Office hours

- 10:30-11:30 on Tuesday OK?

LLS Recap

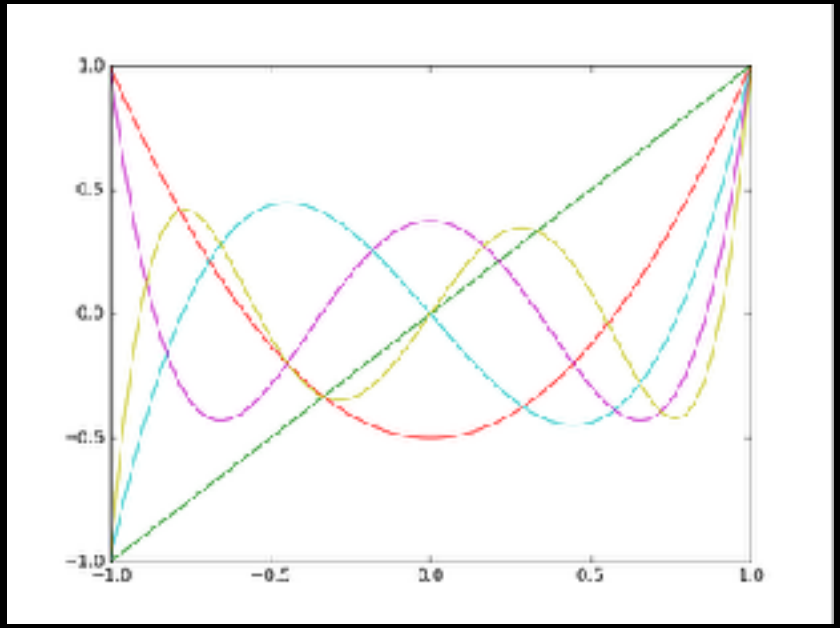
- Assume that $\langle d \rangle = Am$. Then maximum likelihood solution to minimizing χ^2 is $A^T N^{-1} A m = A^T N^{-1} d$
- We also saw that the (co)variance of the errors on m is given by $(A^T N^{-1} A)^{-1}$.

Stability

- Let's fit polynomials (polyfit.py). How did that go? Why?
- Let's ignore N for now, and use SVD of A - $A=USV^T$, where U is orthogonal (and rectangular), S is diagonal, and V is orthogonal (and square).
- $ATA = VSU^TUSV^T=VS^2V^T$. $(ATA)^{-1}=VS^{-2}V^T$, so if an entry of S was very small, it becomes very large.
- By writing out an analytic cancellation, we can get rid of one copy of S, making problem better behaved numerically: $VS^2V^Tm=VSU^Td$. V and S are square and invertible (usually!), so leaves us with $SV^Tm=U^Td$. No squaring of S… or, $m=VS^{-1}U^Td$.
- For polys, real solution is to switch bases to e.g. Legendre, Chebyshev… Good idea to check condition number (ratio of largest to smallest entries of S) before trying LLSQs.

Legendre Polynomials

- Regular polynomials have recurrence relation: $F_{n+1}=xF_n$, with $F_0=1$.
- Legendre polynomials come from different recurrence relation: $(n+1)P_{n+1}=(2n+1)xP_n-nP_{n-1}$.
- They have the nice property that $\int P_nP_m=2/(2n+1)\delta_{mn}$ and are bounded by ± 1.
- If we ignore N, what is the (approximate) condition number of A^TA?



