Inference from the ground up

Building intuition + a primer on sampling

code of conduct

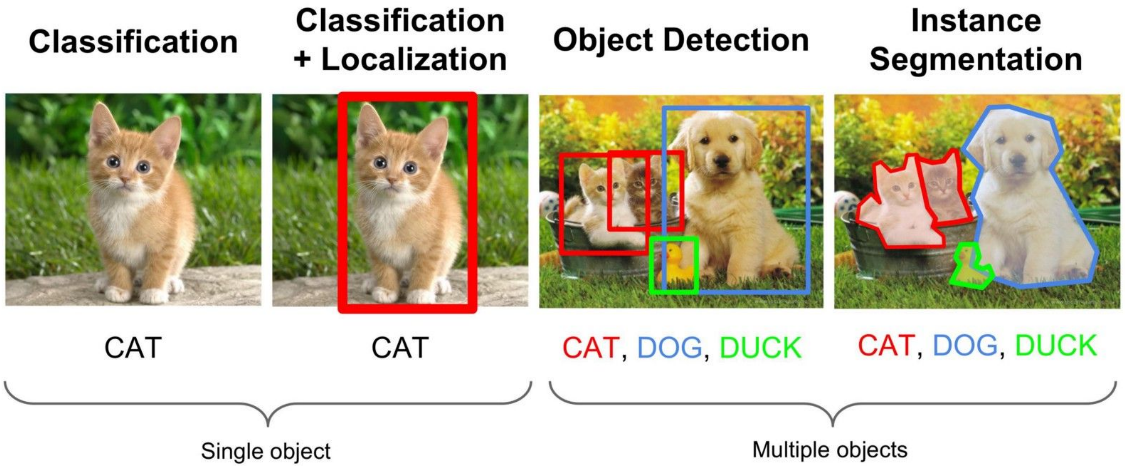
there will be a live coding session:

no laughing at others people’s (my) code

inference

Many ways to infer information from data, e.g., classification

- Is there a categorizable object in an image?



e.g., neural networks, logistic regression

inference

Recommender systems: what will you probably want to watch given what you've already watched?

e.g., clustering++

inference

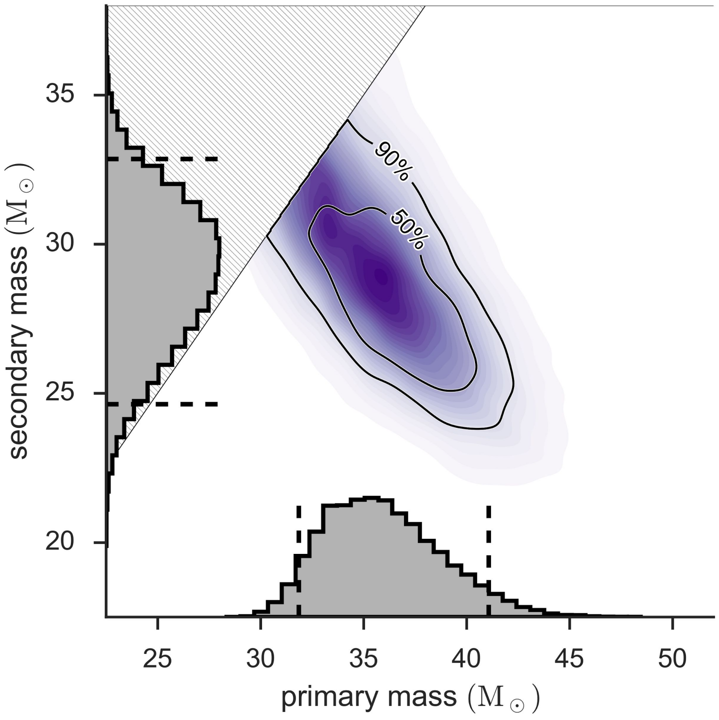
Precision measurement:

does my data contain a signal?

- Many scientific examples (e.g., GWs)

What are the attributes of the signal?

- e.g., GW sky location



Bayesian inference

- Optimal framework for *statistical inference*

Assign probabilities to hypotheses such as:

- Does my data contain a signal?
- How much more likely is it that the data contains a signal vs noise
- If the data contains a signal, does it have such-and-such properties

Bayesian inference

Some features:

- Can be arbitrarily complex, but for many applications there are surprisingly few inputs
  - **Noise statistics**
  - **Theoretical model for signals**

The following example shows how to leverage these two ingredients - in a “Bayesian way” - to demonstrate

- signal detection
- how to extract signal properties

Goals

- How to think about comparing hypotheses (e.g., signal vs noise)
- Build intuition for how to write down a likelihood function
- How to think about defining priors

$$p(\theta|d) = \frac{\mathcal{L}(d|\theta)\pi(\theta)}{Z(d)}$$

- Note how very weak signals can have a high statistical significance if they're extracted optimally

