# Lecture 5 Cogsci 109

More sampling, discretization, and Filtering

Mon. Oct. 8, 2007

# Outline for today

- Announcements
- Some programming issues
  - **More on the dot operator, what exactly does it mean?**
  - **Symbolic manipulation, indices and elements of matrices**
  - **Primes and latex**
- Motivating examples
  - **Discretization examples (therapeutic, visual, auditory)**
  - **Aliasing effects examples**
- Filtering theory
  - **Fourier transforms (concept, demos)- Frequency analysis**
    - Low frequencies vs. high frequencies
  - **Filters - removing unnecessary data**
    - LowPass filter - definition, advantages, disadvantages
      - **Moving average**
      - **Recursive filter**
    - HighPass filter - definition, advantages, disadvantages
      - **From low pass to high pass**

# Announcements

- Homework 2 is due
  - **If you are having issues with the assignment, don’t stress out, come talk to me after class and we’ll figure out what to do**
- Homework 3 will be assigned Wednesday

# Some matlab programming issues

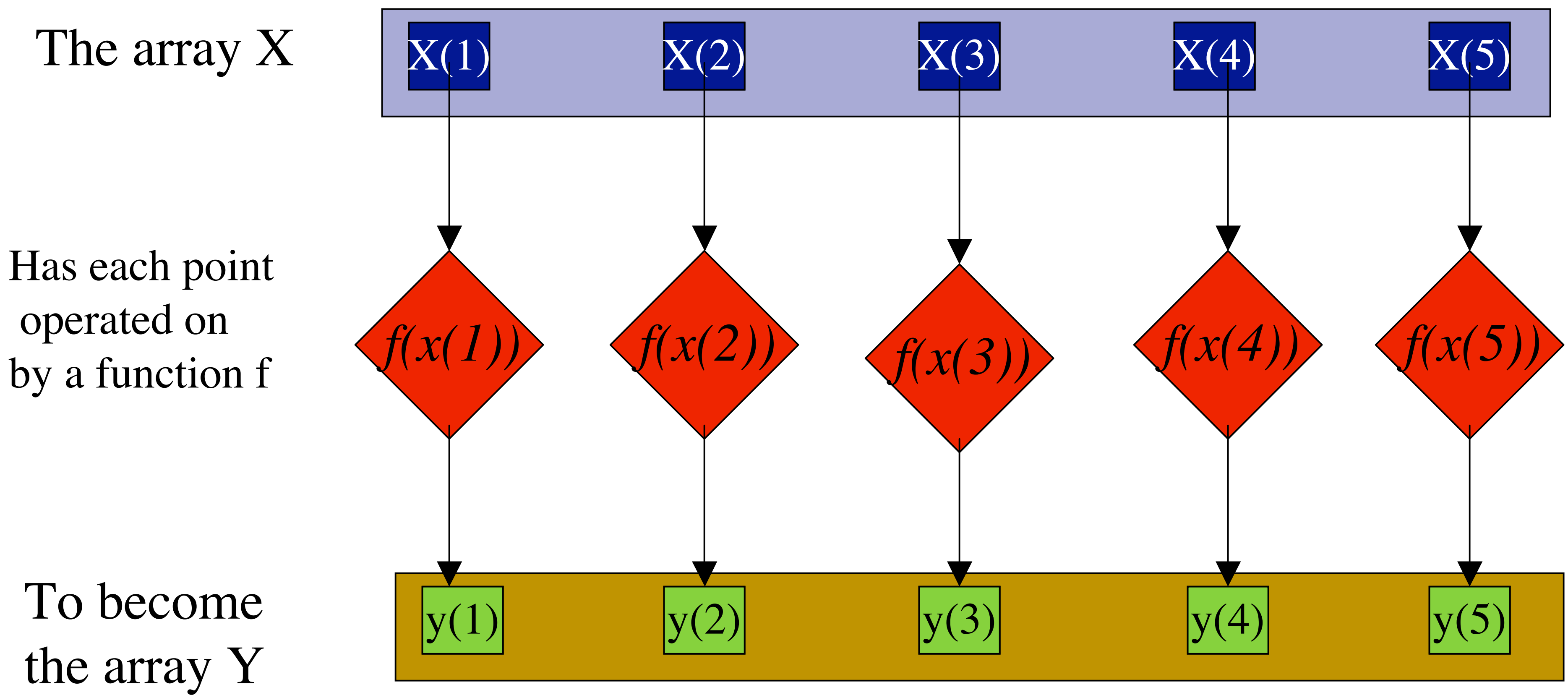
- The dot operator
  - **What exactly does element-wise mean?**
    - Taking *each* value in a variable and performing an operation one element at a time, moving through all the elements in the variable
    - Example:

```
x=[0, 1, 2, 3, 4, 5];
y=x.^2
```

