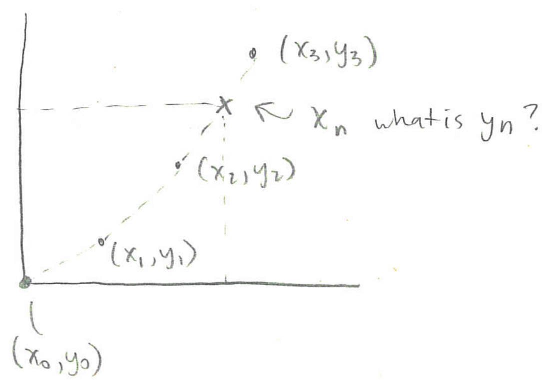
Lecture 18 - Interpolation

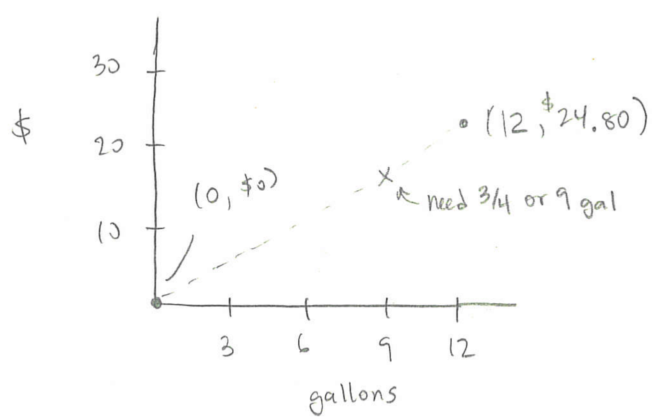
* Prayer / Quiz / AMA

I. Concept of Interpolation

* In interpolation, we have some data $\{x_i, y_i\}$ at discrete points, and we want to know the value of y at some x not in our data set.



Example: Suppose you know it costs \$24.80 to fill up your car which has a 12 gallon tank. How much will it cost to fill up your car if it is 1/4 full?



Try and calculate the answer.

Talk w/ your neighbor to figure it out.

• Record answers on board. Don't tell correct answer yet.

(\$18.60)

* In interpolation we want to fit many simple, piecewise functions that exactly go through all the data.

* we usually use piecewise polynomials

- why polynomials? The polynomial uniquess theorem tells us that we can fit n+1 points exactly with an n^{th} order polynomial.

  e.g. line (1^{st} order) → 2 points

  quadratic (2^{nd} order) → 3 points

  cubic (3^{rd} order) → 4 points

- why piecewise? Runge's (R-oo-ng-a) phenomenon appear (wild oscillations) when we try to fit a high order polynomial to many data points.

* A piecewise, polynomial interpolation that exactly fits every data point is called a spline.

- linear spline (continuous function) ← easiest
- quadratic spline (continuous f, f')
- cubic spline (continuous f, f', f'') ← also very common, good tradeoff between smoothness & simplicity

* Fitting vs. Interpolation

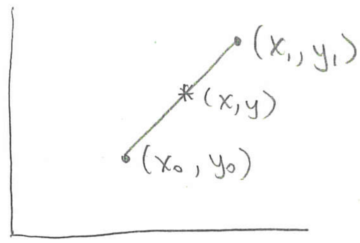
Fitting	Interpolation
Single model	Piecewise Polynomials
Physically Informed	Purely empirical
Noisy Data	Smooth Data
Approximate Fit	Exact Fit.

Python Examples of Interpolation Concepts

II. Interpolation by Hand & in Excel

* Linear Interpolation (a linear spline) is very useful.
You should know how to do this by hand.

* Excel doesn't have an interpolation function. So, you will need to hand code the formula if you need it.



Derivation: set slopes equal

$$\frac{y-y_0}{x-x_0} = \frac{y_1-y_0}{x_1-x_0}$$

solve for y

$$y = y_0 + \frac{y_1-y_0}{x_1-x_0}(x-x_0)$$

$$= y_0\frac{(x_1-x_0)}{(x_1-x_0)} - y_0\frac{(x-x_0)}{(x_1-x_0)} + y_1\frac{(x-x_0)}{x_1-x_0}$$

$$\boxed{y = y_0\frac{(x_1-x)}{(x_1-x_0)} + y_1\frac{(x-x_0)}{x_1-x_0}}$$

(a weighted average based on distance from points)

Example from page 1:

$x_0 = 0$ $\quad x_1 = 12\text{ gal}$ $\quad x = 3/4 \cdot 12\text{ gal} = 9\text{ gal}$

$y_0 = \$0$ $\quad y_1 = \$24.80$

$$y = \$0 \cdot \frac{(12-9)\text{ gal}}{(12-0)\text{ gal}} + \$24.80 \cdot \frac{(9-0)\text{ gal}}{(12-0)\text{ gal}}$$

$$y = \frac{3}{4}\cdot \$24.80 = \$18.60$$

Example provided in Excel
(but not discussed in class)

III. Interpolation in Python

Activity

* Demonstration of linear interpolation using scipy.interpolate.interp1d
* Demonstration of cubic spline using scipy.interpolate.interp1d
* Practice with a linear & cubic spline (same as Excel problem).