CS 188: Artificial Intelligence

Application: AI for Global Nuclear Monitoring

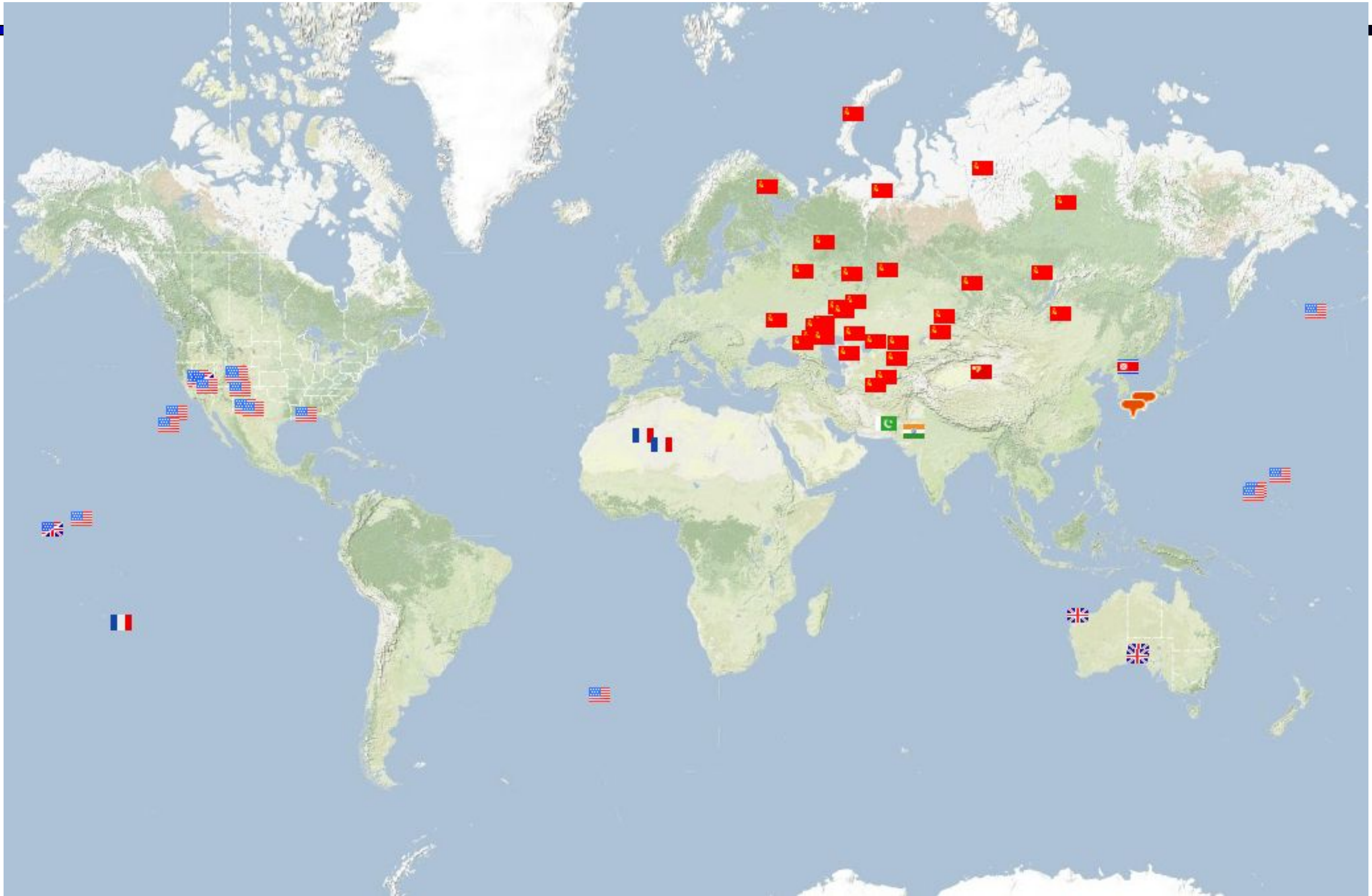
Instructor: Dan Klein and Stuart Russell

University of California, Berkeley

Outline

- Comprehensive Nuclear-Test-Ban Treaty
- Seismology 101
- Bayesian solutions
- NETVISA: A Bayesian model for “blips”
- SIGVISA: A Bayesian model for complete seismograms

2054 nuclear explosions, 300K deaths



CTBT

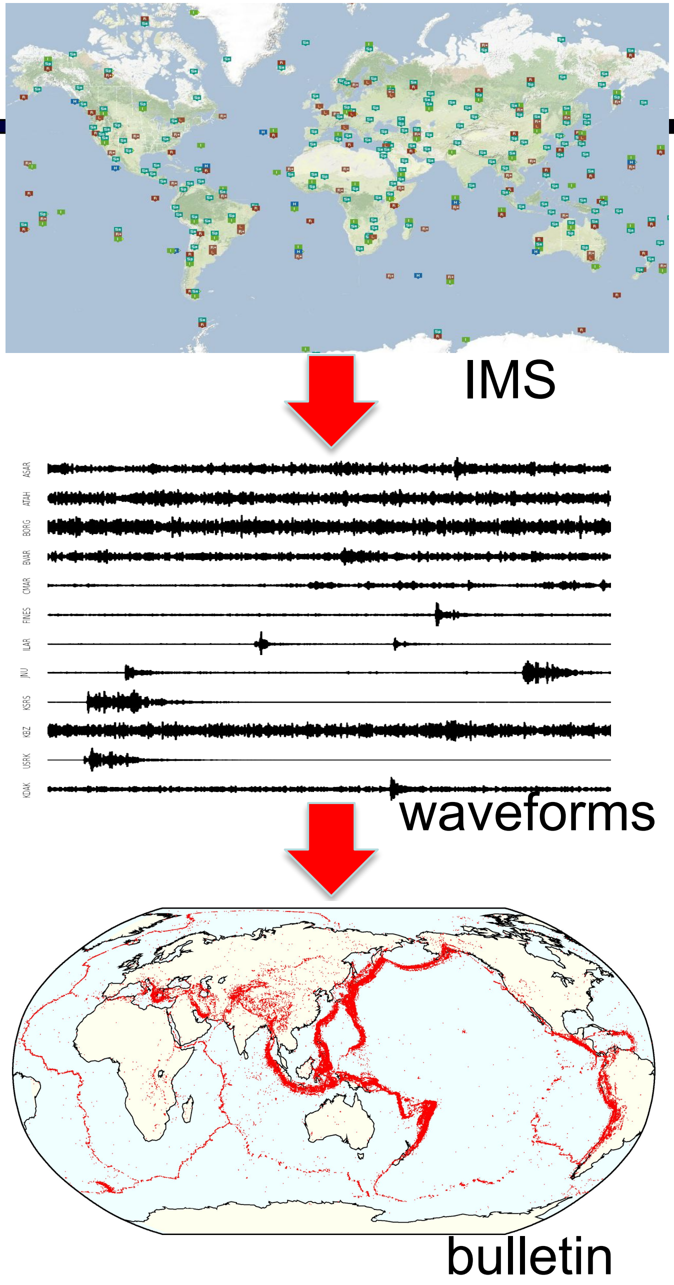
- First formal treaty proposal in 1958 (Eisenhower's letter to Khrushchev)
- *"Failure to achieve a ban on nuclear testing, would have to be classed as the greatest disappointment of any administration -- of any decade -- of any time and of any party."*
- Treaty signed in 1996 (eventually 187 nations)
  - Allows for outside inspection of 1000km^2
- US Senate refused to ratify in 1998
  - "too hard to monitor"

321 monitoring stations (170 seismic)

Global seismic monitoring

- Goal: improve monitoring sensitivity and accuracy

- **Evidence**: waveforms from 170 seismic stations
- **Query**: what happened?
- **Model**: geophysics of event occurrence, signal transmission, detection, noise



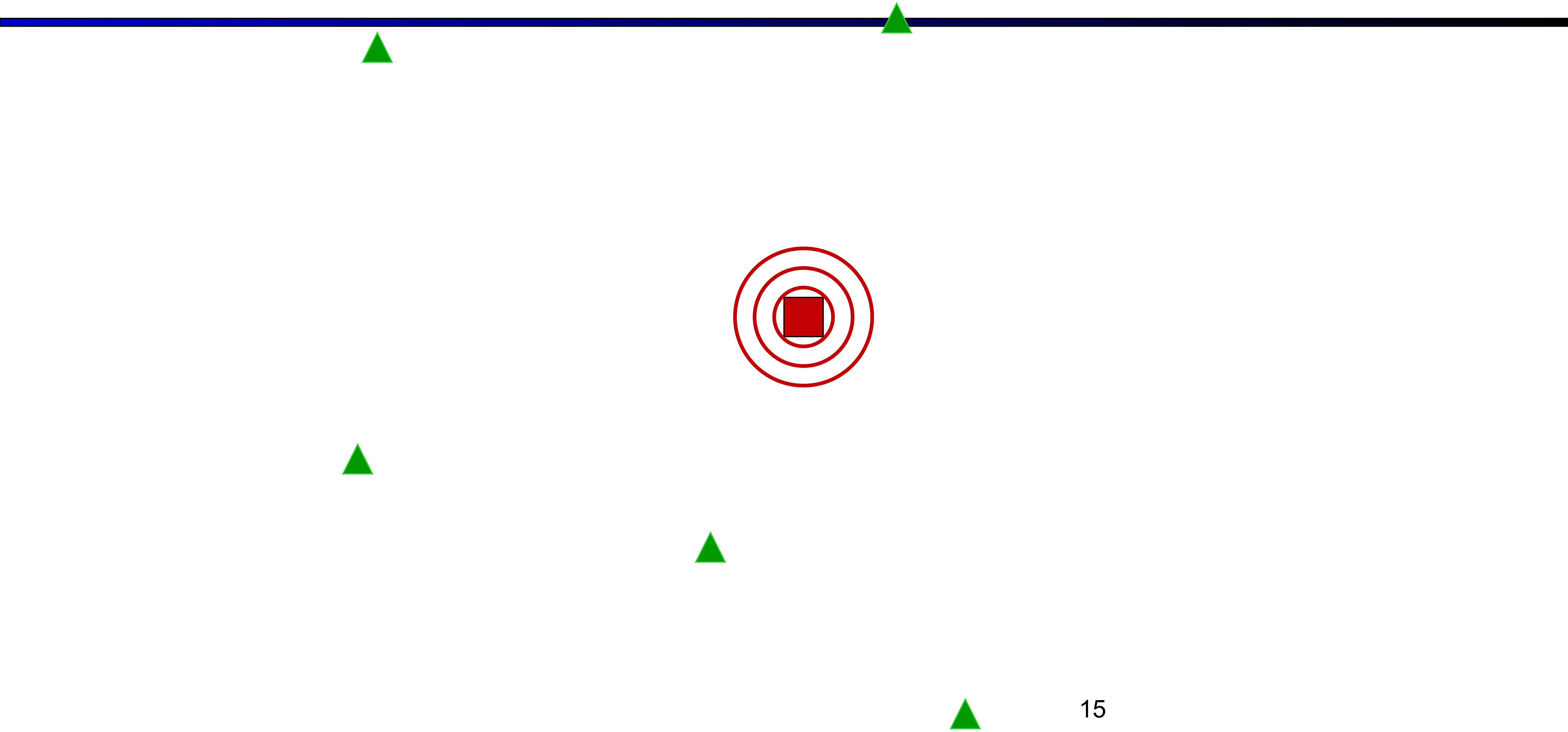
Is this a hard problem?

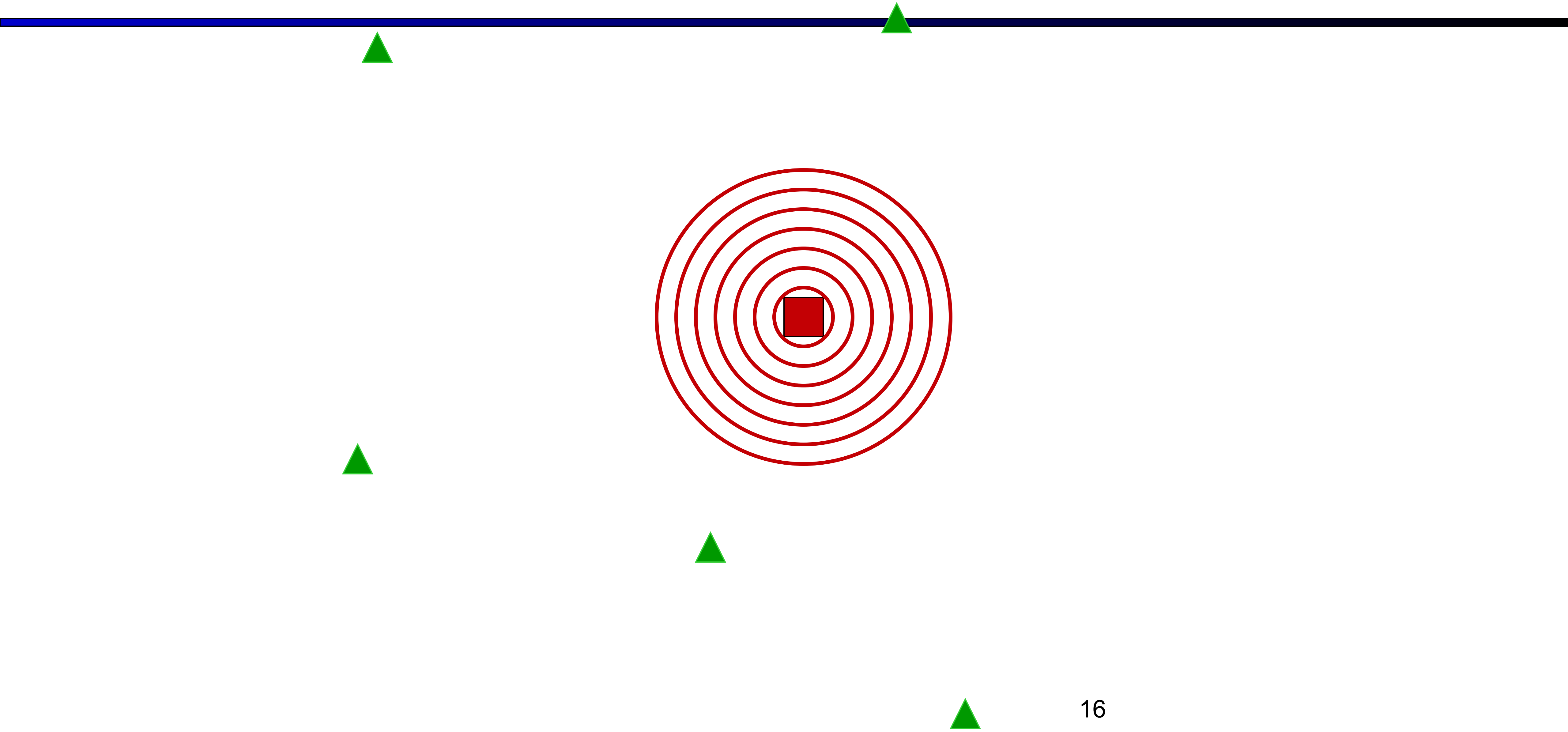
- CTBTO system developed over 10 years, $100M software (GA) plus $1B sensors + network (IMS)
  - Recall 69%, precision 47%
  - 16 human analysts correct or discard GA-proposed events, create new events, produce LEB (“ground truth”)
  - Unreliable below magnitude 4 (1kT)
    - Adversaries can “decouple” test to reduce apparent magnitude

