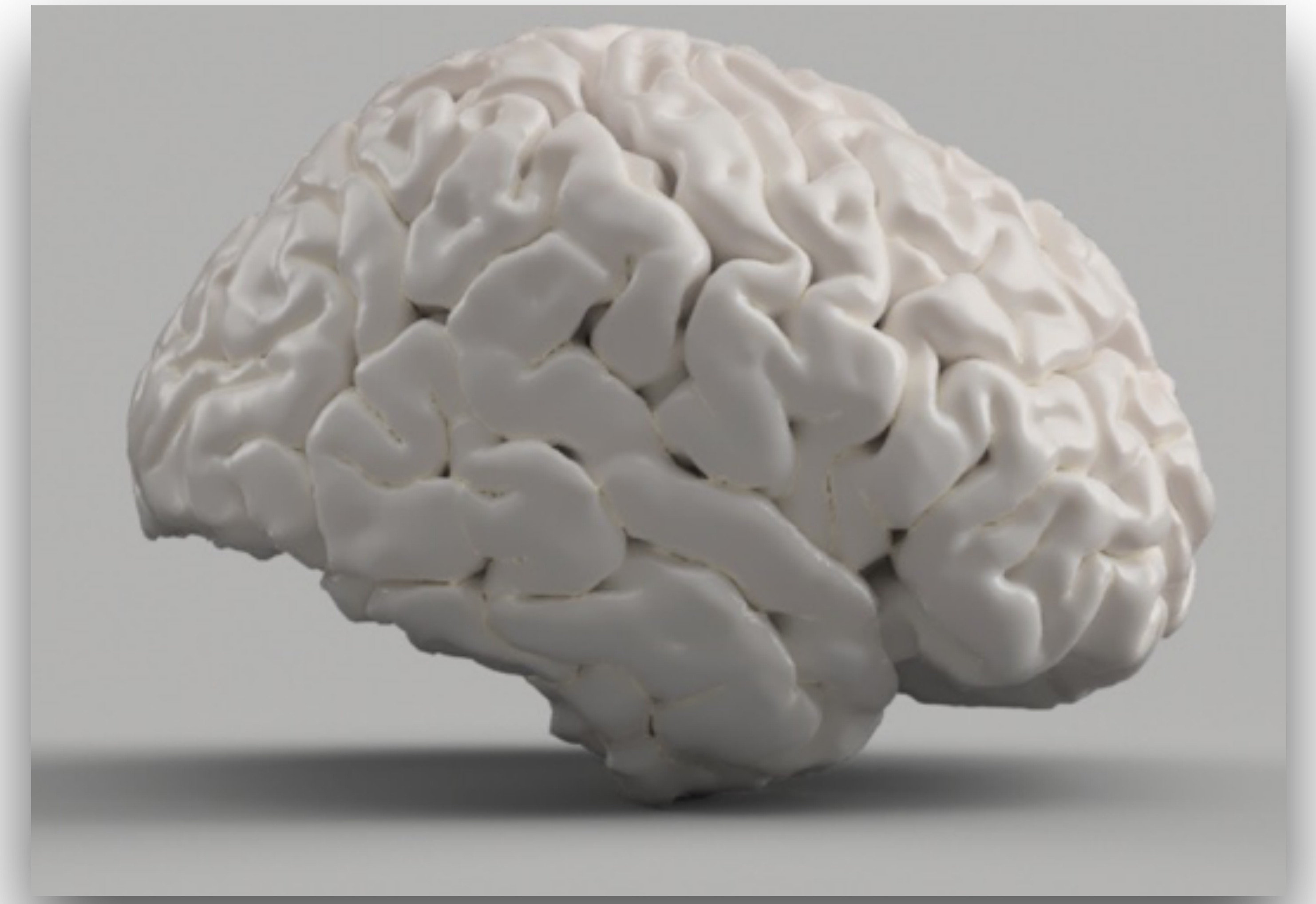
COGS109: Lecture 4



Sampling, Discretization, Filtering

July 11, 2023

Modeling and Data Analysis

Summer Session 1, 2023

C. Alex Simpkins Jr., Ph.D.
RDPRobotics LLC | Dept. of CogSci, UCSD

Plan for the lecture

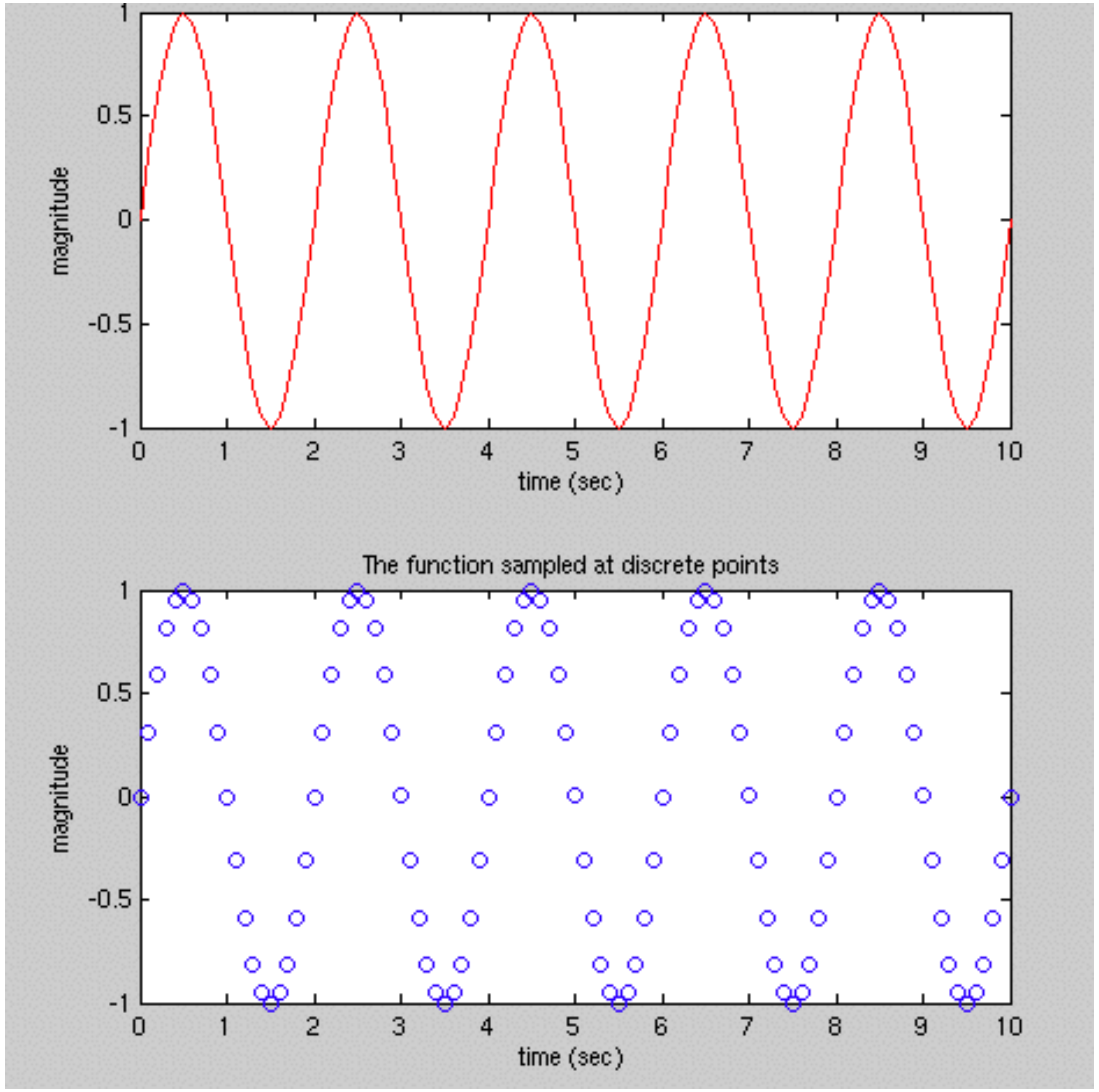
- Discussion of groups, repos and timing
- Discussion of paper review
- Sampling, discretization and filtering

