Example 89. Python numpy and scipy are powerful scientific libraries for Python. While numpy provides core functionality, scipy implements more specialized routines such as interpolation.

```
>>> from numpy import linspace, pi, sin
```

```
>>> from scipy import interpolate
```

Comment. We previously used the sin function available in the math Python standard library. The numpy library offers its own sin function with additional features. For instance, try sin([1,2]). This evaluates $\sin(x)$ at both $x=1$ and $x=2$. On the other hand, this results in an error with the sin function not from numpy.

Let us interpolate $f(x)=\sin(x)$ using 3 points, namely $x_0=0$, $x_1=\frac{\pi}{2}$, $x_2=\pi$. We begin by making lists of the x and y values as follows:

```
>>> xpoints = [0, pi/2, pi]
```

```
>>> ypoints = [sin(x) for x in xpoints]
```

Comment. As pointed out in the previous comment, we can even simply use ypoints = sin(xpoints). (The result would be a numpy array instead of a standard list but, for basic purposes, these behave alike. The numpy library introduces and uses arrays for additional features and performance for scientific computations.)

Let us check that xpoints and ypoints hold the expected values:

```
>>> xpoints

[0, 1.5707963267948966, 3.141592653589793]
```

```
>>> ypoints

[0.0, 1.0, 1.2246467991473532e-16]
```

We now ask scipy to create the interpolating polynomial:

```
>>> poly = interpolate.lagrange(xpoints, ypoints)
```

The resulting polynomial can be evaluated at any other point (such as $x=\pi/4$) and we can access its coefficients (which tell us that the polynomial is approximately $-0.41x^2+1.27x$):

```
>>> poly(pi/4)

0.75
```

```
>>> sin(pi/4)

0.7071067811865475
```

```
>>> poly.coeffs

[-0.40528473456935116, 1.2732395447351628, 0.0]
```

Homework. Show that the exact interpolation polynomial is $\frac{4}{\pi}x-\frac{4}{\pi^2}x^2$.

Finally, let us plot the sine function together with the polynomial interpolation. In the code below we use matplotlib, a powerful and widely used plotting library.

```
>>> import matplotlib.pyplot as plt
```

```
>>> xplot = linspace(0, pi, 100)
```

```
>>> plt.plot(xpoints, ypoints, 'o', xplot, sin(xplot), '-', xplot, poly(xplot), ':')
```

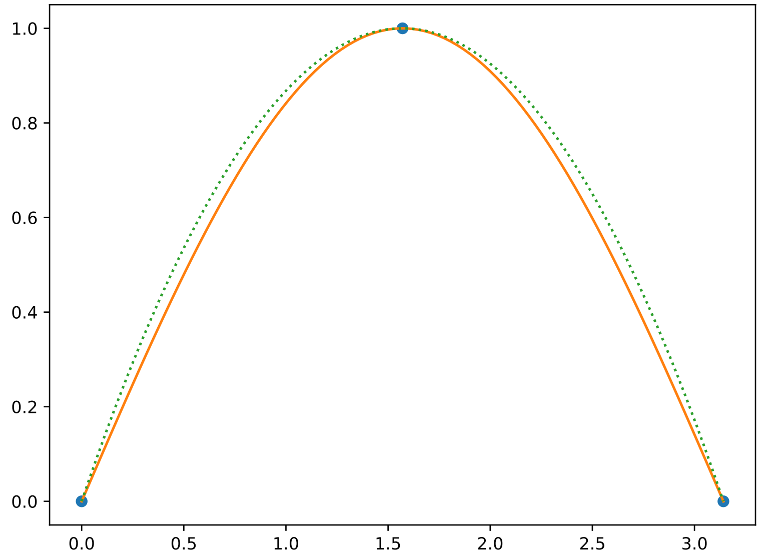
```
>>> plt.show()
```

Comment. Note that we are making three plots in one line here (namely, we plot the three points, we plot sine, and we plot the polynomial interpolation).