What you see is the final result, but what is happening is:

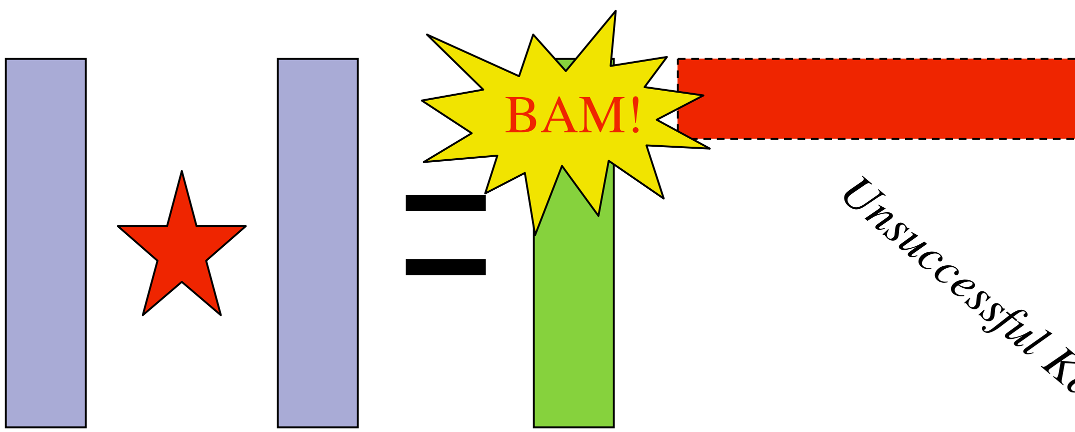
y(1)=0^2, then y(2)=1^2, then y(3)=2^2, then y(4)=3^2, then y(5)=5^2

# Visual concept of the effects of the dot operator



# What happens if you don’t use the dot operator?

- Usually you get an error if you’re trying to perform some calculation like y=x^2, because Matlab tries to do this:




*And it becomes a problem since we’re then trying to multiply an nx1 matrix by an nx1 matrix, and The second n and the first 1 are different sizes!!!*

- Or you get a matrix operation…

# More programming issues in Matlab

- **When is it appropriate NOT to use the dot operator?**
  - When we're wanting to perform matrix operations, such as the matrix A times the vector b, or another matrix of appropriate size

$$A = \begin{bmatrix} 1 & 2 \\ 3 & 1 \end{bmatrix}, b = \begin{bmatrix} 1 \\ 3 \end{bmatrix}$$

