approximation

November 3, 2025

0.1 Notebook 3: Approximation

[ ]:
```
using LaTeXStrings
using LinearAlgebra
using Plots
using Polynomials
```

0.1.1 [Exercise 1] Least squares approximation

The objective of this exercise is to approximate some given data `x`, `y` using a polynomial of a given degree, potentially much lower than the number of data points.

1. Without using any library, write a function `polyfit(x, y, d)` which, given data points $(x_1, y_1), \dots, (x_N, y_N)$ and an integer $0 \leq d \leq N-1$, calculates the unique polynomial $p \in \mathbb{R}[x]$ of degree at most d minimizing the total error

$$E := \frac{1}{2} \sum_{n=0}^{N} \left| p(x_n) - y_n \right|^2.$$

Your function should return a vector containing the $d+1$ coefficients of p, starting from the constant term.

Hint (click to display)

Within the function, you can proceed as follows. First, create the following matrix and vector:

$$\mathbf{A} = \begin{pmatrix} 1 & x_0 & \dots & x_0^d \\ \vdots & \vdots & & \vdots \\ 1 & x_N & \dots & x_N^d \end{pmatrix}, \qquad \mathbf{b} = \begin{pmatrix} y_0 \\ \vdots \\ y_N \end{pmatrix}.$$

Note that the error E rewrites as follows:

$$E(\alpha) = \frac{1}{2} \left\| \mathsf{A}\alpha - b \right\|^2,$$

where α is a vector containing the coefficients of the polynomial, in order of increasing degree, and $\|\cdot\|$ is the Euclidean norm. In other words, the function E is a quadratic function of the vector of coefficients of the polynomial. Writing $\nabla E(\alpha) = 0$ leads to the so-called **normal equations**:

$$\mathsf{A}^\top \mathsf{A} \alpha = \mathsf{A}^\top \mathbf{b}.$$

This is a linear system with a square invertible matrix on the left-hand side, which can be solved using the backslash operator `\`; in fact you can write just `A\b` instead of `(A'A)\(A'b)`, because the operator `\` performs a least squares approximation by default.

[ ]:
```
function polyfit(x, y, d)
    # Your code here
end
```

[ ]:
```
n = 10 ; x = 1:n ; y = randn(n)
@assert polyfit([0.], [0.], 0)   [0.]
@assert polyfit(1:5, 1:5, 1)   [0., 1.]
@assert polyfit(x, y, 0)   [sum(y)/n]
@assert polyfit(x, y, 0)   [sum(y)/n]
@assert polyfit(x, y, 2)   fit(x, y, 2).coeffs
```

2. Write also a function `polyval( , X)` to evaluate the polynomial

$$p(x) = \alpha_0 + \alpha_1 x + ... + \alpha_d x^d$$

at all the points in `X`. The function should return the result in a vector.

[ ]:
```
function polyval( , X)
    # Your code here
end
```

[ ]:
```
n = 10 ;   = randn(n)
@assert polyval([0.], [1., 2., 3.]) == [0., 0., 0.]
@assert polyval( , [0., 1.])   [ [1], sum( )]
```

Use the data given below, of the altitude of a marble in free fall as a function of time on a remote planet, to test your code. The experiment was performed on a different planet. Can you find which one? See Gravitational acceleration.

[ ]:
```
# Time since dropping the marble
x = [0., 1., 2., 3., 4., 5.]

# Altitude of the marble
y = [100., 98.26, 93.56, 81.79, 71.25, 53.22]

# Fit using polyfit
  = polyfit(x, y, 2)

# Evalutate at X
X = LinRange(0, 5, 200)
Y = polyval( , X)

plot(X, Y, label="My approximation")
scatter!(x, y, label="Data")
```

Least squares approximation using `Polynomials.fit` (click to display)

We learnt in the previous notebook that `Polynomials.fit` could be employed for polynomial interpolation. This function can also be used for fitting data by minimizing the sum of squared residuals, which can be achieved as follows:

[ ]:
```
# This returns structure of type `Polynomial`, with associated degree 2
p = fit(x, y, 2)
@show p

# The coefficients can be obtained as follows
@show p.coeffs

# The structure `p` behaves like a function
@show p(0)

X = LinRange(0, 5, 200)
plot(X, p.(X), label="Polynomials.jl approximation")
```

0.1.2 [Exercise 2] Least squares approximation for data analysis

The dataset loaded through the following Julia commands contains data on the vapor pressure of mercury as a function of the temperature.

[ ]:
```
import RDatasets
data = RDatasets.dataset("datasets", "pressure")
```

Using `Polynomials.jl`, find a low-dimensional mathematical model of the form

$$p(T) = \exp(\alpha_0 + \alpha_1 T + \alpha_2 T^2 + \alpha_3 T^3) \tag{1}$$

for the pressure as a function of the temperature. Plot the approximation together with the data.

[ ]:
```

```