To plot just sine, simplify the `plot` command to `plt.plot(xplot, sin(xplot), '-')`. The `'-'` connects the 100 points (with x-coordinates from `xplot`) by a line. Replace it, for instance, with `'r-.'` to get a red dotted line.

https://matplotlib.org/stable/tutorials/introductory/pyplot.html

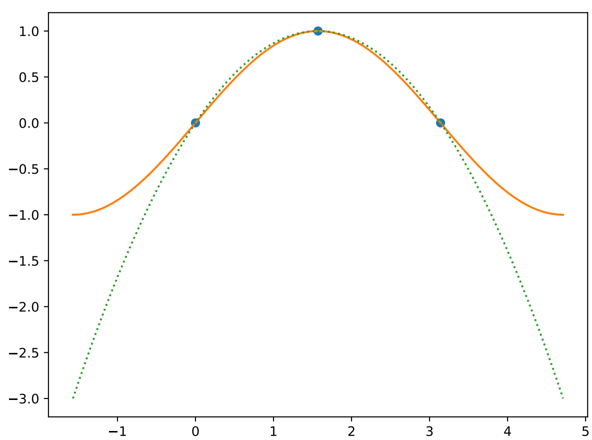
The resulting output should look as follows:

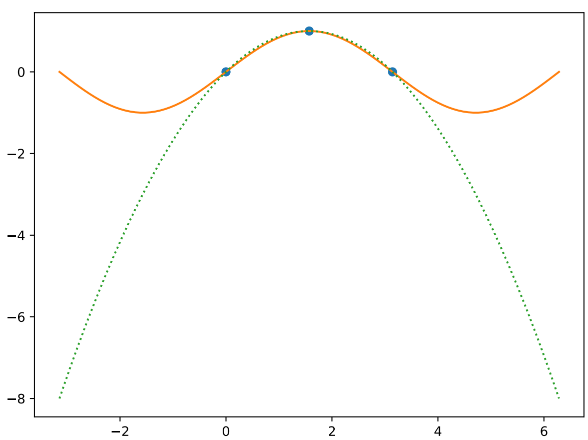


This shows pretty decent interpolation on the interval $[0, \pi]$.

Which function is which?! (You can tell from the fact that we dotted one graph or from the plots below.)

On the other hand, here are the same plots on $\left[-\frac{\pi}{2}, \frac{3\pi}{2}\right]$ and $[-\pi, 2\pi]$:





Homework. Adjust our code above (only the line `linspace(0, pi, 100)` needs to be changed) to produce these two plots.

As we can see (and as we probably expected), the polynomial interpolation does not approximate the sine function well outside the interval $[0, \pi]$.

Comment. Given the three interpolation points $0, \pi/2, \pi$, an attempt to approximate the function at values much less than 0 or much larger than π (that is, outside of the range of our data) is typically referred to as **extrapolation**.

Armin Straub
straub@southalabama.edu

Example 90. (review) Determine the minimal polynomial interpolating $(0,1),(1,2),(2,5)$.

Solution. (Lagrange, review) The interpolating polynomial in Lagrange form is:

$$\begin{aligned} p(x) &= 1\frac{(x-1)(x-2)}{(0-1)(0-2)} + 2\frac{(x-0)(x-2)}{(1-0)(1-2)} + 5\frac{(x-0)(x-1)}{(2-0)(2-1)} \\ &= \frac{1}{2}(x-1)(x-2) - 2x(x-2) + \frac{5}{2}x(x-1) \\ &= x^2+1 \end{aligned}$$

Solution. (Newton, divided differences)

$$\begin{array}{llll} 0: & 1 & & \\ & & \frac{2-1}{1-0}=1 & \\ 1: & 2 & & \frac{3-1}{2-0}=1 \\ & & \frac{5-2}{2-1}=3 & \\ 2: & 5 & & \end{array}$$

Accordingly, reading the coefficients from the top edge of the triangle:

$$p(x) = 1 + 1(x-0) + 1(x-0)(x-1) = x^2+1$$

A mean value theorem for divided differences

Review. The **mean value theorem** (see Theorem 55; the special case $M=0$ of Taylor's theorem) states that, if $f(x)$ is differentiable, then

$$f[a,b] = \frac{f(b)-f(a)}{b-a} = f'(\xi)$$

for some ξ between a and b.