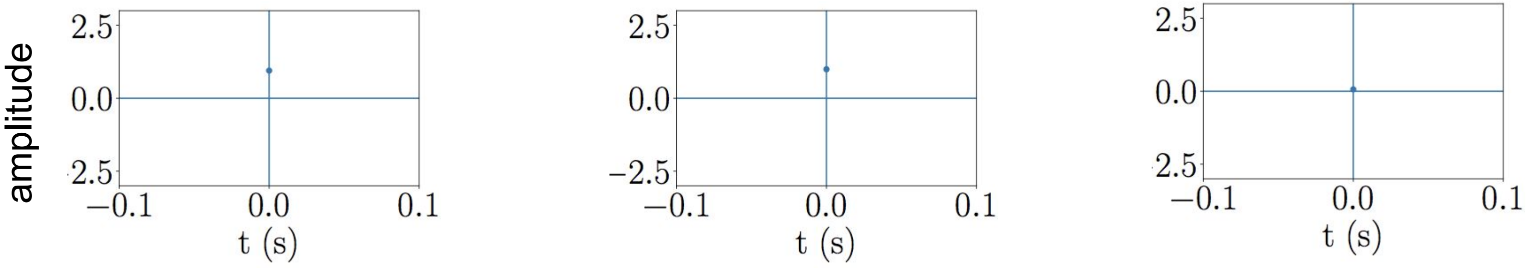
Signal Detection and Parameter Estimation

An example motivated by time-series analysis

Example: detecting impulses

Background/noise measurements

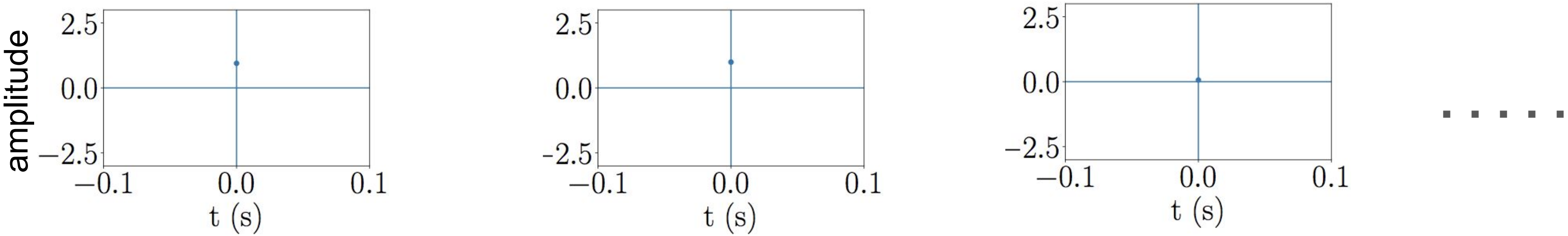
Key point: data are random variables

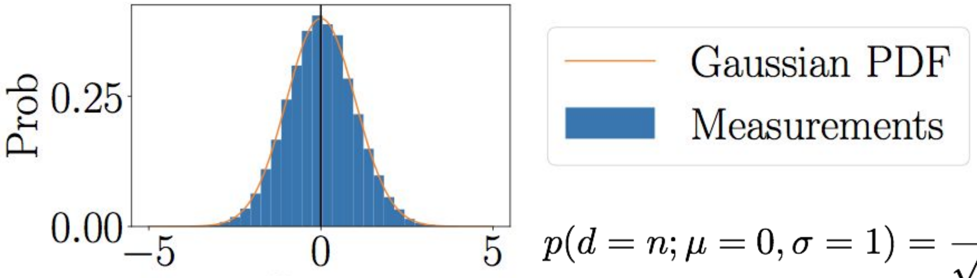


. . . . . .

Example: detecting impulses

Background/noise measurements



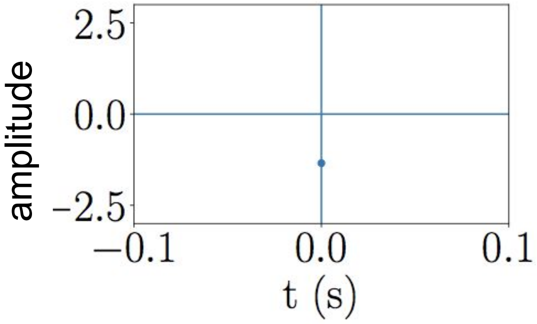


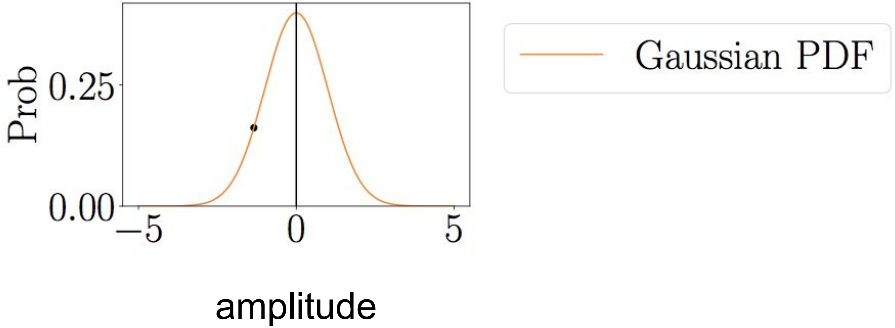
$$p(d = n; \mu = 0, \sigma = 1) = \frac{1}{\sqrt{2\pi 1^2}} \exp\left[-\frac{1}{2}\frac{(n-0)^2}{1^2} \right]$$

