# Machine Learning with Python

## MTH786U/P 2022/23

## Lecture 5: From ridge regression to the LASSO

**Nicola Perra, Queen Mary University of London (QMUL)**

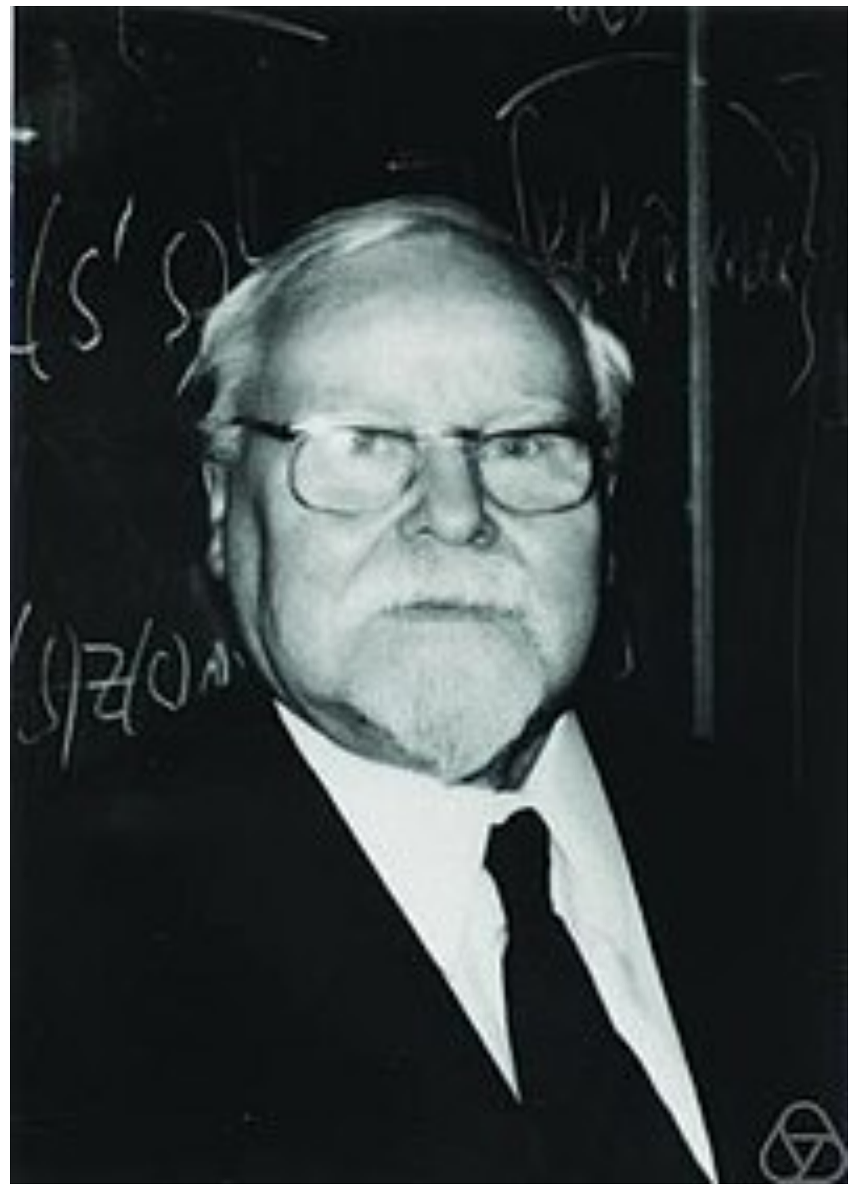
n.perra@qmul.ac.uk

# Recap: Ridge regression

Two weeks ago we learned about the minimisation problem

$$\hat{\mathbf{w}} = \arg\min_{\mathbf{w}} \left\{ \frac{1}{2}\|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2 + \frac{\alpha}{2}\|\mathbf{w}\|^2 \right\}$$

that is known as *Tikhonov regularisation* or *ridge regression*



Andrey Tikhonov, 1906 - 1993

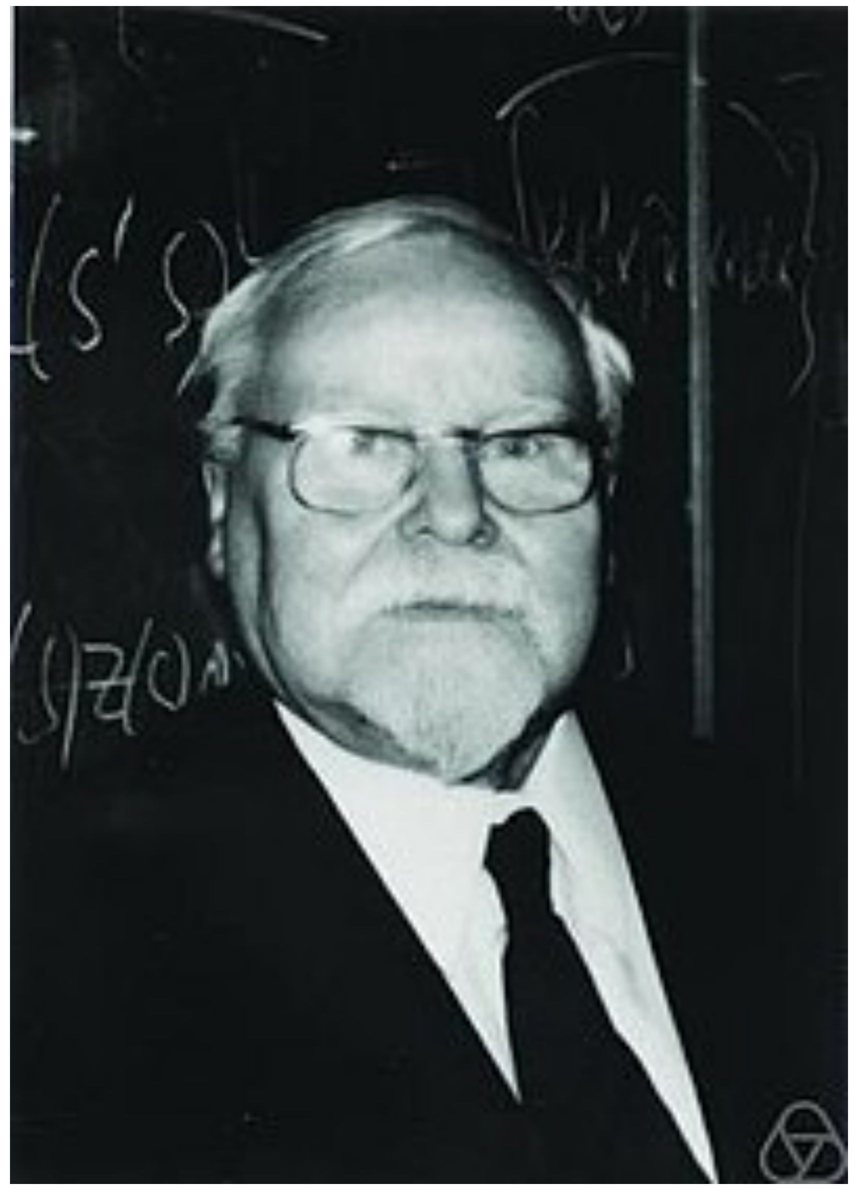
# Recap: Ridge regression

Two weeks ago we learned about the minimisation problem

$$\hat{\mathbf{w}} = \arg\min_{\mathbf{w}} \left\{ \frac{1}{2}\|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2 + \frac{\alpha}{2}\|\mathbf{w}\|^2 \right\}$$

Standard regression term

that is known as *Tikhonov regularisation* or *ridge regression*



Andrey Tikhonov, 1906 - 1993

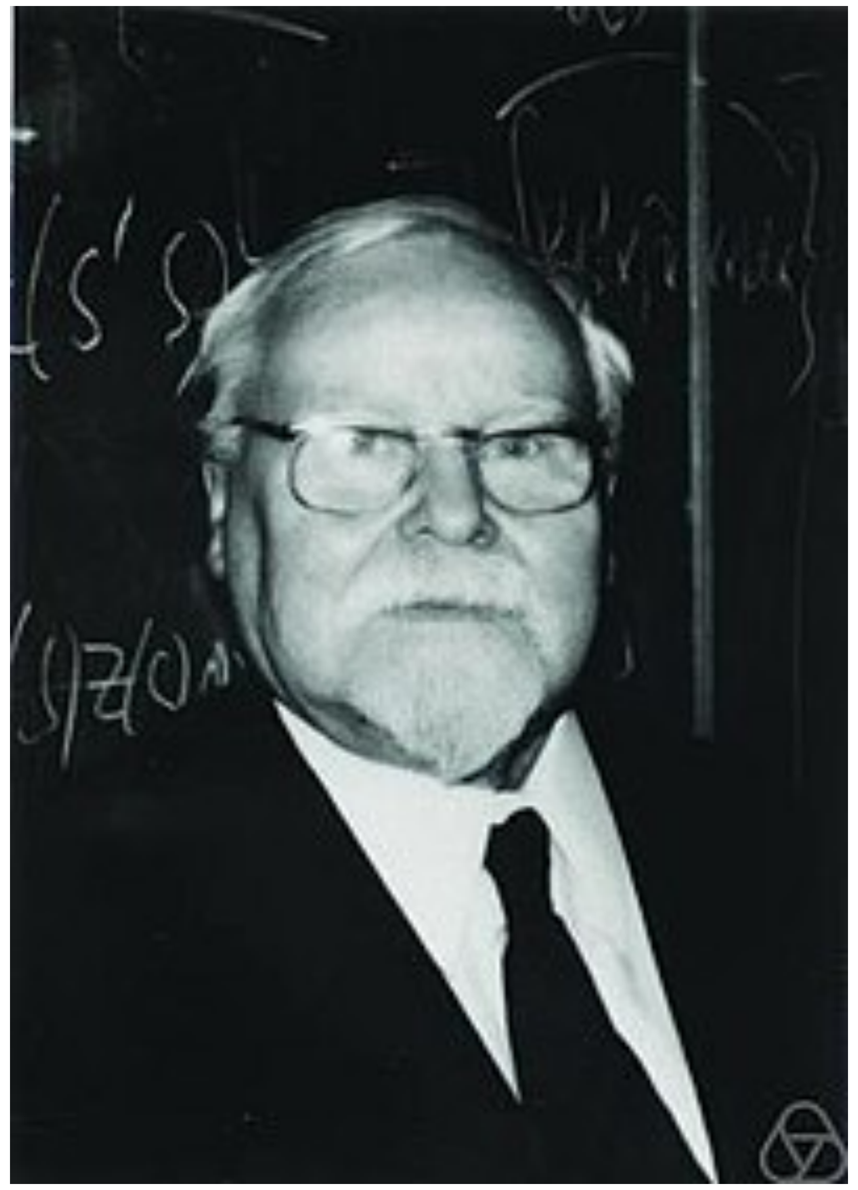
# Recap: Ridge regression

Two weeks ago we learned about the minimisation problem

$$\hat{\mathbf{w}} = \arg\min_{\mathbf{w}} \left\{ \frac{1}{2}\|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2 + \frac{\alpha}{2}\|\mathbf{w}\|^2 \right\}$$

Standard regression term    Regularisation term

that is known as *Tikhonov regularisation* or *ridge regression*



Andrey Tikhonov, 1906 - 1993

# Recap: Ridge regression

Two weeks ago we learned about the minimisation problem

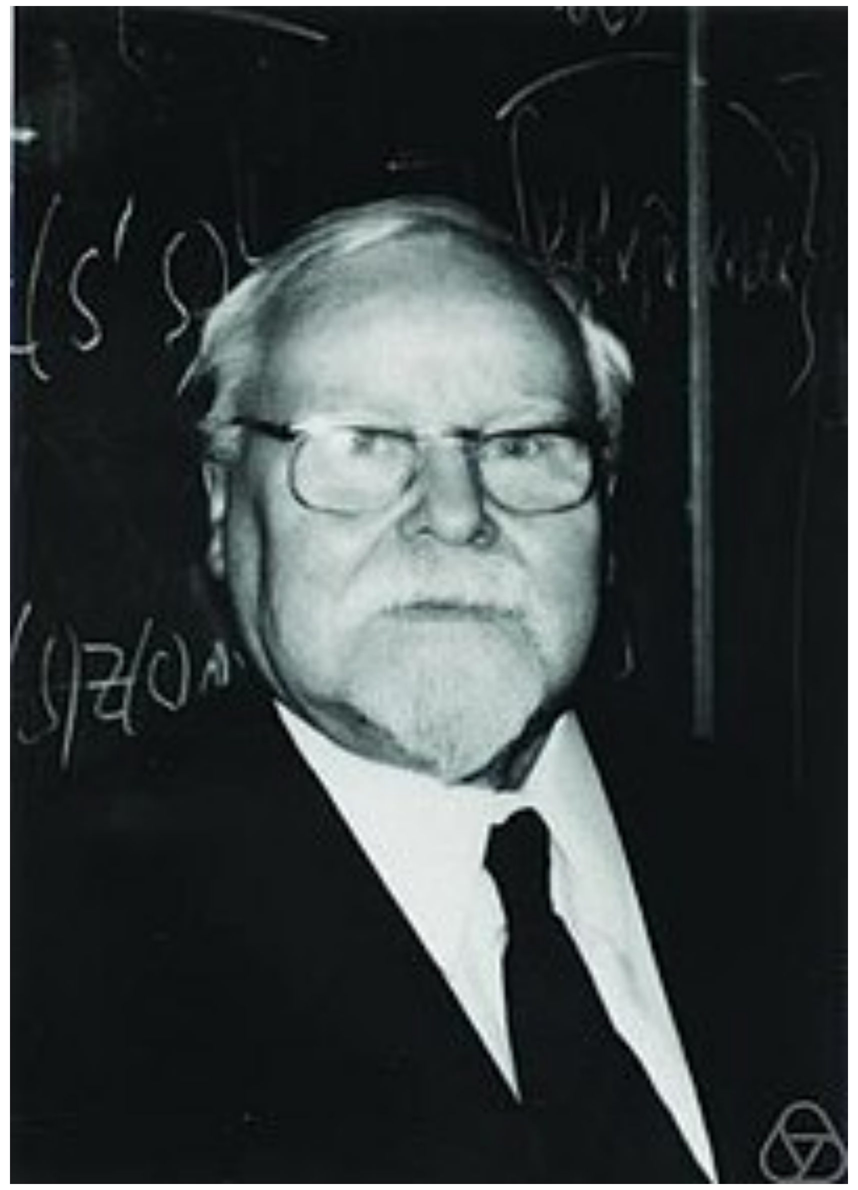
$$\hat{\mathbf{w}} = \arg\min_{\mathbf{w}} \left\{ \frac{1}{2}\|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2 + \frac{\alpha}{2}\|\mathbf{w}\|^2 \right\}$$

Standard regression term

Regularisation term

Regularisation parameter

that is known as *Tikhonov regularisation* or *ridge regression*



Andrey Tikhonov, 1906 - 1993

# Variational regularisation

A more general form of the previous problem is variational regularisation

$$\hat{\mathbf{w}} = \arg\min_{\mathbf{w}} \{L(\mathbf{w}) + R(\mathbf{w})\}$$

# Variational regularisation

A more general form of the previous problem is variational regularisation

$$\hat{\mathbf{w}} = \arg\min_{\mathbf{w}} \{L(\mathbf{w}) + R(\mathbf{w})\}$$

Data term/
Regression term

# Variational regularisation

A more general form of the previous problem is variational regularisation

$$\hat{\mathbf{w}} = \arg\min_{\mathbf{w}} \{L(\mathbf{w}) + R(\mathbf{w})\}$$

Data term/ Regression term

Regularisation term

# Variational regularisation

A more general form of the previous problem is variational regularisation

$$\hat{\mathbf{w}} = \arg\min_{\mathbf{w}} \{L(\mathbf{w}) + R(\mathbf{w})\}$$

Data term/
Regression term

Regularisation
term

Previous example: $L(\mathbf{w}) = \frac{1}{2}\|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2$ $R(\mathbf{w}) = \frac{\alpha}{2}\|\mathbf{w}\|^2$

# ℓ1 regularisation / the lasso

Variational regularisation: $\hat{\mathbf{w}} = \arg\min_{\mathbf{w}} \{L(\mathbf{w}) + R(\mathbf{w})\}$

# $\ell$1 regularisation / the lasso

Variational regularisation: $\hat{\mathbf{w}} = \arg\min_{\mathbf{w}} \{L(\mathbf{w}) + R(\mathbf{w})\}$

Choose $R(\mathbf{w}) = \alpha \|\mathbf{w}\|_1 := \alpha \sum_{k=1}^{n} |w_k|$ and $L(\mathbf{w}) = \frac{1}{2}\|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2$

# $\ell 1$ regularisation / the lasso

Variational regularisation: $\hat{\mathbf{w}} = \arg\min_{\mathbf{w}} \{L(\mathbf{w}) + R(\mathbf{w})\}$

Choose $R(\mathbf{w}) = \alpha \|\mathbf{w}\|_1 := \alpha \sum_{k=1}^{n} |w_k|$ and $L(\mathbf{w}) = \frac{1}{2}\|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2$

$$\Rightarrow \qquad \hat{\mathbf{w}} = \arg\min_{\mathbf{w}} \left\{ \frac{1}{2}\|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2 + \alpha \|\mathbf{w}\|_1 \right\}$$

# ℓ1 regularisation / the lasso

Variational regularisation: $\hat{\mathbf{w}} = \arg\min_{\mathbf{w}} \{L(\mathbf{w}) + R(\mathbf{w})\}$

Choose $R(\mathbf{w}) = \alpha\|\mathbf{w}\|_1 := \alpha \sum_{k=1}^{n} |w_k|$ and $L(\mathbf{w}) = \frac{1}{2}\|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2$

$$\Rightarrow \qquad \hat{\mathbf{w}} = \arg\min_{\mathbf{w}} \left\{ \frac{1}{2}\|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2 + \alpha\|\mathbf{w}\|_1 \right\}$$

What is the advantage of using the one-norm over the two-norm?

# $\ell$1 regularisation / the lasso

Variational regularisation: $\hat{\mathbf{w}} = \arg\min_{\mathbf{w}} \{L(\mathbf{w}) + R(\mathbf{w})\}$

Choose $R(\mathbf{w}) = \alpha \|\mathbf{w}\|_1 := \alpha \sum_{k=1}^{n} |w_k|$ and $L(\mathbf{w}) = \frac{1}{2}\|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2$

$$\Rightarrow \qquad \hat{\mathbf{w}} = \arg\min_{\mathbf{w}} \left\{ \frac{1}{2}\|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2 + \alpha \|\mathbf{w}\|_1 \right\}$$

What is the advantage of using the one-norm over the two-norm?

Sparsity!

# ℓ1 regularisation / the lasso

$$\hat{\mathbf{w}} = \arg\min_{\mathbf{w}} \left\{ \frac{1}{2} \|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2 + \alpha \|\mathbf{w}\|_1 \right\}$$

Sparsity means that only relatively few elements of $\hat{\mathbf{w}}$ will be non-zero

# ℓ1 regularisation / the lasso

$$\hat{\mathbf{w}} = \arg\min_{\mathbf{w}} \left\{ \frac{1}{2} \|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2 + \alpha \|\mathbf{w}\|_1 \right\}$$

Sparsity means that only relatively few elements of $\hat{\mathbf{w}}$ will be non-zero

Sparsity ≅ simplicity! (Occam's razor)

# $\ell$1 regularisation / the lasso

$$\hat{\mathbf{w}} = \arg\min_{\mathbf{w}} \left\{ \frac{1}{2} ||\mathbf{X}\mathbf{w} - \mathbf{y}||^2 + \alpha ||\mathbf{w}||_1 \right\}$$

Sparsity means that only relatively few elements of $\hat{\mathbf{w}}$ will be non-zero

Sparsity ≅ simplicity! (Occam's razor)

Implicit reduction of parameters

# ℓ1 regularisation / the lasso

$$\hat{\mathbf{w}} = \arg\min_{\mathbf{w}} \left\{ \frac{1}{2} \|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2 + \alpha \|\mathbf{w}\|_1 \right\}$$

Sparsity means that only relatively few elements of $\hat{\mathbf{w}}$ will be non-zero

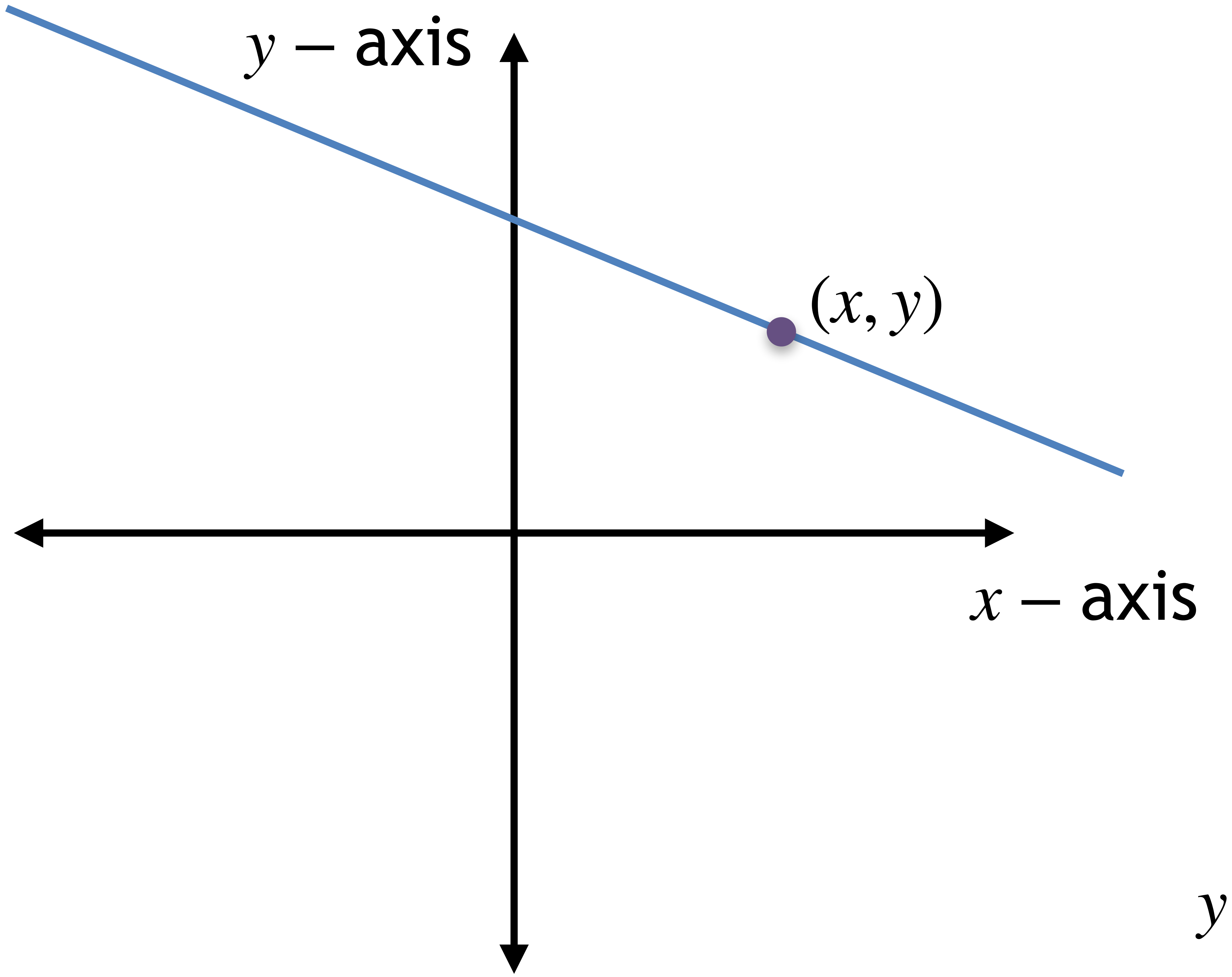
Sparsity ≅ simplicity! (Occam's razor)

Implicit reduction of parameters

LASSO = Least Absolute Shrinkage and Selection Operator

# $\ell 1$ regularisation / the lasso

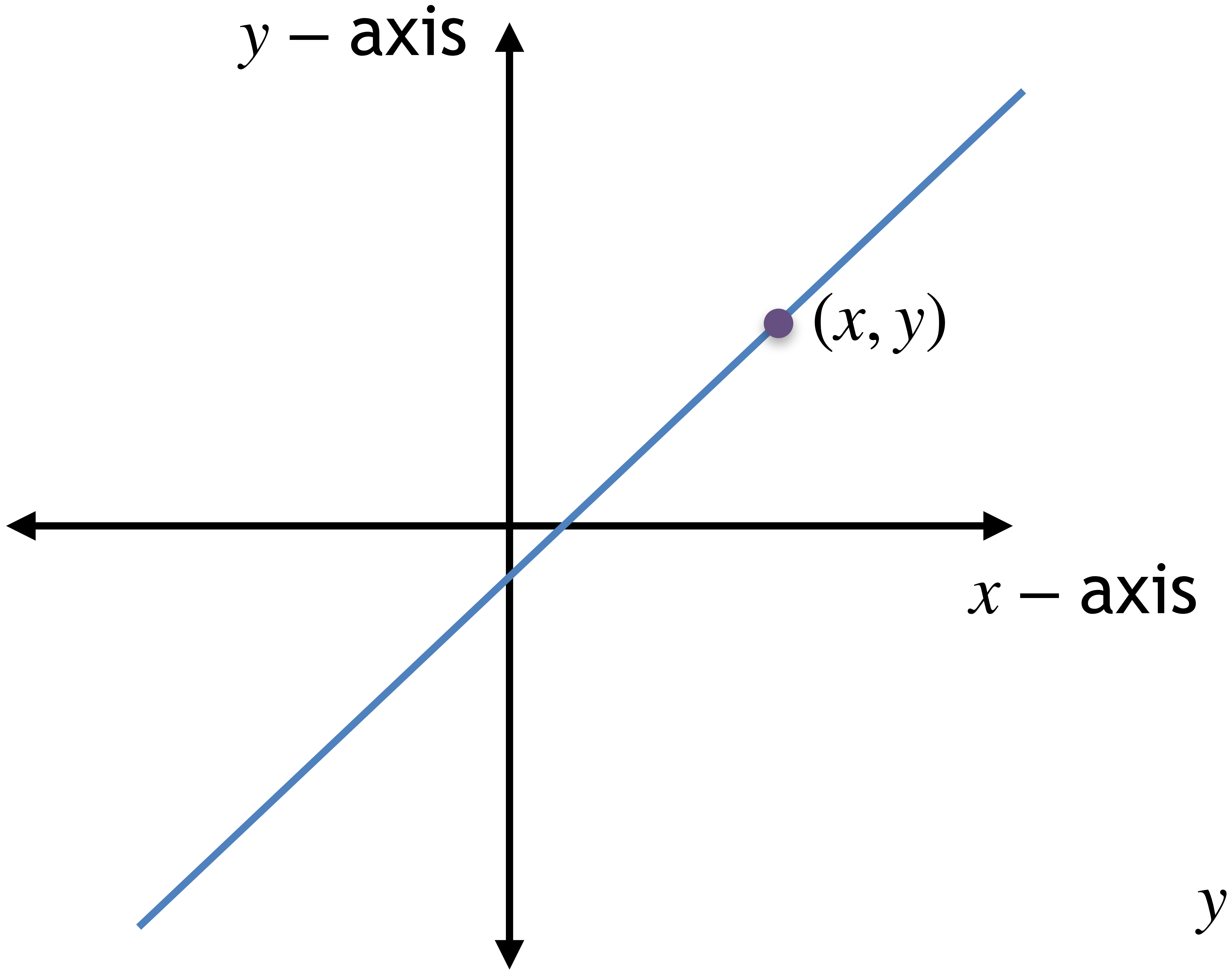
Example: fit line with just one input/output data sample $(x, y)$




$$y = w_1 x + w_0$$

# $\ell 1$ regularisation / the lasso

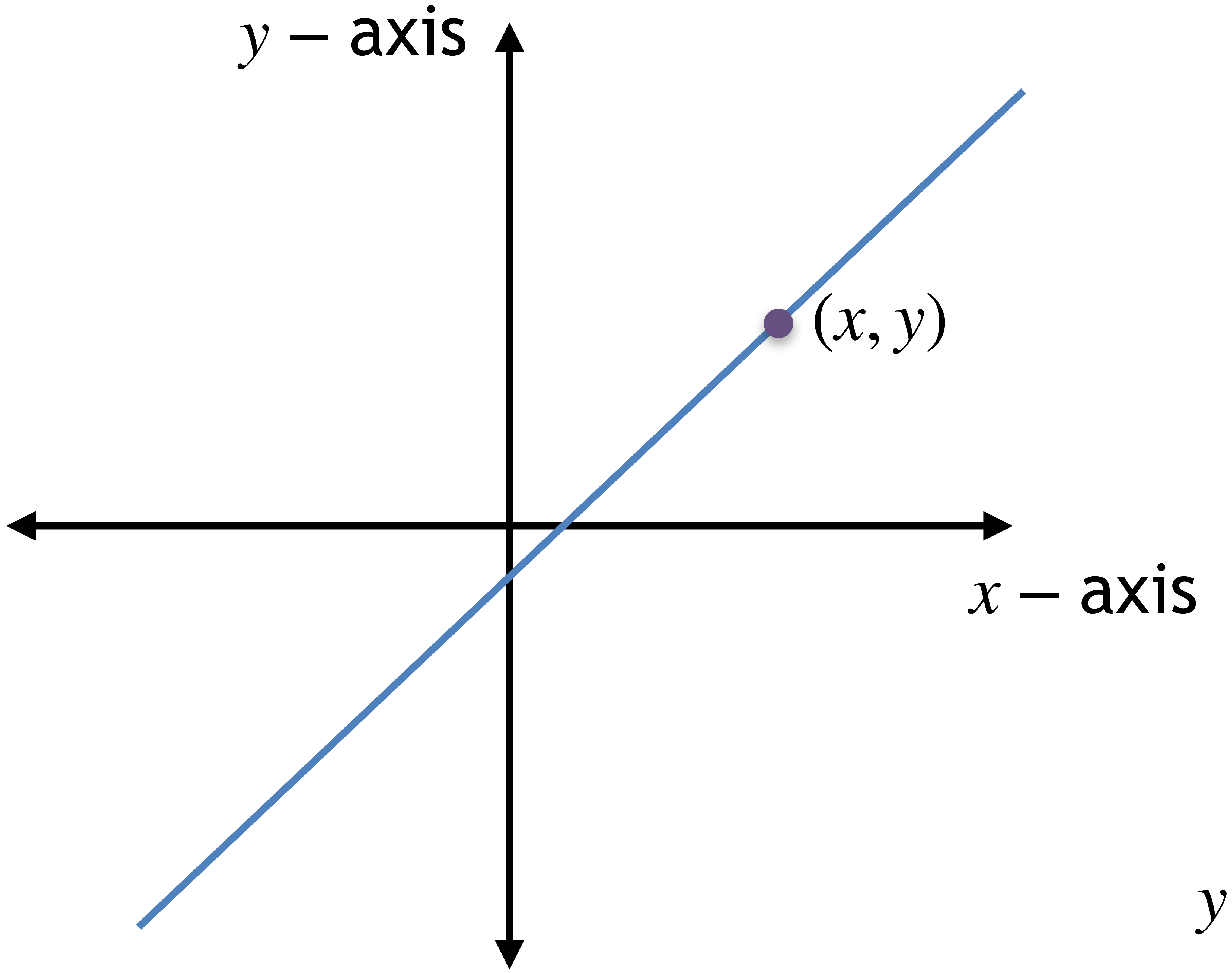
Example: fit line with just one input/output data sample $(x, y)$




