In [ ]:

```
import sys
!{sys.executable} -m pip install pandas numpy matplotlib scipy pynverse
```

In [2]:

```
import scipy
import pandas
import pynverse
import scipy.stats
import numpy as np
import matplotlib.pyplot as plt
from scipy.optimize import fmin

π = np.pi
```

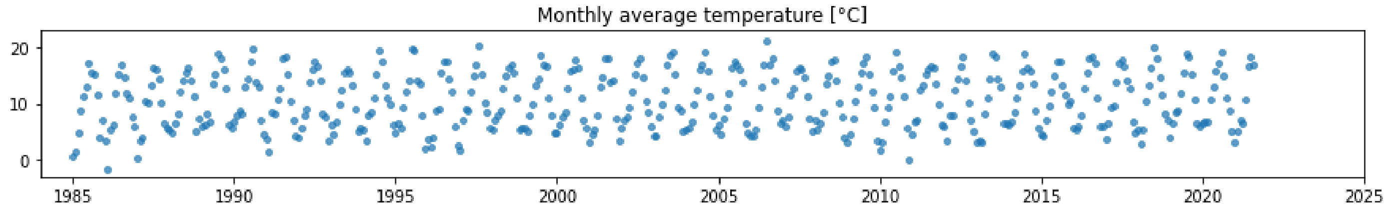
Developing a Better Temperature Model

An Exploration of how Sine Waves are Hot Garbage

In the first few Data Science lectures, we developed a model of the following climate data:

In [3]:

```
url = 'https://www.cl.cam.ac.uk/teaching/2122/DataSci/data/climate.csv'
climate = pandas.read_csv(url)

climate['t'] = climate.yyyy + (climate.mm-1)/12    # timestamp in years
climate['temp'] = (climate.tmin + climate.tmax)/2  # monthly avg temp

df = climate.loc[(climate.station=='Cambridge') & (climate.yyyy>=1985)]

fig,ax = plt.subplots(figsize=(15,1.7))
ax.scatter(df.t, df.temp, s=15, alpha=0.7)
ax.set_xlim([1984,2025])
ax.set_ylim([-3,23])
ax.set_title('Monthly average temperature [°C]')

plt.show()
```



We used the following model, which assumed that the data was a sine wave, with a gradual linear increase, plus some random normally distributed noise.

$$Temp_i = \alpha \sin\big(2\pi(t+\phi)\big) + c + \gamma t + \mathrm{Normal}(0, \sigma^2)$$

Our justification for using a sine wave was that the distance between the Earth and the Sun varies over the course of the year — closest in the Summer (meaning greater temperatures) and furthest in the Winter (meaning lower temperatures). Since this variation is smooth and periodic, we figured that a sine wave was probably good enough. This, however, is dumb and we should feel bad about ourselves.

This notebook explores how to use ellipse geometry and Kepler's laws of planetary motion to model the distance between the Earth and the Sun over time.

Setting up the Geometry

Kepler's 3 laws of planetary motion are as follows [Wikipedia]

1. The orbit of a planet is an ellipse with the Sun at one of the two foci.
2. A line segment joining a planet and the Sun sweeps out equal areas during equal intervals of time.
3. The square of a planet's orbital period is proportional to the cube of the length of the semi-major axis of its orbit.

We will therefore model the orbit of the Earth around the Sun as an axis-aligned ellipse (with the semi-major axis along the horizontal) centred on the origin with semi-major axis length a and eccentricity E. We can calculate the semi-minor axis length b and the distance between the centre of the ellipse and either of the focal points f as follows:

$$f = \frac{1}{2}aE$$
$$b = \sqrt{a^2 - f^2}$$

(Note that we are assuming that E is constant. However, in reality the eccentricity of the Earth's orbit follows a Milankovitch cycle, varying between around 0.000055 to around 0.0679. This is believed to be mostly due to the gravitational pull of Jupiter and Saturn. Nonetheless, taking E to be constant is vital if I want to maintain even a shred of sanity, so if this bothers you, you are cordially invited to deal with those emotions in whatever manner you see fit. [Wikipedia])

