



THE UNIVERSITY *of* EDINBURGH
informatics

Advanced Robotics

2 – Intro to optimisation - least-squares minimisation

22 Sep 2026

Steve Tonneau - School of Informatics
University of Edinburgh

Course objective (reminder):

Control a robot in an environment such that it accomplishes a motion task

Model of the robot (and the environment)

- Geometry / Dynamics state

Let's start with this

- Constraints (collisions, forces etc)

Mathematical definition of a task as a (differentiable) function

- $f(q) = 0$ means the task is satisfied

Motion generated using an optimal control formulation

Course objective (reminder):

Control a robot in an environment such that it accomplishes a motion task

Model of the robot (and the environment)

- Geometry / Dynamics state

Let's start with this...

... But before that ... Let's talk about optimisation (just a bit)

- Constraints (collisions, forces etc)

Mathematical definition of a task as a (differentiable) function

- $f(q) = 0$ means the task is satisfied

Motion generated using an optimal control formulation

Lecture objective:

Starting from well-known notions from secondary school:

- ❑ Progressively get familiar with the concept of optimisation
- ❑ Brush-off basic Matrix operations

Your objectives for the lecture:

- ❑ The concept of minimising an objective through gradient analysis
- ❑ The notion of constraint (we probably won't have time)

NB: Today's techniques don't work in most cases in robotics (because of non linearities)

This is a new lecture based on last year's observations

Any feedback is welcome. This lecture might not seem like a robotics one but it is.

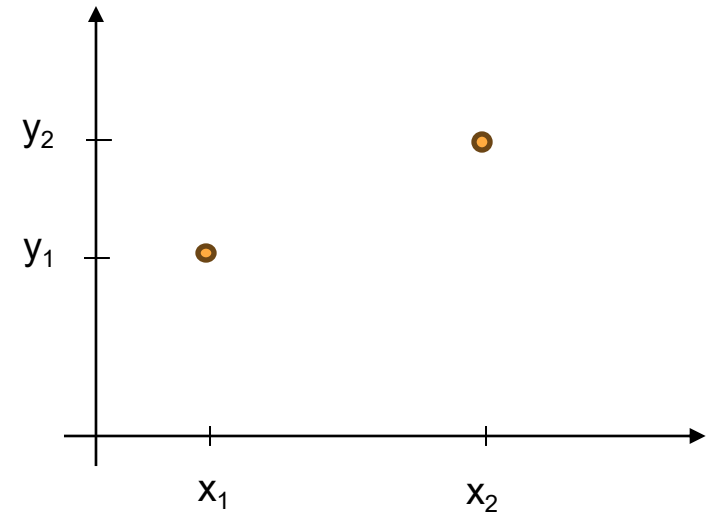
Back to secondary school

Given two samples (x_1, y_1) and (x_2, y_2)
reconstruct a trajectory $y=f(x)$

- ❑ Assuming $f(x)$ is *linear* (follows a line)

Example of application – 1D robot

- ❑ x axis is time
- ❑ y axis is position
- ❑ (x, y) state punctually estimated using on board-sensing => noise



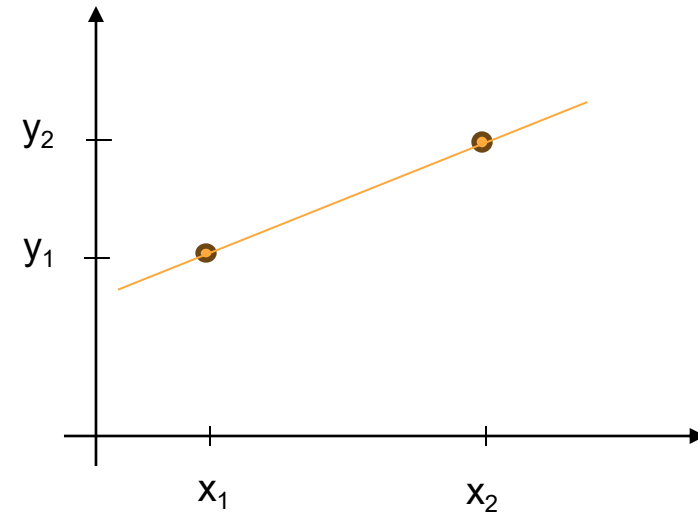
How do we solve this ?



Given two samples (x_1, y_1) and (x_2, y_2)
reconstruct a trajectory $y=f(x)$

□ Assuming $f(x)$ is *linear* (follows a line)

Let's work on the board. Solution on slides afterwards



How do we solve this ?



