Metodi Numerici dell'Informatica

Polynomial regression

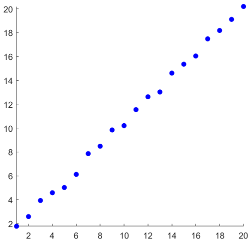
Emanuele Rodolà
rodola@di.uniroma1.it

2nd semester a.y. 2022/2023 · March 07, 2023

Motivation

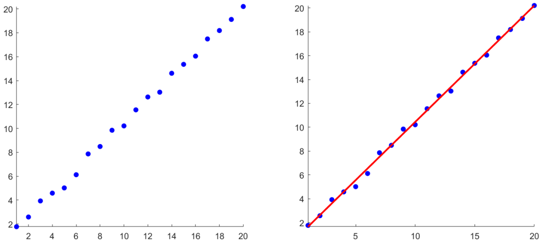
Linear regression

Consider the following fitting problem:



Linear regression

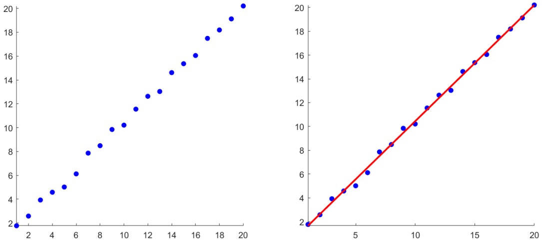
Consider the following fitting problem:



$$y_i = ax_i + b$$

Linear regression

Consider the following fitting problem:



$$f_{\Theta}(x_i) = y_i$$

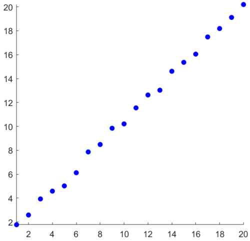
Model: linear + bias

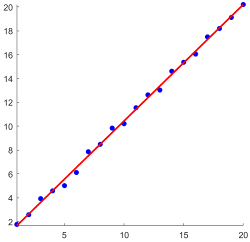
Unknown parameters: $\Theta = \{a, b\}$

Data: n pairs (x_i, y_i); the x_i are called the regressors

Linear regression

Consider the following fitting problem:





$$f_\Theta(x_i) = y_i$$

Model: linear + bias

Unknown parameters: $\Theta = \{a, b\}$

Data: n pairs (x_i, y_i); the x_i are called the regressors

Given a and b, we have a mapping that gives new output from new input.

Linear regression

The equations:

$$f_\Theta(x_i) = y_i$$

must approximately hold for all $i = 1, \ldots, n$.

Linear regression

The equations:

$$f_\Theta(x_i) = y_i$$

must approximately hold for all $i = 1, \ldots, n$.

Problem: Choose a and b that minimize the mean squared error (MSE) between input and predicted output:

$$\epsilon = \min_{a,b \in \mathbb{R}} \frac{1}{n} \sum_{i=1}^{n} (y_i - f_\Theta(x_i))^2$$

Linear regression

The equations:

$$f_\Theta(x_i) = y_i$$

must approximately hold for all $i = 1, \ldots, n$.

Problem: Choose a and b that minimize the mean squared error (MSE) between input and predicted output:

$$\epsilon = \min_{a,b \in \mathbb{R}} \sum_{i=1}^{n} (y_i - f_\Theta(x_i))^2$$

vinyl scratch

What is this weird mathematical expression?

Optimization

We are looking at what is typically called a minimization problem.

The general form for a minimization problem is:

$$\epsilon = \min_{\mathbf{x}} f(\mathbf{x})$$

Optimization

We are looking at what is typically called a minimization problem.

The general form for a minimization problem is:

$$\epsilon = \min_{\mathbf{x}} f(\mathbf{x})$$

Solving the problem means to find:

- A minimizer $\mathbf{x}^* = \arg\min_{\mathbf{x}} f(\mathbf{x})$

Optimization

We are looking at what is typically called a minimization problem.

The general form for a minimization problem is:

$$\epsilon = \min_{\mathbf{x}} f(\mathbf{x})$$

Solving the problem means to find:

- A minimizer $\mathbf{x}^* = \arg\min_{\mathbf{x}} f(\mathbf{x})$
- A minimum $\epsilon = f(\mathbf{x}^*)$

Optimization

We are looking at what is typically called a minimization problem.

The general form for a minimization problem is:

$$\epsilon = \min_{\mathbf{x}} f(\mathbf{x})$$

Solving the problem means to find:

- A minimizer $\mathbf{x}^* = \arg\min_{\mathbf{x}} f(\mathbf{x})$
- A minimum $\epsilon = f(\mathbf{x}^*)$

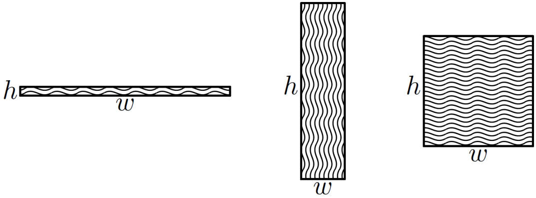
There are no general recipes that work well for all problems!

In general, the algorithm you choose depends on the properties of f.

The research area is broadly called optimization.

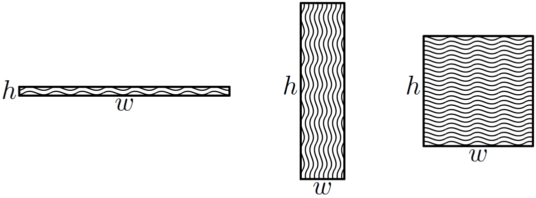
Example

Suppose a rectangle has width w and height h. A classic geometry problem is to maximize area with a fixed perimeter equal to 1:



Example

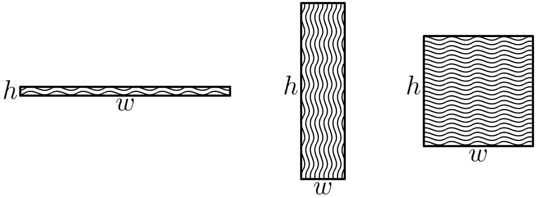
Suppose a rectangle has width w and height h. A classic geometry problem is to maximize area with a fixed perimeter equal to 1:



$$\begin{aligned} \max_{w,h\in\mathbb{R}} \quad & wh \\ \text{subject to} \quad & 2w + 2h = 1 \end{aligned}$$

Example

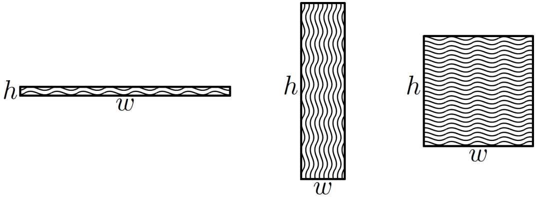
Suppose a rectangle has width w and height h. A classic geometry problem is to maximize area with a fixed perimeter equal to 1:



$$\begin{aligned} \max_{w,h\in\mathbb{R}} \quad & wh \\ \text{s.t.} \quad & 2w + 2h - 1 = 0 \end{aligned}$$