amplitude

Example: detecting impulses

Background/noise measurements

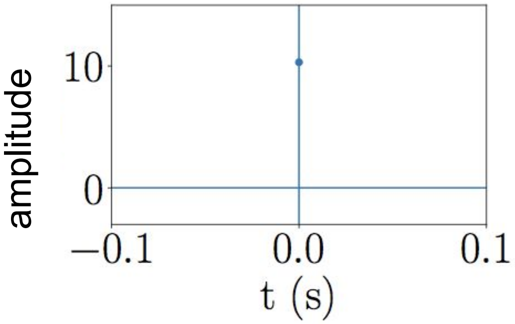


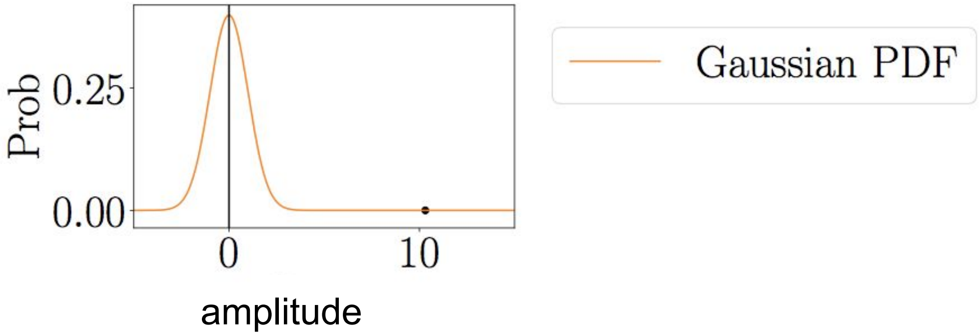


Example: detecting impulses

“Foreground” measurement: impulse of amplitude = 11

Below is the probability that the data is noise



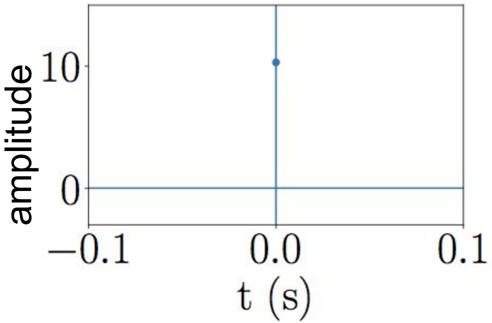


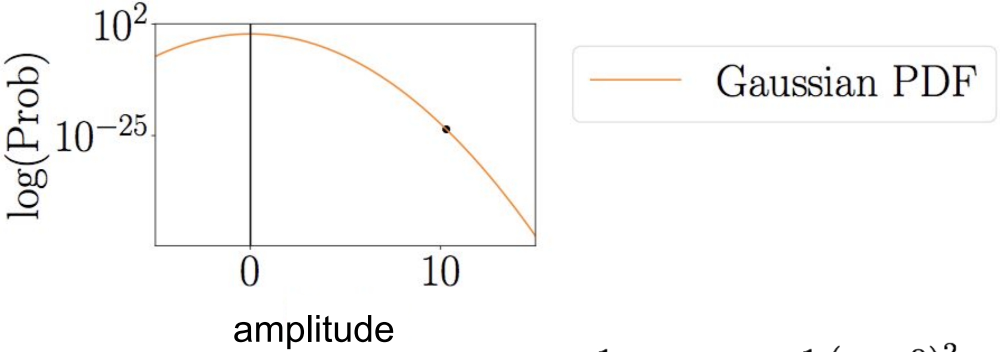
$$p(d = n; \mu = 0, \sigma = 1) = \frac{1}{\sqrt{2\pi 1^2}} \exp\left[-\frac{1}{2}\frac{(n-0)^2}{1^2}\right]$$

Example: detecting impulses

“Foreground” measurement: impulse of strain = 11

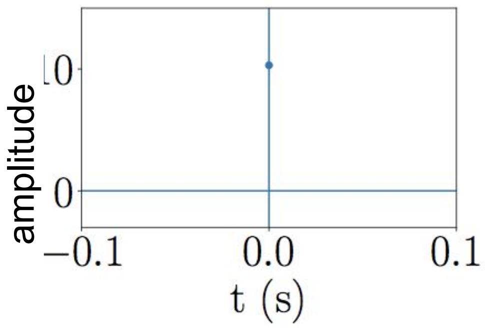
Below is the probability that the data is noise

