Symbolic Computation

A Polynomial:

$$p(t) = \sum_{i=0}^{n} a_i t^i$$

In Maple:

```
p := sum(a[i]*t^(i-1), i=1..n+1);
```

Maple knows about some relationships. For example,

$$\pi = \int_{-\infty}^{+\infty} \frac{\sin x}{x} dx.$$

In Maple:

```
int(sin(x)/x,x=-infinity..infinity);
```

Leibnitz' formula

$$\frac{\pi}{4} = 1 - \frac{1}{3} + \frac{1}{5} - \frac{1}{7} + \cdots ,$$

Euler's formula

$$\frac{\pi^2}{6} = \sum_{n=1}^{\infty} \frac{1}{n^2} .$$

Suppose we wish to study the convergence of Leibnitz' formula for π.

Let

$$Pn = 1 - \frac{1}{3} + \frac{1}{5} - \frac{1}{7} + \cdots + \frac{(-1)^n}{2n+1}$$

$$= \sum_{i=0}^{n} \frac{(-1)^i}{2i+1} .$$

A Maple procedure to compute this partial series is:

```
P := proc (n)
     local i;
     sum( (-1)^i / (2*i+1), i=0..n );
end:
```

```
> P(0);
                              1

> P(1);
                             2/3

> P(10);
                          11757173
                          --------
                          14549535
```

Notice that the assigned result is in a symbolic fractional form. To evaluate the result numerically, we type

```
> evalf(");
                         .8080789524
```

where the **"** refers to the previous result. A numeric approximation for $\pi/4$ is

```
> evalf(Pi/4);
                         .7853981635
```

which differs considerably from our approximation. However, observe that

```
> evalf(P(1000));
                         .7856479136
```

So, 1000 terms in the series only buys us three decimal places of accuracy.

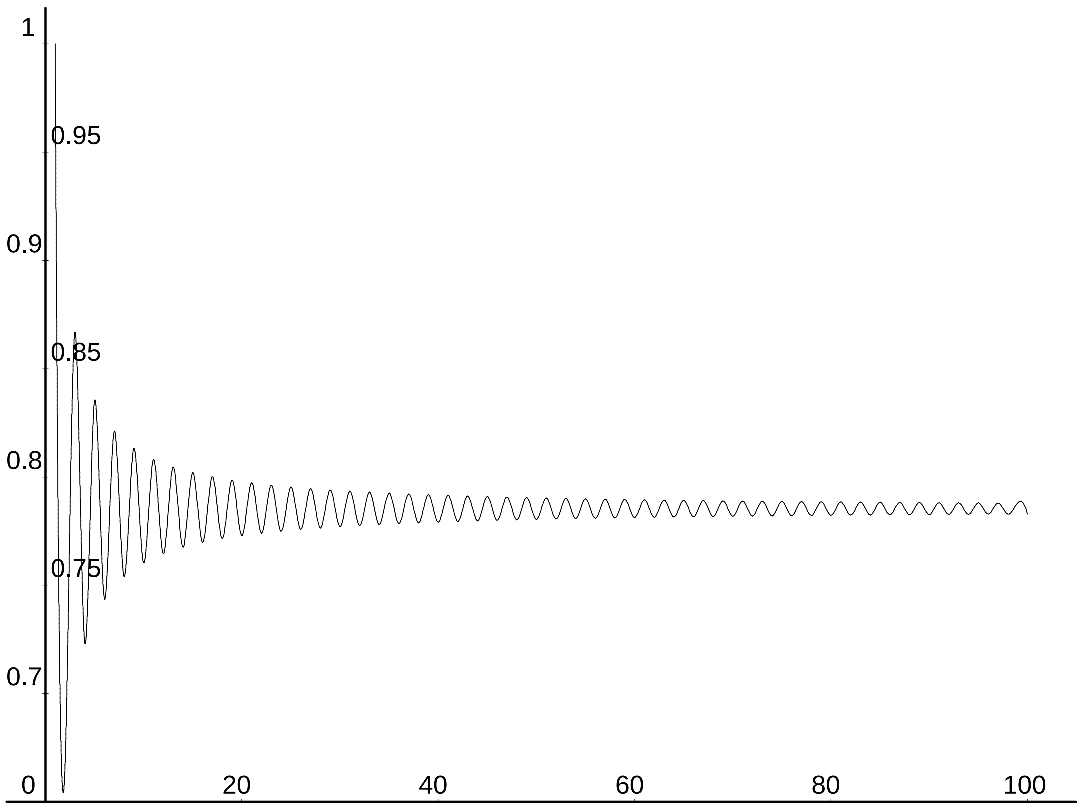
To see just how bad the convergence is, let us plot the behaviour of **`P(n)`** as **`n`** increases.

