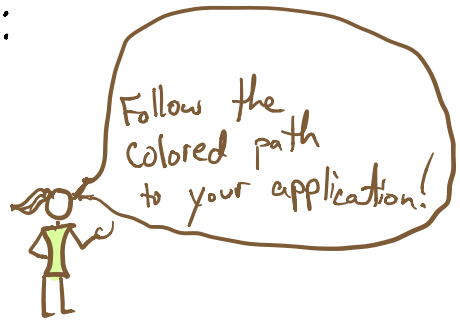
Integration in Python sketchnote

By John Kitchin

There are 3 main applications:

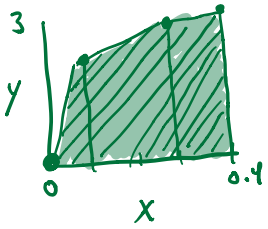


1. Integrating Data
2. Integrating Functions
3. Solving differential equations

Integrating data

Data is represented in arrays:

```
x = [0, 0.1, 0.3, 0.4]
y = [0, 2.0, 2.5, 3.0]
```



//// ≈ the area under the curve defined by the points. using the trapezoid rule

We use `numpy.trapz(y, x)`

or

`scipy.integrate.simps(y, x)`

to approximate $\int_{x_0}^{x_f} y(x)\,dx$

Integrating functions

If you know f(x) in analytical form, and can write a Python function for it, e.g.

```
def f(x):
    return some function of x
```

Then we use scipy.integrate.quad

$$\int_a^b f(x)\,dx = \text{quad}(f, a, b)$$

a, b must be numbers, or numpy.inf for infinity

quad returns the integral <u>and</u> an error estimate.

```
I, err = quad(f, a, b)
```

This assumes f(x) is well-behaved on the interval (a, b)

Integrating differential equations

When we have a first order differential equation that looks like:

$$y' = f(x, y) \quad \text{with} \quad y(x_0) = y_0$$

All the y' have to be on the left

only functions of x, y on the right

These are not OK: not 1st order

$$y'' = f(x, y, y')$$

$$y' = f(x, y, y')$$

y' on the right.

Then we use scipy.integrate.solve_ivp to get the solution $y(x)$, That solution looks like an array of x (or t) values, and the corresponding y-values.

```
sol = solve_ivp(f, tspan, y0)
```

- f: the f(x,y)
- tspan: (x_0, x_f) integration range
- y0: Array of initial conditions (can be a list)

↓

Data structure

sol.t = 1D array of t points we got the solution for

sol.y = 2D array of y values, each row is a solution to the equations

for 1 equation the yvalues are in sol.y[0]

The solver chooses the t-points adaptively to get an accurate answer at the endpoint. To get the points we want use the t_eval optional argument like this.

unpacks X_0 to start, X_f to end

```
tspan = X0, Xf
sol = solve_ivp(f, tspan, y0, t_eval = np.linspace(*tspan))
```

then sol.t will have all the points you put into t_eval and corresponding y-values.

$$\int_{top}^{bottom} this\ dx = boring$$

Maybe next time will be better!