Mining large datasets: advice for the laptop seismologist

Peter Shearer

IGPP, U.C. San Diego

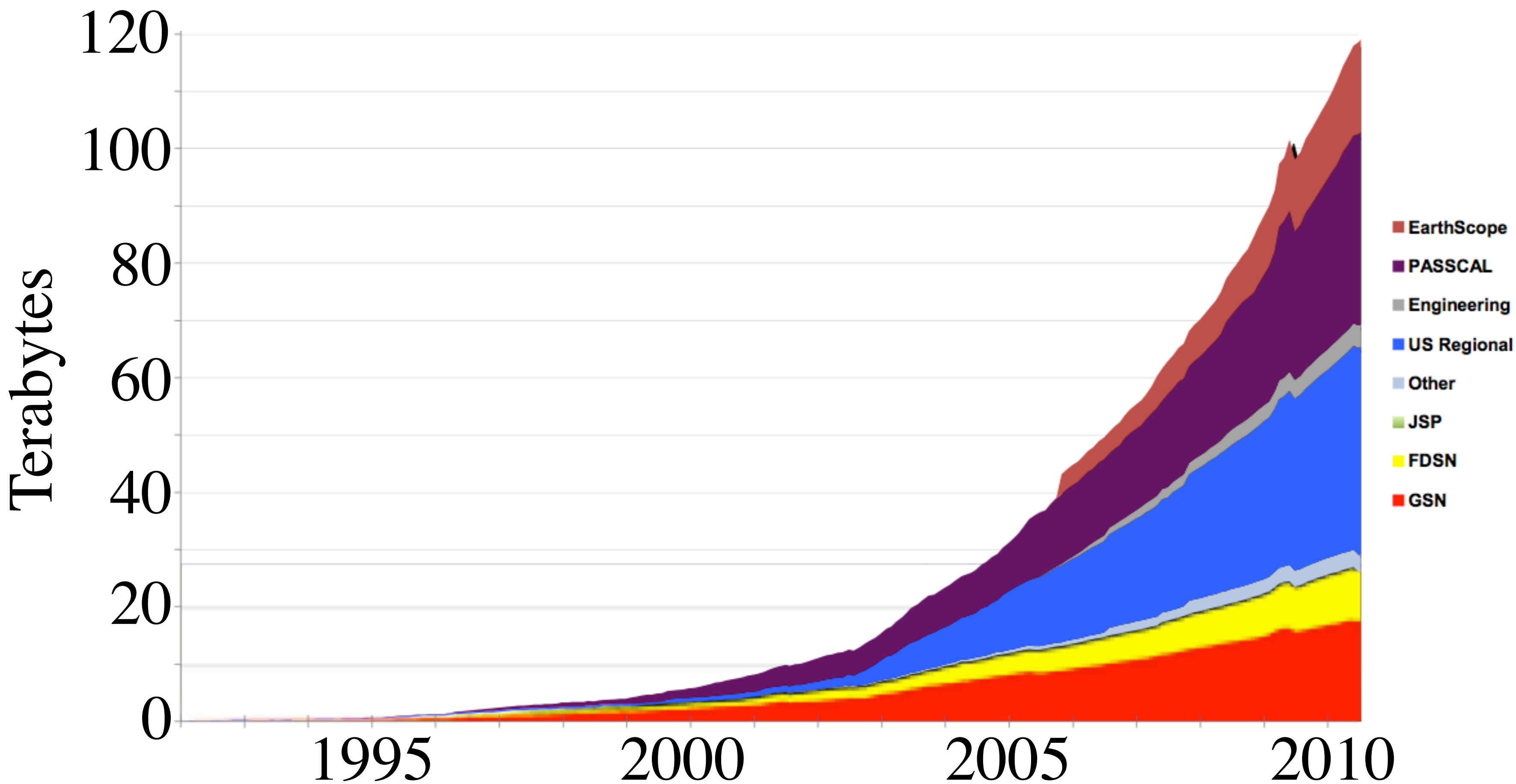
QUEST Workshop

July 13, 2011

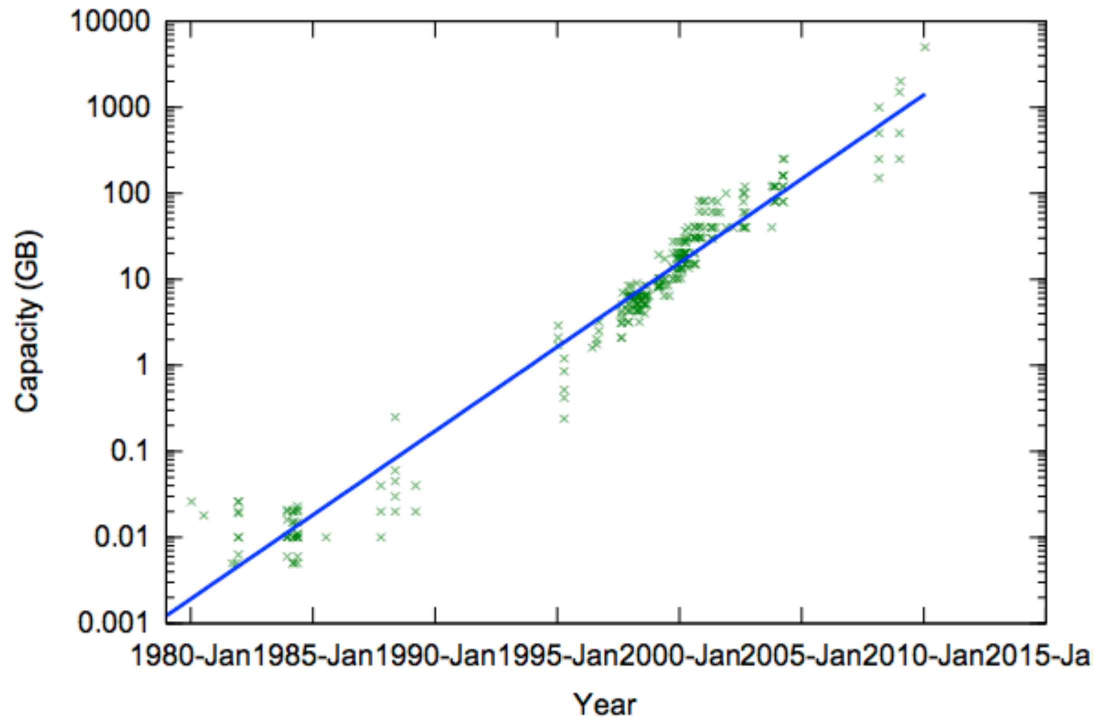
UCSD

UEST

Growth of IRIS seismic data archive

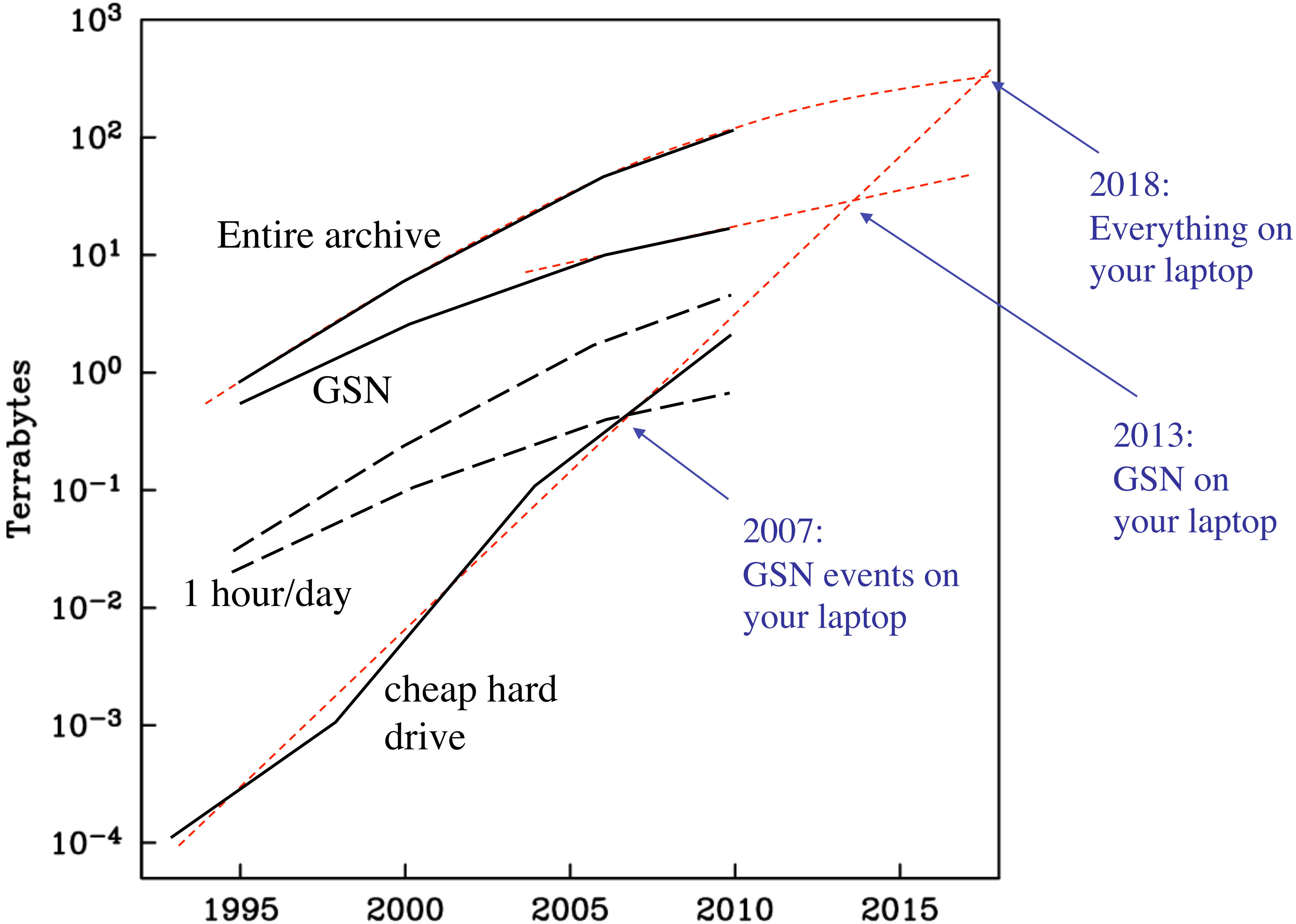


Computer storage has grown as power law

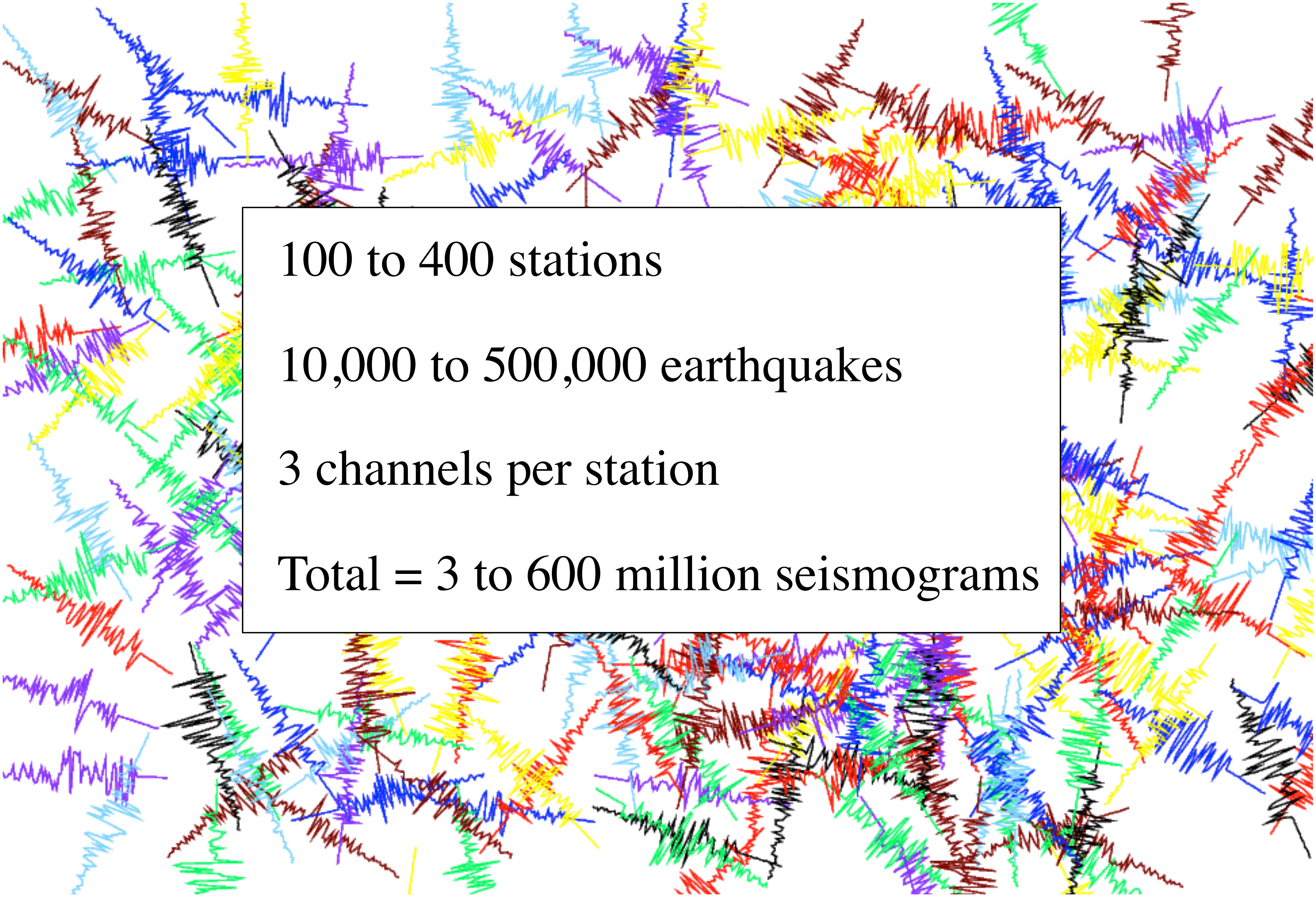


"Kryder's Law": disk space doubles every year

Predicting the future



What is a "large" seismic data set?



100 to 400 stations

10,000 to 500,000 earthquakes

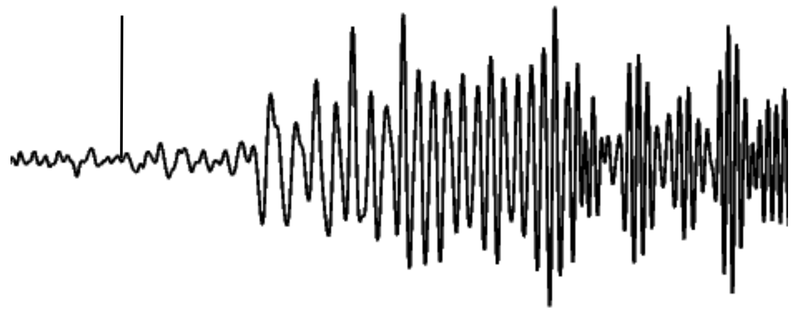
3 channels per station

Total = 3 to 600 million seismograms

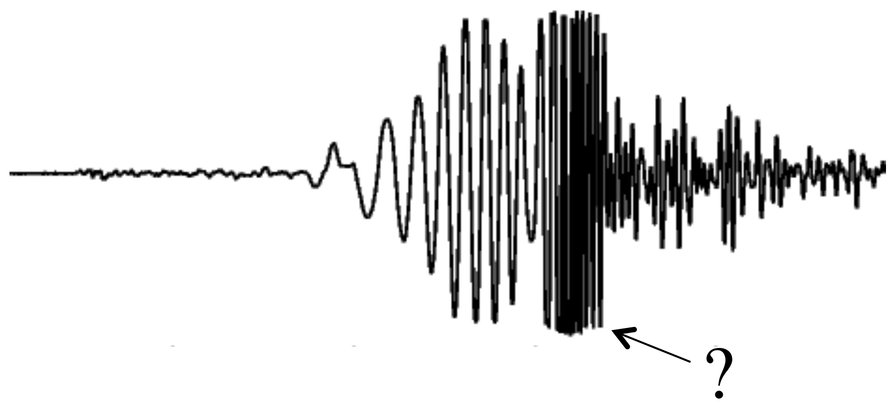
One seismogram

Data problems

Spikes



Clipping



Poor signal-to-noise

What to do about data problems?

Old School:

Look at the data by hand.

Don't trust "black box" analysis methods.