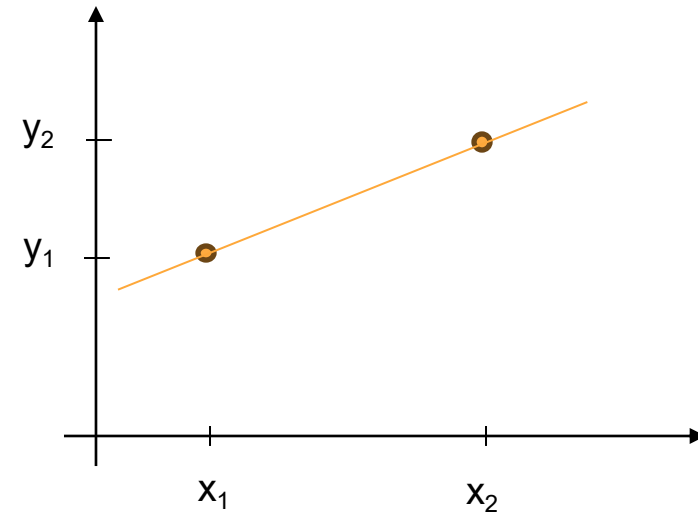
Given two samples (x_1, y_1) and (x_2, y_2)
reconstruct a trajectory $y=f(x)$

□ Assuming $f(x)$ is *linear* (follows a line)

$$\Rightarrow y_1 = w_0 x_1 + w_1$$

$$y_2 = w_0 x_2 + w_1$$

The unknown is $\mathbf{w} = [w_0, w_1] \in \mathbb{R}^2$ vectors in lower case bold



How do we solve this ?



Given two samples (x_1, y_1) and (x_2, y_2)
reconstruct a trajectory $y=f(x)$

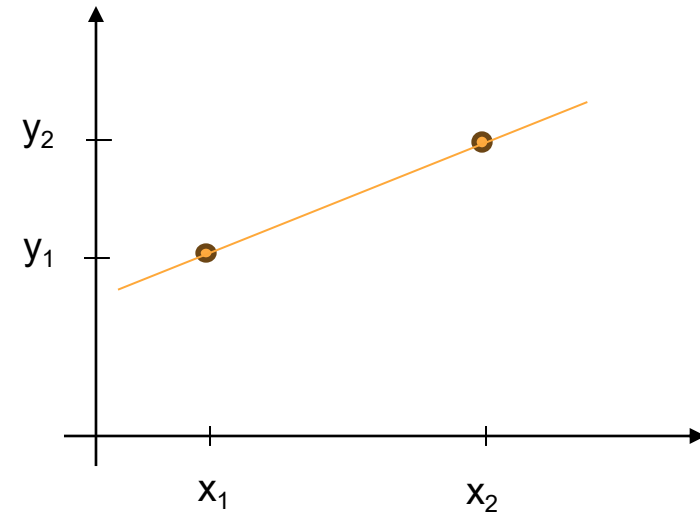
□ Assuming $f(x)$ is *linear* (follows a line)

$$\begin{aligned} \Rightarrow y_1 &= w_0 x_1 + w_1 \\ y_2 &= w_0 x_2 + w_1 \end{aligned}$$

The unknown is $\mathbf{w} = [w_0, w_1] \in \mathbb{R}^2$ vectors in lower case bold

$$y_1 - y_2 = w_0(x_1 - x_2)$$

$$\begin{aligned} w_0 &= \frac{y_1 - y_2}{x_1 - x_2} \\ w_1 &= y_1 - w_0 x_1 \end{aligned}$$



Solving the equations in matrix form

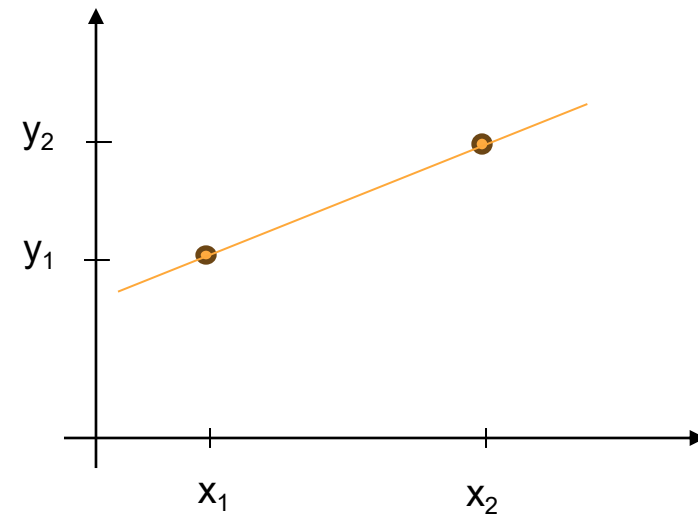


Given two samples (x_1, y_1) and (x_2, y_2)
reconstruct a trajectory $y=f(x)$

□ Assuming $f(x)$ is *linear* (follows a line)

$$\Rightarrow \begin{aligned} y_1 &= w_0 x_1 + w_1 \\ y_2 &= w_0 x_2 + w_1 \end{aligned}$$

The unknown is $\mathbf{w} = [w_0, w_1] \in \mathbb{R}^2$ vectors in lower case bold



