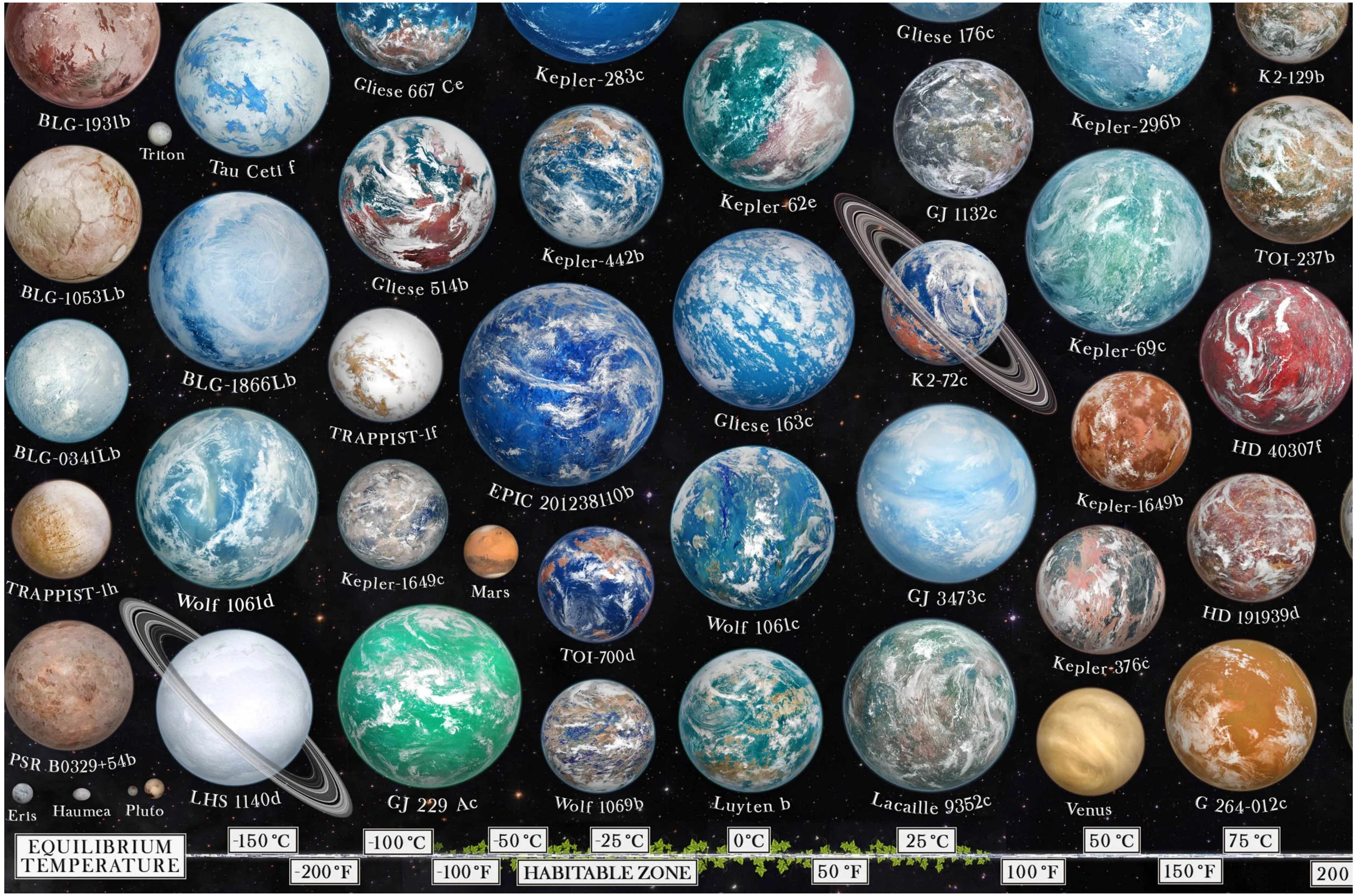
PHYS 246 LECTURE 4: EXOPLANET DETECTION



https://lkwagner.github.io/IntroductionToComputationalPhysics/intro.html

ANNOUNCEMENTS/NOTES

We found a way to accept ipynb files directly. For HW 4, please just upload a single ipynb file (no need to print to PDF). Should be much easier (Thanks to Will!)

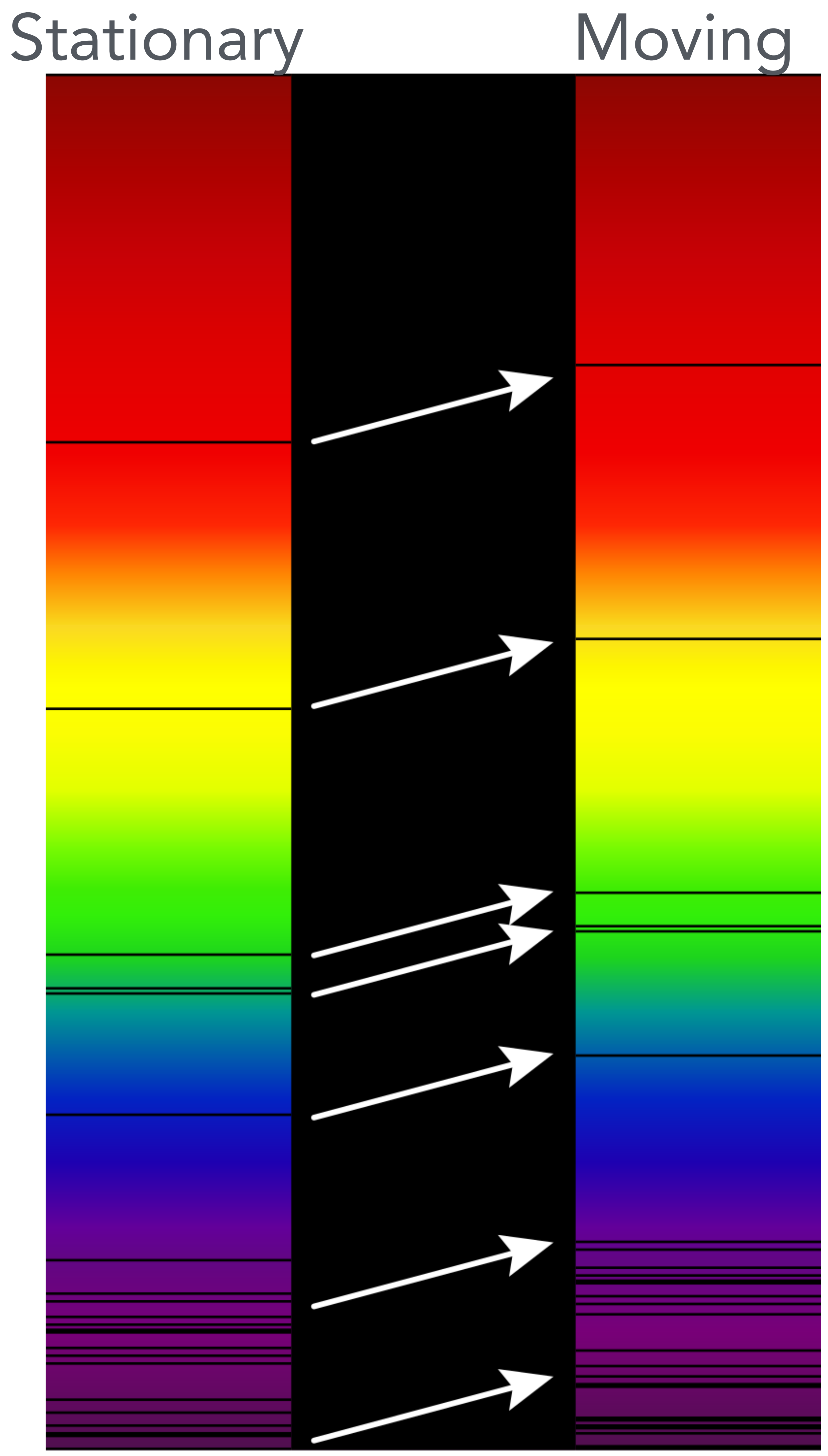
Don't forget to label axes, include docstrings.