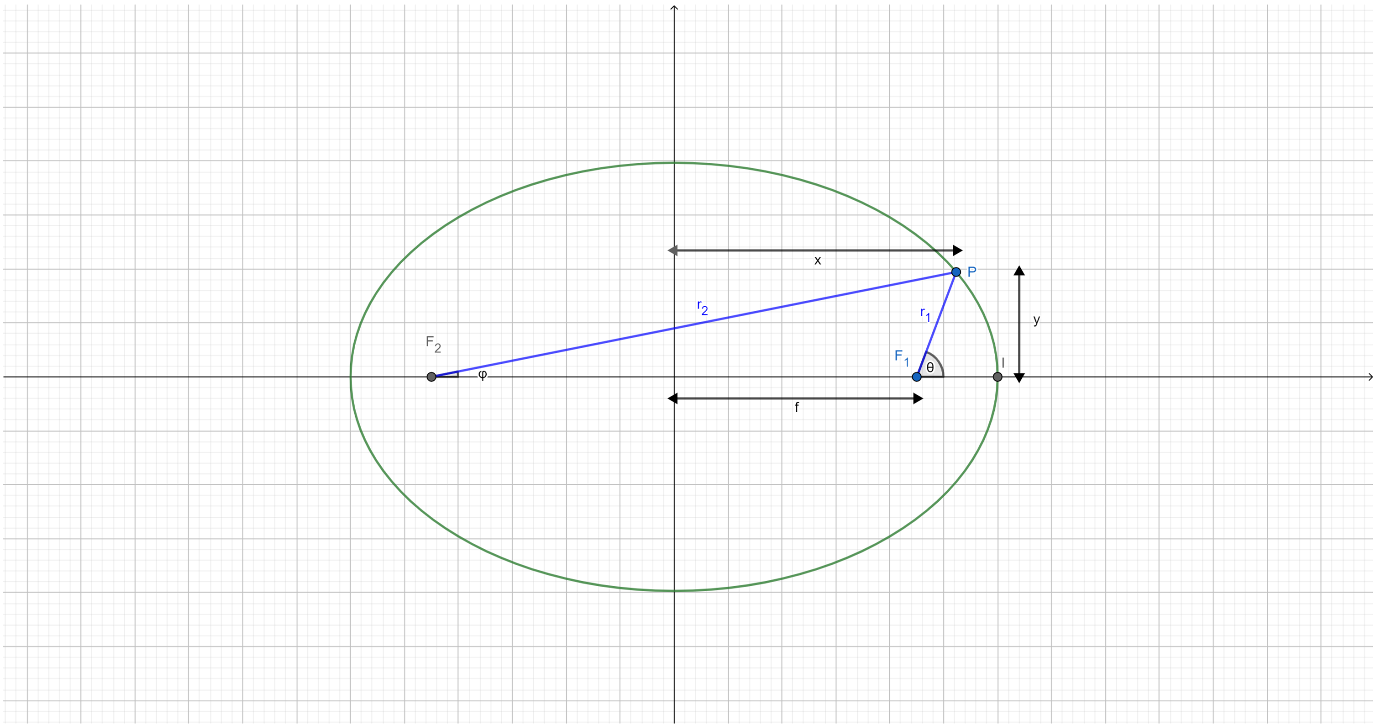
The two foci of the ellipse are points F_1 and F_2 where F_1 is the Sun. The Earth will be at point P.

If I is the point $(a, 0)$, the angles $\angle IF_1P = \theta$ and $\angle IF_2P = \phi$.

The distance between F_1 and P is r_1 and the distance between F_2 and P is r_2.

The area of the sector between I, F_1, and P will be called A.

This is all shown on the diagram below (although the label for A is ommitted because I am bad at geogebra and I didn't know how to include it):



We can call T the orbital period of the Earth, and t the ellapsed time. We can assume that $P|_{t=0} = I$.

A whole load of algebra

As a reminder, we are looking to find r_1 (the distance between the Earth and the Sun) as a function of t.

We can write the value y (the height of point P) in two different ways:

$$y = r_1 \sin(\theta)$$
$$y = r_2 \sin(\phi)$$

One property of ellipses is that $r1 + r2 = 2a$ for any point P on the ellipse. Therefore we can say

$$r_1 \sin(\theta) = (2a - r_1) \sin(\phi)$$

or equivalently,

$$\sin(\phi) = \frac{r_1 \sin(\theta)}{2a - r_1}$$

Squaring both sides,

$$\sin^2(\phi) = \frac{r_1^2 \sin^2(\theta)}{(2a - r_1)^2} \tag{1}$$

Similarly, we can write x in two ways:

$$x = r_1 \cos(\theta) + f$$
$$x = r_2 \cos(\phi) - f$$

Therefore,

$$\cos(\phi) = \frac{r_1 \cos(\theta) + 2f}{r_2} = \frac{r_1 \cos(\theta) + 2f}{2a - r_1}$$

Again squaring both sides,

$$\cos^2(\phi) = \frac{(r_1 \cos(\theta) + 2f)^2}{(2a - r_1)^2} \tag{2}$$

Adding (1) and (2) gives

$$\sin^2(\phi) + \cos^2(\phi) = 1 = \frac{r_1^2 \sin^2(\theta)}{(2a - r_1)^2} + \frac{(r_1 \cos(\theta) + 2f)^2}{(2a - r_1)^2}$$

Therefore,

$$(2a - r_1)^2 = r_1^2 \sin^2(\theta) + (r_1 \cos(\theta) + 2f)^2$$

Expanding both sides gives the following:

$$\begin{aligned} 4a^2 + r_1^2 - 4ar_1 &= r_1^2 \sin^2(\theta) + r_1^2 \cos^2(\theta) + 4f^2 + 4fr_1 \cos(\theta) \\ &= r_1^2 + 4f^2 + 4fr_1 \cos(\theta) \end{aligned}$$

Therefore,

$$a^2 - ar_1 = f^2 + fr_1 \cos(\theta)$$

Solving for r_1:

$$r_1 = \frac{a^2 - f^2}{a + f\cos(\theta)}$$

Recalling that $f^2 = a^2 - b^2$, we get

$$r_1 = \frac{b^2}{a + f\cos(\theta)} \tag{3}$$

We now have a formula for r_1 as a function of θ, so if we can find a formula for θ as a function of t, we can substitute this in to get r_1 as a function of t. To find θ as a function of t, we will find a formula for A in two different ways.

The first way is to integrate over θ, summing the areas of the arbitrarily small sectors. In the limit, these sectors are isocoles triangles. Two of its side lengths are $r_1(\theta)$ and the angle between them is $d\theta$. Their areas dA are therefore $\frac{1}{2}(r_1(\theta))^2 \sin(d\theta)$ and as $d\theta$ is arbitrarily small, $dA = \frac{1}{2}(r_1(\theta))^2 d\theta$

$$
\begin{aligned}
A &= \int dA \\
&= \int \frac{1}{2}(r_1(\theta))^2 d\theta \\
&= \int \frac{1}{2}\left(\frac{b^2}{a+f\cos(\theta)}\right)^2 d\theta \\
&= \int \frac{b^4}{2(a+f\cos(\theta))^2} d\theta
\end{aligned}
$$

This integral makes me want to cry, so I would strongly recommend taking my word on the solution. However, since the entire point of this notebook is unneccesary pedantry, the working-out will be in the appendix.