We first create a sequence of values $(0, Pb(0)),\ (1, Pb(1),\ \cdots,\ (n, Pn)$ and then plot the sequence as a connected curve, as follows.

```
> pts := [ [i, P(i)] $ i=0..100]:

> plot(pts);
```

We used a colon above to inform Maple that we do not wish the resulting value of **`pts`** to be displayed.



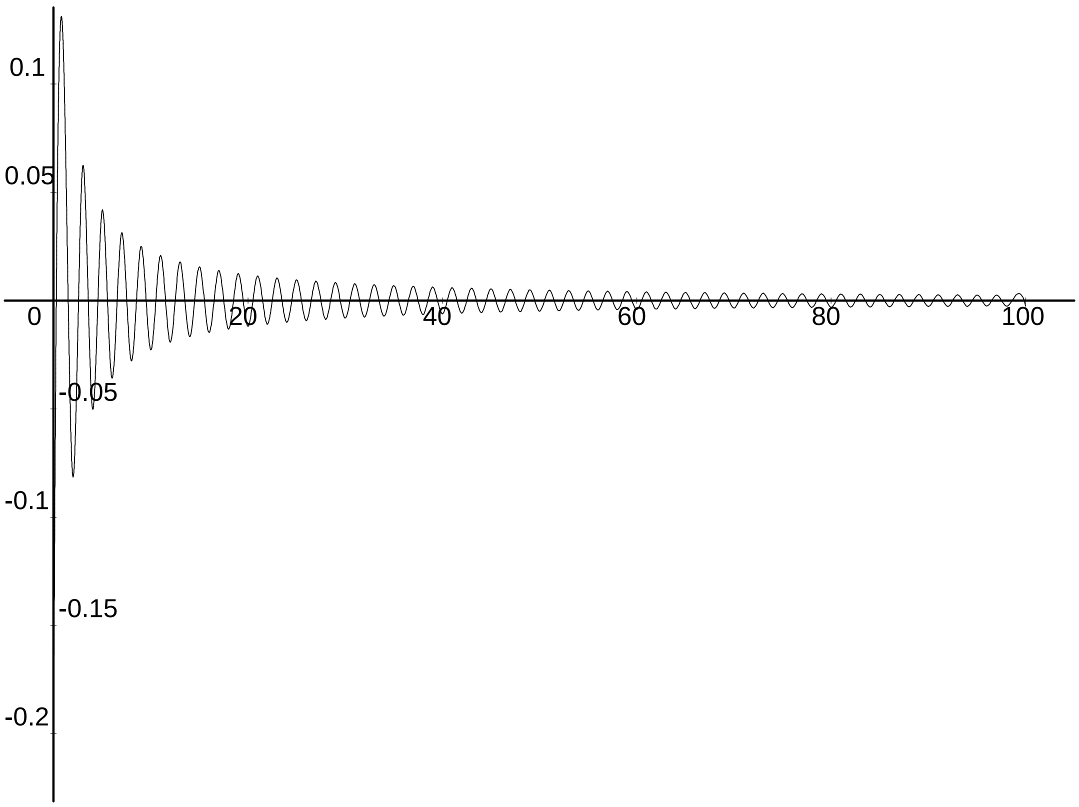
Residual Error

$$\varepsilon_n = \frac{\pi}{4} - Pn.$$

ε_n tells us the signed difference between the limit of P and Pn for any n.

In Maple, we could compute a sequence of such errors as follows:

```
> eps := [ [i, Pi/4 - P(i)] $ i=0..100]:
```