Another helpful tip for comments:

```
x = np.random.random( (3,5))
y = np.random.random( (5,3))

z = x.dot(y) # (3,5) . (5,3) -> (3,3)
```

REDSHIFT VS BLUESHIFT



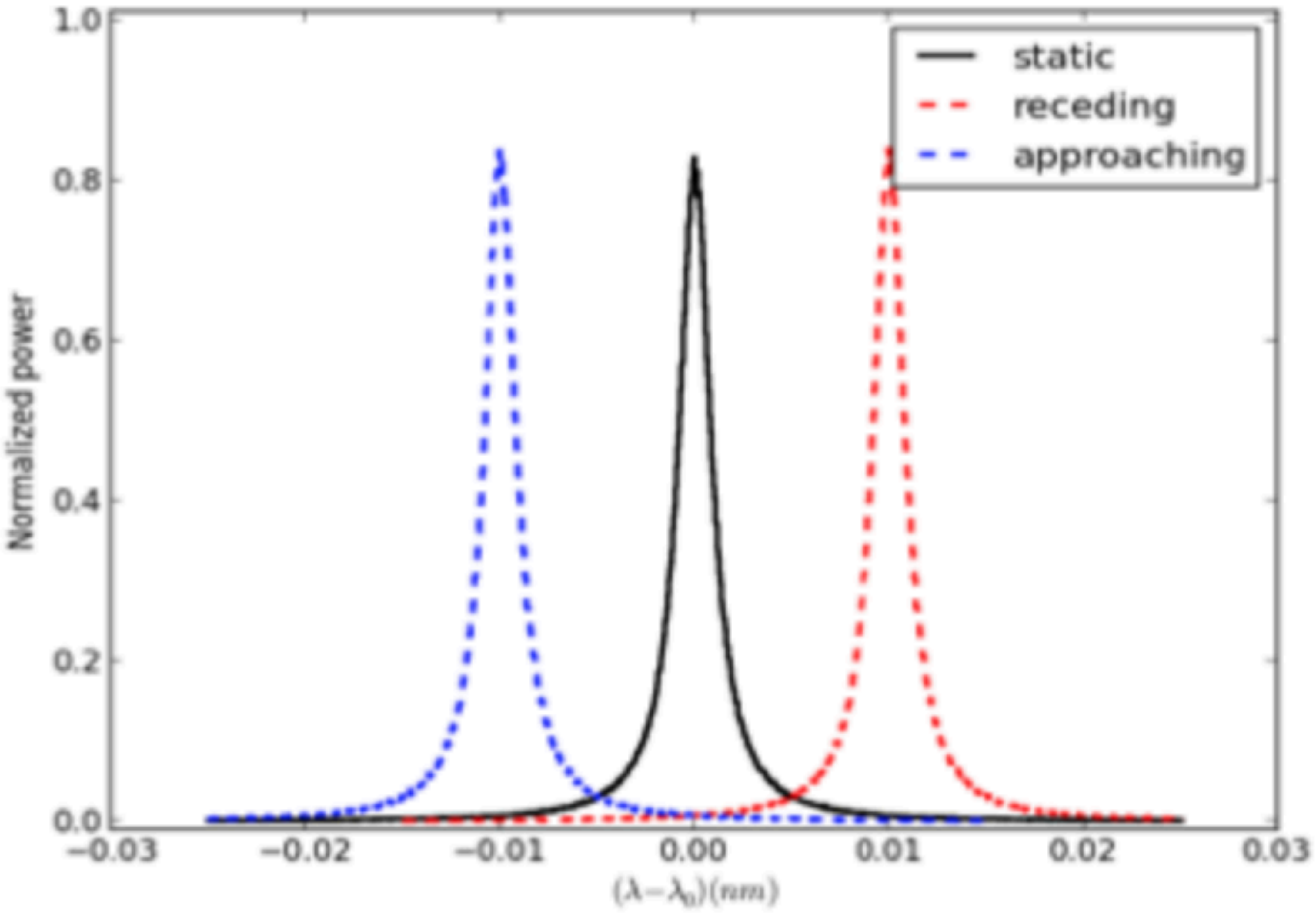
We measure the spectrum of a star.

The missing lines correspond to excitations of elements (differences between energy eigenstates from PHYS 214). We know where they are from a solution of the Schroedinger equation and/or measurements on Earth.

What we actually see is the same lines at slightly different frequencies.

Is the star moving towards us or away from us?