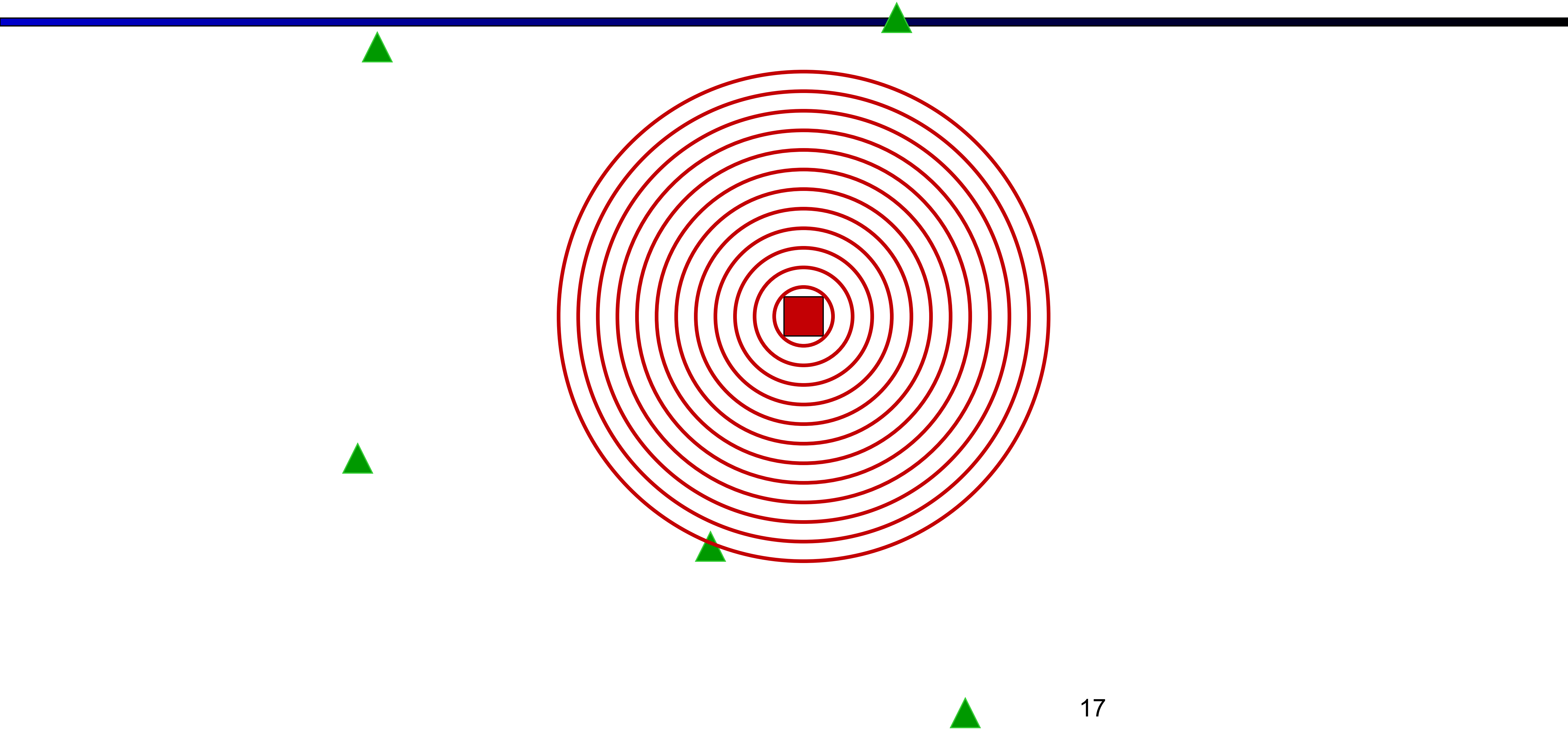
Why is this a hard problem?

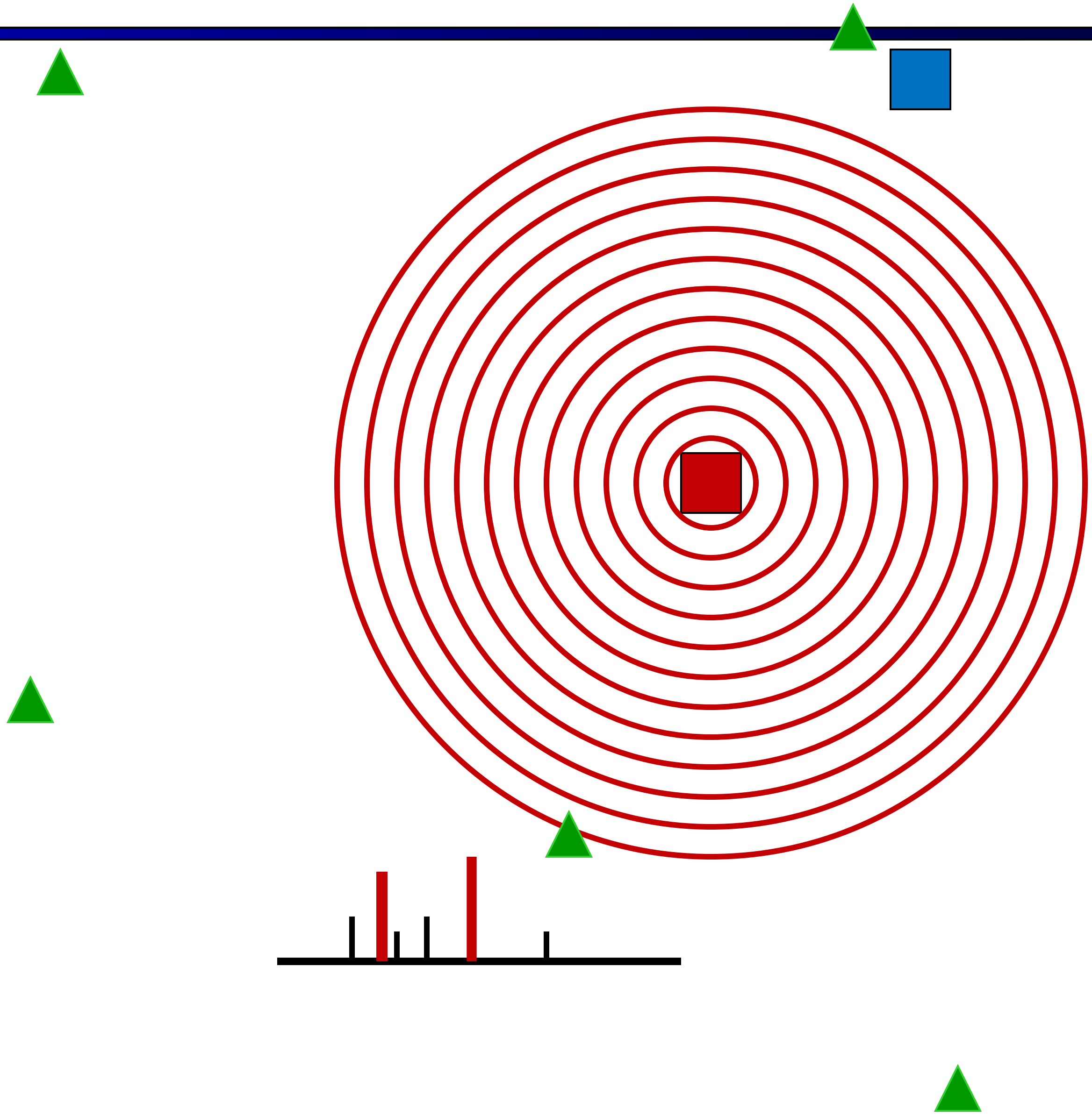
- ~10 000 "detections" per day, *90% false*
- Lots of background noise
- Events generate many different wave types ("phases") with different velocities, paths
- Signals take 15 minutes to several hours to traverse the earth, so they are all mixed up