Sampling, discretization, filtering

- Review of continuous vs. discrete quantities
- Analog vs. Digital
- Discretization, sampling, aliasing
- Filter theory, frequency response, filter types
- Linearity

Continuous vs. Discrete quantities

- Information storage
  - **Continuous** signals have information at every point in time
  - **Discrete** signals have info only at specified intervals (fixed or variable)



Examples of continuous and discrete systems

- Continuous or discrete?
  - # of people in this class
  - # of Time zones
  - Time
  - Answers on multiple choice tests
  - A Sound
  - Body temperature

- Discrete
- Discrete
- Continuous
- Discrete
- Continuous
- Continuous

Analog vs. Digital quantities

- Information storage
  - **Analog** contains infinite information
  - **Digital** contains limited information, depending on the number of bits of information the digital value can store
    - 0 or 1 in each bit means each bit multiplies the possible combinations of numbers by 2
    - 2^4 = 0-15 (a 4-bit number, 16 different values)
    - 2^8 = 0-255 (an 8-bit number, 256 different values)
    - 2^16 = 0-65535 (a 16-bit number, 65536 different values)

More on digital quantities

- Measuring an EEG boils down to recording a sequence of numbers into computer memory, stored in values of a specific size, such as 8 bit numbers.
  - i.e. signal is 0-5V, digitized with 8 bit ***precision*** would yield a ***resolution*** of 5V/256 = 0.020V, or 20mV (mV = ‘milli-Volts’)
  - **Resolution -** defined as the smallest quantity which can be reliably measured
  - **Digital Precision -** The number of bits of information contained in a digital quantity
- Also important for computations
  - Round off errors can accumulate
    - Example
      - 2.245+3.432+1.234 = 6.911
      - 2+3+1 = 6, and that’s only 3 samples! Imagine 1000/sec (1kHz) !
  - More on this later

Discretization

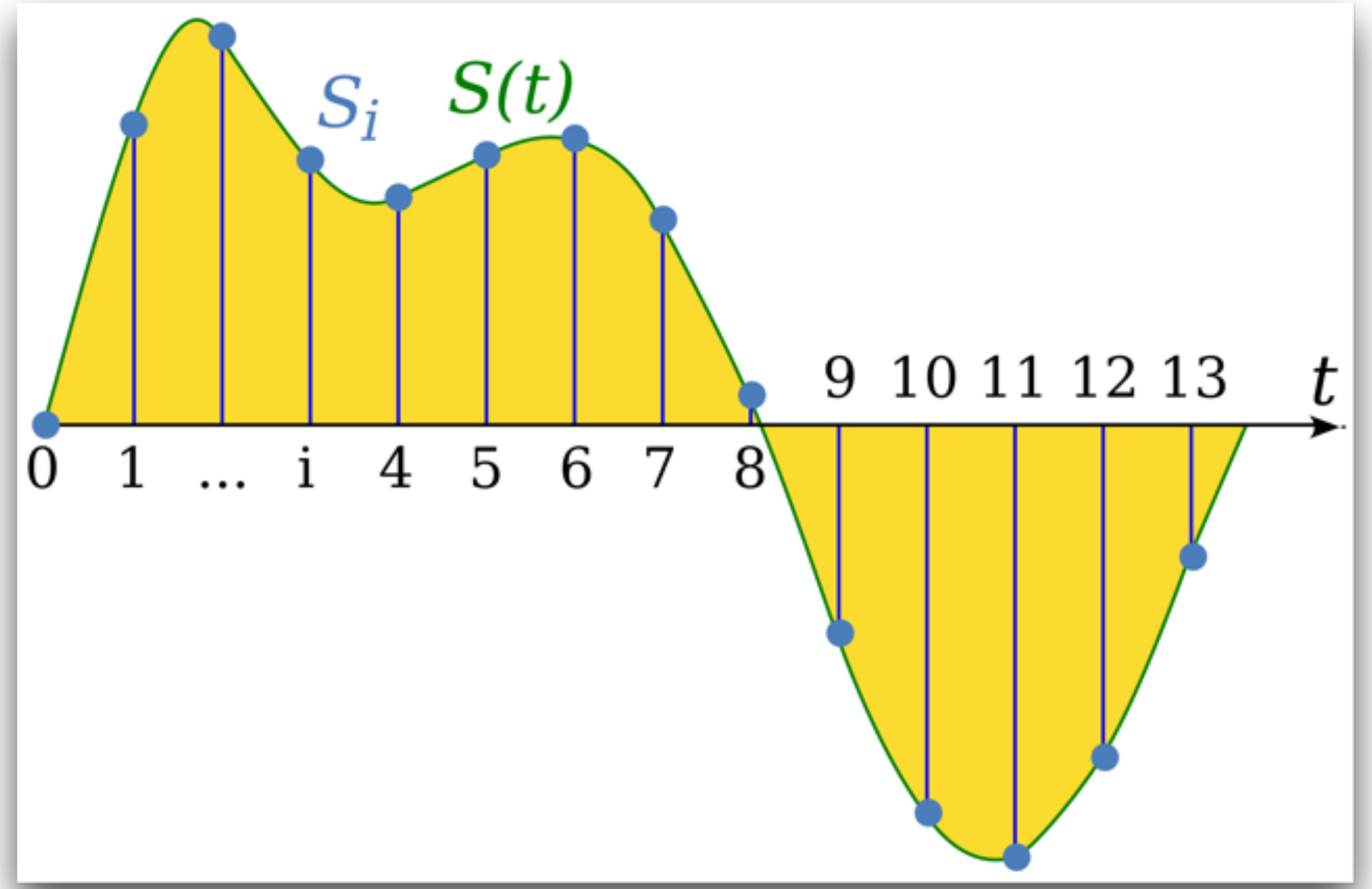
- Measuring a continuous (analog) signal means capturing information at specified (fixed or variable) intervals
  - **Sampling frequency -** the frequency at which data is recorded from a signal (Typically in Hz, ie 5kHz)
- When capturing data, or when manipulating data which has been discretized, there are several issues to consider
  - Aliasing (not the TV show:)
  - Sampling rates
  - Post-processing – filtering data to remove unwanted information while retaining desired information