Example

Suppose a rectangle has width w and height h. A classic geometry problem is to maximize area with a fixed perimeter equal to 1:

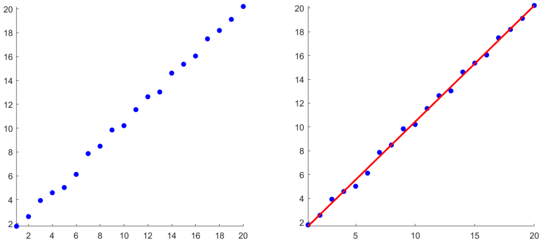


$$\begin{aligned} \max_{w,h\in\mathbb{R}} \quad & wh \\ \text{s.t.} \quad & 2w + 2h - 1 = 0 \end{aligned}$$

This is an example of a constrained problem.

Linear regression

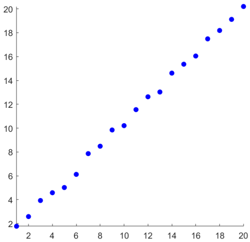
Back to our initial example of the linear fitting problem:

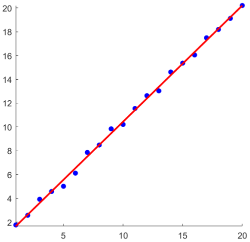


$$y_i = ax_i + b$$

Linear regression

Back to our initial example of the linear fitting problem:





$$y_i = ax_i + b$$

For cleaner and more generic formulas, we simply wrote:

$$f_\Theta(x_i) = y_i$$

where $f_\Theta(x_i) = ax_i + b$.

Linear regression

The equations:

$$f_\Theta(x_i) = y_i$$

must approximately hold for all $i = 1, \ldots, n$.

Linear regression

The equations:

$$f_\Theta(x_i) = y_i$$

must approximately hold for all $i = 1, \dots, n$.

Problem: Choose a and b that minimize the mean squared error (MSE) between input and predicted output:

$$\epsilon = \min_{a,b \in \mathbb{R}} \frac{1}{n} \sum_{i=1}^{n} (y_i - f_\Theta(x_i))^2$$

where $\Theta = \{a, b\}$.

Linear regression

The equations:

$$f_\Theta(x_i) = y_i$$

must approximately hold for all $i = 1, \ldots, n$.

Problem: Choose a and b that minimize the mean squared error (MSE) between input and predicted output:

$$\epsilon = \min_{a,b \in \mathbb{R}} \sum_{i=1}^{n} (y_i - f_\Theta(x_i))^2$$

where $\Theta = \{a, b\}$.

Recall that, in our example, f_Θ is linear:

$$f_\Theta(x_i) = ax_i + b$$

Linear regression

The equations:

$$f_\Theta(x_i) = y_i$$

must approximately hold for all $i = 1, \ldots, n$.

Problem: Choose a and b that minimize the mean squared error (MSE) between input and predicted output:

$$\epsilon = \min_{a,b\in\mathbb{R}} \sum_{i=1}^{n} (y_i - f_\Theta(x_i))^2$$

where $\Theta = \{a, b\}$.

Recall that, in our example, f_Θ is linear:

$$f_\Theta(x_i) = ax_i + b$$

This is called a least-squares approximation problem.

Finding a solution

To find a solution to the least-squares approximation problem that arises in linear regression, we need to introduce two more ingredients:

The notion of convexity

The definition of gradient

Convexity

Convex functions

Jensen's inequality:

$$f(\alpha x + (1-\alpha)y) \le \alpha f(x) + (1-\alpha) f(y)$$

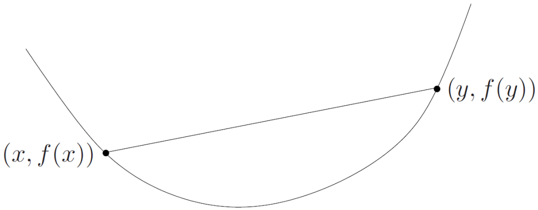
for all x, y and $\alpha \in (0, 1)$

Convex functions

Jensen's inequality:

$$f(\alpha x + (1-\alpha)y) \leq \alpha f(x) + (1-\alpha)f(y)$$

for all x, y and $\alpha \in (0,1)$

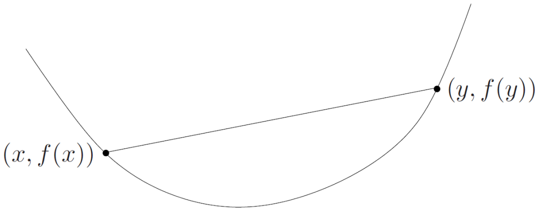


Convex functions

Jensen's inequality:

$$f(\alpha x + (1-\alpha)y) \leq \alpha f(x) + (1-\alpha)f(y)$$

for all x, y and $\alpha \in (0,1)$



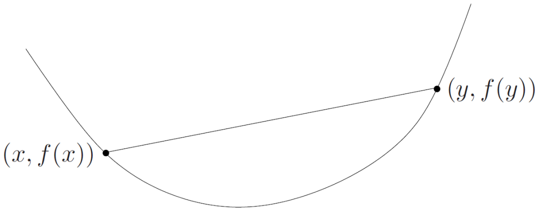
Let us further assume that f is a differentiable function, so that we can compute its derivative $\frac{df}{dx}$ at all points x.

Convex functions

Jensen's inequality:

$$f(\alpha x + (1-\alpha)y) \leq \alpha f(x) + (1-\alpha) f(y)$$

for all x, y and $\alpha \in (0, 1)$



Let us further assume that f is a differentiable function, so that we can compute its derivative $\frac{df}{dx}$ at all points x.

Intuition tells us that the minimizer x is where $\frac{df(x)}{dx} = 0$.