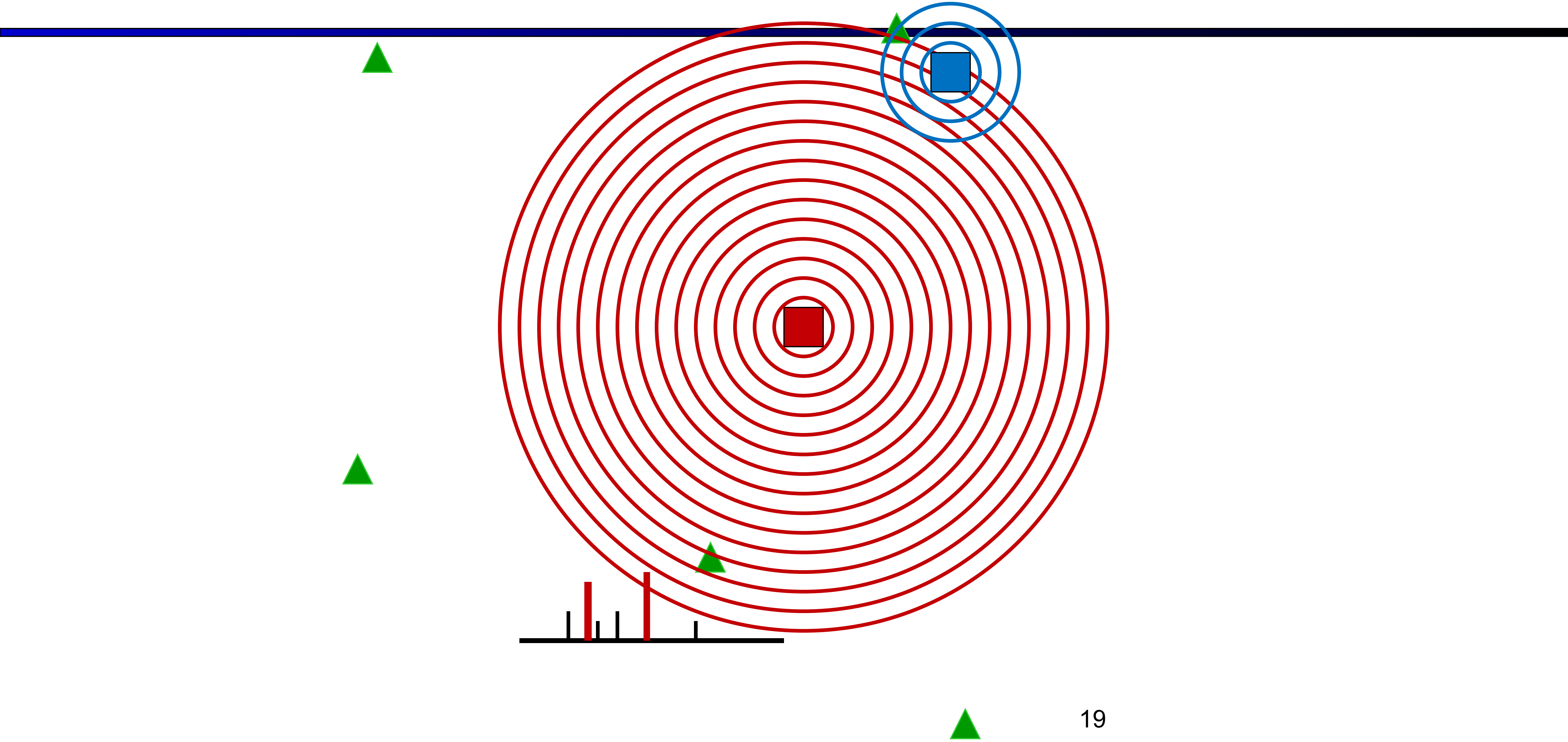
A very simple picture…

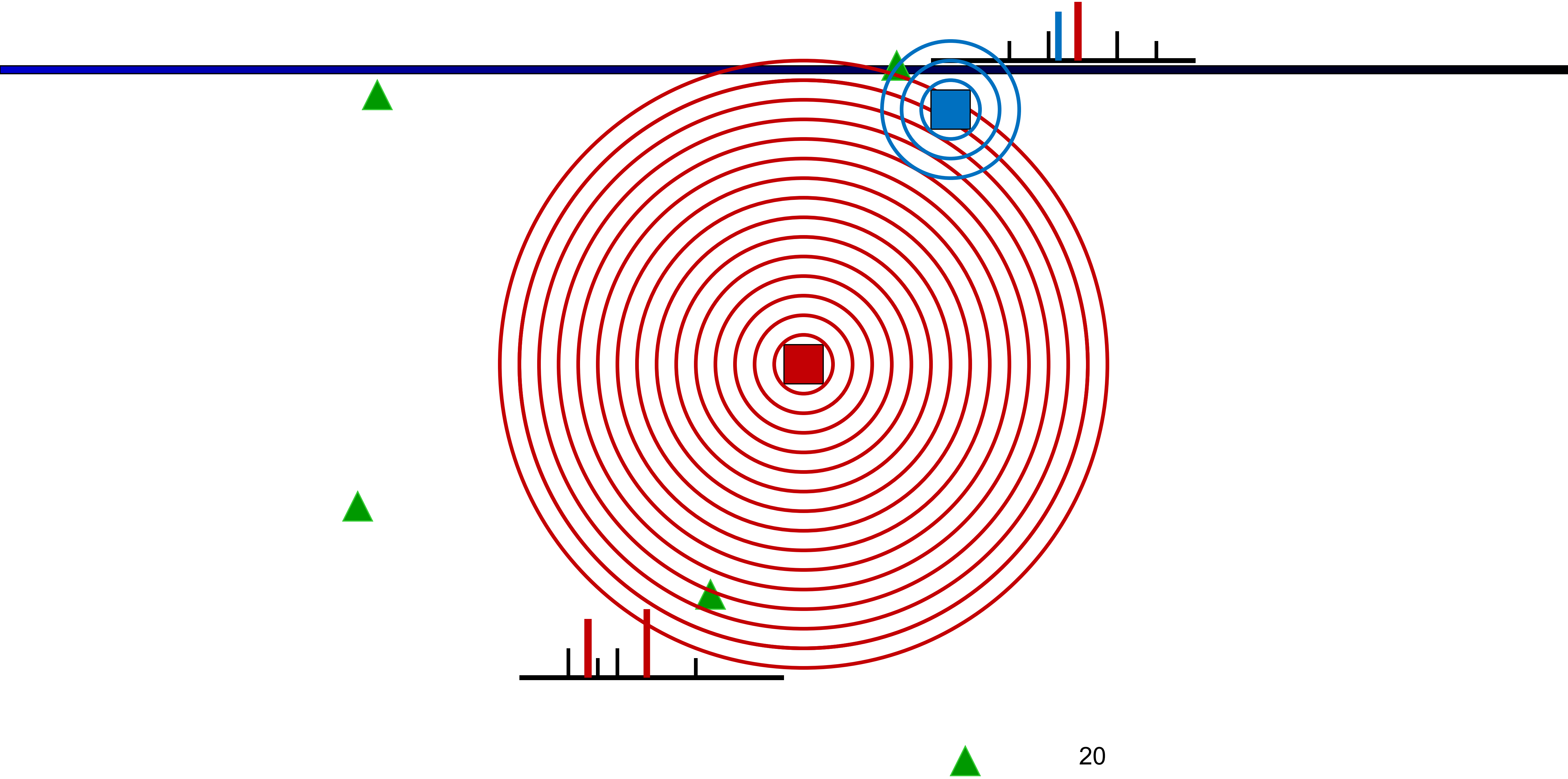


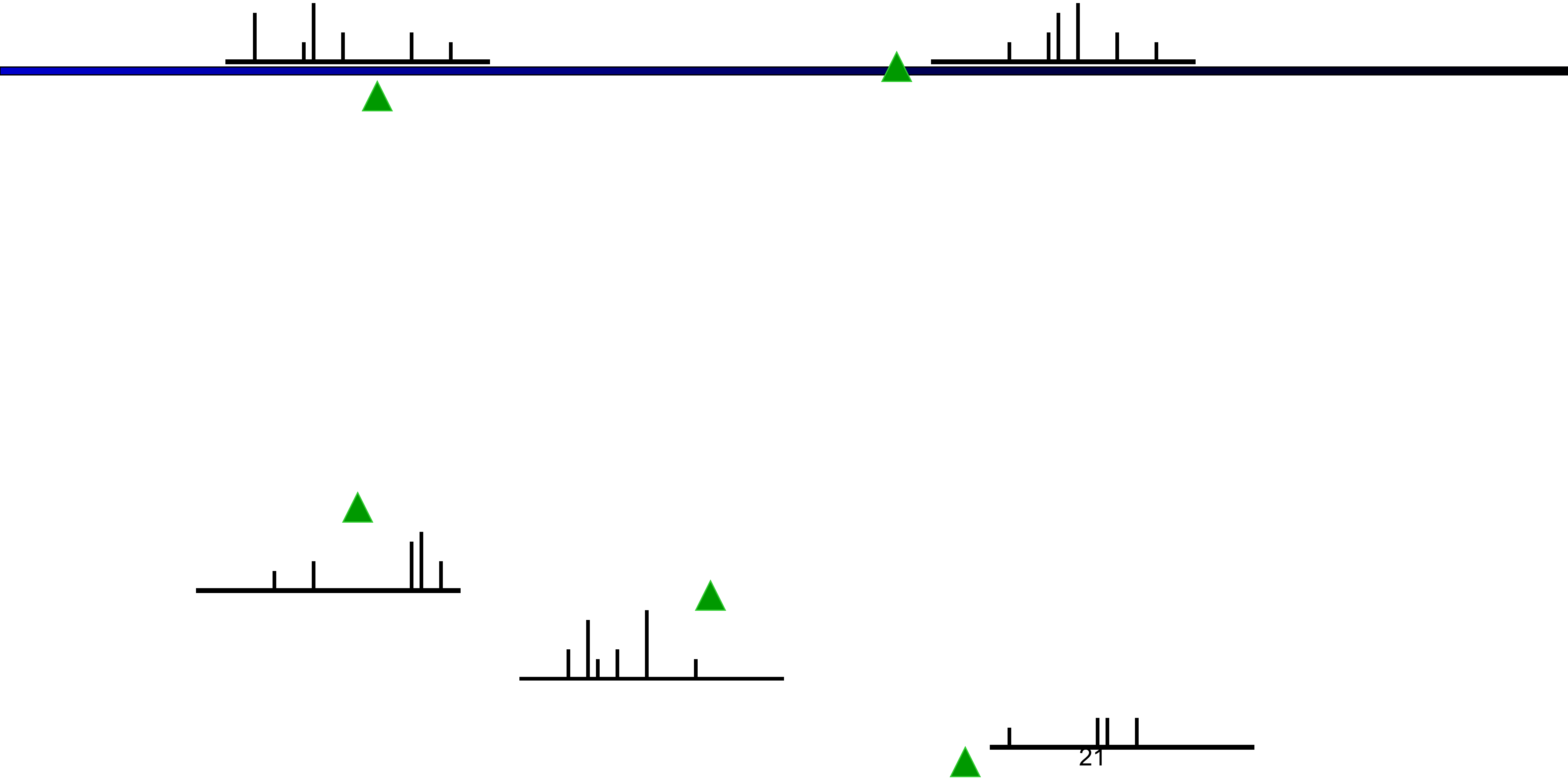












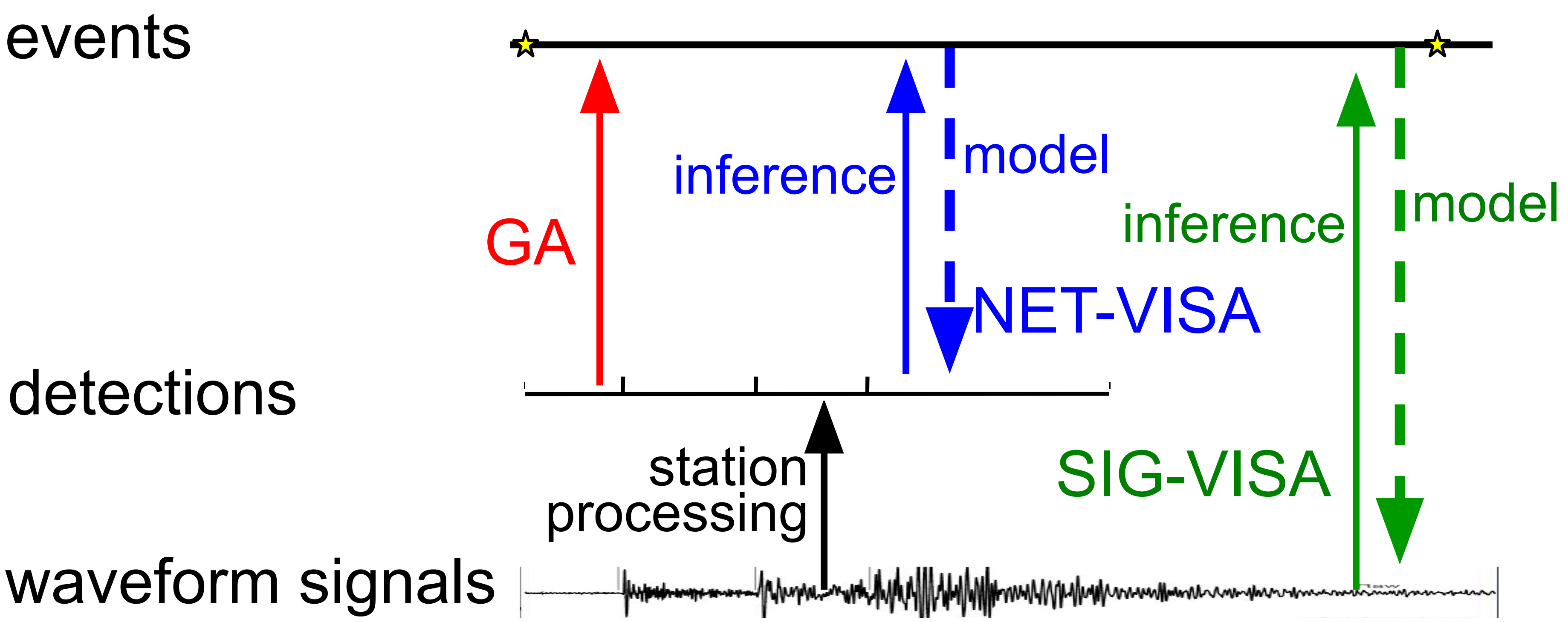
Bayesian monitoring

- Generative, model-based approach
  - P_θ(events) describes natural seismicity + nuclear (uniform)
  - $P\phi$(signals | events) describes forward model (propagation, measurement, noise, i.e., *geophysics and seismometry*)
- Estimate parameters θ and ϕ from historical data
- Bayesian inference inverts the model, given signals

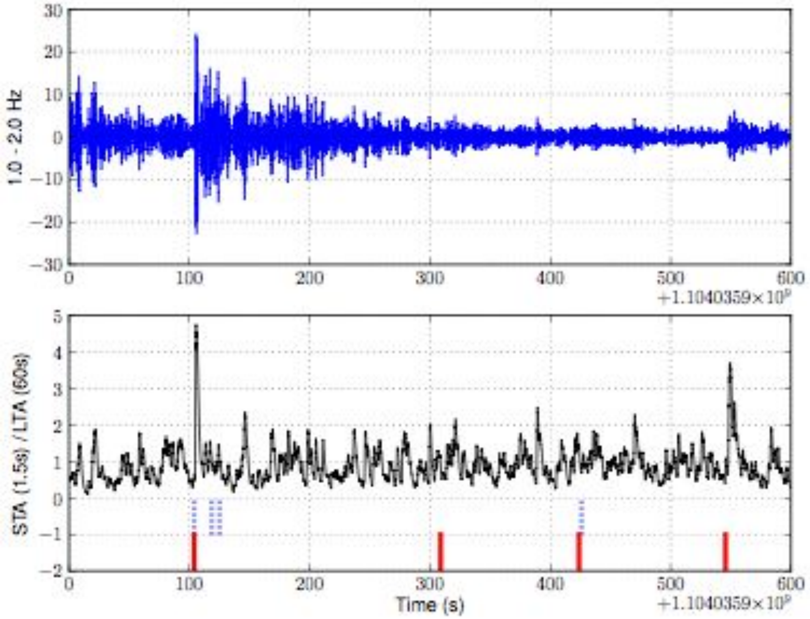
$$P(\text{events} \mid \text{signals}) \sim P\phi(\text{signals} \mid \text{events})\, P_\theta(\text{events})$$

- Advantages:
  - Correct incorporation of evidence
  - Interpretable and improvable by seismologists
  - Better models => better results

Detection-based and signal-based monitoring

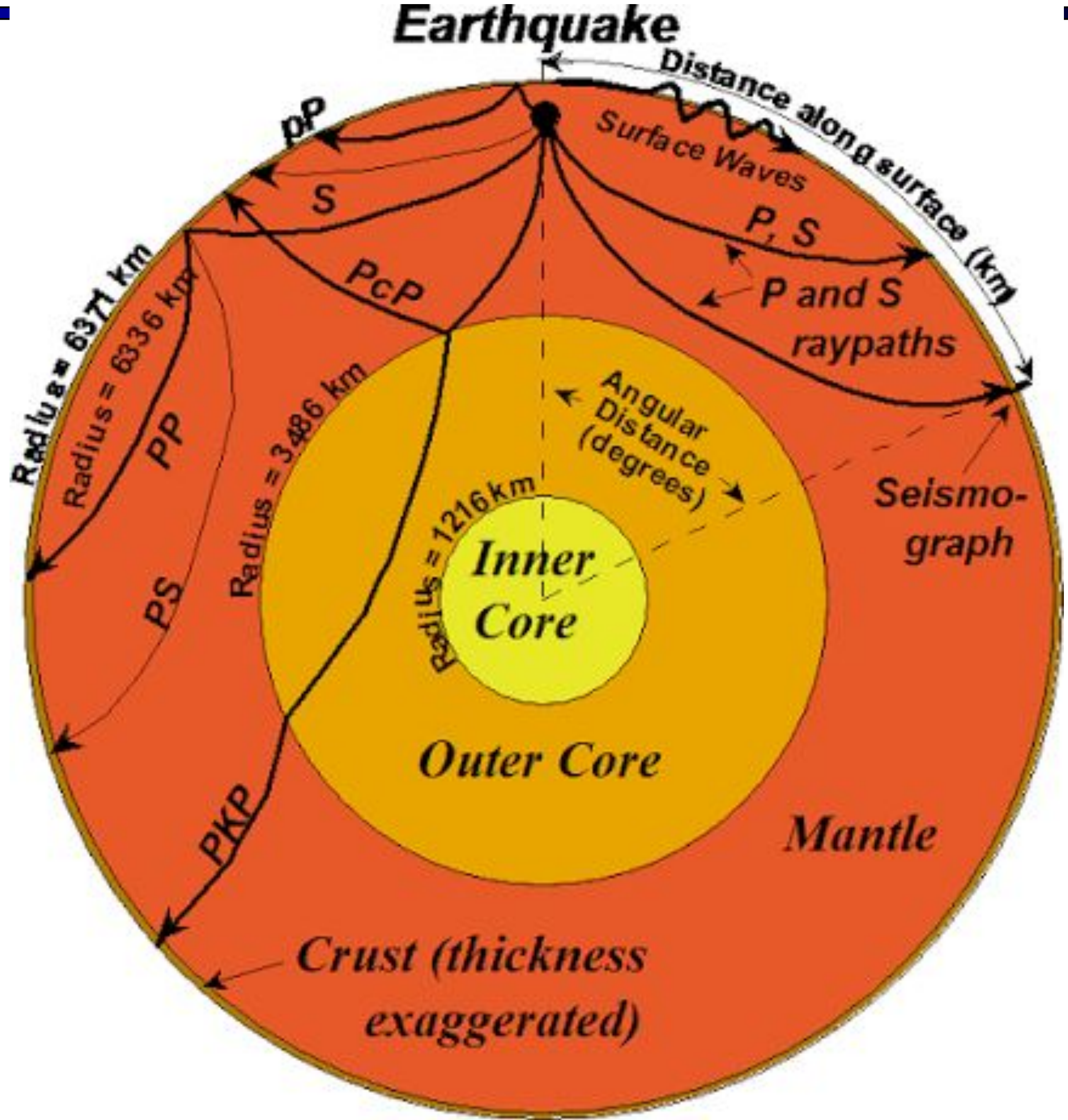


Detections ("blips")



- Local spike in signal value; attributes are:
  - *Onset time*
  - *Amplitude*
  - *Azimuth and slowness* (= direction it arrives from)
  - *Phase type* (= one of 14 distinct wave types)

Different phases: Wave types and path types



Probabilistic programming

- Combine probability theory with the universal (Turing-equivalent) expressive power of
  - Programming languages, or
  - First-order logic
    - E.g., rules of Go = 1 page, vs 1,000,000 pages for a circuit language
- System's knowledge is inspectable, compositional
- Perform inference/learning for any model, data, and query
  - Inference is traceable and rigorously correct
- Combine ***prior knowledge*** and ***data***, cumulatively

#SeismicEvents ~ Poisson[T*λ_e];
***Time(e)* ~** Uniform(0,T)
IsEarthQuake(e) ~ Bernoulli(.999);
Location(e), Depth(e) ~ if IsEarthQuake(e) then NaturalSeismicPrior() else UniformEarthDistribution(0);
Magnitude(e) ~ Exponential(log(10));
#Detections(site = s) ~ Poisson[T*λ_f(s)];
***#Detections(event=e, phase=p, station=s)* =** if IsDetected(e,p,s) then 1 else 0;
***IsDetected(e,p,s)* ~** Logistic[weights(s,p)](Magnitude(e), Depth(e), Distance(e,s))**;**
***OnsetTime(a,s)* ~** if (event(a) = null) then Uniform[0,T] else
Time(event(a)) + GeoTravelTime(Distance(event(a),s),Depth(event(a)),phase(a)) + Laplace(μ_t(s), σ_t(s))
***Amplitude(a,s)* ~** If (event(a) = null) then NoiseAmplitudeDistribution(s)
else AmplitudeModel(Magnitude(event(a)), Distance(event(a),s),Depth(event(a)),phase(a))
***Azimuth(a,s)* ~** If (event(a) = null) then Uniform(0, 360)
else GeoAzimuth(Location(event(a)),Depth(event(a)),phase(a),Site(s)) + LaplaceU(0,σ_a(s))
***Slowness(a,s)* ~** If (event(a) = null) then Uniform(0,20)
else GeoSlowness(Location(event(a)),Depth(event(a)),phase(a),Site(s)) + Laplace(0,σ_a(s))
***ObservedPhase(a,s)* ~** CategoricalPhaseModel(phase(a))