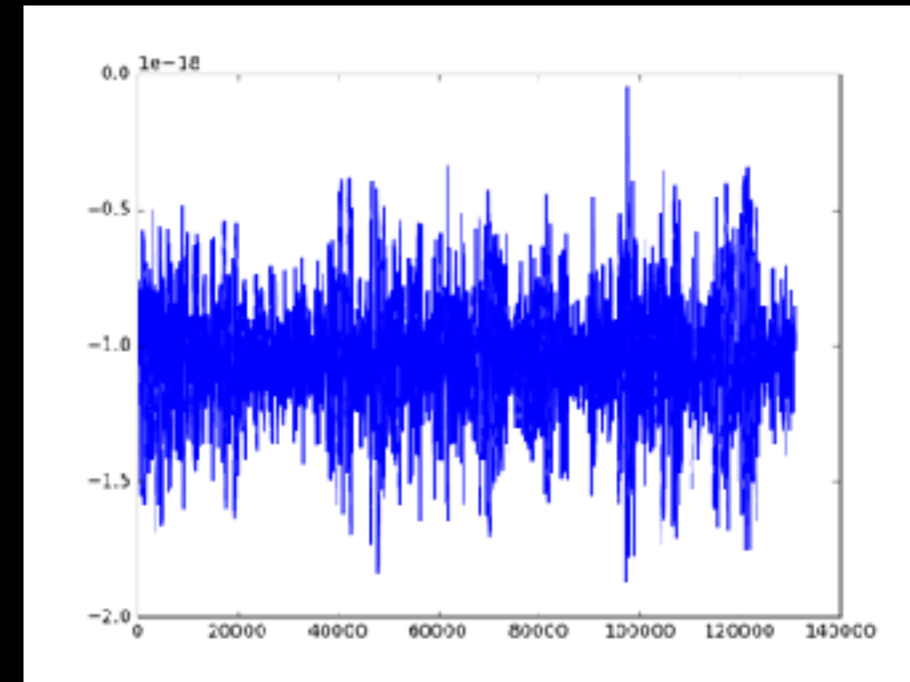
Worked Example

- What is the best-fit mean and error for a set of uncorrelated gaussian variables with same mean but individual errors?

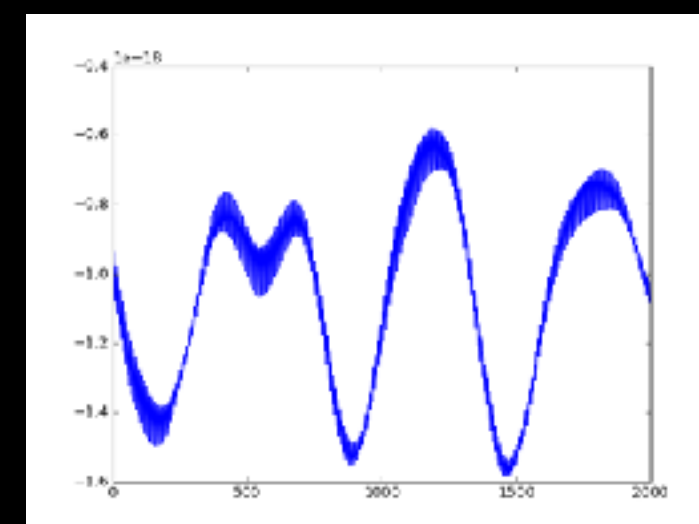
- $A=?$ Show that $A^TN^{-1}A=\sum(\sigma_i^{-2})$, $A^TN^{-1}d=\sum d_i/\sigma_i^2$.