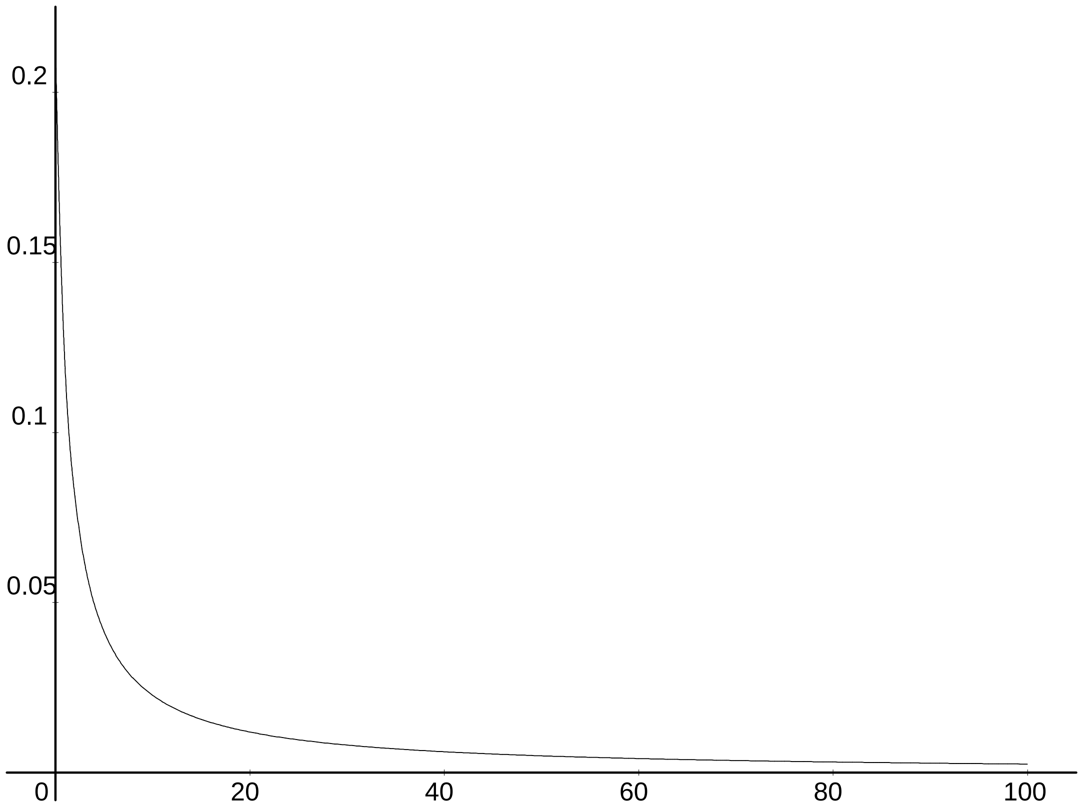


Absolute Error

$$\alpha_n = \left| \frac{\pi}{4} - Pn \right|.$$

In Maple, we can write this as

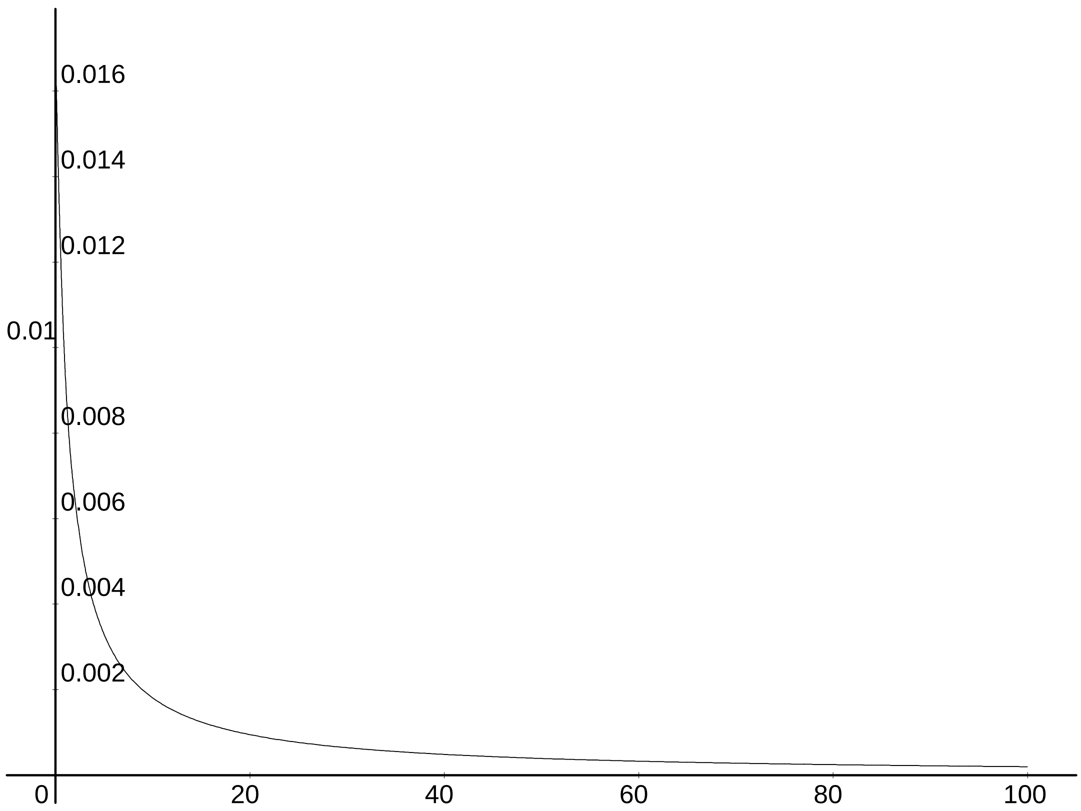
```
> alpha := [ [i, abs(Pi/4 - P(i))] $ i=0..100]:
```