Seismograms are precious. You want to get everything you can out of the data and not waste a single record.

Today: What to do about data problems?

- Not practical to look at millions of records by eye
- Don't waste time trying to "fix" bad records. In data rich environment, you can afford to throw away 10% of your data.
- Devise automated processing methods that are *robust* with respect to data problems.
- Test these methods on subsets of the data using customized graphical user interfaces (GUI).

Strategies for large seismic data sets

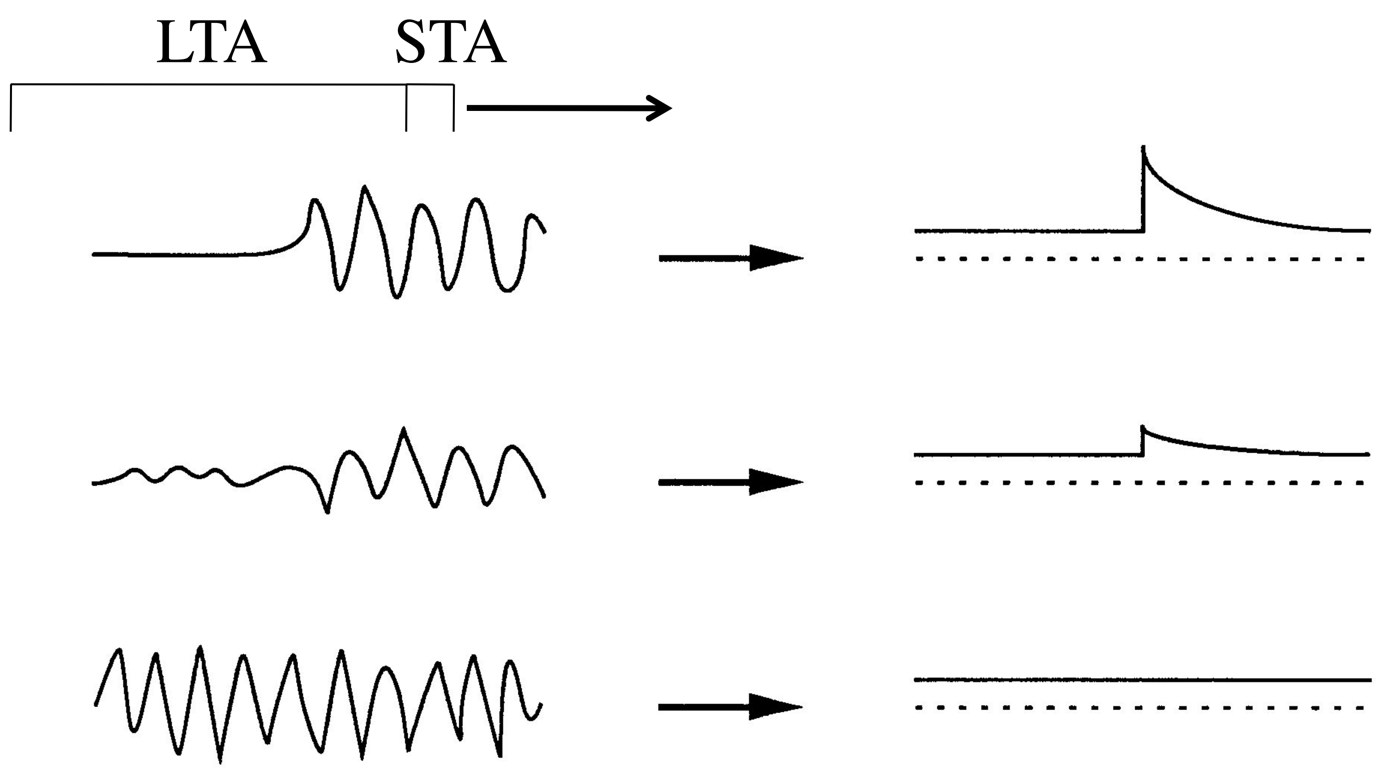
- Analyze entire dataset whenever possible.
- Use simple methods to get sense of data before doing complicated inversions.
- Consider reflection seismology methods like stacking and back-projection.
- Look for unanticipated signals in data, keep an open mind for new problems to work on.
- Avoid any hand-processing of seismograms!