$$y = w_1 x + w_0$$

# $\ell 1$ regularisation / the lasso

Example: fit line with just one input/output data sample $(x, y)$

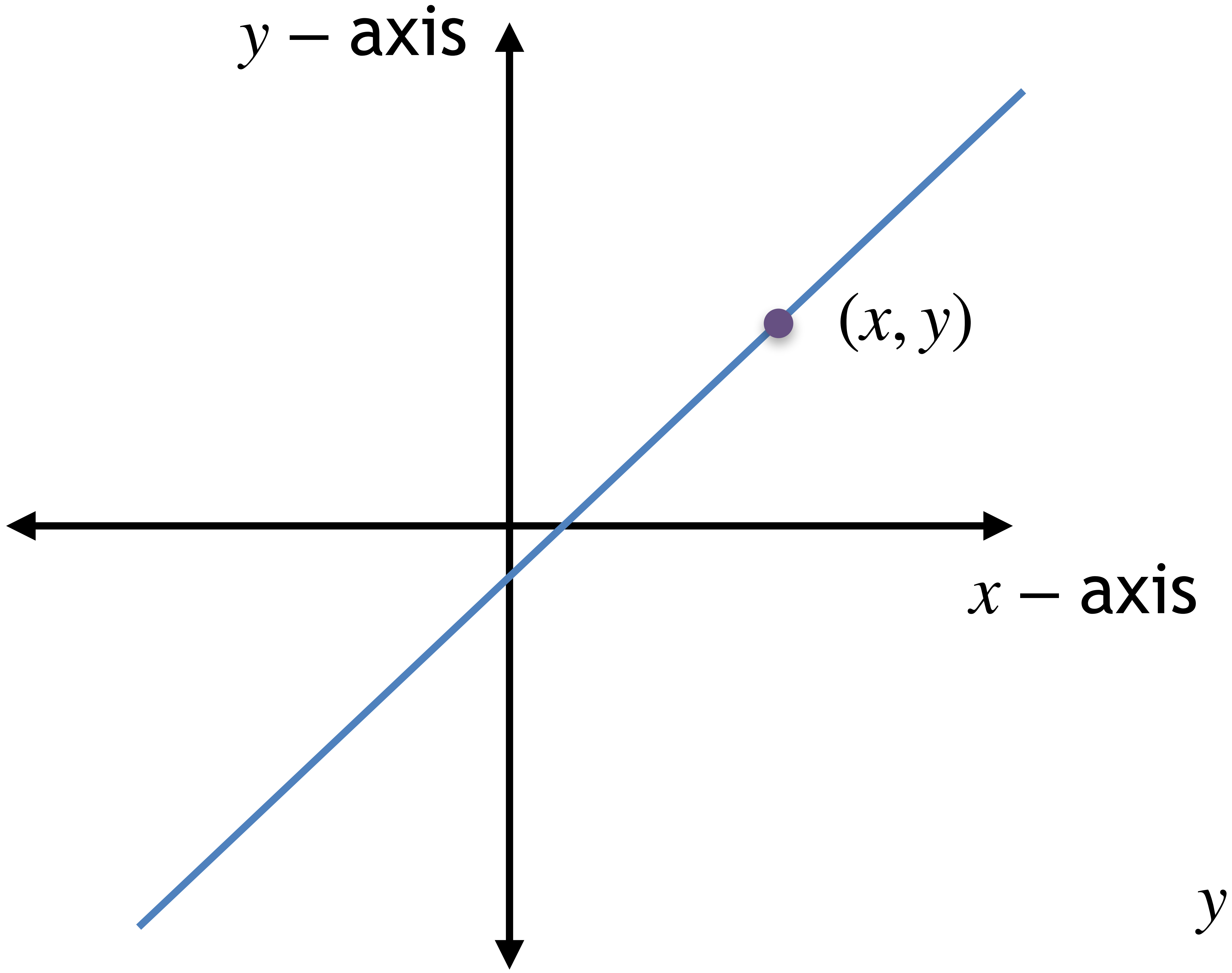



Which solution do we pick?

$$y = w_1 x + w_0$$

# ℓ1 regularisation / the lasso

Example: fit line with just one input/output data sample $(x, y)$

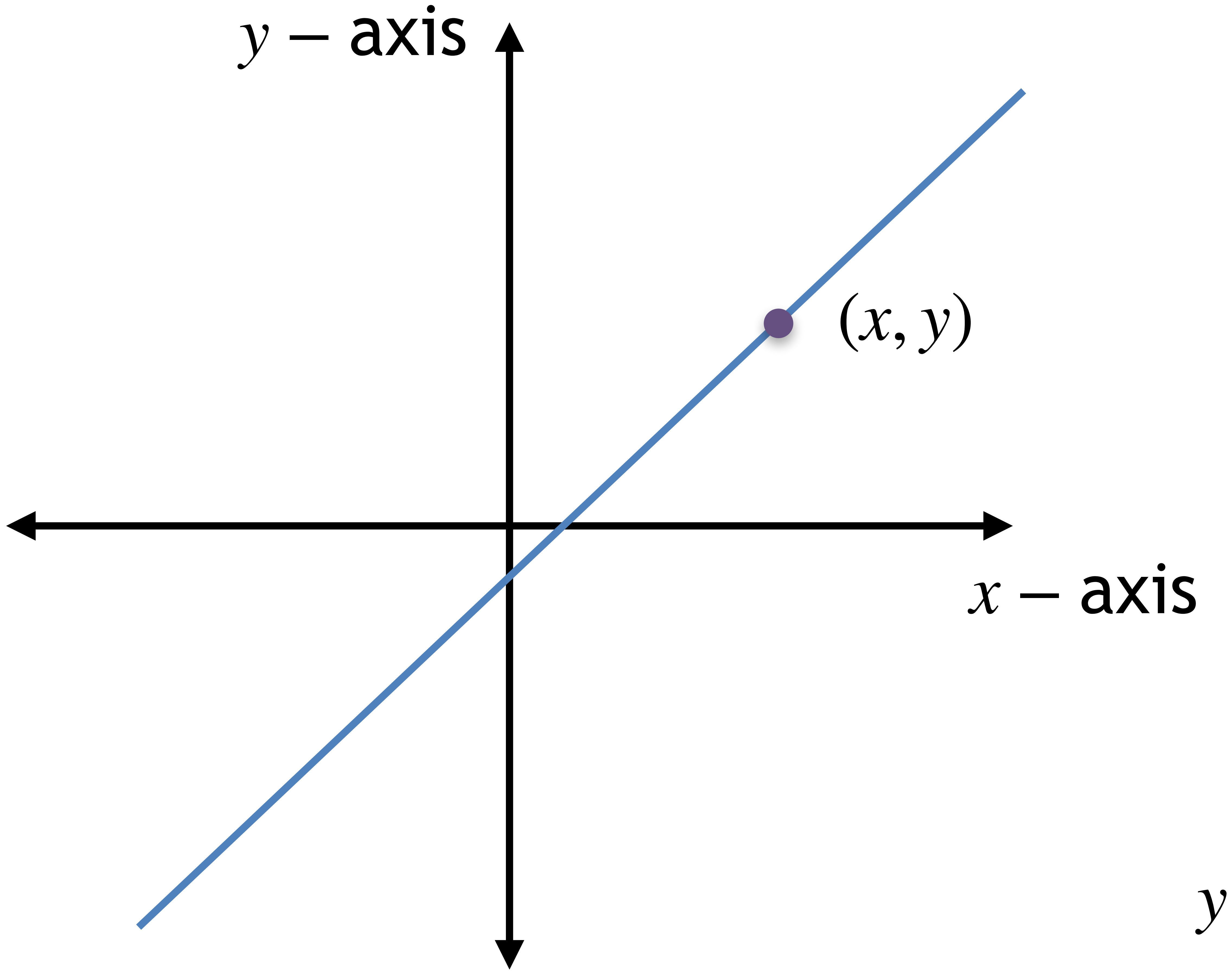



Simplicity idea:

$$y = w_1 x + w_0$$

# $\ell 1$ regularisation / the lasso

Example: fit line with just one input/output data sample $(x, y)$



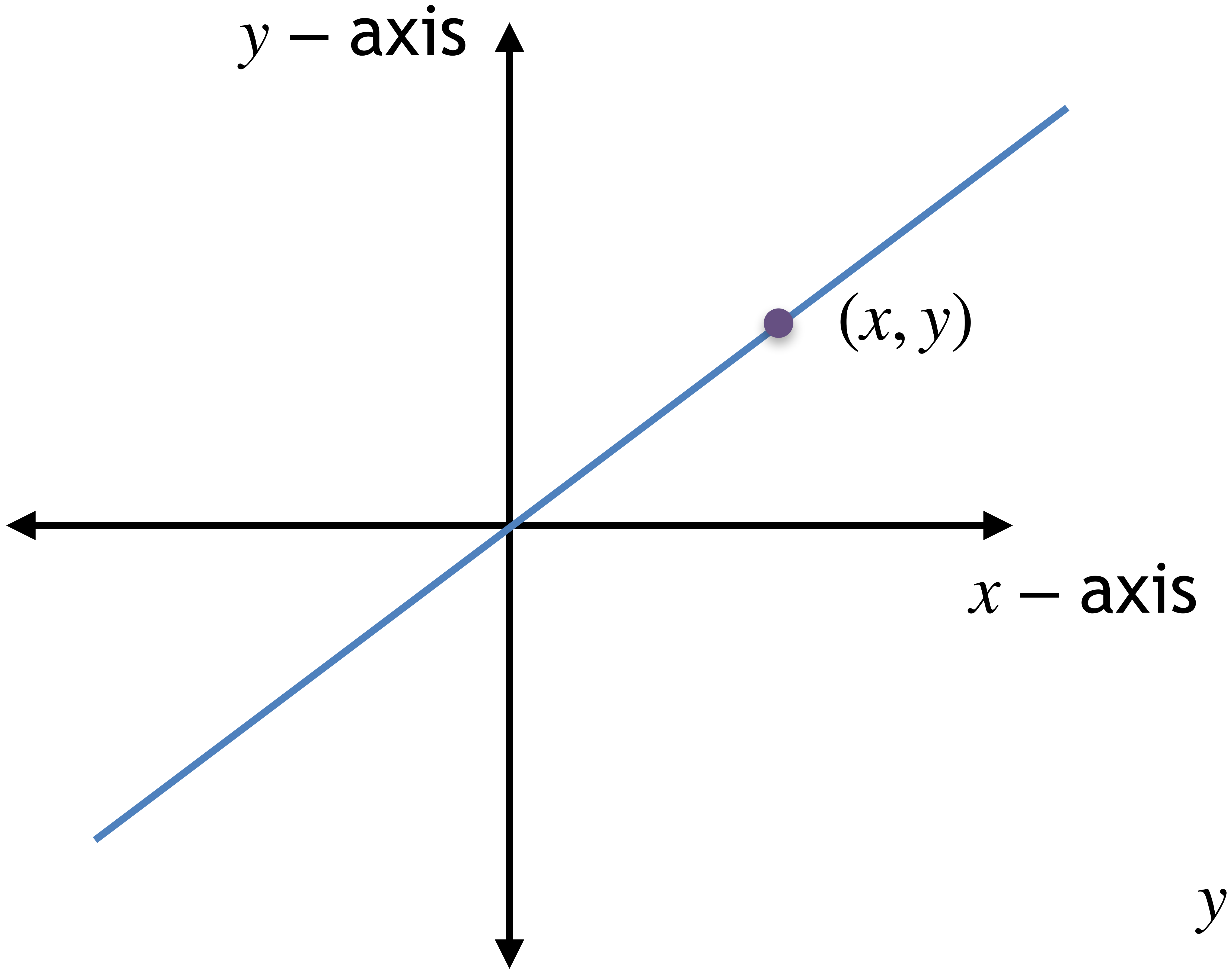

Simplicity idea:

Let either $w_0$ or $w_1$ be zero!

$$y = w_1 x + w_0$$

# ℓ1 regularisation / the lasso

Example: fit line with just one input/output data sample $(x, y)$




Simplicity idea:

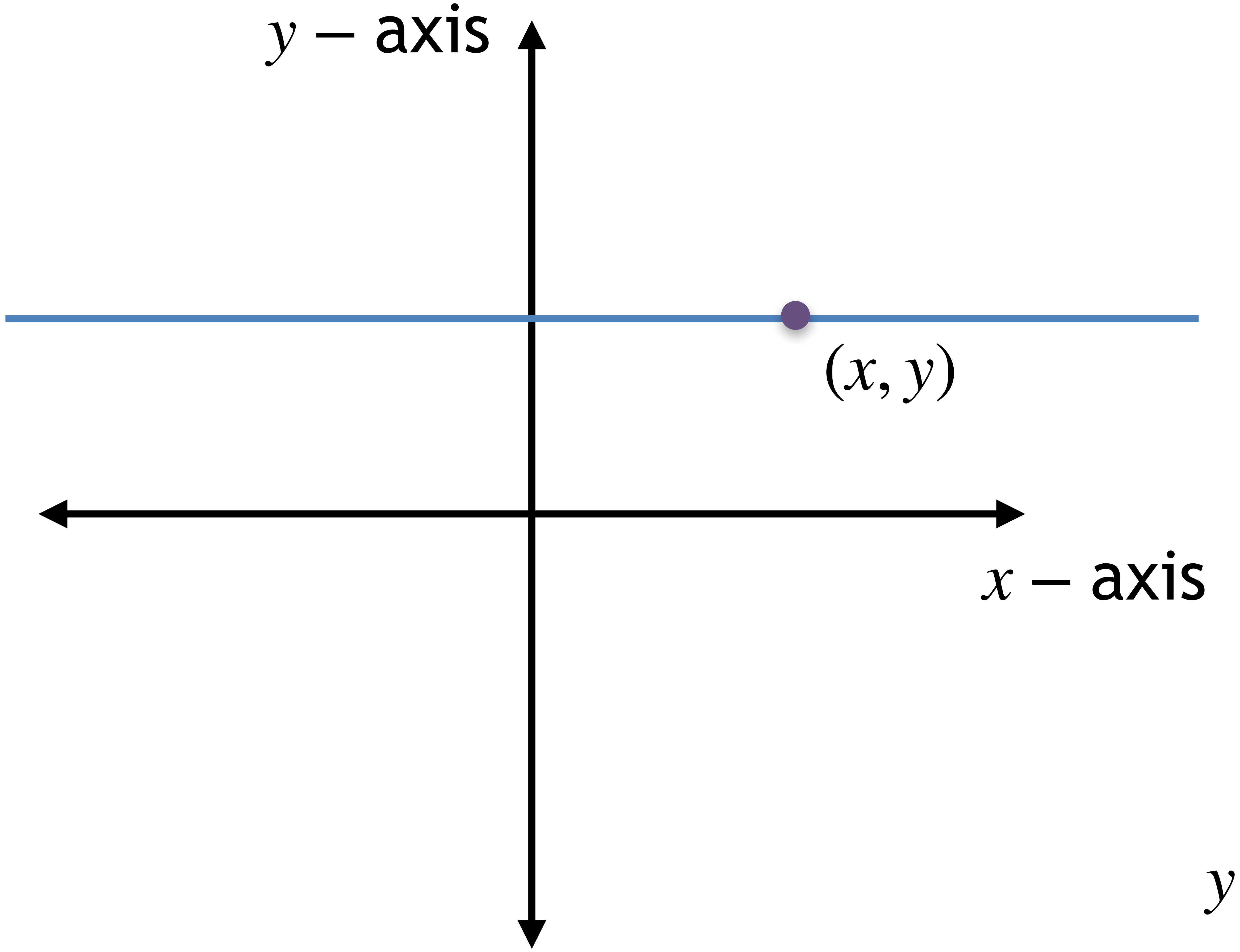
$$w_0 = 0$$

$$y = xw_1$$

$$y = w_1 x + w_0$$

# $\ell$1 regularisation / the lasso

Example: fit line with just one input/output data sample $(x, y)$




Simplicity idea:

$$w_1 = 0$$

$$y = w_0$$

$$y = w_1 x + w_0$$

# $\ell 1$ regularisation / the lasso

# $\ell$1 regularisation / the lasso

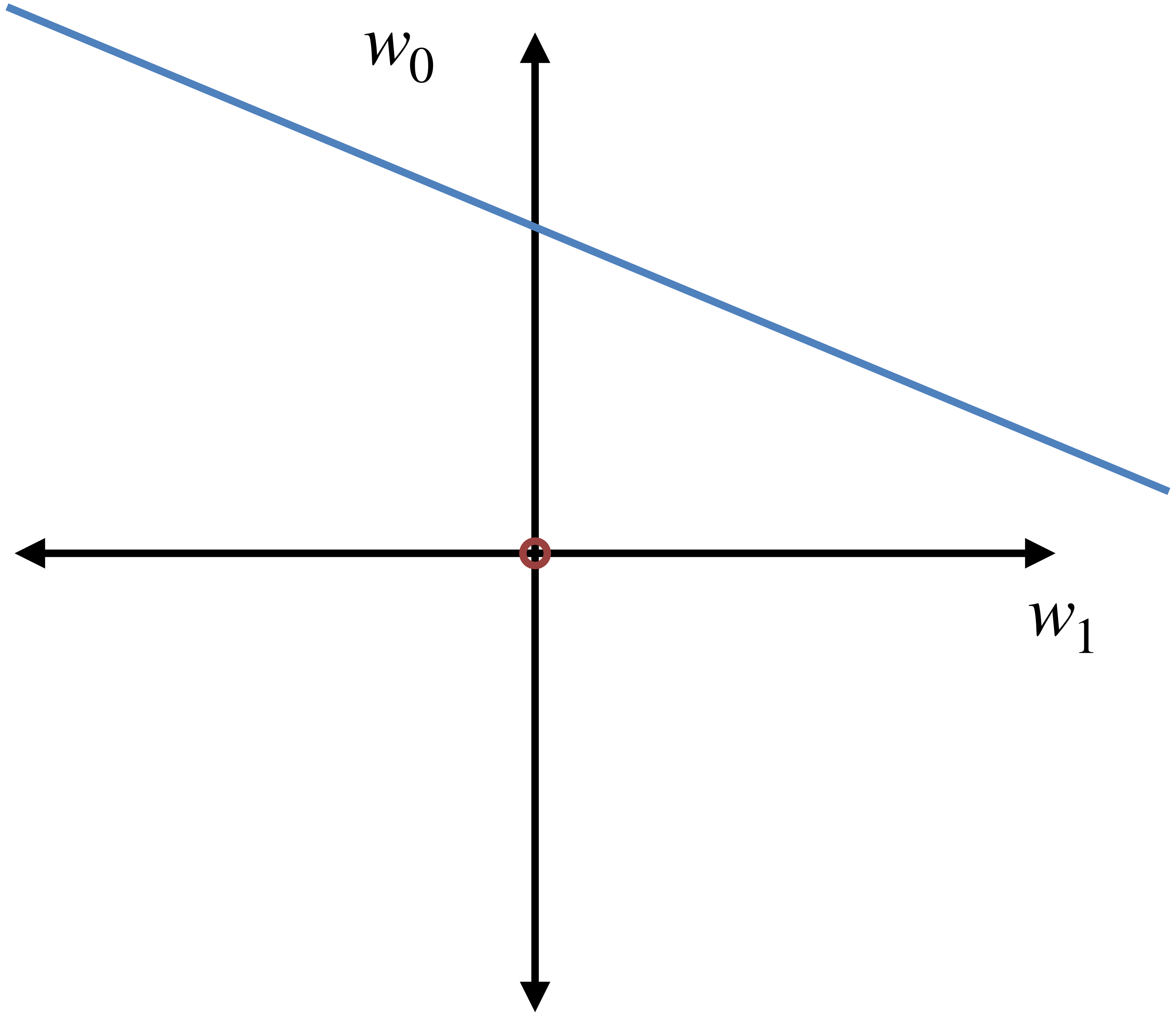
In general, why (or how) does the $\ell^1$ norm make $\hat{w}$ sparse?