Convex functions: Global minima

$$f(\alpha x + (1-\alpha)y) \leq \alpha f(x) + (1-\alpha)f(y)$$

for all x, y and $\alpha \in (0,1)$

Convex functions: Global minima

$$f(x + \alpha(y - x)) \leq (1 - \alpha)f(x) + \alpha f(y)$$

for all x, y and $\alpha \in (0, 1)$

Convex functions: Global minima

$$\frac{f(x+\alpha(y-x))}{\alpha} \leq \frac{(1-\alpha)f(x)+\alpha f(y)}{\alpha}$$

for all x, y and $\alpha \in (0,1)$

Convex functions: Global minima

$$\frac{f(x + \alpha(y - x))}{\alpha} \leq \frac{f(x)}{\alpha} - f(x) + f(y)$$

for all x, y and $\alpha \in (0, 1)$

Convex functions: Global minima

$$\frac{f(x+\alpha(y-x))-f(x)}{\alpha}+f(x)\leq f(y)$$

for all x, y and $\alpha \in (0,1)$

Convex functions: Global minima

$$\lim_{\alpha \to 0} \frac{f(x + \alpha(y - x)) - f(x)}{\alpha} + f(x) \leq f(y)$$

Convex functions: Global minima

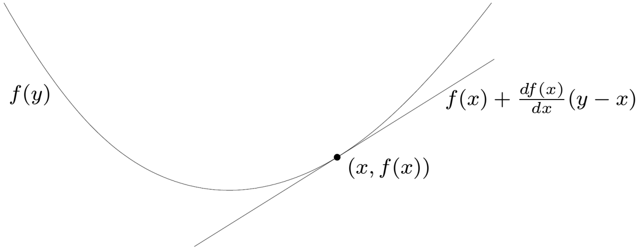
$$\lim_{\alpha\to 0}\frac{f(x+\alpha(y-x))-f(x)}{\alpha(y-x)}(y-x)+f(x)\leq f(y)$$

Convex functions: Global minima

$$\frac{df(x)}{dx}(y - x) + f(x) \leq f(y)$$

Convex functions: Global minima

$$\underbrace{\frac{df(x)}{dx}(y-x)+f(x)}_{\text{1st}-\text{order Taylor of } f(y) \text{ at } x} \quad \leq f(y)$$



Thus, if $\frac{df(x)}{dx}=0$:

$$f(x) \leq f(y)$$

which means that x is a global minimizer of f.

Convex functions: Global minima

To summarize:

If $f(x)$ is convex, then a global minimizer is found by setting $\frac{df(x)}{dx} = 0$ and solving for x.

Convex functions: Global minima

To summarize:

If $f(x)$ is convex, then a global minimizer is found by setting $\frac{df(x)}{dx} = 0$ and solving for x.

The above only makes sense if $f : \mathbb{R} \to \mathbb{R}$.

Convex functions: Global minima

To summarize:

If $f(x)$ is convex, then a global minimizer is found by setting $\frac{df(x)}{dx} = 0$ and solving for x.

The above only makes sense if $f : \mathbb{R} \to \mathbb{R}$.

If $f(\mathbf{x})$ is multi-dimensional, i.e. $f : \mathbb{R}^n \to \mathbb{R}$, do we have a notion of convexity and derivative?

Convex functions: Global minima

To summarize:

If $f(x)$ is convex, then a global minimizer is found by setting $\frac{df(x)}{dx} = 0$ and solving for x.

The above only makes sense if $f : \mathbb{R} \to \mathbb{R}$.

If $f(\mathbf{x})$ is multi-dimensional, i.e. $f : \mathbb{R}^n \to \mathbb{R}$, do we have a notion of convexity and derivative?

If yes, can we find global minimizers as easily as in the former case?

The gradient

Convex functions on $\mathbb{R}^n$

In general we will deal with functions of $n \gg 1$ variables:

$$f : \mathbb{R}^n \to \mathbb{R}$$

Convex functions on $\mathbb{R}^n$

In general we will deal with functions of $n \gg 1$ variables:

$$f : \mathbb{R}^n \to \mathbb{R}$$

The notion of derivative is replaced by the notion of gradient:

$$\nabla_{\mathbf{x}} f(\mathbf{x}) = \begin{pmatrix} \frac{\partial f}{\partial x_1} \\ \vdots \\ \frac{\partial f}{\partial x_n} \end{pmatrix}$$

which is the vector of partial derivatives of f.

Convex functions on $\mathbb{R}^n$

In general we will deal with functions of $n \gg 1$ variables:

$$f : \mathbb{R}^n \to \mathbb{R}$$

The notion of derivative is replaced by the notion of gradient:

$$\nabla_{\mathbf{x}} f(\mathbf{x}) = \begin{pmatrix} \frac{\partial f}{\partial x_1} \\ \vdots \\ \frac{\partial f}{\partial x_n} \end{pmatrix}$$

which is the vector of partial derivatives of f.

Convexity is defined as before:

$$f(\alpha \mathbf{x} + (1-\alpha)\mathbf{y}) \leq \alpha f(\mathbf{x}) + (1-\alpha) f(\mathbf{y})$$

Convex functions on $\mathbb{R}^n$

In general we will deal with functions of $n \gg 1$ variables:

$$f : \mathbb{R}^n \to \mathbb{R}$$

The notion of derivative is replaced by the notion of gradient:

$$\nabla_{\mathbf{x}} f(\mathbf{x}) = \begin{pmatrix} \frac{\partial f}{\partial x_1} \\ \vdots \\ \frac{\partial f}{\partial x_n} \end{pmatrix}$$

which is the vector of partial derivatives of f.

Convexity is defined as before:

$$f(\alpha \mathbf{x} + (1-\alpha)\mathbf{y}) \le \alpha f(\mathbf{x}) + (1-\alpha) f(\mathbf{y})$$

and we also have the global optimality condition:

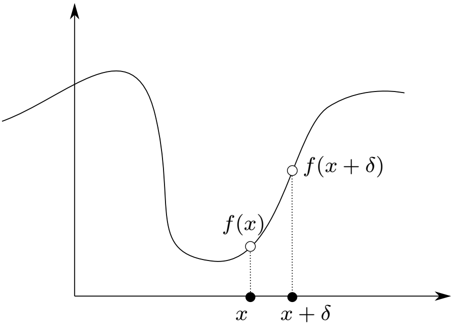
$$\nabla_{\mathbf{x}} f(\mathbf{x}) = \mathbf{0} \quad \Longrightarrow \quad f(\mathbf{x}) \le f(\mathbf{y}) \text{ for all } \mathbf{y} \in \mathbb{R}^n$$

The gradient

The gradient $\nabla_{\mathbf{x}} f(\mathbf{x})$ encodes the direction of steepest ascent of f at point $\mathbf{x}$.