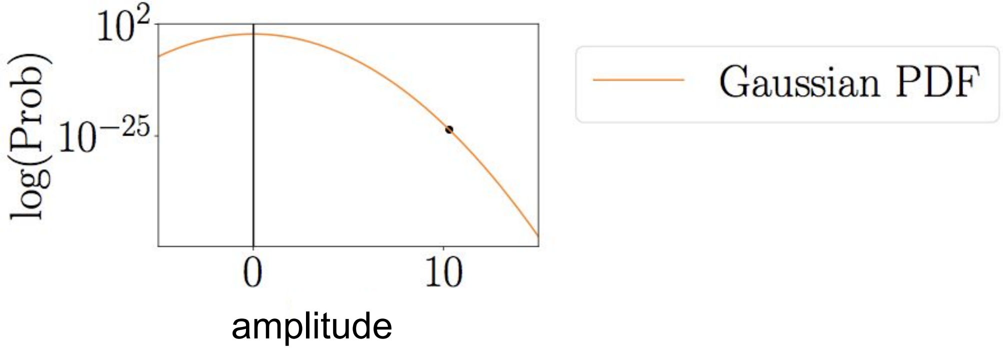




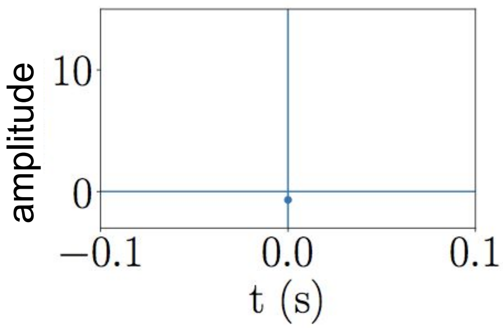
$$p(d=n;\mu=0,\sigma=1)=\frac{1}{\sqrt{2\pi 1^2}}\exp\left[-\frac{1}{2}\frac{(n-0)^2}{1^2}\right]$$

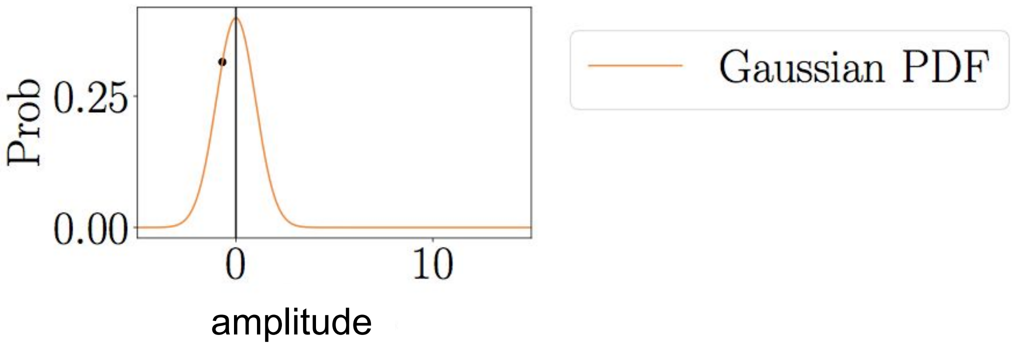
Example: detecting impulses





Subtracting out the signal

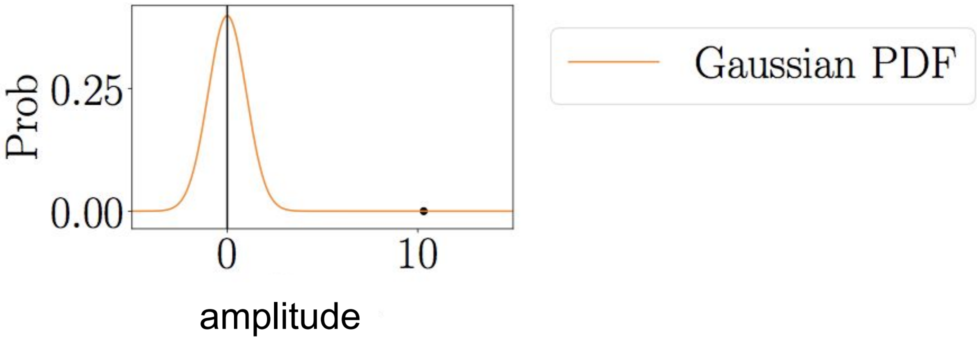




Interlude: Some notation

Likelihood function

$$\mathcal{L}(d|\mathcal{H}_n) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left[-\frac{1}{2}\frac{(d-\mu)^2}{\sigma^2}\right]$$



$$\mathcal{L}(d|\mathcal{H}_s) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left[-\frac{1}{2}\frac{(d-s-\mu)^2}{\sigma^2}\right]$$