Solving the equations in matrix form



Given two samples (x_1, y_1) and (x_2, y_2)
reconstruct a trajectory $y=f(x)$

□ Assuming $f(x)$ is *linear* (follows a line)

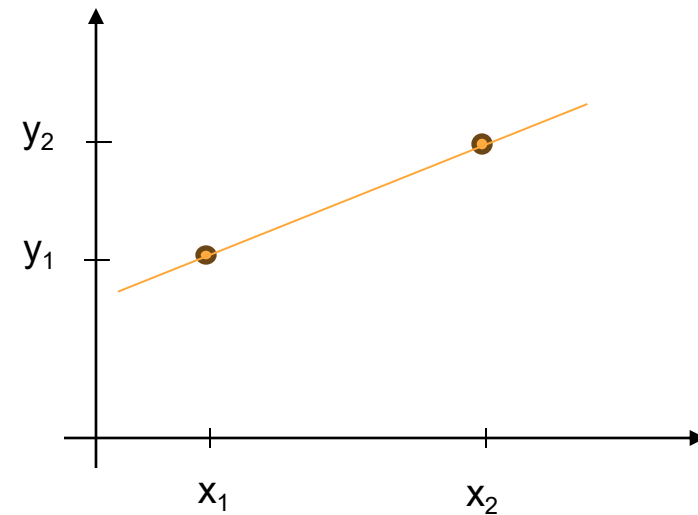
$$\begin{aligned} \Rightarrow y_1 &= w_0 x_1 + w_1 \\ y_2 &= w_0 x_2 + w_1 \end{aligned}$$

The unknown is $\mathbf{w} = [w_0, w_1] \in \mathbb{R}^2$

vectors in lower case bold

$$\underbrace{\begin{bmatrix} y_1 \\ y_2 \end{bmatrix}}_{\mathbf{y}} = \underbrace{\begin{bmatrix} x_1 & 1 \\ x_2 & 1 \end{bmatrix}}_{\mathbf{X}} \underbrace{\begin{bmatrix} w_0 \\ w_1 \end{bmatrix}}_{\mathbf{w}}$$

Matrices in **upper** case bold



Solving the equations in matrix form



Given two samples (x_1, y_1) and (x_2, y_2)
reconstruct a trajectory $y=f(x)$

□ Assuming $f(x)$ is *linear* (follows a line)

$$\begin{aligned} \Rightarrow y_1 &= w_0 x_1 + w_1 \\ y_2 &= w_0 x_2 + w_1 \end{aligned}$$

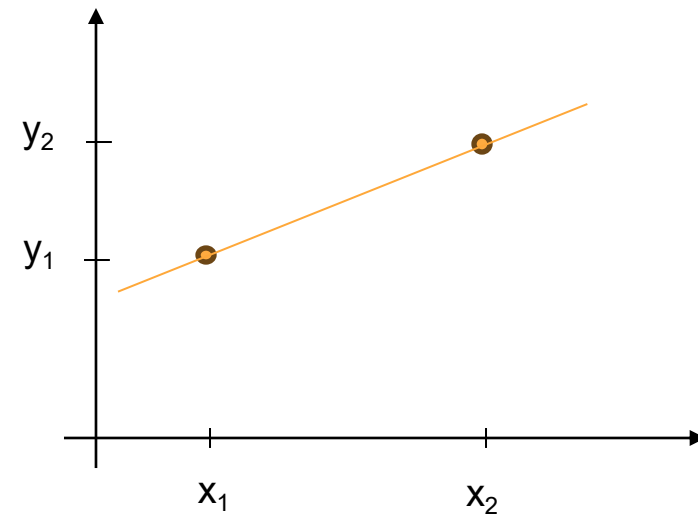
The unknown is $\mathbf{w} = [w_0, w_1] \in \mathbb{R}^2$

vectors in lower case bold

$$\underbrace{\begin{bmatrix} y_1 \\ y_2 \end{bmatrix}}_{\mathbf{y}} = \underbrace{\begin{bmatrix} x_1 & 1 \\ x_2 & 1 \end{bmatrix}}_{\mathbf{X}} \underbrace{\begin{bmatrix} w_0 \\ w_1 \end{bmatrix}}_{\mathbf{w}}$$

Matrices in **upper** case bold

$$\mathbf{w} = \mathbf{X}^{-1} \mathbf{y}$$



Solving the equations in matrix form



Given two samples (x_1, y_1) and (x_2, y_2)
reconstruct a trajectory $y=f(x)$

□ Assuming $f(x)$ is *linear* (follows a line)

$$\Rightarrow \begin{aligned} y_1 &= w_0 x_1 + w_1 \\ y_2 &= w_0 x_2 + w_1 \end{aligned}$$