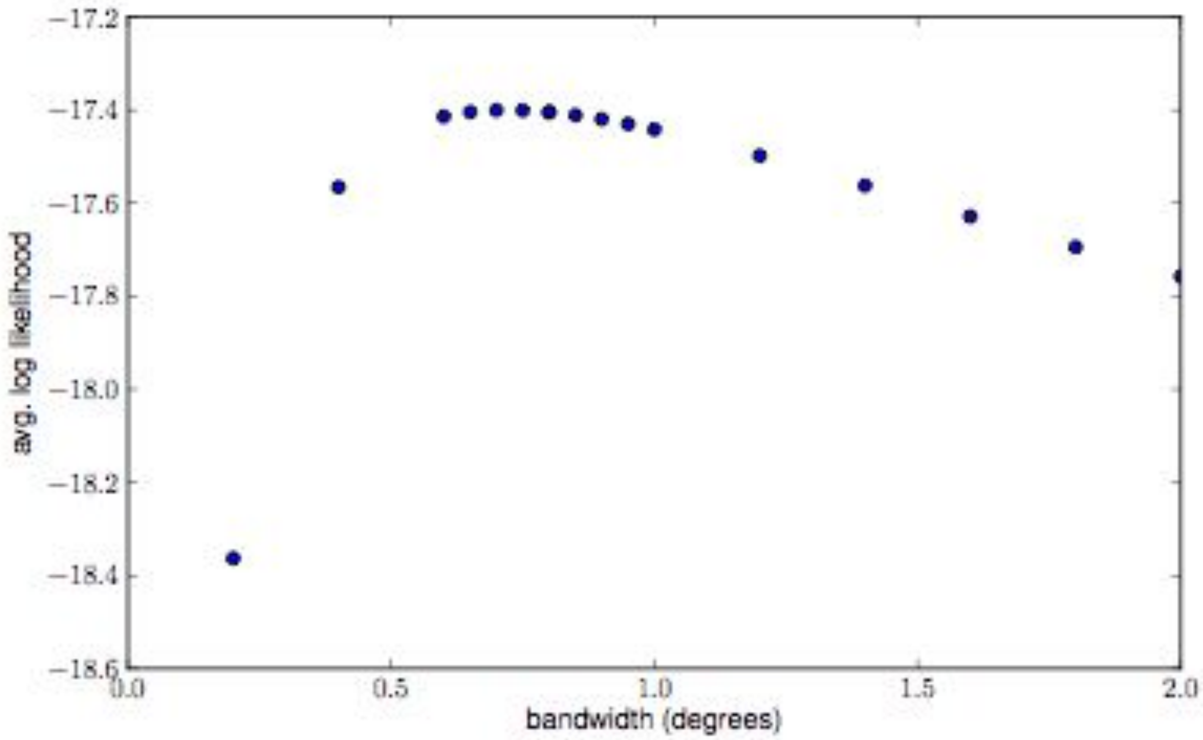
#SeismicEvents ~ Poisson[T*λ_e];
Time(e) **~** Uniform(0,T)
IsEarthQuake(e) ~ Bernoulli(.999);
Location(e), Depth(e) ~ if IsEarthQuake(e) then NaturalSeismicPrior() else UniformEarthDistribution(0);
Magnitude(e) ~ Exponential(log(10));
#Detections(site = s) ~ Poisson[T*λ_f(s)];
***#Detections(event=e, phase=p, station=s)* =** if IsDetected(e,p,s) then 1 else 0;
IsDetected(e,p,s) **~** Logistic[weights(s,p)](Magnitude(e), Depth(e), Distance(e,s))**;**
OnsetTime(a,s) **~** if (event(a) = null) then Uniform[0,T] else
Time(event(a)) + GeoTravelTime(Distance(event(a),s),Depth(event(a)),phase(a)) + Laplace(μ_t(s), σ_t(s))
Amplitude(a,s) **~** If (event(a) = null) then NoiseAmplitudeDistribution(s)
else AmplitudeModel(Magnitude(event(a)), Distance(event(a),s),Depth(event(a)),phase(a))
Azimuth(a,s) **~** If (event(a) = null) then Uniform(0, 360)
else GeoAzimuth(Location(event(a)),Depth(event(a)),phase(a),Site(s)) + LaplaceU(0,σ_a(s))
Slowness(a,s) **~** If (event(a) = null) then Uniform(0,20)
else GeoSlowness(Location(event(a)),Depth(event(a)),phase(a),Site(s)) + Laplace(0,σ_a(s))
ObservedPhase(a,s) **~** CategoricalPhaseModel(phase(a))

Estimating the location prior

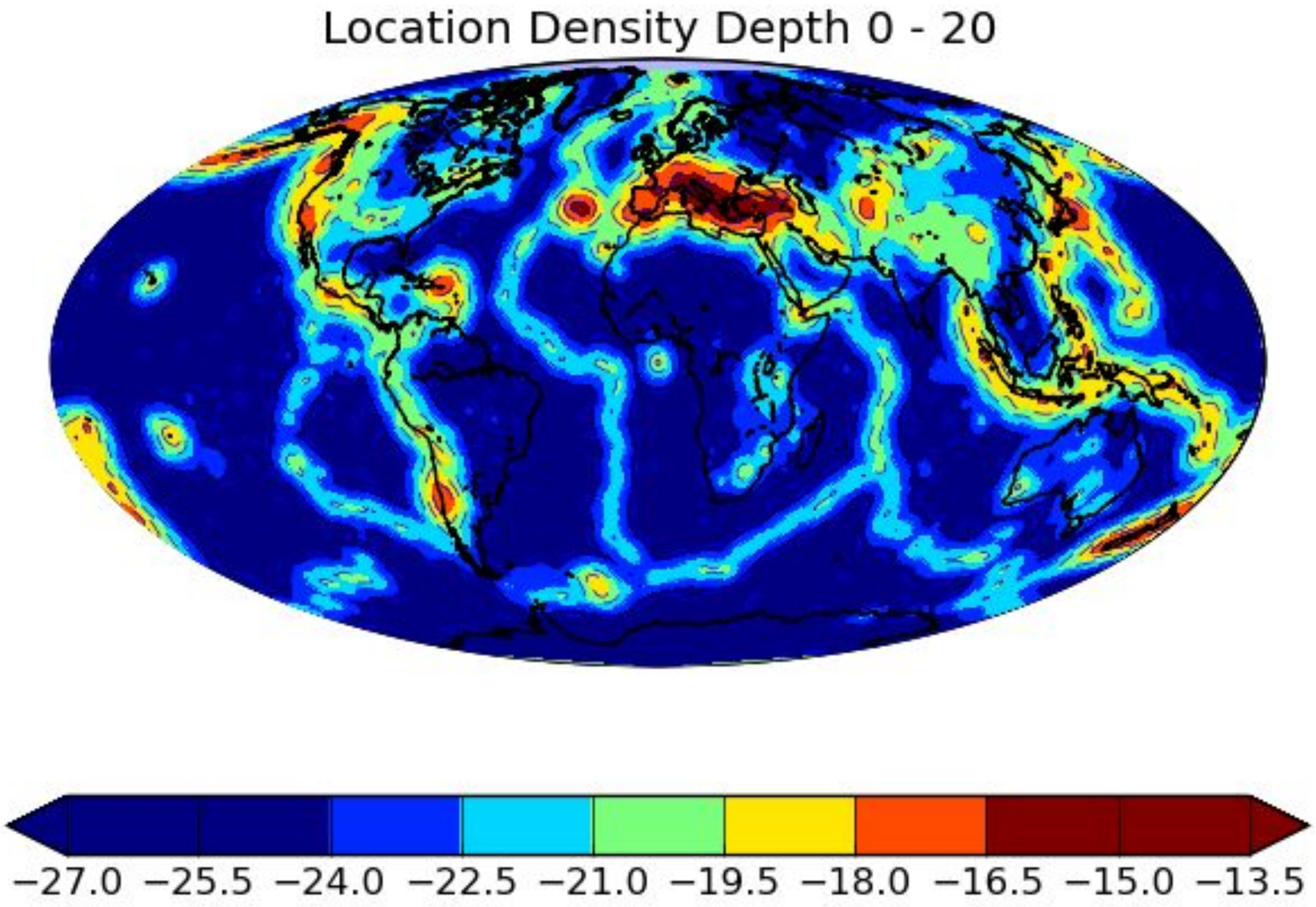
- Kernel density estimate plus uniform component:

$$P_{\theta,l}(e_l) = .001\frac{1}{4\pi R^2} + .999\frac{1}{H}\sum_{h=1}^{H} K_{b,g_l^h}(e_l) \qquad K_{b,x}(y) = \frac{1+1/b^2}{2\pi R^2}\frac{\exp\left(-\Delta_{xy}/b\right)}{1+\exp\left(-\pi/b\right)}$$

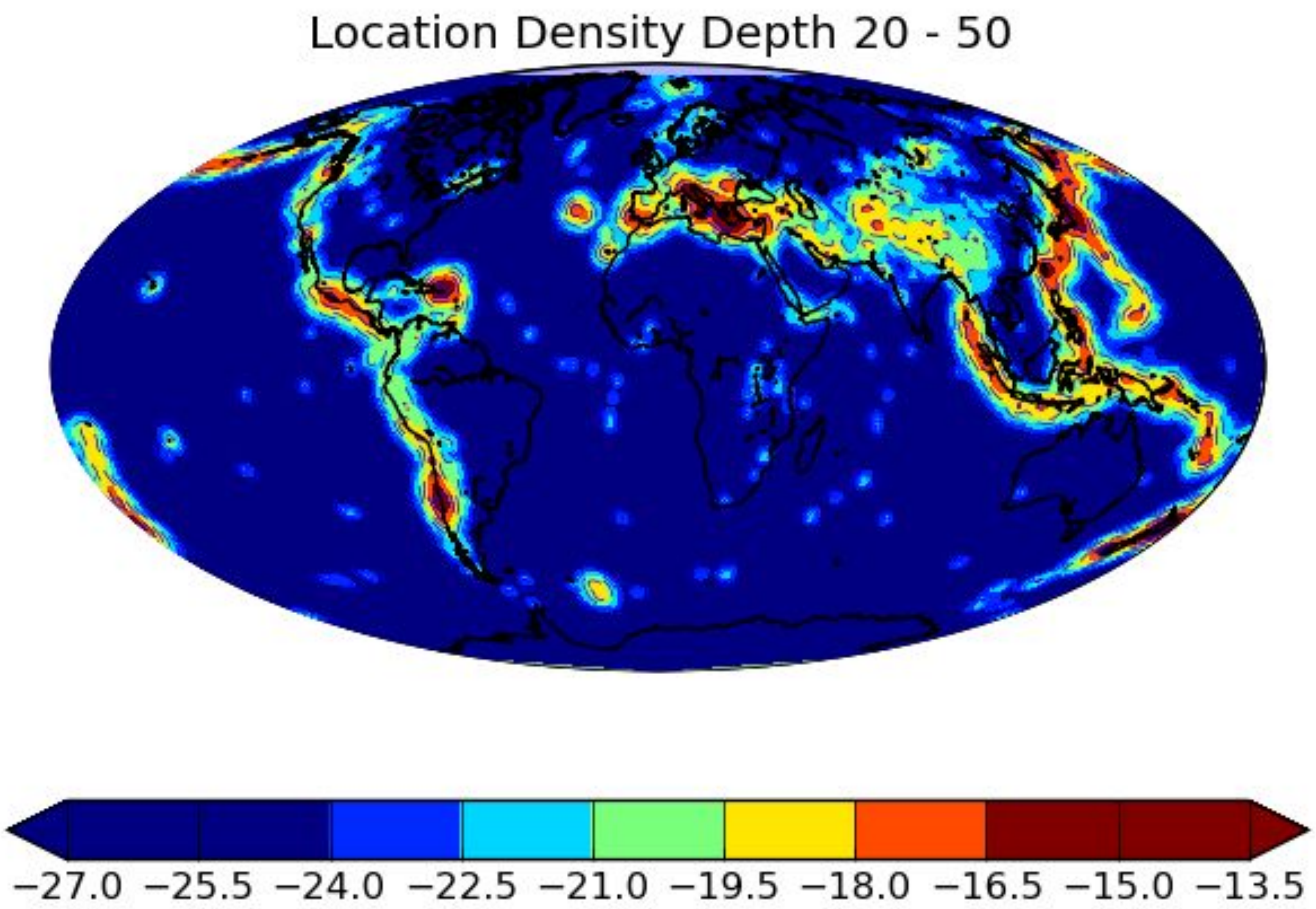
- Kernel width *b* estimated by LOOCV:



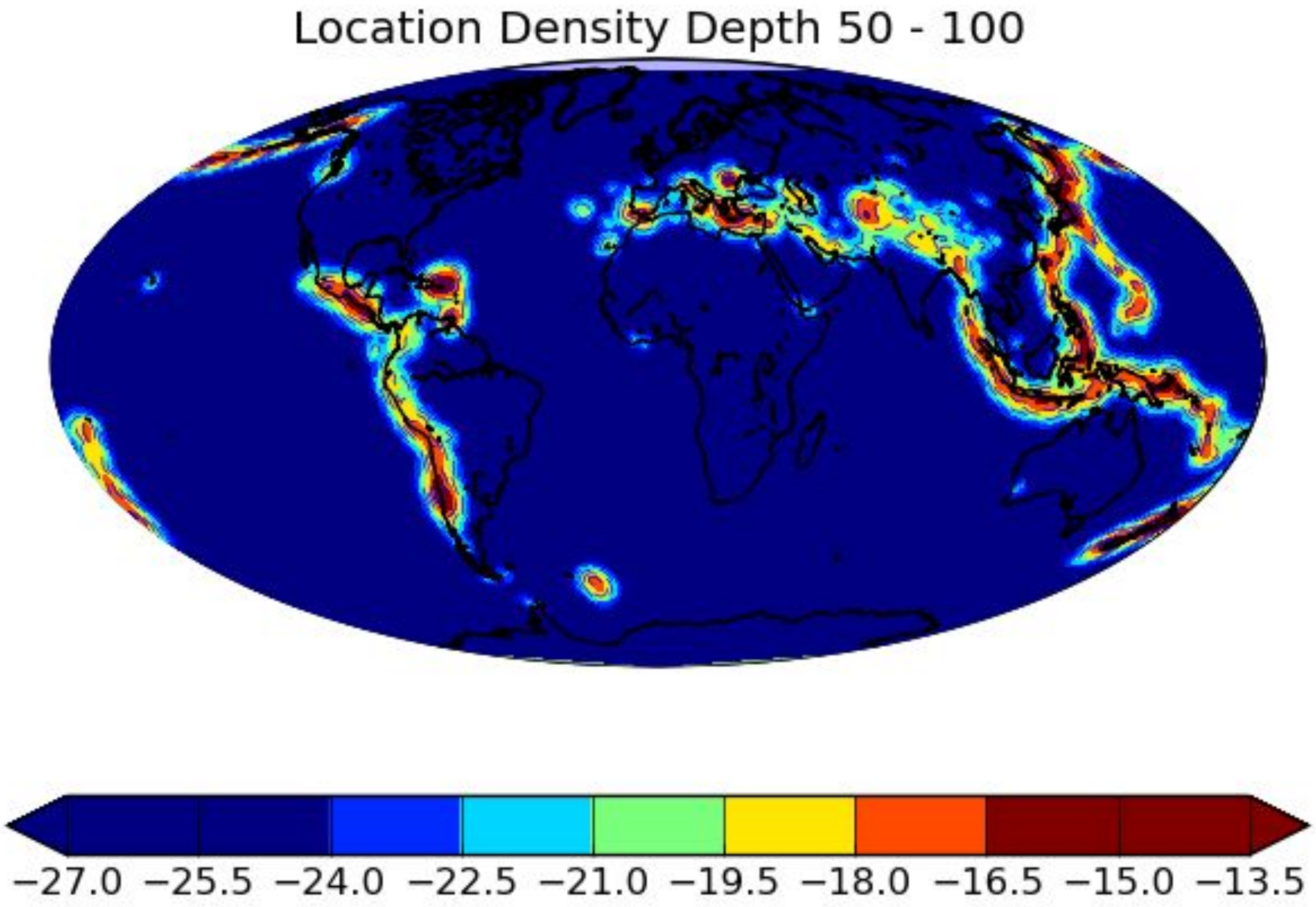
Depth 0 – 20 km


Location Density Depth 0 - 20
−27.0 −25.5 −24.0 −22.5 −21.0 −19.5 −18.0 −16.5 −15.0 −13.5