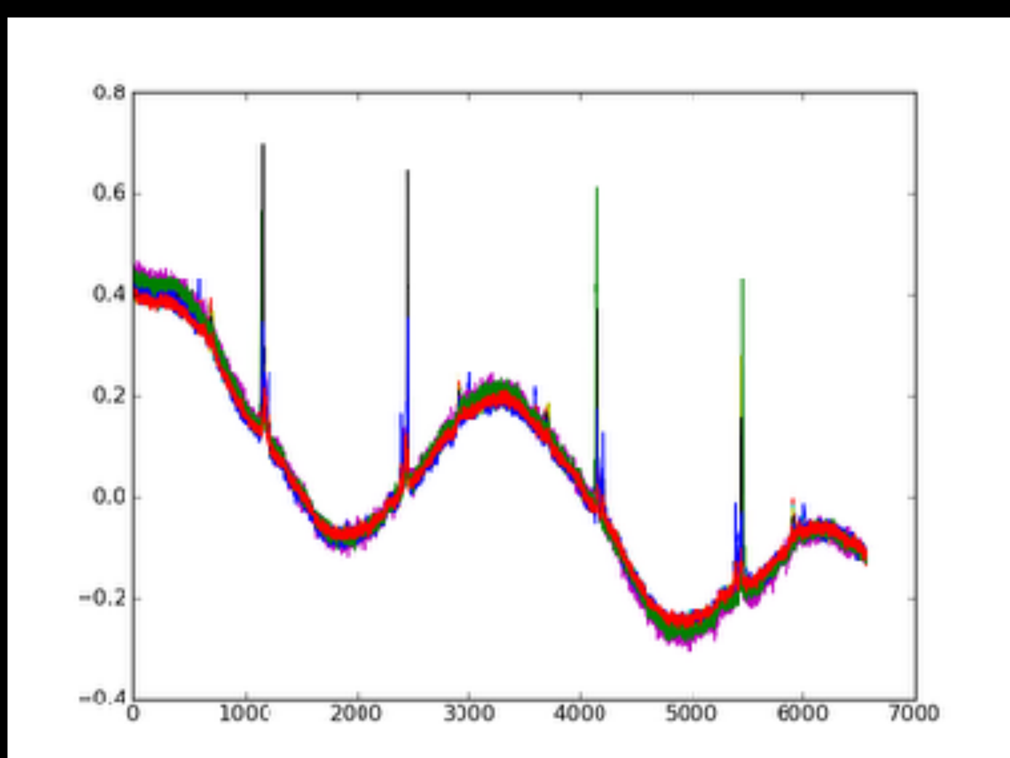
- Define weights $w_i=\sigma_i^{-2}$. Then $m=\sum w_i d_i/\sum w_i$. Variance of our estimator is $1/\sum w_i$.

Correlated Noise

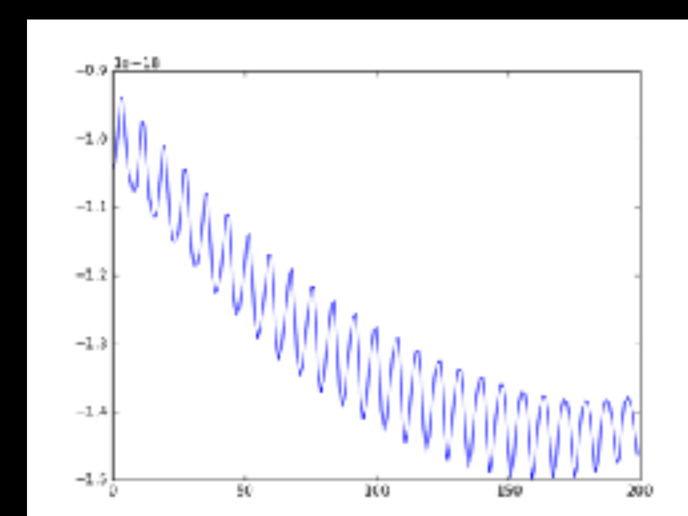


- So far, we have assumed that the noise is independent between data sets.
- Life is sometimes that kind, but very often not. We need tools to deal with this.





Right:LIGO data, with varying levels of zoom.
Left: detector timestreams from Mustang 2 camera @GBT



Fortunately…

- Linear algebra expressions for χ^2 already can handle this.
- Let V be an orthogonal matrix, so $VV^T=V^TV=I$, and $d-Am=r$ (for residual)
- $\chi 2=r^TN^{-1}r=r^TV^TVN^{-1}V^TVr$. Let $r\to Vr$, $N\to VNV^T$, and $\chi 2$ expression is unchanged in new, rotated space.
- Furthermore, (fairly) easy to show that $\langle N_{ij}\rangle=\langle r_ir_j\rangle$.
- So, we can work in this new, rotated space without ever referring to original coordinates. Just need to calculate noise covariances N_{ij}.