Relative Error

Gives error in proportion to actual value:

$$\rho_n = \frac{|\pi/4 - P_n|}{\pi/4} = \frac{\alpha_n}{\pi/4}.$$



Let's Compute π

Consider a circle of diameter 1.

Its radius is ½.

As we shall see in more detail in the next chapter, such a circle can be conveniently defined *parametrically*, which means that we can describe the behaviour of the circle independently in its co-ordinates, say x and y.

In particular if we plot $(x(\theta), y(\theta))$, where

$$x(\theta) = \tfrac{1}{2}\cos\theta,$$

$$y(\theta) = \tfrac{1}{2}\sin\theta,$$

and $\theta \in [0, 2\pi]$, then a circle is the result.

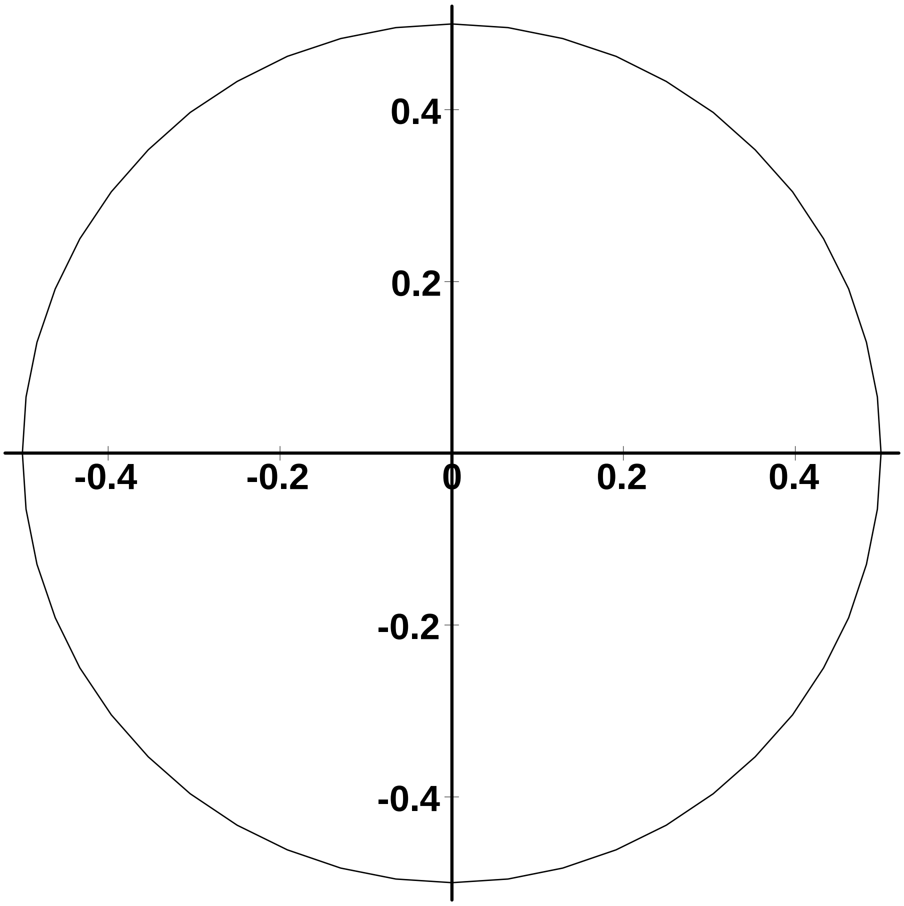
In Maple, we can (verbosely) write:

```
x := proc(t)
    1/2*cos(t);
end:

y := proc(t)
    1/2*sin(t);
end:

circle := proc()
    plot([x(t),y(t), t=0..2*Pi]);
end:
```

Upon invoking **`circle()`**, we get:



The *circumference* of (i.e., the distance around)
a circle of radius r=½ is $2\pi r$, or simply π.