Hypothesis testing

Likelihood ratio: detection statistic

- How much more likely is the signal hypothesis than the noise hypothesis?
  - Form the likelihood ratio:

$$O = \frac{\mathcal{L}(d|\mathcal{H}_s)}{\mathcal{L}(d|\mathcal{H}_n)}$$

Hypothesis testing

Likelihood ratio: detection statistic

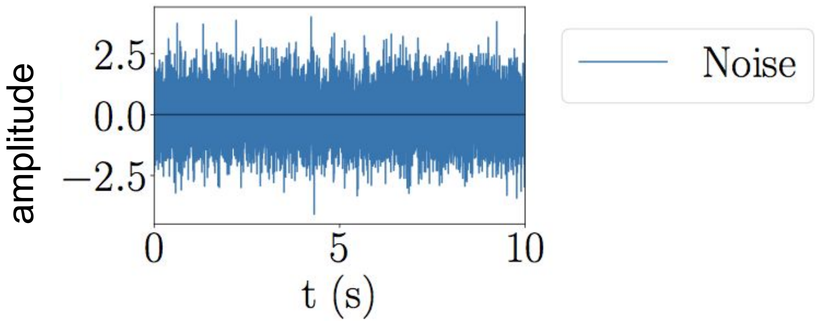
- How much more likely is the signal hypothesis than the noise hypothesis?
  - Form the likelihood ratio:

$$O = \frac{\mathcal{L}(d|\mathcal{H}_s)}{\mathcal{L}(d|\mathcal{H}_n)}$$

In this case: O = exp(23)

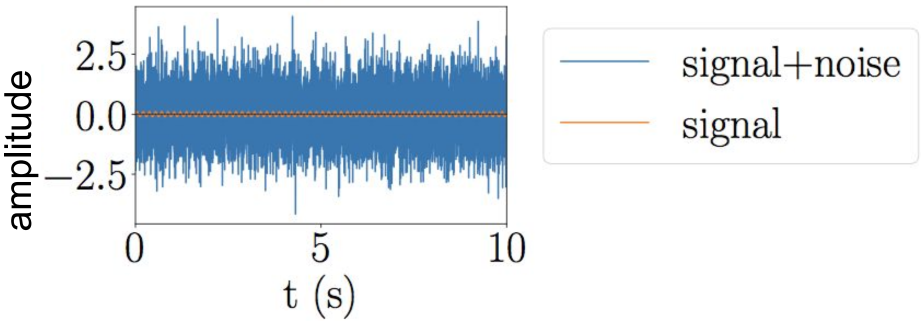
A more interesting example: a quiet sine wave

Time series data



Gaussian noise at each point in time

- mean = 0
- var = 1



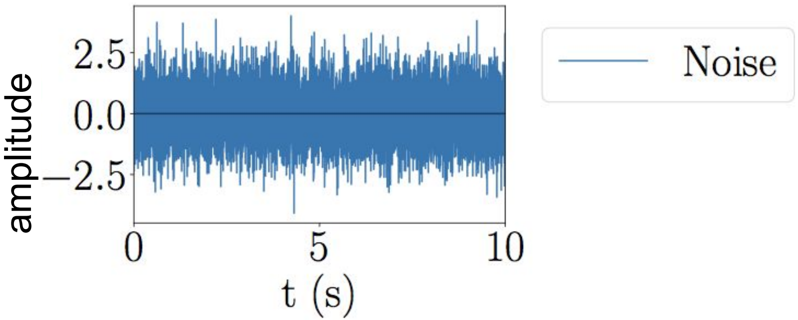
Gaussian noise + sine wave

- f = 6 Hz
- A = 0.1 (var/10)

A more interesting example: a quiet sine wave

Time series data

- Using N=10’000 sample points

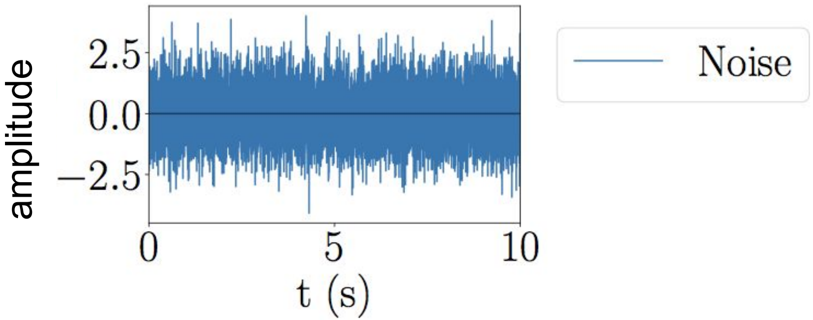


$$\mathcal{L}(d(t_1), d(t_2)...d(t_N)|\mathcal{H}_n) = \mathcal{L}(d(t_1)|\mathcal{H}_n) \text{ and } \mathcal{L}(d(t_2)|\mathcal{H}_n) \text{ and } \ldots \text{ and } \mathcal{L}(d(t_N)|\mathcal{H}_n)$$

$$\mathcal{L}(d|\mathcal{H}_n) = \prod_{i=1}^{N} \frac{1}{\sqrt{2\pi\, 1^2}} \exp\left[-\frac{1}{2}\frac{(d-0)_i^2}{1^2}\right]$$