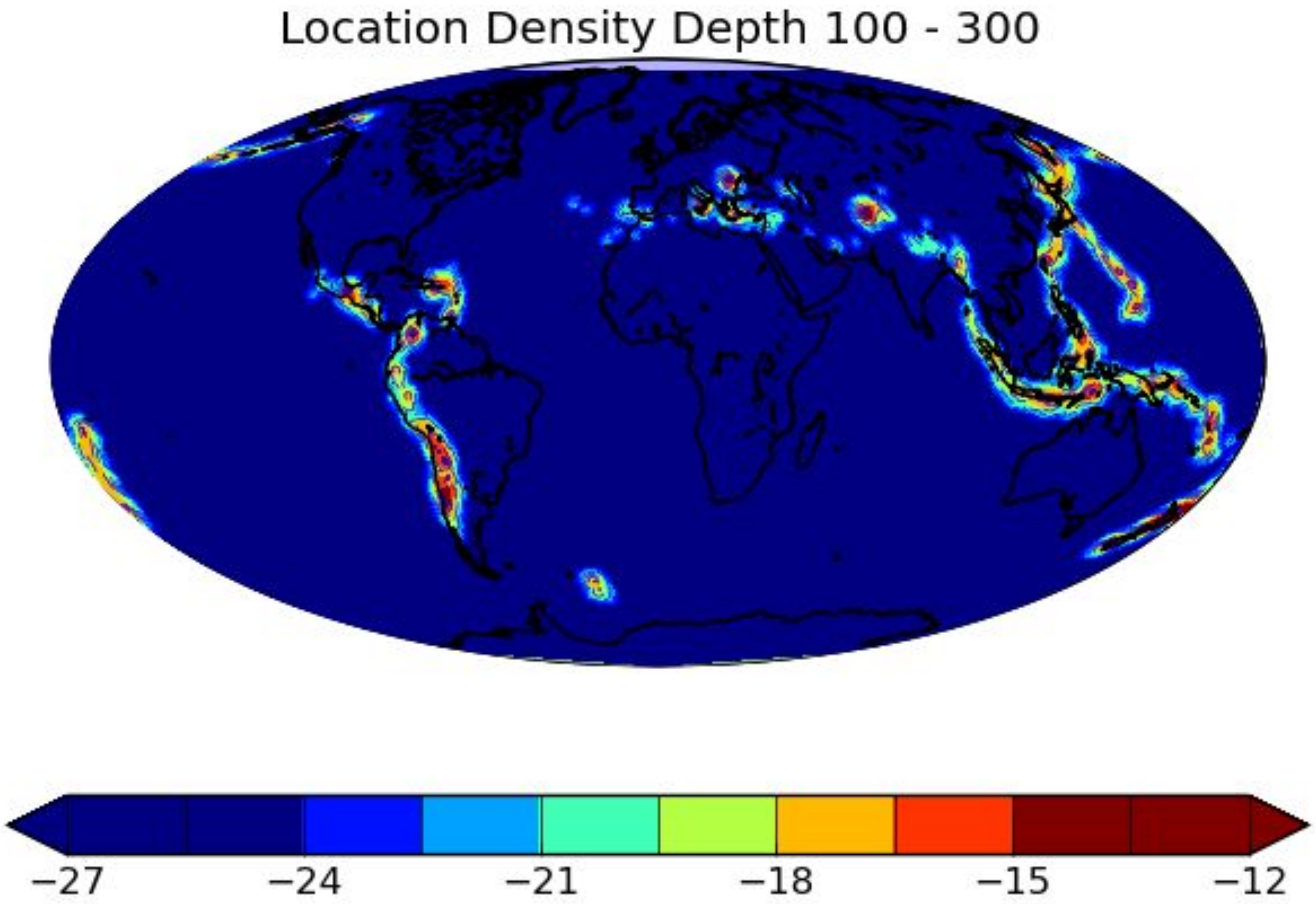
Depth 20 - 50


Location Density Depth 20 - 50
−27.0 −25.5 −24.0 −22.5 −21.0 −19.5 −18.0 −16.5 −15.0 −13.5

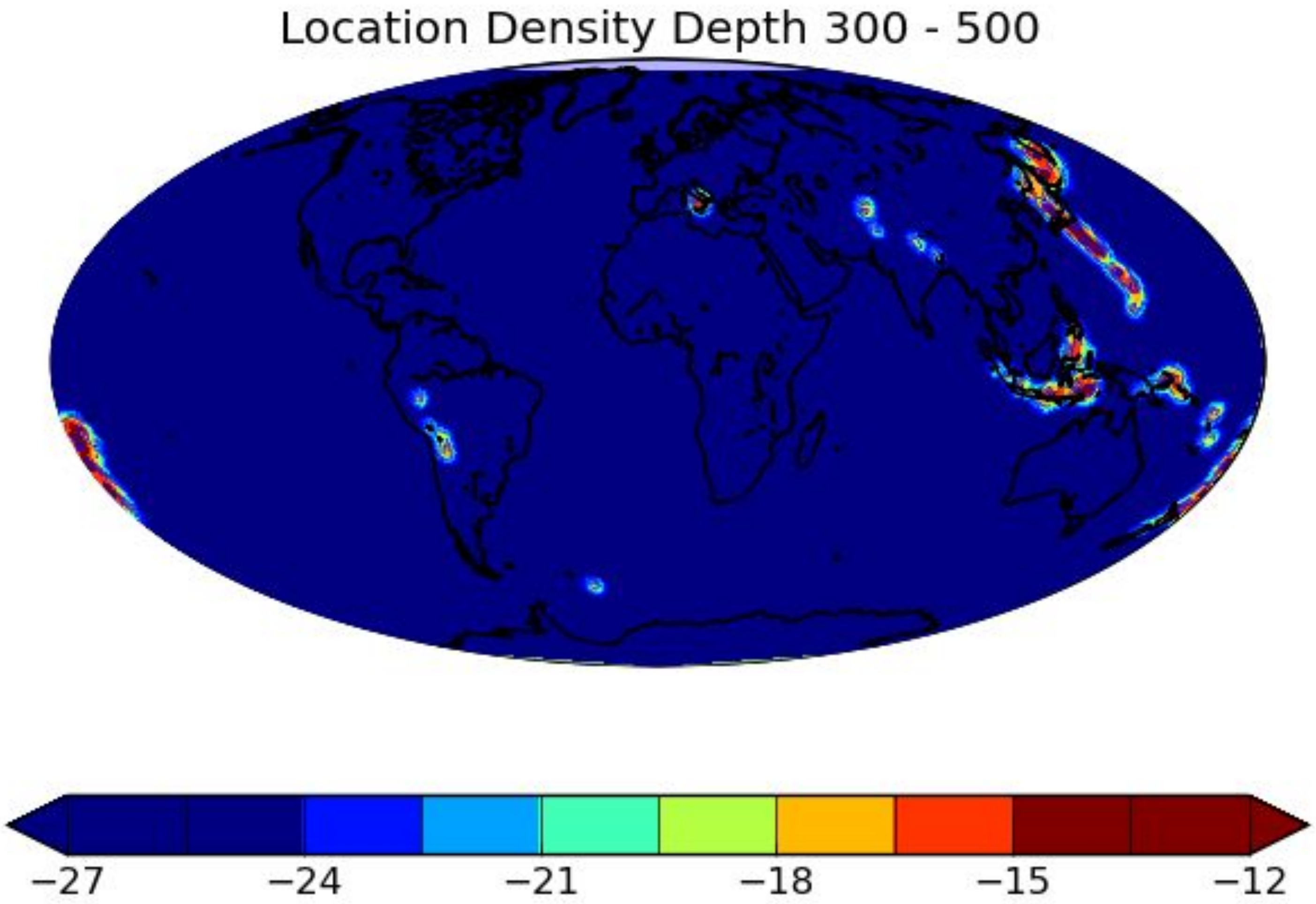
Depth 50 - 100


Location Density Depth 50 - 100
−27.0 −25.5 −24.0 −22.5 −21.0 −19.5 −18.0 −16.5 −15.0 −13.5

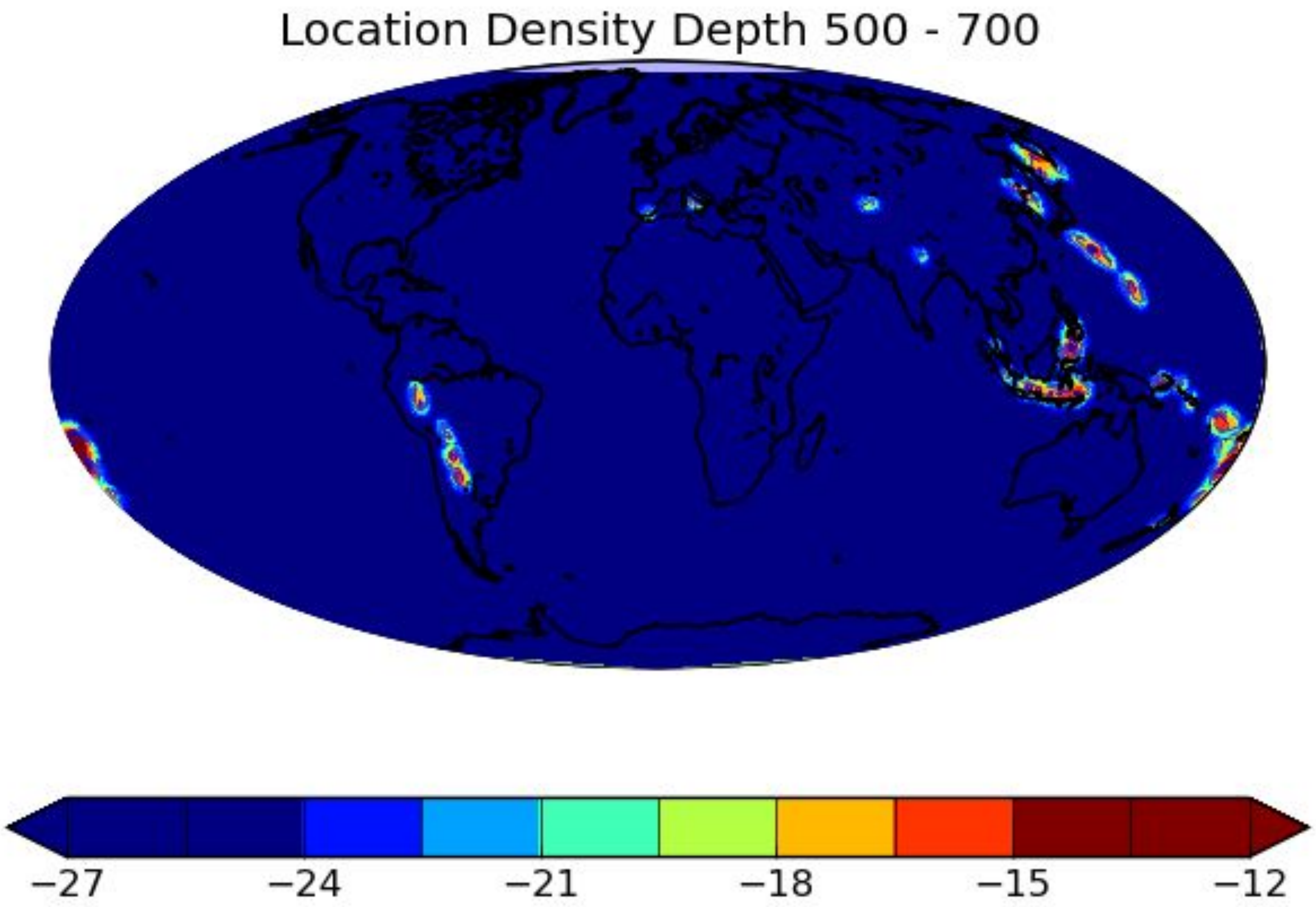
Depth 100 - 300



Depth 300 - 500



Depth 500 - 700



#SeismicEvents ~ Poisson[T*λ_e];
***Time(e)* ~** Uniform(0,T)
IsEarthQuake(e) ~ Bernoulli(.999);
Location(e), Depth(e) ~ if IsEarthQuake(e) then NaturalSeismicPrior() else UniformEarthDistribution(0);
Magnitude(e) ~ Exponential(log(10));
#Detections(site = s) ~ Poisson[T*λ_f(s)];
***#Detections(event=e, phase=p, station=s)* =** if IsDetected(e,p,s) then 1 else 0;
***IsDetected(e,p,s)* ~** Logistic[weights(s,p)](Magnitude(e), Depth(e), Distance(e,s))**;**
***OnsetTime(a,s)* ~** if (event(a) = null) then Uniform[0,T] else
Time(event(a)) + GeoTravelTime(Distance(event(a),s),Depth(event(a)),phase(a)) + Laplace(μ_t(s), σ_t(s))
***Amplitude(a,s)* ~** If (event(a) = null) then NoiseAmplitudeDistribution(s)
else AmplitudeModel(Magnitude(event(a)), Distance(event(a),s),Depth(event(a)),phase(a))
***Azimuth(a,s)* ~** If (event(a) = null) then Uniform(0, 360)
else GeoAzimuth(Location(event(a)),Depth(event(a)),phase(a),Site(s)) + LaplaceU(0,σ_a(s))
***Slowness(a,s)* ~** If (event(a) = null) then Uniform(0,20)
else GeoSlowness(Location(event(a)),Depth(event(a)),phase(a),Site(s)) + Laplace(0,σ_a(s))
***ObservedPhase(a,s)* ~** CategoricalPhaseModel(phase(a))