~1989 — CD-ROM data distribution

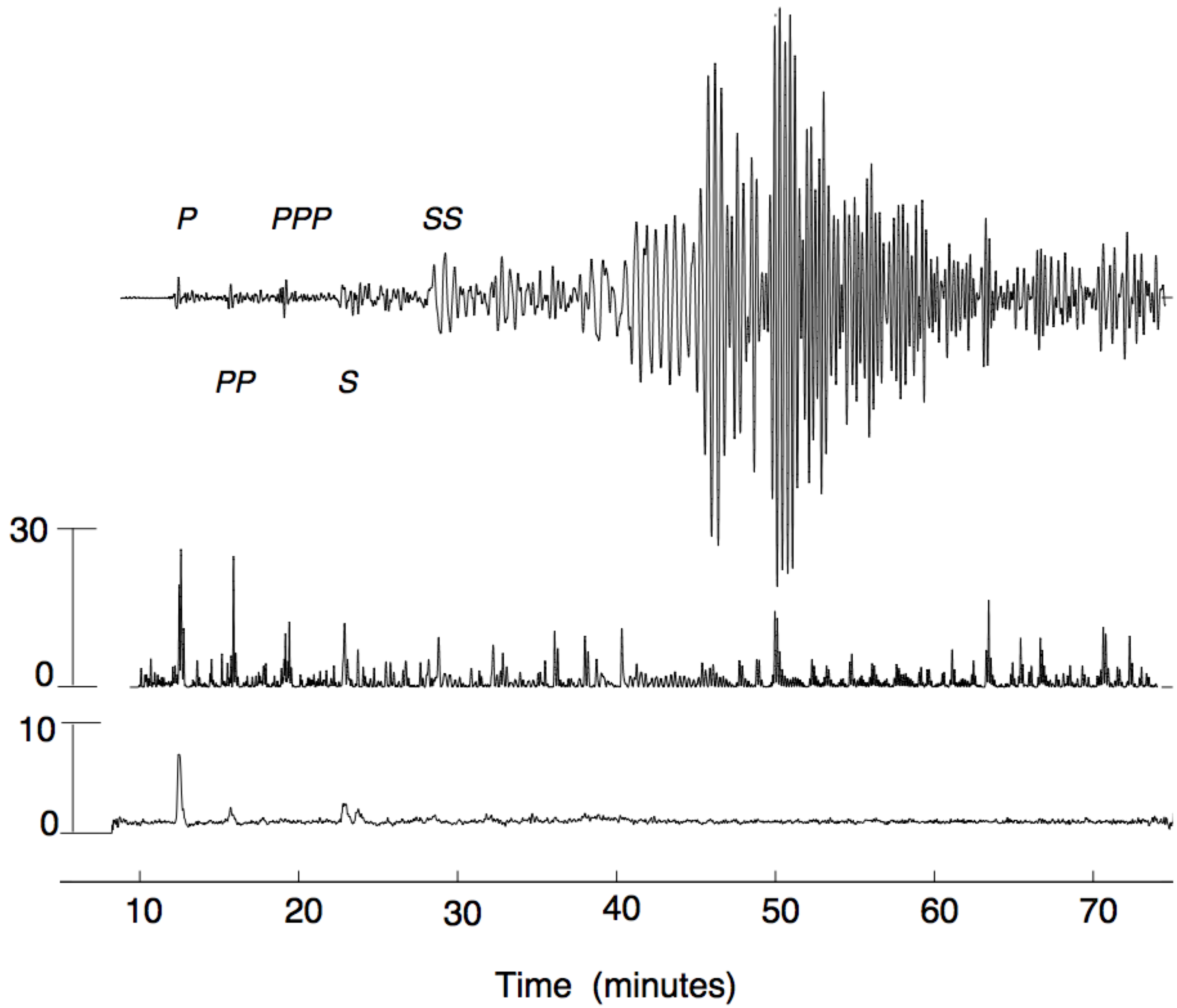
- 650 MB capacity
- Cheap to produce
- Selected events released by NEIC
- First reasonably practical access to large global datasets for individual seismologist

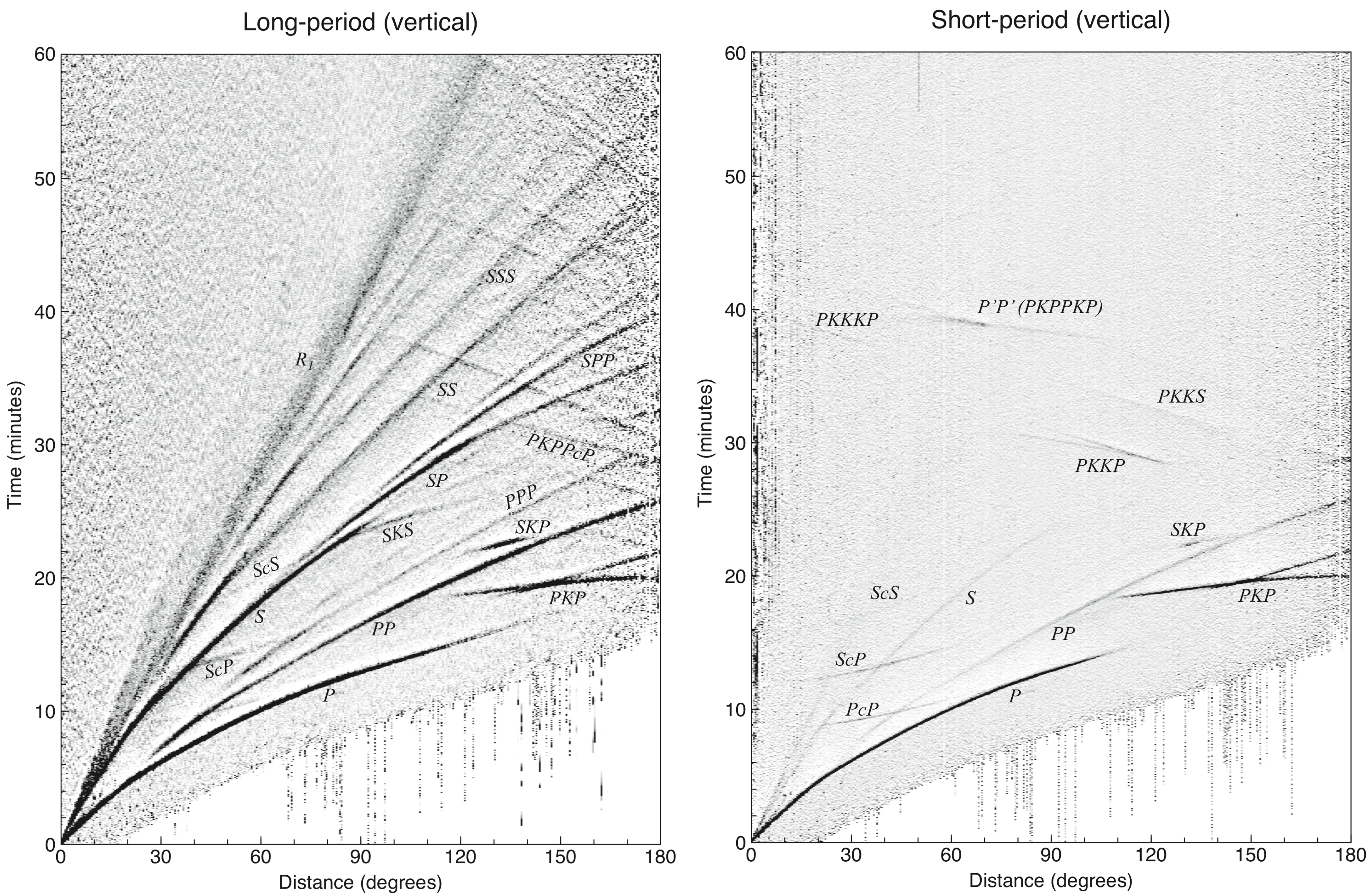
Example: STA/LTA stacking

- Calculate average absolute value in 5 s bins
- Divide each bin by average of previous 24 bins. This normalizes the amplitude of each trace.
- Stack in 0.5° distance bins

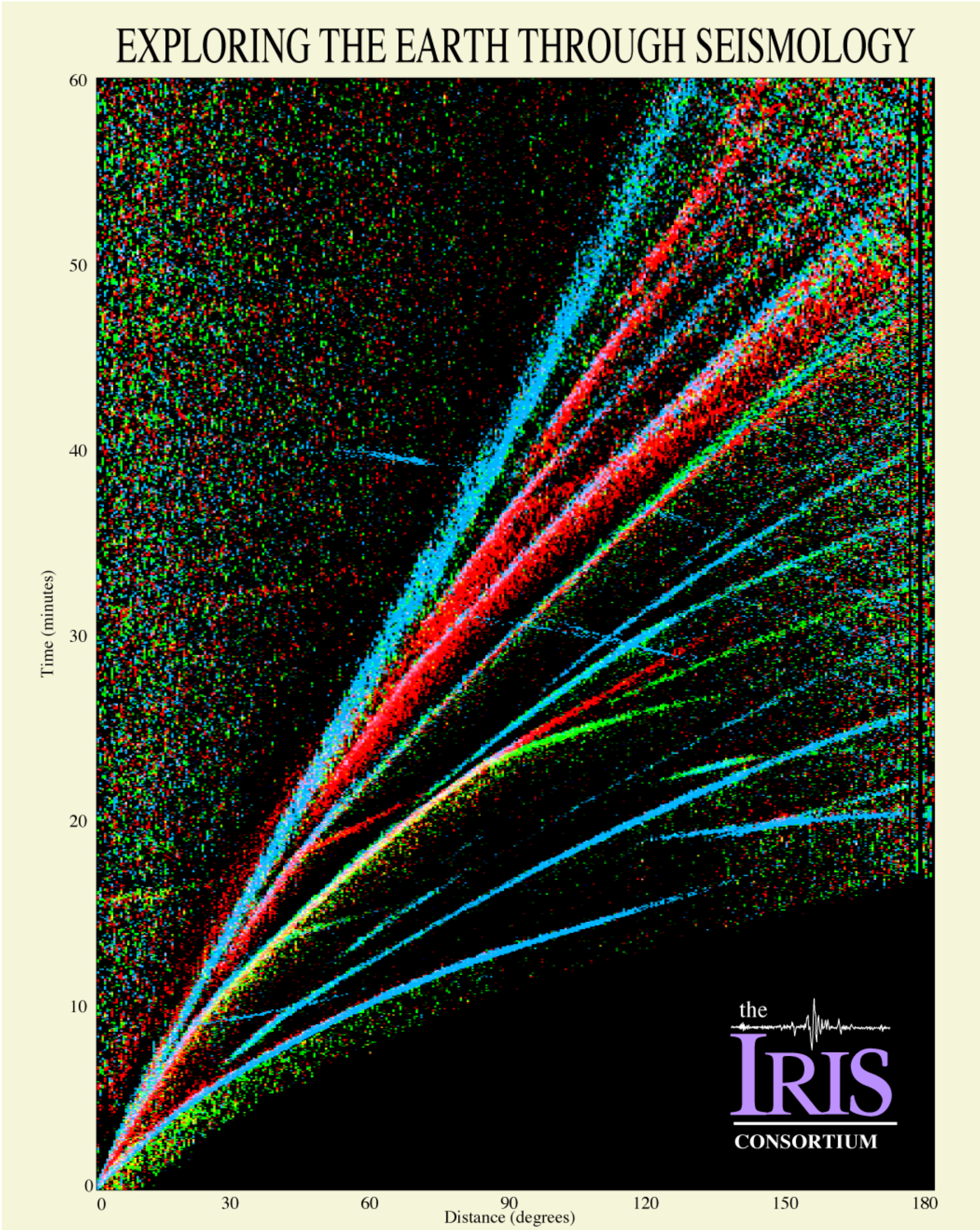


STA/LTA filter applied to one record





from *Astiz et al*. (1996)