A more interesting example: a quiet sine wave

Time series data

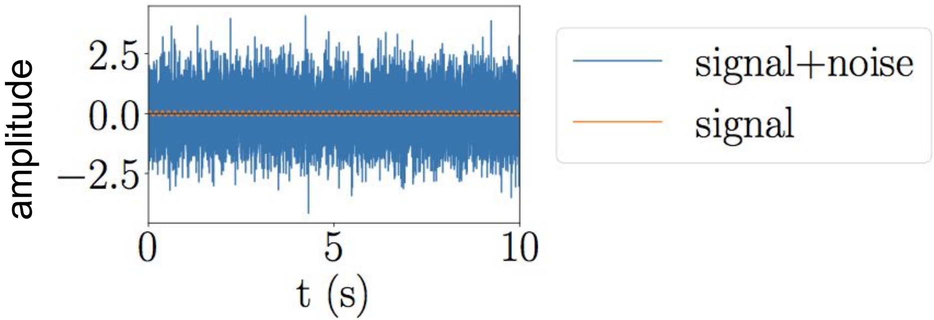


$$\mathcal{L}(d|\mathcal{H}_n) = \prod_{i=1}^{N} \frac{1}{\sqrt{2\pi\, 1^2}} \exp\left[-\frac{1}{2}\frac{(d-0)_i^2}{1^2}\right]$$

- Using N=10’000 sample points
  - log[L(d)] = -5077 (!?)
  - This is a consequence of high-D data
  - Probability ***density*** in shrinks like power of the dimension

A more interesting example: a quiet sine wave

Time series data



$$\mathcal{L}(d|\mathcal{H}_n) = \prod_{i=1}^{N} \frac{1}{\sqrt{2\pi\, 1^2}} \exp\left[-\frac{1}{2} \frac{(d-0)_i^2}{1^2} \right]$$

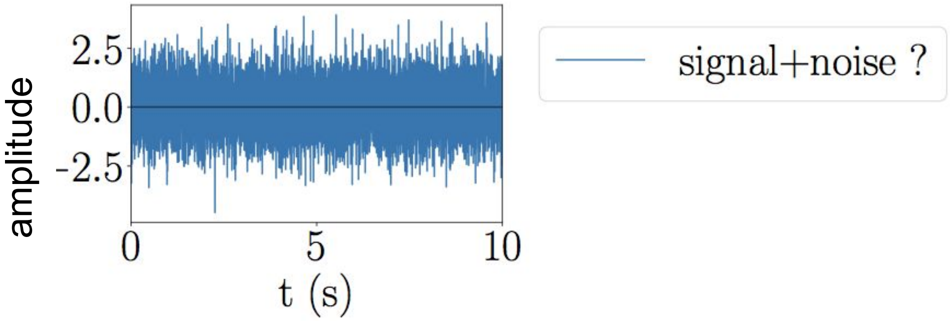
$$\mathcal{L}(d|\mathcal{H}_s) = \prod_{i=1}^{N} \frac{1}{\sqrt{2\pi\, 1^2}} \exp\left[-\frac{1}{2} \frac{(d-s-0)_i^2}{1^2} \right]$$

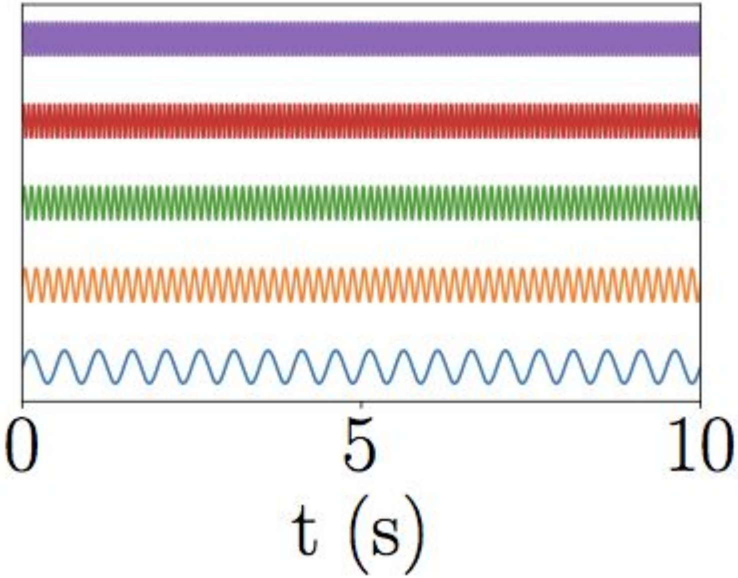
- Using N=10’000 sample points
  - log[L(data=noise)] = -5077
  - log[L(data-signal=noise)] = -5047
  - log[L(signal)/L(noise)] = 30
- exp(30) ~ 10^{13} times more likely to be a signal than noise!

The basics of *search*

What if the exact signal is unknown?