Sampling

- **Sample -** We record data at specific points in time

- **Period -** The time between samples, ***T* [sec]**

- **Sample frequency -** The frequency of sampling, *f* [Hz]



$$f = \frac{1}{\Delta T}$$

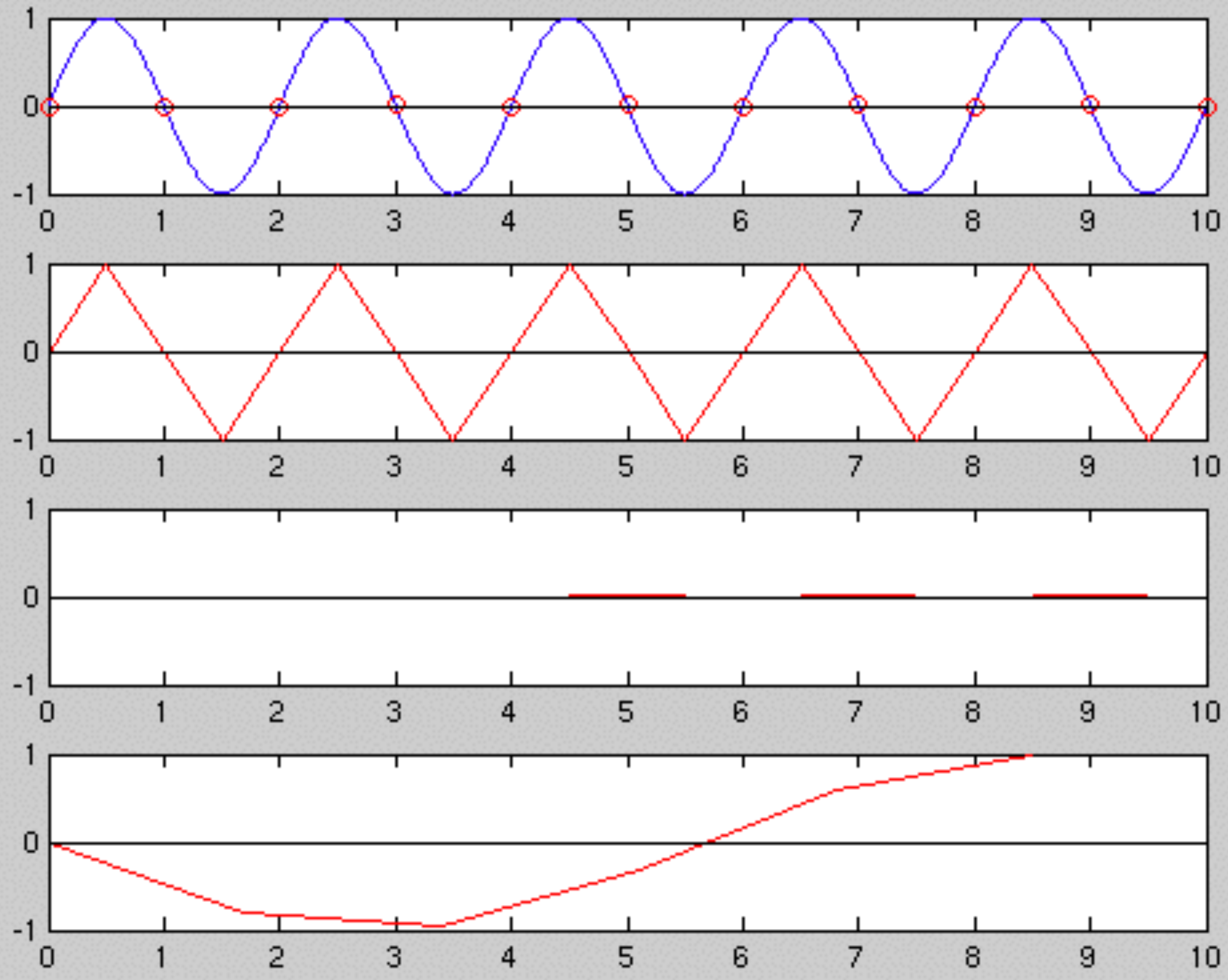
https://en.wikipedia.org/wiki/Sampling_%28signal_processing%29

Nyquist and Sampling

- Stories
  - Running in the dark with periodic lights on the ground, with sharp turns
  - Ping pong (no sound, periodic view of the system)
- As a rule of thumb, you must sample AT LEAST twice as fast as the highest frequency you want to measure
  - **Nyquist frequency -** max freq. that can be measured [Hz]
  - **Nyquist rate -** sampling frequency (which is 2x the nyquist frequency) required to sample at the nyquist frequency
  - 20 times as fast is better
  - Filter out higher frequency components

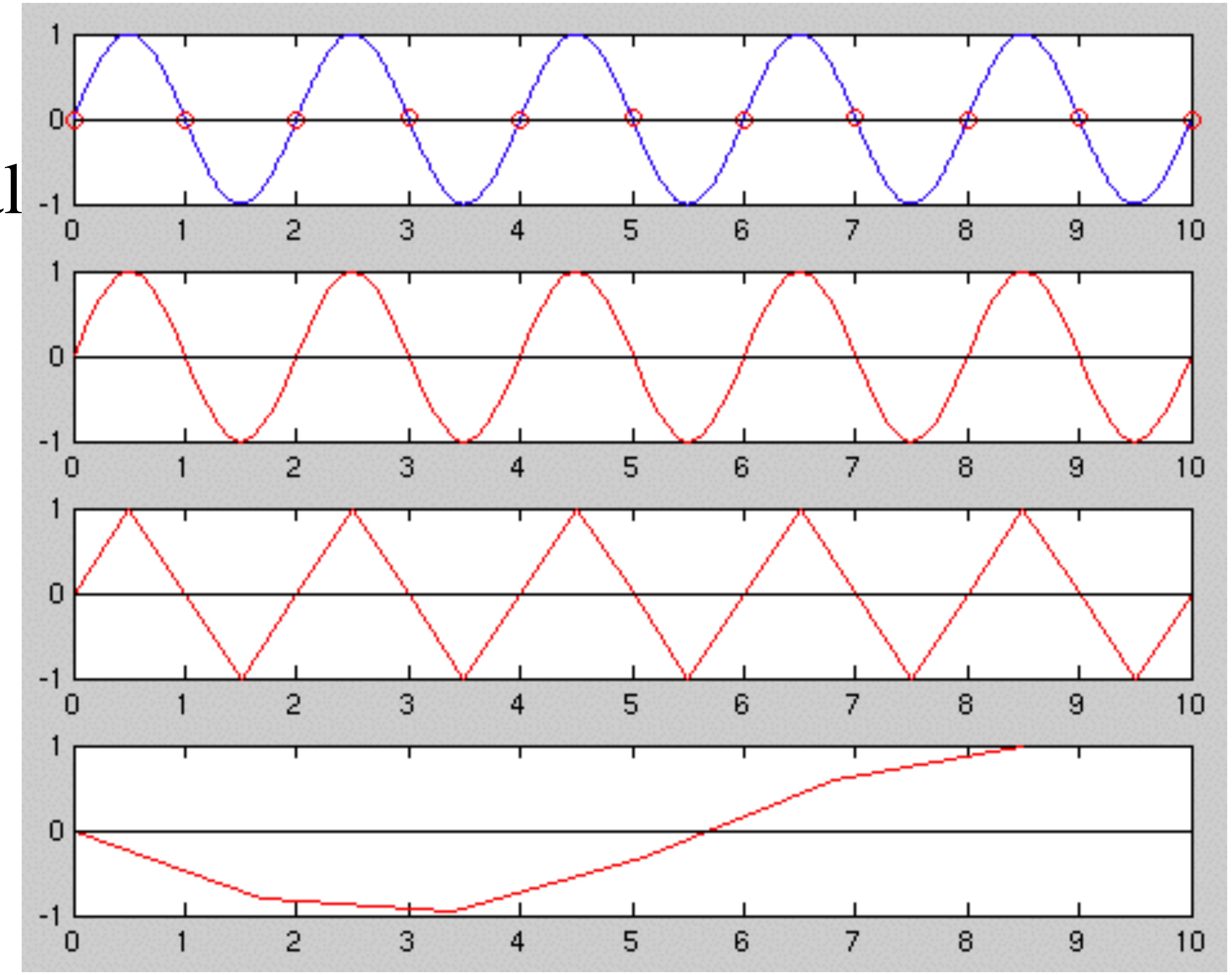
Nyquist frequency

$$f_n = \frac{1}{2\Delta T}$$


1
0
-1
0 1 2 3 4 5 6 7 8 9 10

What do we see in this picture?