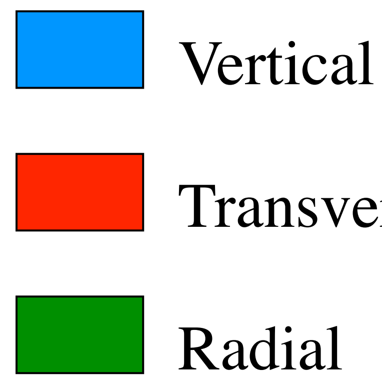


1988–1994

IRIS "Farm" archive

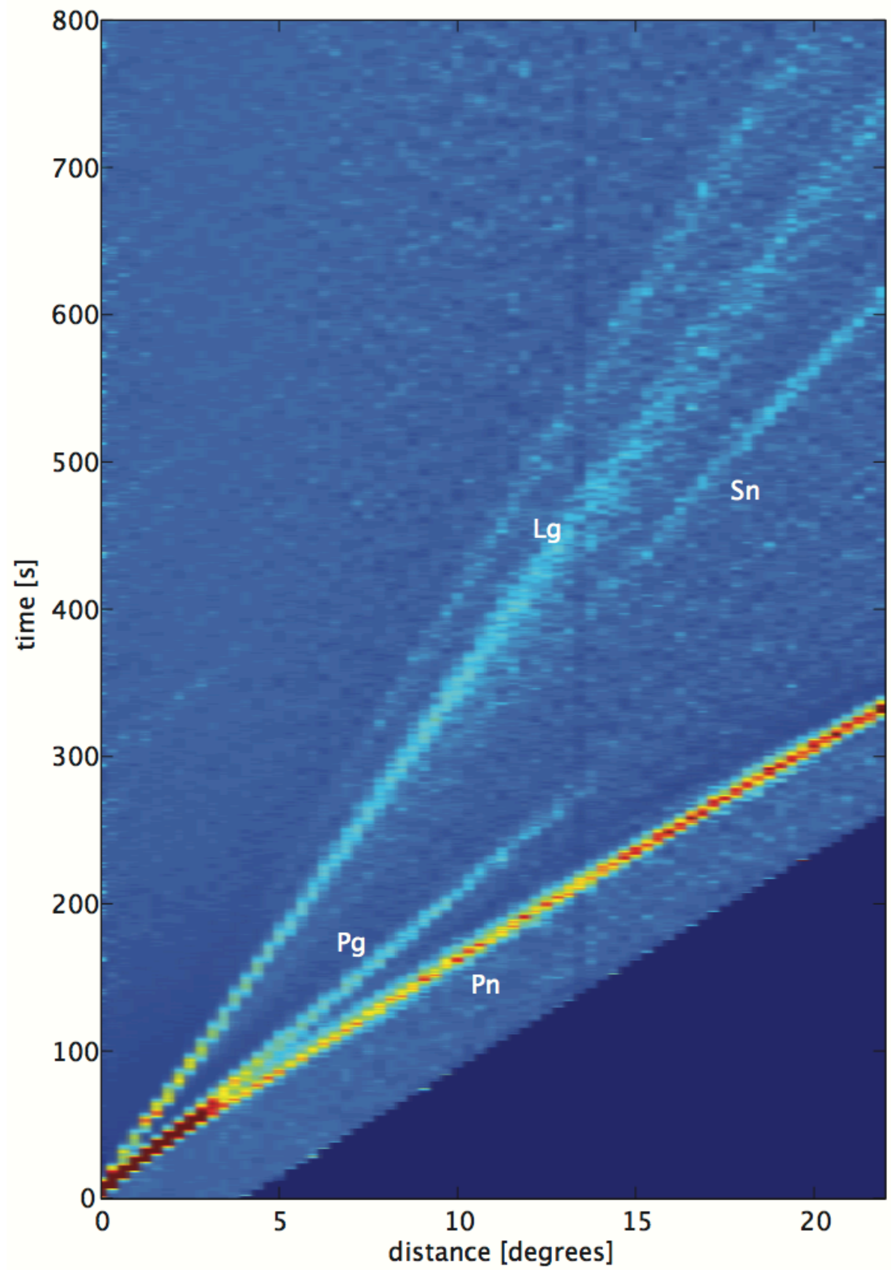
834 earthquakes

27,000 seismograms

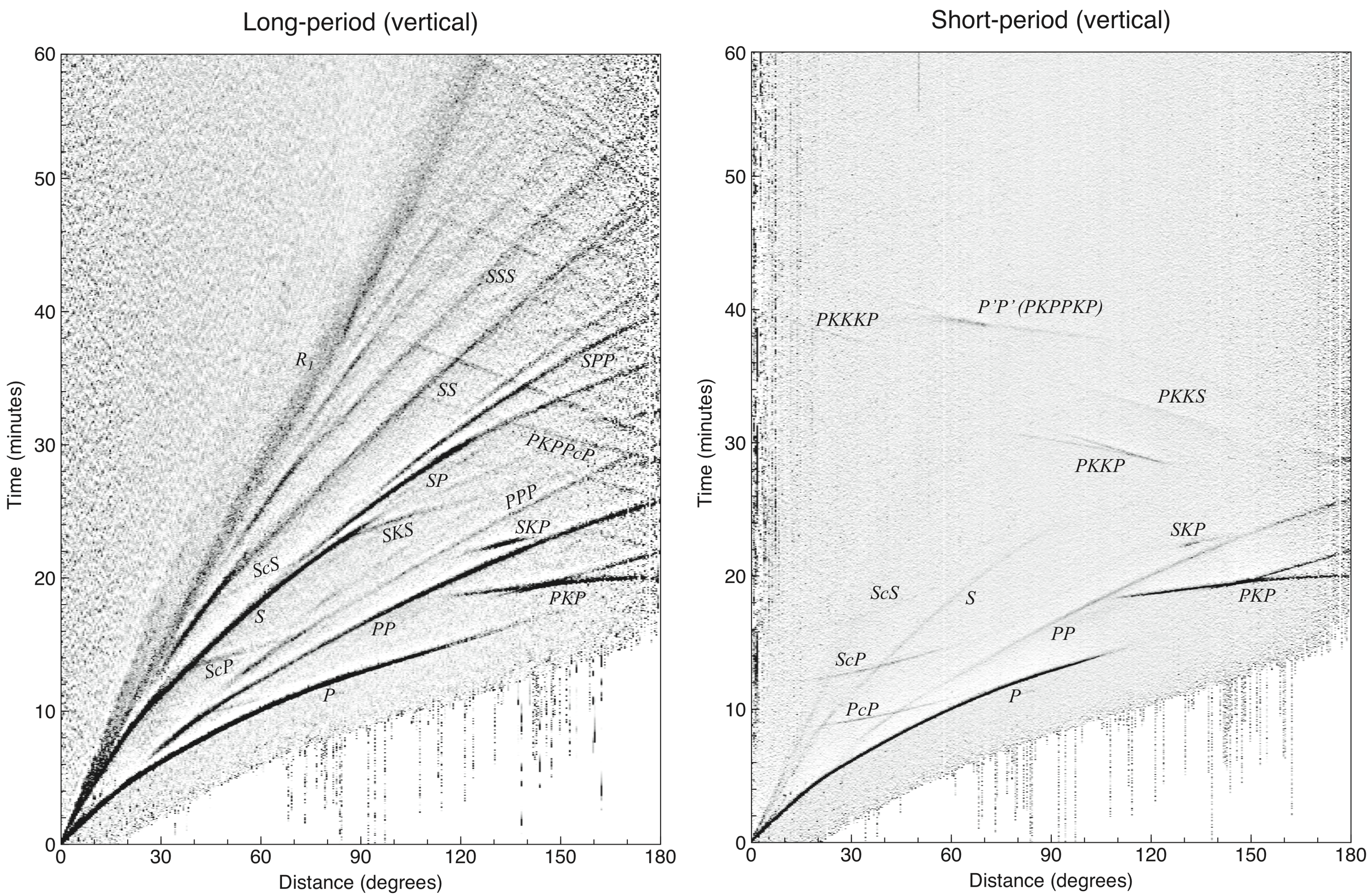


from *Astiz et al*. (1996)

USArray STA/LTA stack



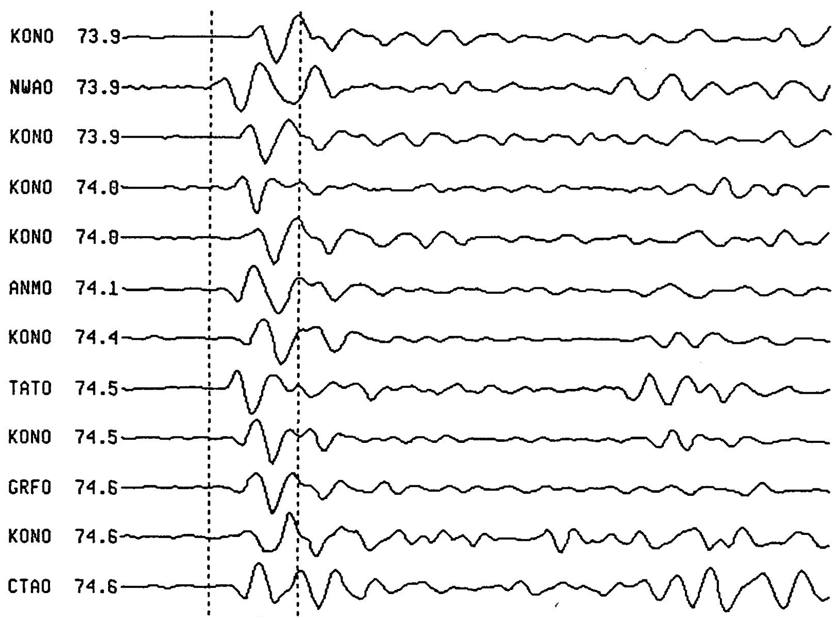
Plot courtesy Janine Buehler



from *Astiz et al*. (1996)