$$
A = \pi ab \left\lfloor \frac{\theta+\pi}{2\pi} \right\rfloor + b^4 \left(\frac{f\sin(\theta)}{2(f-a)(f+a)(f\cos(\theta)+a)} + \frac{a\sqrt{a-f}\arctan\left(\sqrt{\frac{a-f}{f+a}}\tan\frac{\theta}{2}\right)}{(f-a)^2(f+a)^{\frac{3}{2}}} \right) + C
$$

The boundary condition is that $A|_{\theta=0} = 0$. Substituting $\theta = 0$ into the above yields $A = C$. Therefore $C = 0$. Our formula for A as a function of θ is now the following:

$$
A = \pi ab \left\lfloor \frac{\theta+\pi}{2\pi} \right\rfloor + b^4 \left(\frac{f\sin(\theta)}{2(f-a)(f+a)(f\cos(\theta)+a)} + \frac{a\sqrt{a-f}\arctan\left(\sqrt{\frac{a-f}{f+a}}\tan\frac{\theta}{2}\right)}{(f-a)^2(f+a)^{\frac{3}{2}}} \right) \tag{4}
$$

We can also find A as a function of t by using Kepler's 2[nd] law, which implies that A increases linearly with t.

$$
A = kt + D
$$

Using boundary conditions:

$$
\begin{aligned}
A|_{t=0} = 0 &\Rightarrow D = 0 \\
A|_{t=T} = \pi ab &\Rightarrow k = \frac{\pi ab}{T}
\end{aligned}
$$

Therefore, our formula for A as a function of t is:

$$
A = \pi ab \frac{t}{T} \tag{5}
$$

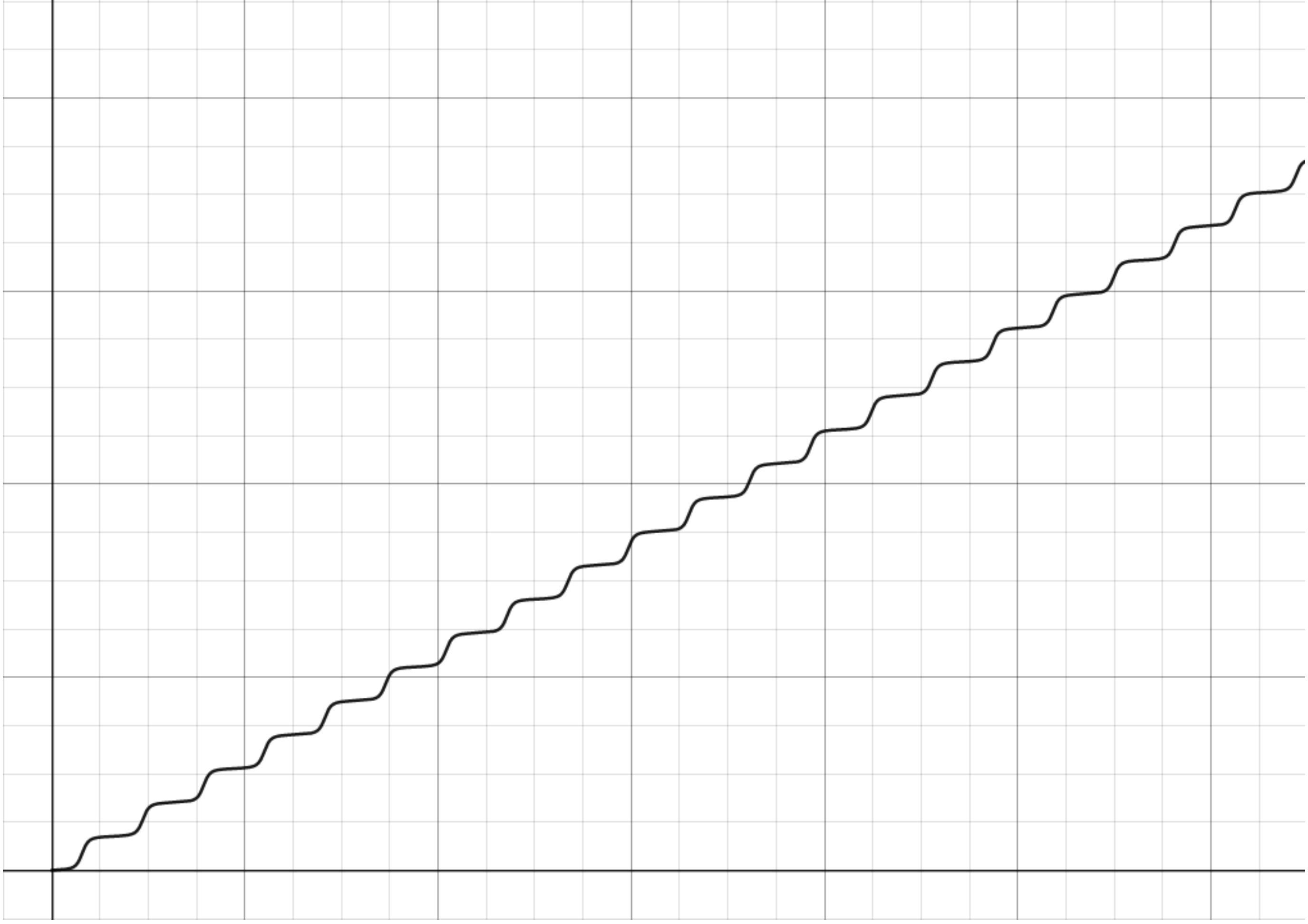
Equating (4) and (5) gives:

$$
\pi ab \frac{t}{T} = \pi ab \left\lfloor \frac{\theta+\pi}{2\pi} \right\rfloor + b^4 \left(\frac{f\sin(\theta)}{2(f-a)(f+a)(f\cos(\theta)+a)} + \frac{a\sqrt{a-f}\arctan\left(\sqrt{\frac{a-f}{f+a}}\tan\frac{\theta}{2}\right)}{(f-a)^2(f+a)^{\frac{3}{2}}} \right)
$$

which we can rearrange to solve for t.