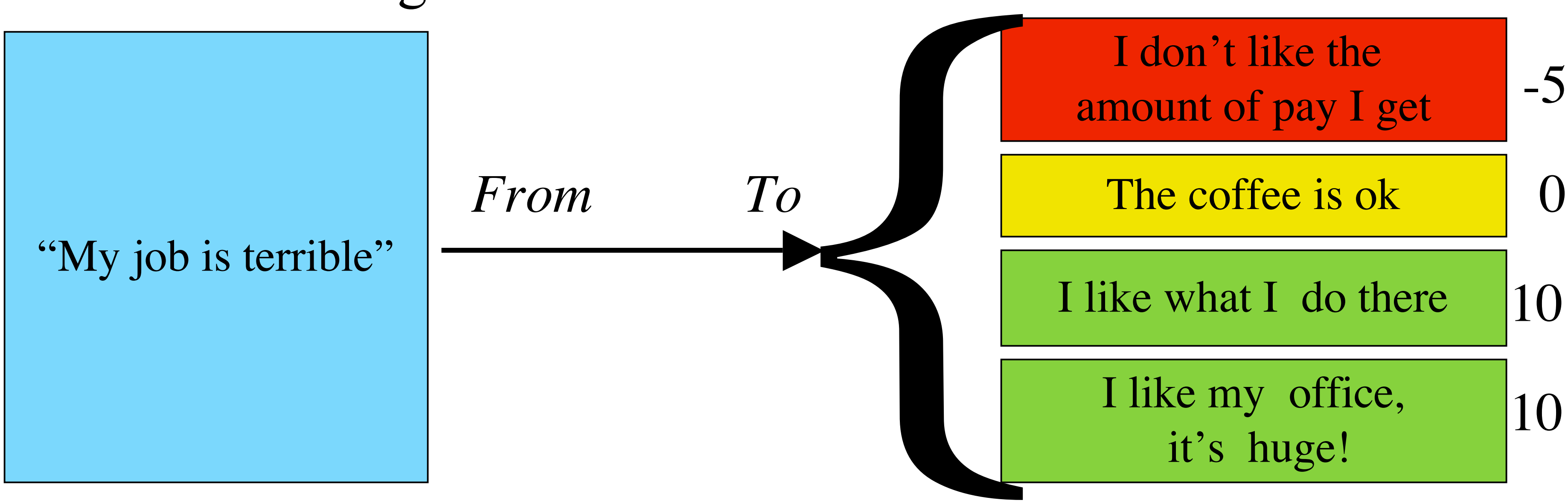
$$A * b = \begin{bmatrix} 1 * 1 + 2 * 3 \\ 3 * 1 + 1 * 3 \end{bmatrix}$$

# Other quick Matlab tips...

- Accessing particular elements in an array or matrix
- Matlab starts its indices at 1, not 0
- Tip: if you want to put primes into the title of a matlab figure, you can use a latex command to make a superscript as follows: x^{|}
- Symbolic manipulation - you can create symbolic variables by using the 'sym' command (type 'help sym' in the matlab command prompt)
  - **Symbolic manipulation can be performed in many ways in matlab (built on maple)**

## Example : Cognitive Therapy - application of discretization strategy to treating depression

- Generalization
- Typically the therapist teaches clients to correctly discretize into separate partitions rather than one continuous generalization



# Examples: Visual discretization

- Color shading

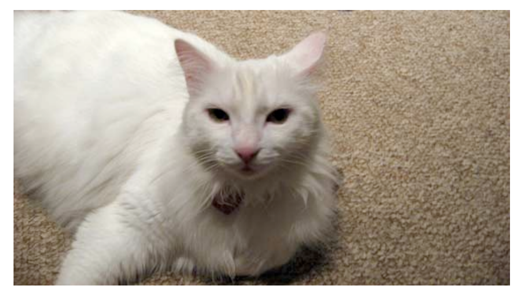


- Color and visual boundaries:

*Few colors and low spatial resolution*

*Low spatial resolution only*

*High spatial resolution and colors*

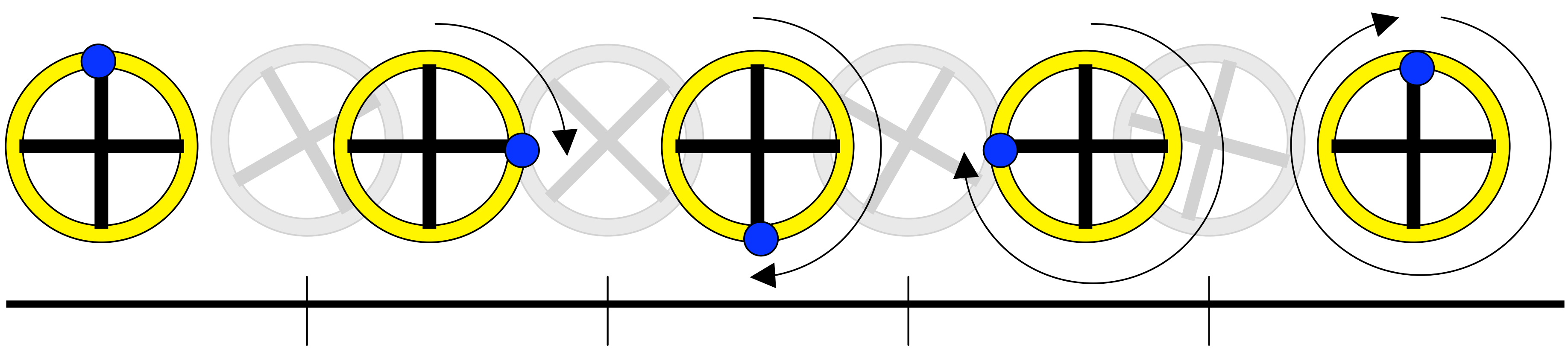


# Auditory examples

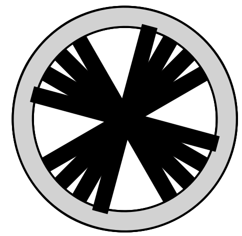
- Sampling rates
  - **Raisin nuts cereal add**