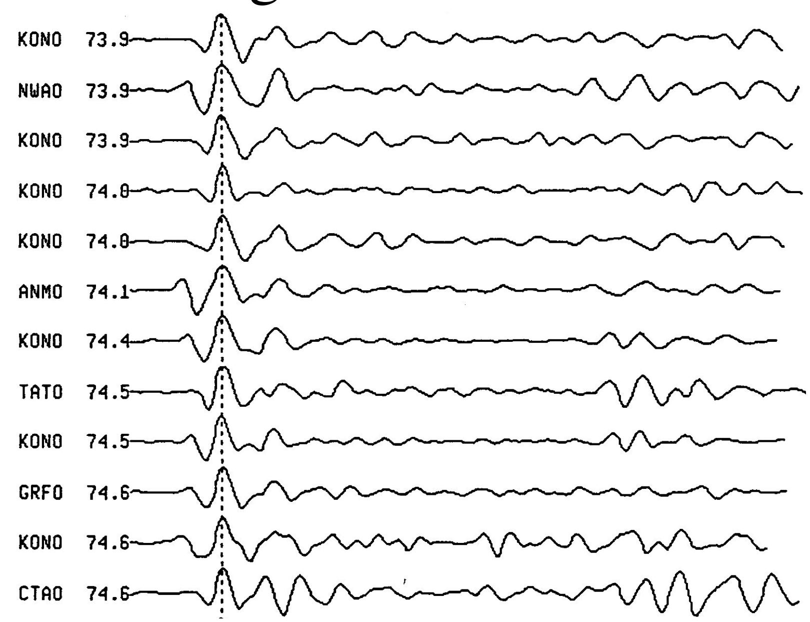
Stacking using a reference phase

Unaligned *SH* waves

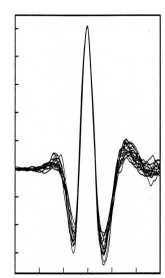


1 minute

Aligned *SH* waves



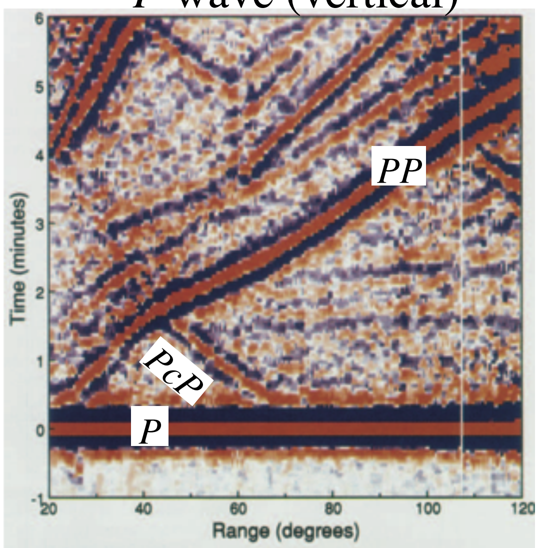
Stack



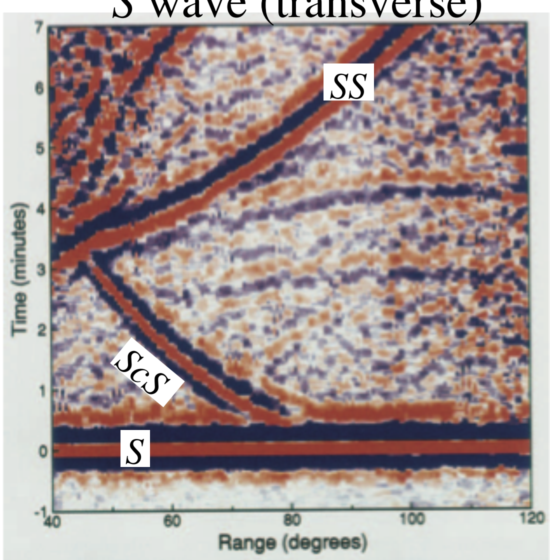
Reference pulse stacks for 20 different range bins

CD-ROM stacks (1991)

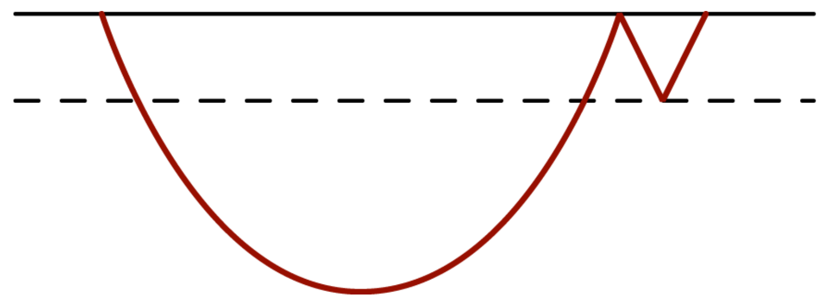
P wave (vertical)



S wave (transverse)



Topside reflections



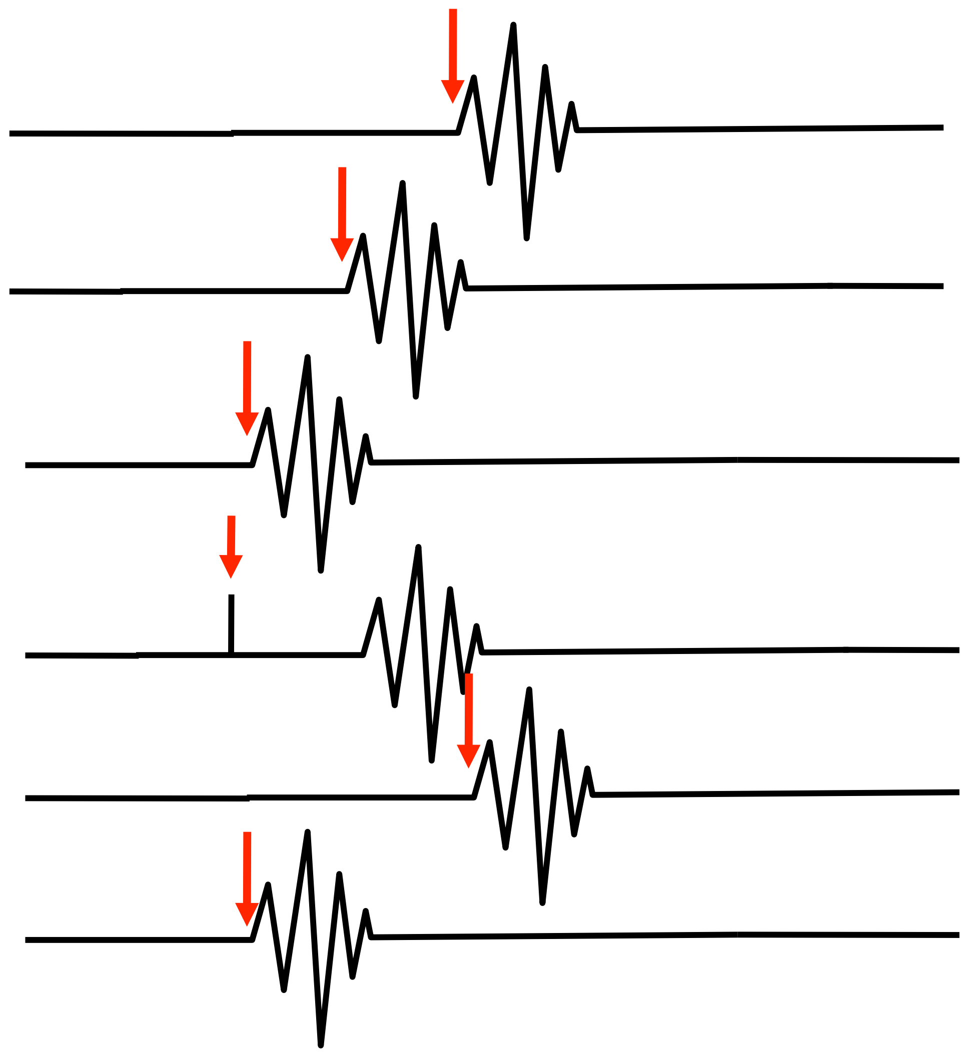
660-km discontinuity

410-km discontinuity

No global 220-km discontinuity

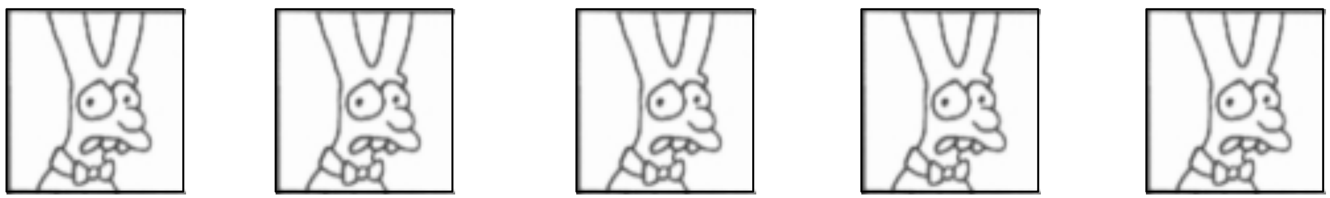
What are robust methods?

Example: noise spike causes bad pick



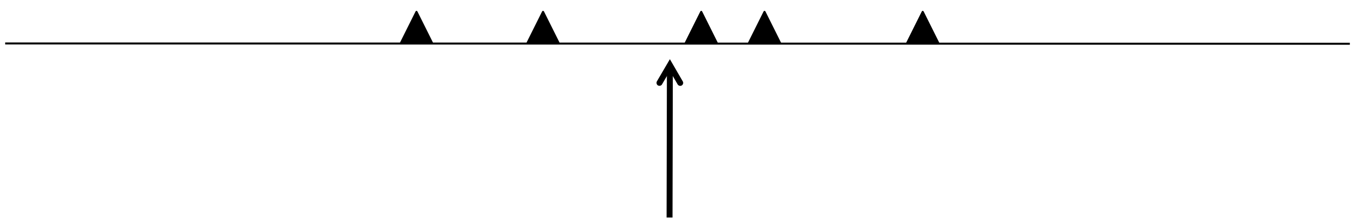
Analogy: estimating length

5 graduate students are told to measure the width of the computer room for a new carpet.



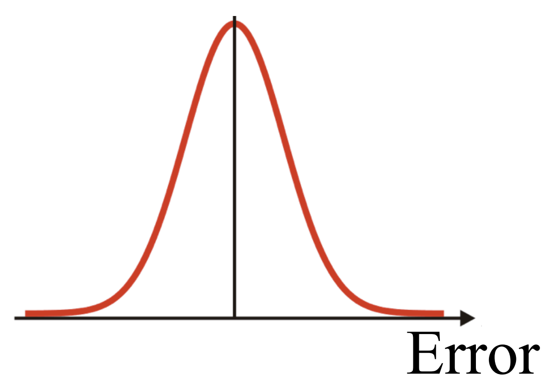
Their advisor averages their measurements to obtain the *best* estimate of the width.

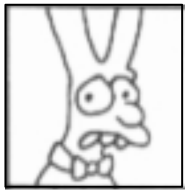
The *average* is actually the *best-fitting least-squares (L2-norm) estimate*!



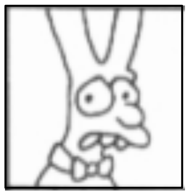
Best-fitting point minimizes the sum of the *squares* of the distances to all the points.

This is the *best* estimate of the true width of the room if the graduate student measuring errors have a *Gaussian distribution centered on zero*.

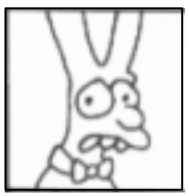




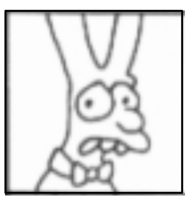
Grad student #1 measures 123 inches



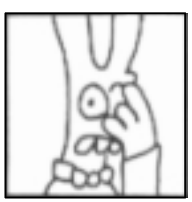
Grad student #2 measures 121 inches



Grad student #3 measures 124 inches

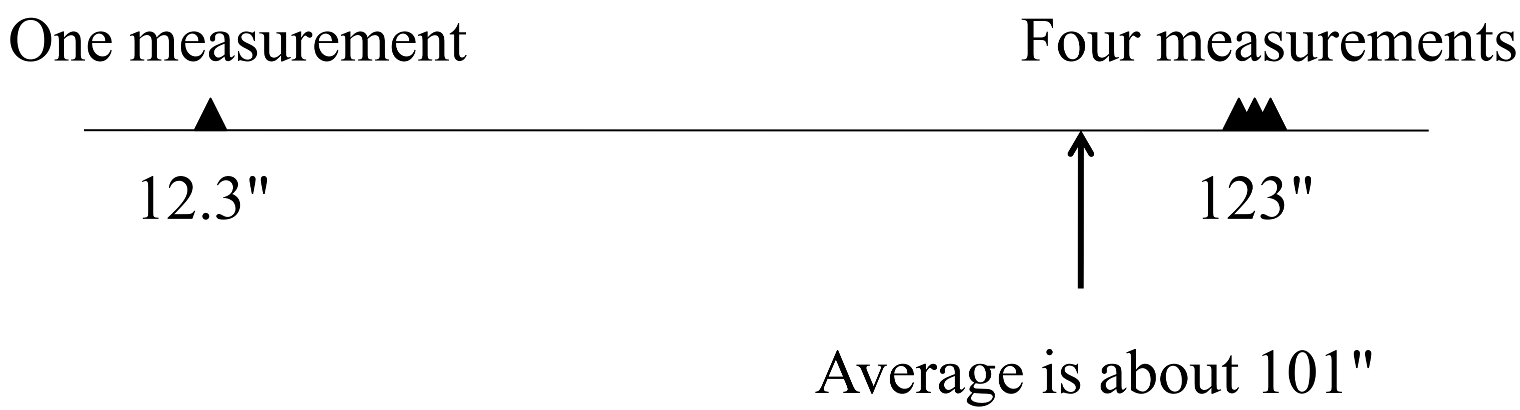


Grad student #4 measures 123 inches



Grad student #5 measures 12.3 inches

Least squares is not a *robust* method



Least squares is very sensitive to *bad data points* (outliers) that fall way outside the Gaussian distribution of the good data points.