$$
t = T \left\lfloor \frac{\theta+\pi}{2\pi} \right\rfloor + \frac{Tb^3}{\pi a} \left(\frac{f\sin(\theta)}{2(f-a)(f+a)(f\cos(\theta)+a)} + \frac{a\sqrt{a-f}\arctan\left(\sqrt{\frac{a-f}{f+a}}\tan\frac{\theta}{2}\right)}{(f-a)^2(f+a)^{\frac{3}{2}}} \right) \tag{6}
$$

This gives us t as a function of θ but what we really need is θ as a function of t. As you might imagine, this function is very difficult to invert. However, inspection of the graph of this function (which looks like it is strictly non-decreasing) would lead us to believe that a well-defined inverse does indeed exist.



Due to my desire to maintain my will to live, we will write code to find the inverse function using numerical methods, rather than solving algebraically. In fact, we will write code to find the inverse of any $\mathbb{R} \to \mathbb{R}$ function, if one exists.

In [4]:

```
def inverse(func, initial_guess=1.0):

    # func is such that f(a) = b

    def sqr_diff(a_prime, b):
        Δb = func(a_prime) - b
        return Δb*Δb

    # estimate(b) should approximate a
    def estimate(b):
        a_prime = np.zeros(b.shape)
        for i, true_b in enumerate(b):
            result = fmin(sqr_diff, initial_guess, args=(true_b,), disp=False)
            a_prime[i] = result[0]
        return a_prime

    return estimate
```

The above code works by taking a function `func`. We can assume that for some input `a`, `func(a)` should return an output `b`. Given some `b`, we want to find a value `a_prime` which is as close as possible to `a`. If `func` is strictly non-decreasing, then this is equivalent to finding an `a_prime` for which `func(a_prime)` is as close as possible to `b`.

We create a function `sqr_diff(a_prime, b)` which calculates the square difference between `func(a_prime)` and `b`. We therefore want to find the `a_prime` which minimises this function.

This is achieved by the `scipy.optimize.fmin` function. However, we have to call this function for each value of `b`. To my knowledge, there is no vectorised version of this function which can be used in this context. This means that calling `estimate` on a large array is *incredibly* slow.

Luckily, there's a package called `pynverse` which does it a lot quicker!

In [5]:

```
def inverse(func):
    def apply(b):
```

```
            return pynverse.inversefunc(func, y_values=b)
    return apply
```

We can now calculate θ as a function of t as follows:

In [6]:

```
def t_from_θ(θ, a, E, T):
    f=E*a*0.5
    b=np.sqrt(a*a-f*f)

    term1 = np.floor((θ+π)/(2.*π)) * T
    term2 = T*b*b*b/(π*a)
    term3 = (f*np.sin(θ))/(2.*(f-a)*(f+a)*(f*np.cos(θ)+a))
    term4 = a*np.sqrt(a-f)*np.arctan(np.sqrt((a-f)/(f+a))*np.tan(θ/2.))
    term5 = (f-a)*(f-a)*(f+a)*np.sqrt(f+a)

    return term1 + term2 * (term3 + term4/term5)

def θ_from_t(t, a, E, T):
    func = lambda θ: t_from_θ(θ, a, E, T)    #Turn t_from_ϑ into a single-input function
    inverted = inverse(func)
    return inverted(t)
```

We can now implement the formula for r_1 as a function of θ, and use it to calculate r_1 as a function of t.

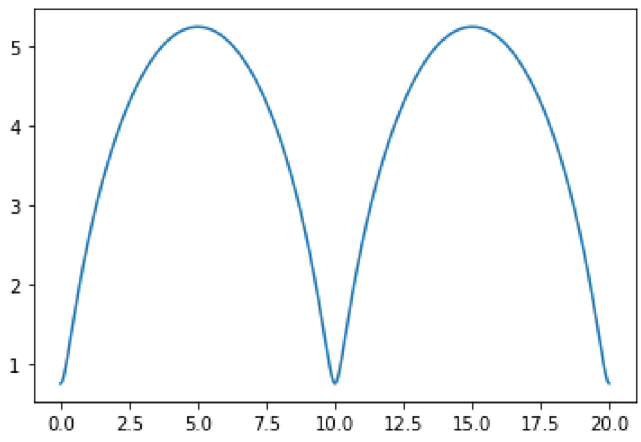
In [7]:

```
def r1_from_θ(θ, a, E, T):
    f=E*a*0.5
    b=np.sqrt(a*a-f*f)

    return b*b/(a + f*np.cos(θ))

def r1_from_t(t, a, E, T):
    θ = θ_from_t(t, a, E, T)
    return r1_from_θ(θ, a, E, T)
```

If we take a look at this model for arbitrary parameters $a = 3, E = 1.5, T = 10$, we see the following graph:

In [8]:

```
a = 3
E = 1.5
T = 10

t = np.arange(0,2*T,0.01)

r1 = r1_from_t(t, a, E, T)

plt.plot(t, r1)
plt.show()
```