- **Aliasing -** the corrupting of a signal by components of higher frequencies overlapping into the lower frequency



How do we solve this?

- Filter out the frequencies we don’t want
  - Low pass filter
  - High pass filter

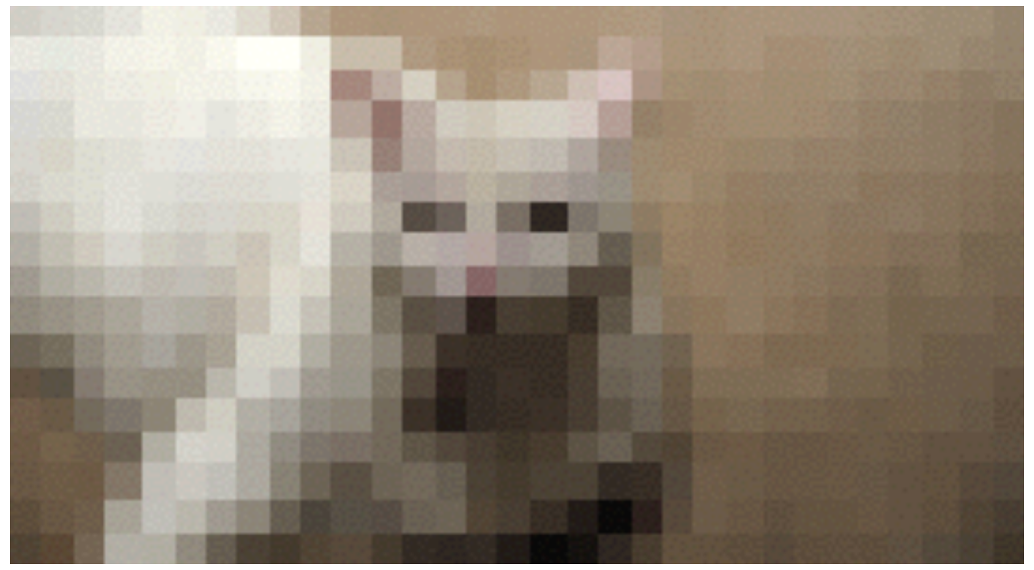
Examples: Visual discretization

- Color shading

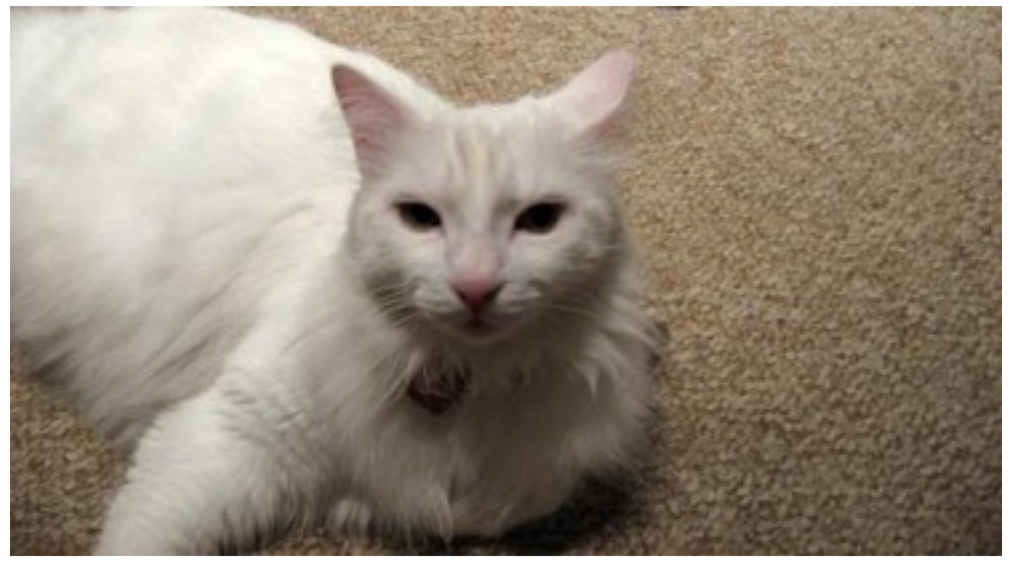
- Color and visual boundaries:

Few colors and low spatial resolution

Low spatial resolution only

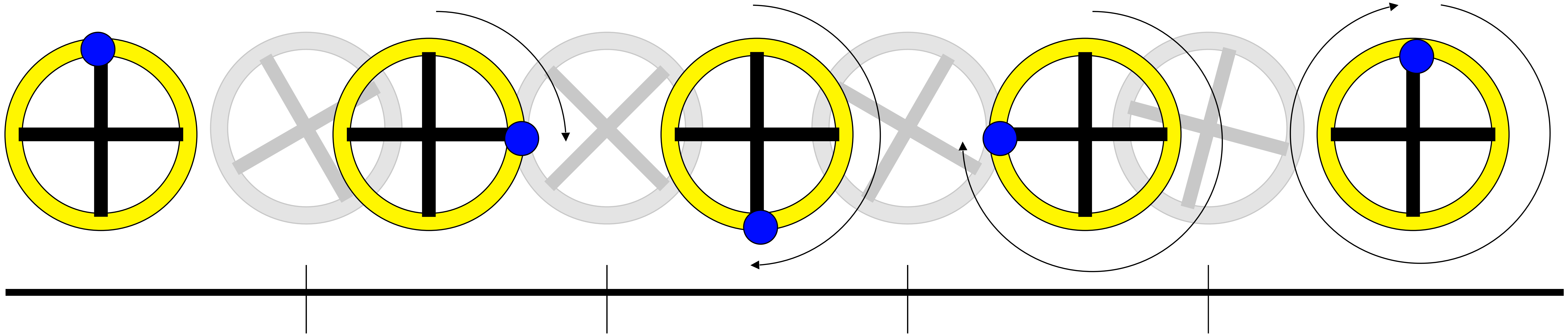


High spatial resolution and colors

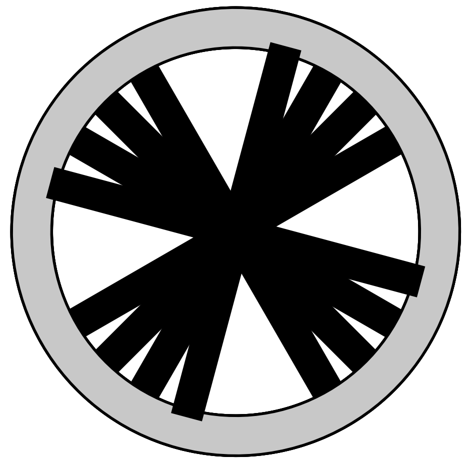


Example: Sampling and Aliasing

- The wheel spokes example…<Live demo>



- We're sampling at too slow a rate to accurately see the spokes rotate, and at a particular rotational velocity of the wheel, we see an 'aliased' reverse rotation!