A robust method: the *median*

Example: How much are houses worth in Beverley Hills?

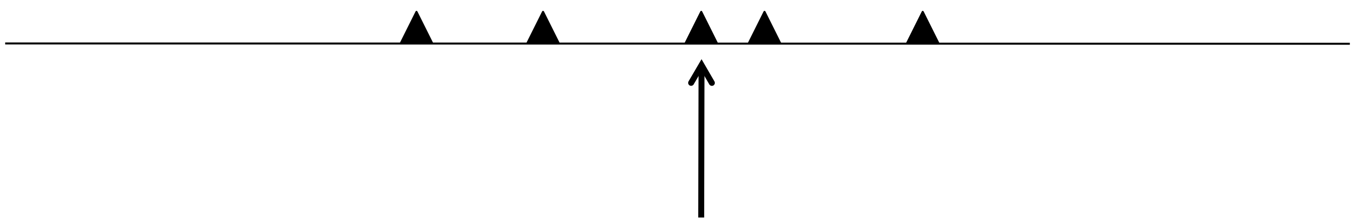
Average sale price: $3.5 million

Median sale price: $1.5 million

(90210 area code)

Beverley Hills housing prices are an example of a long-tailed (non-Gaussian) distribution

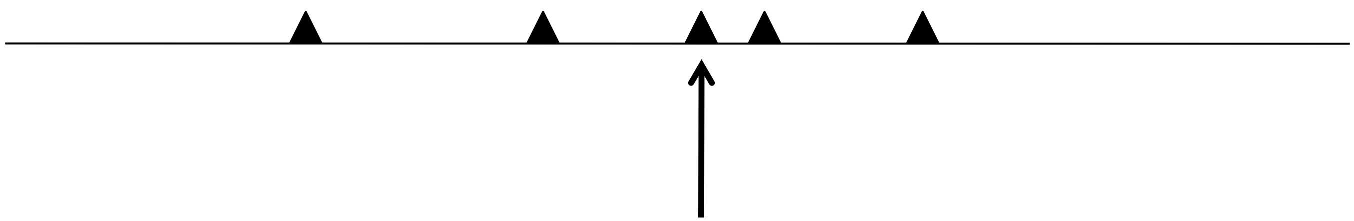
The *median* is actually the *best-fitting least-distance (L1-norm) estimate*!



Median minimizes the sum of the distances to all the points

Notice that the median puts half the points to the left and half to the right. It does not care how far away they are.

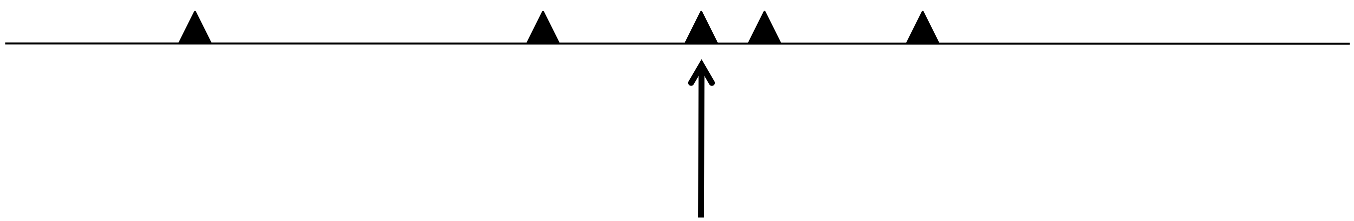
The *median* is actually the *best-fitting least-distance (L1-norm) estimate*!



Median minimizes the sum of the distances to all the points

Notice that the median puts half the points to the left and half to the right. It does not care how far away they are.

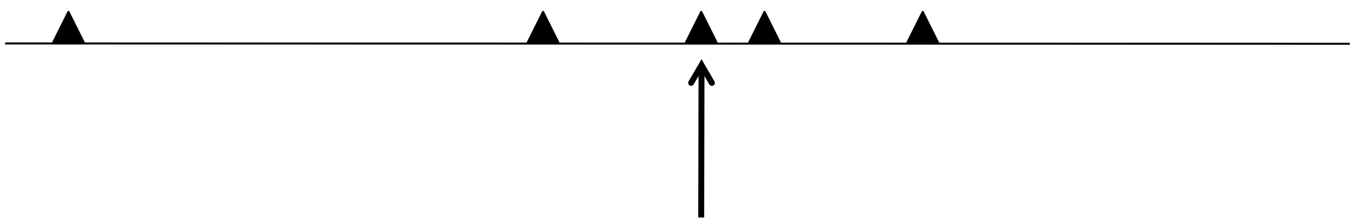
The *median* is actually the *best-fitting least-distance (L1-norm) estimate*!



Median minimizes the sum of the distances to all the points

Notice that the median puts half the points to the left and half to the right. It does not care how far away they are.

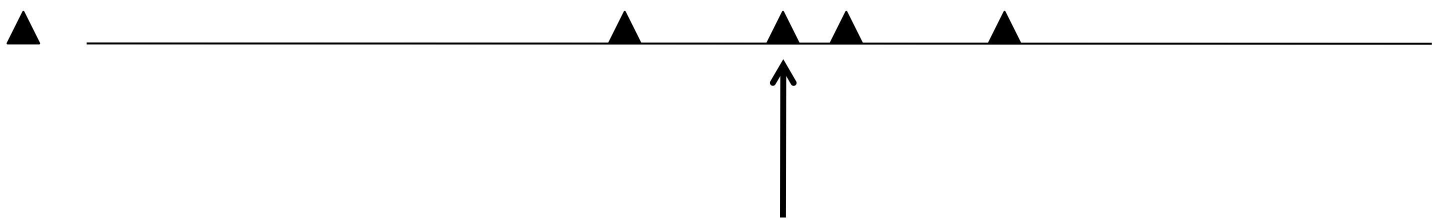
The *median* is actually the *best-fitting least-distance (L1-norm) estimate*!



Median minimizes the sum of the distances to all the points

Notice that the median puts half the points to the left and half to the right. It does not care how far away they are.

The *median* is actually the *best-fitting least-distance (L1-norm) estimate*!



Median minimizes the sum of the distances to all the points

Notice that the median puts half the points to the left and half to the right. It does not care how far away they are.

Moral: Don't blindly use least squares

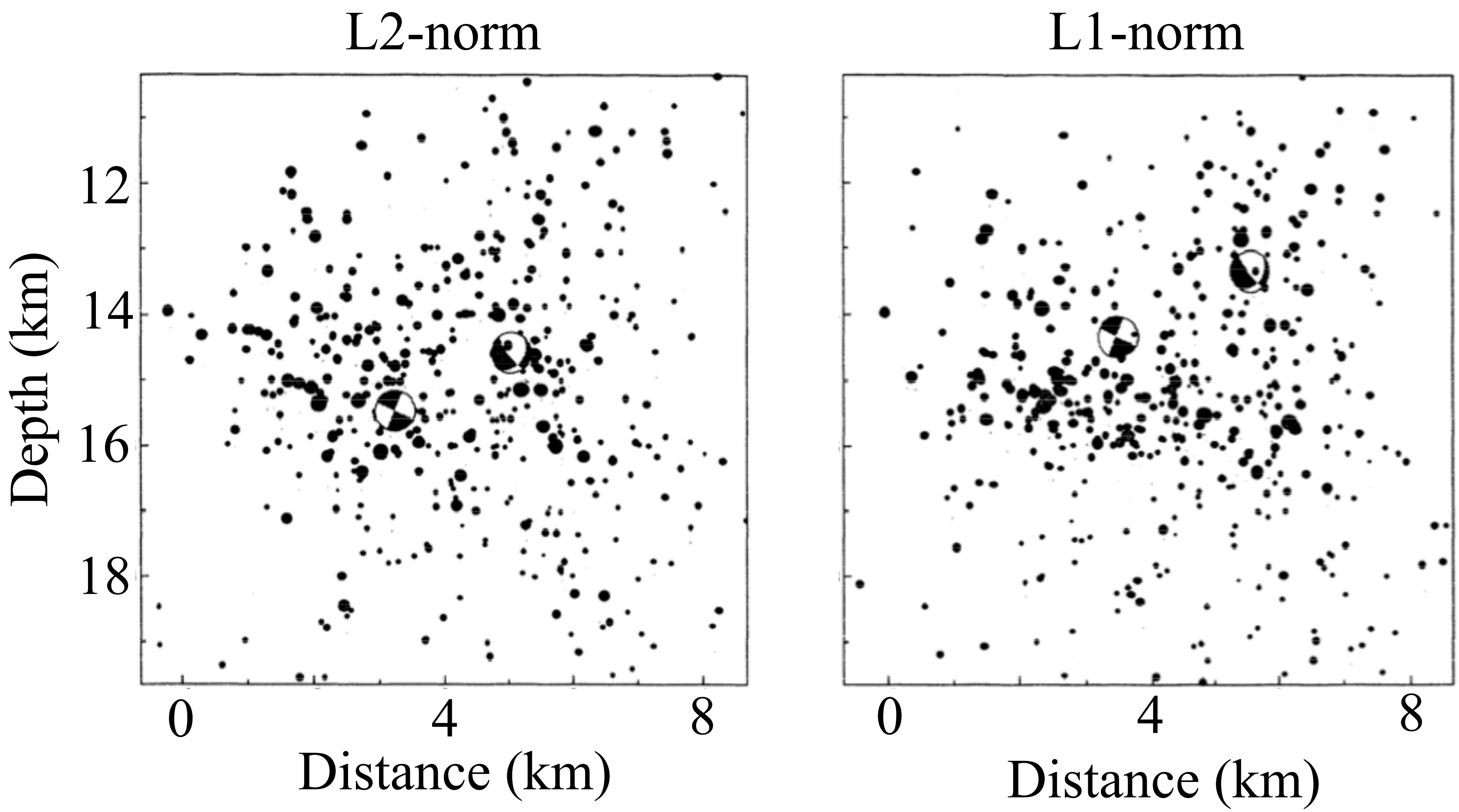
- Consider more robust norms such as L1. Median can be slow, but faster, iterative methods exist. Ask me about my robust mean subroutine *robomean*.