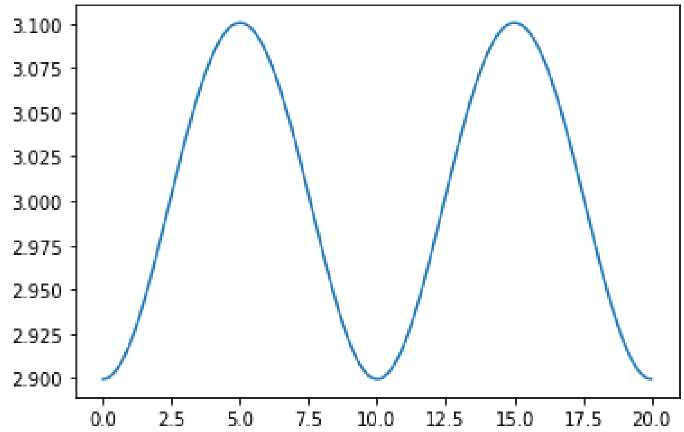


Validation! This does not look very sinuoidal at all. Although, the Earth's eccentricity is much smaller, only ever getting as big as around 0.067 [Wikipedia], so let's see how it looks with that eccentricity.

In [9]:

```
a = 3
E = 0.067
T = 10

t = np.arange(0,2*T,0.01)
```

```
r1 = r1_from_t(t, a, E, T)

plt.plot(t, r1)
plt.show()
```

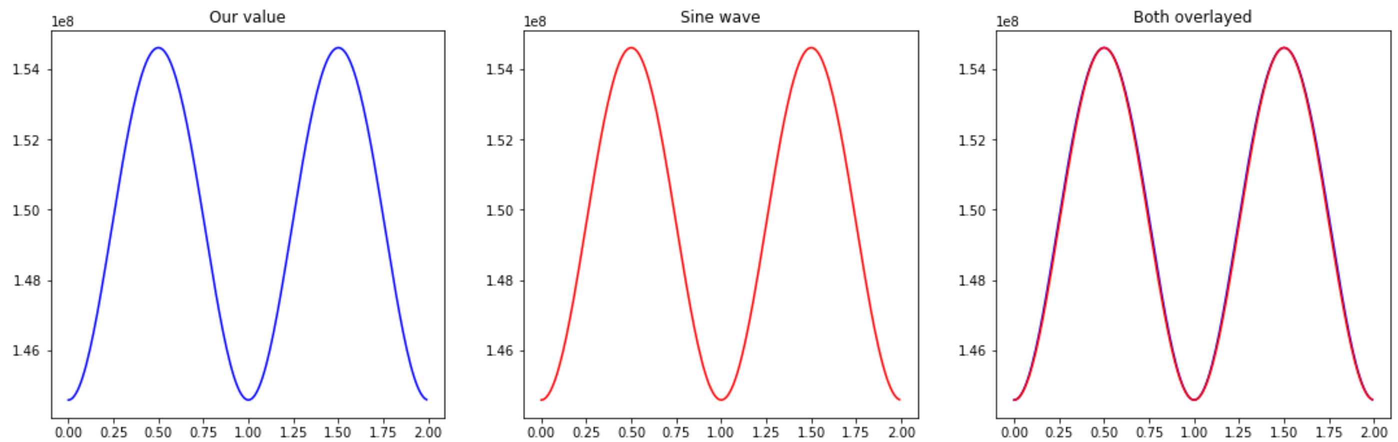


Oof okay granted, that's pretty sinusoidal. But it's not *exactly* equal to a sine wave, and that's something to celebrate!

Here it is plotted with a sine wave for comparison.

In [10]:

```
a = 149.60e6
E = 0.067
T = 1

t = np.arange(0,2*T,0.01)

r1 = r1_from_t(t, a, E, T)

sinusoid = np.interp(-np.cos(t*2*π/T), (-1, 1), (r1.min(), r1.max()))

fig, (ax1, ax2, ax3) = plt.subplots(1, 3)
fig.set_size_inches(18.5, 5.5)

ax1.set_title("Our value")
ax1.plot(t, r1, color="blue")

ax2.set_title("Sine wave")
ax2.plot(t, sinusoid, color="red")

ax3.set_title("Both overlayed")
ax3.plot(t, r1, color="blue")
ax3.plot(t, sinusoid, color="red")
plt.show()
```



It's pretty close, but not the same. I'm considering that a win.

Modelling temperature

The next step is to fit this model to the data. Replacing the sine term from the old model with our new curve we get:

$$Temp_i = \alpha r_1(t + \phi, a, E, T) + c + \gamma t + \text{Normal}(0, \sigma^2)$$

(Note that from now on, ϕ has gone back to meaning phase shift and not an angle)

It should be clear from the above graphs that this model will be a **negligible** improvement over the previous one. Furthermore, this improvement comes at the cost of a **significant** amount of extra computation time. However, we can use this new model to estimate the semi-major axis length, eccentricity, and period of Earth's orbit! Exciting, I know.

In [11]:
```
def rtemp(t, α, ϕ, a, E, T, c, γ, σ):
    pred = α * r1_from_t(t + ϕ, a, E, T) + c + γ * t
    return np.random.normal(loc=pred, scale=σ)
```

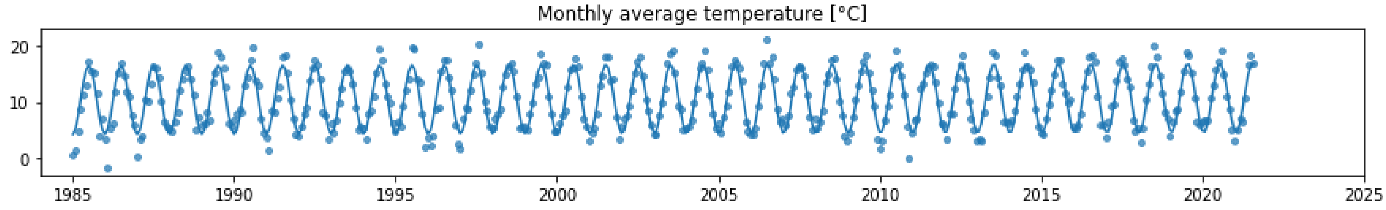
We will maximise over

$$\Theta = (\alpha, \phi, a, E, T, c, \gamma, \sigma) \in \mathbb{R}^8$$

The fitting code below can take a few minutes to run. Also, since the loss function is so complicated, there are a lot of false minima. Therefore, we need to pick very good initial estimates of a, E, and T for the fitting to be effective. Luckily, we can get these values from Wikipedia.