Recall that the Newton form of the polynomial interpolating $f(x)$ at $x = x_0, x_1, \ldots$ is

$$f[x_0] + f[x_0,x_1](x-x_0) + f[x_0,x_1,x_2](x-x_0)(x-x_1) + f[x_0,x_1,x_2,x_3](x-x_0)(x-x_1)(x-x_2) + \ldots$$

Note that this is somewhat similar to the Taylor expansion of $f(x)$ at $x = x_0$, which is

$$f(x_0) + f'(x_0)(x-x_0) + \tfrac{1}{2!}f''(x_0)(x-x_0)^2 + \tfrac{1}{3!}f'''(x_0)(x-x_0)^3 + \ldots$$

Indeed, if all the x_j are equal to x_0 (this is technically not allowed when interpolating, but you can still think of choosing them all close to x_0), then the Newton form would turn into a Taylor polynomial.
In that case, $f[x_0, x_1, \ldots, x_n]$ would become $\frac{1}{n!}f^{(n)}(x_0)$.

With that (as well as the mean value theorem and Taylor's theorem (see Theorem 54)) in mind, the next result does not come as a surprise.

Theorem 91. (mean value theorem for divided differences) If $f(x)$ is differentiable, then

$$f[x_0, x_1, \ldots, x_n] = \frac{f^{(n)}(\xi)}{n!}$$

for some ξ between the smallest and the largest of the x_i.

Proof. Without loss of generality, we may assume that $x_0 < x_1 < \cdots < x_n$ (because divided differences do not depend on the ordering of the points x_i).

Let $P(x)$ be the interpolation polynomial for f at $x_0, x_1, \ldots, x_n$. Then $d(x) = f(x) - P(x)$ has $n+1$ zeros, namely $x_0, x_1, \ldots, x_n$. The mean value theorem implies that between any two zeros of a function, there must be a zero of its derivative (this is often referred to as Rolle's theorem). It therefore follows that $d'(x)$ has n zeros (between x_0 and x_n). Applying the same argument to $d'(x)$, we then find that $d''(x)$ has $n-1$ zeros. Continuing like this, $d^{(n)}(x)$ must have a zero ξ between x_0 and x_n. As such,

$$0 = d^{(n)}(\xi) = f^{(n)}(\xi) - P^{(n)}(\xi).$$

Recall that $P(x)$ is a polynomial of degree n or less, and that its Newton form is

$$P(x) = c_0 + c_1(x - x_0) + c_2(x - x_0)(x - x_1) + \ldots + c_n(x - x_0)(x - x_1)\cdots(x - x_{n-1}),$$

where $c_j = f[x_0, x_1, \ldots, x_j]$. Note that $P^{(n)}(x) = n! c_n = n! f[x_0, x_1, \ldots, x_n]$. We therefore conclude that

$$0 = d^{(n)}(\xi) = f^{(n)}(\xi) - P^{(n)}(\xi) = f^{(n)}(\xi) - n! f[x_0, x_1, \ldots, x_n],$$

which proves the claim. □

Comment. Note that this provides us with a way to numerically approximate an nth derivative $f^{(n)}(x)$. Namely, choose $n+1$ points $x_0, x_1, \ldots, x_n$ near x. Then $f^{(n)}(x) \approx \underbrace{n! f[x_0, x_1, \ldots, x_n]}_{= f^{(n)}(\xi)}$.

Bounding the interpolation error

Theorem 92. (interpolation error) Suppose that $f(x)$ is $n+1$ times continuously differentiable. Let $P_n(x)$ be the interpolating polynomial for $f(x)$ at $x_0, x_1, \ldots, x_n$. Then

$$f(x) - P_n(x) = \underbrace{\frac{f^{(n+1)}(\xi)}{(n+1)!}(x - x_0)(x - x_1)\cdots(x - x_n)}_{\text{interpolation error}}$$

for some ξ between the smallest and the largest of the x_i together with x.

Proof. Let $P_{n+1}(x)$ be the interpolating polynomial for $f(x)$ at $x_0, x_1, \ldots, x_n, x_{n+1}$. We know that

$$\begin{aligned} P_{n+1}(x) &= P_n(x) + f[x_0, x_1, \ldots, x_{n+1}](x - x_0)(x - x_1)\cdots(x - x_n) \\ &= P_n(x) + \frac{f^{(n+1)}(\xi)}{(n+1)!}(x - x_0)(x - x_1)\cdots(x - x_n) \end{aligned}$$

for some ξ between the smallest and the largest of the x_i together with x.