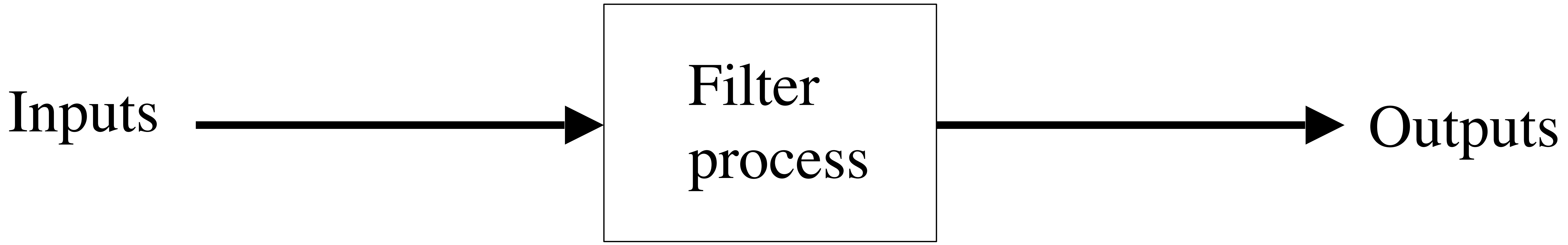


Obviously aliasing can be bad…

- Aliasing can lead to improper interpretations of data
  - **So what do we do about it?**
    - We must first sample at twice the rate of the fastest signal we care about
    - Filter our data (humans do this, and so do cognitive scientists!)

Thus we filter our data…

- **Filter** - an operation or process which alters input data according to some mathematical relationship or heuristic rule to produce output data which is more desirable



Computational filtering

- *Noisy auditory data can be filtered to remove undesired signals*
- *EEG signals can be filtered to remove 60Hz noise from AC lines nearby*
- *Other sensor signals can be filtered to improve results*

Frequency Response

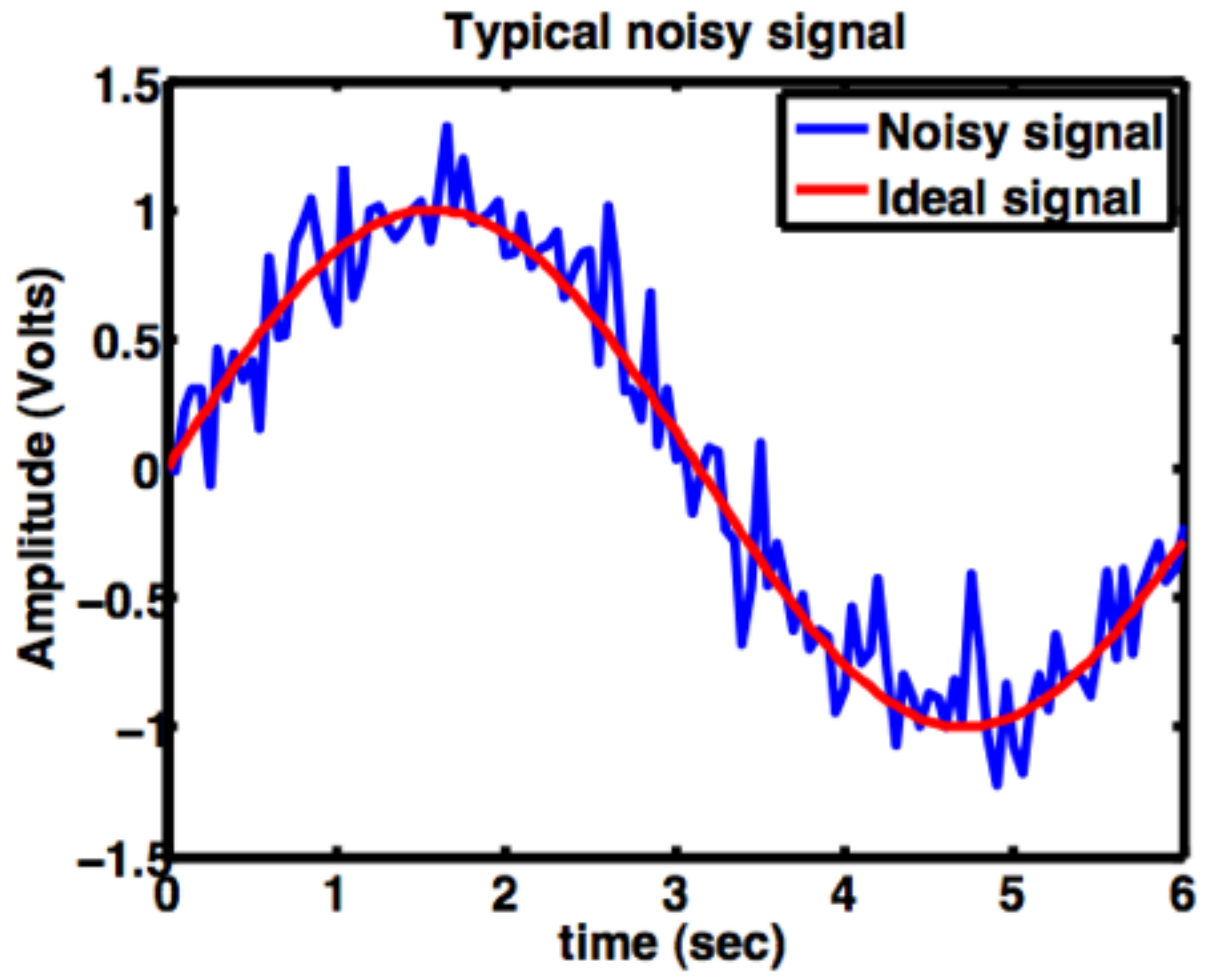
- Linearity of systems vs. nonlinearity
- The response of a linear system to a sinusoidal input is a sinusoidal output with the amplitude and phase shifted in some way
    - <demo volunteer needed>
- This is useful for characterizing the behavior of some signal over a range of possible input frequencies
- Example with the chalk