Doppler effect: measuring relative velocities

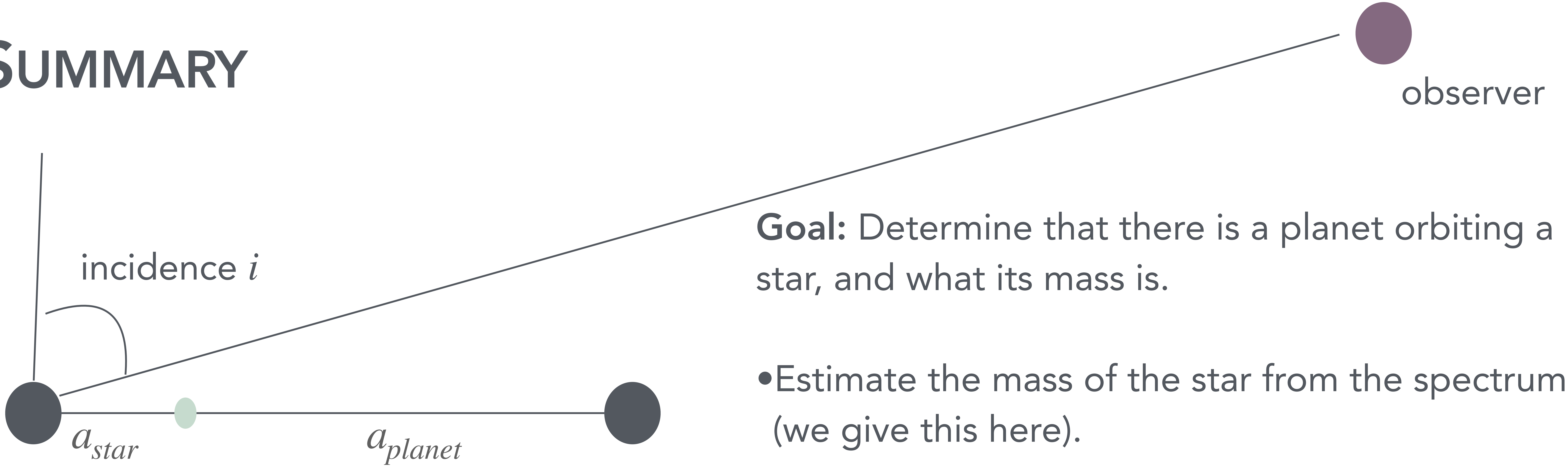


$$\frac{\lambda_{\text{observed}} - \lambda_{\text{source}}}{\lambda_{\text{source}}} = \frac{v_r}{c}$$

"Redshift" vs "blueshift"

This is also how we know the universe is expanding!