What if we want to *find* a signal





The basics of *parameter estimation*

$$p(f|d) \propto \mathcal{L}(d|f, \mathcal{H}_s)\pi(f)$$

- The probability of the signal's frequency depends on the likelihood *and* the prior
  - Choosing a good prior is motivated by, e.g.,
    - Knowledge of how signal's frequencies are distributed in nature (previous measurements)
    - **Ignorance (Maybe we don't know what to expect, so we should treat all values as equally likely)**
    - **Known constraints (signals only have frequencies in some range)**

The basics of *parameter estimation*

$$p(f|d) \propto \mathcal{L}(d|f, \mathcal{H}_s)\pi(f)$$

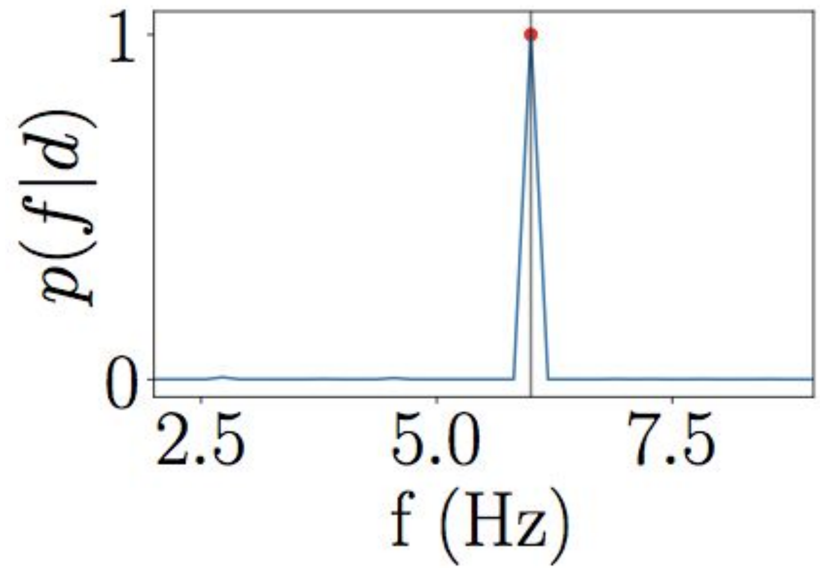
$$\mathcal{L}(d|f, \mathcal{H}_s) = \prod_{i=1}^{N} \frac{1}{\sqrt{2\pi 1^2}} \exp\Big[-\frac{1}{2}\frac{(d_i - \sin(2\pi f t_i) - 0)^2}{1^2}\Big]$$

$$\begin{aligned} \pi(f) &= 1 && (2\mathrm{Hz} \leq f \leq 8\mathrm{Hz}) \\ &= 0 && \text{otherwise} \end{aligned}$$

Exercise:

- Code up (log) likelihood functions for the signal and noise hypothesis
    - The function should take as inputs:
        - Data
        - Noise variance
        - Noise mean
        - A signal
- Useful tips:
    - When multiplying numbers, it’s a good idea to use numpy functions like numpy.dot as this will ensure you code will scale efficiently when your data becomes large
- Add a signal to some noise and compute the likelihood ratio