Given any fixed value t, choose $x_{n+1} = t$ in this formula (so that $P_{n+1}(t) = f(t)$) to conclude that

$$f(t) = P_{n+1}(t) = P_n(t) + \frac{f^{(n+1)}(\xi)}{(n+1)!}(t - x_0)(t - x_1)\cdots(t - x_n),$$

which (after subtracting $P_n(t)$ from both sides) is the claimed expression for the error term (with x replaced by t). □

Example 93. Suppose we approximate $f(x) = \sin(x)$ by the polynomial $P(x)$ interpolating it at $x = 0, \frac{\pi}{2}, \pi$. Without computing $P(x)$, give an upper bound for the error when $x = \frac{\pi}{4}$.

[Compare with Example 89 where we computed and plotted $P(x)$.]

Solution. By Theorem 92, the error is

$$\sin(x) - P(x) = \frac{f^{(3)}(\xi)}{3!}(x-0)\left(x - \frac{\pi}{2}\right)(x-\pi),$$

where ξ is between 0 and π (provided that x is in $[0,\pi]$). Note that $f^{(3)}(x) = -\cos(x)$ so that $|f^{(3)}(\xi)| \leqslant 1$. Hence, the error is bounded by

$$|\sin(x) - P(x)| \leqslant \frac{1}{6}\left|x\left(x - \frac{\pi}{2}\right)(x-\pi)\right|.$$

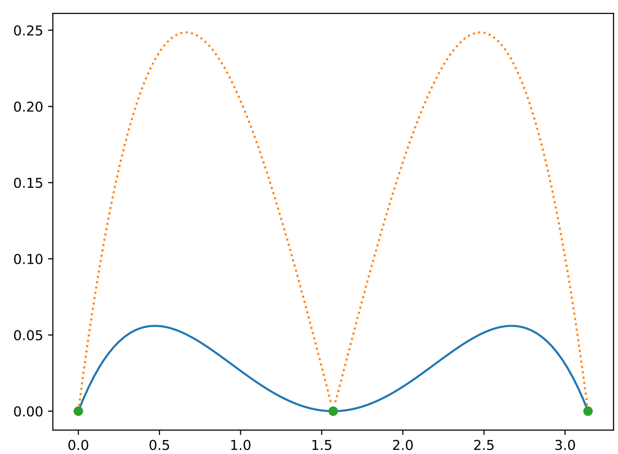
In particular, in the case $x = \frac{\pi}{4}$,

$$\left|\sin\left(\frac{\pi}{4}\right) - P\left(\frac{\pi}{4}\right)\right| \leqslant \frac{1}{6}\left|\frac{\pi}{4}\left(-\frac{\pi}{4}\right)\left(-\frac{3\pi}{4}\right)\right| = \frac{\pi^3}{128} \approx 0.242.$$

For comparison. In this particularly simple case, we can easily calculate the exact error.

Namely, since $\sin(\frac{\pi}{4}) = \frac{1}{\sqrt{2}}$ and $P(\frac{\pi}{4}) = \frac{3}{4}$ (see Example 89), the actual error is $\left|\sin(\frac{\pi}{4}) - P(\frac{\pi}{4})\right| \approx 0.0428$.

Below is a plot of the actual error (in blue) together with our bound (dotted).



Homework. Following what we did in Example 89, try to reproduce this plot.

For which x in $[0,\pi]$ is our bound for the error maximal? What is the bound in that case?

Solution. Recall that our bound for the error is $\frac{1}{6}\left|x(x - \frac{\pi}{2})(x-\pi)\right|$.

$x(x - \frac{\pi}{2})(x-\pi)$ is maximal on $[0,\pi]$ for $x = \left(1 \pm \frac{1}{\sqrt{3}}\right)\frac{\pi}{2} \approx 0.664, 2.478$. (Fill in the details!)

The corresponding error bound is $\frac{1}{72\sqrt{3}}\pi^3 \approx 0.249$.