Thus if we can approximate the circumference we have an approximation to π.

One way to do this is to compute the *perimeter* of a polygon whose edges lie close to the circle.

Let us *inscribe* the circle with a polygon, meaning that the polygon lies completely within the interior of the circle except at the polygon's vertices.

For simplicity, we shall assume the polygon is *regular*, meaning that its edges are all the same length.

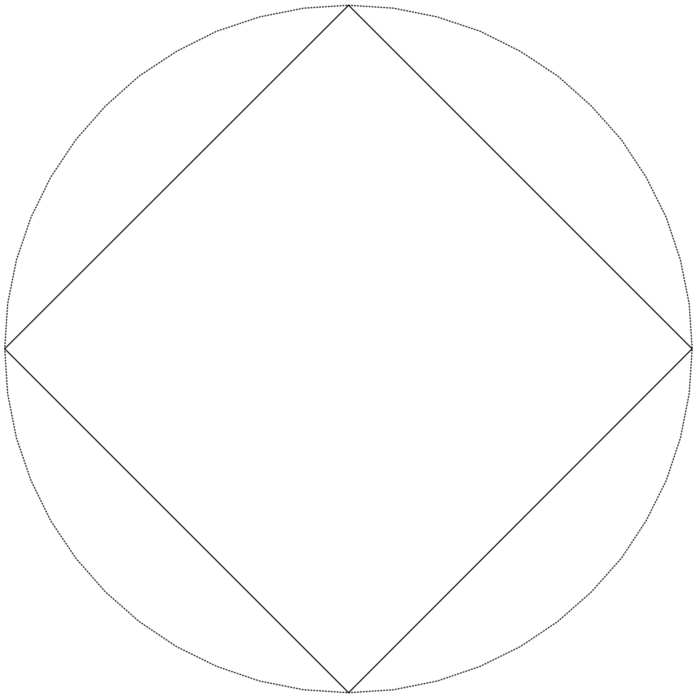
Thus if the polygon has n sides, then its perimeter is just n times the length of one side.
It is very easy to compute regular, inscribed polygons of the unit circle: if we are to have n edges, then we compute n equally paced points on the circle.

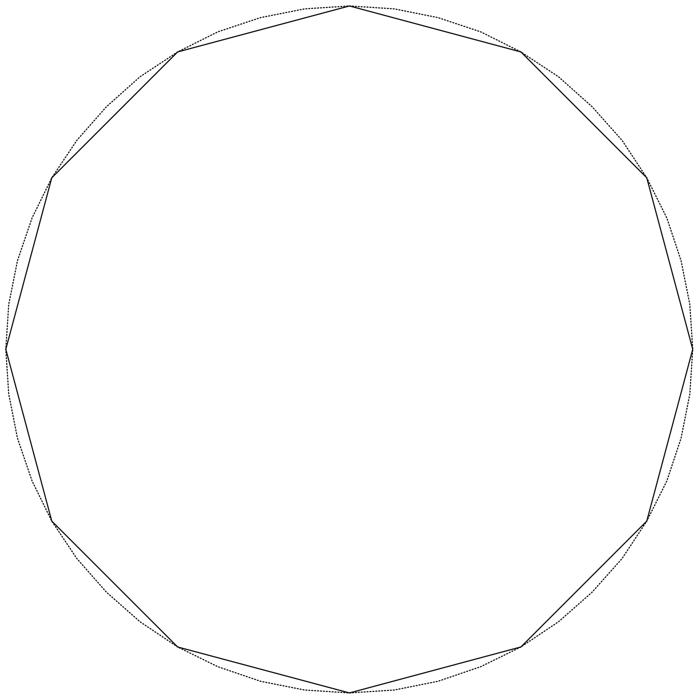
To do this we just compute $x(\theta)$ and $y(\theta)$ for each angle

$$\theta = i\,\frac{2\pi}{n}, \quad i=1, 2, \cdots, n\,.$$

The following code plots both the unit circle and an inscribing regular n-gon.