SUMMARY

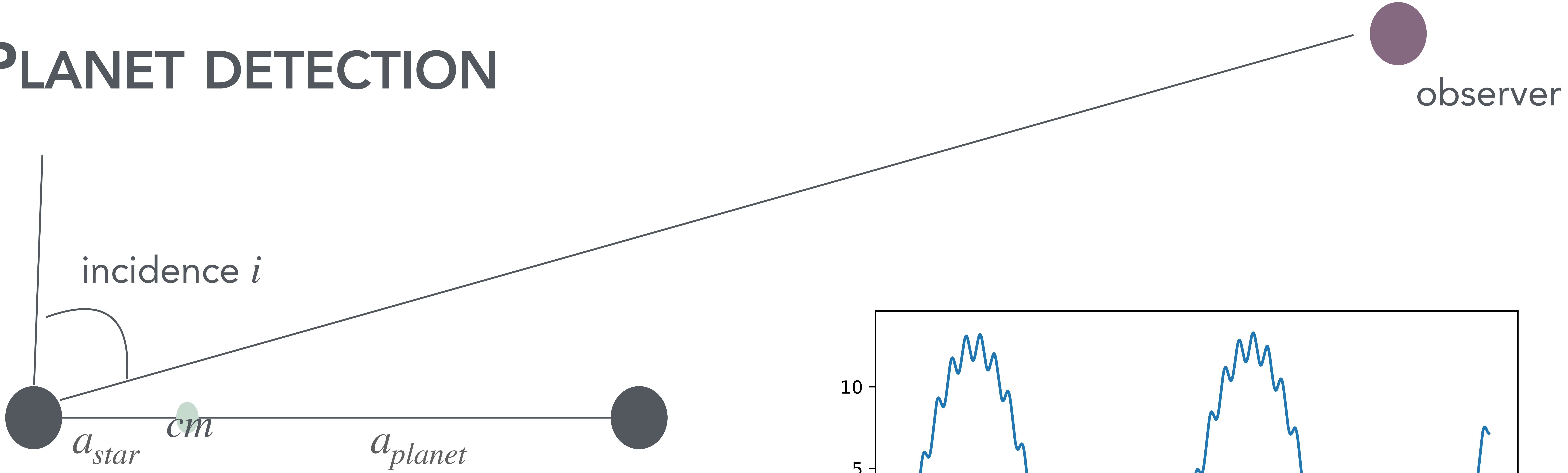


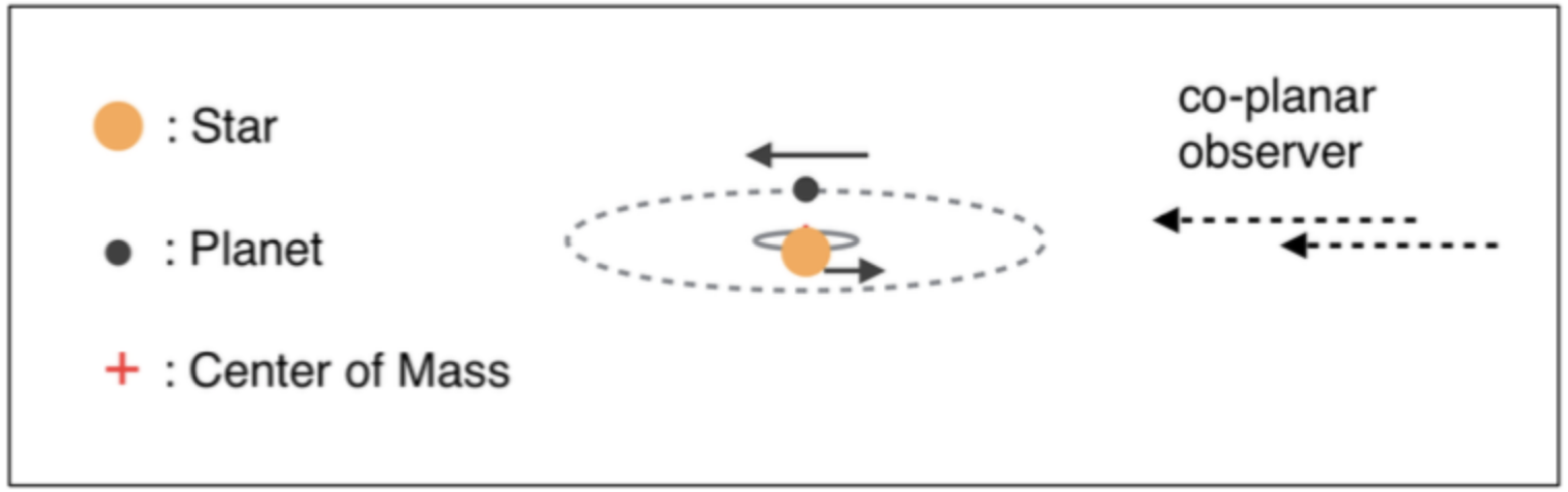
$$m_{\text{projected}} = \left(\frac{m_{\text{star}}^2 P}{2\pi G}\right)^{1/3} v_r$$

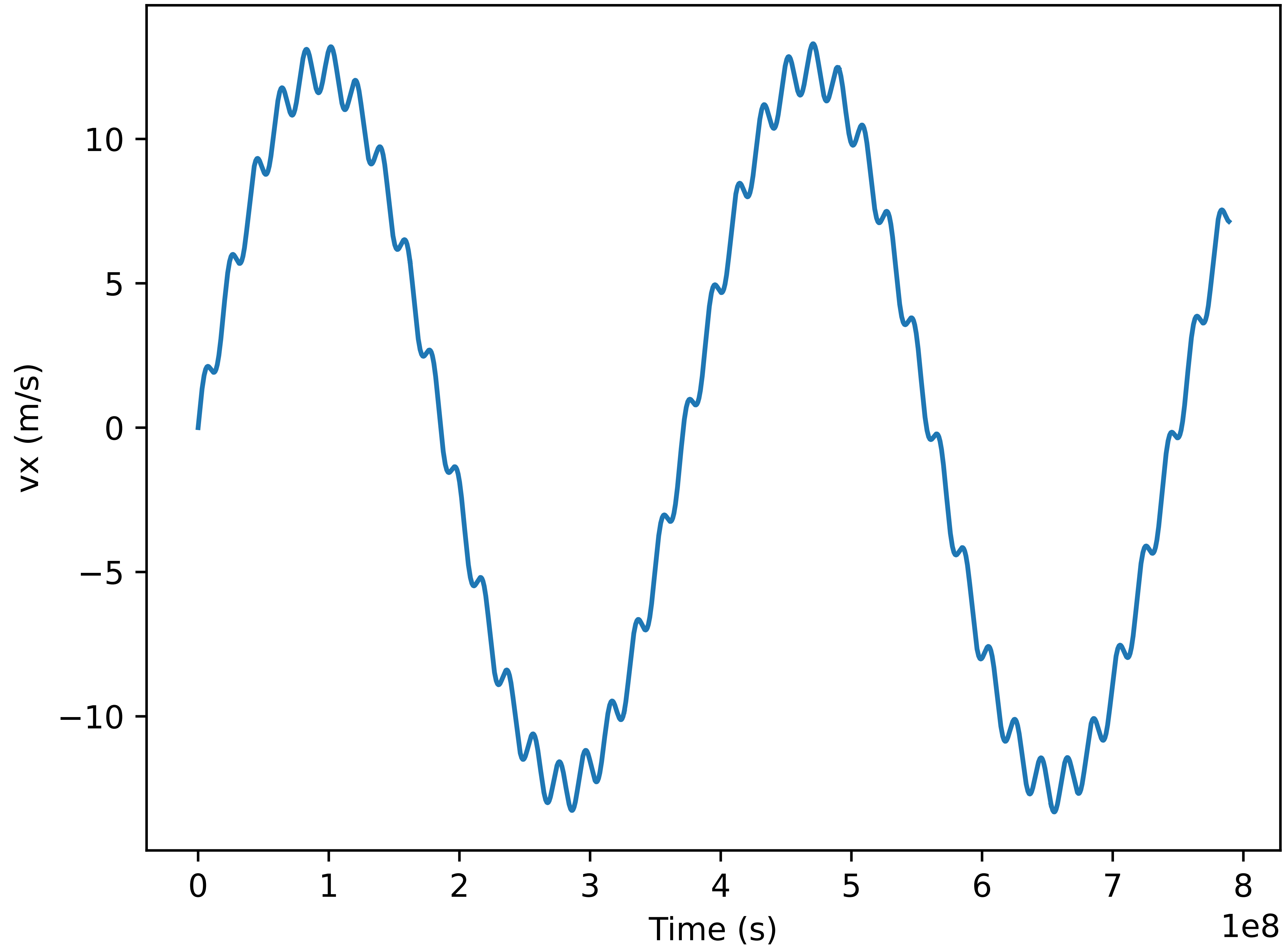
Goal: Determine that there is a planet orbiting a star, and what its mass is.

- Estimate the mass of the star from the spectrum (we give this here).
- Use Doppler effect to measure star's velocity.
- Take the Fourier transform to find the period of the star's velocity P
- Use the max velocity, mass, and period to estimate (a lower bound to) the planet's mass.