Comment. Note that this shows that our earlier error bound for $x = \frac{\pi}{4} \approx 0.785$ was close to the worst case. That is not too much of a surprise since $\frac{\pi}{4}$ sits right between 0 and $\frac{\pi}{2}$ for which the error is 0 by construction.

For comparison. The actual maximal error occurs when $\cos(x) - \frac{4}{\pi} + \frac{8}{\pi^2}x = 0$. (Why?!)

The approximate solutions are $x \approx 0.472, 2.670$ with corresponding (actual) error of 0.0560.

Make sure that you can identify both the x values and the error in the above plot.

Armin Straub
straub@southalabama.edu

Example 94. (homework) Suppose we approximate a function $f(x)$ by the polynomial $P(x)$ interpolating it at $x=-1,-\frac{1}{3},\frac{1}{3},1$. Suppose that we know that $|f^{(n)}(x)|\leqslant n$ for all $x\in[-1,1]$.

(a) Give an upper bound for the error when $x=-\frac{2}{3}$.

(b) Give an upper bound for the error when $x=0$.

(c) Give an upper bound for the error for all $x\in[-1,1]$.

Solution. By Theorem 92, the error is

$$f(x)-P(x)=\frac{f^{(4)}(\xi)}{4!}(x+1)\left(x+\frac{1}{3}\right)\left(x-\frac{1}{3}\right)(x-1)=\frac{f^{(4)}(\xi)}{4!}(x^2-1)\left(x^2-\frac{1}{9}\right),$$

where ξ is between -1 and 1 (provided that $x\in[-1,1]$). Since $\frac{1}{4!}|f^{(4)}(\xi)|\leqslant\frac{4}{4!}=\frac{1}{6}$, the error is bounded by

$$|f(x)-P(x)|\leqslant\frac{1}{6}\left|(x^2-1)\left(x^2-\frac{1}{9}\right)\right|.$$

(a) If $x=-\frac{2}{3}$, then this bound becomes $|f(x)-P(x)|\leqslant\frac{1}{6}\left|(x^2-1)\left(x^2-\frac{1}{9}\right)\right|=\frac{1}{6}\cdot\frac{5}{27}\approx0.0309$.

(b) If $x=0$, then this bound becomes $|f(x)-P(x)|\leqslant\frac{1}{6}\left|(x^2-1)\left(x^2-\frac{1}{9}\right)\right|=\frac{1}{6}\cdot\frac{1}{9}\approx0.0185$.

Comment. It is not surprising that this error bound is better than the one for $x=-\frac{2}{3}$ since, roughly speaking, there are more interpolation nodes around 0.

(c) Consider $g(x)=(x^2-1)\left(x^2-\frac{1}{9}\right)=x^4-\frac{10}{9}x^2+\frac{1}{9}$. We need to compute $\max_{x\in[-1,1]}|g(x)|$.

Since $g(\pm1)=0$, the maximum value of $|g(x)|$ must be attained at a point where $g'(x)=0$.

We compute $g'(x)=4x^3-\frac{20}{9}x$. Hence $g'(x)=0$ if $x=0$ or $x=\pm\frac{\sqrt{5}}{3}$.

Since $|g(0)|=\frac{1}{9}$ and $\left|g\left(\pm\frac{\sqrt{5}}{3}\right)\right|=\frac{16}{81}>\frac{1}{9}$, we conclude that $\max_{x\in[-1,1]}|g(x)|=\frac{16}{81}$.

Thus, our error bound is $\max_{x\in[-1,1]}|f(x)-P(x)|\leqslant\frac{1}{6}\max_{x\in[-1,1]}\left|(x^2-1)\left(x^2-\frac{1}{9}\right)\right|=\frac{1}{6}\cdot\frac{16}{81}\approx0.0329$.

Example 95. Python We can approximate $\frac{1}{6}\max_{x\in[-1,1]}\left|(x^2-1)\left(x^2-\frac{1}{9}\right)\right|=\frac{1}{6}\cdot\frac{16}{81}\approx0.0329$ as follows using 100 points.

```
>>> from numpy import linspace
>>> max([1/6*abs((x**2-1)*(x**2-1/9)) for x in linspace(-1,1,100)])
    0.0328984640831
```