In [13]:
```
def log_lik(t, temp, Θ):
    α, ϕ, a, E, T, c, γ, σ = Θ
    est = rtemp(t, α, ϕ, a, E, T, c, γ, 0)
    lik = scipy.stats.norm.pdf(temp, loc=est, scale=σ)
    lik[np.where(lik == 0)] = 1e-10
    return np.log(lik)

initial_guess = [1e-5, 0., 150e6, 0.0167, 1, -1.485e3, 0., 2.]
Θ_hat = fmin(lambda Θ: -np.sum(log_lik(df.t, df.temp, Θ)), initial_guess, maxiter=5000, ftol=1e-6)

print("")
print(f"Predicted semi-major axis length: {Θ_hat[2]}km (wikipedia value: {149.60e6}km)")
print(f"Predicted eccentricity: {Θ_hat[3]} (wikipedia value: {0.0167086})")
print(f"Predicted period: {Θ_hat[4]} calendar years (google value: {1.0007} calendar years)")

pred = rtemp(df.t, *Θ_hat[:-1], 0)

fig,ax = plt.subplots(figsize=(15,1.7))
ax.scatter(df.t, df.temp, s=15, alpha=0.7)
ax.plot(df.t, pred)
ax.set_xlim([1984,2025])
ax.set_ylim([-3,23])
ax.set_title('Monthly average temperature [°C]')

plt.show()
```

```
Optimization terminated successfully.
         Current function value: 927.777139
         Iterations: 2064
         Function evaluations: 3371

Predicted semi-major axis length: 109937239.663032km (wikipedia value: 149600000.0km)
Predicted eccentricity: 0.016258113424275897 (wikipedia value: 0.0167086)
Predicted period: 1.0015105791935341 calendar years (google value: 1.0007 calendar years)
```



Interestingly enough, out approximations for the semi-major axis length and eccentricity from this model are slightly further from the values from Wikipedia than the initial guesses was. As far as I can tell that's because we are better than NASA and our numbers are more accurate.

Appendix

Solution to the integral

We want to solve the following integral:

$$\begin{aligned} A &= \int \frac{b^4}{2(a + f\cos(x))^2} dx \\ &= \frac{b^4}{2} \int \frac{1}{(a + f\cos(x))^2} dx \end{aligned}$$

Note that we've substituted $x = \theta$ so that I can put it into the online solver

Problem:

$$\int \frac{1}{(f\cos(x) + a)^2} \mathrm{d}x$$

Prepare for tangent half-angle substitution (Weierstrass substitution):

$$= \int \frac{1}{\left(\frac{f\left(1-\tan^2\left(\frac{x}{2}\right)\right)}{\tan^2\left(\frac{x}{2}\right)+1} + a\right)^2} \mathrm{d}x$$

Substitute $u = \tan\left(\frac{x}{2}\right) \longrightarrow \frac{\mathrm{d}u}{\mathrm{d}x} = \frac{\sec^2\left(\frac{x}{2}\right)}{2}$ (steps) $\longrightarrow \mathrm{d}x = \frac{2}{\sec^2\left(\frac{x}{2}\right)}\mathrm{d}u = \frac{2}{u^2+1}\mathrm{d}u$:

$$= 2\int \frac{u^2+1}{\left((a-f)\,u^2 + f + a\right)^2} \mathrm{d}u$$

Now solving:

$$\int \frac{u^2+1}{\left((a-f)\,u^2 + f + a\right)^2} \mathrm{d}u$$

Write $u^2 + 1$ as $\frac{1}{a-f}\left((a-f)\,u^2 + f + a\right) + 1 - \frac{f+a}{a-f}$ and split:

$$= \int \left(\frac{\frac{1}{a-f}\left((a-f)\,u^2 + f + a\right)}{\left((a-f)\,u^2 + f + a\right)^2} + \frac{1 - \frac{f+a}{a-f}}{\left((a-f)\,u^2 + f + a\right)^2}\right) \mathrm{d}u$$

$$= \int \left(\frac{1}{(a-f)\left((a-f)\,u^2 + f + a\right)} + \frac{1 - \frac{f+a}{a-f}}{\left((a-f)\,u^2 + f + a\right)^2}\right) \mathrm{d}u$$

Apply linearity:

$$= \frac{1}{a-f}\int \frac{1}{(a-f)\,u^2 + f + a} \mathrm{d}u + \left(1 - \frac{f+a}{a-f}\right)\int \frac{1}{\left((a-f)\,u^2 + f + a\right)^2} \mathrm{d}u$$

Now solving:

$$\int \frac{1}{(a-f)\,u^2 + f + a} \mathrm{d}u$$

Substitute $v = \frac{\sqrt{a-f}\,u}{\sqrt{f+a}} \longrightarrow \frac{\mathrm{d}v}{\mathrm{d}u} = \frac{\sqrt{a-f}}{\sqrt{f+a}}$ (steps) $\longrightarrow \mathrm{d}u = \frac{\sqrt{f+a}}{\sqrt{a-f}}\mathrm{d}v$:

$$= \int \frac{\sqrt{f+a}}{\sqrt{a-f}\left((f+a)\,v^2 + f + a\right)} \mathrm{d}v$$