PLANET DETECTION







FOURIER TRANSFORM

Would like to write a general function as

$$v(f) = \sum_f c(f) e^{i2\pi fx}$$

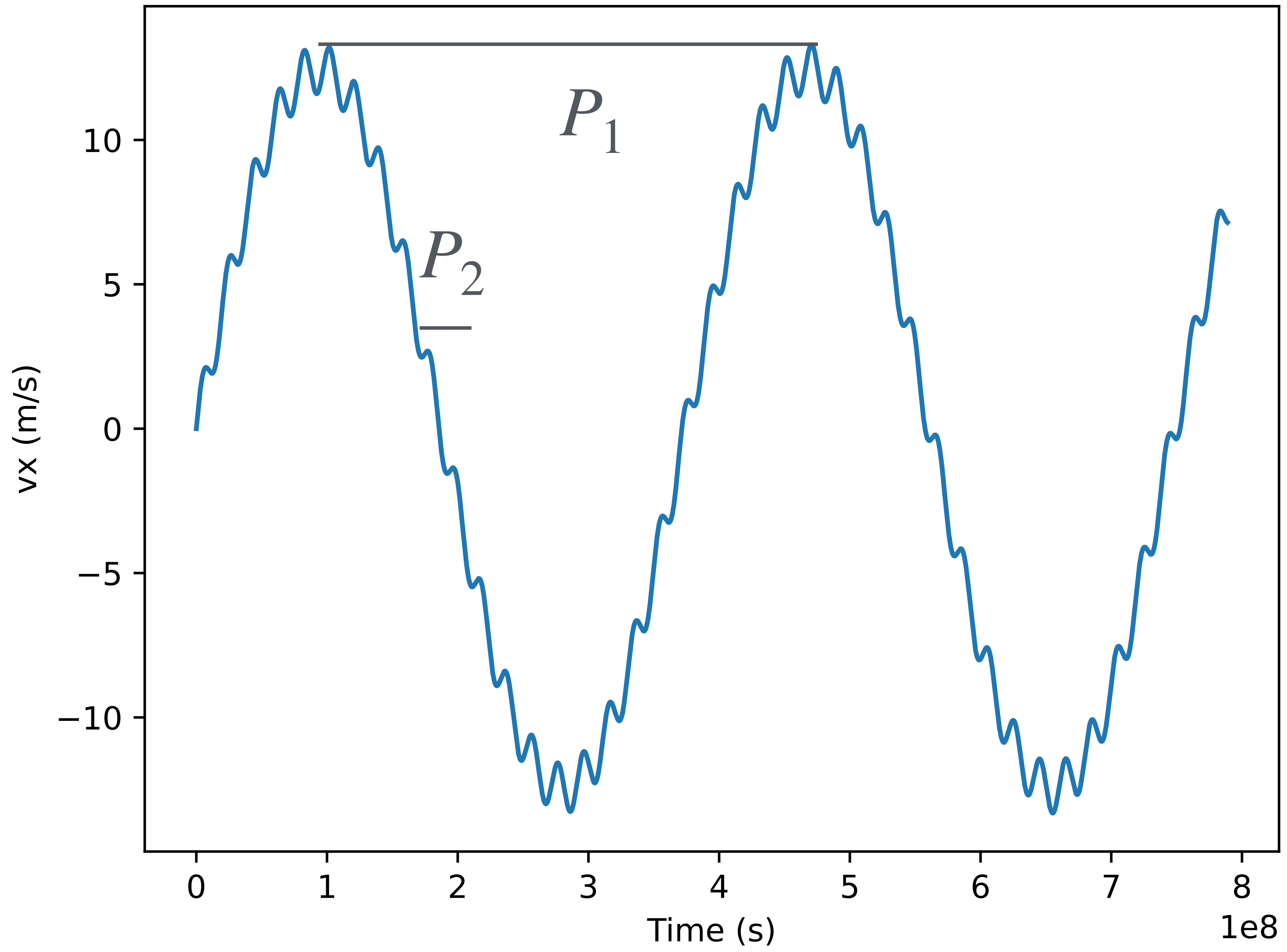
$c(f)$ tells us the frequency components of the function v