Stationary Noise

- An important class of noise is stationary - i.e. no *systematic* change with time.
- In particular, $N_{ij}=\langle n_i n_j \rangle = f(i-j)$ only. This is often close to reality!
- For this case, if we take the Fourier transform of the data, then $\langle F_k F_{k'} \rangle = \sigma_k^2 \delta_{kk'}$ i.e. the noise is uncorrelated in Fourier space.

LLS With Stationary Noise

- The Fourier transform is just an orthogonal matrix, up to (possible) normalization constant.
- Usual discrete FT has $F^T F = nI$ where n is # of data points.
- So, $\chi 2 = r^T F^T N^{-1} F r$ where N is once again diagonal.

χ^2 Example 2

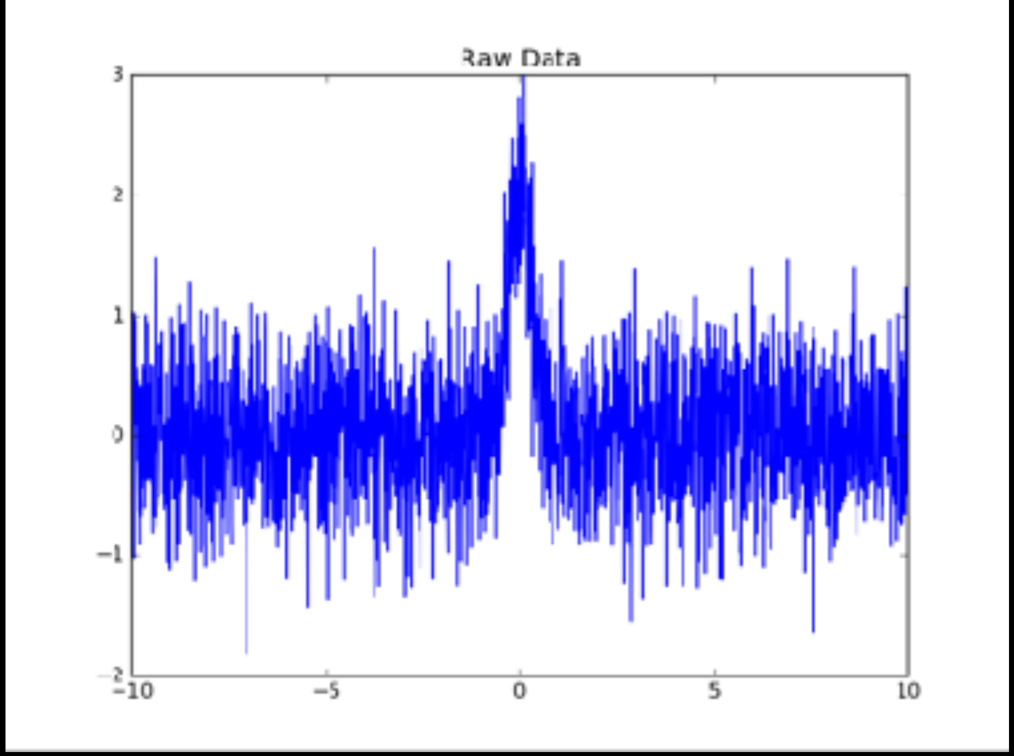
- Let's fit a 1-parameter template to data, but possibly want to shift it (e.g. fitting a for a source amplitude at various positions).
- If A is n by 1, then $A^TN^{-1}A$ is a scalar.
- now we have $m=A^TN^{-1}d/(A^TN^{-1}A)$.
- If N is stationary (i.e. $N_{ij}=f(i-j)$) and we shift the template $A_i \rightarrow A_{i+\delta}$, how does the denominator depend on δ?
- Up to an overall constant, we can make $N^{-1}d$ and dot it against the various shifted A's.