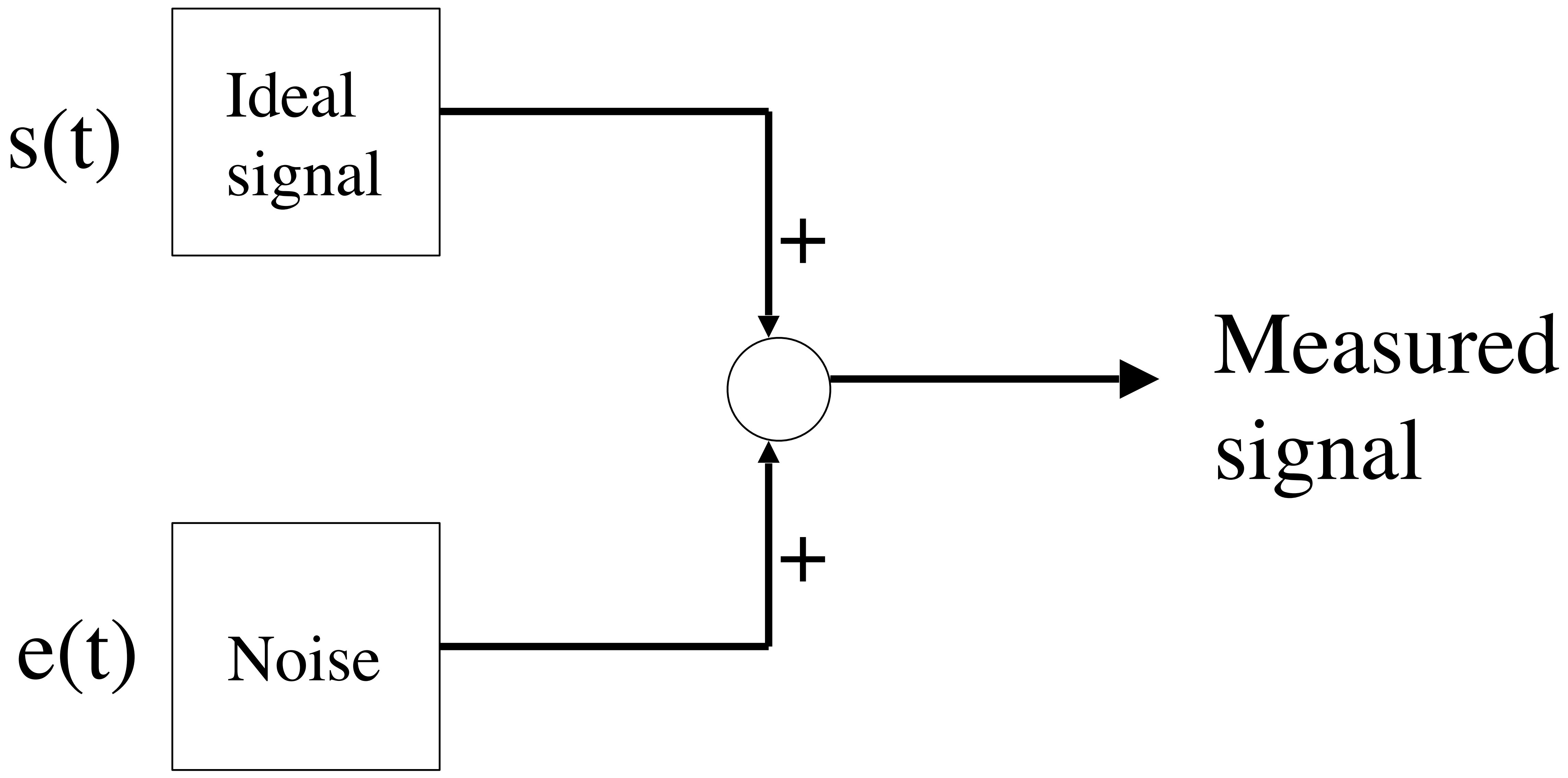
Common filter types in signal processing

- **Low-pass filter** - (ideal) attenuates high frequency data, while allowing low frequency data to pass unchanged
- **High-pass filter** - (ideal) attenuates low frequency data, while allowing high frequency data to pass unchanged
- **Band-pass filter** - (ideal) attenuates all frequencies except a particular frequency band (or bands)
- **Band-stop filter** - (ideal) attenuates one or a selection of frequency ranges of data, allowing all the rest to pass unchanged
- Actual filters are not exactly ideal…which we will discuss

Signals and noise…

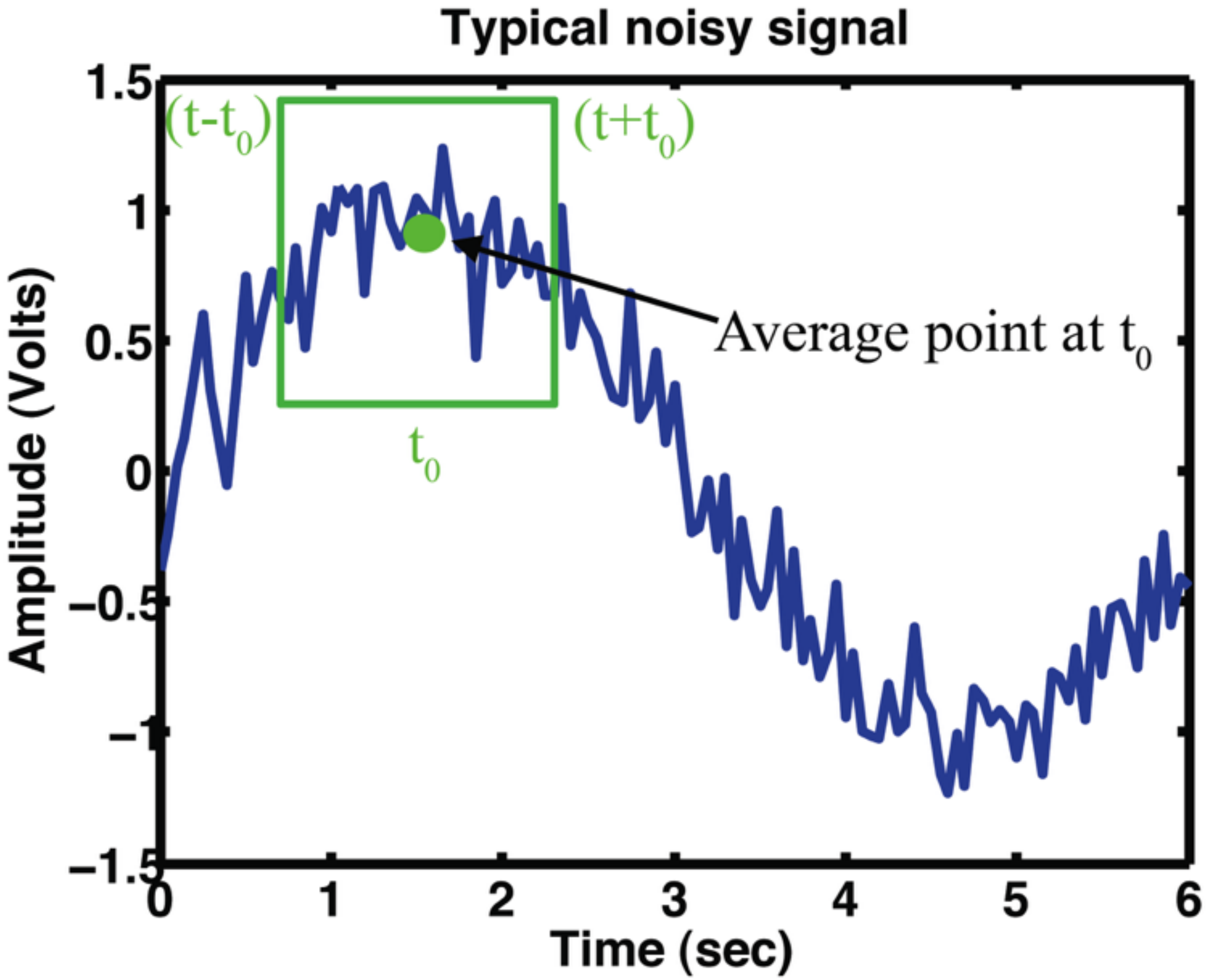
- By making assumptions about the properties of the unwanted 'noise' *e(t)*, we can reconstruct an appropriate *estimate* of the original signal *s(t)*
  - **Noise - any unwanted portion of a signal, lumped together. It may come from multiple sources but tends toward some statistically predictable properties**