# 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!

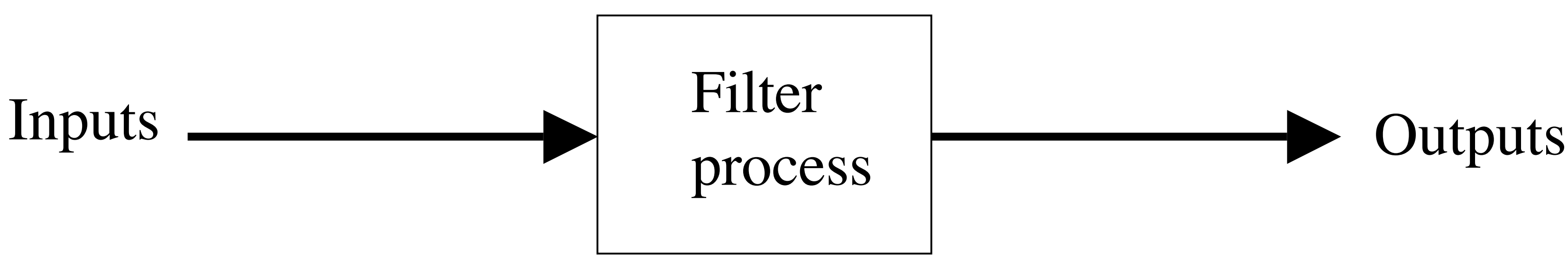


# Obviously aliasing can be bad…

- 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



# Human filtering examples

- **Auditory filtering** *(filtering out unwanted conversations in a crowded room to hear one person)*
- **Conceptual filtering** *(filtering the stream of words and concepts to acquire relevant principles and discard irrelevant ones)*
- **Socio-behavioral filtering** *(filtering the stream of individuals in ones life, removing the undesired individuals while associating with desired individuals - happens by behavioral patterns of living alone, as well as cognitive processes)*

# 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 analysis

- Any time domain signal can be decomposed into a corresponding sum of sine waves
- Sometimes this is an easier way to describe a signal
- Other times this allows us to separate the components we care about from those we don't
- We can compute a frequency-domain representation of a signal by taking the Fourier Transform
  - **Tells us how much energy out of the total energy of the signal is contributed by a particular frequency range**
- Music example

# 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
- 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

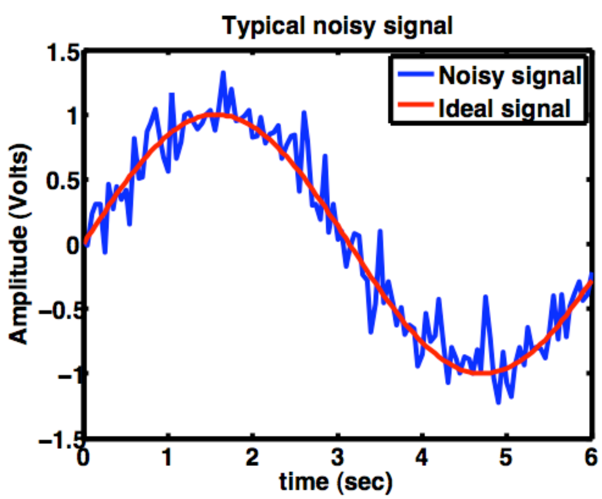
- **Low-pass filter** - (ideal) attenuates high frequency data, while allowing low frequency data to pass unchanged
- **High-pass filter** - (ideal) attenuates low frequency data, while allowing high frequency data to pass unchanged
- **Band-pass filter** - (ideal) attenuates all frequencies except a 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