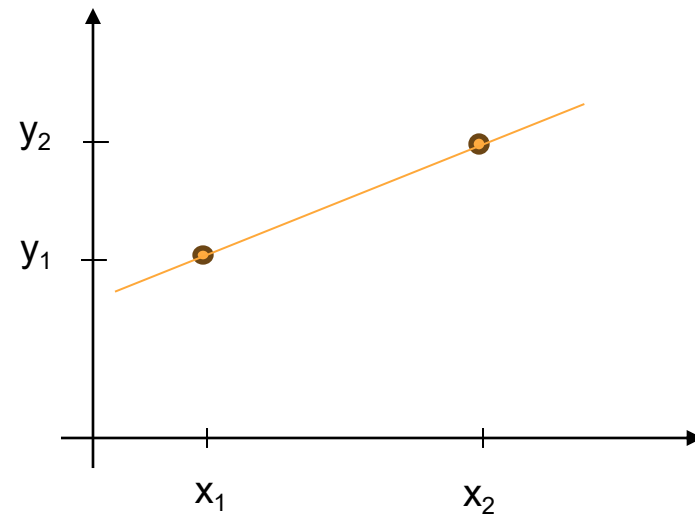
The unknown is $\mathbf{w} = [w_0, w_1] \in \mathbb{R}^2$

vectors in lower case bold

$$\underbrace{\begin{bmatrix} y_1 \\ y_2 \end{bmatrix}}_{\mathbf{y}} = \underbrace{\begin{bmatrix} x_1 & 1 \\ x_2 & 1 \end{bmatrix}}_{\mathbf{X}} \underbrace{\begin{bmatrix} w_0 \\ w_1 \end{bmatrix}}_{\mathbf{w}}$$

Matrices in **upper** case bold

$$\mathbf{w} = \mathbf{X}^{-1} \mathbf{y}$$



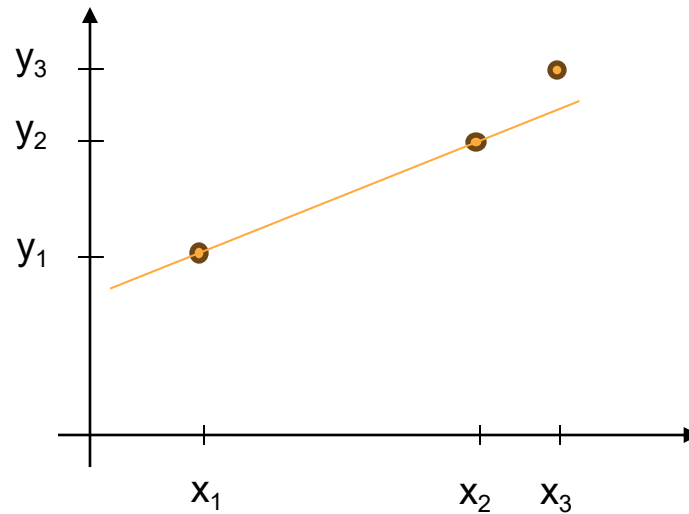
What if $\mathbf{X}$ is not invertible?

Exercise: calculate the inverse of $\mathbf{X}$ and check that you find the desired solution

What if we consider $n > 2$ samples?



❑ Noisy sensors / actuators $\Rightarrow$ not all points on a line



Optimising an objective

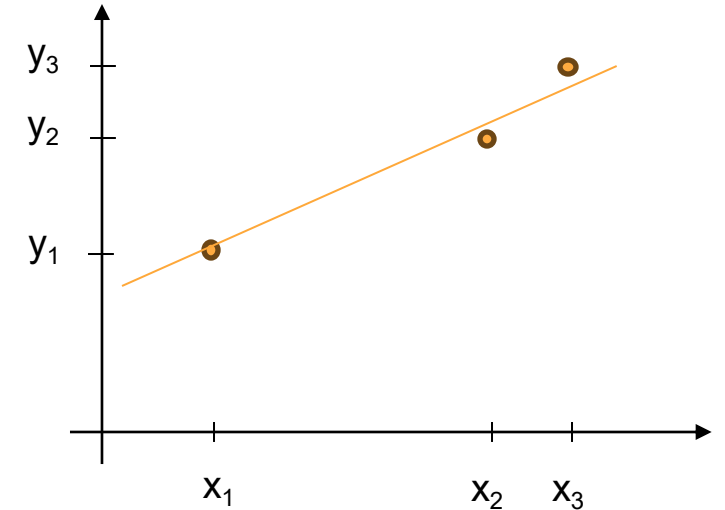


- ❑ Try to approximate “as best as possible”:

Minimise cost / error OR maximise a reward (same thing)

- ❑ What objective?