Simplify:

$$= \frac{1}{\sqrt{a-f}\sqrt{f+a}}\int \frac{1}{v^2+1} \mathrm{d}v$$

Now solving:

$$\int \frac{1}{v^2+1} \mathrm{d}v$$

This is a standard integral:

$$= \arctan(v)$$

Plug in solved integrals:

$$\frac{1}{\sqrt{a-f}\sqrt{f+a}}\int \frac{1}{v^2+1} \mathrm{d}v$$

$$= \frac{\arctan(v)}{\sqrt{a-f}\sqrt{f+a}}$$

Undo substitution $v = \frac{\sqrt{a-f}\,u}{\sqrt{f+a}}$:

$$= \frac{\arctan\left(\frac{\sqrt{a-f}\,u}{\sqrt{f+a}}\right)}{\sqrt{a-f}\sqrt{f+a}}$$

Now solving:

$$\int \frac{1}{\left((a-f)\,u^2 + f + a\right)^2} \mathrm{d}u$$

Apply reduction formula:

$$\int \frac{\quad}{(\mathrm{a}u^2+\mathrm{b})^{\mathrm{n}}}\mathrm{d}u = \frac{\quad}{2\mathrm{b}(\mathrm{n}-1)}\int\frac{\quad}{(\mathrm{a}u^2+\mathrm{b})^{\mathrm{n}-1}}\mathrm{d}u + \frac{\quad}{2\mathrm{b}(\mathrm{n}-1)(\mathrm{a}u^2+\mathrm{b})^{\mathrm{n}-1}}$$

with $\mathrm{a} = a - f, \mathrm{b} = f + a, \mathrm{n} = 2$:

$$= \frac{u}{2(f+a)((a-f)u^2+f+a)} + \frac{1}{2(f+a)}\int\frac{1}{(a-f)u^2+f+a}\mathrm{d}u$$

Now solving:

$$\int\frac{1}{(a-f)u^2+f+a}\mathrm{d}u$$

Use previous result:

$$= \frac{\arctan\left(\frac{\sqrt{a-f}u}{\sqrt{f+a}}\right)}{\sqrt{a-f}\sqrt{f+a}}$$

Plug in solved integrals:

$$\frac{u}{2(f+a)((a-f)u^2+f+a)} + \frac{1}{2(f+a)}\int\frac{1}{(a-f)u^2+f+a}\mathrm{d}u$$

$$= \frac{\arctan\left(\frac{\sqrt{a-f}u}{\sqrt{f+a}}\right)}{2\sqrt{a-f}(f+a)^{\frac{3}{2}}} + \frac{u}{2(f+a)((a-f)u^2+f+a)}$$

Plug in solved integrals:

$$\frac{1}{a-f}\int\frac{1}{(a-f)u^2+f+a}\mathrm{d}u + \left(1-\frac{f+a}{a-f}\right)\int\frac{1}{((a-f)u^2+f+a)^2}\mathrm{d}u$$

$$= \frac{\arctan\left(\frac{\sqrt{a-f}u}{\sqrt{f+a}}\right)}{(a-f)^{\frac{3}{2}}\sqrt{f+a}} + \frac{\left(1-\frac{f+a}{a-f}\right)\arctan\left(\frac{\sqrt{a-f}u}{\sqrt{f+a}}\right)}{2\sqrt{a-f}(f+a)^{\frac{3}{2}}} + \frac{\left(1-\frac{f+a}{a-f}\right)u}{2(f+a)((a-f)u^2+f+a)}$$

Plug in solved integrals:

$$2\int\frac{u^2+1}{((a-f)u^2+f+a)^2}\mathrm{d}u$$

$$= \frac{2\arctan\left(\frac{\sqrt{a-f}u}{\sqrt{f+a}}\right)}{(a-f)^{\frac{3}{2}}\sqrt{f+a}} + \frac{\left(1-\frac{f+a}{a-f}\right)\arctan\left(\frac{\sqrt{a-f}u}{\sqrt{f+a}}\right)}{\sqrt{a-f}(f+a)^{\frac{3}{2}}} + \frac{\left(1-\frac{f+a}{a-f}\right)u}{(f+a)((a-f)u^2+f+a)}$$

Undo substitution $u = \tan\left(\frac{x}{2}\right)$:

$$= \frac{\left(1-\frac{f+a}{a-f}\right)\tan\left(\frac{x}{2}\right)}{(f+a)\left((a-f)\tan^2\left(\frac{x}{2}\right)+f+a\right)} + \frac{2\arctan\left(\frac{\sqrt{a-f}\tan\left(\frac{x}{2}\right)}{\sqrt{f+a}}\right)}{(a-f)^{\frac{3}{2}}\sqrt{f+a}} + \frac{\left(1-\frac{f+a}{a-f}\right)\arctan\left(\frac{\sqrt{a-f}\tan\left(\frac{x}{2}\right)}{\sqrt{f+a}}\right)}{\sqrt{a-f}(f+a)^{\frac{3}{2}}}$$

The problem is solved:

$$\int\frac{1}{(f\cos(x)+a)^2}\mathrm{d}x$$

$$= \frac{\left(1-\frac{f+a}{a-f}\right)\tan\left(\frac{x}{2}\right)}{(f+a)\left((a-f)\tan^2\left(\frac{x}{2}\right)+f+a\right)} + \frac{2\arctan\left(\frac{\sqrt{a-f}\tan\left(\frac{x}{2}\right)}{\sqrt{f+a}}\right)}{(a-f)^{\frac{3}{2}}\sqrt{f+a}} + \frac{\left(1-\frac{f+a}{a-f}\right)\arctan\left(\frac{\sqrt{a-f}\tan\left(\frac{x}{2}\right)}{\sqrt{f+a}}\right)}{\sqrt{a-f}(f+a)^{\frac{3}{2}}} + C$$