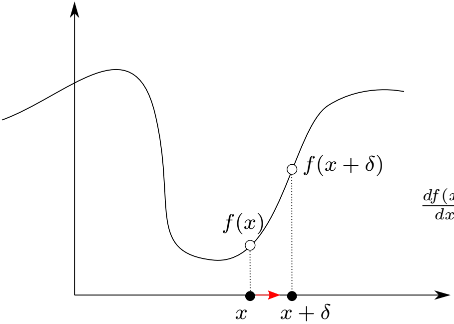
The gradient

The gradient $\nabla_{\mathbf{x}} f(\mathbf{x})$ encodes the direction of steepest ascent of f at point $\mathbf{x}$. In the simple 1D case:



The gradient

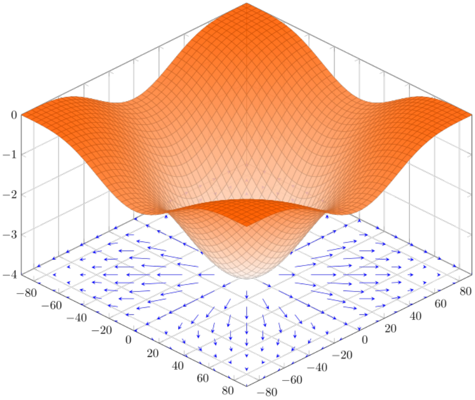
The gradient $\nabla_{\mathbf{x}} f(\mathbf{x})$ encodes the direction of steepest ascent of f at point $\mathbf{x}$. In the simple 1D case:



$$\frac{df(x)}{dx} = \lim_{\delta \to 0} \frac{f(x+\delta) - f(x)}{\delta}$$

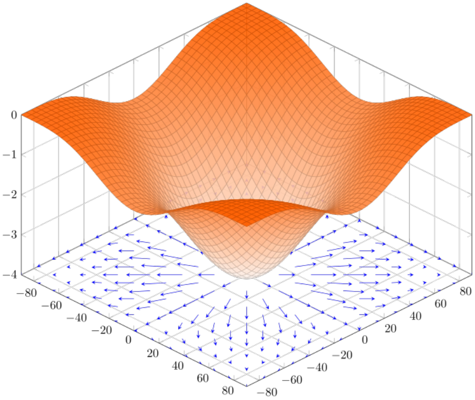
The gradient

The gradient $\nabla_{\mathbf{x}} f(\mathbf{x})$ encodes the direction of steepest ascent of f at point $\mathbf{x}$. In the more general case:



The gradient

The gradient $\nabla_{\mathbf{x}} f(\mathbf{x})$ encodes the direction of steepest ascent of f at point $\mathbf{x}$. In the more general case:



The length of the gradient vector encodes its steepness.

Convex functions: Global minima

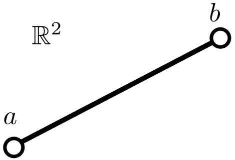
To summarize:

If $f(x)$ is convex, then a global minimizer is found by setting $\frac{df(x)}{dx} = 0$ and solving for x.

If $f(\mathbf{x})$ is convex, then a global minimizer is found by setting $\nabla_{\mathbf{x}} f(\mathbf{x}) = \mathbf{0}$ and solving for $\mathbf{x}$.

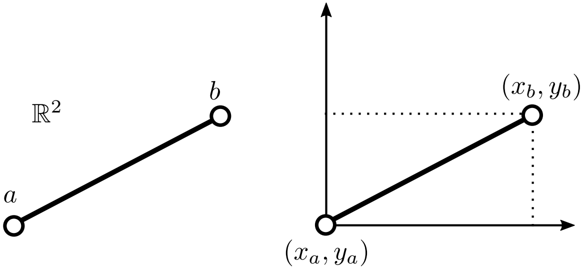
Vector lengths

How to measure the length of the gradient? Let's first start from the definition of Euclidean distance, which measures the length of any straight line connecting two points:



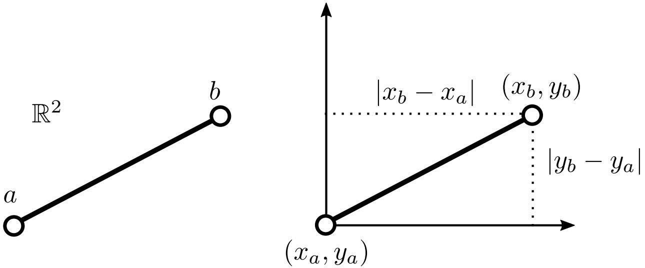
Vector lengths

How to measure the length of the gradient? Let's first start from the definition of Euclidean distance, which measures the length of any straight line connecting two points:



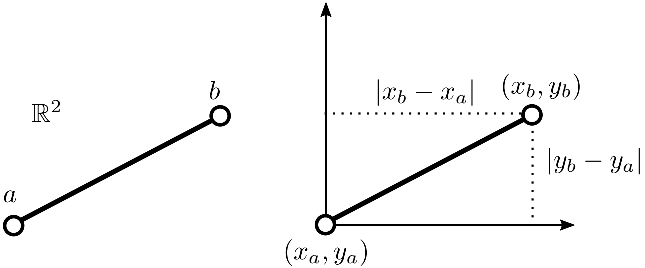
Vector lengths

How to measure the length of the gradient? Let's first start from the definition of Euclidean distance, which measures the length of any straight line connecting two points:



Vector lengths

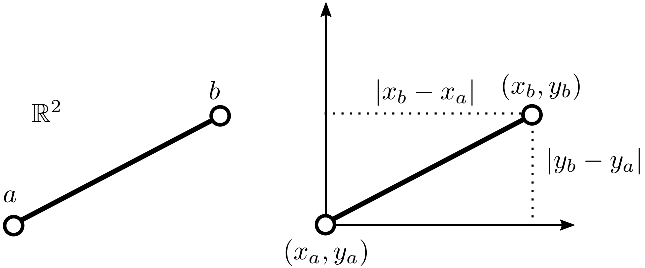
How to measure the length of the gradient? Let's first start from the definition of Euclidean distance, which measures the length of any straight line connecting two points:



Apply Pythagoras' theorem: $d(a, b) = (|x_b - x_a|^2 + |y_b - y_a|^2)^{\frac{1}{2}}$

Vector lengths

How to measure the length of the gradient? Let's first start from the definition of Euclidean distance, which measures the length of any straight line connecting two points:



Apply Pythagoras' theorem: $d(a,b) = (|x_b - x_a|^2 + |y_b - y_a|^2)^{\frac{1}{2}}$

In matrix notation:

$$d(\mathbf{a}, \mathbf{b}) = \|\mathbf{a} - \mathbf{b}\|_2$$

where $\mathbf{a} = \begin{pmatrix} x_a \\ y_a \end{pmatrix}$ and $\mathbf{b} = \begin{pmatrix} x_b \\ y_b \end{pmatrix}$

Vector lengths

Thus, with this definition of distance:

$$d(\mathbf{a}, \mathbf{b}) = \|\mathbf{a} - \mathbf{b}\|_2$$

We can simply define the length of a vector $\mathbf{x}$ as the distance from the origin to $\mathbf{x}$:

$$\|\mathbf{x} - \mathbf{0}\|_2 = \|\mathbf{x}\|_2 = \sqrt{\mathbf{x}^\top \mathbf{x}}$$

Vector lengths

Thus, with this definition of distance:

$$d(\mathbf{a}, \mathbf{b}) = \|\mathbf{a} - \mathbf{b}\|_2$$

We can simply define the length of a vector $\mathbf{x}$ as the distance from the origin to $\mathbf{x}$:

$$\|\mathbf{x} - \mathbf{0}\|_2 = \|\mathbf{x}\|_2 = \sqrt{\mathbf{x}^\top \mathbf{x}}$$