# ℓ1 regularisation / the lasso

The solution of the problem

$$y = w_0 + w_1 x$$

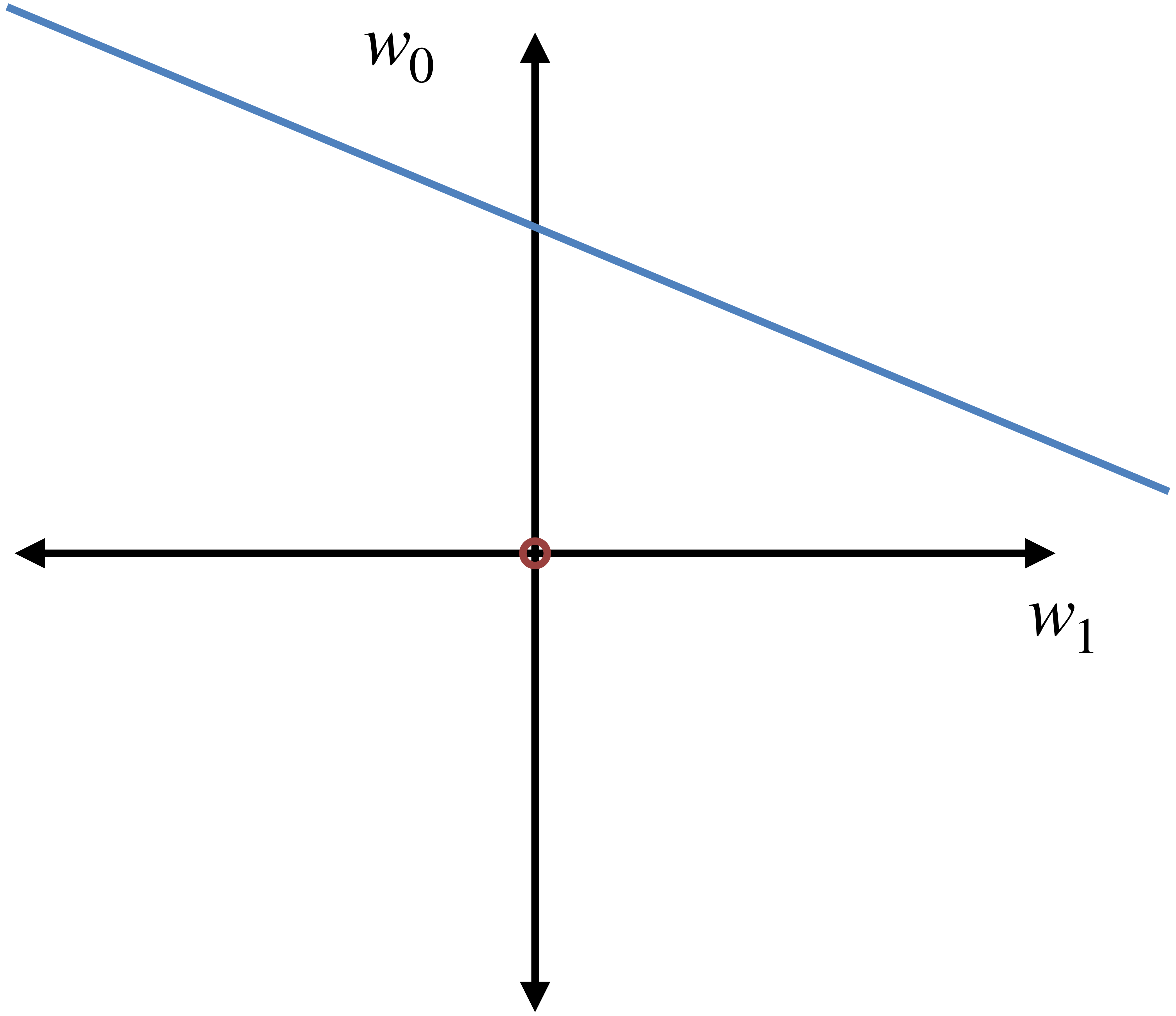
is a point in this space




We can indeed write

$$w_0 = y - w_1 x$$

# ℓ1 regularisation / the lasso



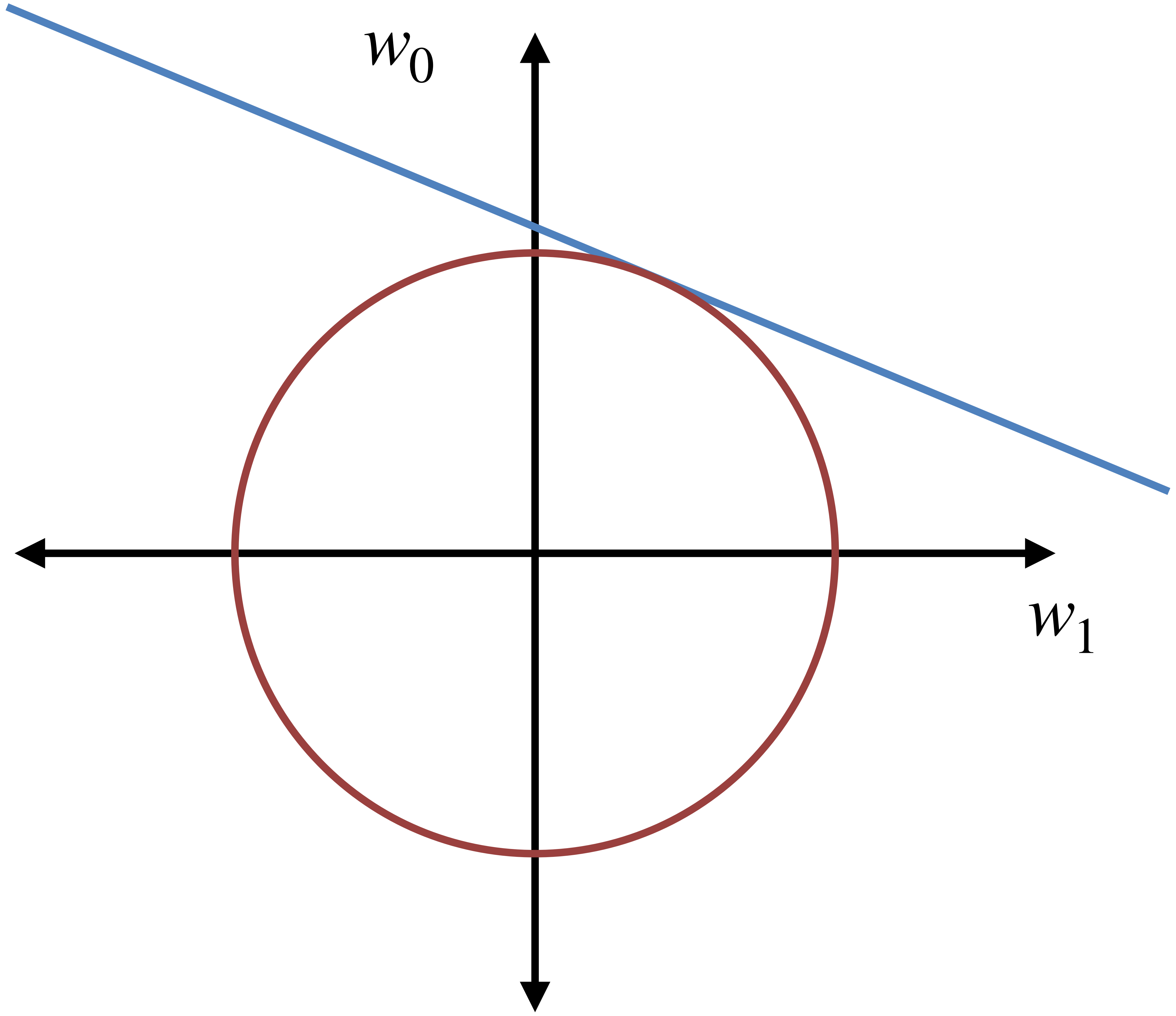

Minimise

$$\sqrt{w_0^2 + w_1^2}$$

subject to

$$w_0 = -w_1 x + y$$

# ℓ1 regularisation / the lasso



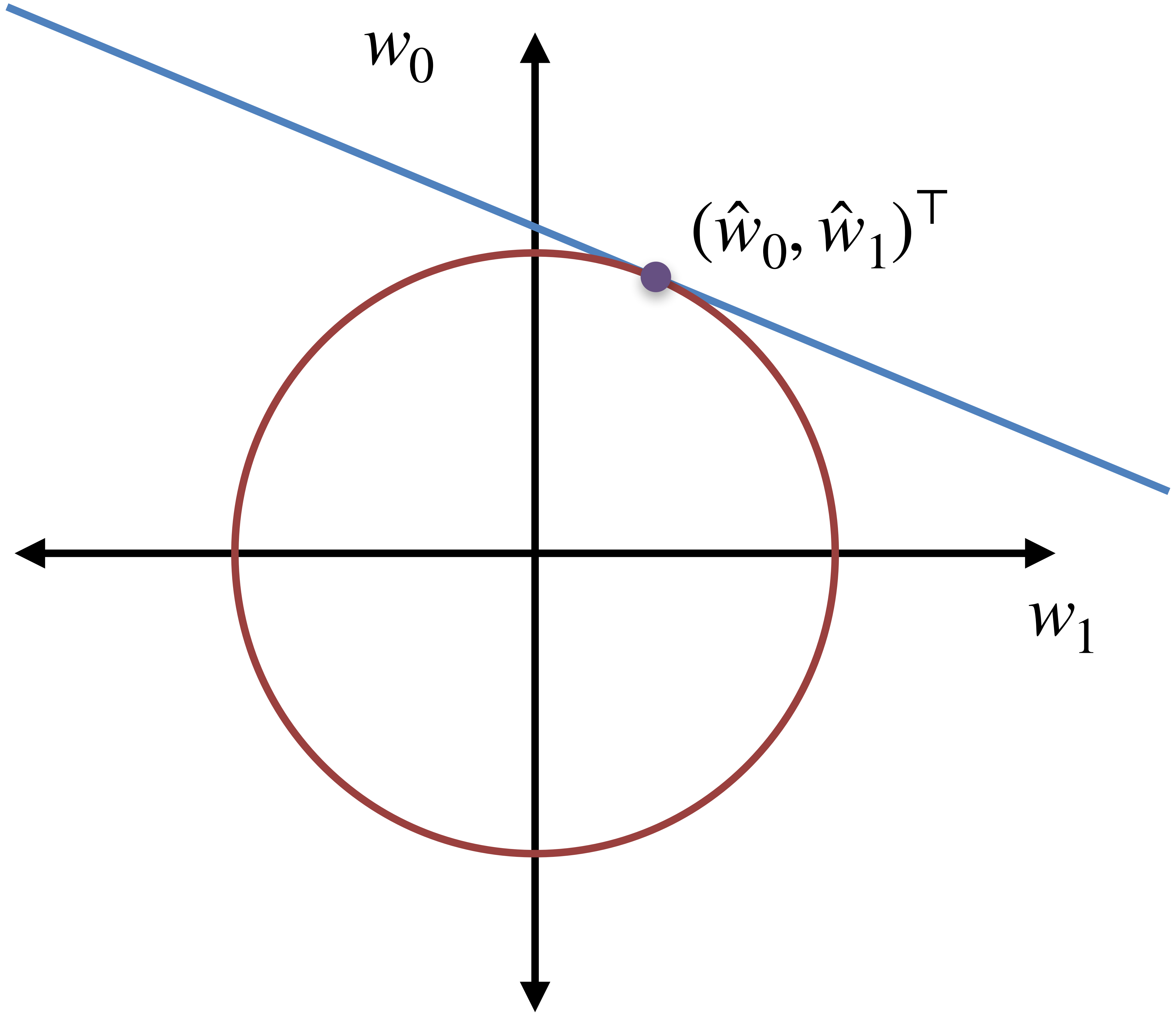

Minimise

$$\sqrt{w_0^2 + w_1^2}$$

subject to

$$w_0 = -w_1 x + y$$

# ℓ1 regularisation / the lasso



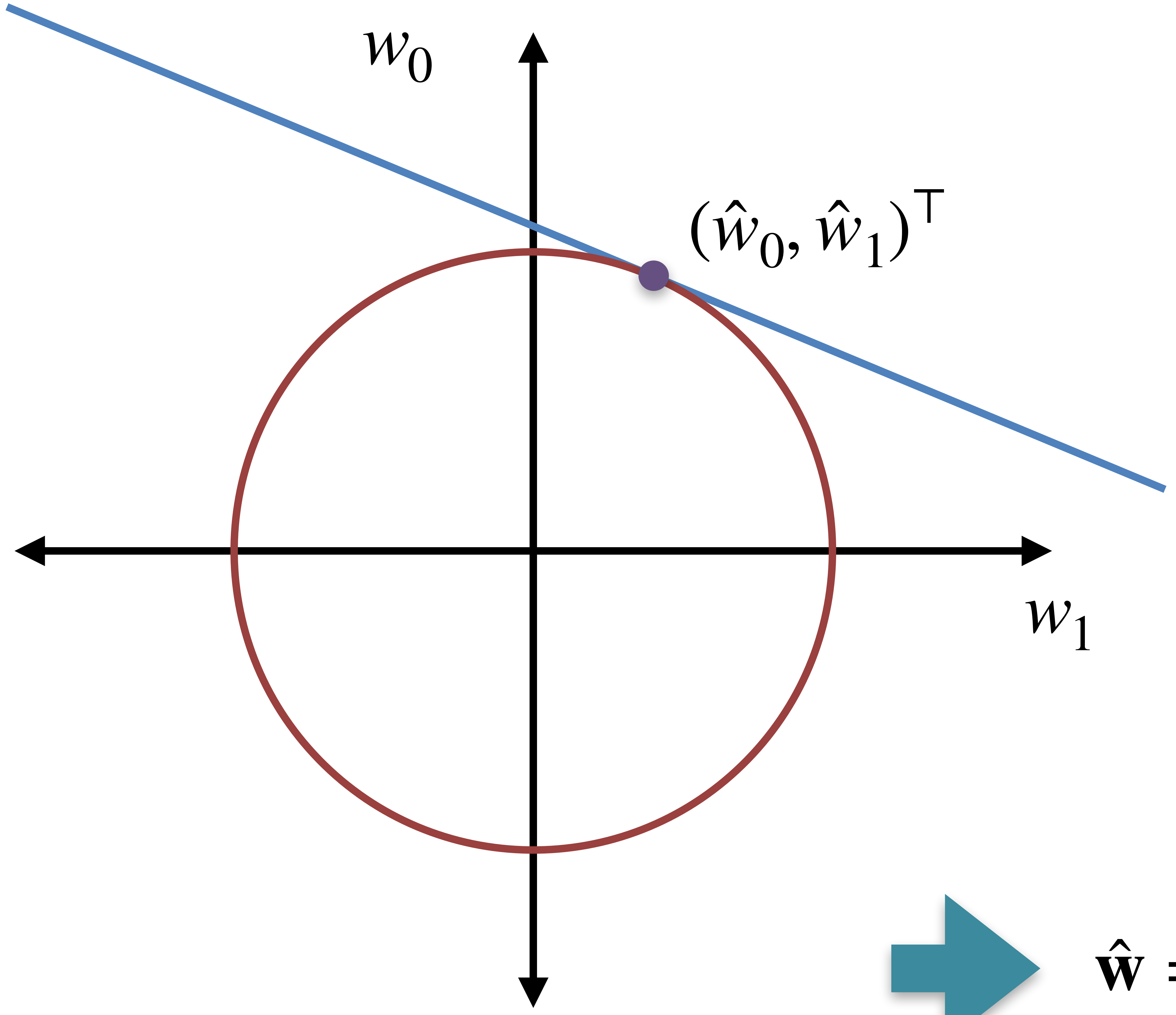

Minimise

$$\sqrt{w_0^2 + w_1^2}$$

subject to

$$w_0 = -w_1 x + y$$

# ℓ1 regularisation / the lasso




Minimise

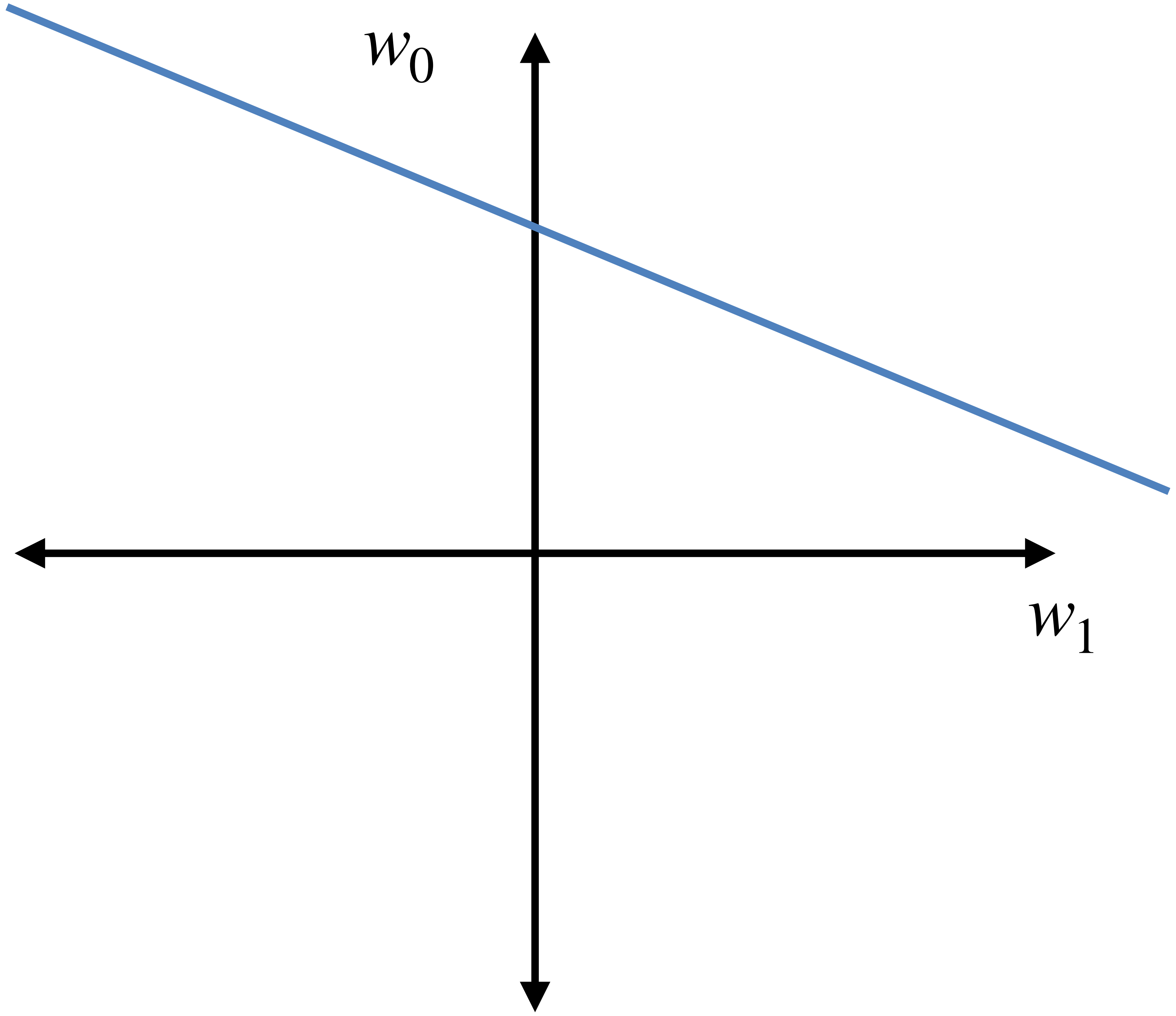
$$\sqrt{w_0^2 + w_1^2}$$

subject to

$$w_0 = -w_1 x + y$$

⇒ $\hat{\mathbf{w}} = (\hat{w}_0, \hat{w}_1)^\top$ most likely not sparse

# ℓ1 regularisation / the lasso



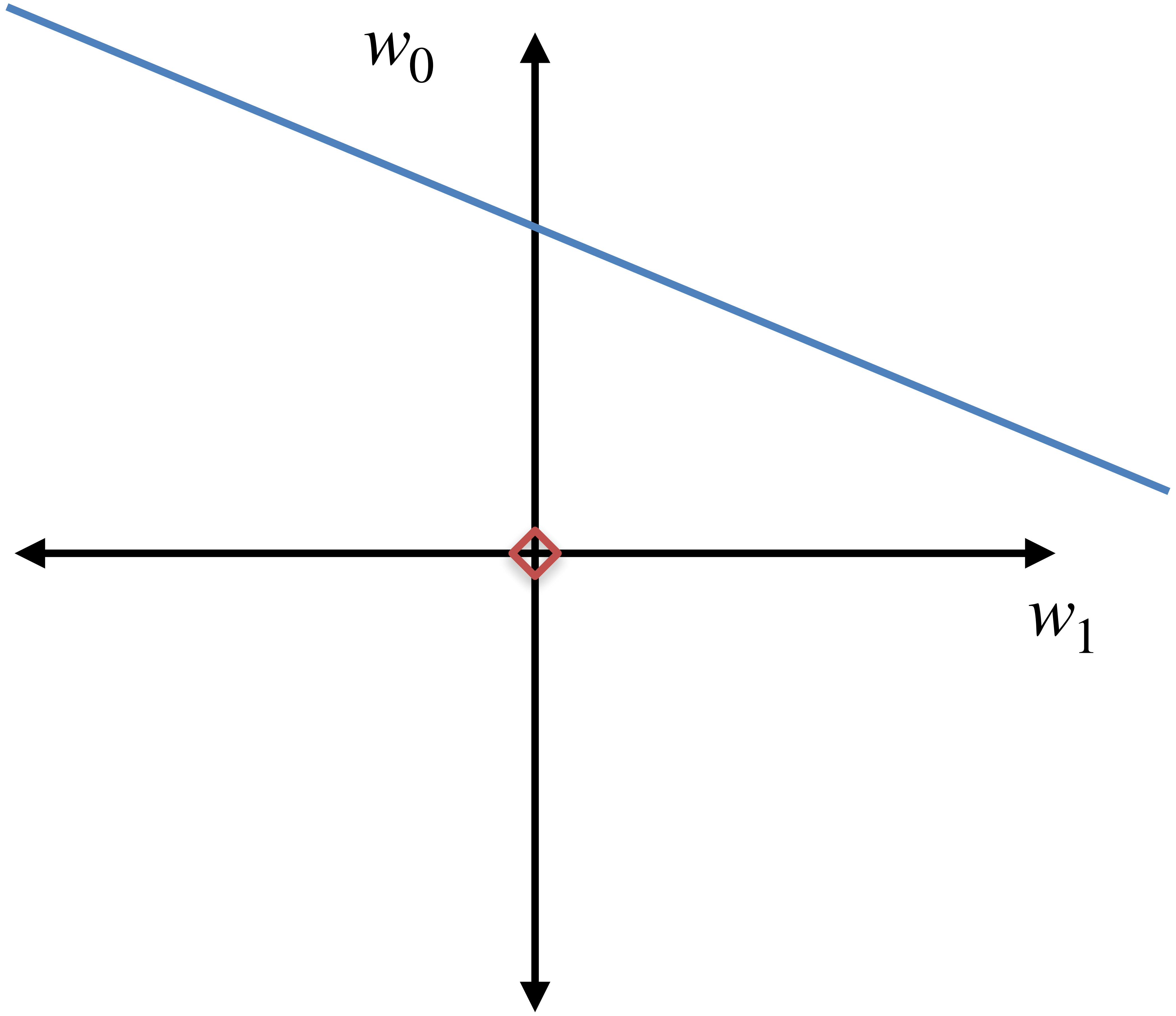

Minimise

$$|w_0| + |w_1|$$

subject to

$$w_0 = -w_1 x + y$$

# ℓ1 regularisation / the lasso



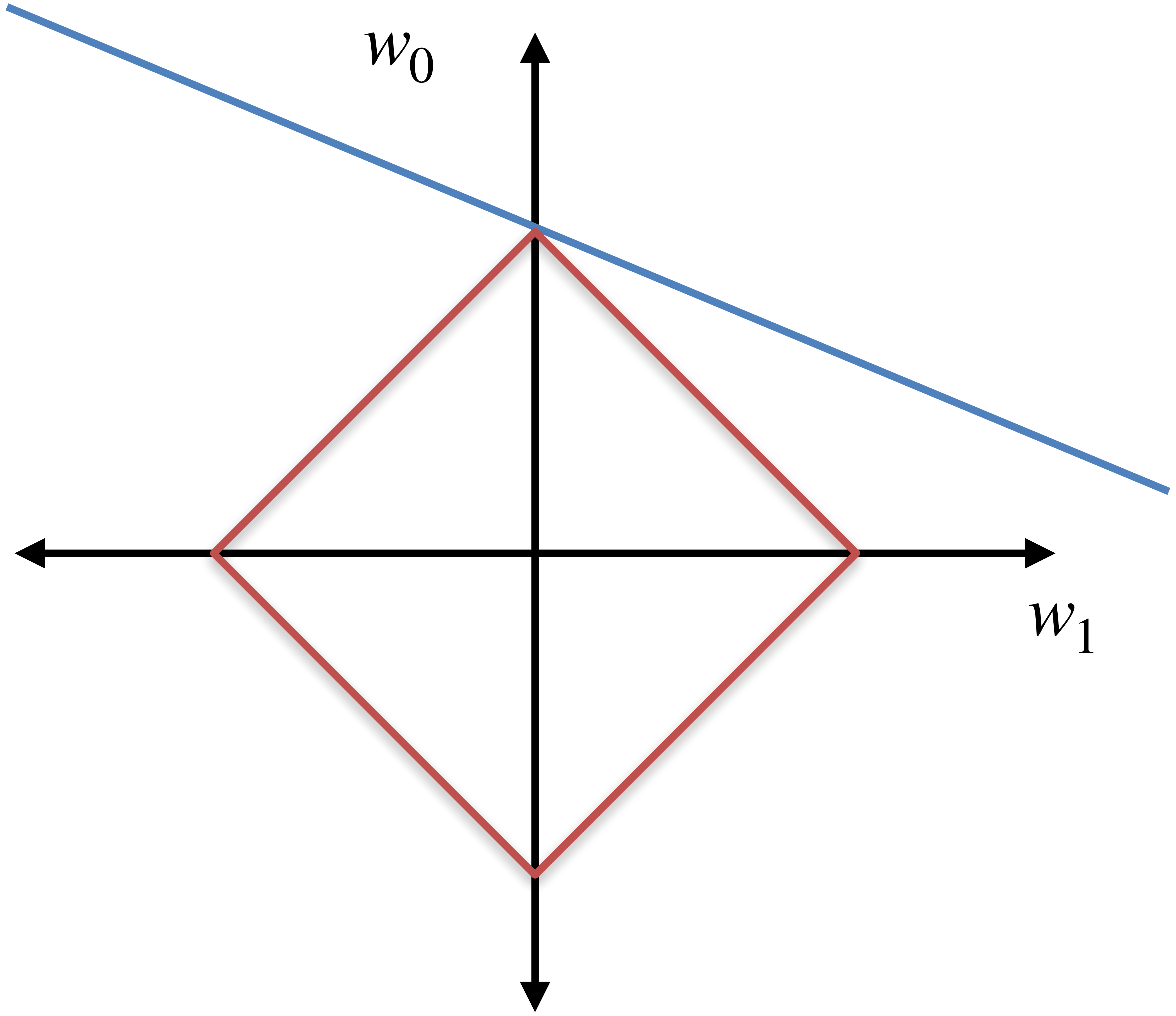

Minimise

$$|w_0| + |w_1|$$

subject to

$$w_0 = -w_1 x + y$$

# ℓ1 regularisation / the lasso



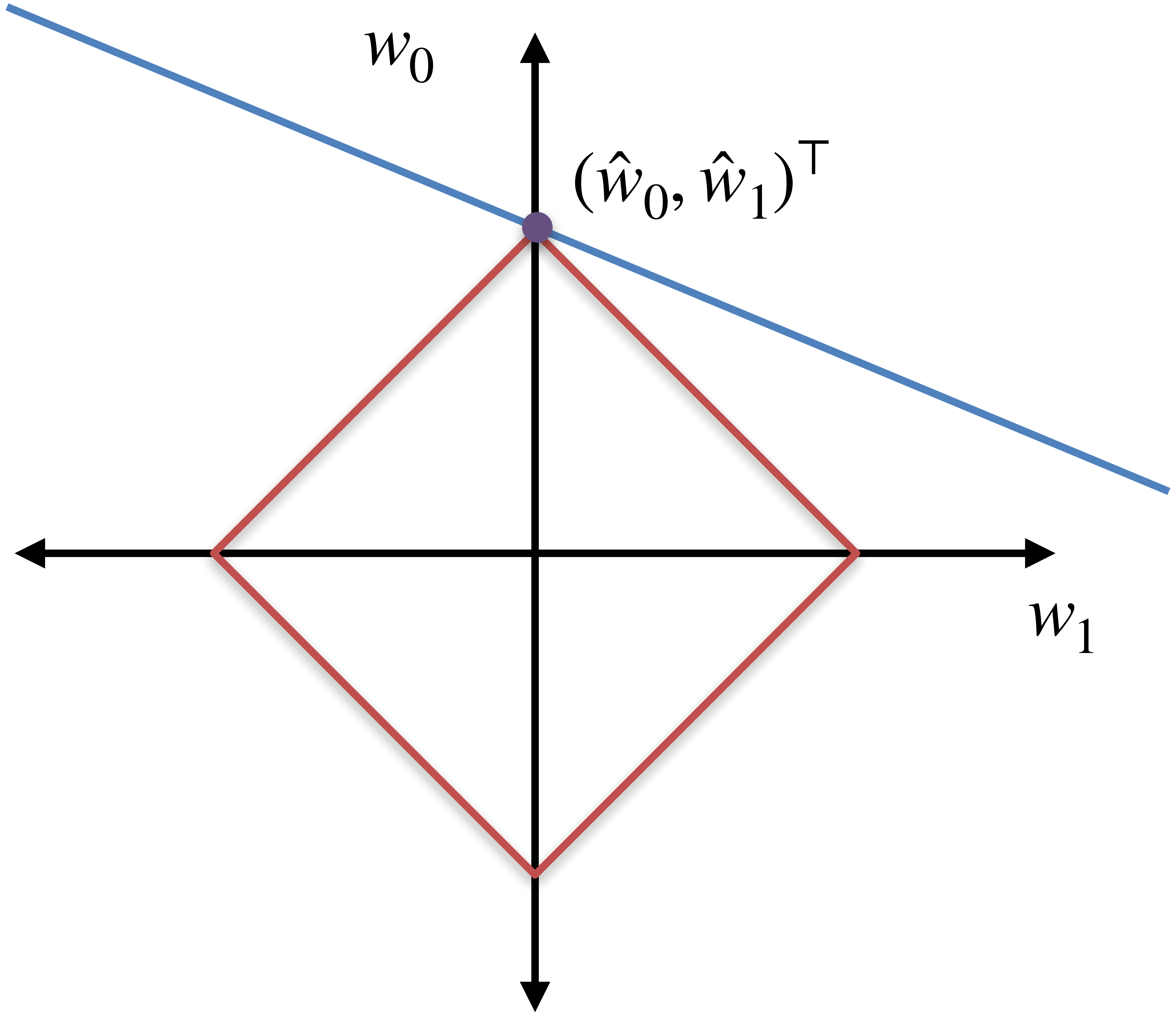

Minimise

$$|w_0| + |w_1|$$

subject to

$$w_0 = -w_1 x + y$$

# ℓ1 regularisation / the lasso



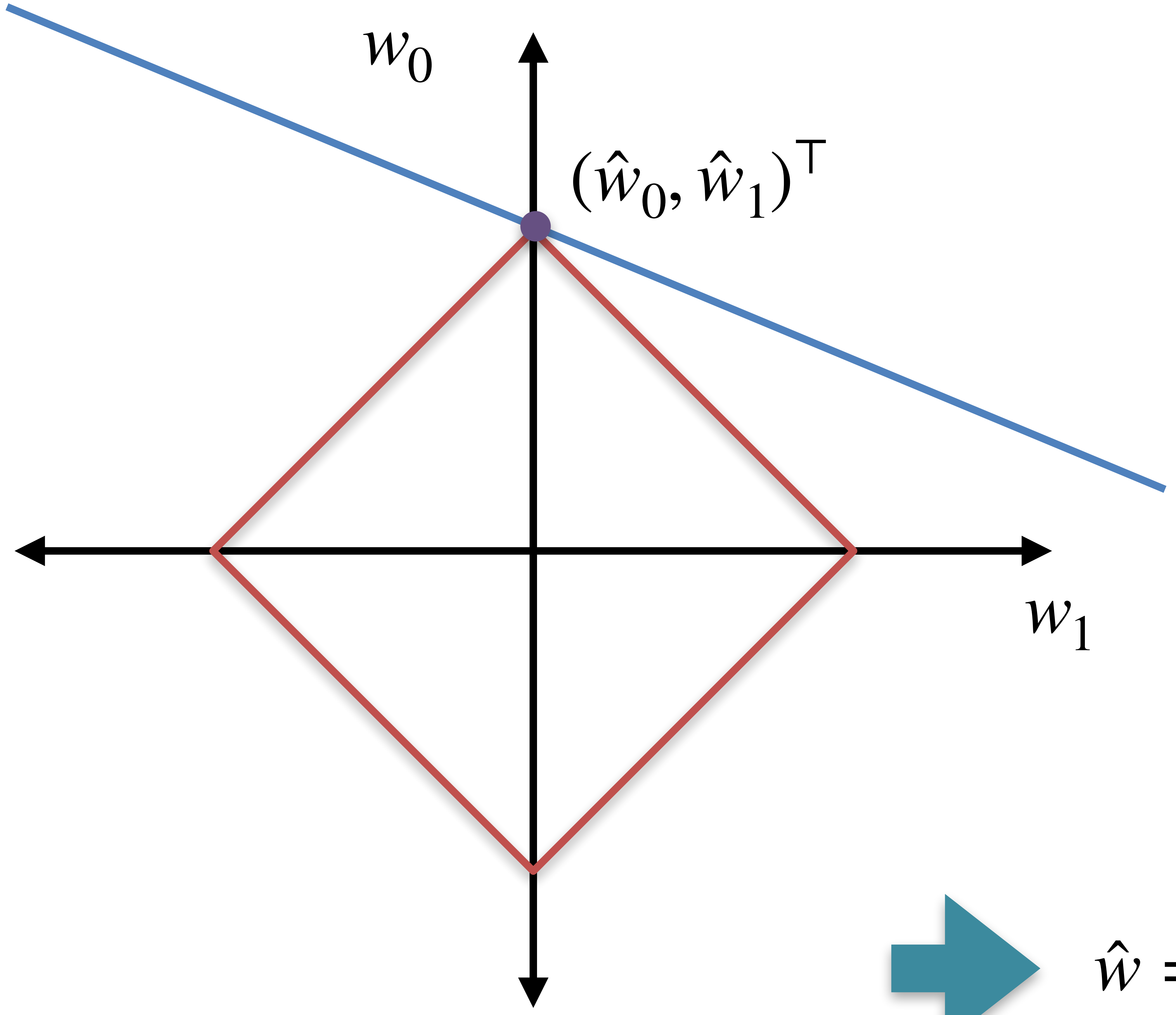

Minimise

$$|w_0| + |w_1|$$

subject to

$$w_0 = -w_1 x + y$$

# ℓ1 regularisation / the lasso




Minimise

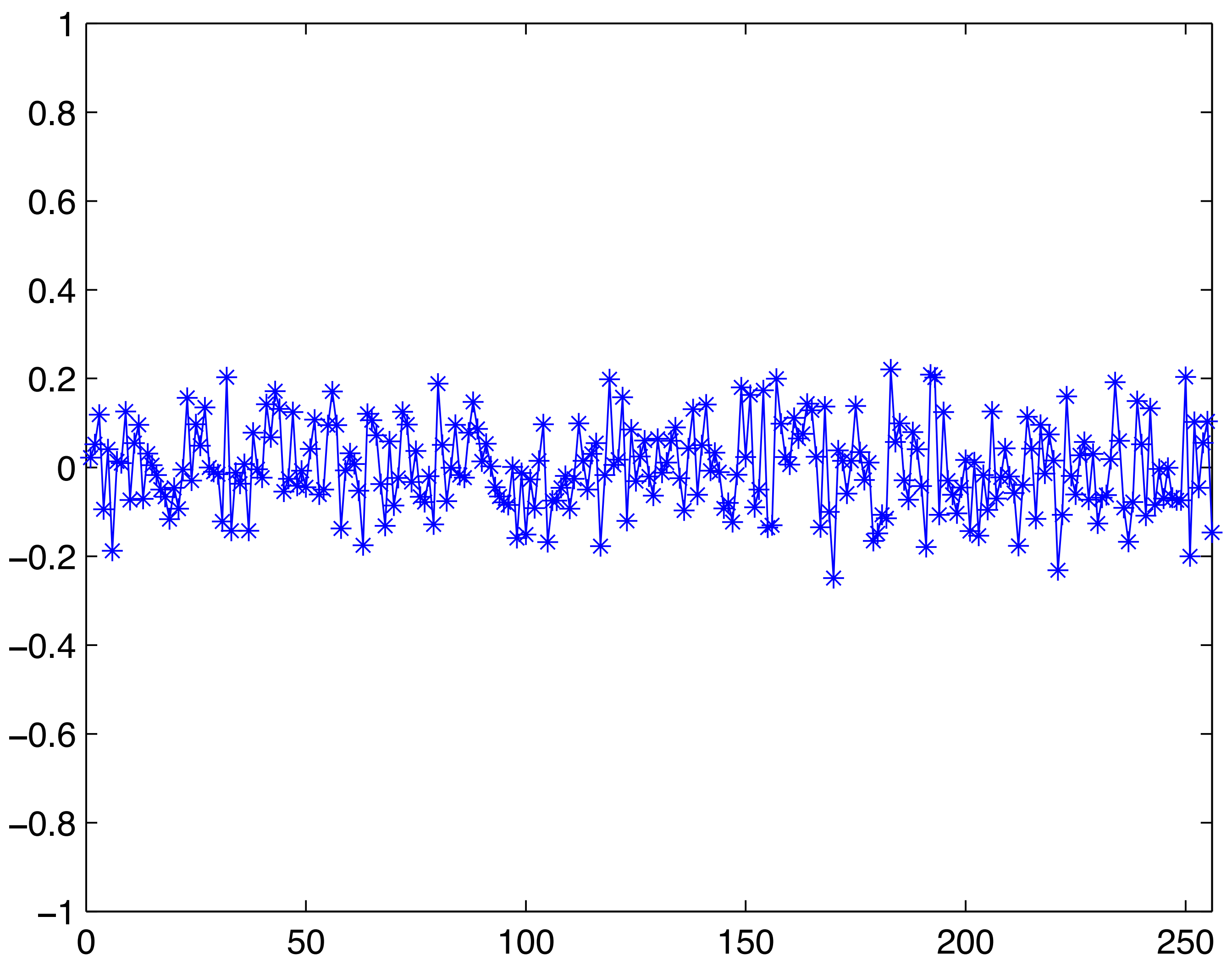
$$|w_0| + |w_1|$$

subject to

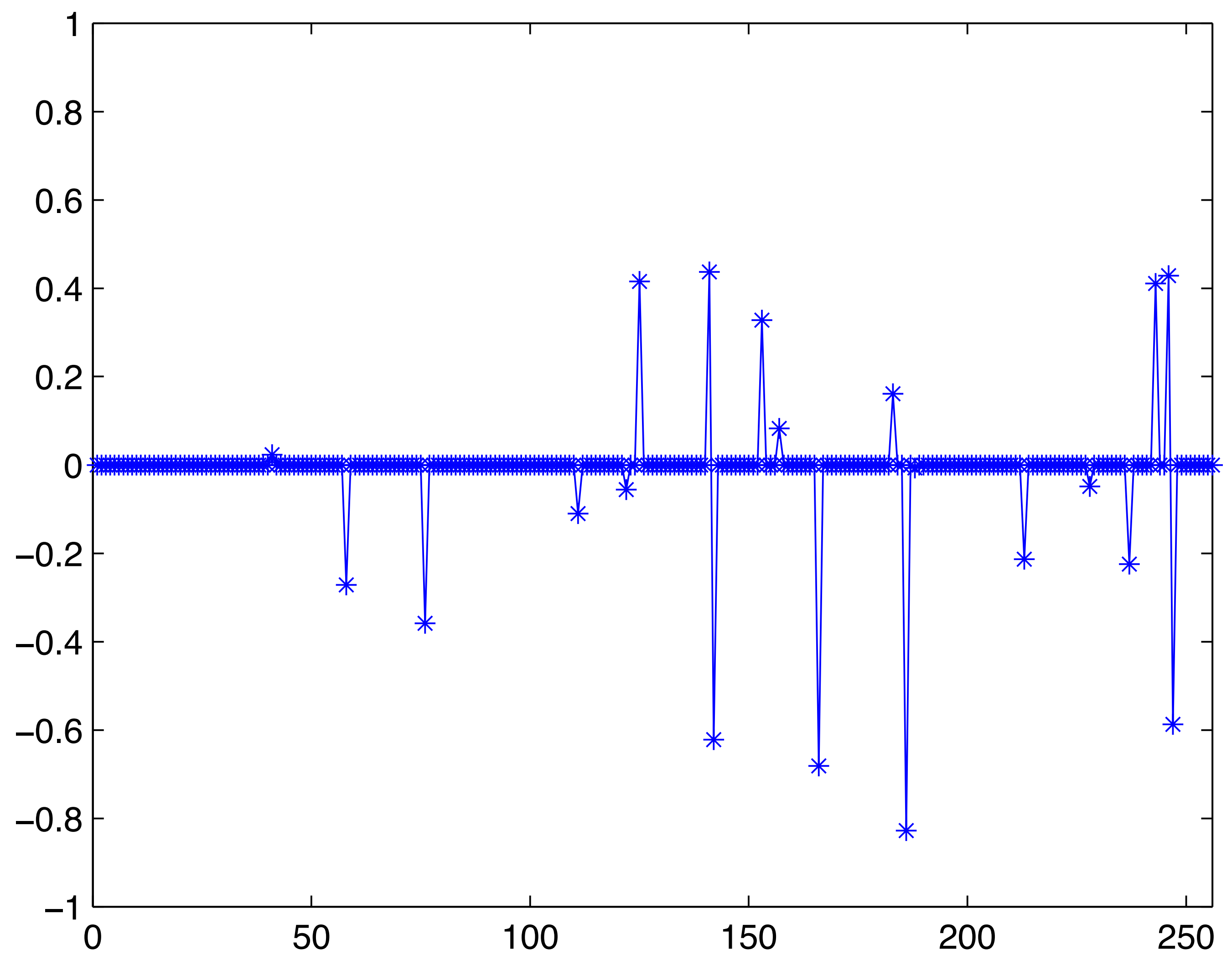
$$w_0 = -w_1 x + y$$

$\hat{w} = (\hat{w}_0, \hat{w}_1)^\top$ most likely sparse! One of the coordinates must be zero