- Or apply iterative outlier identification and removal.

- There are many valid strategies, but *don't ignore the problem*. Almost all large datasets are non-Gaussian.

Example: Earthquake location from arrival time picks

Aftershocks of 1987 *M* 5.9 Whittier Narrows earthquake

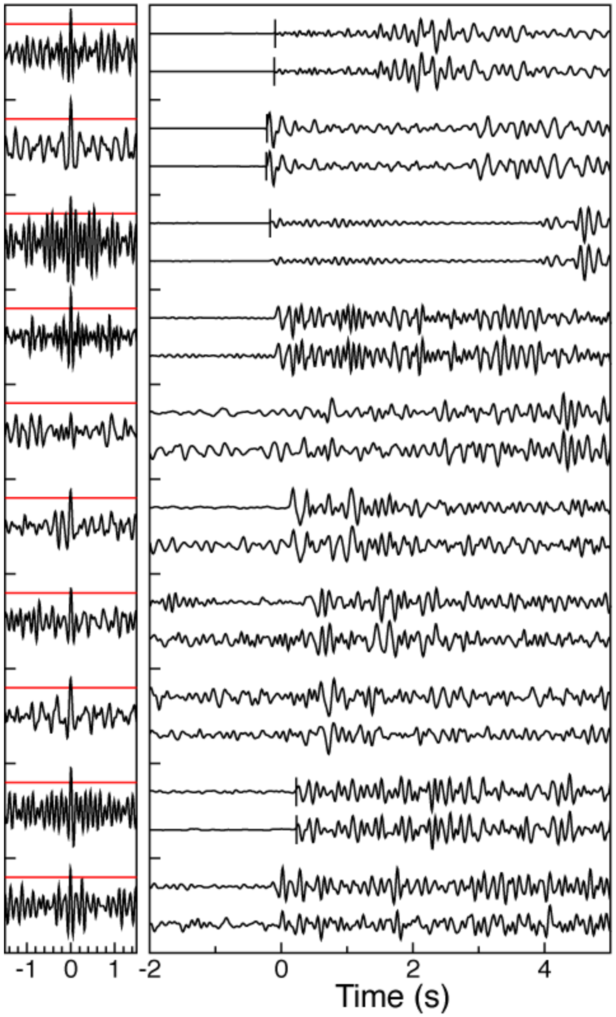


from Shearer (1997)

Caltech/UCSD Southern California Relocation Project

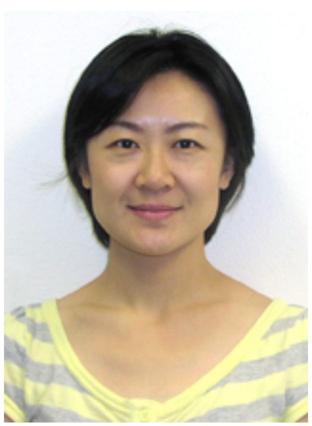
- 1981 to 2005 waveforms now online at Caltech
- Cross-correlation completed for 94 million event pairs
- Relocated catalogs now available at SCEDC
- Latest is LSH catalog (Lin et al., 2007)

Egill Hauksson

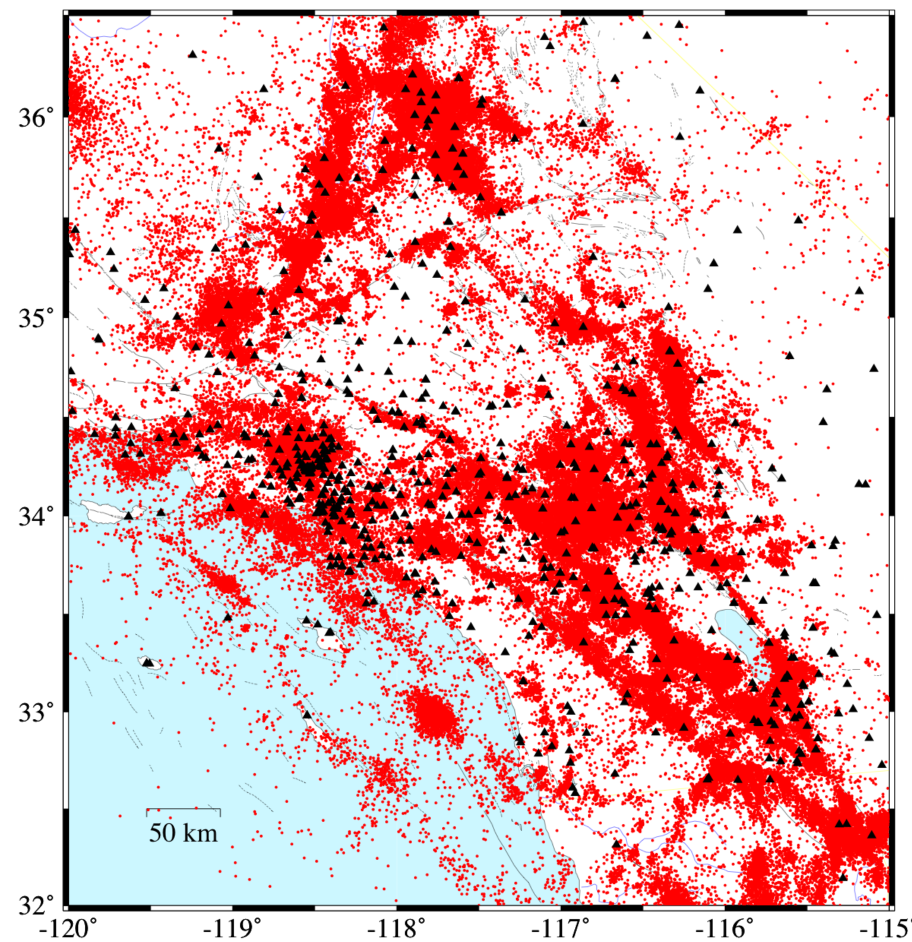


LSH Catalog

- Study Period: from 1981 to 2005
- 452,943 events
- *P*- and *S*- phase arrival times
- Waveform data
- 783 SCSN stations