Often, for simplicity and to avoid computing square roots, we will consider squared distances and norms:

$$\|\mathbf{x}\|_2^2 = \mathbf{x}^\top \mathbf{x}$$

Least squares

Linear regression: Finding a solution

$$\min_{a,b\in\mathbb{R}} \sum_{i=1}^{n} (y_i - a x_i - b)^2$$

Linear regression: Finding a solution

$$\mathbf{\Theta}^* = \arg\min_{\mathbf{\Theta} \in \mathbb{R}^2} \ell(\mathbf{\Theta})$$

where $\ell : \mathbb{R}^2 \to \mathbb{R}$ is defined as:

$$\ell(a, b) = \sum_{i=1}^{n} (y_i - a x_i - b)^2$$

Linear regression: Finding a solution

$$\mathbf{\Theta}^* = \arg\min_{\mathbf{\Theta}\in\mathbb{R}^2} \ell(\mathbf{\Theta})$$

where $\ell : \mathbb{R}^2 \to \mathbb{R}$ is defined as:

$$\ell(a,b) = \sum_{i=1}^{n} (y_i - ax_i - b)^2$$

A solution is found by setting $\nabla_{\mathbf{\Theta}} \ell(\mathbf{\Theta}) = \mathbf{0}$:

$$\nabla_{\mathbf{\Theta}} \sum_{i=1}^{n} (y_i - ax_i - b)^2 = \sum_{i=1}^{n} \nabla_{\mathbf{\Theta}} (y_i - ax_i - b)^2$$

Linear regression: Finding a solution

$$\boldsymbol{\Theta}^* = \arg\min_{\boldsymbol{\Theta} \in \mathbb{R}^2} \ell(\boldsymbol{\Theta})$$

where $\ell : \mathbb{R}^2 \to \mathbb{R}$ is defined as:

$$\ell(a, b) = \sum_{i=1}^{n} (y_i - a x_i - b)^2$$

A solution is found by setting $\nabla_{\boldsymbol{\Theta}} \ell(\boldsymbol{\Theta}) = \mathbf{0}$:

$$\begin{aligned}
\nabla_{\boldsymbol{\Theta}} \sum_{i=1}^{n} (y_i - a x_i - b)^2 &= \sum_{i=1}^{n} \nabla_{\boldsymbol{\Theta}} (y_i - a x_i - b)^2 \\
&= \sum_{i=1}^{n} \nabla_{\boldsymbol{\Theta}} (\cancel{y_i^2} + a^2 x_i^2 + b^2 - 2a x_i y_i - 2b y_i + 2ab x_i)
\end{aligned}$$

Linear regression: Finding a solution

$$\boldsymbol{\Theta}^* = \arg\min_{\boldsymbol{\Theta}\in\mathbb{R}^2} \ell(\boldsymbol{\Theta})$$

where $\ell : \mathbb{R}^2 \to \mathbb{R}$ is defined as:

$$\ell(a, b) = \sum_{i=1}^{n}(y_i - ax_i - b)^2$$

A solution is found by setting $\nabla_{\boldsymbol{\Theta}}\ell(\boldsymbol{\Theta}) = \mathbf{0}$:

$$\begin{aligned}
\nabla_{\boldsymbol{\Theta}} \sum_{i=1}^{n}(y_i - ax_i - b)^2 &= \sum_{i=1}^{n} \nabla_{\boldsymbol{\Theta}}(y_i - ax_i - b)^2 \\
&= \sum_{i=1}^{n} \nabla_{\boldsymbol{\Theta}}(\cancel{y_i^2} + a^2x_i^2 + b^2 - 2ax_iy_i - 2by_i + 2abx_i) \\
&= \sum_{i=1}^{n} \begin{pmatrix} 2ax_i^2 - 2x_iy_i + 2bx_i \\ 2b - 2y_i + 2ax_i \end{pmatrix}
\end{aligned}$$

Linear regression: Finding a solution

$$\mathbf{\Theta}^* = \arg\min_{\mathbf{\Theta} \in \mathbb{R}^2} \ell(\mathbf{\Theta})$$

where $\ell : \mathbb{R}^2 \to \mathbb{R}$ is defined as:

$$\ell(a, b) = \sum_{i=1}^{n} (y_i - ax_i - b)^2$$

A solution is found by setting $\nabla_{\mathbf{\Theta}} \ell(\mathbf{\Theta}) = \mathbf{0}$:

$$\begin{aligned}
\nabla_{\mathbf{\Theta}} \sum_{i=1}^{n} (y_i - ax_i - b)^2 &= \sum_{i=1}^{n} \nabla_{\mathbf{\Theta}} (y_i - ax_i - b)^2 \\
&= \sum_{i=1}^{n} \nabla_{\mathbf{\Theta}} (\cancel{y_i^2} + a^2 x_i^2 + b^2 - 2ax_i y_i - 2by_i + 2abx_i) \\
&= \begin{pmatrix} \sum_{i=1}^{n} 2ax_i^2 - 2x_i y_i + 2bx_i \\ \sum_{i=1}^{n} 2b - 2y_i + 2ax_i \end{pmatrix}
\end{aligned}$$

Linear regression: Finding a solution

$$\mathbf{\Theta}^* = \arg\min_{\mathbf{\Theta}\in\mathbb{R}^2} \ell(\mathbf{\Theta})$$

where $\ell : \mathbb{R}^2 \to \mathbb{R}$ is defined as:

$$\ell(a,b) = \sum_{i=1}^{n}(y_i - ax_i - b)^2$$

A solution is found by setting $\nabla_{\mathbf{\Theta}}\ell(\mathbf{\Theta}) = \mathbf{0}$:

$$\nabla_{\mathbf{\Theta}} \sum_{i=1}^{n}(y_i - ax_i - b)^2 = \begin{pmatrix} \sum_{i=1}^{n} 2ax_i^2 - 2x_iy_i + 2bx_i \\ \sum_{i=1}^{n} 2b - 2y_i + 2ax_i \end{pmatrix}$$

We get 2 linear equations in the 2 unknowns a, b:

$$\begin{pmatrix} \sum_{i=1}^{n} ax_i^2 + bx_i - x_iy_i \\ \sum_{i=1}^{n} ax_i + b - y_i \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

We need to familiarize with matrix calculus.

When we use a numerical method, we manipulate matrices and vectors.

Linear regression: Matrix notation

The linear regression problem is so called, because it is linear in the parameters a, b.