# ℓ1 regularisation / the lasso

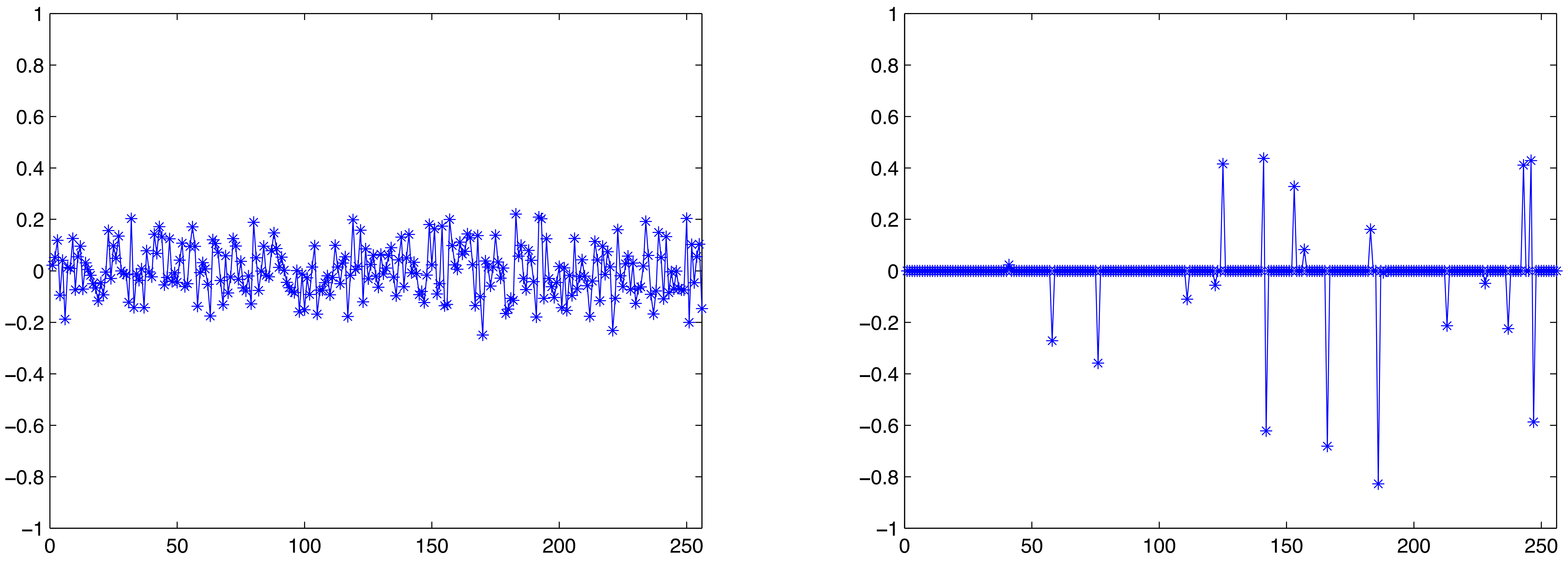



(a) Dense




(b) Sparse

# ℓ1 regularisation / the lasso

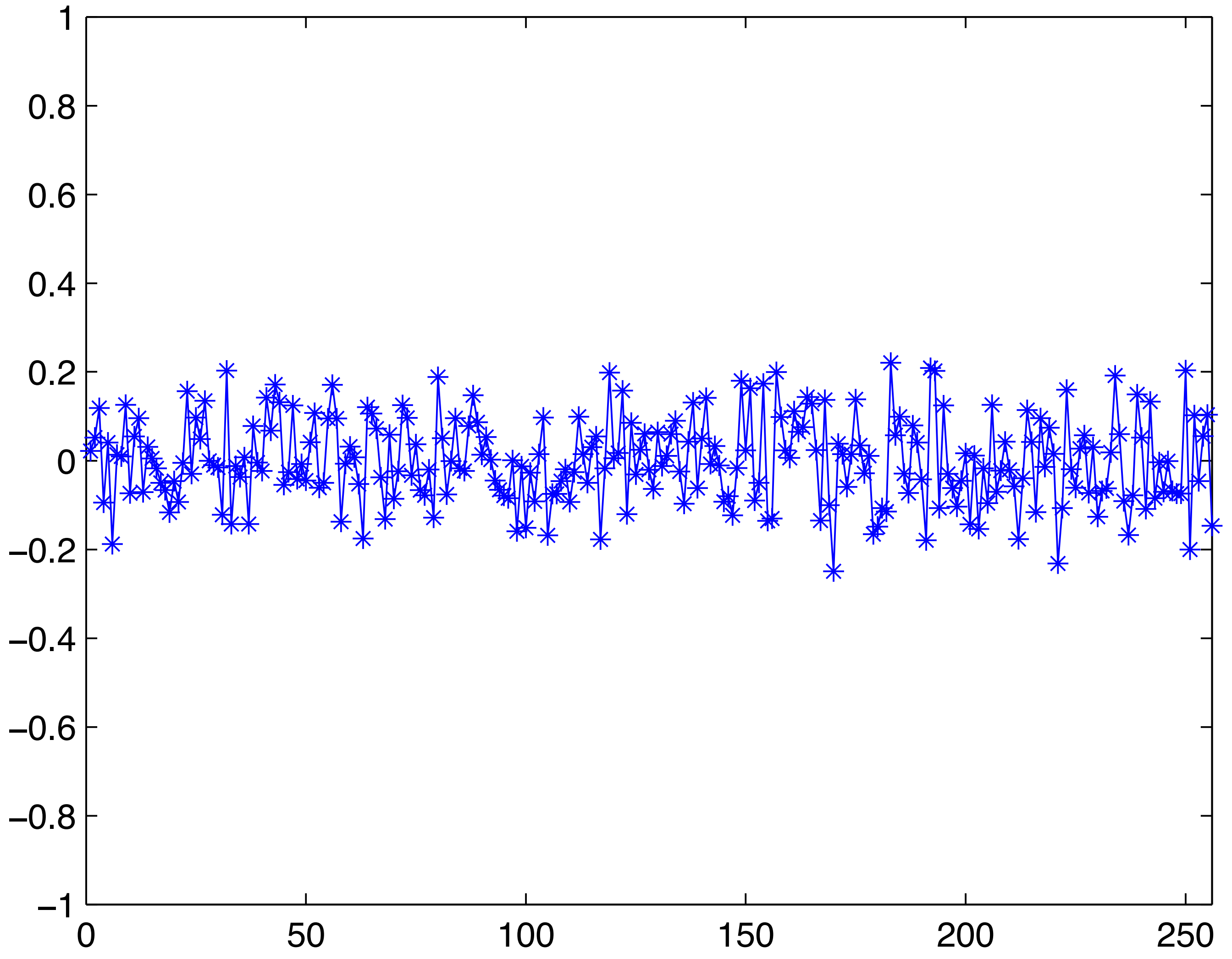



(a) Dense

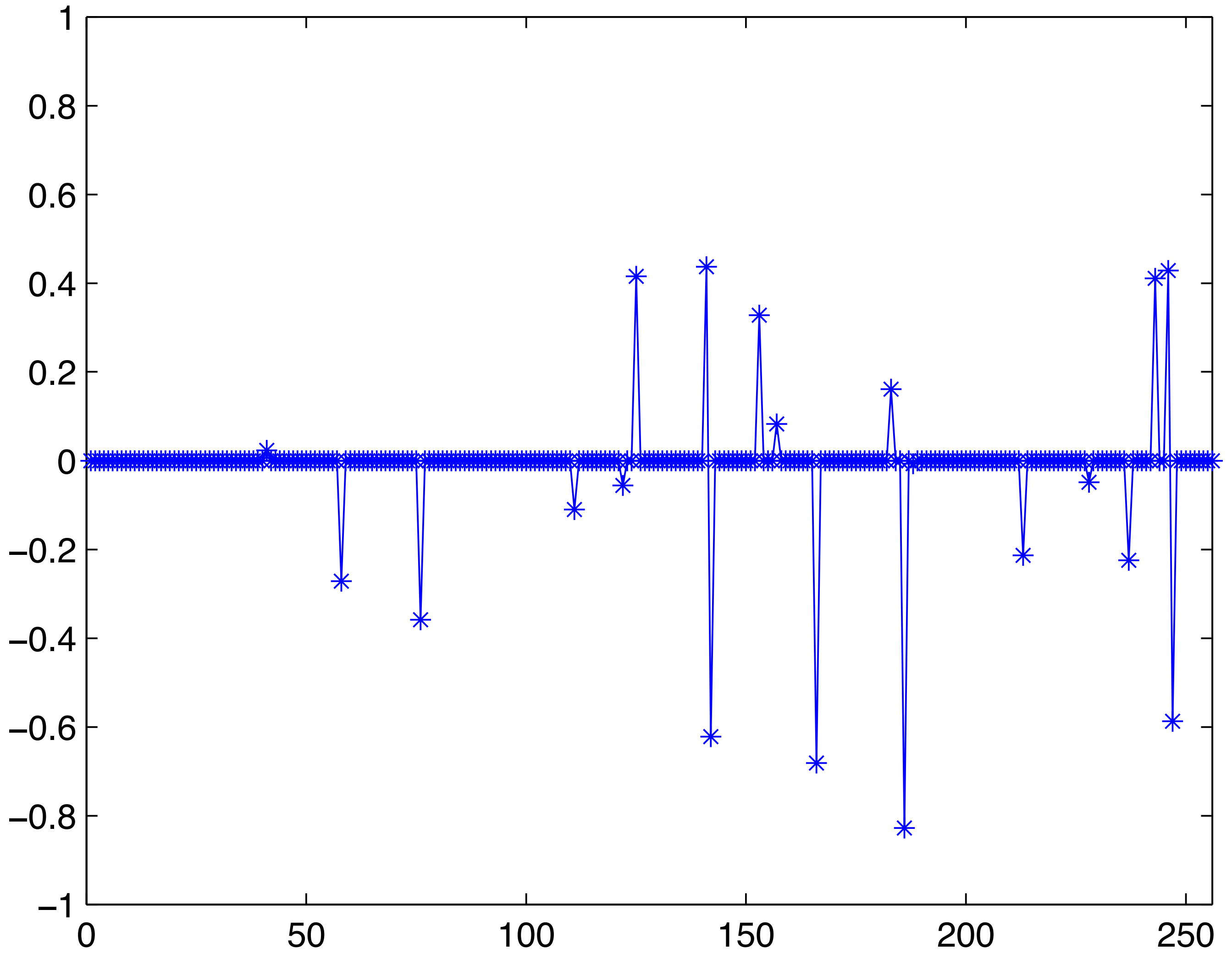
(b) Sparse

$\|\text{signal in a)}\|_2 \approx 1.5431$ $\|\text{signal in a)}\|_1 \approx 20.061$

# ℓ1 regularisation / the lasso




(a) Dense




(b) Sparse

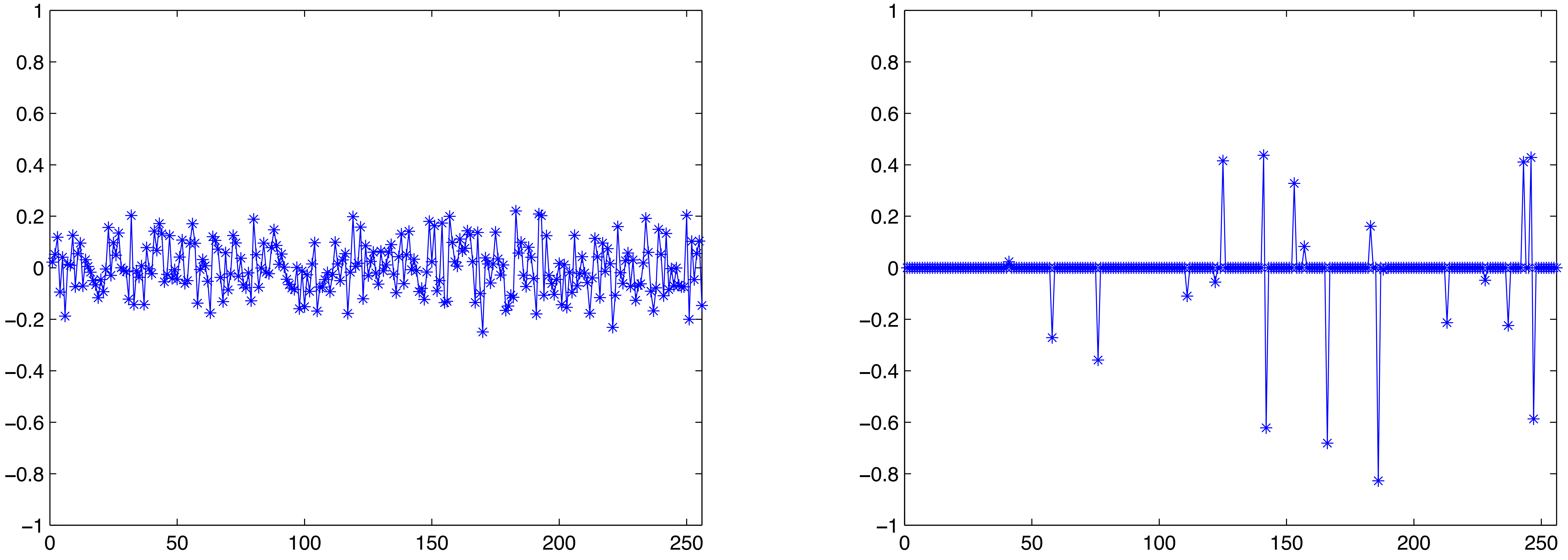
$\|\text{signal in a)}\|_2 \approx 1.5431$

$\|\text{signal in b)}\|_2 \approx 1.7472$

$\|\text{signal in a)}\|_1 \approx 20.061$

$\|\text{signal in b)}\|_1 \approx 6.2931$

# ℓ1 regularisation / the lasso




(a) Dense

(b) Sparse

$\|\text{signal in a)}\|_2 \approx 1.5431$

$\|\text{signal in a)}\|_1 \approx 20.061$

$\|\text{signal in b)}\|_2 \approx 1.7472$

$\|\text{signal in b)}\|_1 \approx 6.2931$

Lasso would select the sparse solution!

# HOW TO SOLVE LASSO OR MORE IN GENERAL OPTIMIZATION PROBLEMS?

# Why optimisation?

In the previous lectures, we have studied regression problems of the form

$$\hat{\mathbf{w}} = \arg \min_{\mathbf{w} \in \mathbb{R}^{d+1}} E(\mathbf{w})$$

# Why optimisation?

In the previous lectures, we have studied regression problems of the form

$$\hat{\mathbf{w}} = \arg \min_{\mathbf{w} \in \mathbb{R}^{d+1}} E(\mathbf{w})$$

For

$$E(\mathbf{w}) = \text{MSE}(\mathbf{w}) = \frac{1}{2s} \sum_{i=1}^{s} |f(x_i, w) - y_i|^2 \quad ,$$

where $f$ is linear in $w$, we have seen that we can compute $\hat{w}$ by solving a linear system of equations

# Why optimisation?

In the previous lectures, we have studied regression problems of the form

$$\hat{\mathbf{w}} = \arg \min_{\mathbf{w} \in \mathbb{R}^{d+1}} E(\mathbf{w})$$

For

$$E(\mathbf{w}) = \frac{1}{2s} \sum_{i=1}^{s} | f(x_i, w) - y_i |^2 + \frac{\alpha}{2} \|\mathbf{w}\|^2 \quad ,$$

where $f$ is linear in $w$, we have seen that we can compute $\hat{w}$ by solving a linear system of equations

# Why optimisation?

In the previous lectures, we have studied regression problems of the form

$$\hat{\mathbf{w}} = \arg \min_{\mathbf{w} \in \mathbb{R}^{d+1}} E(\mathbf{w})$$

For

$$E(\mathbf{w}) = \frac{1}{2s} \sum_{i=1}^{s} |f(x_i, w) - y_i|^2 + \frac{\alpha}{2} \|\mathbf{w}\|^2 \quad ,$$

where $f$ is linear in $w$, we have seen that we can compute $\hat{w}$ by solving a linear system of equations

But: how do we minimise $E$ in general?

# Grid search?

How about using grid search?

Evaluate a function $E$ at points on a grid and record smallest value

# Grid search?

How about using grid search?

Evaluate a function $E$ at points on a grid and record smallest value

Advantages:

# Grid search?

How about using grid search?

Evaluate a function $E$ at points on a grid and record smallest value

Advantages:

- works for any kind of function!

# Grid search?

How about using grid search?

Evaluate a function $E$ at points on a grid and record smallest value

Advantages:

- works for any kind of function!
- very easy to implement

# Grid search?

How about using grid search?

Evaluate a function $E$ at points on a grid and record smallest value

Advantages:

- works for any kind of function!
- very easy to implement

Disadvantages:

# Grid search?

How about using grid search?

Evaluate a function $E$ at points on a grid and record smallest value

Advantages:

- works for any kind of function!
- very easy to implement

Disadvantages:

- computationally infeasible for large no. of parameters

# Grid search?

How about using grid search?

Evaluate a function $E$ at points on a grid and record smallest value

Advantages:

- works for any kind of function!
- very easy to implement

Disadvantages:

- computationally infeasible for large no. of parameters
- no guarantee that we compute a minimum

# Smooth optimisation

Smooth functions (continuously differentiable) allow the application of more systematic searches compared to grid search

$$E \in C^1(\mathbb{R}^{d+1}) \qquad \Rightarrow \qquad \nabla E \text{ exists and is continuous}$$

# Smooth optimisation

Example for smooth optimisation: gradient descent

$$\mathbf{w}^{k+1} = \mathbf{w}^k - \tau \nabla E(\mathbf{w}^k)$$

for some $\mathbf{w}^0 \in \mathbb{R}^n$ and a constant $\tau > 0$ .

Procedure to find a minimum of w!

# Gradient descent

Gradient descent is an iterative procedure.
Let us remember that the gradient points the direction of max growth

# Gradient descent

Gradient descent is an iterative procedure.
Let us remember that the gradient points the direction of max growth

$$\mathbf{w}^1 = \mathbf{w}^0 - \tau \nabla E(\mathbf{w}^0)$$

# Gradient descent

Gradient descent is an iterative procedure.
Let us remember that the gradient points the direction of max growth

$$\mathbf{w}^1 = \mathbf{w}^0 - \tau \nabla E(\mathbf{w}^0)$$
$$\mathbf{w}^2 = \mathbf{w}^1 - \tau \nabla E(\mathbf{w}^1)$$

# Gradient descent

Gradient descent is an iterative procedure.
Let us remember that the gradient points the direction of max growth

$$\mathbf{w}^1 = \mathbf{w}^0 - \tau \nabla E(\mathbf{w}^0)$$
$$\mathbf{w}^2 = \mathbf{w}^1 - \tau \nabla E(\mathbf{w}^1)$$
$$= \mathbf{w}^0 - \tau \nabla E(\mathbf{w}^0) - \tau \nabla E(\mathbf{w}^0 - \tau \nabla E(\mathbf{w}^0))$$

# Gradient descent

Gradient descent is an iterative procedure.
Let us remember that the gradient points the direction of max growth

$$\mathbf{w}^1 = \mathbf{w}^0 - \tau \nabla E(\mathbf{w}^0)$$
$$\mathbf{w}^2 = \mathbf{w}^1 - \tau \nabla E(\mathbf{w}^1)$$
$$= \mathbf{w}^0 - \tau \nabla E(\mathbf{w}^0) - \tau \nabla E(\mathbf{w}^0 - \tau \nabla E(\mathbf{w}^0))$$
$$\mathbf{w}^3 = \mathbf{w}^2 - \tau \nabla E(\mathbf{w}^2)$$

# Gradient descent

Gradient descent is an iterative procedure.
Let us remember that the gradient points the direction of max growth

$$\begin{aligned}
\mathbf{w}^1 &= \mathbf{w}^0 - \tau \nabla E(\mathbf{w}^0) \\
\mathbf{w}^2 &= \mathbf{w}^1 - \tau \nabla E(\mathbf{w}^1) \\
&= \mathbf{w}^0 - \tau \nabla E(\mathbf{w}^0) - \tau \nabla E(\mathbf{w}^0 - \tau \nabla E(\mathbf{w}^0)) \\
\mathbf{w}^3 &= \mathbf{w}^2 - \tau \nabla E(\mathbf{w}^2) \\
&\vdots
\end{aligned}$$

# Gradient descent

Gradient descent is an iterative procedure.
Let us remember that the gradient points the direction of max growth

$$
\begin{aligned}
\mathbf{w}^1 &= \mathbf{w}^0 - \tau \nabla E(\mathbf{w}^0) \\
\mathbf{w}^2 &= \mathbf{w}^1 - \tau \nabla E(\mathbf{w}^1) \\
&= \mathbf{w}^0 - \tau \nabla E(\mathbf{w}^0) - \tau \nabla E(\mathbf{w}^0 - \tau \nabla E(\mathbf{w}^0)) \\
\mathbf{w}^3 &= \mathbf{w}^2 - \tau \nabla E(\mathbf{w}^2) \\
&\vdots \\
\mathbf{w}^k &= \mathbf{w}^{k-1} - \tau \nabla E(\mathbf{w}^{k-1})
\end{aligned}
$$

# Gradient descent

Gradient descent is an iterative procedure.
Let us remember that the gradient points the direction of max growth

$$\begin{aligned}
\mathbf{w}^1 &= \mathbf{w}^0 - \tau \nabla E(\mathbf{w}^0) \\
\mathbf{w}^2 &= \mathbf{w}^1 - \tau \nabla E(\mathbf{w}^1) \\
&= \mathbf{w}^0 - \tau \nabla E(\mathbf{w}^0) - \tau \nabla E(\mathbf{w}^0 - \tau \nabla E(\mathbf{w}^0)) \\
\mathbf{w}^3 &= \mathbf{w}^2 - \tau \nabla E(\mathbf{w}^2) \\
&\vdots \\
\mathbf{w}^k &= \mathbf{w}^{k-1} - \tau \nabla E(\mathbf{w}^{k-1})
\end{aligned}$$

Every step of the procedure is also known as an iterate or update

# Gradient descent

Gradient descent is an iterative procedure.
Let us remember that the gradient points the direction of max growth

$$\mathbf{w}^1 = \mathbf{w}^0 - \tau \nabla E(\mathbf{w}^0)$$

$$\mathbf{w}^2 = \mathbf{w}^1 - \tau \nabla E(\mathbf{w}^1)$$

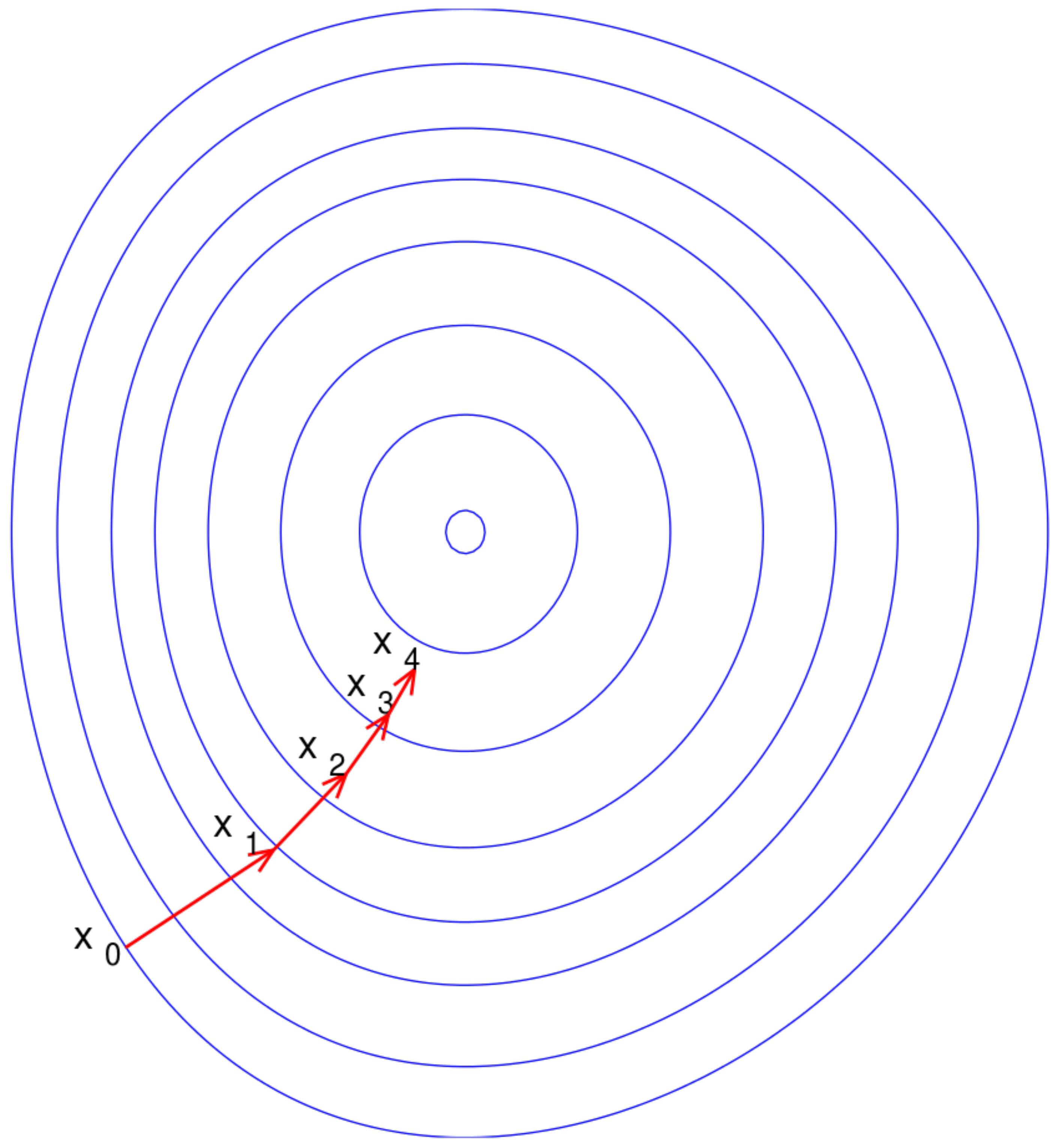
$$= \mathbf{w}^0 - \tau \nabla E(\mathbf{w}^0) - \tau \nabla E(\mathbf{w}^0 - \tau \nabla E(\mathbf{w}^0))$$

$$\mathbf{w}^3 = \mathbf{w}^2 - \tau \nabla E(\mathbf{w}^2)$$

$$\vdots$$

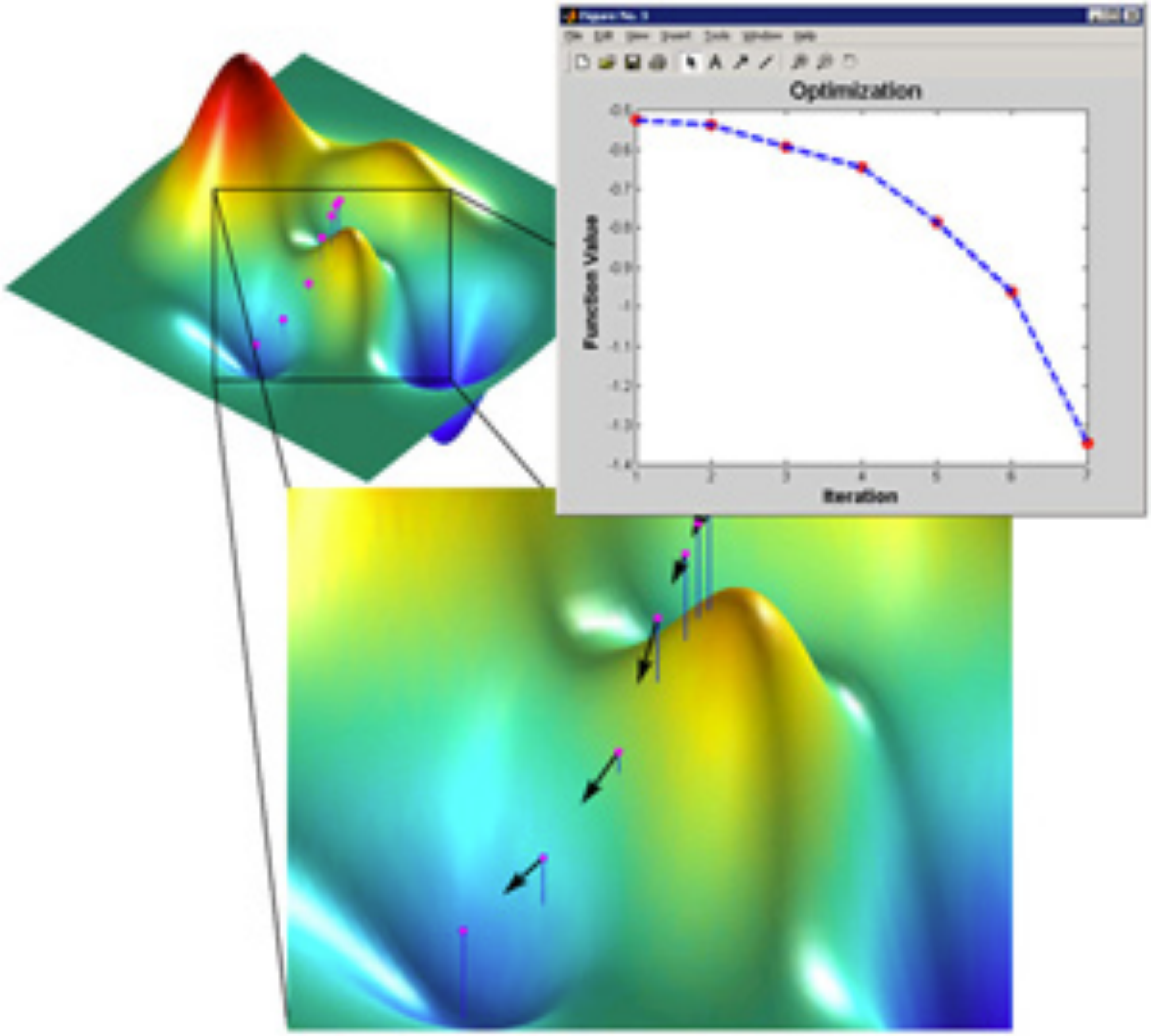
$$\mathbf{w}^k = \mathbf{w}^{k-1} - \tau \nabla E(\mathbf{w}^{k-1})$$