Detection probability model

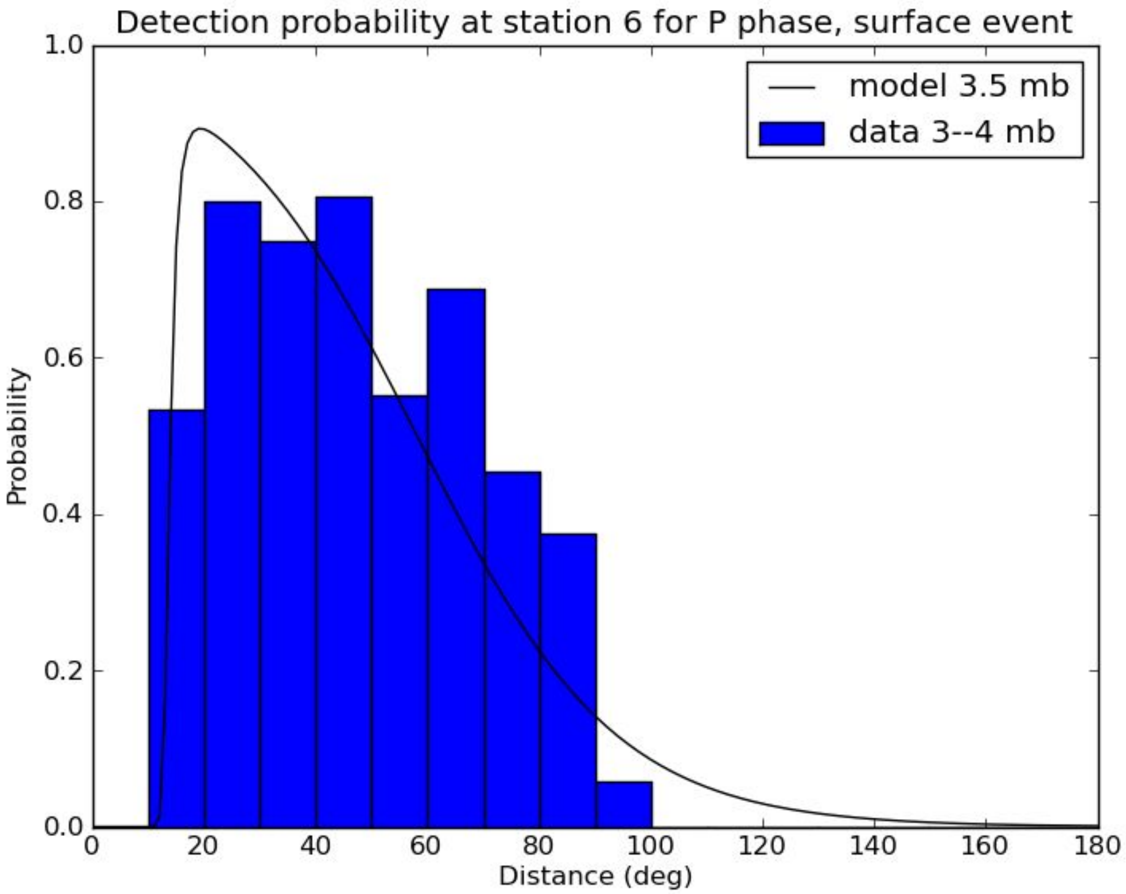
- Detection probability $P_{\phi,d}^{jk}(e_i)$ for phase *j* of event e_i at station *k* is given by a logistic regression with input features derived from event magnitude e_m, depth e_d, and distance Δ_{ik}:

$$\log\left(\frac{P_{\phi,d}^{jk}(e^i)}{1 - P_{\phi,d}^{jk}(e^i)}\right) = \sum_{w \in \mathcal{F}_d} \mu_d^{wjk} \cdot w(e_m^i, e_d^i, \Delta_{ik})$$

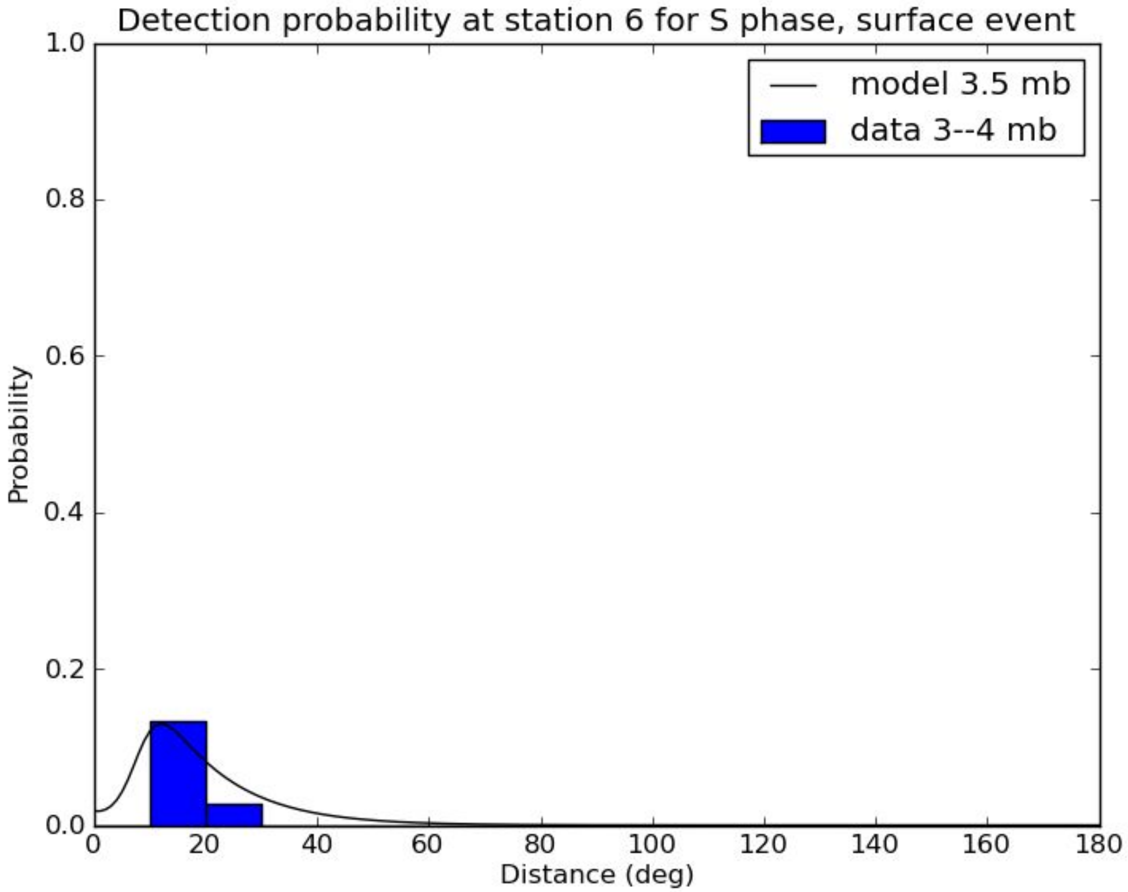
- Station-specific parameters μ_d^{wjk} estimated by hierarchical Bayesian model to allow for shrinkage towards global mean for stations with sparse data

Detection probability as a function of distance (station 6, magnitude 3.5)

P phase

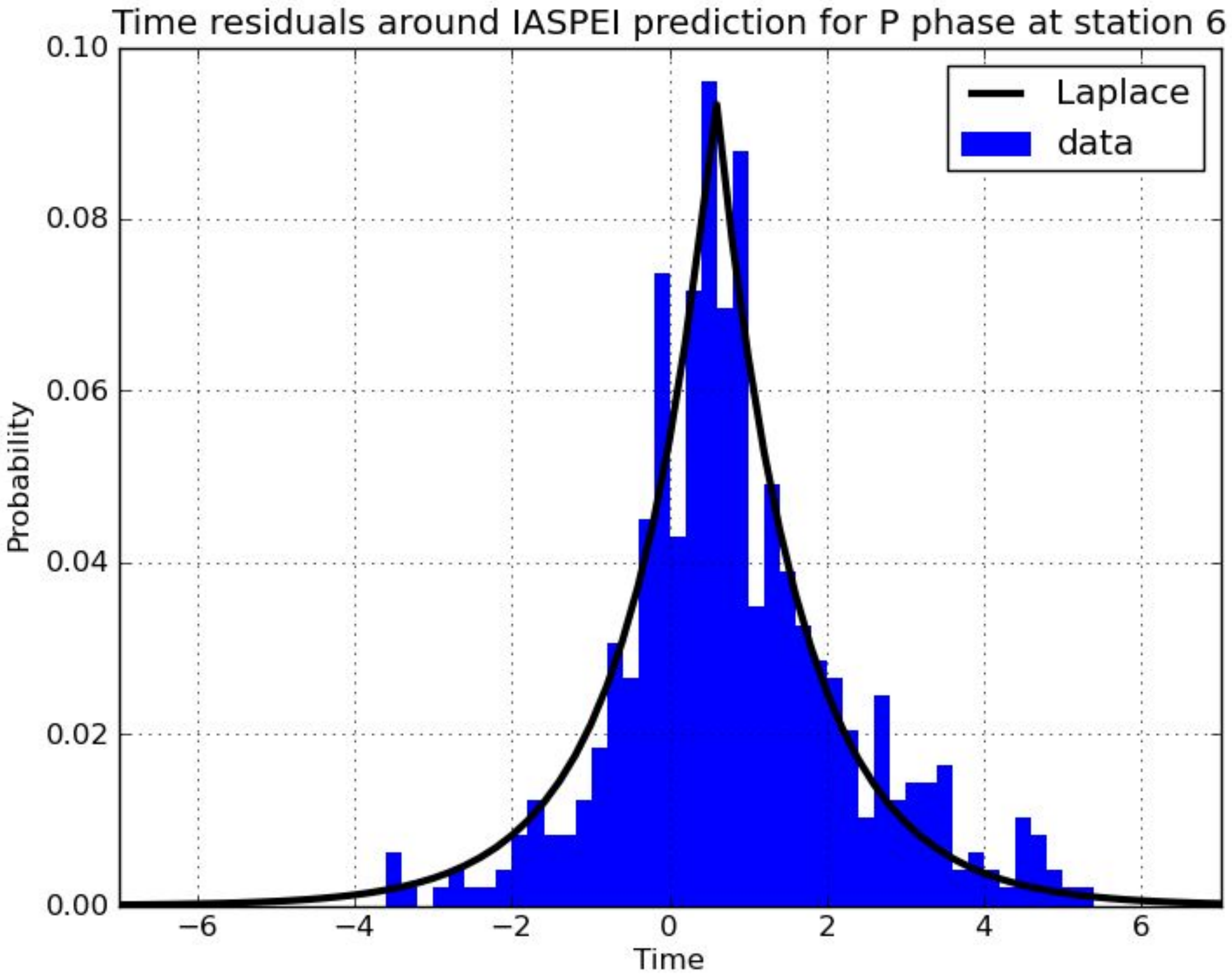


S phase



#SeismicEvents ~ Poisson[T*λ_e];
***Time(e)* ~** Uniform(0,T)
IsEarthQuake(e) ~ Bernoulli(.999);
Location(e), Depth(e) ~ if IsEarthQuake(e) then NaturalSeismicPrior() else UniformEarthDistribution(0);
Magnitude(e) ~ Exponential(log(10));
#Detections(site = s) ~ Poisson[T*λ_f(s)];
***#Detections(event=e, phase=p, station=s)* =** if IsDetected(e,p,s) then 1 else 0;
***IsDetected(e,p,s)* ~** Logistic[weights(s,p)](Magnitude(e), Depth(e), Distance(e,s))**;**
***OnsetTime(a,s)* ~** if (event(a) = null) then Uniform[0,T] else
Time(event(a)) + GeoTravelTime(Distance(event(a),s),Depth(event(a)),phase(a)) + Laplace(μ_t(s), σ_t(s))
***Amplitude(a,s)* ~** If (event(a) = null) then NoiseAmplitudeDistribution(s)
else AmplitudeModel(Magnitude(event(a)), Distance(event(a),s),Depth(event(a)),phase(a))
***Azimuth(a,s)* ~** If (event(a) = null) then Uniform(0, 360)
else GeoAzimuth(Location(event(a)),Depth(event(a)),phase(a),Site(s)) + LaplaceU(0,σ_a(s))
***Slowness(a,s)* ~** If (event(a) = null) then Uniform(0,20)
else GeoSlowness(Location(event(a)),Depth(event(a)),phase(a),Site(s)) + Laplace(0,σ_a(s))
***ObservedPhase(a,s)* ~** CategoricalPhaseModel(phase(a))

Travel-time residual (station 6)



#SeismicEvents ~ Poisson[T*λ_e];

Time(e) ~ Uniform(0,T)

IsEarthQuake(e) ~ Bernoulli(.999);

Location(e), Depth(e) ~ if IsEarthQuake(e) then NaturalSeismicPrior() else UniformEarthDistribution(0);

Magnitude(e) ~ Exponential(log(10));

#Detections(site = s) ~ Poisson[T*λ_f(s)];

#Detections(event=e, phase=p, station=s) = if IsDetected(e,p,s) then 1 else 0;

IsDetected(e,p,s) ~ Logistic[weights(s,p)](Magnitude(e), Depth(e), Distance(e,s));

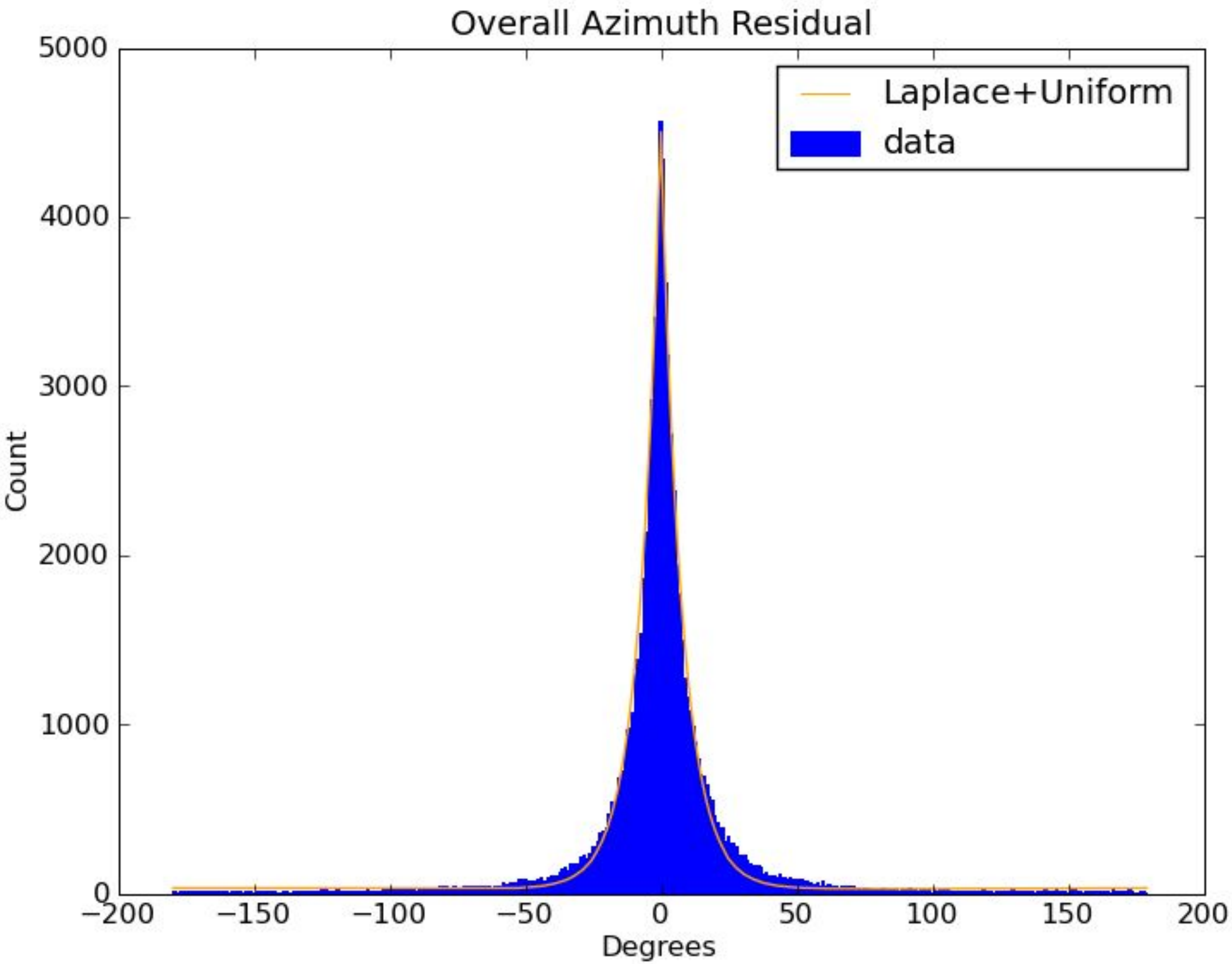
OnsetTime(a,s) ~ if (event(a) = null) then Uniform[0,T] else
Time(event(a)) + GeoTravelTime(Distance(event(a),s),Depth(event(a)),phase(a)) + Laplace(μ_t(s), σ_t(s))