This is evident if we switch to matrix notation:

$$\underbrace{\begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix}}_{\mathbf{y}} = \underbrace{\begin{pmatrix} x_1 & 1 \\ x_2 & 1 \\ \vdots & \vdots \\ x_n & 1 \end{pmatrix}}_{\mathbf{X}} \underbrace{\begin{pmatrix} a \\ b \end{pmatrix}}_{\boldsymbol{\theta}}$$

Linear regression: Matrix notation

$$\underbrace{\begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix}}_{\mathbf{y}} = \underbrace{\begin{pmatrix} x_1 & 1 \\ x_2 & 1 \\ \vdots & \vdots \\ x_n & 1 \end{pmatrix}}_{\mathbf{X}} \underbrace{\begin{pmatrix} a \\ b \end{pmatrix}}_{\boldsymbol{\theta}}$$

This expresses all the equations $y_i = ax_i + b$ at once and makes the linearity w.r.t. a, b evident.

The MSE is simply:

$$\ell(\boldsymbol{\theta}) = \|\mathbf{y} - \mathbf{X}\boldsymbol{\theta}\|_2^2$$

Linear regression: Matrix notation

$$\underbrace{\begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix}}_{\mathbf{y}} = \underbrace{\begin{pmatrix} x_1 & 1 \\ x_2 & 1 \\ \vdots & \vdots \\ x_n & 1 \end{pmatrix}}_{\mathbf{X}} \underbrace{\begin{pmatrix} a \\ b \end{pmatrix}}_{\boldsymbol{\theta}}$$

This expresses all the equations $y_i = ax_i + b$ at once and makes the linearity w.r.t. a, b evident.

The MSE is simply:

$$\ell(\boldsymbol{\theta}) = (\mathbf{y} - \mathbf{X}\boldsymbol{\theta})^\top (\mathbf{y} - \mathbf{X}\boldsymbol{\theta})$$

Linear regression: Matrix notation

$$\underbrace{\begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix}}_{\mathbf{y}} = \underbrace{\begin{pmatrix} x_1 & 1 \\ x_2 & 1 \\ \vdots & \vdots \\ x_n & 1 \end{pmatrix}}_{\mathbf{X}} \underbrace{\begin{pmatrix} a \\ b \end{pmatrix}}_{\boldsymbol{\theta}}$$

This expresses all the equations $y_i = ax_i + b$ at once and makes the linearity w.r.t. a, b evident.

The MSE is simply:

$$\ell(\boldsymbol{\theta}) = \mathbf{y}^\top \mathbf{y} - 2\mathbf{y}^\top \mathbf{X}\boldsymbol{\theta} + \boldsymbol{\theta}^\top \mathbf{X}^\top \mathbf{X}\boldsymbol{\theta}$$

Linear regression: Matrix notation

$$\underbrace{\begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix}}_{\mathbf{y}} = \underbrace{\begin{pmatrix} x_1 & 1 \\ x_2 & 1 \\ \vdots & \vdots \\ x_n & 1 \end{pmatrix}}_{\mathbf{X}} \underbrace{\begin{pmatrix} a \\ b \end{pmatrix}}_{\boldsymbol{\theta}}$$

This expresses all the equations $y_i = ax_i + b$ at once and makes the linearity w.r.t. a, b evident.

The MSE is simply:

$$\ell(\boldsymbol{\theta}) = \mathbf{y}^\top \mathbf{y} - 2\mathbf{y}^\top \mathbf{X}\boldsymbol{\theta} + \boldsymbol{\theta}^\top \mathbf{X}^\top \mathbf{X}\boldsymbol{\theta}$$

Setting the gradient w.r.t. $\boldsymbol{\theta}$ to zero:

$$-2\mathbf{X}^\top \mathbf{y} + 2\mathbf{X}^\top \mathbf{X}\boldsymbol{\theta} = \mathbf{0}$$

Linear regression: Matrix notation

$$\underbrace{\begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix}}_{\mathbf{y}} = \underbrace{\begin{pmatrix} x_1 & 1 \\ x_2 & 1 \\ \vdots & \vdots \\ x_n & 1 \end{pmatrix}}_{\mathbf{X}} \underbrace{\begin{pmatrix} a \\ b \end{pmatrix}}_{\boldsymbol{\theta}}$$

This expresses all the equations $y_i = ax_i + b$ at once and makes the linearity w.r.t. a, b evident.

The MSE is simply:

$$\ell(\boldsymbol{\theta}) = \mathbf{y}^\top \mathbf{y} - 2\mathbf{y}^\top \mathbf{X}\boldsymbol{\theta} + \boldsymbol{\theta}^\top \mathbf{X}^\top \mathbf{X}\boldsymbol{\theta}$$

Setting the gradient w.r.t. $\boldsymbol{\theta}$ to zero:

$$\mathbf{X}^\top \mathbf{X}\boldsymbol{\theta} = \mathbf{X}^\top \mathbf{y}$$

Linear regression: Matrix notation

$$\underbrace{\begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix}}_{\mathbf{y}} = \underbrace{\begin{pmatrix} x_1 & 1 \\ x_2 & 1 \\ \vdots & \vdots \\ x_n & 1 \end{pmatrix}}_{\mathbf{X}} \underbrace{\begin{pmatrix} a \\ b \end{pmatrix}}_{\boldsymbol{\theta}}$$

This expresses all the equations $y_i = ax_i + b$ at once and makes the linearity w.r.t. a, b evident.

The MSE is simply:

$$\ell(\boldsymbol{\theta}) = \mathbf{y}^\top \mathbf{y} - 2\mathbf{y}^\top \mathbf{X}\boldsymbol{\theta} + \boldsymbol{\theta}^\top \mathbf{X}^\top \mathbf{X}\boldsymbol{\theta}$$

Setting the gradient w.r.t. $\boldsymbol{\theta}$ to zero:

$$\boldsymbol{\theta} = (\mathbf{X}^\top \mathbf{X})^{-1} \mathbf{X}^\top \mathbf{y}$$

We get a closed form solution to our problem.

Linear regression: Higher dimensions

In the general case, the data points $(\mathbf{x}_i, \mathbf{y}_i)$ are vectors in $\mathbb{R}^d$:

$$\mathbf{y}_i = \mathbf{A}\mathbf{x}_i + \mathbf{b} \quad \text{for } i = 1, \ldots, n$$

Linear regression: Higher dimensions

In the general case, the data points $(\mathbf{x}_i, \mathbf{y}_i)$ are vectors in $\mathbb{R}^d$:

$$\mathbf{y}_i = \mathbf{A}\mathbf{x}_i + \mathbf{b} \quad \text{for } i = 1, \ldots, n$$