- ❑ If perfect match exists, we want this
- ❑ On average all points are “close enough”



Optimising an objective

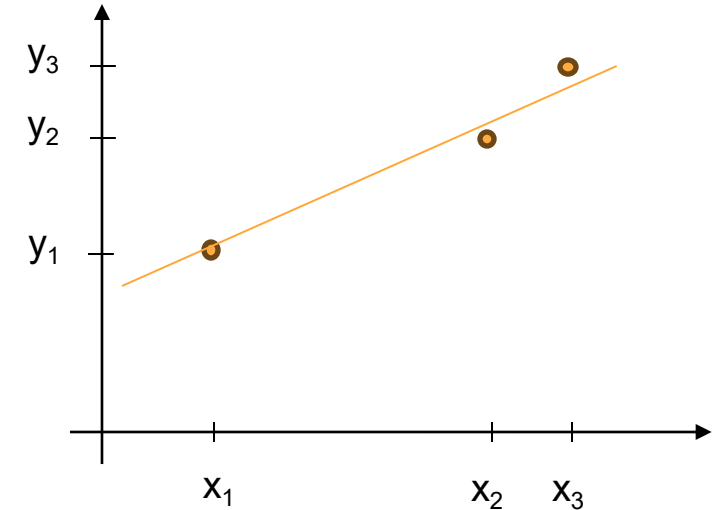


- ❑ Try to approximate “as best as possible”:

Minimise cost / error OR maximise a reward (same thing)

- ❑ What objective?

- ❑ If perfect match exists, we want this
- ❑ On average all points are “close enough”



- ❑ Minimise the residual error between sample and line prediction:

$$r_i = y_i - (w_0x_i + w_1), \forall i = \{1, \dots, n\}$$

Square it to deal with negative values:

$$l(\mathbf{w}) = \sum_{i=1}^n r_i^2$$

Does this satisfy our objectives?

How to minimise $l(\mathbf{w})$?

□ Minimise $l(\mathbf{w}) = \sum_{i=1}^n r_i^2$ where $r_i = y_i - (w_0 x_i + w_1), \forall i = \{1, \dots, n\}$

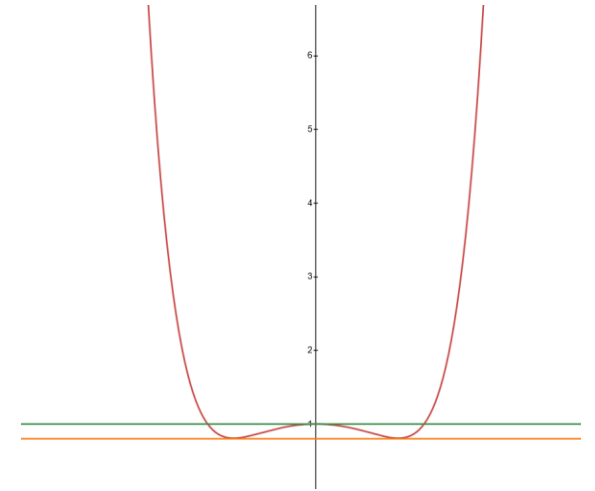
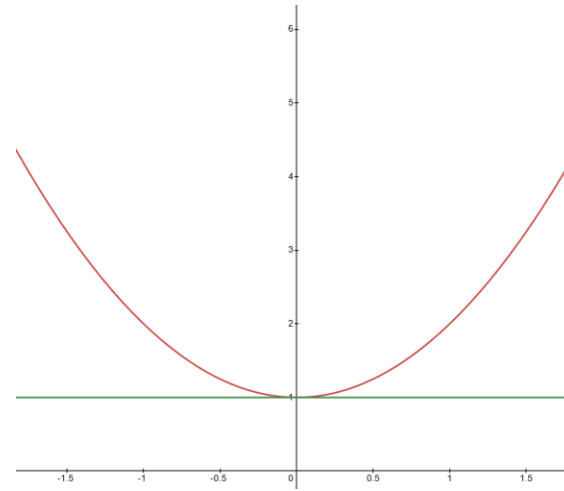
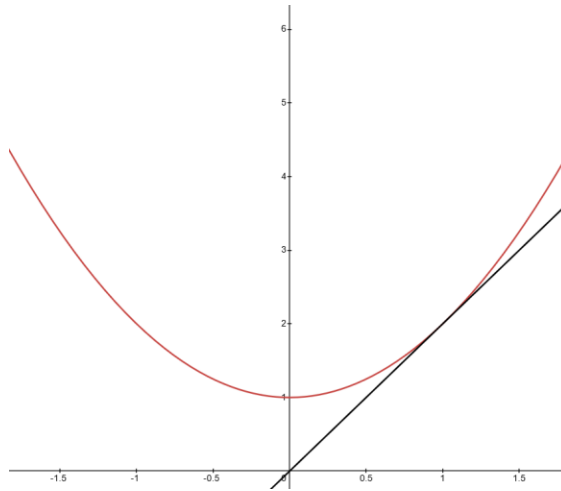
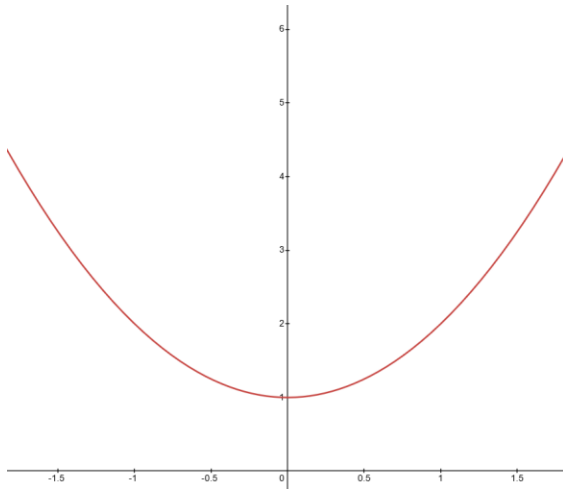
□ Matrix form is