Example 2 ctd.

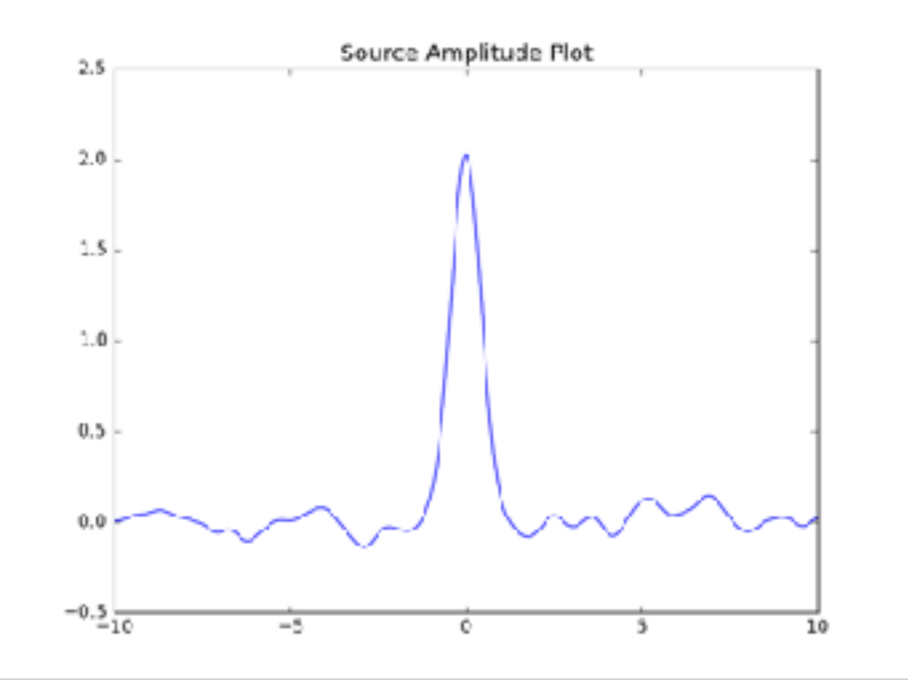
- What if we wanted to plot amplitude in terms of SNR?
- recall $Var(m)=(A^TN^{-1}A)^{-1}$, so $\sigma(m)=(A^TN^{-1}A)^{-1/2}$.
- So, $SNR=signal/noise=A^TN^{-1}d/A^TN^{-1}A/(A^TN^{-1}A)^{-1/2}=A^TN^{-1}d/\sqrt{(A^TN^{-1}A)}$.