Stacking all data points into matrices $\tilde{\mathbf{X}} = \begin{pmatrix} | & | & \\ \mathbf{x}_1 & \mathbf{x}_2 & \cdots \\ | & | & \end{pmatrix}$ and $\mathbf{Y}$, we get:

$$\underbrace{\begin{pmatrix} y_{11} & \cdots & y_{1d} \\ y_{21} & \cdots & y_{2d} \\ \vdots & & \vdots \\ y_{n1} & \cdots & y_{nd} \end{pmatrix}}_{\mathbf{Y}^\top} = \underbrace{\begin{pmatrix} x_{11} & \cdots & x_{1d} & 1 \\ x_{21} & \cdots & x_{2d} & 1 \\ \vdots & & \vdots & \vdots \\ x_{n1} & \cdots & x_{nd} & 1 \end{pmatrix}}_{\mathbf{X}^\top := (\tilde{\mathbf{X}}^\top | \mathbf{1})} \underbrace{\begin{pmatrix} a_{11} & \cdots & a_{1d} \\ \vdots & & \vdots \\ a_{d1} & \cdots & a_{dd} \\ b_1 & \cdots & b_d \end{pmatrix}}_{\boldsymbol{\Theta}}$$

According to which, for each output data point $\mathbf{y}_i$ we have:

$$\underbrace{\begin{pmatrix} y_{i1} \\ \vdots \\ y_{id} \end{pmatrix}}_{\mathbf{y}_i} = \begin{pmatrix} \sum_{j=1}^d a_{j1} x_{ij} + b_1 \\ \vdots \\ \sum_{j=1}^d a_{jd} x_{ij} + b_d \end{pmatrix}$$

Linear regression: Higher dimensions

In the general case, the data points $(\mathbf{x}_i, \mathbf{y}_i)$ are vectors in $\mathbb{R}^d$:

$$\mathbf{y}_i = \mathbf{A}\mathbf{x}_i + \mathbf{b} \quad \text{for } i = 1, \ldots, n$$

Stacking all data points into matrices $\tilde{\mathbf{X}} = \begin{pmatrix} | & | & \\ \mathbf{x}_1 & \mathbf{x}_2 & \cdots \\ | & | & \end{pmatrix}$ and $\mathbf{Y}$, we get:

$$\underbrace{\begin{pmatrix} y_{11} & \cdots & y_{1d} \\ y_{21} & \cdots & y_{2d} \\ \vdots & & \vdots \\ y_{n1} & \cdots & y_{nd} \end{pmatrix}}_{\mathbf{Y}^\top} = \underbrace{\begin{pmatrix} x_{11} & \cdots & x_{1d} & 1 \\ x_{21} & \cdots & x_{2d} & 1 \\ \vdots & & \vdots & \vdots \\ x_{n1} & \cdots & x_{nd} & 1 \end{pmatrix}}_{\mathbf{X}^\top := (\tilde{\mathbf{X}}^\top | \mathbf{1})} \underbrace{\begin{pmatrix} a_{11} & \cdots & a_{1d} \\ \vdots & & \vdots \\ a_{d1} & \cdots & a_{dd} \\ b_1 & \cdots & b_d \end{pmatrix}}_{\boldsymbol{\Theta}}$$

The MSE reads:

$$\ell(\boldsymbol{\Theta}) = \|\mathbf{Y}^\top - \mathbf{X}^\top\boldsymbol{\Theta}\|_2^2 = \mathrm{tr}(\mathbf{Y}^\top\mathbf{Y}) - 2\mathrm{tr}(\mathbf{Y}\mathbf{X}^\top\boldsymbol{\Theta}) + \mathrm{tr}(\boldsymbol{\Theta}^\top\mathbf{X}\mathbf{X}^\top\boldsymbol{\Theta})$$

Linear regression: Higher dimensions

In the general case, the data points $(\mathbf{x}_i, \mathbf{y}_i)$ are vectors in $\mathbb{R}^d$:

$$\mathbf{y}_i = \mathbf{A}\mathbf{x}_i + \mathbf{b} \quad \text{for } i = 1, \dots, n$$

Stacking all data points into matrices $\tilde{\mathbf{X}} = \begin{pmatrix} | & | & \\ \mathbf{x}_1 & \mathbf{x}_2 & \cdots \\ | & | & \end{pmatrix}$ and $\mathbf{Y}$, we get:

$$\underbrace{\begin{pmatrix} y_{11} & \cdots & y_{1d} \\ y_{21} & \cdots & y_{2d} \\ \vdots & & \vdots \\ y_{n1} & \cdots & y_{nd} \end{pmatrix}}_{\mathbf{Y}^\top} = \underbrace{\begin{pmatrix} x_{11} & \cdots & x_{1d} & 1 \\ x_{21} & \cdots & x_{2d} & 1 \\ \vdots & & \vdots & \vdots \\ x_{n1} & \cdots & x_{nd} & 1 \end{pmatrix}}_{\mathbf{X}^\top := (\tilde{\mathbf{X}}^\top | \mathbf{1})} \underbrace{\begin{pmatrix} a_{11} & \cdots & a_{1d} \\ \vdots & & \vdots \\ a_{d1} & \cdots & a_{dd} \\ b_1 & \cdots & b_d \end{pmatrix}}_{\boldsymbol{\Theta}}$$

The MSE reads:

$$\ell(\boldsymbol{\Theta}) = \|\mathbf{Y}^\top - \mathbf{X}^\top \boldsymbol{\Theta}\|_2^2 = \mathrm{tr}(\mathbf{Y}^\top \mathbf{Y}) - 2\mathrm{tr}(\mathbf{Y}\mathbf{X}^\top \boldsymbol{\Theta}) + \mathrm{tr}(\boldsymbol{\Theta}^\top \mathbf{X}\mathbf{X}^\top \boldsymbol{\Theta})$$

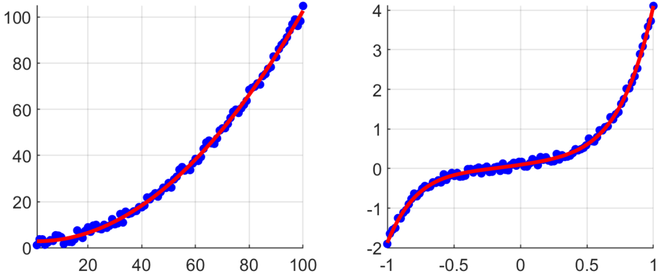
The closed form solution of $\nabla_{\boldsymbol{\Theta}} \ell(\boldsymbol{\Theta}) = \mathbf{0}$ is:

$$\boldsymbol{\Theta} = (\mathbf{X}\mathbf{X}^\top)^{-1} \mathbf{X}\mathbf{Y}^\top$$

Fitting polynomials

Polynomial regression

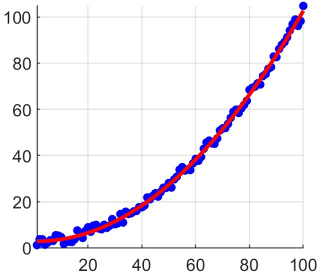
Instead of linear functions, can we fit higher-order polynomials?