Every step of the procedure is also known as an iterate or update



# Gradient descent

Gradient descent is an iterative procedure

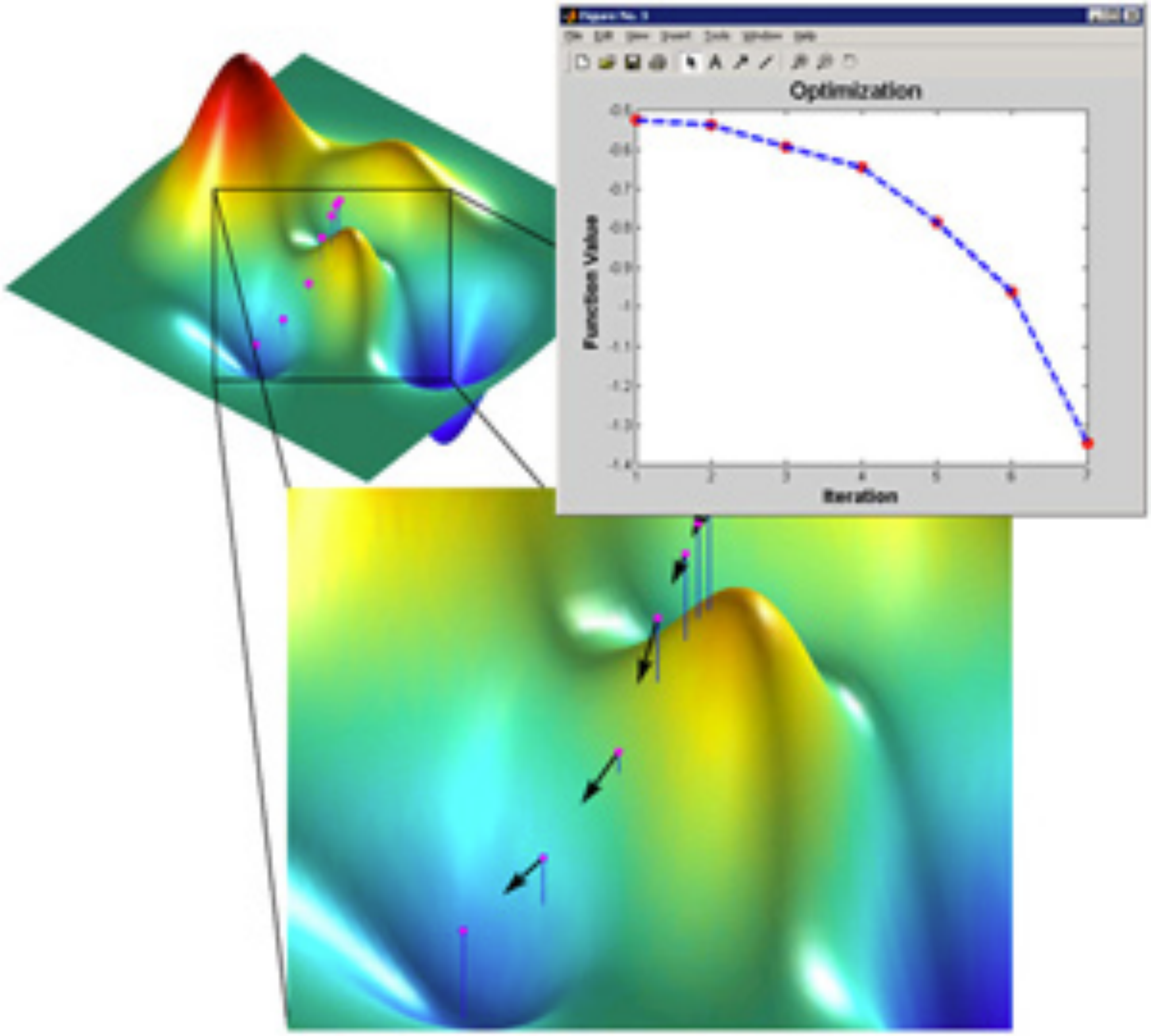



Every step of the procedure is also known as an iterate or update

# Gradient descent

Gradient descent is an iterative procedure

$$\mathbf{w}^1 = \mathbf{w}^0 - \tau \nabla E(\mathbf{w}^0)$$

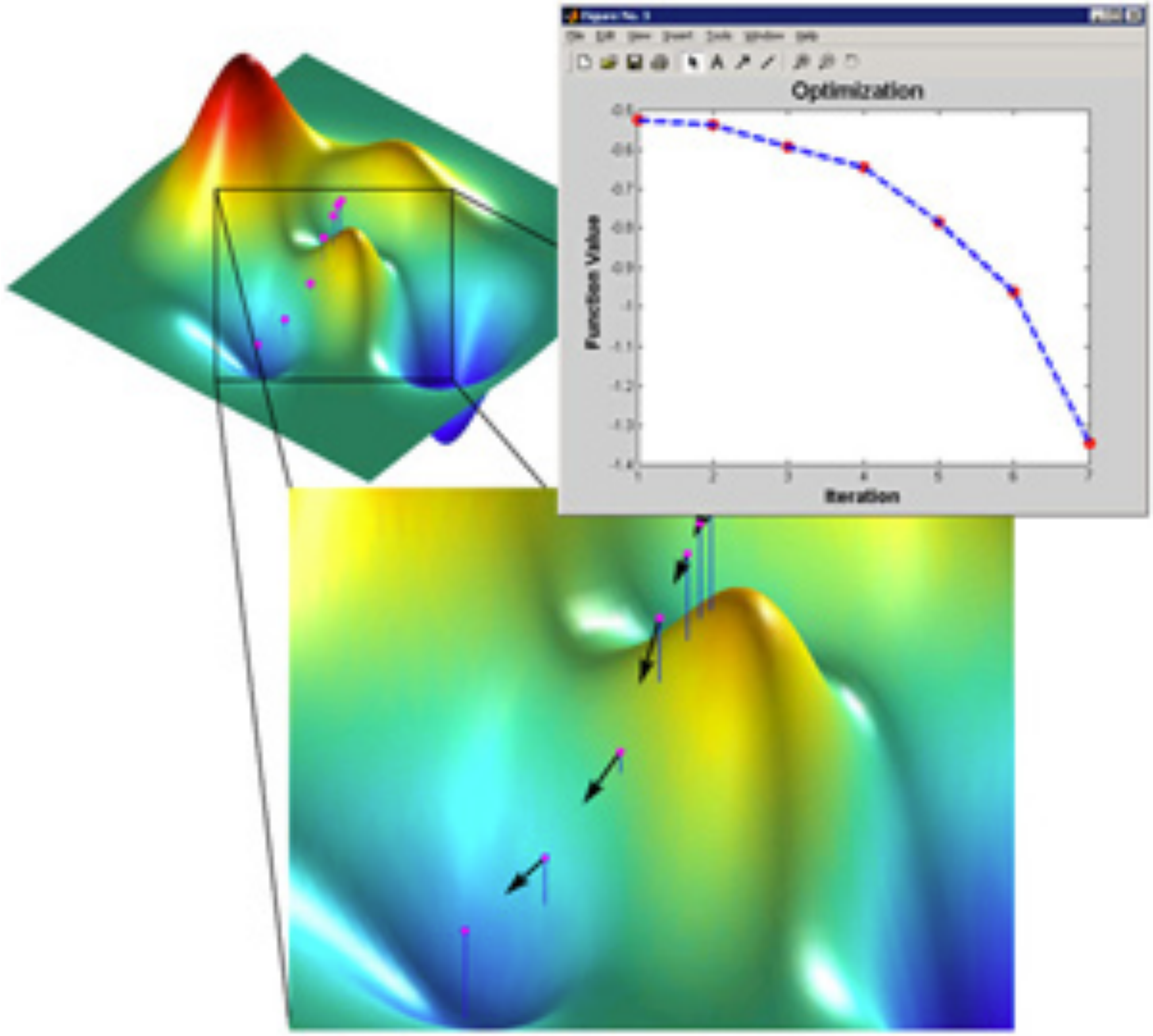



Every step of the procedure is also known as an iterate or update

# Gradient descent

Gradient descent is an iterative procedure

$$\mathbf{w}^1 = \mathbf{w}^0 - \tau \nabla E(\mathbf{w}^0)$$
$$\mathbf{w}^2 = \mathbf{w}^1 - \tau \nabla E(\mathbf{w}^1)$$




©Mathworks

Every step of the procedure is also known as an iterate or update

# Gradient descent

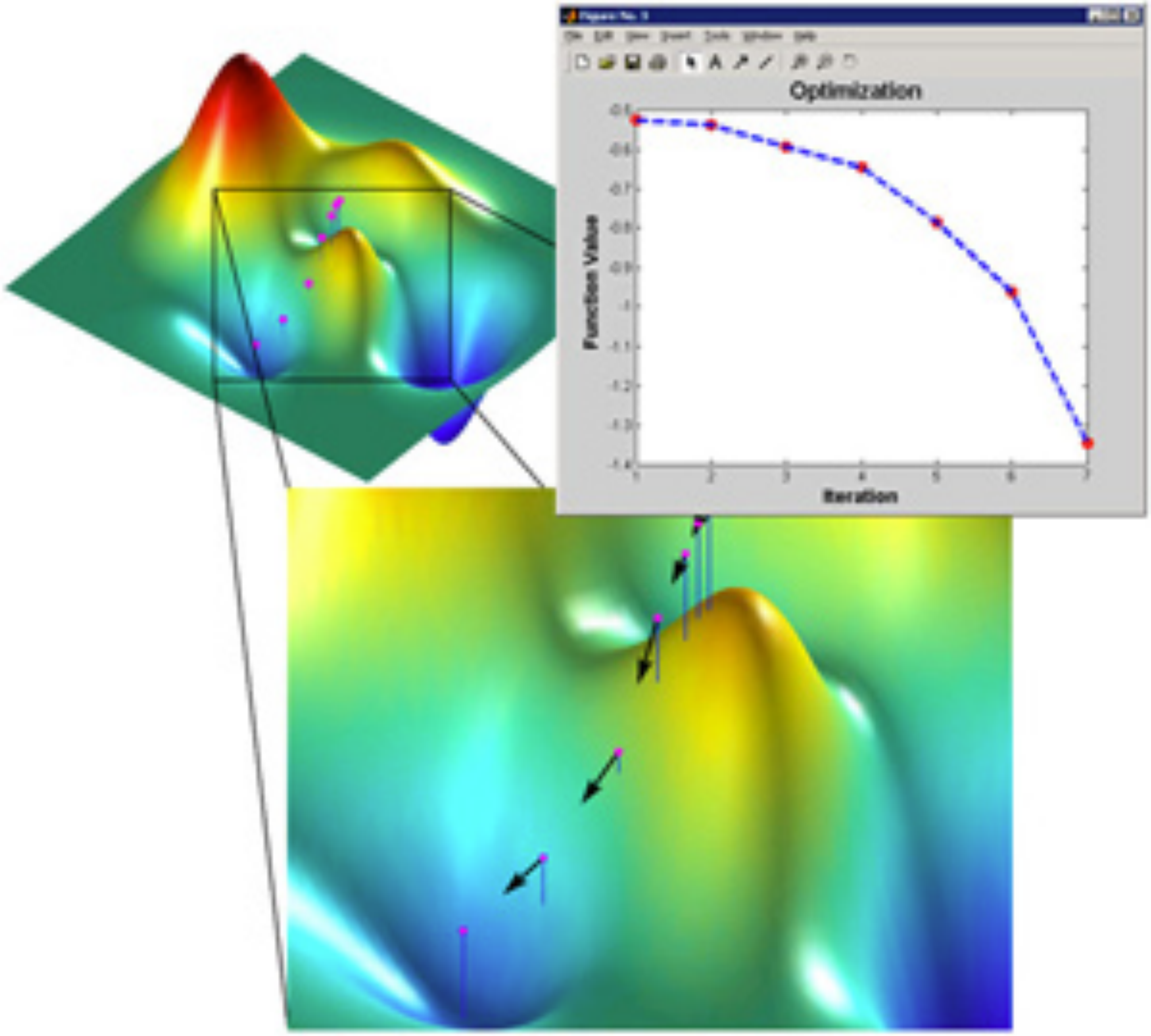
Gradient descent is an iterative procedure

$$\mathbf{w}^1 = \mathbf{w}^0 - \tau \nabla E(\mathbf{w}^0)$$

$$\mathbf{w}^2 = \mathbf{w}^1 - \tau \nabla E(\mathbf{w}^1)$$

$$= \mathbf{w}^0 - \tau \nabla E(\mathbf{w}^0) - \tau \nabla E(\mathbf{w}^0 - \tau \nabla E(\mathbf{w}^0))$$

Every step of the procedure is also known as an iterate or update




©Mathworks

# Gradient descent

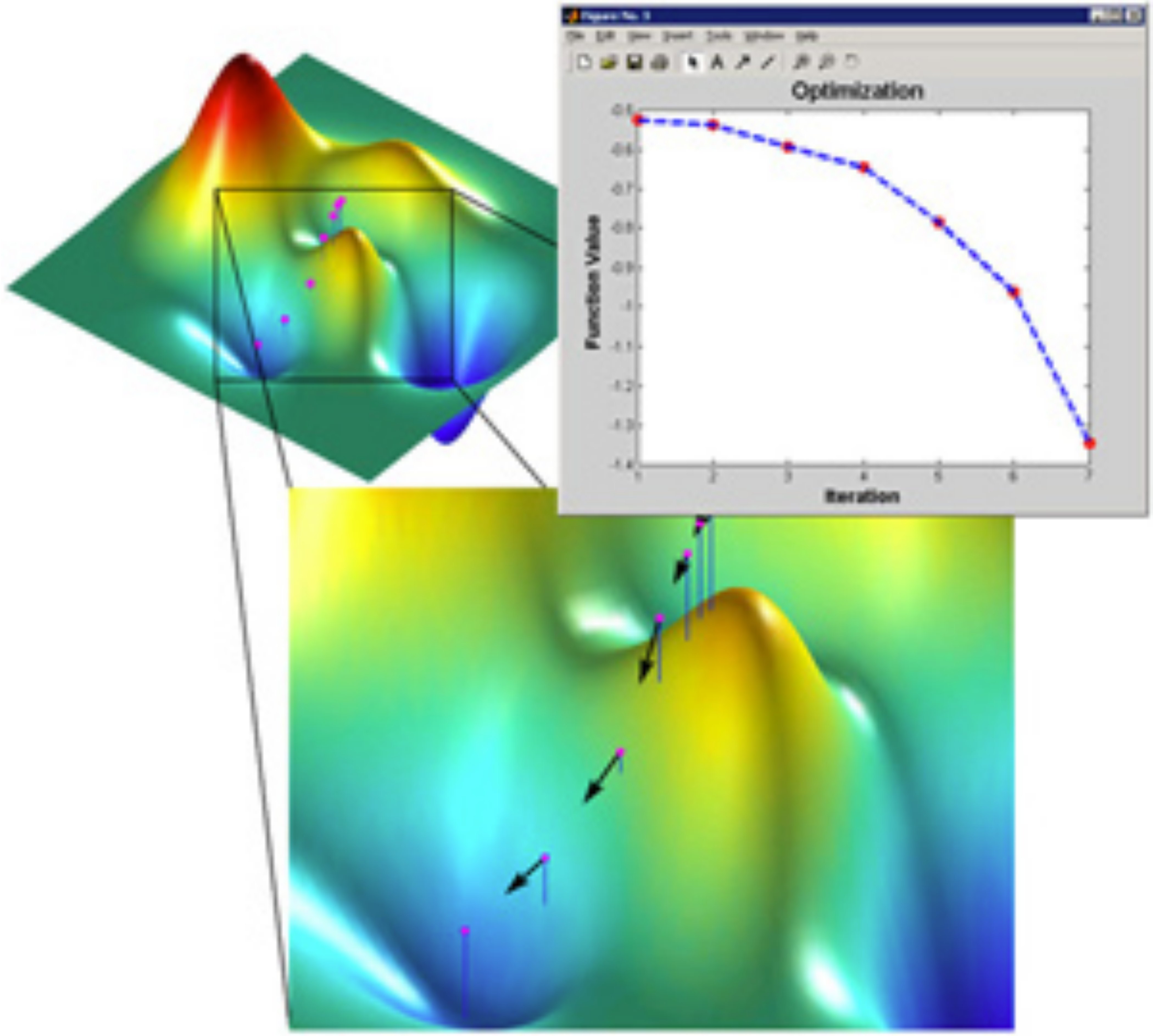
Gradient descent is an iterative procedure

$$\mathbf{w}^1 = \mathbf{w}^0 - \tau \nabla E(\mathbf{w}^0)$$

$$\mathbf{w}^2 = \mathbf{w}^1 - \tau \nabla E(\mathbf{w}^1)$$

$$= \mathbf{w}^0 - \tau \nabla E(\mathbf{w}^0) - \tau \nabla E(\mathbf{w}^0 - \tau \nabla E(\mathbf{w}^0))$$

$$\mathbf{w}^3 = \mathbf{w}^2 - \tau \nabla E(\mathbf{w}^2)$$




©Mathworks

Every step of the procedure is also known as an iterate or update

# Gradient descent

Gradient descent is an iterative procedure

$$\mathbf{w}^1 = \mathbf{w}^0 - \tau \nabla E(\mathbf{w}^0)$$

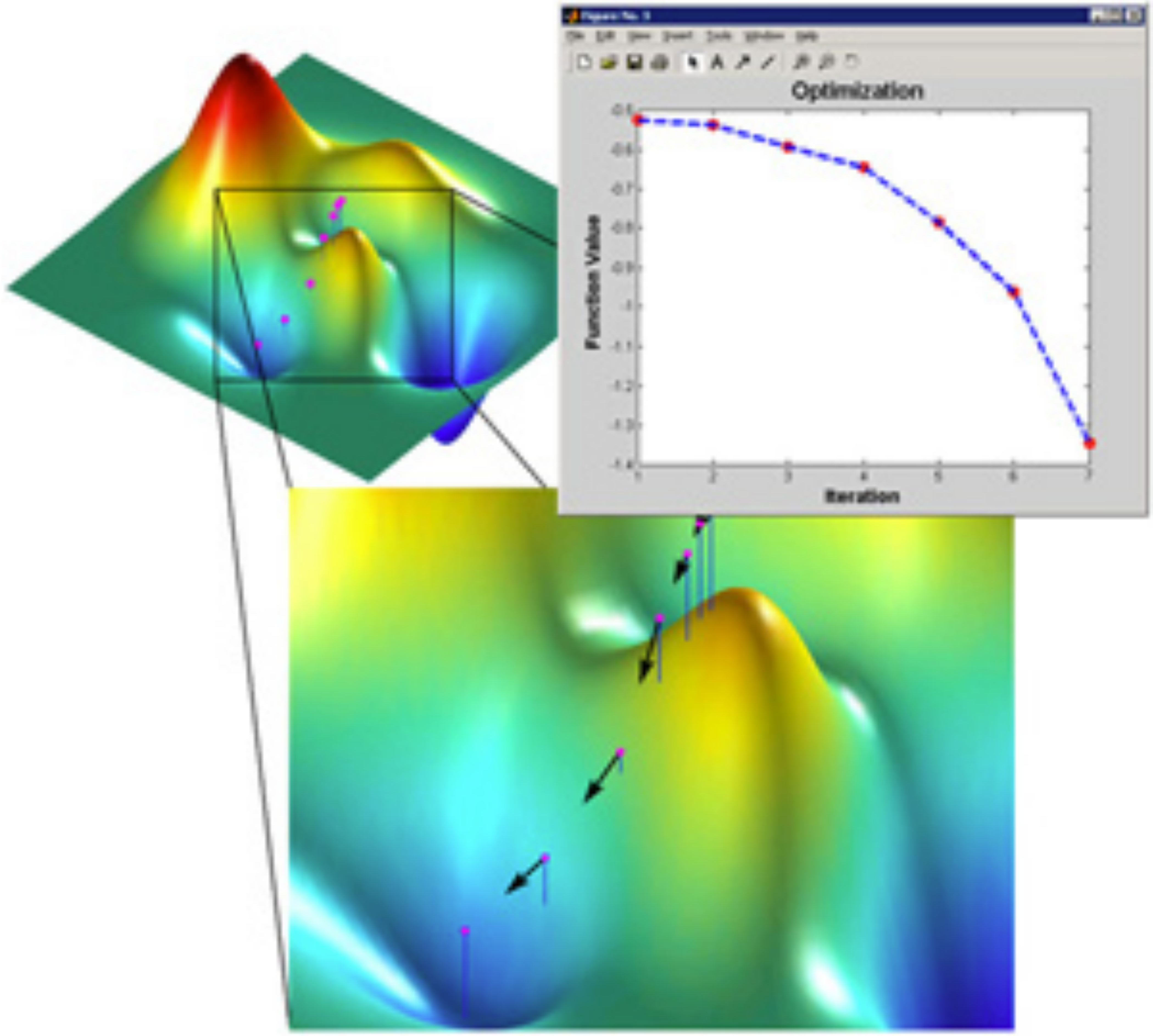
$$\mathbf{w}^2 = \mathbf{w}^1 - \tau \nabla E(\mathbf{w}^1)$$

$$= \mathbf{w}^0 - \tau \nabla E(\mathbf{w}^0) - \tau \nabla E(\mathbf{w}^0 - \tau \nabla E(\mathbf{w}^0))$$

$$\mathbf{w}^3 = \mathbf{w}^2 - \tau \nabla E(\mathbf{w}^2)$$

$$\vdots$$

Every step of the procedure is also known as an iterate or update



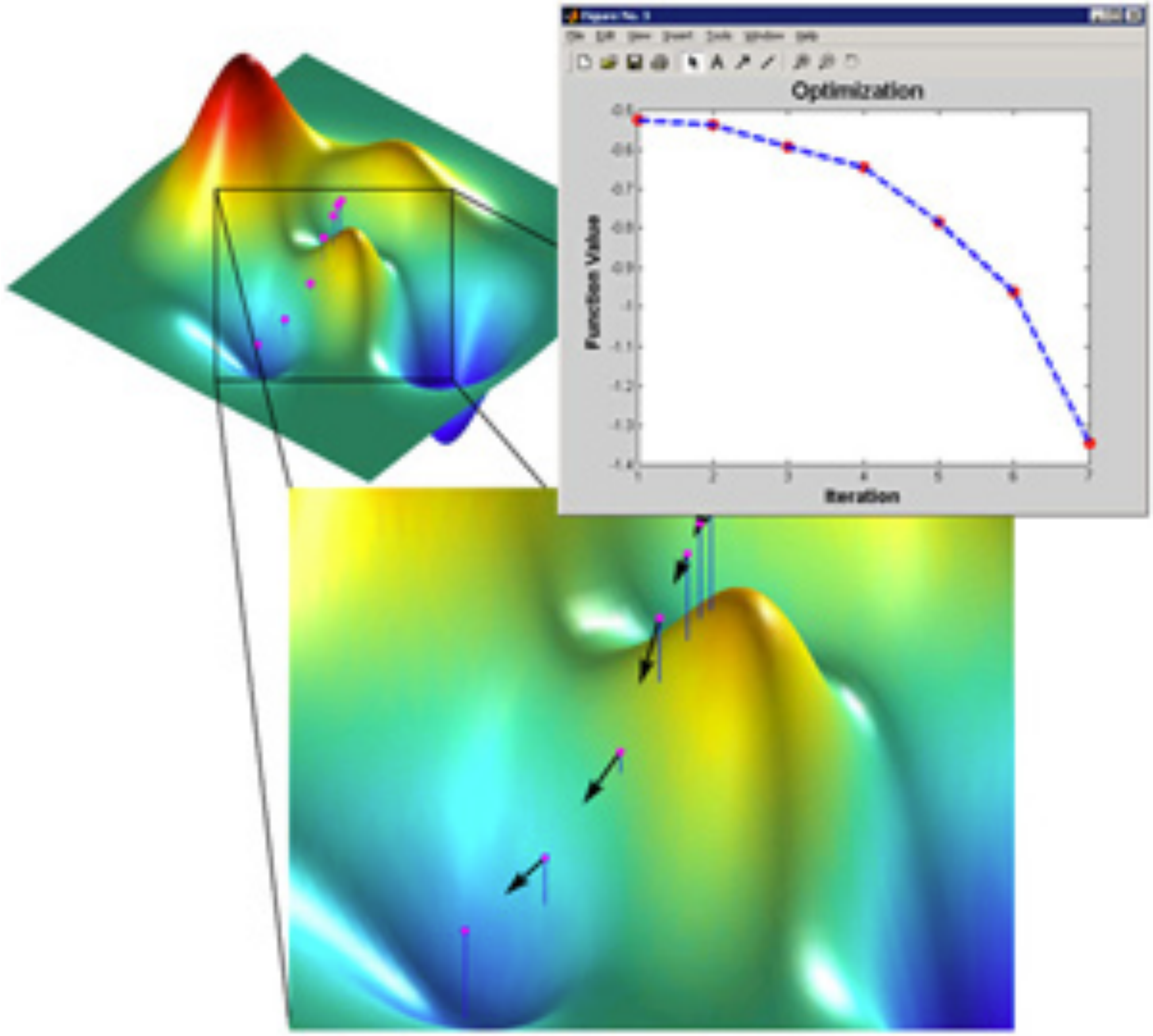

©Mathworks

# Gradient descent

Gradient descent is an iterative procedure

$$\mathbf{w}^1 = \mathbf{w}^0 - \tau \nabla E(\mathbf{w}^0)$$
$$\mathbf{w}^2 = \mathbf{w}^1 - \tau \nabla E(\mathbf{w}^1)$$
$$= \mathbf{w}^0 - \tau \nabla E(\mathbf{w}^0) - \tau \nabla E(\mathbf{w}^0 - \tau \nabla E(\mathbf{w}^0))$$
$$\mathbf{w}^3 = \mathbf{w}^2 - \tau \nabla E(\mathbf{w}^2)$$
$$\vdots$$
$$\mathbf{w}^k = \mathbf{w}^{k-1} - \tau \nabla E(\mathbf{w}^{k-1})$$

Every step of the procedure is also known as an iterate or update




©Mathworks

# Gradient descent: examples

One parameter MSE-model: $\text{MSE}(w) = \frac{1}{2s} \sum_{i=1}^{s} |w - y_i|^2$

# Gradient descent: examples

One parameter MSE-model:

$$\mathrm{MSE}(w) = \frac{1}{2s}\sum_{i=1}^{s} |w - y_i|^2$$

Gradient:

$$\nabla \mathrm{MSE}(w) = w - \frac{1}{s}\sum_{i=1}^{s} y_i$$

# Gradient descent: examples

One parameter MSE-model: $$\mathrm{MSE}(w) = \frac{1}{2s}\sum_{i=1}^{s} |w - y_i|^2$$

Gradient: $$\nabla \mathrm{MSE}(w) = w - \frac{1}{s}\sum_{i=1}^{s} y_i$$

We have learnt that $$\nabla \mathrm{MSE}(w) = w - \frac{1}{s}\sum_{i=1}^{s} y_i = 0 \rightarrow \hat{w} = \bar{y}$$

# Gradient descent: examples

# Gradient descent: examples

Gradient descent:

$$w^{k+1} = w^k - \tau \left( w^k - \frac{1}{s} \sum_{i=1}^{s} y_i \right) = (1 - \tau) w^k + \frac{\tau}{s} \sum_{i=1}^{s} y_i$$

# Gradient descent: examples

Gradient descent: $$w^{k+1} = w^k - \tau\left(w^k - \frac{1}{s}\sum_{i=1}^{s} y_i\right) = (1-\tau)w^k + \frac{\tau}{s}\sum_{i=1}^{s} y_i$$

For $\tau = 1$ $$w^{k+1} = \frac{1}{s}\sum_{i=1}^{s} y_i$$

# Gradient descent: examples

Gradient descent: $$w^{k+1} = w^k - \tau\left(w^k - \frac{1}{s}\sum_{i=1}^{s} y_i\right) = (1-\tau)w^k + \frac{\tau}{s}\sum_{i=1}^{s} y_i$$

For $\tau = 1$ $$w^{k+1} = \frac{1}{s}\sum_{i=1}^{s} y_i$$

For a general value of $\tau$?