Low-pass filtering

- So the effect is this



More on linearity vs. nonlinearity

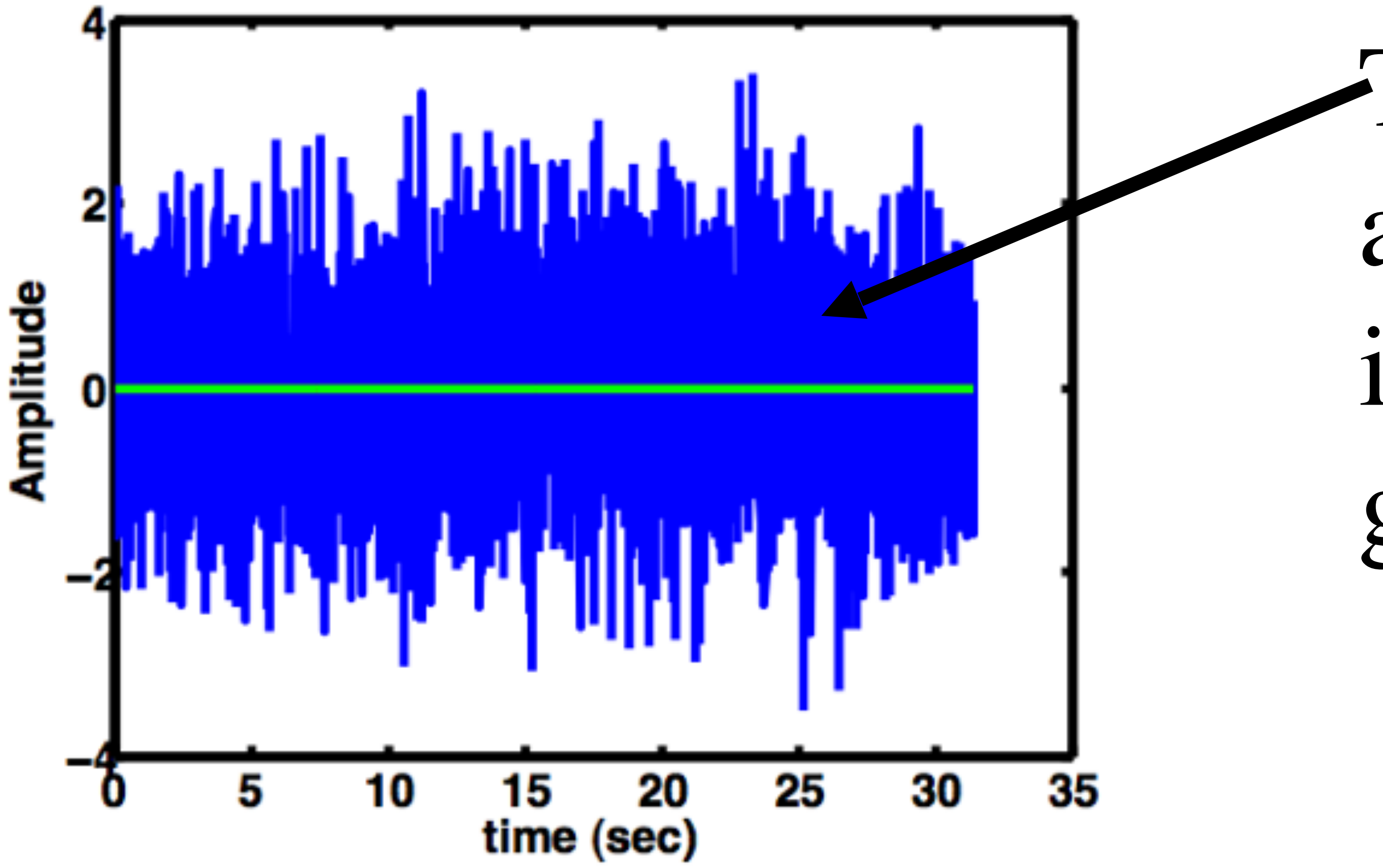
- Power
  - **A linear system is a system whose dependent variables are related to its independent variables by a power of one**
- Linear systems have these particular properties (and they are very favorable)
  - **Additive** $T[x_1(n) + x_2(n)] = T[x_1(n)] + T[x_2(n)]$
  - **Homogeneous** $T[cx(n)] = cT[x(n)]$
- Linear differential equations are more well-understood than nonlinear differential equations

Fourier transforms

- Frequency domain example : Musical note vs. the sound
    - **More parsimonious to describe a song in terms of its notes than time domain signal (when creating a ‘model’ for a song which can be communicated)**

We return to noisy data which we want to 'clean up'

- We do this by removing undesired components of the signal
- One way to do this is *averaging* out the noise
- If it's Gaussian and additive…

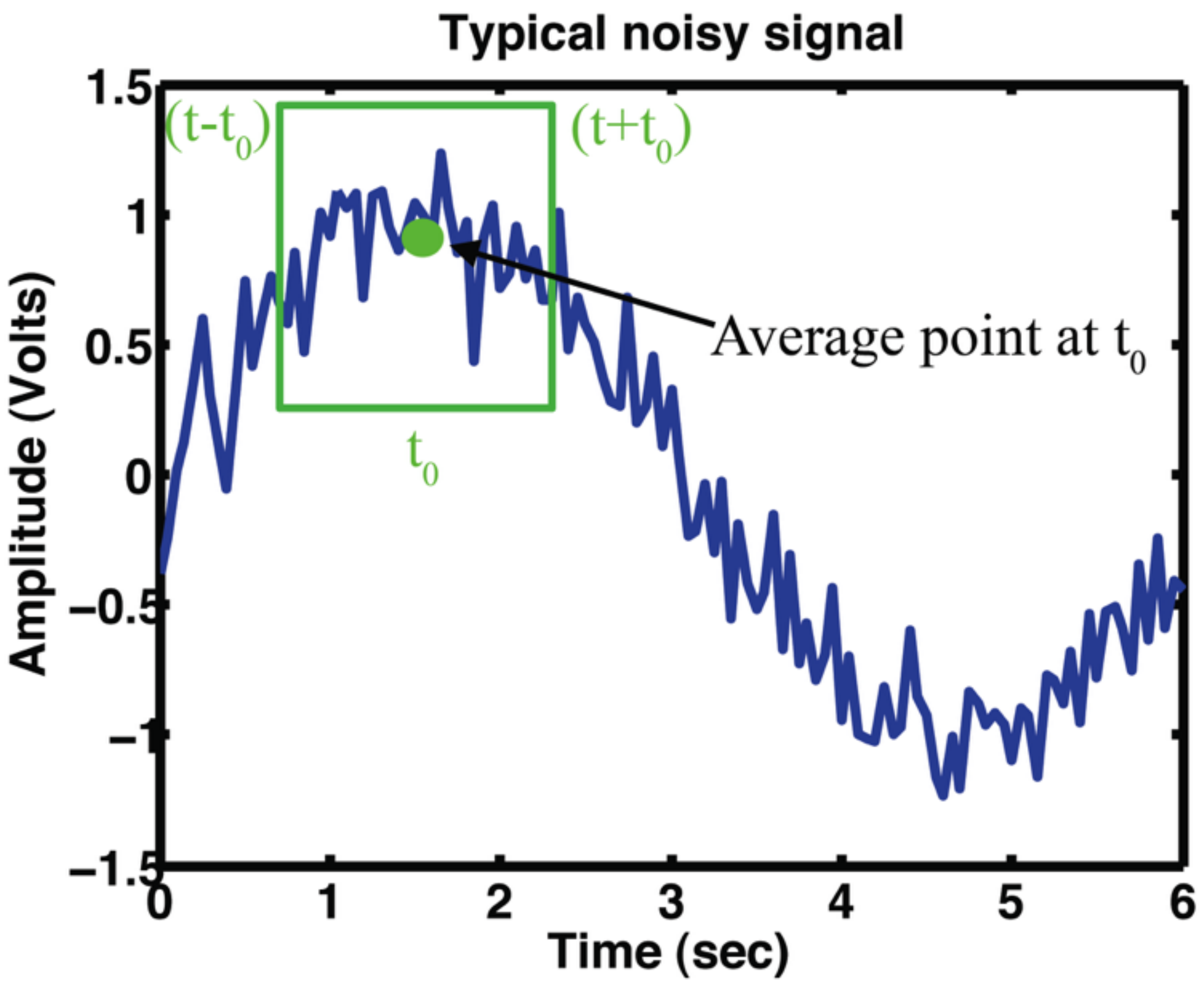


This is gaussian noise, and the average of this is approximately the green line, 0