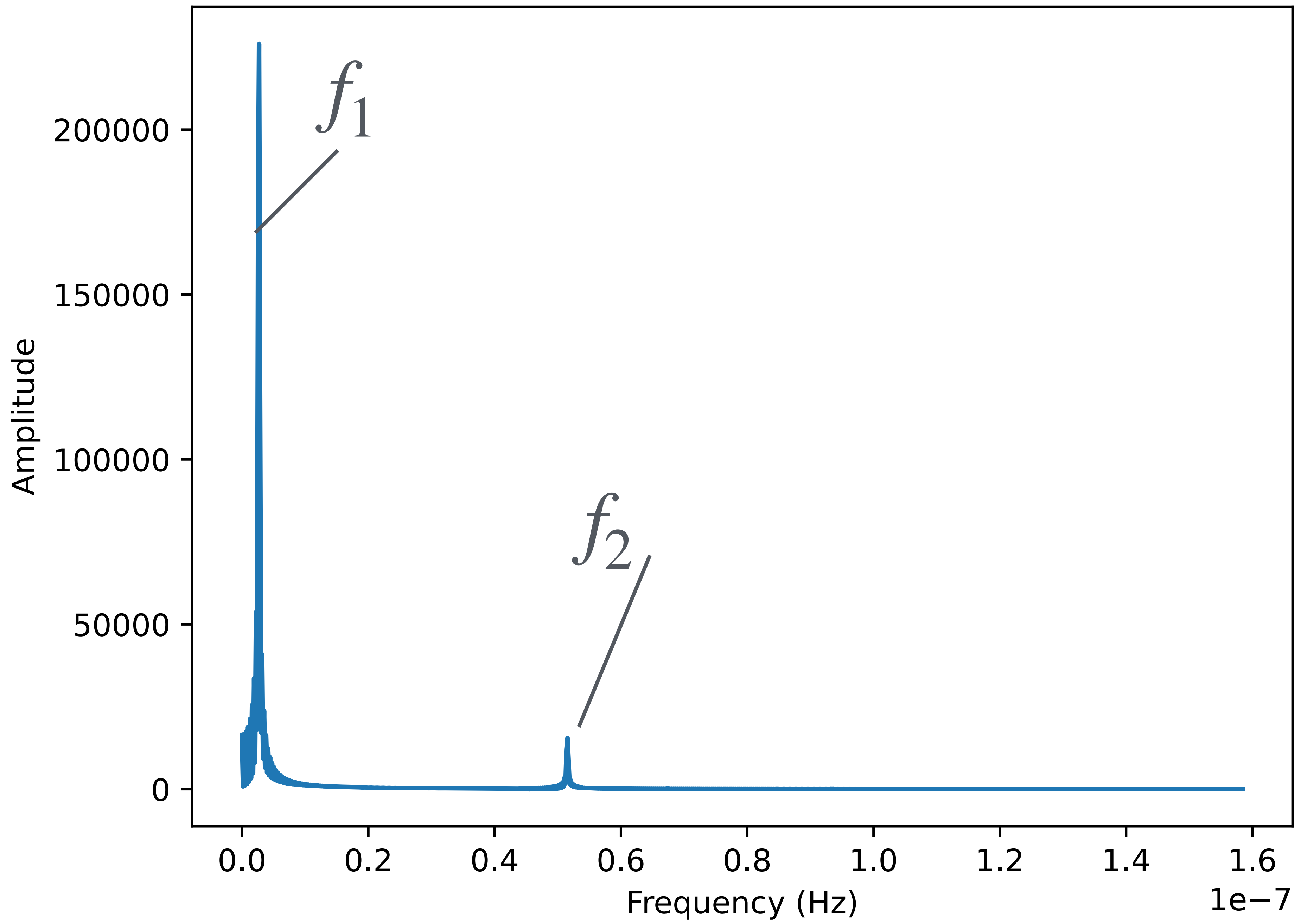
Note that

$$\int e^{-ikx} e^{ik'x} dx = \delta(k - k')$$

So

$$c(f) = \int e^{-i2\pi ft} v(t) dt$$



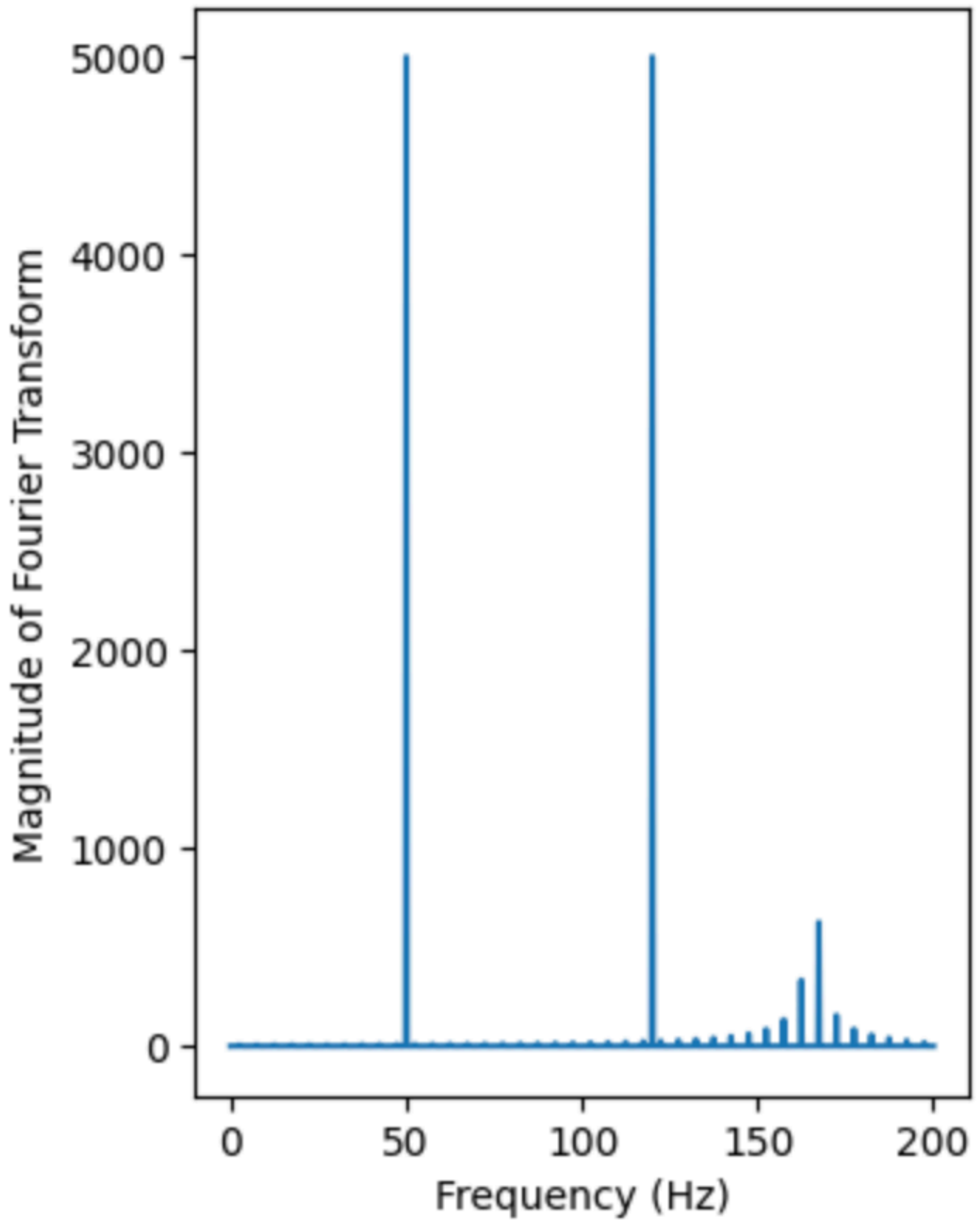


TESTING YOUR FOURIER TRANSFORM

```python
def fourier_transform(t, v, f):
    """
    Compute the Fourier transform of a signal from v(t) to v(f) at the frequencies specified in f.
    t: np.array (npoints,) time (s)
    v: np.array (npoints,) velocity (V)
    f: np.array (number_frequency,) frequency (Hz)
    returns:
    v_f: np.array (number_frequency,) Fourier transform of v at frequencies f
    """
```

```python
t = np.linspace(0, 4001, 10000)
v = np.sin(2 * np.pi * 50 * t) + np.sin(2 * np.pi * 120 * t)
f = np.linspace(0, 200, 1001)
```

```python
V_f_loop = fourier_transform(t, v, f)
```



SPEED

Which do you think is fastest?