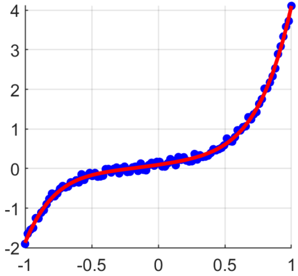
2nd-order **5th**-order

Polynomial regression

Instead of linear functions, can we fit higher-order polynomials?



2nd-order

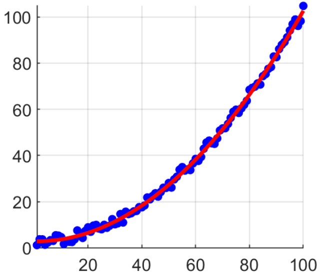


5th-order

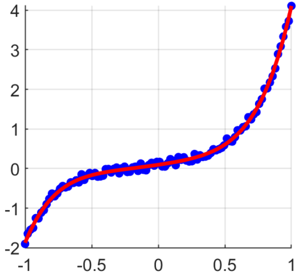
The number of parameters grows with the order.

Polynomial regression

Instead of linear functions, can we fit higher-order polynomials?



2nd-order



5th-order

The number of parameters grows with the order.

More data are needed to make an informed decision on the order.

Polynomial regression

Remark: Despite the name, polynomial regression is still linear in the parameters. It is polynomial with respect to the data.

Linear regression with polynomial features

Remark: Despite the name, polynomial regression is still linear in the parameters. It is polynomial with respect to the data.

Linear regression with polynomial features

Remark: Despite the name, polynomial regression is still linear in the parameters. It is polynomial with respect to the data.

$$y_i = a_3 x_i^3 + a_2 x_i^2 + a_1 x_i + b \quad \text{for all data points } i = 1, \ldots, n$$

Linear regression with polynomial features

Remark: Despite the name, polynomial regression is still linear in the parameters. It is polynomial with respect to the data.

$$y_i = b + \sum_{j=1}^{k} a_j x_i^j \quad \text{for all data points } i = 1, \dots, n$$

Linear regression with polynomial features

Remark: Despite the name, polynomial regression is still linear in the parameters. It is polynomial with respect to the data.

$$y_i = b + \sum_{j=1}^{k} a_j x_i^j \quad \text{for all data points } i = 1, \ldots, n$$

Linear regression with polynomial features

Remark: Despite the name, polynomial regression is still linear in the parameters. It is polynomial with respect to the data.

$$y_i = b + \sum_{j=1}^{k} a_j x_i^j \quad \text{for all data points } i = 1, \ldots, n$$

In matrix notation:

$$\underbrace{\begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix}}_{\mathbf{y}} = \underbrace{\begin{pmatrix} x_1^k & x_1^{k-1} & \cdots & x_1 & 1 \\ x_2^k & x_2^{k-1} & \cdots & x_2 & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ x_n^k & x_n^{k-1} & \cdots & x_n & 1 \end{pmatrix}}_{\mathbf{X}} \underbrace{\begin{pmatrix} a_k \\ a_{k-1} \\ \vdots \\ a_1 \\ b \end{pmatrix}}_{\boldsymbol{\theta}}$$

Linear regression with polynomial features

Remark: Despite the name, polynomial regression is still linear in the parameters. It is polynomial with respect to the data.

$$y_i = b + \sum_{j=1}^{k} a_j x_i^j \quad \text{for all data points } i = 1, \ldots, n$$

In matrix notation:

$$\underbrace{\begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix}}_{\mathbf{y}} = \underbrace{\begin{pmatrix} x_1^k & x_1^{k-1} & \cdots & x_1 & 1 \\ x_2^k & x_2^{k-1} & \cdots & x_2 & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ x_n^k & x_n^{k-1} & \cdots & x_n & 1 \end{pmatrix}}_{\mathbf{X}} \underbrace{\begin{pmatrix} a_k \\ a_{k-1} \\ \vdots \\ a_1 \\ b \end{pmatrix}}_{\boldsymbol{\theta}}$$

The same exact least-squares solution as with linear regression applies, with the requirement that $k < n$.

Polynomial fitting

An application of the Stone-Weierstrass theorem tells us:

If f is continuous on the interval $[a, b]$, then for every $\epsilon > 0$ there exists a polynomial p such that $|f(x) - p(x)| < \epsilon$ for all x.

Polynomial fitting

An application of the Stone-Weierstrass theorem tells us:

If f is continuous on the interval $[a, b]$, then for every $\epsilon > 0$ there exists a polynomial p such that $|f(x) - p(x)| < \epsilon$ for all x.

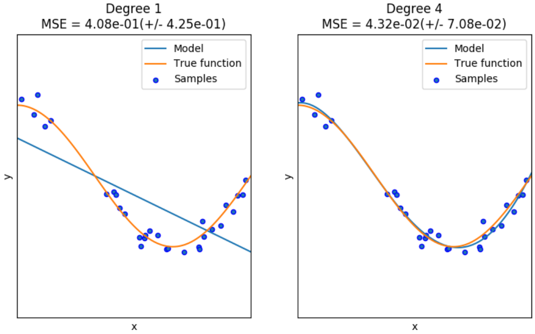
Thus, we can try to fit a polynomial in many cases.

Polynomial fitting

An application of the Stone-Weierstrass theorem tells us:

If f is continuous on the interval $[a, b]$, then for every $\epsilon > 0$ there exists a polynomial p such that $|f(x) - p(x)| < \epsilon$ for all x.

Thus, we can try to fit a polynomial in many cases.

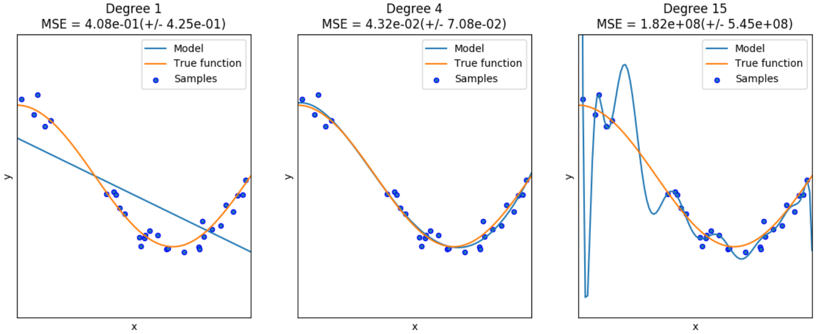


Polynomial fitting

An application of the Stone-Weierstrass theorem tells us:

If f is continuous on the interval $[a, b]$, then for every $\epsilon > 0$ there exists a polynomial p such that $|f(x) - p(x)| < \epsilon$ for all x.

Thus, we can try to fit a polynomial in many cases.



Suggested reading

For convexity and optimality, read Sections 3.1.1 and 3.1.3 of the book:

S. Boyd & L. Vandenberghe, "Convex optimization". Cambridge University Press, 2009

Public download link: `https://web.stanford.edu/~boyd/cvxbook/bv_cvxbook.pdf`