Amplitude(a,s) ~ If (event(a) = null) then NoiseAmplitudeDistribution(s)
else AmplitudeModel(Magnitude(event(a)), Distance(event(a),s),Depth(event(a)),phase(a))

Azimuth(a,s) ~ If (event(a) = null) then Uniform(0, 360)
else GeoAzimuth(Location(event(a)),Depth(event(a)),phase(a),Site(s)) + LaplaceU(0,σ_a(s))

Slowness(a,s) ~ If (event(a) = null) then Uniform(0,20)
else GeoSlowness(Location(event(a)),Depth(event(a)),phase(a),Site(s)) + Laplace(0,σ_a(s))

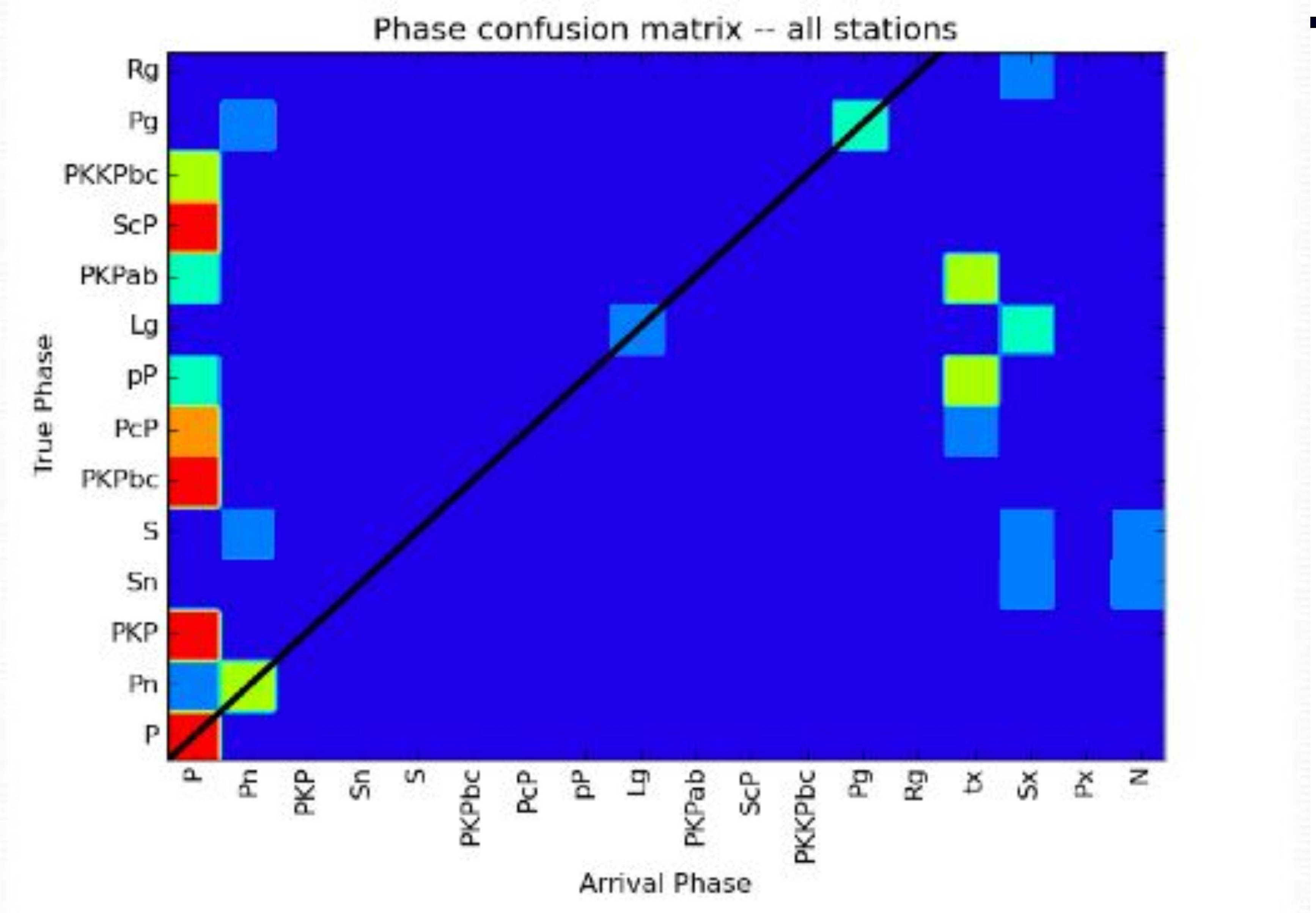
ObservedPhase(a,s) ~ CategoricalPhaseModel(phase(a))



Overall Azimuth Residual
Laplace+Uniform
data
Count
Degrees
5000
4000
3000
2000
1000
0
−200
−150
−100
−50
0
50
100
150
200

#SeismicEvents ~ Poisson[T*λ_e];
Time(e) ~ Uniform(0,T)
IsEarthQuake(e) ~ Bernoulli(.999);
Location(e), Depth(e) ~ if IsEarthQuake(e) then NaturalSeismicPrior() else UniformEarthDistribution(0);
Magnitude(e) ~ Exponential(log(10));
#Detections(site = s) ~ Poisson[T*λ_f(s)];
#Detections(event=e, phase=p, station=s) = if IsDetected(e,p,s) then 1 else 0;
IsDetected(e,p,s) ~ Logistic[weights(s,p)](Magnitude(e), Depth(e), Distance(e,s));
OnsetTime(a,s) ~ if (event(a) = null) then Uniform[0,T] else
Time(event(a)) + GeoTravelTime(Distance(event(a),s),Depth(event(a)),phase(a)) + Laplace(μ_t(s), σ_t(s))
Amplitude(a,s) ~ If (event(a) = null) then NoiseAmplitudeDistribution(s)
else AmplitudeModel(Magnitude(event(a)), Distance(event(a),s),Depth(event(a)),phase(a))
Azimuth(a,s) ~ If (event(a) = null) then Uniform(0, 360)
else GeoAzimuth(Location(event(a)),Depth(event(a)),phase(a),Site(s)) + LaplaceU(0,σ_a(s))
Slowness(a,s) ~ If (event(a) = null) then Uniform(0,20)
else GeoSlowness(Location(event(a)),Depth(event(a)),phase(a),Site(s)) + Laplace(0,σ_a(s))
ObservedPhase(a,s) ~ CategoricalPhaseModel(phase(a))

Phase confusion matrix



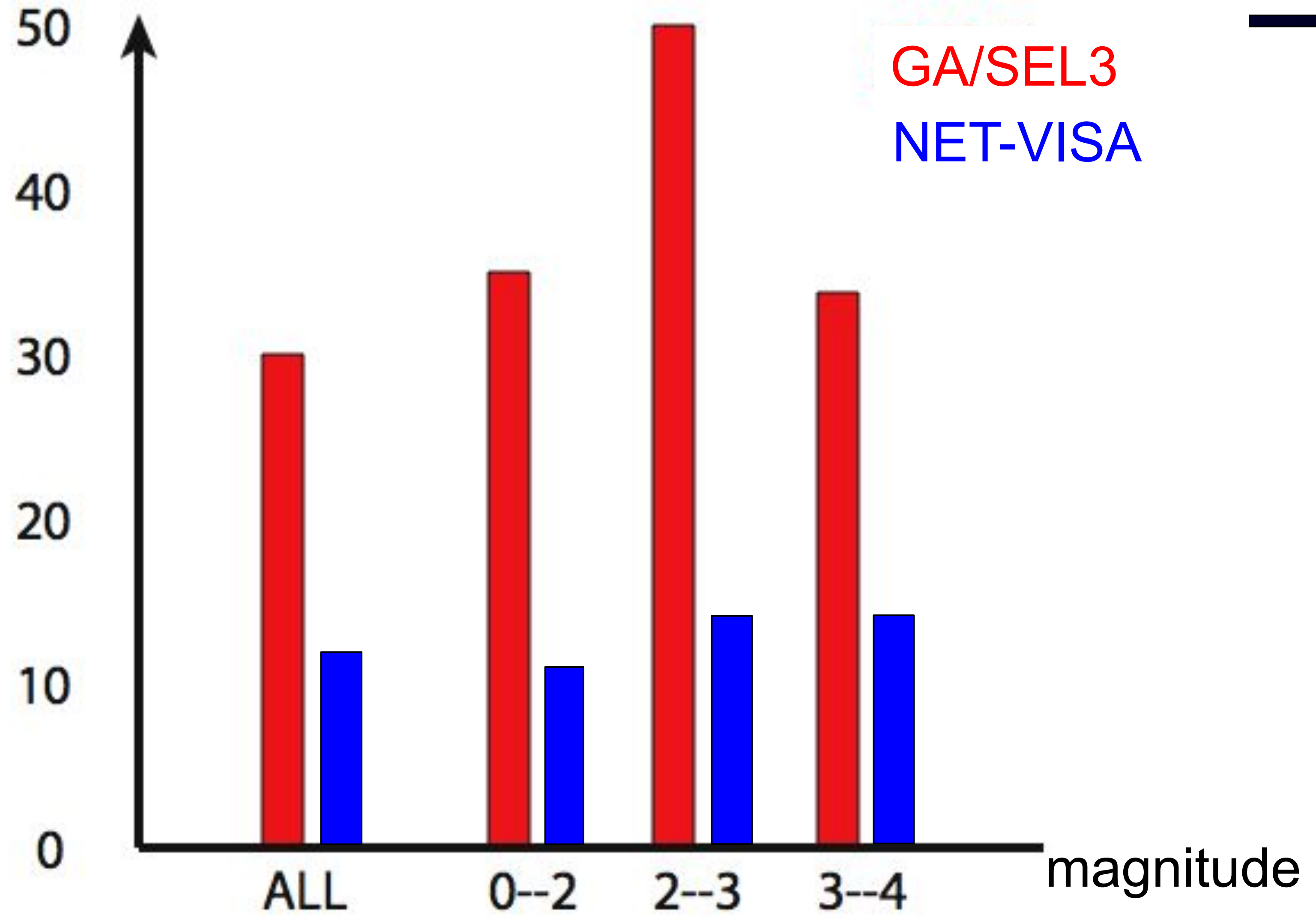
Inference

- NET-VISA aims for most likely explanation (MAP)
  - MCMC-like search with 4 moves:
    - Birth move suggests new events to account for new arrivals
    - Improve-arrival move selects a better event for a given arrival
    - Improve-event optimizes event location, time given its arrivals
    - Death move kills events with no arrivals or negative likelihood
- Runs faster than real time on a laptop with zero GPUs
  - Huge events (Sumatra, Tohoku) handled using ~100 nodes
- Key point for “client”: computing posterior probabilities takes the *algorithm* off the table; to get better answers,
  - Improve the model, or
  - Add more sensors

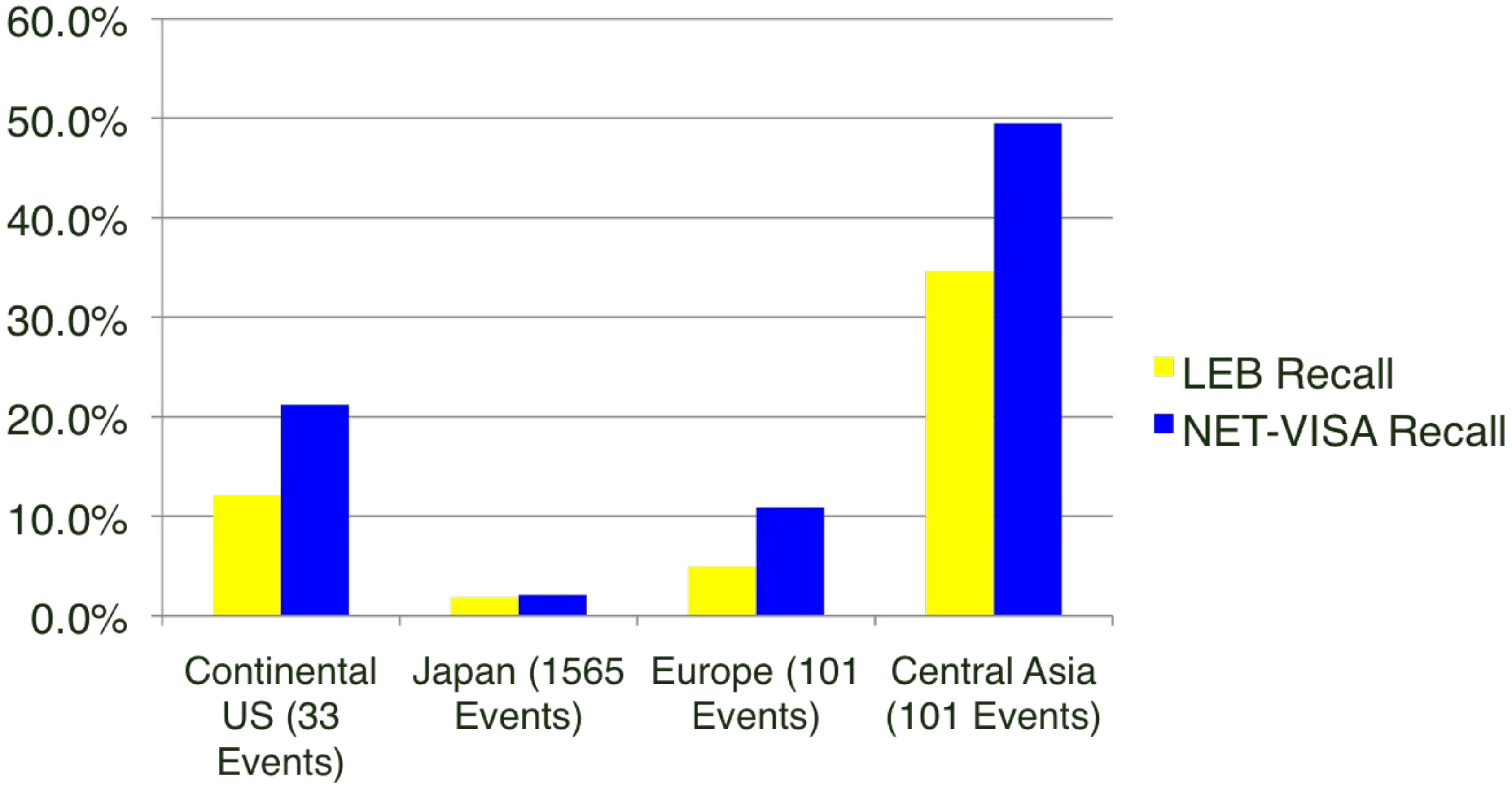
Evaluation

- 11 weeks of training data from all IMS stations
- Test on 1 week of unseen data
- Evaluated using human-expert LEB as “ground truth”

