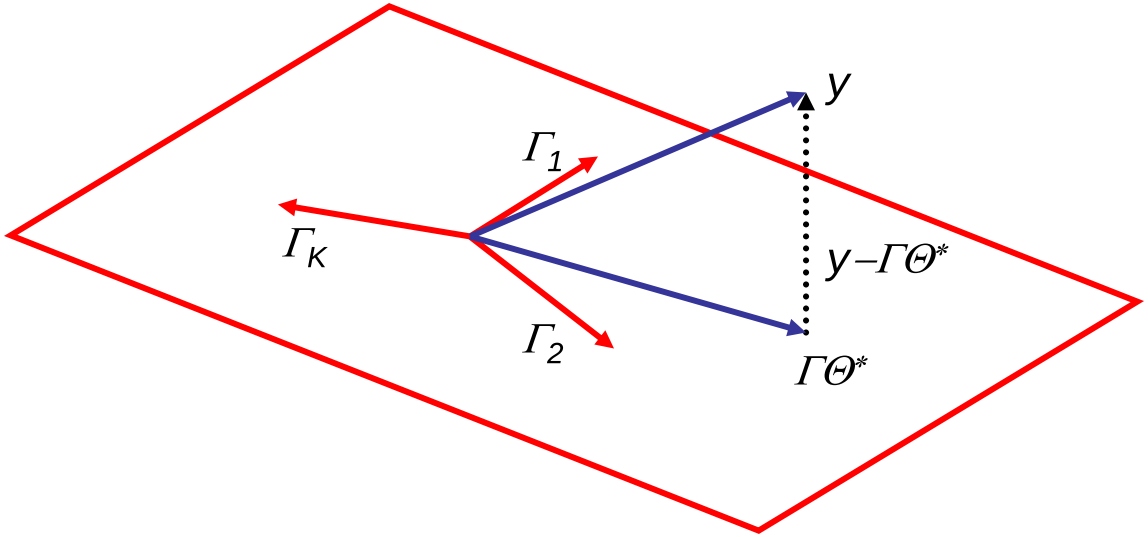
- the vector

$$\boxed{\Gamma(x)\Theta = \begin{bmatrix} | & & | \\ \Gamma_1 & \dots & \Gamma_K \\ | & & | \end{bmatrix} \begin{bmatrix} \theta_1 \\ \vdots \\ \theta_K \end{bmatrix} = \begin{bmatrix} | \\ \sum_i \theta_i \Gamma_i \\ | \end{bmatrix}}$$

is a linear combination of the column vectors Γ_i

Geometric interpretation

- this means that $\Gamma\Theta$ is a vector in the column space of $(\Gamma_1, \ldots, \Gamma_K)$