$$\underbrace{\begin{bmatrix} r_1 \\ \dots \\ r_n \end{bmatrix}}_{\mathbf{r}} = \underbrace{\begin{bmatrix} y_1 \\ \dots \\ y_n \end{bmatrix}}_{\mathbf{y}} - \underbrace{\begin{bmatrix} x_1 & 1 \\ \dots & \dots \\ x_n & 1 \end{bmatrix}}_{\mathbf{X}} \underbrace{\begin{bmatrix} w_0 \\ w_1 \end{bmatrix}}_{\mathbf{w}}$$

□ We thus want to find the minimum of $l(\mathbf{w}) = \mathbf{r}^T \mathbf{r} = (\mathbf{y} - \mathbf{X}\mathbf{w})^T (\mathbf{y} - \mathbf{X}\mathbf{w})$

How to minimise $l(\mathbf{w})$?

□ Necessary (not sufficient) condition for a minimum: gradient is **0** (stationary point)



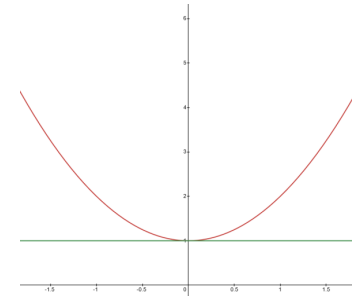
To minimise $l(\mathbf{w})$ we compute the gradient

$$\nabla_{\mathbf{w}} l(\mathbf{w}) = \frac{d}{d\mathbf{w}} (\mathbf{r}^T \mathbf{r}) = \left[\frac{\partial l}{\partial w_0}, \frac{\partial l}{\partial w_1} \right] \in \mathbb{R}^2$$

To minimise $l(\mathbf{w})$ we compute the gradient

$$\nabla_{\mathbf{w}} l(\mathbf{w}) = \frac{d}{d\mathbf{w}} (\mathbf{r}^T \mathbf{r}) = \left[\frac{\partial l}{\partial w_0}, \frac{\partial l}{\partial w_1} \right] \in \mathbb{R}^2$$

Objective: set this gradient to 0



To minimise $l(\mathbf{w})$ we compute the gradient

$$\nabla_{\mathbf{w}} l(\mathbf{w}) = \frac{d}{d\mathbf{w}} (\mathbf{r}^T \mathbf{r}) = \left[\frac{\partial l}{\partial w_0}, \frac{\partial l}{\partial w_1} \right] \in \mathbb{R}^2$$

$$\frac{d}{d\mathbf{w}} (\mathbf{r}^T \mathbf{r}) = \frac{d}{d\mathbf{w}} \left(\sum_i r_i^2 \right)$$

To minimise $l(\mathbf{w})$ we compute the gradient

$$\nabla_{\mathbf{w}} l(\mathbf{w}) = \frac{d}{d\mathbf{w}} (\mathbf{r}^T \mathbf{r}) = \left[\frac{\partial l}{\partial w_0}, \frac{\partial l}{\partial w_1} \right] \in \mathbb{R}^2$$

$$\frac{d}{d\mathbf{w}} (\mathbf{r}^T \mathbf{r}) = \frac{d}{d\mathbf{w}} \left(\sum_i r_i^2 \right)$$

Chain rule:

$$\frac{d}{dx} f(g(x)) = f'(g(x)) \cdot g'(x)$$

To minimise $l(\mathbf{w})$ we compute the gradient

$$\nabla_{\mathbf{w}} l(\mathbf{w}) = \frac{d}{d\mathbf{w}} (\mathbf{r}^T \mathbf{r}) = \left[\frac{\partial l}{\partial w_0}, \frac{\partial l}{\partial w_1} \right] \in \mathbb{R}^2$$

$$\frac{d}{d\mathbf{w}} (\mathbf{r}^T \mathbf{r}) = \frac{d}{d\mathbf{w}} \left(\sum_i r_i^2 \right)$$

$$\frac{d}{d\mathbf{w}} (r_i^2) = 2r_i \frac{dr_i}{d\mathbf{w}}$$

Chain rule:

$$\frac{d}{dx} f(g(x)) = f'(g(x)) \cdot g'(x)$$

To minimise $l(\mathbf{w})$ we compute the gradient

$$\nabla_{\mathbf{w}} l(\mathbf{w}) = \frac{d}{d\mathbf{w}} (\mathbf{r}^T \mathbf{r}) = \left[\frac{\partial l}{\partial w_0}, \frac{\partial l}{\partial w_1} \right] \in \mathbb{R}^2$$

$$\frac{d}{d\mathbf{w}} (\mathbf{r}^T \mathbf{r}) = \frac{d}{d\mathbf{w}} \left(\sum_i r_i^2 \right)$$

$$\frac{d}{d\mathbf{w}} (r_i^2) = 2r_i \frac{dr_i}{d\mathbf{w}}$$

$$\frac{d}{d\mathbf{w}} (\mathbf{r}^T \mathbf{r}) = 2\mathbf{r}^T \frac{d\mathbf{r}}{d\mathbf{w}}$$

Chain rule:

$$\frac{d}{dx} f(g(x)) = f'(g(x)) \cdot g'(x)$$

To minimise $l(\mathbf{w})$ we compute the gradient

$$\frac{d}{d\mathbf{w}}(\mathbf{r}^T \mathbf{r}) = 2\mathbf{r}^T \frac{d\mathbf{r}}{d\mathbf{w}}$$

To minimise $l(\mathbf{w})$ we compute the gradient

$$\frac{d}{d\mathbf{w}}(\mathbf{r}^T \mathbf{r}) = 2\mathbf{r}^T \frac{d\mathbf{r}}{d\mathbf{w}}$$

$$\begin{aligned}\frac{d\mathbf{r}}{d\mathbf{w}} &= \frac{d}{d\mathbf{w}}(\mathbf{y} - \mathbf{X}\mathbf{w}) \\ &= \frac{d}{d\mathbf{w}}(-\mathbf{X}\mathbf{w})\end{aligned}$$

$$\frac{d\mathbf{r}}{d\mathbf{w}} = -\mathbf{X}$$

To minimise $l(\mathbf{w})$ we compute the gradient

$$\begin{aligned}\frac{d\mathbf{r}}{d\mathbf{w}} &= \frac{d}{d\mathbf{w}}(\mathbf{y} - \mathbf{X}\mathbf{w}) \\ &= \frac{d}{d\mathbf{w}}(-\mathbf{X}\mathbf{w})\end{aligned}$$

$$\frac{d\mathbf{r}}{d\mathbf{w}} = -\mathbf{X}$$

$$\frac{d}{d\mathbf{w}}(\mathbf{r}^T \mathbf{r}) = 2\mathbf{r}^T \frac{d\mathbf{r}}{d\mathbf{w}}$$

$$\begin{aligned}\frac{\partial}{\partial \mathbf{w}}(\mathbf{r}^T \mathbf{r}) &= 2\mathbf{r}^T(-\mathbf{X}) \\ &= 2(\mathbf{y} - \mathbf{X}\mathbf{w})^T(-\mathbf{X})\end{aligned}$$