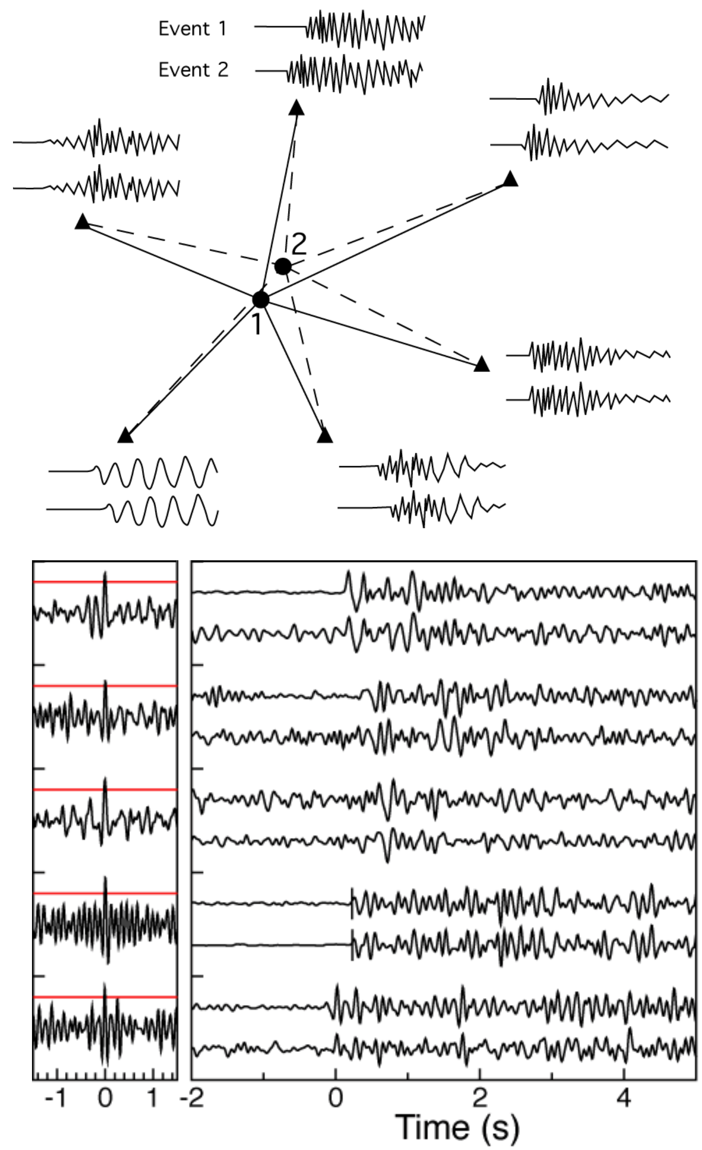


Guoqing Lin

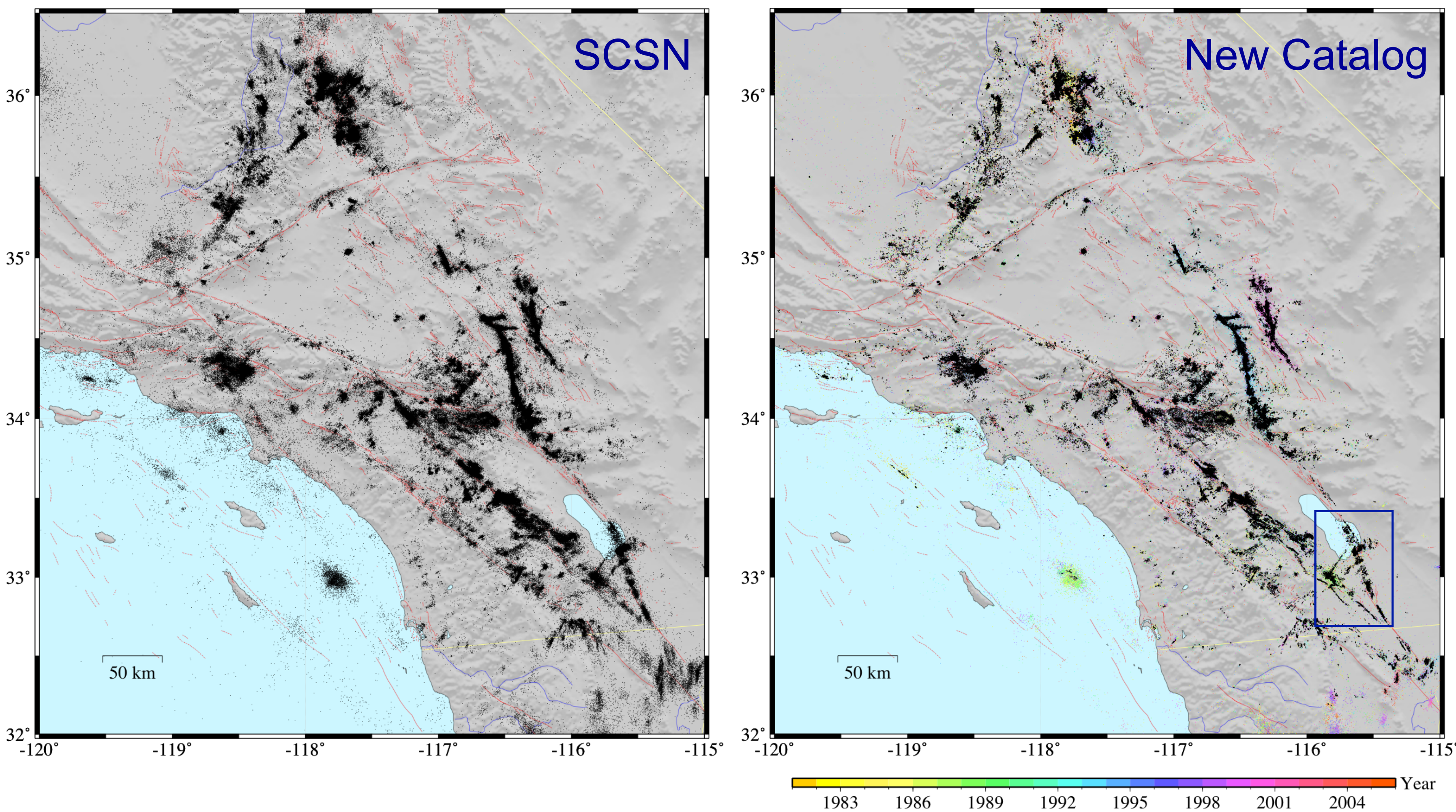


Waveform Cross-correlation

- 1981 to 2005 seismograms from 450,000 events now online at Caltech.
- Time-domain cross-correlation method with sub-sample precision, applied to filtered waveforms from all stations, channels and components.
- Cross-correlation completed for 95 million event pairs, including all pairs separated by < 2 km in source-specific station term (SSST) catalog computed from phase picks.
- About 10 times more cross-correlations than our previous analysis.

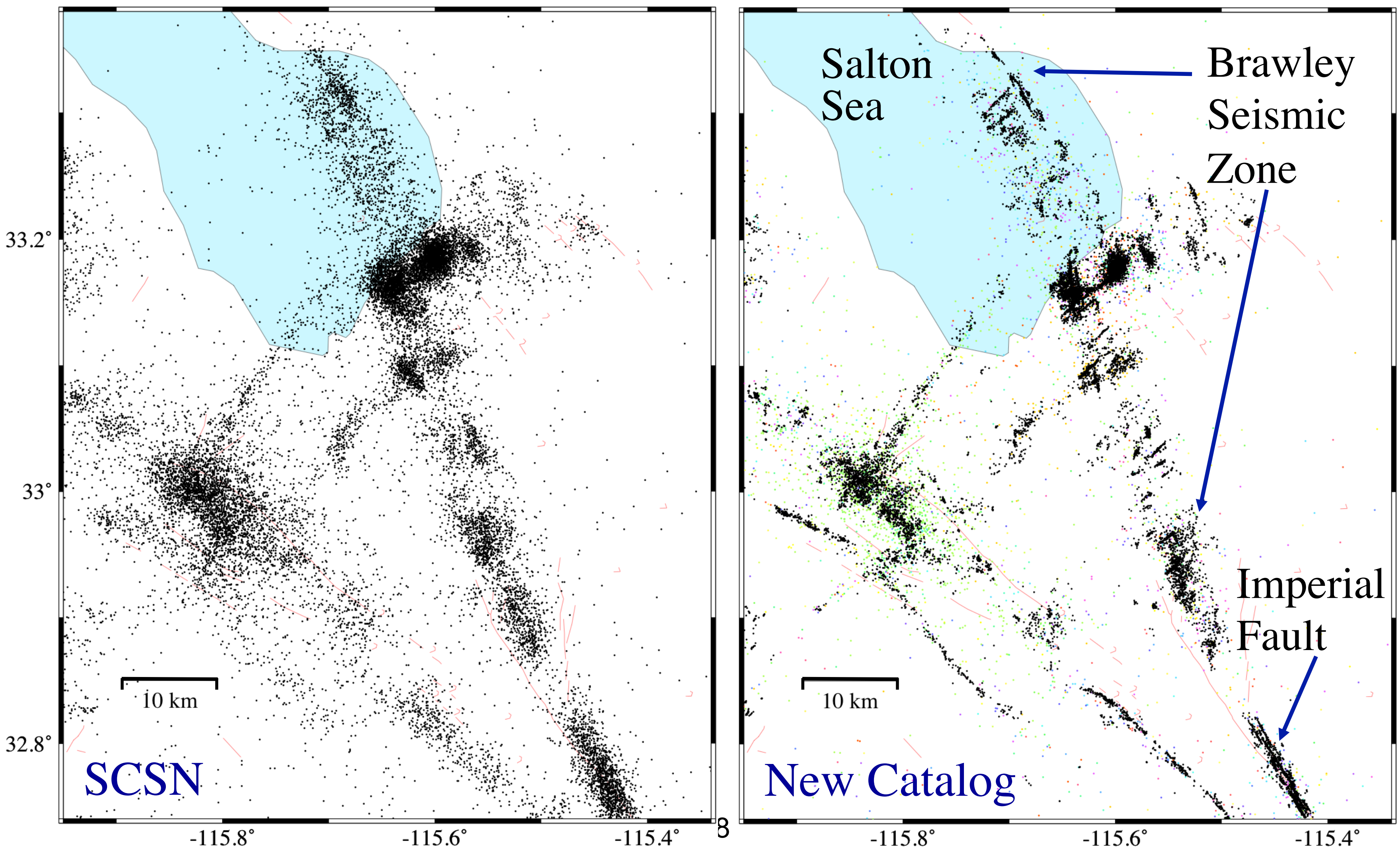


Location Comparison



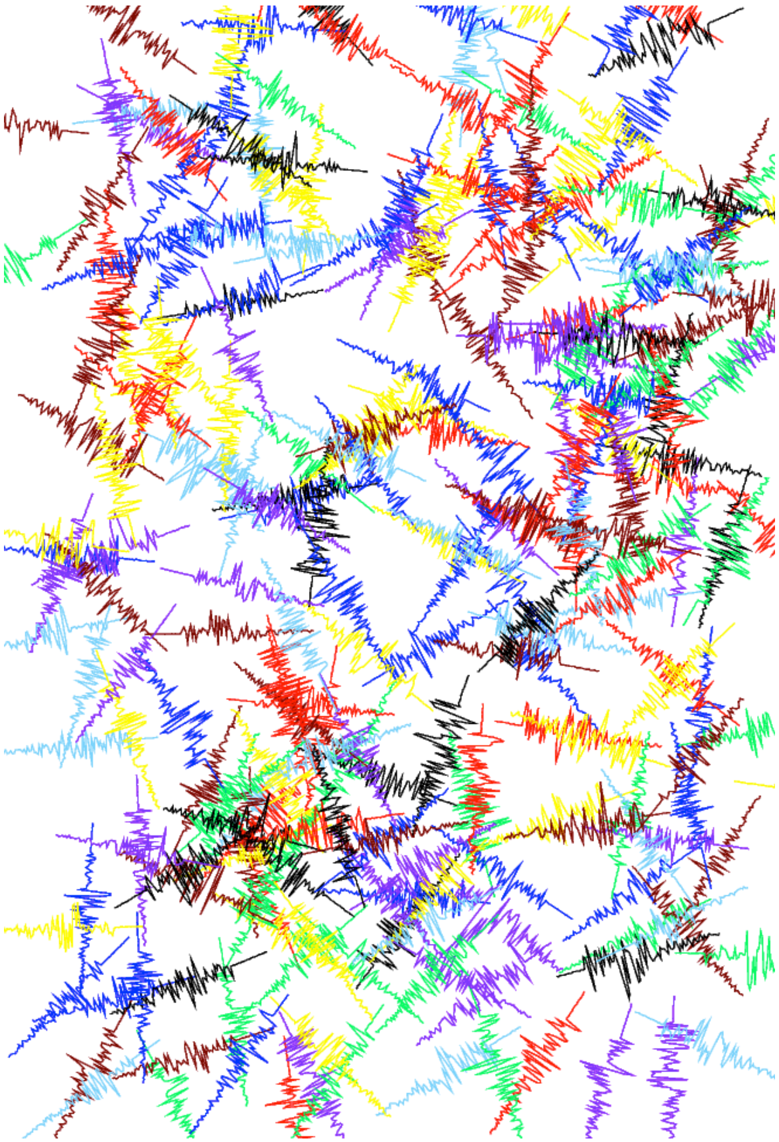
- Black dots show similar event clusters, relocated using cross-correlation data
- ~25% of events don't correlate, are plotted in color by year

Location Comparison



Personal experiences

- Commercial databases are hard to use for large-scale data processing. They store things in different places and rarely do exactly what you want.
- SEED is a distribution format, not a working data analysis format. You will need to convert to ah, SAC, gfs, etc. (ask me about EFS format).
- Store big data files in large binary blocks for fast I/O. Format should be simple and easy to understand.
- Matlab is slow. I prefer "real" languages like Fortran or C.



Personal experiences, continued

- Learn to write UNIX scripts. They make everything much more repeatable.
- Your time is more valuable than computer time! Don't worry about rerunning things when necessary to save yourself time.
- Computers keep getting better and cheaper. Buy a new computer every two to three years.

```
foreach year (1997 1998 1999 2000)

set xdrdir = /Volumes/LaCie_1TB_Drive_A/HVO_XDR_files/${year}

set sacdir = /Volumes/LaCie_1TB_Drive_A/SAC/${year}

foreach month (01 02 03 04 05 06 07 08 09 10 11 12)

cd ${xdrdir}/$month
rm -f junk*
ls *.XDR >! junk1
sed 's/.XDR//g' junk1 >! xdrlist
rm -f junk*

cd ${sacdir}/$month

foreach cuspid (`cat ${xdrdir}/${month}/xdrlist`)
echo ${cuspid}

rm -fr ${cuspid}.dir
mkdir ${cuspid}.dir
cd ${cuspid}.dir

echo ${xdrdir}/${month}/${cuspid}.XDR

/home/shearer/PROG/HVO/CONVERSION/WOLFE/ah/AHUNCAT/ahuncat ${cuspid} < $
{xdrdir}/${month}/${cuspid}.XDR

ls ${cuspid}.* >! filelist

foreach file (`cat filelist `)

/home/shearer/PROG/HVO/CONVERSION/WOLFE/lpmauna_picks/AH2ASC/ahtimeandpol < $
{file} > ${file}.pick

/home/shearer/PROG/HVO/CONVERSION/WOLFE/fcu/AH2SAC/ah2sac_lin < ${file} > $
{file}.sac
#rm -f ${file}

end
cd ../
end
end
```

Advice for large-scale data analysis

- First test automated processing methods using GUI on small problems.
- Modify code until it seems to be working.
- Anticipate where things might go wrong and add suitable warning flags in program.
- Develop ways to examine output to make sure things are working right.
- Don't assume you are done the first time the code runs completely without crashing. Things can still go wrong. Always keep testing and thinking of possible problems.
- We are blessed to work in a data-rich field. Don't forget to have fun with the data!