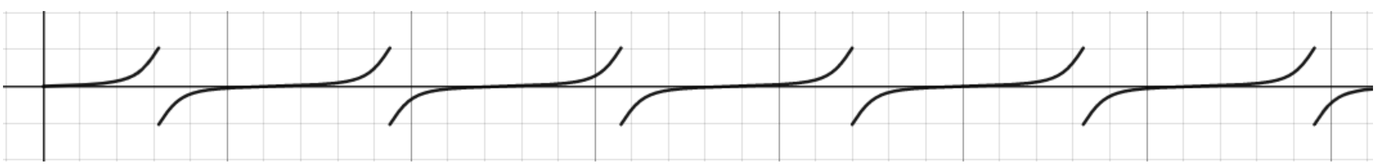
Rewrite/simplify:

$$= \frac{f\sin(x)}{(f-a)(f+a)(f\cos(x)+a)} + \frac{2a\sqrt{a-f}\arctan\left(\frac{\sqrt{a-f}\tan\left(\frac{x}{2}\right)}{\sqrt{f+a}}\right)}{(f-a)^2(f+a)^{\frac{3}{2}}} + C$$

Substituting this in, we get

$$A = b^4\left(\frac{f\sin(\theta)}{2(f-a)(f+a)(f\cos(\theta)+a)} + \frac{a\sqrt{a-f}\arctan\left(\sqrt{\frac{a-f}{f+a}}\tan\frac{\theta}{2}\right)}{(f-a)^2(f+a)^{\frac{3}{2}}}\right) + C$$

However, something in the weird wonderful world of arctan substitutions means that the actual curve we get repeats in the range $A \in [-\frac{1}{2}\pi ab, \frac{1}{2}\pi ab]$ as shown below.

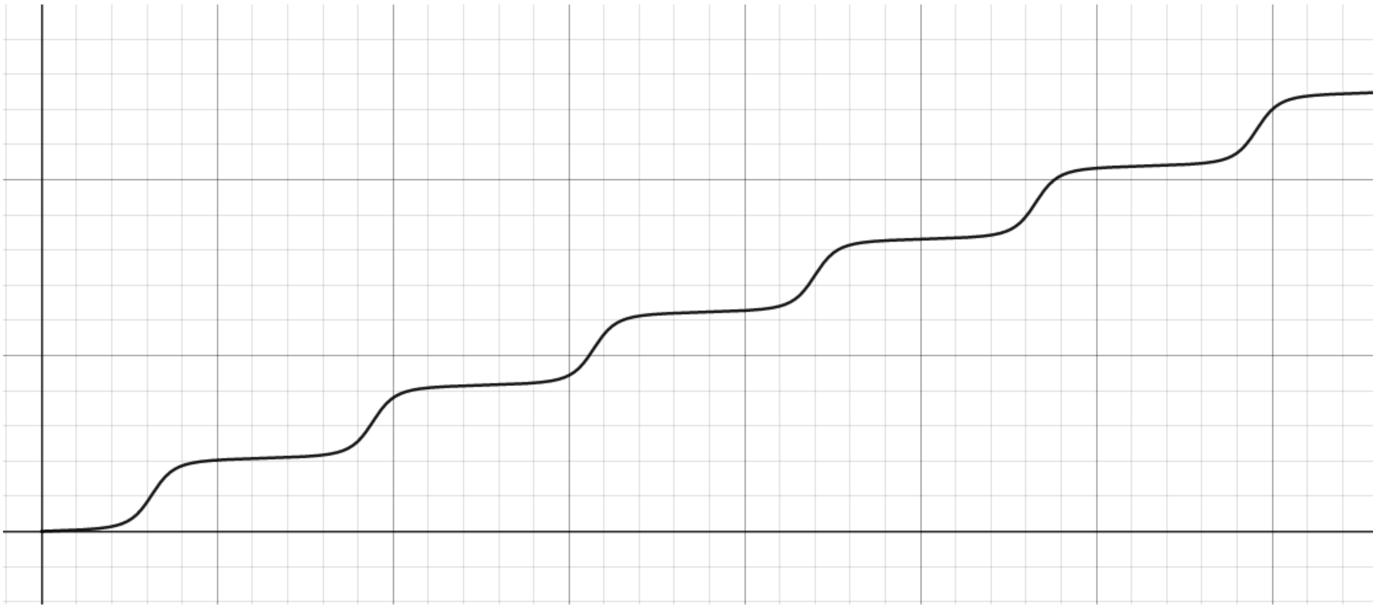


This corresponds to the fact A can be thought of as wrapping around from half of the ellipse to negative half of the ellipse as θ increases over the π radians threshold and then again after every integer multiple of 2π radians.

We therefore add on the following term:

$$\pi ab \left\lfloor \frac{\theta + \pi}{2\pi} \right\rfloor$$

which adds on the area of the ellipse (πab) for every integer multiple of 2π radians along the θ axis, offset by a phase of π radians. This correction is shown in the graph below:



This leaves us with our final formula of

$$A = \pi ab \left\lfloor \frac{\theta + \pi}{2\pi} \right\rfloor + b^4 \left(\frac{f \sin(\theta)}{2(f-a)(f+a)(f\cos(\theta)+a)} + \frac{a\sqrt{a-f}\,\arctan\left(\sqrt{\frac{a-f}{f+a}}\tan\frac{\theta}{2}\right)}{(f-a)^2(f+a)^{\frac{3}{2}}} \right) + C$$