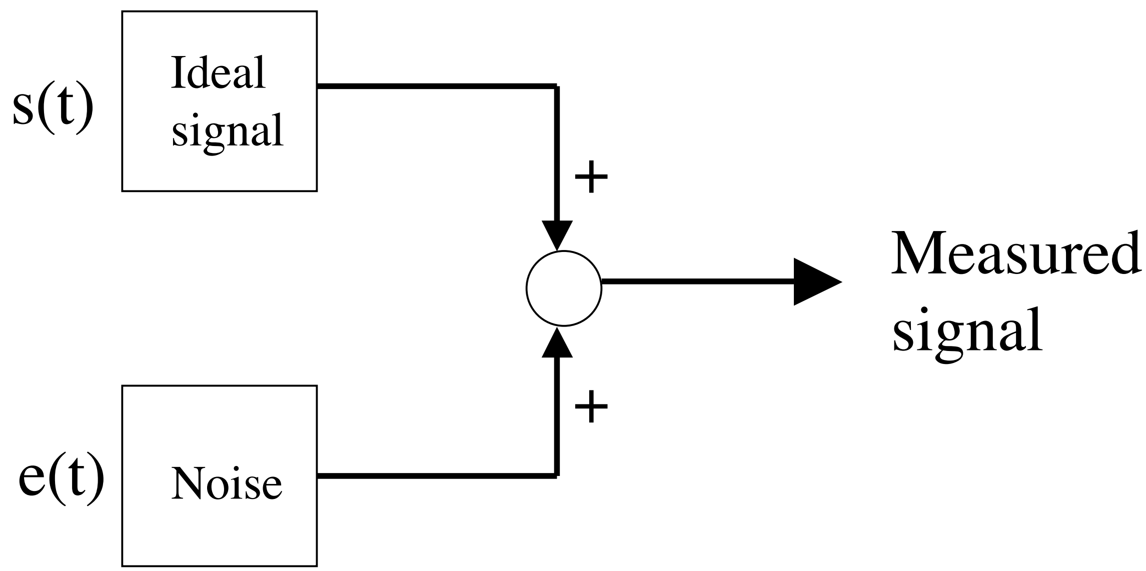
# Filters we'll go through in the next couple of days…

- Low-pass filter
- High-pass filter

# 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***

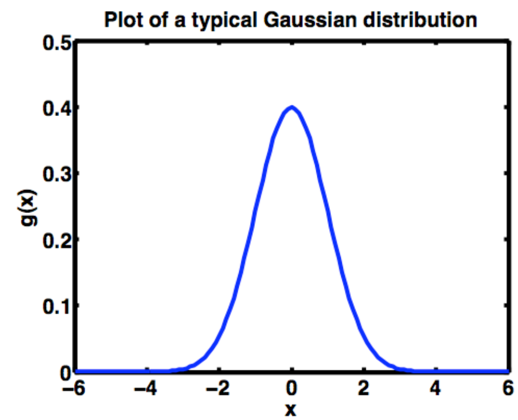


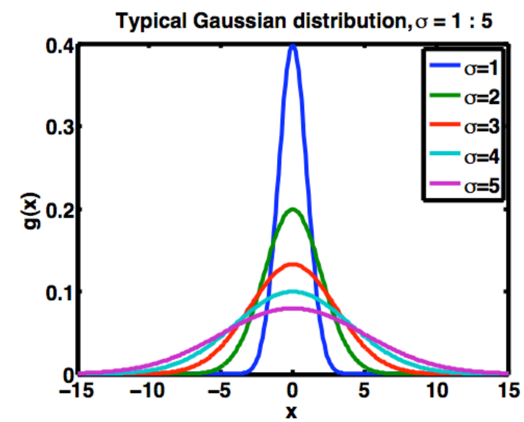
# Gaussian quick review

- Gaussian distributions have particular properties
- A.k.a. The 'Normal curve'

$$g(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\overline{x})^2}{2\sigma^2}}$$




- Has a mean and variance
  - **Typically with noise**
    - Mean (average) = 0
    - Some variance $\sigma^2$



# Low-pass filtering

- If we assume that the high frequency noise we don't care about is *Gaussian*, the noise behaves in a statistically predictable way
  - **Average (or 'mean') = 0**
  - **Therefore one logical method of low pass filtering is by averaging over multiple sample points:**

$$\int_{-\infty}^{+\infty} e(t)dt = 0$$

# Low-pass filtering II

- So the effect is this