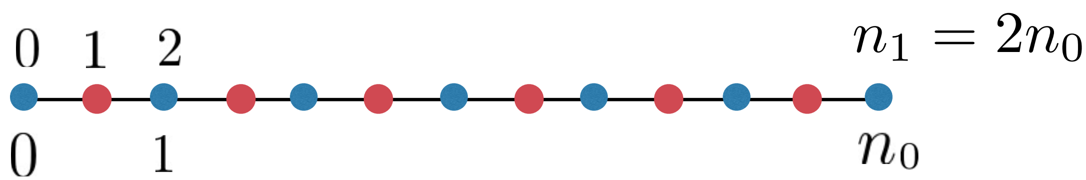
Romberg integration

Idea: Use two or more trapezoidal integral estimates, extrapolate

- step sizes (decreasing order) $h_0, h_1, \ldots, h_m$, integral estimates $I_0, I_1, \ldots, I_m$
- use polynomial of order n to fit and extrapolate to h=0
- error for given h scales as h^2 (+ higher even powers only)
- use polynomial P(x) with $x=h^2$

Simplest case: 2 points (m=1), using $h_0=(b-a)/n_0$ and $h_1=h_0/2$ ($x_1=x_0/4$)



Function evaluation once only for each point needed

$$I_0 = I_\infty + \epsilon x_0, \quad I_1 = I_\infty + \epsilon x_0/4$$

$$\rightarrow \quad I_\infty = \frac{4}{3} I_1 - \frac{1}{3} I_0 \quad + O(h_0^4) \quad [O(x_0^2)]$$

reducing h by 50%

- error should be 1/4 of previous
- ε is unknown factor, eliminated

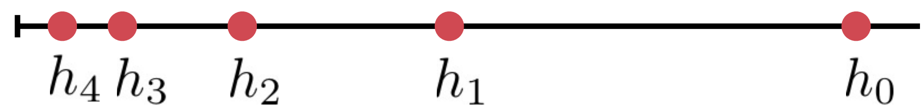
Computation cost doubled, error reduced by two powers of h_0!

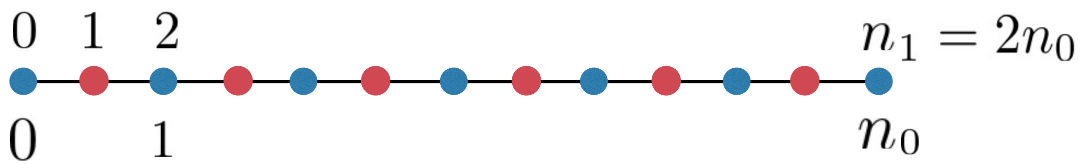
Generalizes easily to the case of m estimates (Friday)

General case; $h_0, h_1, \ldots, h_m \rightarrow I_1, I_2, \ldots, I_m$

For each i, let $h_{i+1} = h_i/2$ ($x_{i+1} = x_i/4$)

- save old sum, add new points





How to construct a polynomial of order n going throug n+1 point pairs (x_i,y_i)

$$P(x) = \sum_{i=0}^{m} y_i \prod_{k\neq i} \frac{x - x_k}{x_i - x_k}$$

$x_i=h_i^2$

Evaluate (this is the extrapolation) at x=0 (h=0)

$$I_\infty = \sum_{i=0}^{m} I_i \prod_{k\neq i} \frac{-h_0^2 2^{-2k}}{h_0^2(2^{-2i} - 2^{-2k})} = (-1)^m \sum_{i=0}^{m} I_i \prod_{k\neq i} \frac{1}{2^{2(k-i)} - 1}$$

Error decreases very rapidly: $O(h^{2(m+1)})$

Implemented in romberg.jl (both closed and open cases)