# Gradient descent: examples

General linear MSE-model: $\text{MSE}(\mathbf{w}) = \frac{1}{2s} \| \mathbf{X}\mathbf{w} - \mathbf{y} \|^2$

# Gradient descent: examples

General linear MSE-model: $\text{MSE}(\mathbf{w}) = \frac{1}{2s} \| \mathbf{X}\mathbf{w} - \mathbf{y} \|^2$

Recall: $\nabla \text{MSE}(\mathbf{w}) = \frac{1}{s}\mathbf{X}^\top \left(\mathbf{X}\mathbf{w} - \mathbf{y}\right)$

# Gradient descent: examples

General linear MSE-model: $\text{MSE}(\mathbf{w}) = \frac{1}{2s} \| \mathbf{X}\mathbf{w} - \mathbf{y} \|^2$

Recall: $\nabla \text{MSE}(\mathbf{w}) = \frac{1}{s}\mathbf{X}^\top \left(\mathbf{X}\mathbf{w} - \mathbf{y}\right)$

Gradient descent: $\mathbf{w}^{k+1} = \mathbf{w}^k + \frac{\tau}{s}\mathbf{X}^\top(\mathbf{y} - \mathbf{X}\mathbf{w}^k)$

# Gradient descent: examples

General linear MSE-model: $\text{MSE}(\mathbf{w}) = \frac{1}{2s} \parallel \mathbf{X}\mathbf{w} - \mathbf{y} \parallel^2$

Recall: $\nabla \text{MSE}(\mathbf{w}) = \frac{1}{s}\mathbf{X}^\top \left(\mathbf{X}\mathbf{w} - \mathbf{y}\right)$

Gradient descent:

$$\mathbf{w}^{k+1} = \mathbf{w}^k + \frac{\tau}{s}\mathbf{X}^\top(\mathbf{y} - \mathbf{X}\mathbf{w}^k)$$

$$= \left(I - \frac{\tau}{s}\mathbf{X}^\top\mathbf{X}\right)\mathbf{w}^k + \frac{\tau}{s}\mathbf{X}^T\mathbf{y}$$

# Gradient descent: examples

General linear MSE-model: $\text{MSE}(\mathbf{w}) = \frac{1}{2s} \parallel \mathbf{X}\mathbf{w} - \mathbf{y} \parallel^2$

Recall: $\nabla \text{MSE}(\mathbf{w}) = \frac{1}{s}\mathbf{X}^\top \left(\mathbf{X}\mathbf{w} - \mathbf{y}\right)$

Gradient descent:

$$\mathbf{w}^{k+1} = \mathbf{w}^k + \frac{\tau}{s}\mathbf{X}^\top(\mathbf{y} - \mathbf{X}\mathbf{w}^k)$$

$$= \left(I - \frac{\tau}{s}\mathbf{X}^\top\mathbf{X}\right)\mathbf{w}^k + \frac{\tau}{s}\mathbf{X}^T\mathbf{y} \qquad \xrightarrow{k\to\infty} \left(\mathbf{X}^\top\mathbf{X}\right)^{-1}\mathbf{X}^\top\mathbf{y}$$

# Gradient descent: examples

General linear MSE-model: $\text{MSE}(\mathbf{w}) = \frac{1}{2s} \parallel \mathbf{X}\mathbf{w} - \mathbf{y} \parallel^2$

Recall: $\nabla \text{MSE}(\mathbf{w}) = \frac{1}{s}\mathbf{X}^\top \left(\mathbf{X}\mathbf{w} - \mathbf{y}\right)$

Gradient descent:

$$\mathbf{w}^{k+1} = \mathbf{w}^k + \frac{\tau}{s}\mathbf{X}^\top(\mathbf{y} - \mathbf{X}\mathbf{w}^k)$$

$$= \left(I - \frac{\tau}{s}\mathbf{X}^\top\mathbf{X}\right)\mathbf{w}^k + \frac{\tau}{s}\mathbf{X}^T\mathbf{y} \qquad \xrightarrow{k\to\infty} \left(\mathbf{X}^\top\mathbf{X}\right)^{-1}\mathbf{X}^\top\mathbf{y}$$

Does this work for any $\tau$?

# Gradient descent

Why (and when) does it work?

# Gradient descent

Why (and when) does it work?

Assumption: $E$ is Lipschitz-continuous with constant $L$ (or L-smooth), i.e.

$$\|\nabla E(\mathbf{x}) - \nabla E(\mathbf{y})\| \leq L\|\mathbf{x} - \mathbf{y}\| \qquad \forall x, y \in \mathbb{R}^n$$

# Gradient descent

Why (and when) does it work?

Assumption: $E$ is Lipschitz-continuous with constant $L$ (or L-smooth), i.e.

$$\|\nabla E(\mathbf{x}) - \nabla E(\mathbf{y})\| \leq L\|\mathbf{x} - \mathbf{y}\| \qquad \forall x, y \in \mathbb{R}^n$$

Then the function

$$G(x) := \frac{L}{2}\|\mathbf{x}\|^2 - E(\mathbf{x})$$

is convex for all $\mathbf{x} \in \mathbb{R}^n$.

# Gradient descent

Why (and when) does it work?

Assumptions:
- the function $E$ is $\tau^{-1}$ smooth
- the function $G(\mathbf{w}) := \frac{1}{2\tau}||\mathbf{w}||^2 - E(\mathbf{w})$ is convex

for all $\mathbf{w} \in \mathbb{R}^n$

# Gradient descent

Why (and when) does it work?

Assumptions:
- the function $E$ is $\tau^{-1}$ smooth
- the function $G(\mathbf{w}) := \frac{1}{2\tau}\|\mathbf{w}\|^2 - E(\mathbf{w})$ is convex

  for all $\mathbf{w} \in \mathbb{R}^n$

Then (converge theorem) we can show

1. that $E(\mathbf{w}^{k+1}) \leq E(\mathbf{w}^k)$
2. as well as $\lim_{k\to\infty} E(\mathbf{w}^k) = E(\hat{\mathbf{w}})$ with rate $1/k$

# Gradient descent

Why (and when) does it work?

Assumptions:
- the function $E$ is $\tau^{-1}$ smooth
- the function $G(\mathbf{w}) := \frac{1}{2\tau}\|\mathbf{w}\|^2 - E(\mathbf{w})$ is convex

  for all $\mathbf{w} \in \mathbb{R}^n$

Then (converge theorem) we can show

1. that $E(\mathbf{w}^{k+1}) \leq E(\mathbf{w}^k)$
2. as well as $\lim_{k\to\infty} E(\mathbf{w}^k) = E(\hat{\mathbf{w}})$ with rate $1/k$

Proof:

# Gradient descent

Why (and when) does it work?

Assumptions:
- the function $E$ is $\tau^{-1}$ smooth
- the function $G(\mathbf{w}) := \frac{1}{2\tau}\|\mathbf{w}\|^2 - E(\mathbf{w})$ is convex

  for all $\mathbf{w} \in \mathbb{R}^n$

Then (converge theorem) we can show

1. that $E(\mathbf{w}^{k+1}) \leq E(\mathbf{w}^k)$
2. as well as $\lim_{k\to\infty} E(\mathbf{w}^k) = E(\hat{\mathbf{w}})$ with rate $1/k$

Proof: in the lecture notes, but not examinable! 😉

# Gradient descent: examples

What is the value of $\tau$ that allows convergence?

# Gradient descent: examples

What is the value of $\tau$ that allows convergence?

$$E(\mathbf{w}) = \frac{1}{2s}\|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2 \rightarrow \nabla E(\mathbf{w}) = \frac{1}{s}\mathbf{X}^\top(\mathbf{X}\mathbf{w} - \mathbf{y})$$

# Gradient descent: examples

What is the value of $\tau$ that allows convergence?

$$E(\mathbf{w}) = \frac{1}{2s}\|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2 \rightarrow \nabla E(\mathbf{w}) = \frac{1}{s}\mathbf{X}^\top(\mathbf{X}\mathbf{w} - \mathbf{y})$$

$$\|\nabla E(\mathbf{w}) - \nabla E(\mathbf{v})\| = \frac{1}{s}\|\mathbf{X}^\top\mathbf{X}(\mathbf{w} - \mathbf{v})\|$$

# Gradient descent: examples

What is the value of $\tau$ that allows convergence?

$$E(\mathbf{w}) = \frac{1}{2s}\|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2 \rightarrow \nabla E(\mathbf{w}) = \frac{1}{s}\mathbf{X}^\top(\mathbf{X}\mathbf{w} - \mathbf{y})$$

$$\|\nabla E(\mathbf{w}) - \nabla E(\mathbf{v})\| = \frac{1}{s}\|\mathbf{X}^\top\mathbf{X}(\mathbf{w} - \mathbf{v})\| \leq \frac{1}{s}\|\mathbf{X}^\top\mathbf{X}\|\,\|(\mathbf{w} - \mathbf{v})\|$$

# Gradient descent: examples

What is the value of $\tau$ that allows convergence?

$$E(\mathbf{w}) = \frac{1}{2s}\|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2 \rightarrow \nabla E(\mathbf{w}) = \frac{1}{s}\mathbf{X}^\top(\mathbf{X}\mathbf{w} - \mathbf{y})$$

$$\|\nabla E(\mathbf{w}) - \nabla E(\mathbf{v})\| = \frac{1}{s}\|\mathbf{X}^\top\mathbf{X}(\mathbf{w} - \mathbf{v})\| \leq \frac{1}{s}\|\mathbf{X}^\top\mathbf{X}\|\,\|(\mathbf{w} - \mathbf{v})\|$$

Hence the function is $\tau^{-1}$ smooth and converge is guaranteed for $\dfrac{1}{\tau} = \dfrac{\|\mathbf{X}^\top\mathbf{X}\|}{s}$

# Gradient descent: examples

What is the value of $\tau$ that allows convergence?

$$E(\mathbf{w}) = \frac{1}{2s}\|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2 \to \nabla E(\mathbf{w}) = \frac{1}{s}\mathbf{X}^\top(\mathbf{X}\mathbf{w} - \mathbf{y})$$

$$\|\nabla E(\mathbf{w}) - \nabla E(\mathbf{v})\| = \frac{1}{s}\|\mathbf{X}^\top\mathbf{X}(\mathbf{w} - \mathbf{v})\| \leq \frac{1}{s}\|\mathbf{X}^\top\mathbf{X}\|\,\|(\mathbf{w} - \mathbf{v})\|$$

Hence the function is $\tau^{-1}$ smooth and converge is guaranteed for $\dfrac{1}{\tau} = \dfrac{\|\mathbf{X}^\top\mathbf{X}\|}{s}$

This implies convergence for any $\tau \leq \dfrac{s}{\|\mathbf{X}^\top\mathbf{X}\|}$

# Gradient descent

Assumptions:

- the function $E$ is $\tau^{-1}$ smooth
- the function $G(\mathbf{w}) := \frac{1}{2\tau}\|\mathbf{w}\|^2 - E(\mathbf{w})$ is convex

  for all $\mathbf{w} \in \mathbb{R}^n$

What can we do if the assumptions are not met?

# Gradient descent

Assumptions:
- the function $E$ is $\tau^{-1}$ smooth
- the function $G(\mathbf{w}) := \frac{1}{2\tau}\|\mathbf{w}\|^2 - E(\mathbf{w})$ is convex

  for all $\mathbf{w} \in \mathbb{R}^n$

What can we do if the assumptions are not met?

Backtracking:

# Gradient descent

Assumptions:

- the function $E$ is $\tau^{-1}$ smooth
- the function $G(\mathbf{w}) := \frac{1}{2\tau}\|\mathbf{w}\|^2 - E(\mathbf{w})$ is convex

  for all $\mathbf{w} \in \mathbb{R}^n$

What can we do if the assumptions are not met?

Backtracking: compute $\mathbf{w}^{k+1}$ and check $E(\mathbf{w}^{k+1}) \leq E(\mathbf{w}^k)$

$$\begin{cases} \text{keep } \tau \text{ as it is} & \text{if } E(\mathbf{w}^{k+1}) \leq E(\mathbf{w}^k) \\ \text{decrease } \tau & \text{if } E(\mathbf{w}^{k+1}) > E(\mathbf{w}^k) \end{cases}$$

# Gradient descent

Remark: in the (modern) machine learning literature…

# Gradient descent

Remark: in the (modern) machine learning literature…

…gradient descent is also known as batch gradient descent

# Gradient descent

Remark: in the (modern) machine learning literature…

…gradient descent is also known as batch gradient descent

…the stepsize $\tau$ is also known as the learning rate (bad name)

# SOLVING LASSO

# LASSO

How can we solve the LASSO computationally?

$$\mathbf{w}_\alpha = \arg\min_{\mathbf{w}} \left\{ \frac{1}{2}\|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2 + \alpha\|\mathbf{w}\|_1 \right\}$$

# LASSO

How can we solve the LASSO computationally?

$$\mathbf{w}_\alpha = \arg\min_{\mathbf{w}} \left\{ \frac{1}{2} \|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2 + \alpha \|\mathbf{w}\|_1 \right\}$$

Can we just compute $\nabla E(\mathbf{w}_\alpha) = 0$ for $E(\mathbf{w}) := \|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2/2 + \alpha\|\mathbf{w}\|_1$?

# LASSO

How can we solve the LASSO computationally?

$$\mathbf{w}_\alpha = \arg\min_{\mathbf{w}} \left\{ \frac{1}{2} \|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2 + \alpha \|\mathbf{w}\|_1 \right\}$$

Can we just compute $\nabla E(\mathbf{w}_\alpha) = 0$ for $E(\mathbf{w}) := \|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2/2 + \alpha\|\mathbf{w}\|_1$?

We cannot do this, since $E$ is not differentiable!

# LASSO

How can we solve the LASSO computationally?

$$\mathbf{w}_\alpha = \arg\min_{\mathbf{w}} \left\{ \frac{1}{2} \|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2 + \alpha \|\mathbf{w}\|_1 \right\}$$

Can we use the same machinery we developed for the other problems?

# LASSO

No!

$$\mathbf{w}_\alpha = \arg\min_{\mathbf{w}} \left\{ \frac{1}{2}\|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2 + \alpha\|\mathbf{w}\|_1 \right\}$$

The l1 norm is not differentiable in zero

# LASSO

No!

$$\mathbf{w}_\alpha = \arg\min_{\mathbf{w}} \left\{ \frac{1}{2} \|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2 + \alpha \|\mathbf{w}\|_1 \right\}$$

The l1 norm is not differentiable in zero

We can smooth the one-norm to make this problem differentiable!

# LASSO

No!

$$\mathbf{w}_\alpha = \arg\min_{\mathbf{w}} \left\{ \frac{1}{2}\|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2 + \alpha\|\mathbf{w}\|_1 \right\}$$

The l1 norm is not differentiable in zero

We can smooth the one-norm to make this problem differentiable!

Note that we can write

$$|\mathbf{w}| = \max_{p\in[-1,1]} \mathbf{w}p$$

# LASSO

How can we solve the LASSO computationally?

$$\mathbf{w}_\alpha = \arg\min_{\mathbf{w}} \left\{ \frac{1}{2}\|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2 + \alpha\|\mathbf{w}\|_1 \right\}$$

We can smooth the one-norm to make this problem differentiable!

We can modify slightly the l1 norm to smooth the function

$$|\mathbf{w}|_\tau = \max_{p\in[-1,1]} wp \quad - \quad \frac{\tau}{2}|p|^2$$

# LASSO

Note that we can write

$$|\mathbf{w}|_{\tau} = \max_{p \in [-1,1]} wp \quad - \quad \frac{\tau}{2}|p|^2$$

# LASSO

Note that we can write

$$|\mathbf{w}|_{\tau} = \max_{p\in[-1,1]} wp \quad - \quad \frac{\tau}{2}|p|^2$$

This problem has a closed form solution

$$\hat{p} = \arg\max_{p\in[-1,1]} wp - \frac{\tau}{2}|p|^2$$

# LASSO

Note that we can write

$$|\mathbf{w}|_{\tau} = \max_{p\in[-1,1]} wp \quad - \quad \frac{\tau}{2}|p|^2$$

This problem has a closed form solution

$$\hat{p} = \arg\max_{p\in[-1,1]} wp - \frac{\tau}{2}|p|^2$$

$$\Leftrightarrow \qquad \hat{p} = \begin{cases} 1 & w > \tau \\ \frac{w}{\tau} & |w| \leq \tau \\ -1 & w < -\tau \end{cases}$$

# LASSO

Note that we can write

$$|\mathbf{w}|_{\tau} = \max_{p\in[-1,1]} wp \quad - \quad \frac{\tau}{2}|p|^2$$

This problem has a closed form solution

$$\hat{p} = \arg\max_{p\in[-1,1]} wp - \frac{\tau}{2}|p|^2$$

$$\Leftrightarrow \qquad \hat{p} = \begin{cases} 1 & w > \tau \\ \frac{w}{\tau} & |w| \leq \tau \\ -1 & w < -\tau \end{cases}$$

Why?

# LASSO

We need to solve

$$|\mathbf{w}|_\tau = \max_{p\in[-1,1]} wp \quad - \quad \frac{\tau}{2}|p|^2$$

# LASSO

We need to solve

$$|\mathbf{w}|_\tau = \max_{p\in[-1,1]} wp \quad - \quad \frac{\tau}{2}|p|^2$$

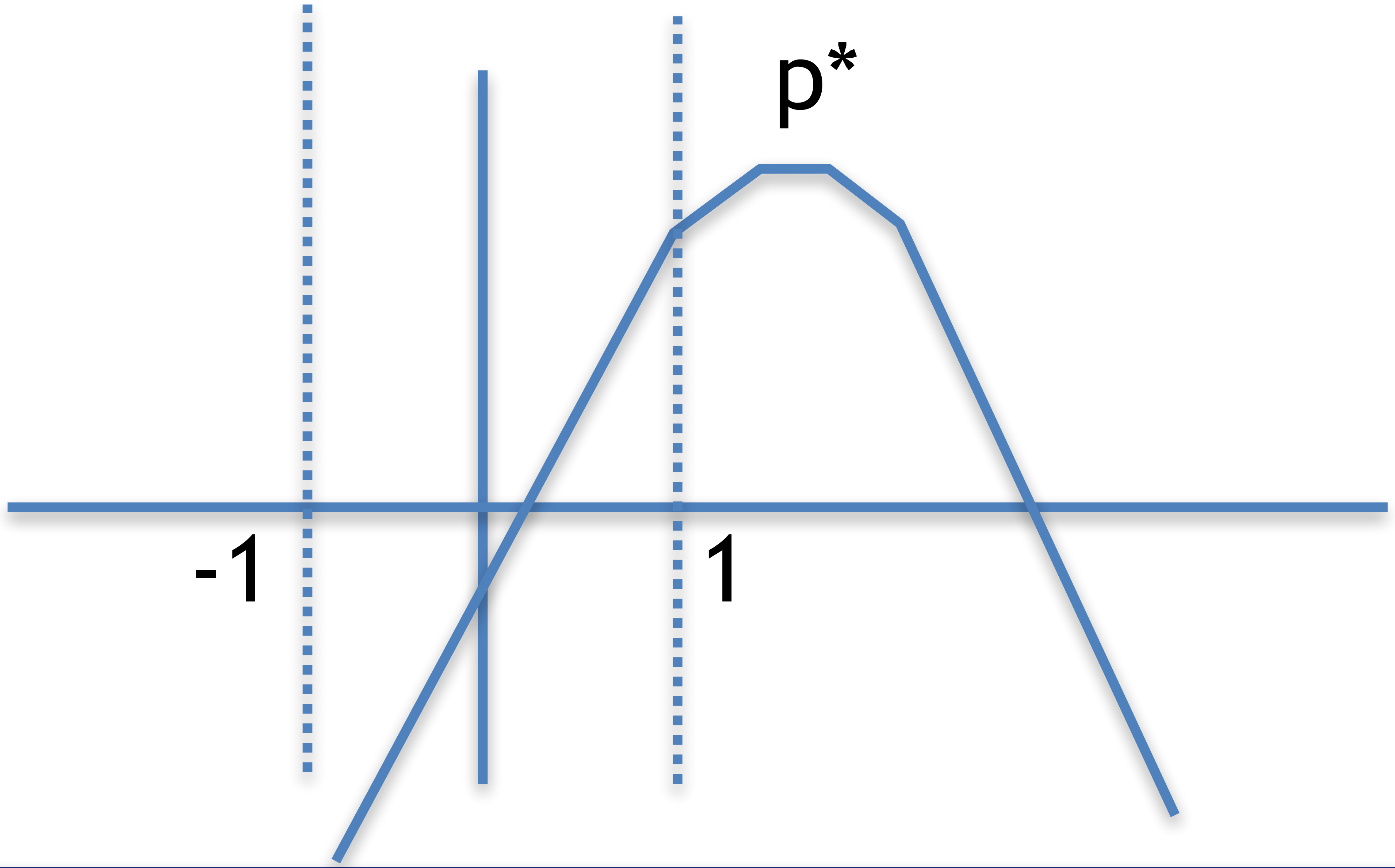
The function we are trying to maximize is a parabola of this type

# LASSO

We need to solve

$$|\mathbf{w}|_{\tau} = \max_{p\in[-1,1]} wp \quad - \quad \frac{\tau}{2}|p|^2$$

The function we are trying to maximize is a parabola of this type




p is bounded by -1 and 1

# LASSO

How do we get the max?

$$|\mathbf{w}|_\tau = \max_{p\in[-1,1]} wp \quad - \quad \frac{\tau}{2}|p|^2$$

# LASSO

How do we get the max?

$$|\mathbf{w}|_\tau = \max_{p\in[-1,1]} wp \quad - \quad \frac{\tau}{2}|p|^2$$

Compute the gradient!

# LASSO

How do we get the max?

$$|\mathbf{w}|_\tau = \max_{p\in[-1,1]} wp \quad - \quad \frac{\tau}{2}|p|^2$$

Compute the gradient!

$$\nabla|\mathbf{w}|_\tau = w - \tau p \quad \rightarrow \hat{p} = \frac{w}{\tau}$$

# LASSO

How do we get the max?

$$|\mathbf{w}|_\tau = \max_{p\in[-1,1]} wp \quad - \quad \frac{\tau}{2}|p|^2$$

Compute the gradient!

$$\nabla|\mathbf{w}|_\tau = w - \tau p \qquad \to \hat{p} = \frac{w}{\tau}$$

$$1 \le \hat{p} \le 1 \to -\tau \le w \le \tau$$

# LASSO

How do we get the max?

$$|\mathbf{w}|_\tau = \max_{p\in[-1,1]} wp \quad - \quad \frac{\tau}{2}|p|^2$$

Compute the gradient!

$$\nabla|\mathbf{w}|_\tau = w - \tau p \qquad \rightarrow \hat{p} = \frac{w}{\tau}$$

$$1 \leq \hat{p} \leq 1 \rightarrow -\tau \leq w \leq \tau$$

$$|w| \leq \tau$$

# LASSO

Hence, for $|w| \leq \tau$ the max is obtained substituting p hat in the expression

Hence

$$|\mathbf{w}|_{\tau} = \frac{w^2}{2\tau}$$

# LASSO

Hence, for $|w| \leq \tau$ the max is obtained substituting p hat in the expression

$$|\mathbf{w}| = \max_{p\in[-1,1]} wp - \frac{\tau}{2}|p|^2$$

Hence

$$|\mathbf{w}|_\tau = \frac{w^2}{2\tau}$$

# LASSO

Hence, for $|w| \leq \tau$ the max is obtained substituting p hat in the expression

$$|\mathbf{w}| = \max_{p \in [-1,1]} wp - \frac{\tau}{2}|p|^2 = w\frac{w}{\tau} - \frac{\tau}{2}\frac{w^2}{\tau^2} = \frac{w^2}{2\tau}$$

Hence

$$|\mathbf{w}|_\tau = \frac{w^2}{2\tau}$$

# LASSO

For $w > \tau$ instead $\hat{p} > 1$

# LASSO

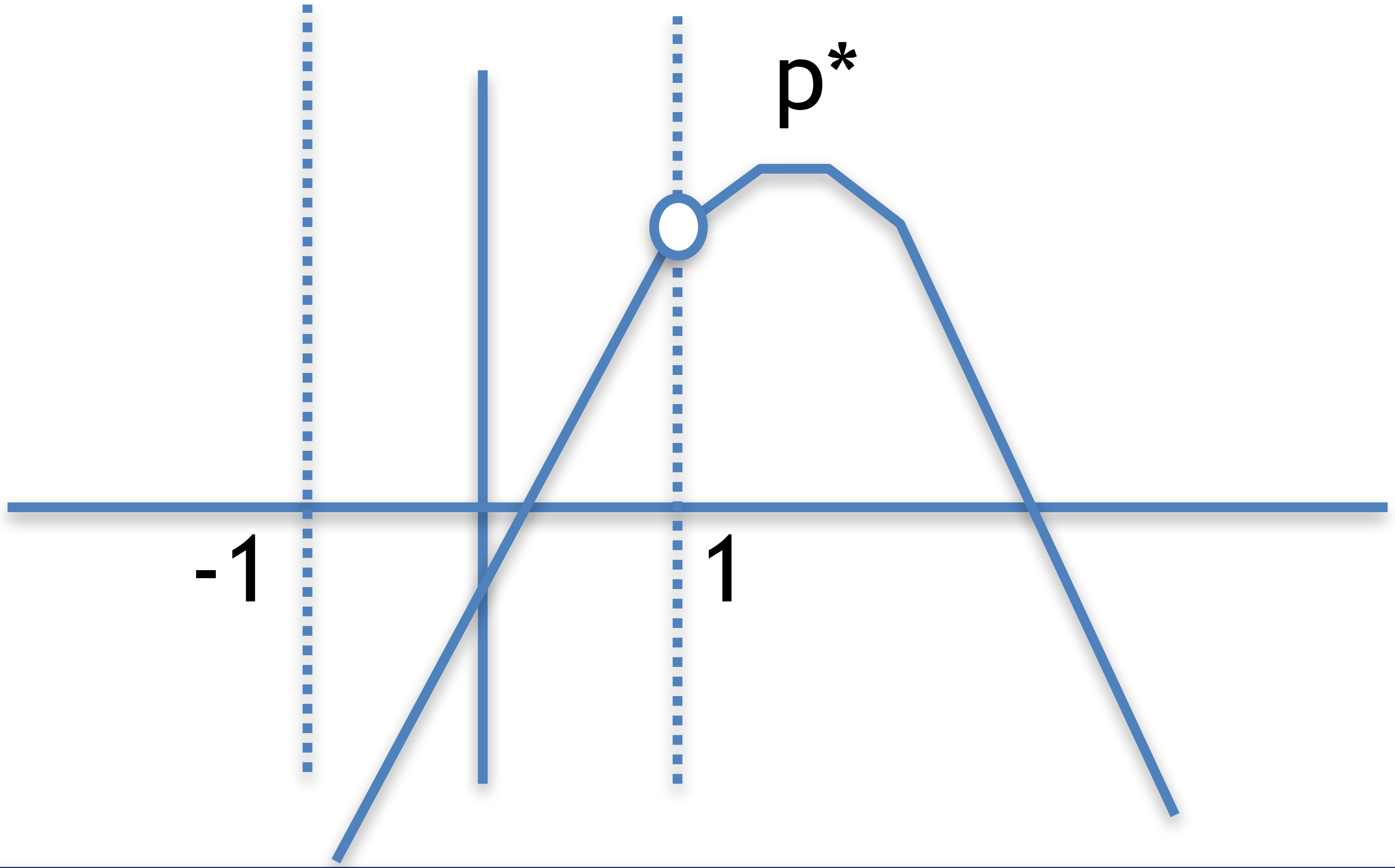
For $w > \tau$ instead $\hat{p} > 1$

If the max is larger than one than the parabola is indeed like this

# LASSO

For $w > \tau$ instead $\hat{p} > 1$

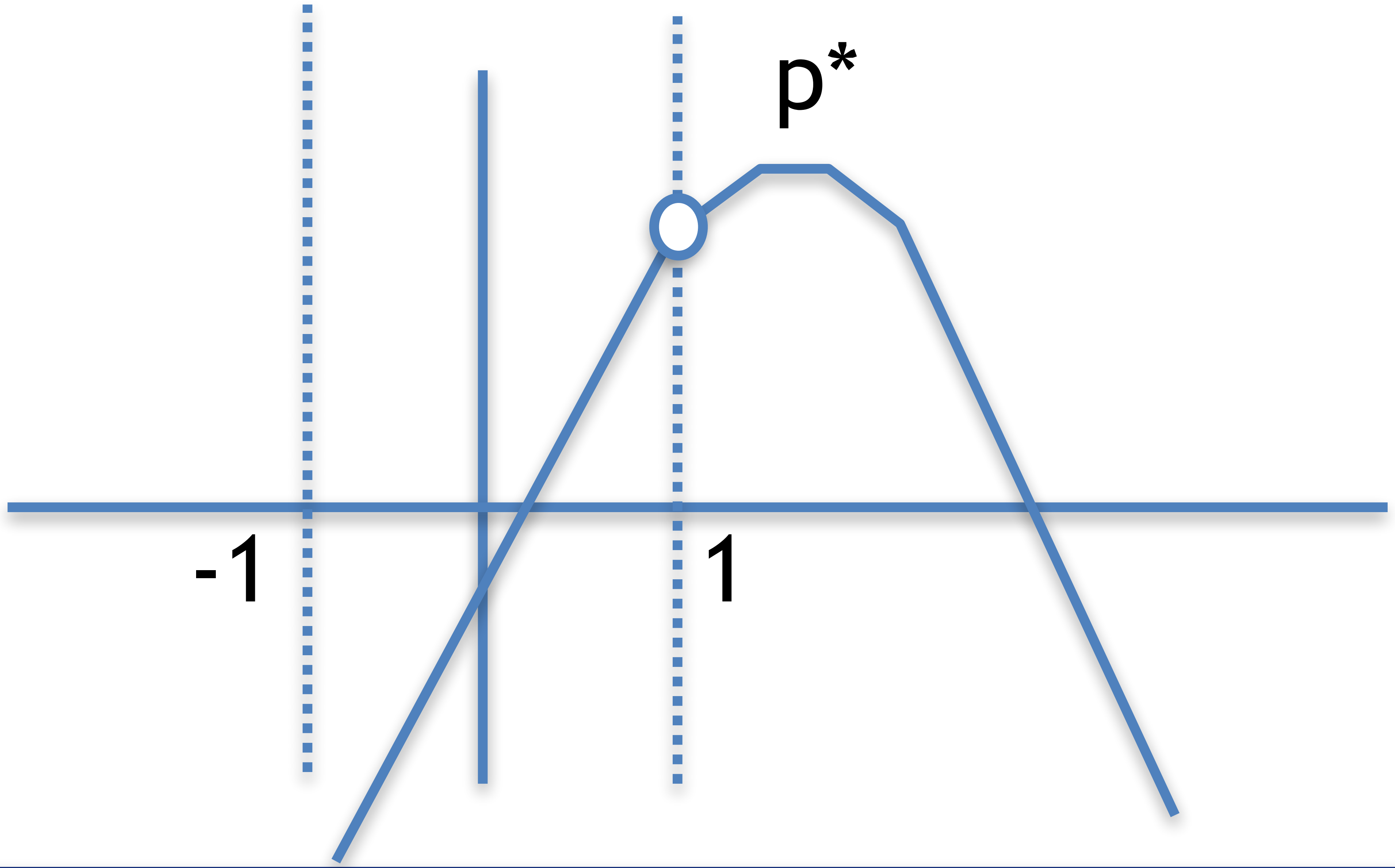
If the max is larger than one than the parabola is indeed like this