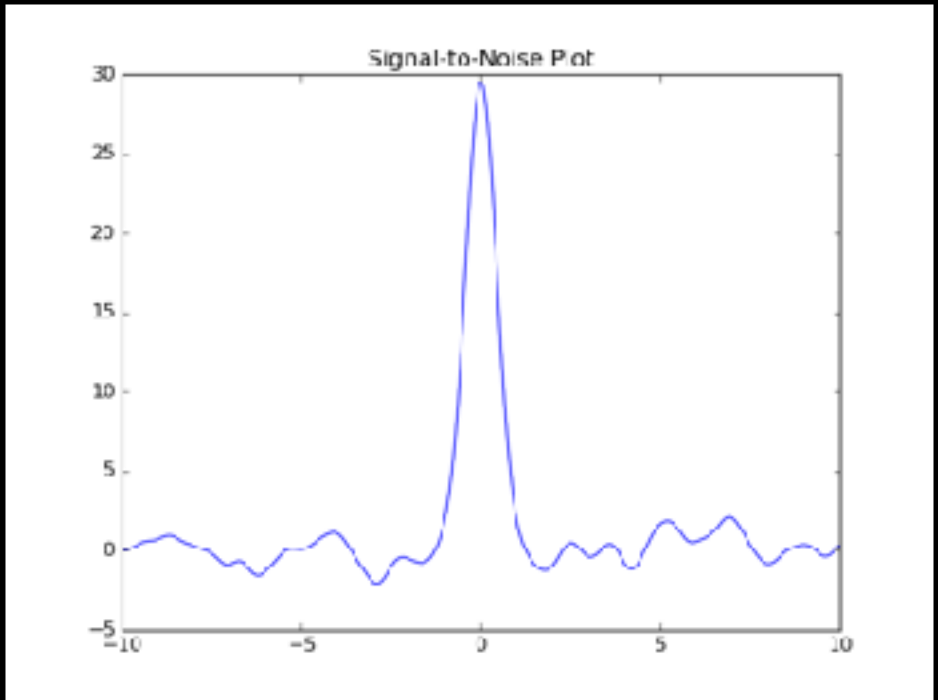
Code Example

```python
import numpy
from matplotlib import pyplot as plt

dx=0.01
noise=0.5
Ninv=1.0/noise**2
x=numpy.arange(-10,10,dx)
n=len(x)


x0=0
amp_true=2.0

sig=0.3
template=numpy.exp(-0.5*(x-x0)**2/sig**2)

dat=template*amp_true+numpy.random.randn(n)*noise

snr=numpy.zeros(n)
amp=numpy.zeros(n)
dat_filt=Ninv*dat
denom=(numpy.dot(template,Ninv*template))
rt_denom=numpy.sqrt(denom)
for i in range(n):
    template=numpy.exp(-0.5*(x-x[i])**2/sig**2)
    rhs=numpy.dot(template,dat_filt)
    snr[i]=rhs/rt_denom
    amp[i]=rhs/denom
```




```python
plt.clf();
plt.plot(x,snr);
plt.title('Signal-to-Noise Plot')
plt.savefig('snr_plot.png')

plt.clf();
plt.plot(x,amp)
plt.title('Source Amplitude Plot')
plt.savefig('amp_plot.png')

plt.clf();
plt.plot(x,dat)
plt.title('Raw Data')
plt.savefig('dat_raw.png')
```





But wait!

- We took $\sum d(t)a(t-\tau)$. But, this is just the correlation of d with a. We can do this quickly using Fourier transforms.

- Alternatively, let $a^{\lozenge}=a(-t)$. Then this is $\sum d(t)a^{\lozenge}(\tau-t)=d\otimes a^{\lozenge}$. By convolution theorem, this is $IFT(FT(d)FT(a^{\lozenge}))$.

- However, $FT=\sum f(x)\exp(-2\pi ikx/N)$. $FT(f(-x))=\sum f(-x)\exp(-2\pi ikx/N)=\sum f(x)\exp(2\pi ikx/N)=F^{*}(k)$. So, our output is just $IFT(FT(d)FT^{*}(a))$.

- NB - if a is symmetric, then A(k) is real (why?) and we can skip the conjugates.

How about changing noise?

- Often we're searching for things where noise is varying - say, searching for sources in a map that is deeper in the middle.
- Top: $A^T N^{-1} d$ is easy - correlate A with $N^{-1}d$.
- Bottom: $A^T N^{-1} A$ we can correlate N^{-1} with A^2.
- Amplitude stays ratio of the two (bottom is now a vector instead of a scalar), and SNR is top/$\sqrt{}$(bottom)

Matched Filter

- We are beginning on a very widespread class of techniques called *matched filters*.
- In various forms, they show up in GW analysis, photometry in optical images, finding galaxy clusters in CMB maps, finding radar return echos…
- Extendable to non-independent noise, multiple (possibly correlated) simultaneous/multi-frequency datasets, many others.
- All arises from writing down χ^2 and minimizing. Make a habit of this.