1
```
V_f = np.zeros_like(f, dtype=complex)  # initialize Fourier transform array
for i in range(len(f)):
    for j in range(len(t)):
        V_f[i]+= v[j] * np.exp(-2j * np.pi * f[i] * t[j])   # compute Fourier transform at frequency f[i]
return V_f
```

2
```
V_f = np.zeros_like(f, dtype=complex)  # initialize Fourier transform array
for i in range(len(f)):
    V_f[i] = np.sum(v * np.exp(-2j * np.pi * f[i] * t))   # compute Fourier transform at frequency f[i]
return V_f
```

3
```
V_f = np.sum(v[:,np.newaxis] * np.exp(-2j * np.pi * f[np.newaxis, :] * t[:, np.newaxis]), axis=0)
return V_f
```

```
Time for loop version: 5.1826 seconds.
Time for one-loop numpy version: 0.1300 seconds.
Time for all numpy version: 0.1536 seconds.
```

DERIVATION OF $m_{\text{projected}} = \left(\frac{m_{\text{star}}^2 P}{2\pi G}\right)^{1/3} v_r$

3rd law

$$m_{star}a_{star} = m_{planet}a_{planet}$$

$$P = \frac{2\pi a_{planet}}{v_{planet}} = \frac{2\pi a_{star}}{v_{star}}$$

Kepler

$$\frac{G m_{\text{star}} m_{\text{planet}}}{(a_{\text{planet}} + a_{\text{start}})^2} = \frac{m_{\text{planet}} v_{\text{planet}}^2}{a_{\text{planet}}}$$

$$\frac{G m_{\text{star}}}{a_{\text{planet}}^2} \approx \frac{v_{\text{planet}}^2}{a_{\text{planet}}} = \frac{1}{a_{\text{planet}}}\left(\frac{2\pi a_{\text{planet}}}{P}\right)^2$$

$$G m_{\text{start}} \approx \left(\frac{2\pi}{P}\right)^2 a_{\text{planet}}^3$$

Algebra

$$\begin{aligned}
GM_{\text{star}} &\approx \left(\frac{2\pi a_{\text{planet}}}{P}\right)^2 a_{\text{planet}} \\
&= \left(\frac{2\pi a_{\text{star}}}{P}\frac{m_{\text{star}}}{m_{\text{planet}}}\right)^2 a_{\text{star}}\frac{m_{\text{star}}}{m_{\text{planet}}} \\
&= \left(v_{\text{star}}\frac{m_{\text{star}}}{m_{\text{planet}}}\right)^2 \frac{P}{2\pi}\frac{2\pi a_{\text{star}}}{P}\frac{m_{\text{star}}}{m_{\text{planet}}} \\
&= \left(v_{\text{star}}\frac{m_{\text{star}}}{m_{\text{planet}}}\right)^2 \frac{P}{2\pi} v_{\text{star}}\frac{m_{\text{star}}}{m_{\text{planet}}} = \frac{P}{2\pi}\left(v_{\text{star}}\frac{m_{\text{star}}}{m_{\text{planet}}}\right)^3
\end{aligned}$$

$$\begin{aligned}
G m_{\text{star}} &\approx \frac{P}{2\pi}\left(v_{\text{star}}\frac{m_{\text{star}}}{m_{\text{planet}}}\right)^3 \\
G\frac{m_{\text{planet}}^3}{m_{\text{star}}^2} &= \frac{P}{2\pi} v_{\text{star}}^3 \\
\frac{m_{\text{planet}}^3}{m_{\text{star}}^2} &= \frac{P\,(v_{\text{star}}\sin i)^3}{2\pi G \sin^3 i} \\
m_{\text{planet}}^3 \sin^3 i &= m_{\text{star}}^2 \frac{P}{2\pi G}(v_{\text{star}}\sin i)^3 \\
m_{\text{planet}} \sin i &= \left(\frac{m_{\text{star}}^2 P}{2\pi G}\right)^{1/3} v_{\text{star}}\sin i \\
&= \left(\frac{m_{\text{star}}^2 P}{2\pi G}\right)^{1/3} v_r \\
&\equiv m_{\text{projected}}.
\end{aligned}$$

DOPPLER EFFECT

Static Source

λ

Velocity

Car (source) and person (receiver) are both at rest

Frequency

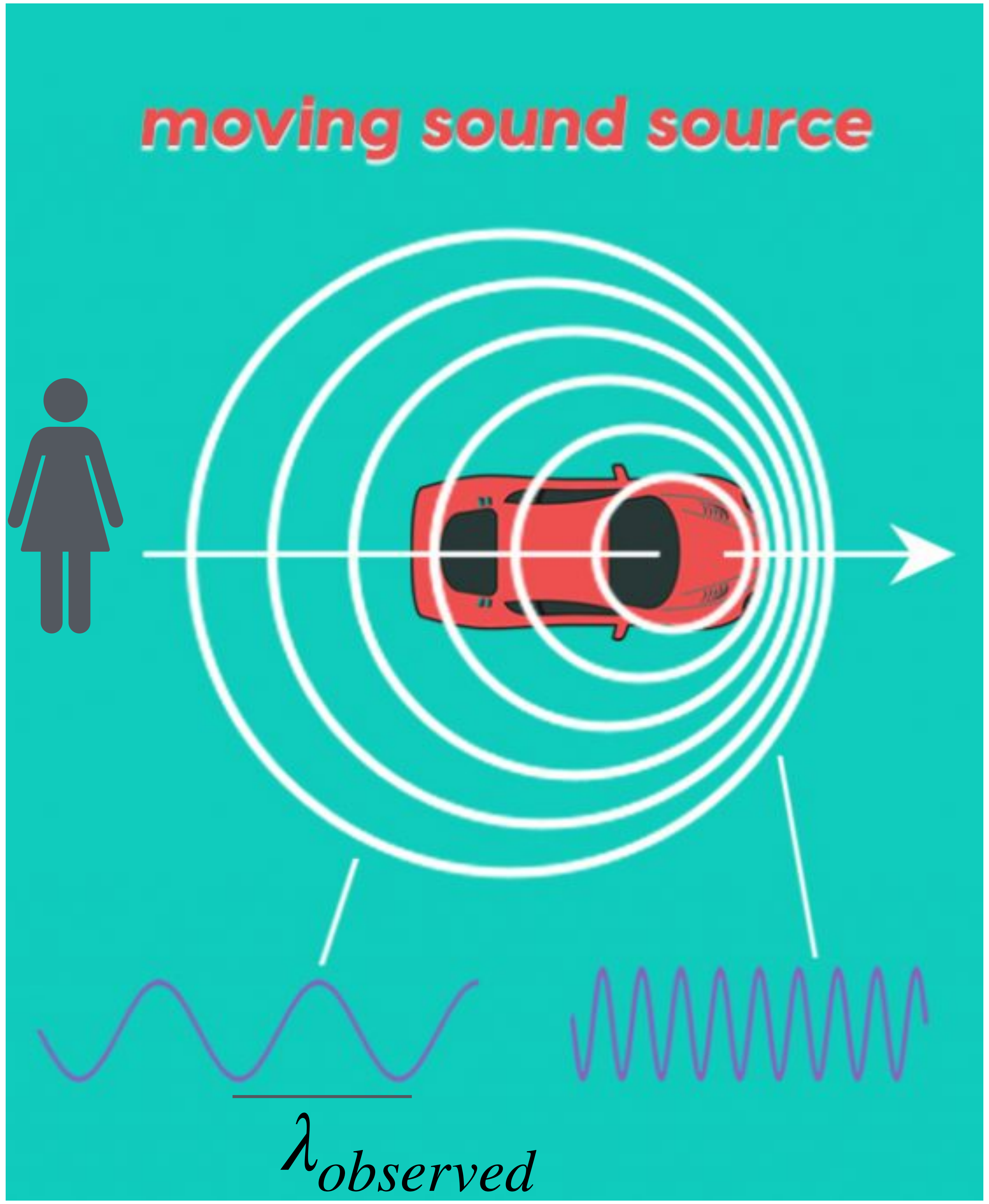
$$f_{source} = \frac{v}{\lambda_{source}}$$