# LASSO

For $w > \tau$ instead $\hat{p} > 1$

If the max is larger than one than the parabola is indeed like this

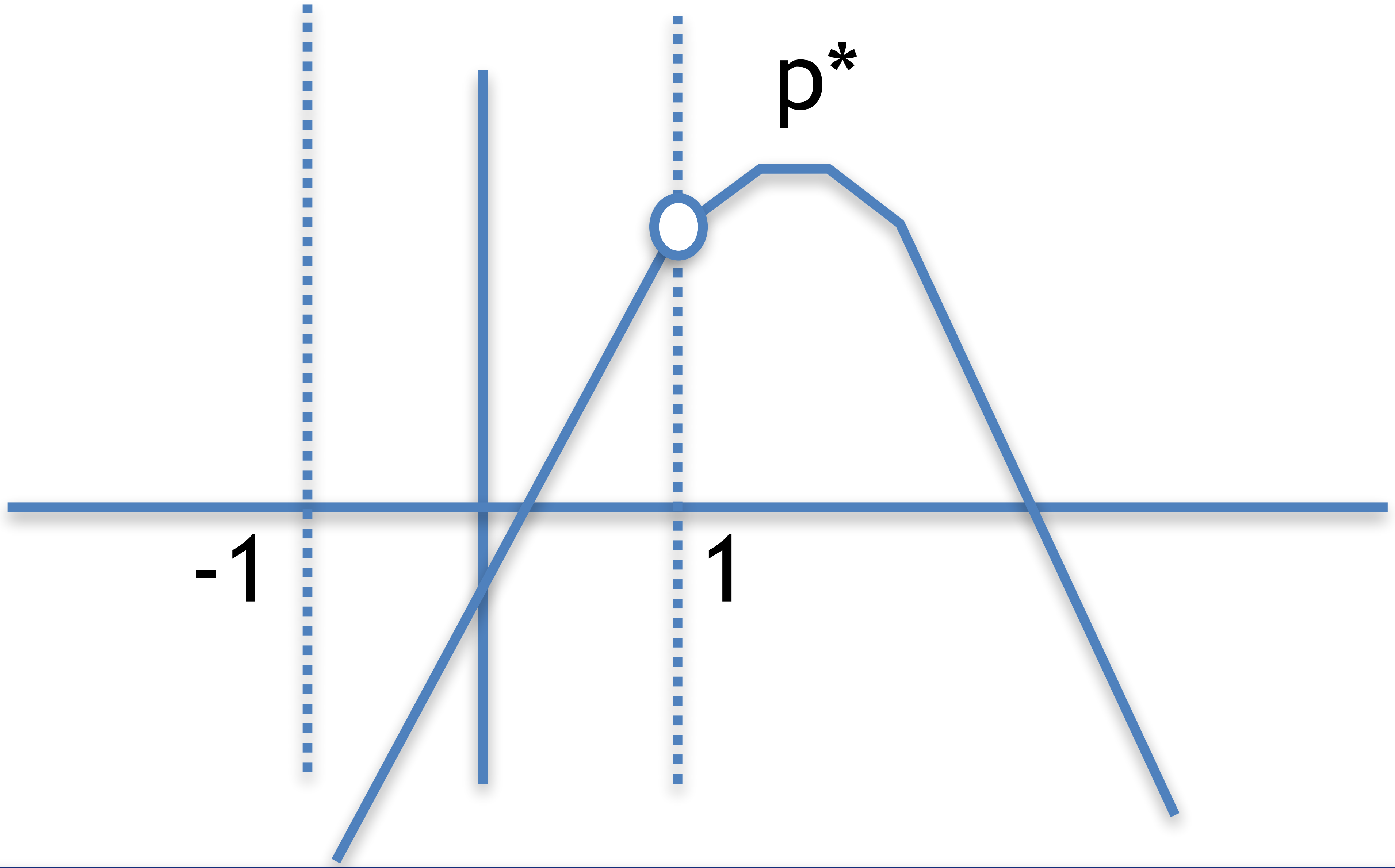



And the max is for p=1

# LASSO

For $w > \tau$ instead $\hat{p} > 1$

If the max is larger than one than the parabola is indeed like this




And the max is for p=1

This implies $|w|_\tau = w - \frac{\tau}{2}$

# LASSO

For $w < -\tau$ instead $\hat{p} < -1$

# LASSO

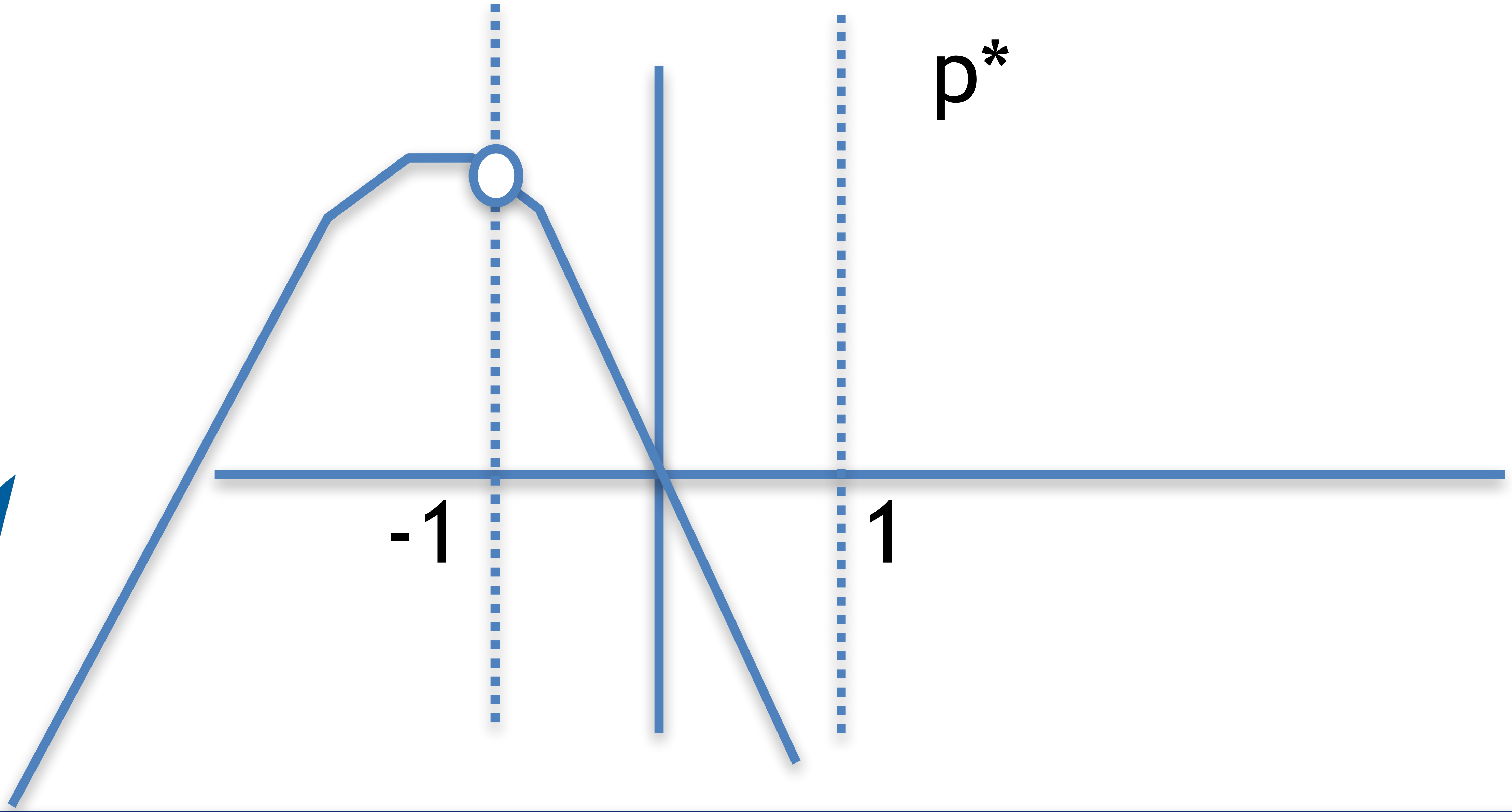
For $w < -\tau$ instead $\hat{p} < -1$

If the max is smaller than one than the parabola is instead like this

# LASSO

For $w < -\tau$ instead $\hat{p} < -1$

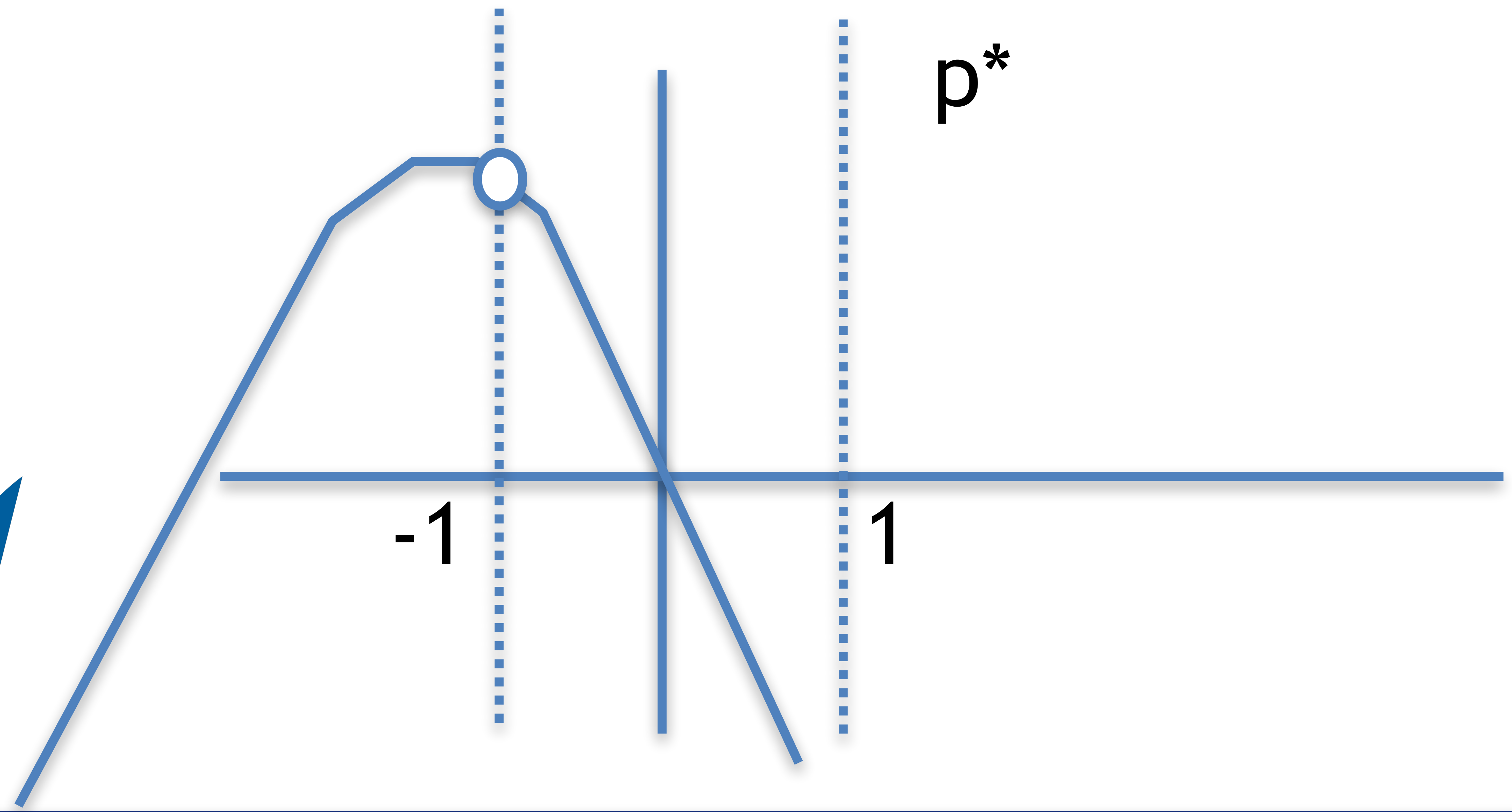
If the max is smaller than one than the parabola is instead like this



# LASSO

For $w < -\tau$ instead $\hat{p} < -1$

If the max is smaller than one than the parabola is instead like this

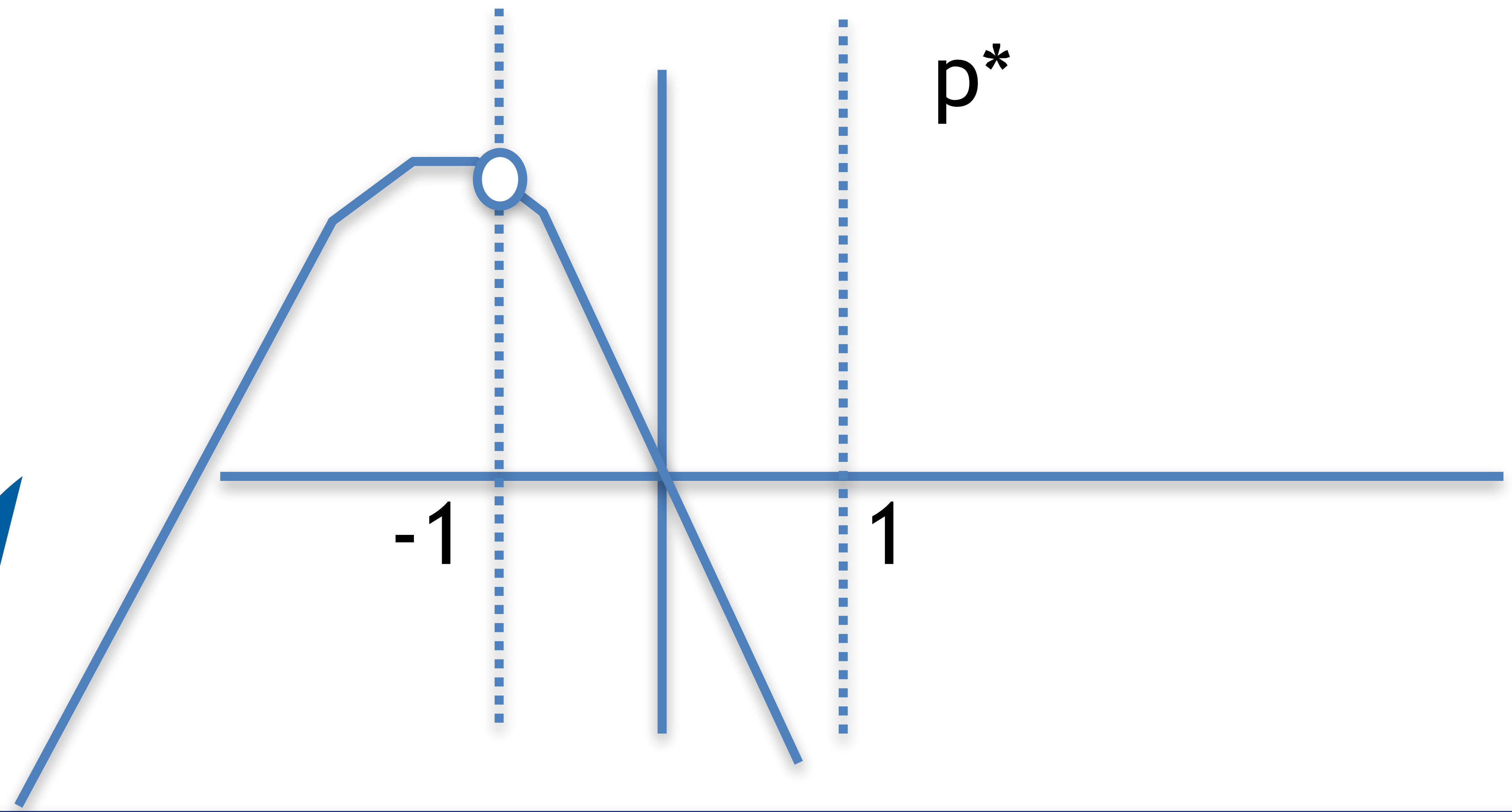



And the max is for p=-1

# LASSO

For $w < -\tau$ instead $\hat{p} < -1$

If the max is smaller than one than the parabola is instead like this




And the max is for p=-1

This implies $|w|_\tau = -w - \frac{\tau}{2}$

# LASSO

Hence

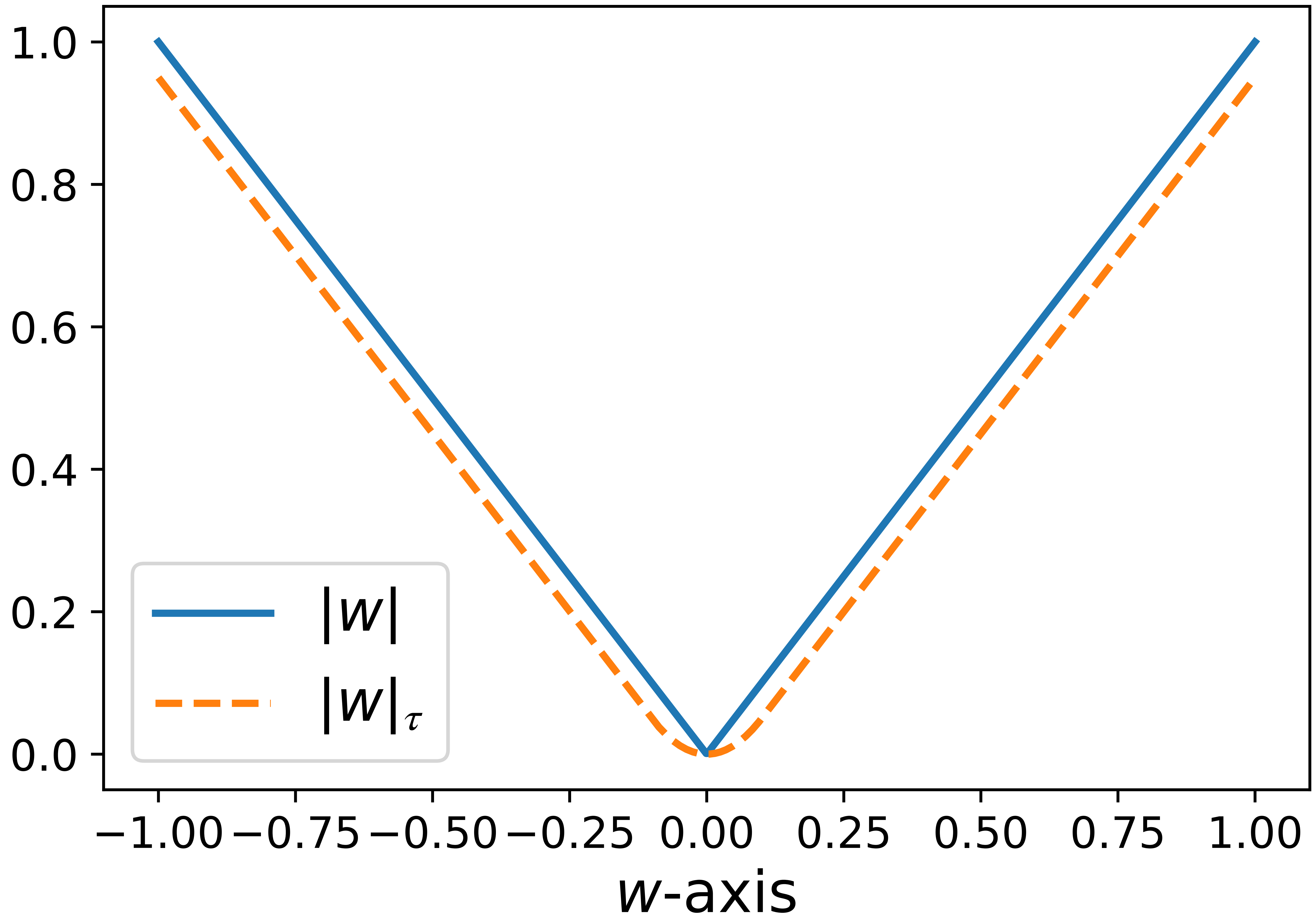
$$|\mathbf{w}|_\tau = \max_{p\in[-1,1]} wp \quad - \quad \frac{\tau}{2}|p|^2$$

has a closed form solution

$$\hat{p} = \arg\max_{p\in[-1,1]} wp - \frac{\tau}{2}|p|^2$$

$$\Longrightarrow \quad \hat{p} = \begin{cases} 1 & w > \tau \\ \frac{w}{\tau} & |w| \le \tau \\ -1 & w < -\tau \end{cases} \quad \Longrightarrow \quad |\mathbf{w}|_\tau = \begin{cases} |w| - \frac{\tau}{2} & |w| > \tau \\ \frac{1}{2\tau}|w|^2 & |w| \le \tau \end{cases}$$

# LASSO




$$\tau = \frac{1}{10}$$

# LASSO

The change in the l1 norm allows us to write

$$|\mathbf{w}|_\tau = \begin{cases} |w| - \frac{\tau}{2} & |w| > \tau \\ \frac{1}{2\tau}|w|^2 & |w| \le \tau \end{cases}$$

for which we observe

$$\nabla |\mathbf{w}|_\tau = \begin{cases} 1 & w > \tau \\ \frac{1}{\tau} w & |w| \le \tau \\ -1 & w < -\tau \end{cases}$$

# LASSO

The change in the l1 norm allows us to write

$$|\mathbf{w}|_\tau = \begin{cases} |w| - \frac{\tau}{2} & |w| > \tau \\ \frac{1}{2\tau}|w|^2 & |w| \le \tau \end{cases}$$

for which we observe

$$\nabla |\mathbf{w}|_\tau = \begin{cases} 1 & w > \tau \\ \frac{1}{\tau} w & |w| \le \tau \\ -1 & w < -\tau \end{cases}$$

as well as

$$|\mathbf{w}|_\tau \le |\mathbf{w}| \le |\mathbf{w}|_\tau + \frac{\tau}{2}$$

# LASSO

We can therefore get a differentiable problem by replacing

$$\|\mathbf{w}\|_1 = \sum_{j=0}^{d} |w_j| \quad \text{with} \quad H_\tau(\mathbf{w}) = \sum_{j=0}^{d} |w_j|_\tau$$

Huber loss function

# LASSO

We can therefore get a differentiable problem by replacing

$$\|\mathbf{w}\|_1 = \sum_{j=0}^{d} |w_j| \quad \text{with} \quad H_\tau(\mathbf{w}) = \sum_{j=0}^{d} |w_j|_\tau$$

Huber loss function

Smoothed LASSO:

$$\mathbf{w}_\alpha = \arg\min_{\mathbf{w}\in\mathbb{R}^{d+1}} \left\{ \frac{1}{2}\|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2 + \alpha H_\tau(\mathbf{w}) \right\}$$

# LASSO

We can therefore get a differentiable problem by replacing

$$\|\mathbf{w}\|_1 = \sum_{j=0}^{d} |w_j| \quad \text{with} \quad H_\tau(\mathbf{w}) = \sum_{j=0}^{d} |w_j|_\tau$$

Huber loss function

Smoothed LASSO:

$$\mathbf{w}_\alpha = \arg \min_{\mathbf{w}\in\mathbb{R}^{d+1}} \left\{ \frac{1}{2}\|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2 + \alpha H_\tau(\mathbf{w}) \right\}$$

How can we solve this problem?

# LASSO

Smoothed LASSO: $$\mathbf{w}_\alpha = \arg\min_{\mathbf{w}\in\mathbb{R}^{d+1}} \left\{ \frac{1}{2}\|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2 + \alpha H_\tau(\mathbf{w}) \right\}$$

How can we solve this problem?

One variant:

# LASSO

Smoothed LASSO: $$\mathbf{w}_\alpha = \arg \min_{\mathbf{w}\in\mathbb{R}^{d+1}} \left\{ \frac{1}{2} \|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2 + \alpha H_\tau(\mathbf{w}) \right\}$$

How can we solve this problem?

One variant: gradient descent for $E_\tau(\mathbf{w}) := \frac{1}{2}\|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2 + \alpha\, H_\tau(\mathbf{w})$:

# LASSO

Smoothed LASSO: $$\mathbf{w}_\alpha = \arg\min_{\mathbf{w}\in\mathbb{R}^{d+1}} \left\{ \frac{1}{2}\|\mathbf{X}\mathbf{w}-\mathbf{y}\|^2 + \alpha H_\tau(\mathbf{w}) \right\}$$

How can we solve this problem?

One variant: gradient descent for $E_\tau(\mathbf{w}) := \frac{1}{2}\|\mathbf{X}\mathbf{w}-\mathbf{y}\|^2 + \alpha\, H_\tau(\mathbf{w})$:

$$\mathbf{w}^{k+1} = \mathbf{w}^k - \tau \nabla E(\mathbf{w}^k)$$

# LASSO

Smoothed LASSO: $$\mathbf{w}_\alpha = \arg\min_{\mathbf{w}\in\mathbb{R}^{d+1}} \left\{ \frac{1}{2}\|\mathbf{X}\mathbf{w}-\mathbf{y}\|^2 + \alpha H_\tau(\mathbf{w}) \right\}$$

How can we solve this problem?

One variant: gradient descent for $E_\tau(\mathbf{w}) := \frac{1}{2}\|\mathbf{X}\mathbf{w}-\mathbf{y}\|^2 + \alpha\, H_\tau(\mathbf{w})$:

$$\mathbf{w}^{k+1} = \mathbf{w}^k - \tau \nabla E(\mathbf{w}^k)$$

$$\mathbf{w}^{k+1} = \mathbf{w}^k - \tau \left( \mathbf{X}^\top(\mathbf{X}\mathbf{w}^k - \mathbf{y}) + \alpha \nabla H_\tau(\mathbf{w}^k) \right)$$

# LASSO

Smoothed LASSO: $$\mathbf{w}_\alpha = \arg\min_{\mathbf{w}\in\mathbb{R}^{d+1}} \left\{ \frac{1}{2}\|\mathbf{X}\mathbf{w}-\mathbf{y}\|^2 + \alpha H_\tau(\mathbf{w}) \right\}$$

How can we solve this problem?