Optimum at gradient = 0

$$2(\mathbf{y} - \mathbf{X}\mathbf{w})^T(-\mathbf{X}) = 0$$

$$(\mathbf{y} - \mathbf{X}\mathbf{w})^T(-\mathbf{X}) = 0$$

Optimum at gradient = 0

$$2(\mathbf{y} - \mathbf{X}\mathbf{w})^T(-\mathbf{X}) = 0$$

$$(\mathbf{y} - \mathbf{X}\mathbf{w})^T(-\mathbf{X}) = 0$$

$$((\mathbf{y} - \mathbf{X}\mathbf{w})^T(-\mathbf{X}))^T = 0$$

Transpose of a scalar is equal to the scalar

Optimum at gradient = 0

$$2(\mathbf{y} - \mathbf{X}\mathbf{w})^T(-\mathbf{X}) = 0$$

$$(\mathbf{y} - \mathbf{X}\mathbf{w})^T(-\mathbf{X}) = 0$$

$$((\mathbf{y} - \mathbf{X}\mathbf{w})^T(-\mathbf{X}))^T = 0$$

$$(-\mathbf{X})^T(\mathbf{y} - \mathbf{X}\mathbf{w}) = 0$$

Transpose of a scalar is equal to the scalar

$$(\mathbf{AB})^T = \mathbf{B}^T \mathbf{A}^T, (\mathbf{A}^T)^T = \mathbf{A}$$

Optimum at gradient = 0

$$2(\mathbf{y} - \mathbf{X}\mathbf{w})^T(-\mathbf{X}) = 0$$

$$(\mathbf{y} - \mathbf{X}\mathbf{w})^T(-\mathbf{X}) = 0$$

$$((\mathbf{y} - \mathbf{X}\mathbf{w})^T(-\mathbf{X}))^T = 0$$

$$(-\mathbf{X})^T(\mathbf{y} - \mathbf{X}\mathbf{w}) = 0$$

$$\mathbf{X}^T\mathbf{X}\mathbf{w} = \mathbf{X}^T\mathbf{y}$$

$$\mathbf{w} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}$$

Transpose of a scalar is equal to the scalar

$$(\mathbf{AB})^T = \mathbf{B}^T\mathbf{A}^T, (\mathbf{A}^T)^T = \mathbf{A}$$

Optimum at gradient = 0

$$2(\mathbf{y} - \mathbf{X}\mathbf{w})^T(-\mathbf{X}) = 0$$

$$(\mathbf{y} - \mathbf{X}\mathbf{w})^T(-\mathbf{X}) = 0$$

$$((\mathbf{y} - \mathbf{X}\mathbf{w})^T(-\mathbf{X}))^T = 0$$

$$(-\mathbf{X})^T(\mathbf{y} - \mathbf{X}\mathbf{w}) = 0$$

$$\mathbf{X}^T\mathbf{X}\mathbf{w} = \mathbf{X}^T\mathbf{y}$$

$$\mathbf{w} = \underbrace{(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T}_{\text{pseudo-inverse of X}}\mathbf{y}$$

Transpose of a scalar is equal to the scalar

$$(\mathbf{AB})^T = \mathbf{B}^T\mathbf{A}^T, (\mathbf{A}^T)^T = \mathbf{A}$$

pseudo-inverse of X

Optimum at gradient = 0

$$2(\mathbf{y} - \mathbf{X}\mathbf{w})^T(-\mathbf{X}) = 0$$

$$(\mathbf{y} - \mathbf{X}\mathbf{w})^T(-\mathbf{X}) = 0$$

$$((\mathbf{y} - \mathbf{X}\mathbf{w})^T(-\mathbf{X}))^T = 0$$

$$(-\mathbf{X})^T(\mathbf{y} - \mathbf{X}\mathbf{w}) = 0$$

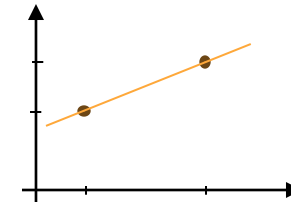
$$\mathbf{X}^T\mathbf{X}\mathbf{w} = \mathbf{X}^T\mathbf{y}$$

$$\mathbf{w} = \underbrace{(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T}_{\text{pseudo-inverse of X}}\mathbf{y}$$

$$\mathbf{w} = \mathbf{X}^{-1}\mathbf{y}$$

Transpose of a scalar is equal to the scalar

$$(\mathbf{AB})^T = \mathbf{B}^T\mathbf{A}^T, (\mathbf{A}^T)^T = \mathbf{A}$$



Optimum at gradient = 0

$$2(\mathbf{y} - \mathbf{X}\mathbf{w})^T(-\mathbf{X}) = 0$$

$$(\mathbf{y} - \mathbf{X}\mathbf{w})^T(-\mathbf{X}) = 0$$

$$((\mathbf{y} - \mathbf{X}\mathbf{w})^T(-\mathbf{X}))^T = 0$$

$$(-\mathbf{X})^T(\mathbf{y} - \mathbf{X}\mathbf{w}) = 0$$

$$\mathbf{X}^T\mathbf{X}\mathbf{w} = \mathbf{X}^T\mathbf{y}$$

$$\mathbf{w} = \underbrace{(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T}_{\text{pseudo-inverse of X}}\mathbf{y}$$

pseudo-inverse of X

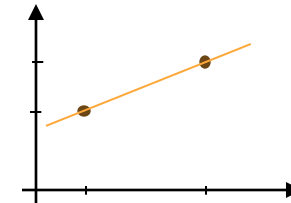
Transpose of a scalar is equal to the scalar

$$(\mathbf{AB})^T = \mathbf{B}^T\mathbf{A}^T, (\mathbf{A}^T)^T = \mathbf{A}$$

Exact vs approximate solution depends
on whether X is invertible!

Although pseudo-inverse not always defined (underconstrained)

$$\mathbf{w} = \mathbf{X}^{-1}\mathbf{y}$$



In conclusion

- ❑ Optimisation is essentially working with the gradients of a function
 - ❑ Setting it to 0 does not guarantee global optimum (except in some cases)
 - ❑ We need to be able to invert matrices / approximate something close enough
- ❑ Least squares is a widely used technique
 - ❑ Constraints require extra work => Can we set constraints into the cost ?
 - ❑ Inversion is really a problem (numerical instability)
- ❑ Exercice. What is $y=f(y)$ is a polynomial of degree 3 (or higher) ? Would unconstrained least square still work?

Homework for next week

- ❑ Self run the python tutorial if you need
- ❑ Make sure your environment is setup on DICE and run tutorial 0
- ❑ Ask questions on ~~Piazza~~ EdStem if you do not understand something