Wave's velocity

Wavelength

Doppler effect

Redshift: $\downarrow f$ or in other words $\uparrow \lambda$



$$\lambda_{\text{observed}} > \lambda_{\text{source}}$$

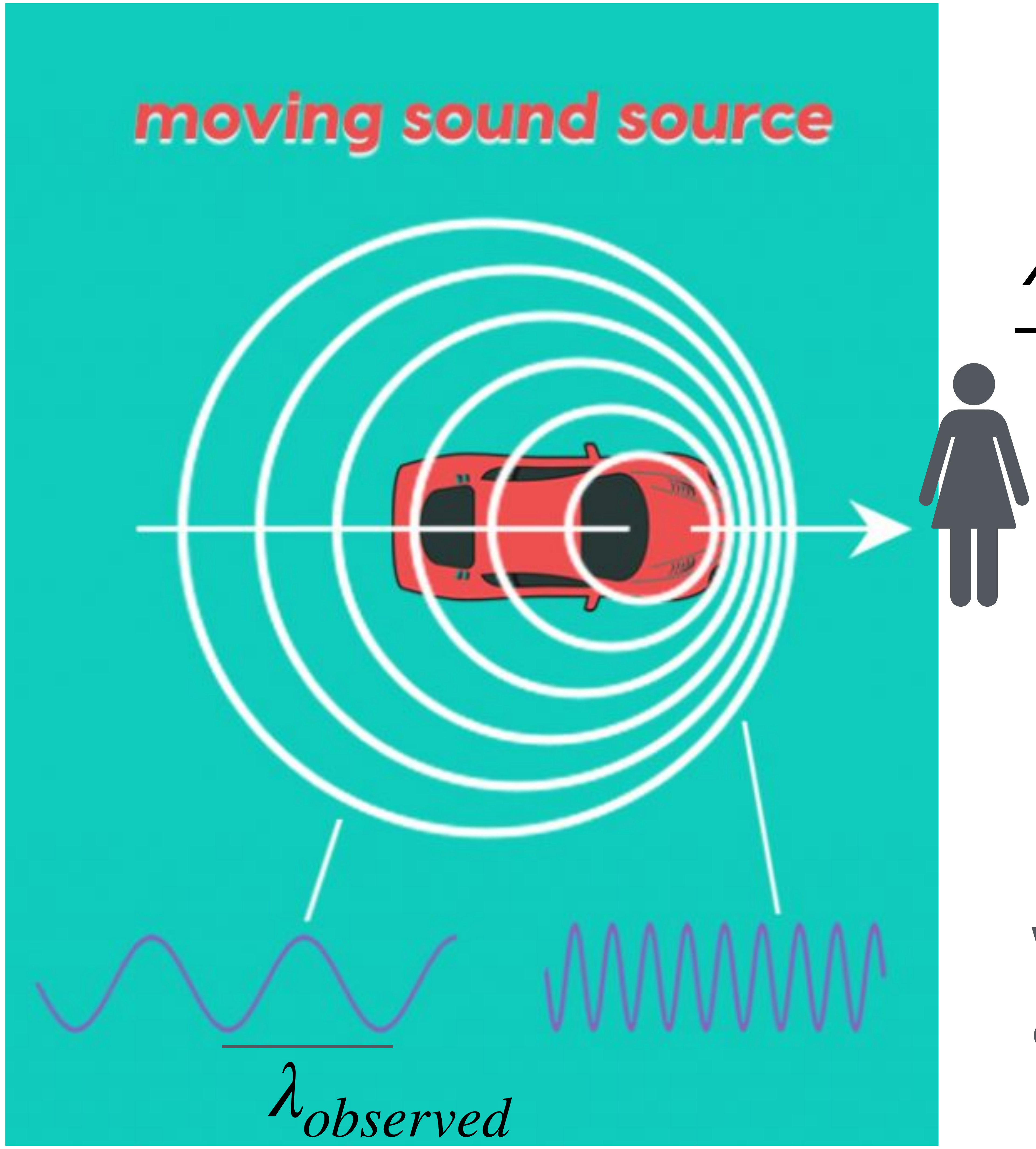
$$\frac{\lambda_{\text{observed}} - \lambda_{\text{source}}}{\lambda_{\text{source}}} = \frac{v_r}{c}$$

$$\frac{f_{\text{source}} - f_{\text{observed}}}{f_{\text{observed}}} = \frac{v_r}{c}$$

When $v_r > 0$, motion away from observer

DOPPLER EFFECT

Blueshift: $\uparrow f$ or in other words $\downarrow \lambda$



$$\lambda_{\text{observed}} < \lambda_{\text{source}}$$

$$\frac{\lambda_{\text{observed}} - \lambda_{\text{source}}}{\lambda_{\text{source}}} = \frac{v_r}{c}$$

$$\frac{f_{\text{source}} - f_{\text{observed}}}{f_{\text{observed}}} = \frac{v_r}{c}$$

When $v_r < 0$, motion towards observer