$-5 + 5 = 0$

How to do it

- Decide on a 'window' of data to average over, which is narrower than the fastest component to your changing signal
- Sum up over that window of points and divide by the number of points (average)



Continuous form

$$x_f(t) = \int_{t-t_0}^{t+t_0} x(\tau)d\tau$$

Discrete form

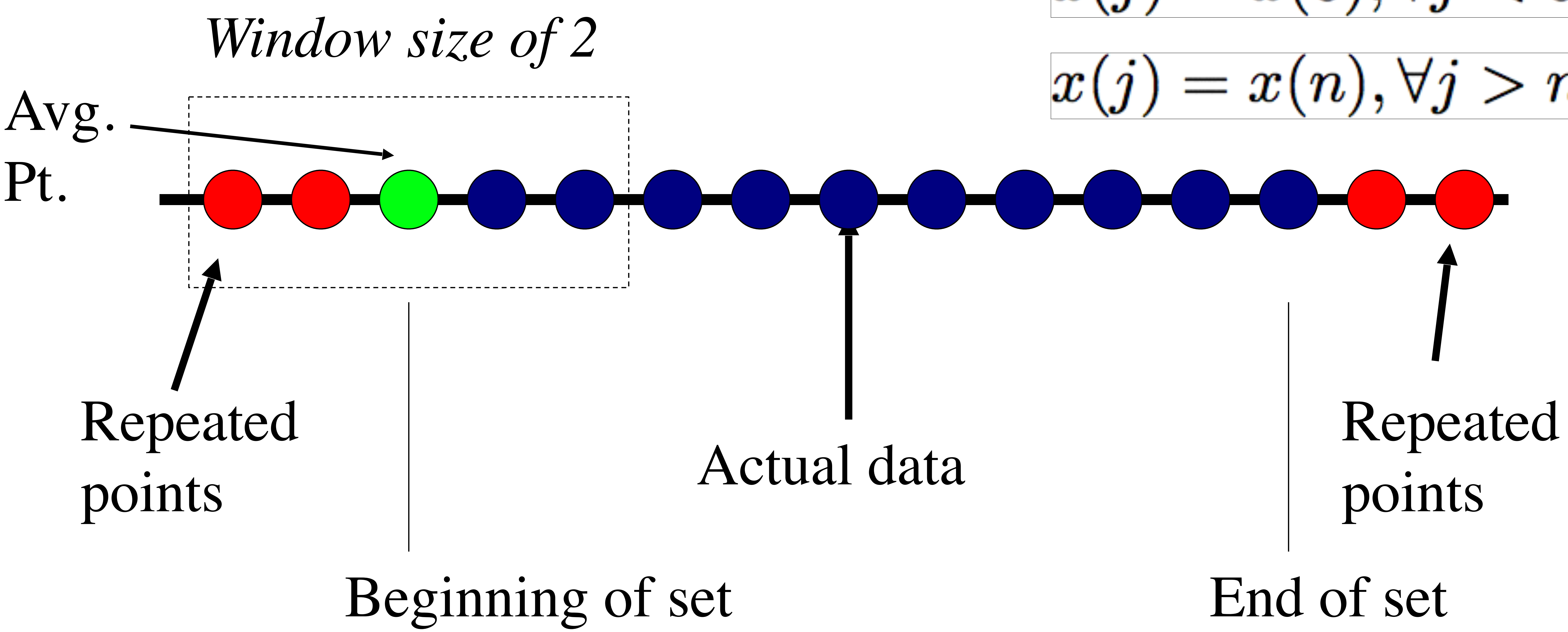
$$x_f(i) = \frac{1}{2k+1}\sum_{j=i-k}^{i+k} x(j)$$

A few details

- What about at the ends of the data where we don't have information before (at the beginning of the data set) or after (at the end of the data set)?
  - **Copy the first or last point and repeat as necessary**

$$x(j) = x(0), \forall j < 0$$

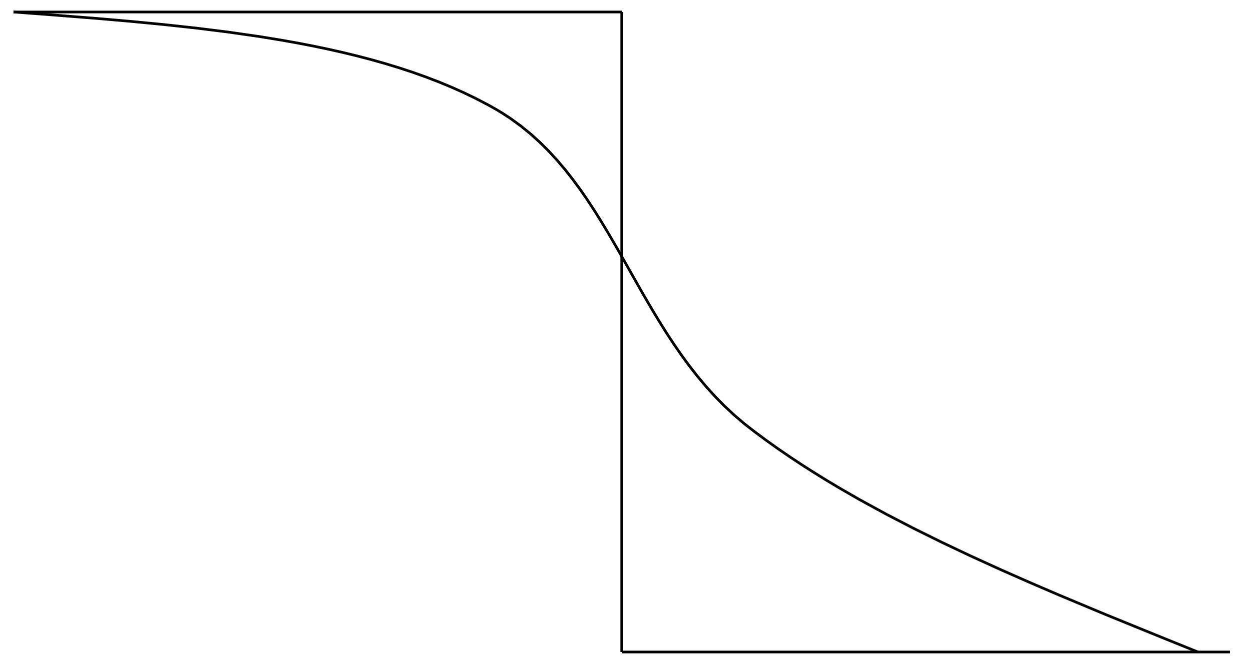
$$x(j) = x(n), \forall j > n$$



Disadvantages…

- Need to have all data in memory already, so it isn't an 'online' filter

- Causality
  - **If we care about an exact event timing, this is a poor filter to use:**



Signal anticipates changes!

Solution

- Recursive filter
  - Solves causality issue
  - Easy to implement as we saw last time