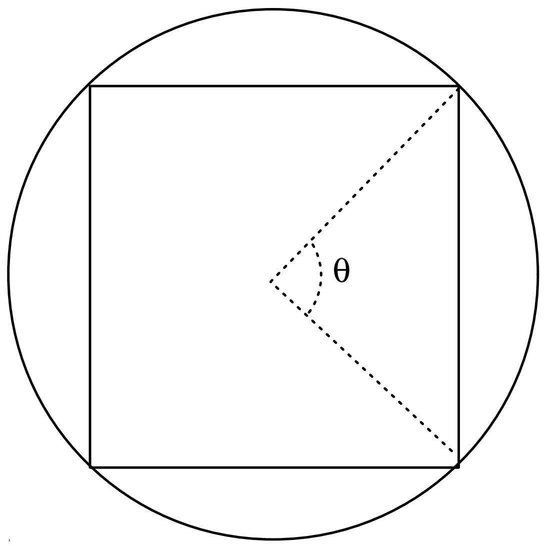
```
ngon := proc(n)
    local i,pts;
    pts := [[x(i*2*Pi/n),y(i*2*Pi/n)] $ i=0..n];
    plot({[x(t),y(t), t=0..2*Pi],pts},style=LINE);
end:
```





Let θ_n be the angle measured from the centre of the circle between two endpoints of any edge of a regular n-gon that inscribes the unit circle.

For example, for an inscribing square, $\theta_4 = \pi/2$.



In general, $\theta_n = 2\pi/n$, for a regular, inscribing polygon on n vertices.

Exercise:
Show that the length d_n of the edge of a regular n-gon is

$$d_n = \sqrt{(1-\cos\theta_n)/2}.$$

What is d_4?

What is the corresponding perimeter of the inscribing square?

From this exercise, the perimeter P_n is

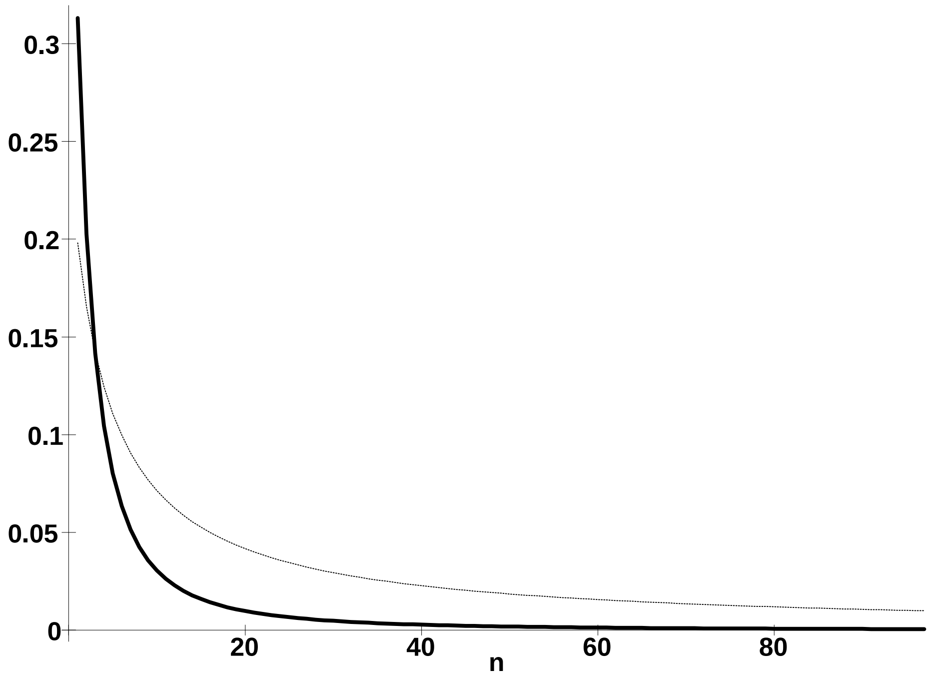
$$P_n = n\, d_n = n\sqrt{\left(1-\cos\frac{2\pi}{n}\right)/2} \quad .$$

%

We can implement this formula using Maple as follows.

```
Pn := proc (n)        # perimeter of regular n-gon
    local i;
    n * sqrt( 1/2*(1-cos(2*Pi/n)) );
end:
```

When we plot the absolute error of this new formula against Leibnitz' formula:



What's wrong with this approach?