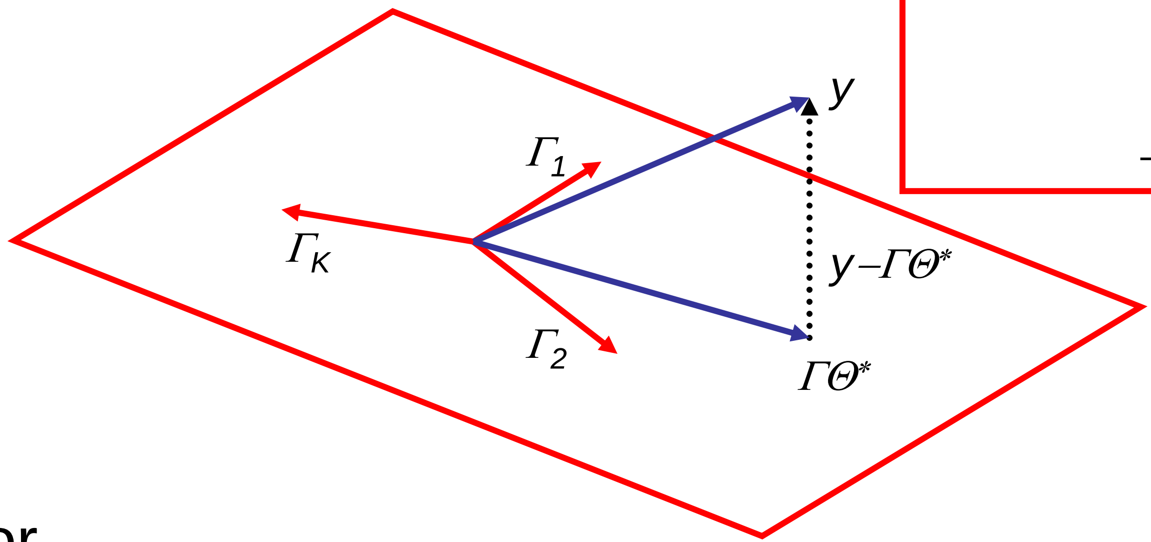


- assume that y is as shown
  - what is the value of $\Gamma\Theta$ closest to y?
  - it has to be the projection of y on the hyper-plane

Geometric interpretation

- let's denote this by $\Gamma\Theta^*$. Then,
  - $y-\Gamma\Theta^*$ is in the null space of Γ, i.e.

$$\left(y-\Gamma\Theta^*\right)\perp\Gamma_1 \\ \perp\Gamma_2 \\ \vdots \\ \perp\Gamma_K$$



- or

$$\begin{cases}\Gamma_1^T\left(y-\Gamma\Theta^*\right)=0\\ \vdots \\ \Gamma_K^T\left(y-\Gamma\Theta^*\right)=0\end{cases} \Leftrightarrow \left\{\begin{bmatrix}- & \Gamma_1^T & -\\ & \vdots & \\ - & \Gamma_K^T & -\end{bmatrix}\left(y-\Gamma\Theta^*\right)=0 \Leftrightarrow \Gamma^T\left(y-\Gamma\Theta^*\right)=0\right.$$

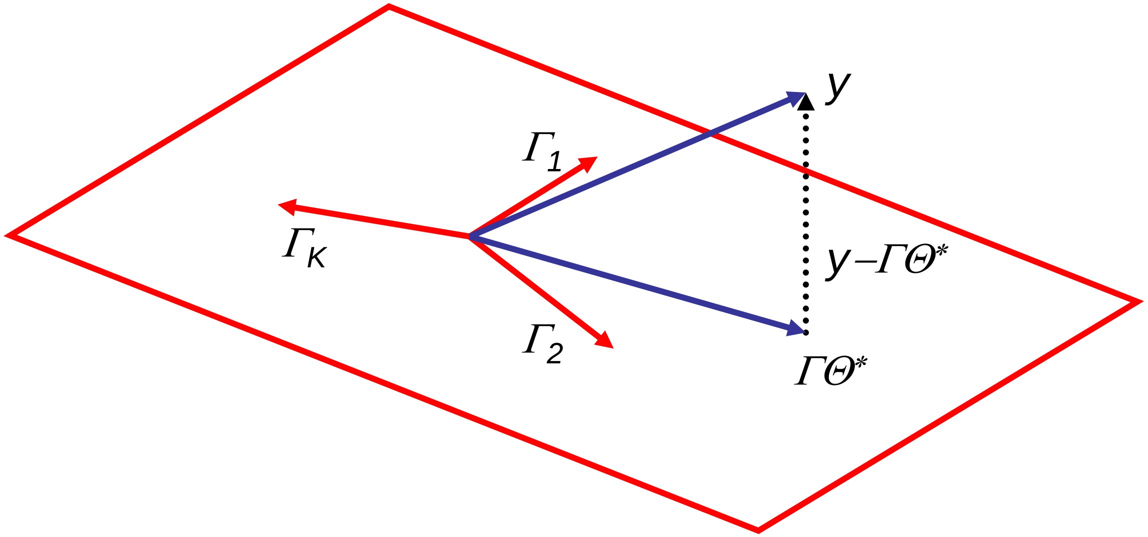
Geometric interpretation

- from which

$$\Gamma^T\left(y - \Gamma\Theta^*\right) = 0 \Leftrightarrow \Gamma^T y = \Gamma^T \Gamma\Theta^*$$

and we get our well known equation

$$\boxed{\Theta^* = \left[\Gamma(x)^T \Gamma(x)\right]^{-1} \Gamma(x)^T y}$$



Any questions?