One variant: gradient descent for $E_\tau(\mathbf{w}) := \frac{1}{2}\|\mathbf{X}\mathbf{w}-\mathbf{y}\|^2 + \alpha\, H_\tau(\mathbf{w})$:

$$\mathbf{w}^{k+1} = \mathbf{w}^k - \tau \nabla E(\mathbf{w}^k)$$

$$\mathbf{w}^{k+1} = \mathbf{w}^k - \tau \left( \mathbf{X}^\top(\mathbf{X}\mathbf{w}^k - \mathbf{y}) + \alpha \nabla H_\tau(\mathbf{w}^k) \right)$$

We have two competing terms due to the structure of E

# LASSO

Alternative: forward-forward splitting for $E_\tau(\mathbf{w}) := \frac{1}{2\alpha}\|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2 + \ H_\tau(\mathbf{w})$

# LASSO

Alternative: forward-forward splitting for $E_\tau(\mathbf{w}) := \frac{1}{2\alpha}\|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2 + \ H_\tau(\mathbf{w})$

$$\mathbf{w}^{k+\frac{1}{2}} = \mathbf{w}^k - \frac{\tau}{\alpha}\mathbf{X}^\top(\mathbf{X}\mathbf{w}^{\mathbf{k}} - \mathbf{y})$$

We move first towards the opposite of the max variation of MSE

# LASSO

Alternative: forward-forward splitting for $E_\tau(\mathbf{w}) := \frac{1}{2\alpha}\|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2 + H_\tau(\mathbf{w})$

$$\mathbf{w}^{k+\frac{1}{2}} = \mathbf{w}^k - \frac{\tau}{\alpha}\mathbf{X}^\top(\mathbf{X}\mathbf{w}^\mathbf{k} - \mathbf{y})$$

We move first towards the opposite of the max variation of MSE

$$\mathbf{w}^{k+1} = \mathbf{w}^{k+\frac{1}{2}} - \tau \nabla H_\tau(\mathbf{w}^{k+\frac{1}{2}})$$

We move then towards the opposite of the max variation of the Huber loss function

# LASSO

Alternative: forward-forward splitting for $E_\tau(\mathbf{w}) := \frac{1}{2\alpha}\|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2 + H_\tau(\mathbf{w})$

Converge for $\frac{\tau}{\alpha} \leq \|\mathbf{X}^\top \mathbf{X}\|^{-1}$

$$\mathbf{w}^{k+\frac{1}{2}} = \mathbf{w}^k - \frac{\tau}{\alpha}\mathbf{X}^\top(\mathbf{X}\mathbf{w}^{\mathbf{k}} - \mathbf{y})$$

We move first towards the opposite of the max variation of MSE

$$\mathbf{w}^{k+1} = \mathbf{w}^{k+\frac{1}{2}} - \tau \nabla H_\tau(\mathbf{w}^{k+\frac{1}{2}})$$

We move then towards the opposite of the max variation of the Huber loss function

# LASSO

Alternative: forward-forward splitting for $E_\tau(\mathbf{w}) := \frac{1}{2\alpha}\|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2 + H_\tau(\mathbf{w})$

Converge for $\frac{\tau}{\alpha} \leq \|\mathbf{X}^\top\mathbf{X}\|^{-1}$

$$\mathbf{w}^{k+\frac{1}{2}} = \mathbf{w}^k - \frac{\tau}{\alpha}\mathbf{X}^\top(\mathbf{X}\mathbf{w}^{\mathbf{k}} - \mathbf{y})$$

We move first towards the opposite of the max variation of MSE

Converge for any $\tau$

$$\mathbf{w}^{k+1} = \mathbf{w}^{k+\frac{1}{2}} - \tau\nabla H_\tau(\mathbf{w}^{k+\frac{1}{2}})$$

We move then towards the opposite of the max variation of the Huber loss function

# LASSO

Alternative: forward-forward splitting for $E_\tau(\mathbf{w}) := \frac{1}{2\alpha}\|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2 + H_\tau(\mathbf{w})$

Converge for $\frac{\tau}{\alpha} \leq \|\mathbf{X}^\top \mathbf{X}\|^{-1}$

$$\mathbf{w}^{k+\frac{1}{2}} = \mathbf{w}^k - \frac{\tau}{\alpha}\mathbf{X}^\top(\mathbf{X}\mathbf{w}^{\mathbf{k}} - \mathbf{y})$$

We move first towards the opposite of the max variation of MSE

Converge for any $\tau$

$$\mathbf{w}^{k+1} = \mathbf{w}^{k+\frac{1}{2}} - \tau \nabla H_\tau(\mathbf{w}^{k+\frac{1}{2}})$$

We move then towards the opposite of the max variation of the Huber loss function

Hence we select $\frac{\tau}{\alpha} \leq \|\mathbf{X}^\top \mathbf{X}\|^{-1}$

# LASSO

Alternative: forward-forward splitting for $E_\tau(\mathbf{w}) := \frac{1}{2\alpha}\|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2 + \; H_\tau(\mathbf{w})$:

$$\mathbf{w}^{k+\frac{1}{2}} = \mathbf{w}^k - \frac{\tau}{\alpha}\mathbf{X}^\top(\mathbf{X}\mathbf{w}^{\mathbf{k}} - \mathbf{y})$$

$$\mathbf{w}^{k+1} = \mathbf{w}^{k+\frac{1}{2}} - \tau \nabla H_\tau(\mathbf{w}^{k+\frac{1}{2}})$$

Note the following:

$$w - \tau \nabla |w|_\tau \; = w - \begin{cases} \tau & w > \tau \\ w & |w| \leq \tau \\ -\tau & w < -\tau \end{cases}$$

# LASSO

Alternative: forward-forward splitting for $E_\tau(\mathbf{w}) := \frac{1}{2\alpha}\|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2 + H_\tau(\mathbf{w})$:

$$\mathbf{w}^{k+\frac{1}{2}} = \mathbf{w}^k - \frac{\tau}{\alpha}\mathbf{X}^\top(\mathbf{X}\mathbf{w}^\mathbf{k} - \mathbf{y})$$

$$\mathbf{w}^{k+1} = \mathbf{w}^{k+\frac{1}{2}} - \tau \nabla H_\tau(\mathbf{w}^{k+\frac{1}{2}})$$

Note the following:

$$w - \tau \nabla |w|_\tau = w - \begin{cases} \tau & w > \tau \\ w & |w| \leq \tau \\ -\tau & w < -\tau \end{cases} = \begin{cases} w - \tau & w > \tau \\ 0 & |w| \leq \tau \\ w + \tau & w < -\tau \end{cases}$$

# LASSO

Alternative: forward-forward splitting for $E_\tau(\mathbf{w}) := \frac{1}{2\alpha}\|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2 + H_\tau(\mathbf{w})$:

$$\mathbf{w}^{k+\frac{1}{2}} = \mathbf{w}^k - \frac{\tau}{\alpha}\mathbf{X}^\top(\mathbf{X}\mathbf{w}^\mathbf{k} - \mathbf{y})$$

$$\mathbf{w}^{k+1} = \mathbf{w}^{k+\frac{1}{2}} - \tau \nabla H_\tau(\mathbf{w}^{k+\frac{1}{2}})$$

Note the following:

$$w - \tau \nabla |w|_\tau = w - \begin{cases} \tau & w > \tau \\ w & |w| \leq \tau \\ -\tau & w < -\tau \end{cases} = \begin{cases} w - \tau & w > \tau \\ 0 & |w| \leq \tau \\ w + \tau & w < -\tau \end{cases}$$

$=:$ $\text{soft}_\tau(w)$ (soft-thresholding)

# LASSO

Alternative: forward-forward splitting for $E_\tau(\mathbf{w}) := \frac{1}{2\alpha}\|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2 + H_\tau(\mathbf{w})$:

$$\mathbf{w}^{k+\frac{1}{2}} = \mathbf{w}^k - \frac{\tau}{\alpha}\mathbf{X}^\top(\mathbf{X}\mathbf{w}^k - \mathbf{y})$$

$$\mathbf{w}^{k+1} = \mathbf{w}^{k+\frac{1}{2}} - \tau \nabla H_\tau(\mathbf{w}^{k+\frac{1}{2}})$$

# LASSO

Alternative: forward-forward splitting for $E_\tau(\mathbf{w}) := \frac{1}{2\alpha}\|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2 + H_\tau(\mathbf{w})$:

$$\mathbf{w}^{k+\frac{1}{2}} = \mathbf{w}^k - \frac{\tau}{\alpha}\mathbf{X}^\top(\mathbf{X}\mathbf{w}^k - \mathbf{y})$$

$$\mathbf{w}^{k+1} = \mathbf{w}^{k+\frac{1}{2}} - \tau \nabla H_\tau(\mathbf{w}^{k+\frac{1}{2}})$$

The last term can be written as

# LASSO

Alternative: forward-forward splitting for $E_\tau(\mathbf{w}) := \frac{1}{2\alpha}\|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2 + H_\tau(\mathbf{w})$:

$$\mathbf{w}^{k+\frac{1}{2}} = \mathbf{w}^k - \frac{\tau}{\alpha}\mathbf{X}^\top(\mathbf{X}\mathbf{w}^k - \mathbf{y})$$

$$\mathbf{w}^{k+1} = \mathbf{w}^{k+\frac{1}{2}} - \tau \nabla H_\tau(\mathbf{w}^{k+\frac{1}{2}})$$

The last term can be written as

$$\mathbf{w}^{k+1} = \mathrm{soft}_\tau(\mathbf{w}^{k+\frac{1}{2}})$$

# LASSO

Alternative: forward-forward splitting for $E_\tau(\mathbf{w}) := \frac{1}{2\alpha}\|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2 + H_\tau(\mathbf{w})$:

$$\mathbf{w}^{k+\frac{1}{2}} = \mathbf{w}^k - \frac{\tau}{\alpha}\mathbf{X}^\top(\mathbf{X}\mathbf{w}^k - \mathbf{y})$$

$$\mathbf{w}^{k+1} = \mathbf{w}^{k+\frac{1}{2}} - \tau \nabla H_\tau(\mathbf{w}^{k+\frac{1}{2}})$$

The last term can be written as

$$\mathbf{w}^{k+1} = \text{soft}_\tau(\mathbf{w}^{k+\frac{1}{2}})$$

Hence, the soft-thresholding of the previous expression

# LASSO

Alternative: forward-forward splitting for $E_\tau(\mathbf{w}) := \frac{1}{2\alpha}\|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2 + H_\tau(\mathbf{w})$:

$$\mathbf{w}_j^{k+1} = \text{soft}_\tau\left(\left(\mathbf{w}^k - \frac{\tau}{\alpha}\mathbf{X}^\top(\mathbf{X}\mathbf{w}^k - \mathbf{y})\right)_j\right) \qquad \forall j \in \{1, \ldots, d+1\}$$

# LASSO

Alternative: forward-forward splitting for $E_\tau(\mathbf{w}) := \frac{1}{2\alpha}\|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2 + H_\tau(\mathbf{w})$:

$$\mathbf{w}_j^{k+1} = \text{soft}_\tau\left(\left(\mathbf{w}^k - \frac{\tau}{\alpha}\mathbf{X}^\top(\mathbf{X}\mathbf{w}^k - \mathbf{y})\right)_j\right) \qquad \forall j \in \{1, \ldots, d+1\}$$

This algorithm is also known as *ISTA (= iterative soft-thresholding algorithm)*

# LASSO

Alternative: forward-forward splitting for $E_\tau(\mathbf{w}) := \frac{1}{2\alpha}\|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2 + H_\tau(\mathbf{w})$:

$$\mathbf{w}_j^{k+1} = \text{soft}_\tau\left(\left(\mathbf{w}^k - \frac{\tau}{\alpha}\mathbf{X}^\top(\mathbf{X}\mathbf{w}^k - \mathbf{y})\right)_j\right) \qquad \forall j \in \{1, \ldots, d+1\}$$

This algorithm is also known as *ISTA (= iterative soft-thresholding algorithm)*

Special case of *proximal gradient descent*

$$\mathbf{w}^{k+1} = (I + \tau\partial R)^{-1}\left(\mathbf{w}^k - \tau\nabla L(\mathbf{w}^k)\right)$$

# Proximal gradient method

Suppose we want to minimise $E(\mathbf{w}) = L(\mathbf{w}) + R(\mathbf{w})$

# Proximal gradient method

Suppose we want to minimise $E(\mathbf{w}) = L(\mathbf{w}) + R(\mathbf{w})$

Assumptions:

1. $L$ is differentiable, i.e., $\nabla L(\mathbf{w})$ exists
2. $R$ has a simple proximal map, i.e.,

$$\text{prox}_{\tau R}(\mathbf{z}) := \arg\min_{\mathbf{x}} \left\{ \frac{1}{2}\|\mathbf{x} - \mathbf{z}\|^2 + \tau R(\mathbf{x}) \right\}$$

is easy to compute

# Proximal gradient method

Suppose we want to minimise $E(\mathbf{w}) = L(\mathbf{w}) + R(\mathbf{w})$

Assumptions: 1. $L$ is differentiable, i.e., $\nabla L(\mathbf{w})$ exists

2. $R$ has a simple proximal map, i.e.,

$$\text{prox}_{\tau R}(\mathbf{z}) := \arg\min_{\mathbf{x}} \left\{ \frac{1}{2}\|\mathbf{x} - \mathbf{z}\|^2 + \tau R(\mathbf{x}) \right\}$$

is easy to compute

Then: $$\mathbf{w}^{k+1} = \text{prox}_{\tau R}\left(\mathbf{w}^k - \tau \nabla L(\mathbf{w}^k)\right)$$

# Proximal gradient method

Suppose we want to minimise $E(\mathbf{w}) = L(\mathbf{w}) + R(\mathbf{w})$

Assumptions:

1. $L$ is differentiable, i.e., $\nabla L(\mathbf{w})$ exists
2. $R$ has a simple proximal map, i.e.,

$$\text{prox}_{\tau R}(\mathbf{z}) := \arg\min_{\mathbf{x}} \left\{ \frac{1}{2}\|\mathbf{x} - \mathbf{z}\|^2 + \tau R(\mathbf{x}) \right\}$$

is easy to compute

Then: $$\mathbf{w}^{k+1} = \text{prox}_{\tau R}\left(\mathbf{w}^k - \tau \nabla L(\mathbf{w}^k)\right)$$ Proximal gradient method

# Proximal gradient method

For the choice $R(x) = \frac{1}{2}\|x\|^2$ this reads as

$$\text{prox}_{\frac{\tau}{2}\|\cdot\|^2}(z) = \arg\min_x \left\{ \frac{1}{2}\|x - z\|^2 + \frac{\tau}{2}\|x\|^2 \right\}$$

# Proximal gradient method

For the choice $R(x) = \frac{1}{2}\|x\|^2$ this reads as

$$\text{prox}_{\frac{\tau}{2}\|\cdot\|^2}(z) = \arg\min_x \left\{ \frac{1}{2}\|x - z\|^2 + \frac{\tau}{2}\|x\|^2 \right\}$$

Forget for a second the proximal map, we know how to solve that problem!

# Proximal gradient method

$$\text{prox}_{\frac{\tau}{2}\|\cdot\|^2}(z) = \arg\min_x \left\{ \frac{1}{2}\|x - z\|^2 + \frac{\tau}{2}\|x\|^2 \right\}$$

This is a simple convex optimisation problem. If we define $E(x) := \frac{1}{2}\|x - z\|^2 + \frac{\tau}{2}\|x\|^2$, we obtain $\nabla E(x) = x - z + \tau x$. The global minimiser satisfies

$$\nabla E(\hat{x}) = 0 \qquad \Leftrightarrow \qquad \hat{x} = \frac{z}{1 + \tau}$$

# Proximal gradient method

$$\text{prox}_{\frac{\tau}{2}\|\cdot\|^2}(z) = \arg\min_x \left\{ \frac{1}{2}\|x - z\|^2 + \frac{\tau}{2}\|x\|^2 \right\}$$

This is a simple convex optimisation problem. If we define $E(x) := \dfrac{1}{2}\|x - z\|^2 + \dfrac{\tau}{2}\|x\|^2$, we obtain $\nabla E(x) = x - z + \tau x$. The global minimiser satisfies

$$\nabla E(\hat{x}) = 0 \qquad \Leftrightarrow \qquad \hat{x} = \frac{z}{1 + \tau}$$

$$\Longrightarrow \qquad \text{prox}_{\frac{\tau}{2}\|\cdot\|^2}(z) = \frac{z}{1 + \tau}$$

# Proximal gradient method

Example for a proximal map

$$\text{prox}_{\tau R}(\mathbf{z}) := \arg\min_{\mathbf{x}} \left\{ \frac{1}{2}\|\mathbf{x}-\mathbf{z}\|^2 + \tau R(\mathbf{x}) \right\} \qquad :$$

# Proximal gradient method

Example for a proximal map

$$\text{prox}_{\tau R}(\mathbf{z}) := \arg\min_{\mathbf{x}} \left\{ \frac{1}{2}\|\mathbf{x} - \mathbf{z}\|^2 + \tau R(\mathbf{x}) \right\} \qquad :$$

For the choice $R(\mathbf{x}) = \begin{cases} 0 & \mathbf{x} \in C \\ \infty & \mathbf{x} \notin C \end{cases}$ this reads as

$$\text{prox}_{\tau R}(\mathbf{z}) = \arg\min_{\mathbf{x}} \left\{ \frac{1}{2}\|\mathbf{x} - \mathbf{z}\|^2 + \tau R(\mathbf{x}) \right\}$$

# Proximal gradient method

Example for a proximal map

$$\text{prox}_{\tau R}(\mathbf{z}) := \arg\min_{\mathbf{x}} \left\{ \frac{1}{2}\|\mathbf{x}-\mathbf{z}\|^2 + \tau R(\mathbf{x}) \right\} \qquad :$$

For the choice $R(\mathbf{x}) = \begin{cases} 0 & \mathbf{x} \in C \\ \infty & \mathbf{x} \notin C \end{cases}$ this reads as

$$\text{prox}_{\tau R}(\mathbf{z}) = \arg\min_{\mathbf{x}} \left\{ \frac{1}{2}\|\mathbf{x}-\mathbf{z}\|^2 + \tau R(\mathbf{x}) \right\} = \arg\min_{\mathbf{x}\in C} \left\{ \frac{1}{2}\|\mathbf{x}-\mathbf{z}\|^2 \right\}$$

# Proximal gradient method

Example for a proximal map

$$\text{prox}_{\tau R}(\mathbf{z}) := \arg\min_{\mathbf{x}} \left\{ \frac{1}{2}\|\mathbf{x} - \mathbf{z}\|^2 + \tau R(\mathbf{x}) \right\}$$

For the choice $R(\mathbf{x}) = \begin{cases} 0 & \mathbf{x} \in C \\ \infty & \mathbf{x} \notin C \end{cases}$ this reads as

$$\text{prox}_{\tau R}(\mathbf{z}) = \arg\min_{\mathbf{x}} \left\{ \frac{1}{2}\|\mathbf{x} - \mathbf{z}\|^2 + \tau R(\mathbf{x}) \right\} = \arg\min_{\mathbf{x}\in C} \left\{ \|\mathbf{x} - \mathbf{z}\| \right\}$$

# Proximal gradient method

Example for a proximal map

$$\text{prox}_{\tau R}(\mathbf{z}) := \arg\min_{\mathbf{x}} \left\{ \frac{1}{2}\|\mathbf{x}-\mathbf{z}\|^2 + \tau R(\mathbf{x}) \right\}$$

For the choice $R(\mathbf{x}) = \begin{cases} 0 & \mathbf{x} \in C \\ \infty & \mathbf{x} \notin C \end{cases}$ this reads as

$$\text{prox}_{\tau R}(\mathbf{z}) = \arg\min_{\mathbf{x}} \left\{ \frac{1}{2}\|\mathbf{x}-\mathbf{z}\|^2 + \tau R(\mathbf{x}) \right\} = \arg\min_{\mathbf{x}\in C} \left\{ \|\mathbf{x}-\mathbf{z}\| \right\}$$

Projection onto convex set $C$!

# Proximal gradient method

Example for a proximal map

$$\text{prox}_{\tau R}(\mathbf{z}) := \arg\min_{\mathbf{x}} \left\{ \frac{1}{2}\|\mathbf{x}-\mathbf{z}\|^2 + \tau R(\mathbf{x}) \right\}$$

For the choice $R(\mathbf{x}) = \begin{cases} 0 & \mathbf{x} \in C \\ \infty & \mathbf{x} \notin C \end{cases}$ this reads as

$$\text{prox}_{\tau R}(\mathbf{z}) = \arg\min_{\mathbf{x}} \left\{ \frac{1}{2}\|\mathbf{x}-\mathbf{z}\|^2 + \tau R(\mathbf{x}) \right\} = \arg\min_{\mathbf{x}\in C} \left\{ \|\mathbf{x}-\mathbf{z}\| \right\}$$

Projection onto convex set $C$!

This might be important in some real applications where we have some constraints on the x!

# Proximal gradient method

Example for a proximal map

$$\text{prox}_{\tau R}(\mathbf{z}) := \arg\min_{\mathbf{x}} \left\{ \frac{1}{2}\|\mathbf{x}-\mathbf{z}\|^2 + \tau R(\mathbf{x}) \right\}$$

For the choice $R(\mathbf{x}) = \begin{cases} 0 & \mathbf{x} \in C \\ \infty & \mathbf{x} \notin C \end{cases}$ this reads as

$$\text{prox}_{\tau R}(\mathbf{z}) = \arg\min_{\mathbf{x}} \left\{ \frac{1}{2}\|\mathbf{x}-\mathbf{z}\|^2 + \tau R(\mathbf{x}) \right\} = \arg\min_{\mathbf{x}\in C} \left\{ \|\mathbf{x}-\mathbf{z}\| \right\}$$

Projection onto convex set $C$!

This might be important in some real applications where we have some constraints on the x!

Example: $C = \{x \in \mathbb{R} \mid x \in [0,1]\}$

# Constrained optimisation

Special case:

$$R(w) = \begin{cases} 0 & w \in C \\ \infty & w \notin C \end{cases}$$

$C$ = convex set = constraint-set

# Constrained optimisation

Special case:

$$R(w) = \begin{cases} 0 & w \in C \\ \infty & w \notin C \end{cases}$$

$C$ = convex set = constraint-set

$$\Rightarrow \quad \text{prox}_{\tau R}(z) = \arg\min_{w \in \mathbb{R}^n} \|w - z\|^2 + R(w)$$

$$= \arg\min_{w \in C} \|w - z\|^2 = \text{proj}_C(z)$$

# Constrained optimisation

Special case:

$$R(w) = \begin{cases} 0 & w \in C \\ \infty & w \notin C \end{cases}$$

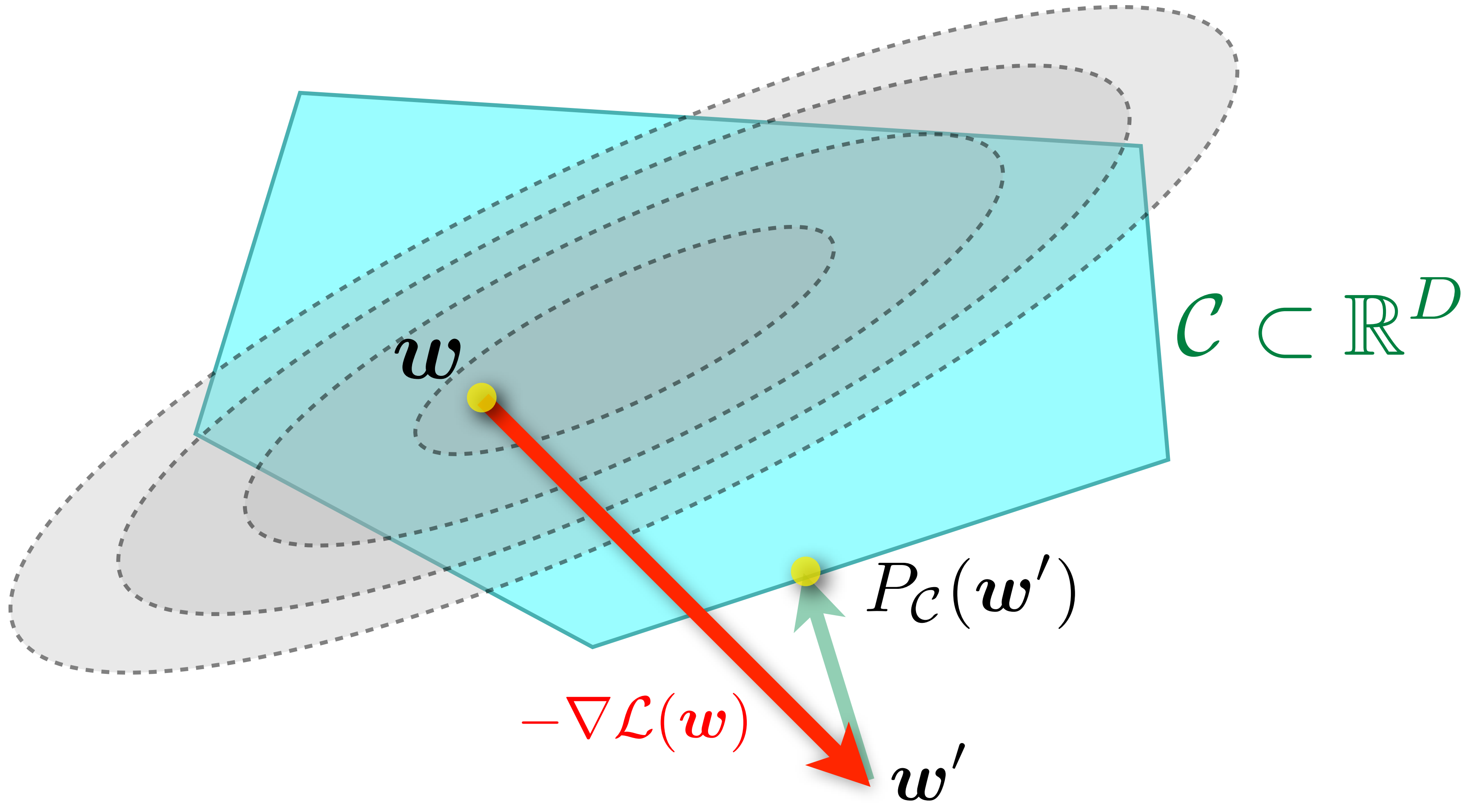
$C$ = convex set = constraint-set

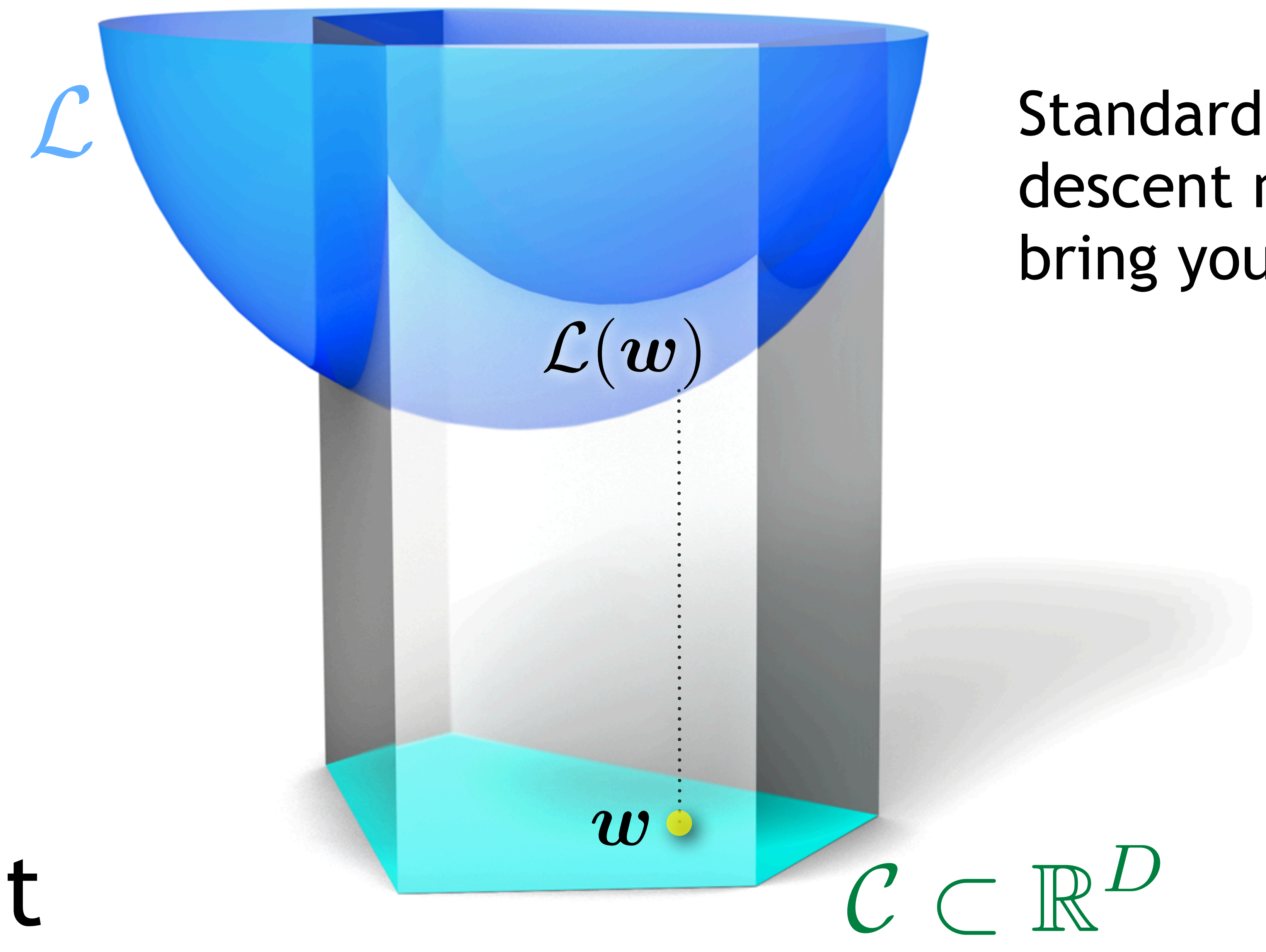
$$\Rightarrow \quad \text{prox}_{\tau R}(z) = \arg\min_{w \in \mathbb{R}^n} \|w - z\|^2 + R(w)$$

$$= \arg\min_{w \in C} \|w - z\|^2 = \text{proj}_C(z)$$

$$\Rightarrow \quad w^{k+1} = \text{proj}_C\left(w^k - \tau \nabla L(w^k)\right)$$

Projected gradient descent




Standard gradient descent might bring you out!

# Proximal gradient method

Suppose we want to minimise $E(\mathbf{w}) = L(\mathbf{w}) + R(\mathbf{w})$

Assumptions:

1. $L$ is differentiable, i.e. $\nabla L(\mathbf{w})$ exists

2. $R$ has a simple proximal map, i.e.

$$\text{prox}_{\tau R}(\mathbf{z}) := \arg\min_{\mathbf{x}} \left\{ \frac{1}{2}\|\mathbf{x} - \mathbf{z}\|^2 + \tau R(\mathbf{x}) \right\}$$

is easy to compute

Proximal gradient method: $\mathbf{w}^{k+1} = \text{prox}_{\tau R}\left(\mathbf{w}^k - \tau \nabla L(\mathbf{w}^k)\right)$

# Proximal gradient descent

Minimise variational regularisation $L(\mathbf{w}) + R(\mathbf{w})$ iteratively via

$$\mathbf{w}^{k+1} = (I + \tau\partial R)^{-1}\left(\mathbf{w}^k - \tau\nabla L(\mathbf{w}^k)\right)$$

where the *proximal map* is defined as

$$(I + \tau\partial R)^{-1}(\mathbf{z}) := \arg\min_{\mathbf{w}\in\mathbb{R}^{d+1}} \left\{ \frac{1}{2}\|\mathbf{w} - \mathbf{z}\|^2 + \tau R(\mathbf{w}) \right\}$$