The basics of *parameter estimation*

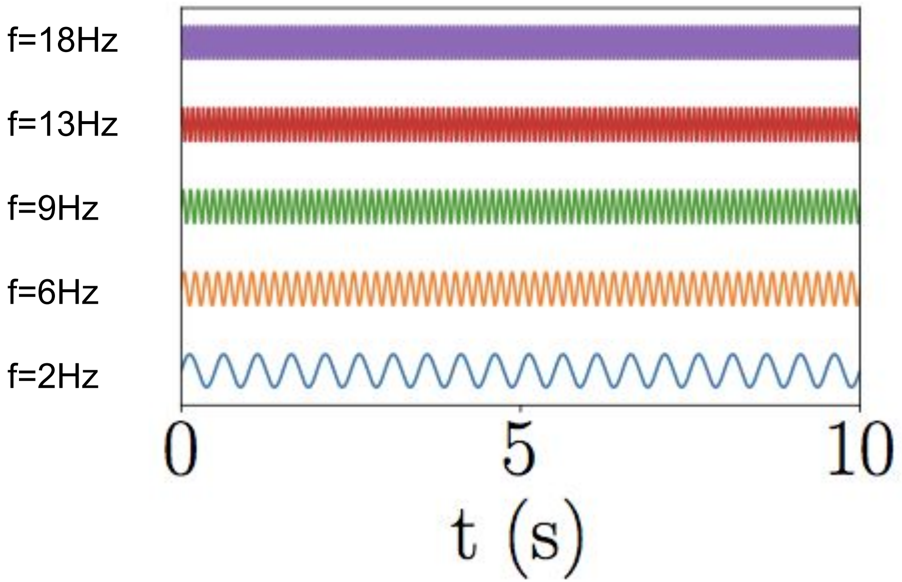


- ρ=30 (as we found when we knew the exact signal)
-

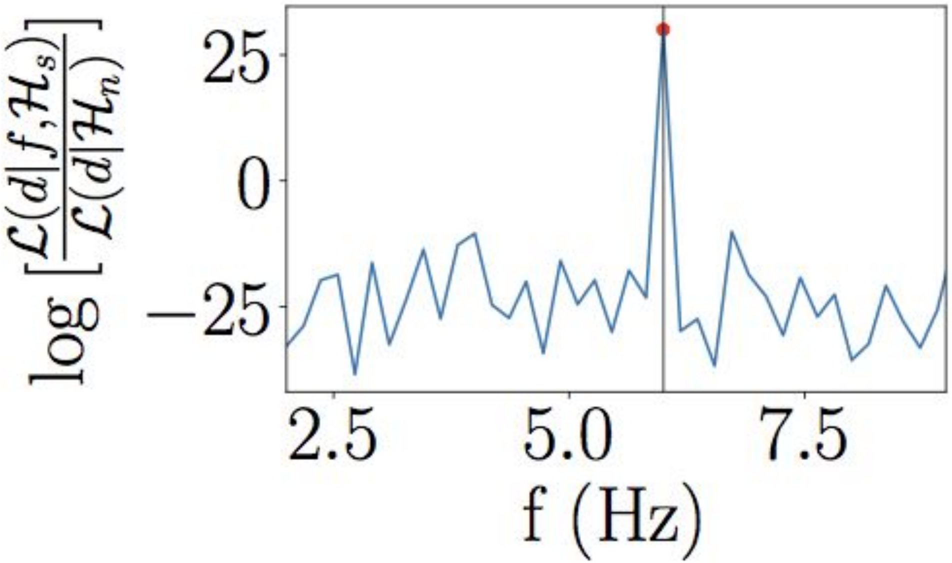
unkown signals

What if the exact signal is unknown?

What if we want to *find* a signal



The basics of *search*



- ρ=30 (as we found when we knew the exact signal)
- This is also the basis of parameter estimation!

Exercise: Recreating the experiment in software

- Use numpy to generate some Gaussian random variables
  - Hint: the numpy.random module contains a convenient function to do this

- Create a histogram of your random numbers
  - Note how the histogram converges as you generate more numbers
  - Compare your (normalized) histogram to the exact probability of a number being Gaussian

A primer on sampling

- Previous example used a simple search over a single parameter (frequency)
  - Numerically simple
  - Scales poorly with number of parameters
- Example:
  - Measure a satellite's trajectory for a few seconds/minutes/hours using GPS
    - What is the (initial) position, velocity, acceleration of the satellite

Position (3) + velocity (3) + acceleration (3) = 9 parameters

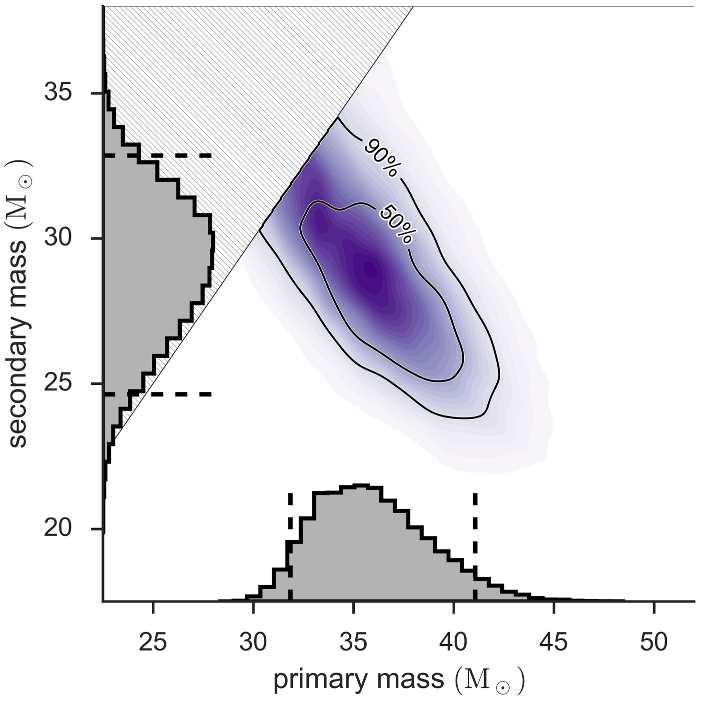
- Gridding up N points in each dimension and searching over this space scales like N^9 !

A primer on sampling

- Do we need to consider N^9 trails point to get an accurate measurement of the variables?
  - Noise in the data sets ultimate accuracy of any measurement
  - Systematic error scales like 1/sqrt(M) where M is the number of measurements

- Probably only need a few thousand "sample" points in position/velocity/acceleration to get a measurement that's not dominated by systematic errors
- *Sampling* provides a way to do this
  - Idea is that we only need a relatively small number of samples for the parameters where most of the probability is contained (and ignore the rest)
  - E.g., sampling a population to measure height distribution

A primer on sampling



A primer on sampling

Drawing points with the right probability requires specialized algorithms/tools:

- MCMC
- Nested sampling

The sample in a random, unbiased way