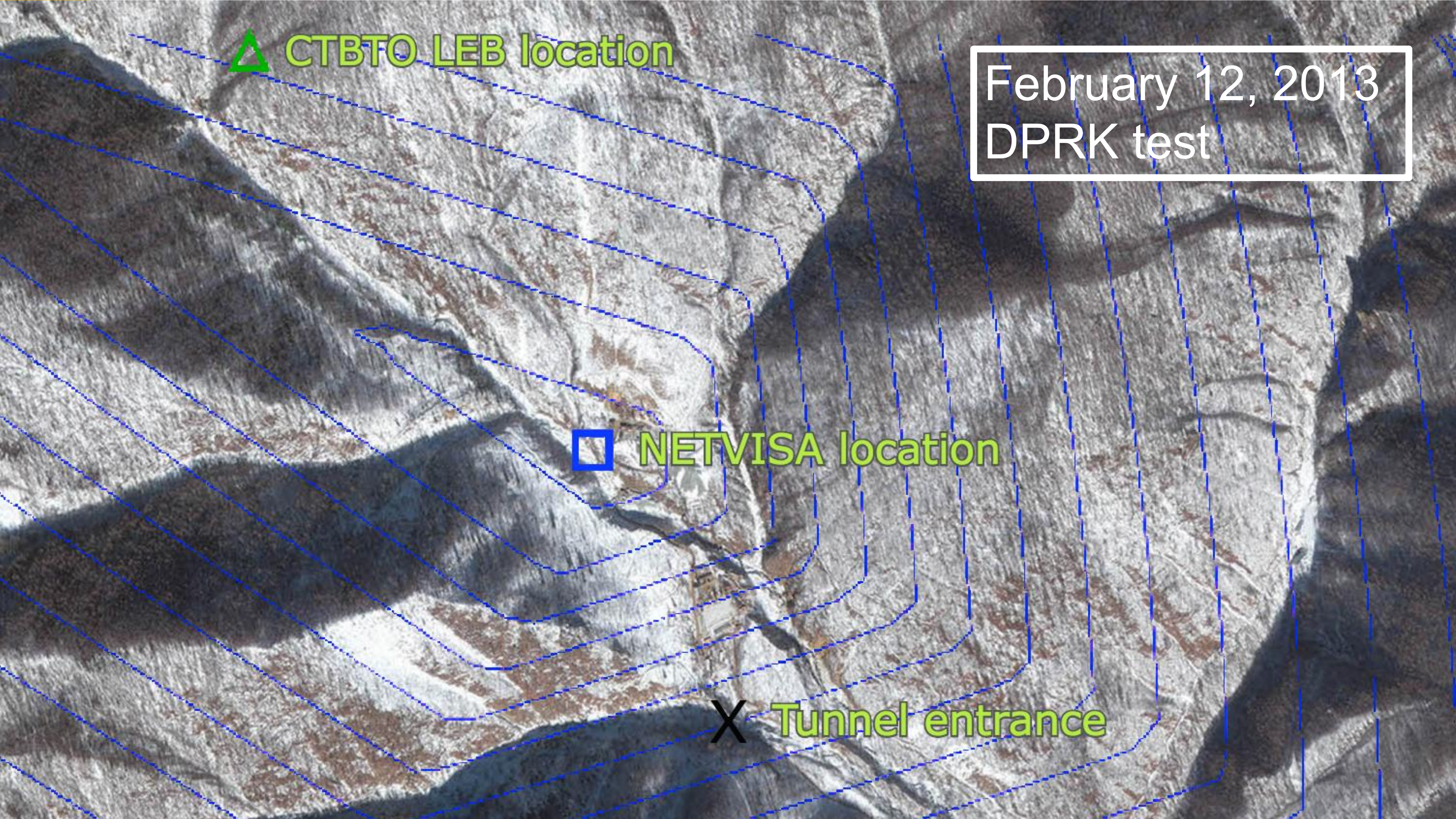
Detection failure rate



Detecting Events Found by Regional Networks



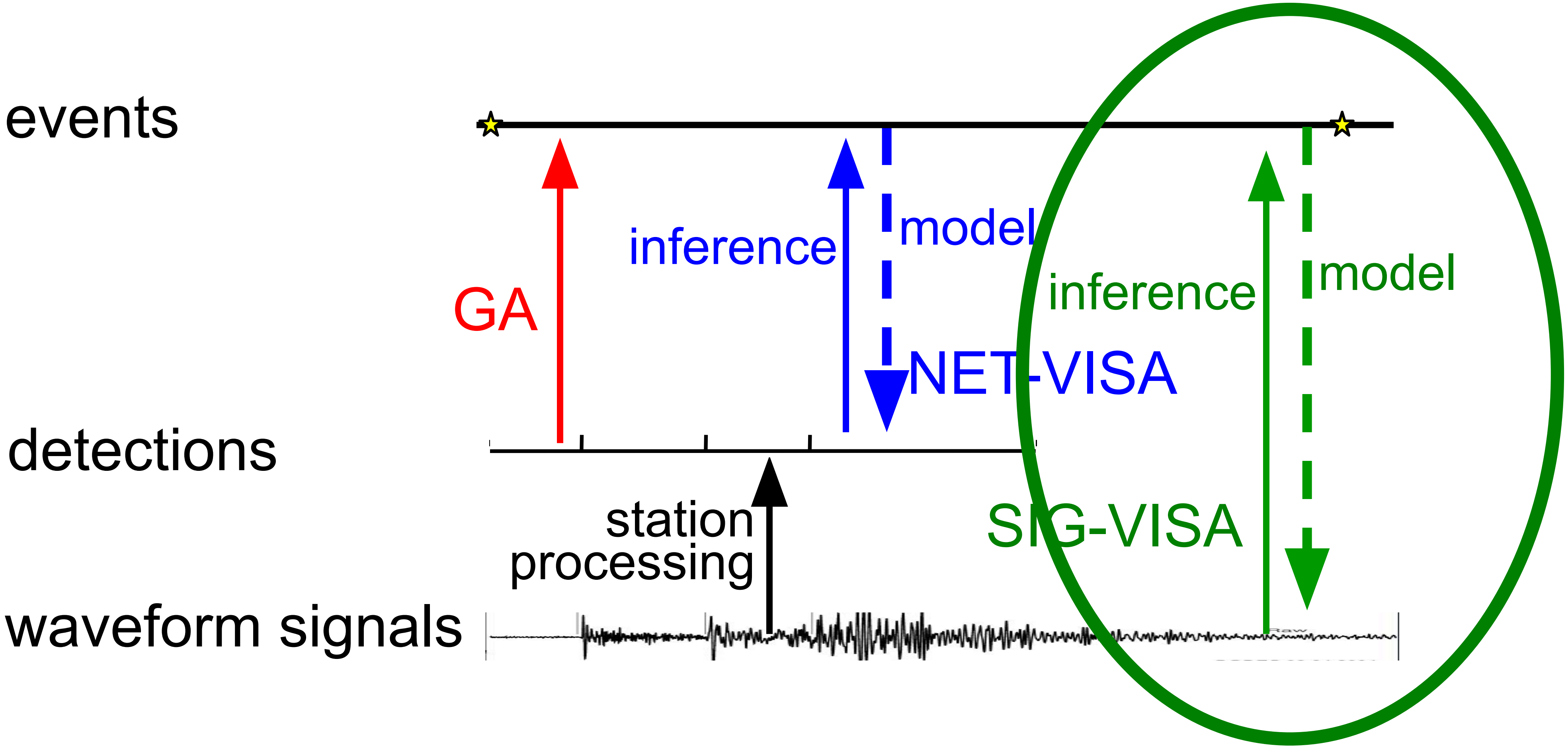
NET-VISA recall 20-100% better than human experts


CTBTO LEB location
February 12, 2013
DPRK test
NETVISA location
Tunnel entrance

Why does NETVISA work better?

- Better modeling
  - Non-Gaussian error distributions (GA uses least squares => Gaussian)
  - Location priors for natural events
  - Detection probabilities
- *<u>Non</u>*-detections!!
- Combining evidence properly
- More detections associated with each event
- Much better at finding *low-evidence* events

Detection-based and signal-based monitoring



SIGVISA model adds…

General waveform shape and coda decay rate

Superposition of signals

Spatial continuity of travel time residuals

Spatial continuity + repeatability of waveforms

And inference naturally yields….

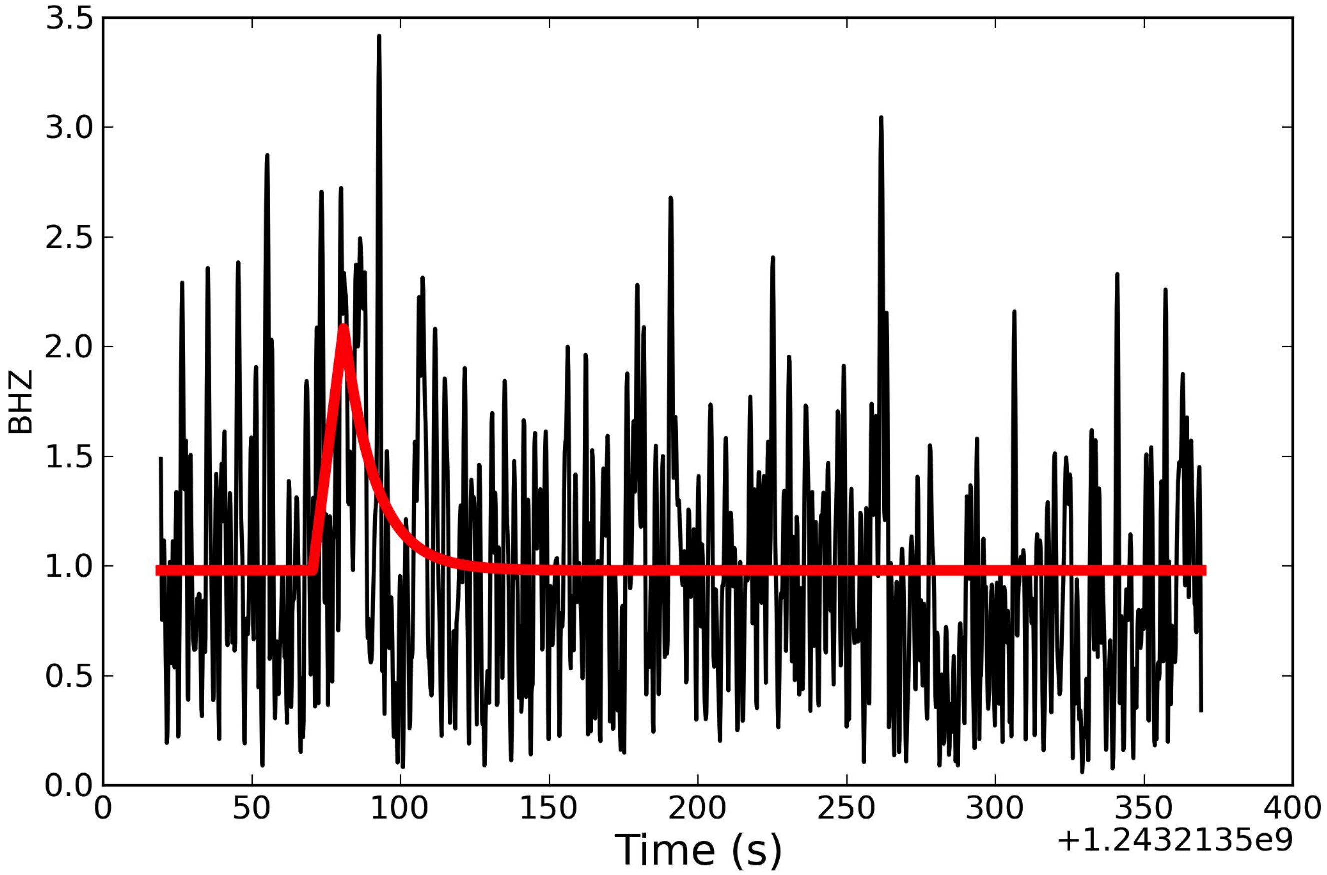
Sub-threshold detection, global beamforming

Locations from single-station detections

Accurate locations via “double-differencing”

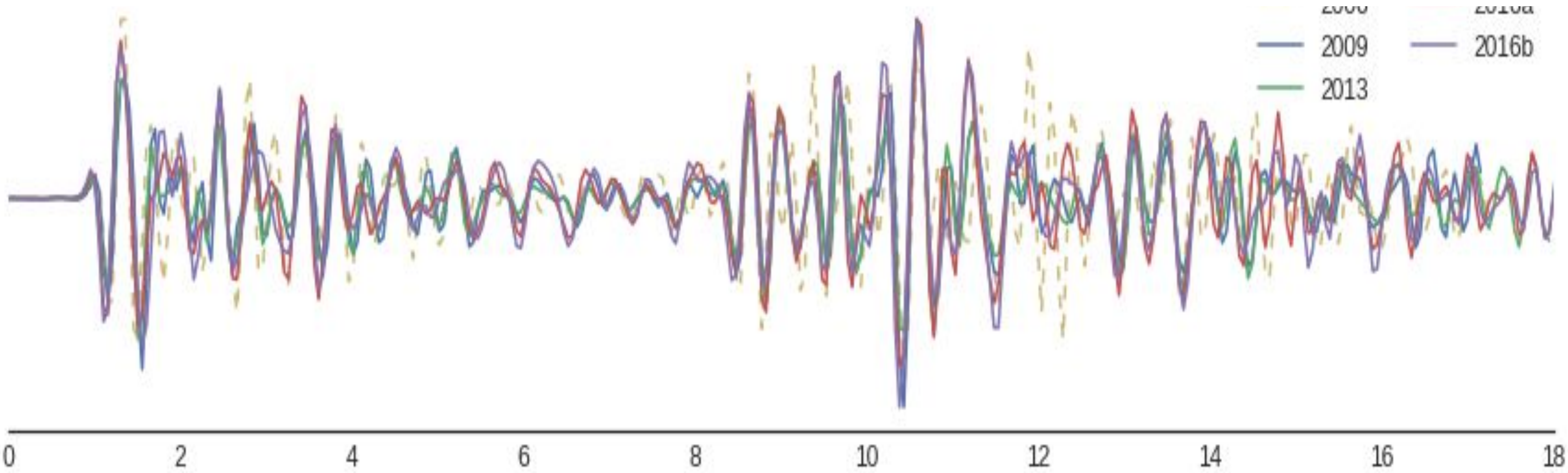
Deconvolution of overlapping arrivals

Improved sensitivity, no hard-threshold effect

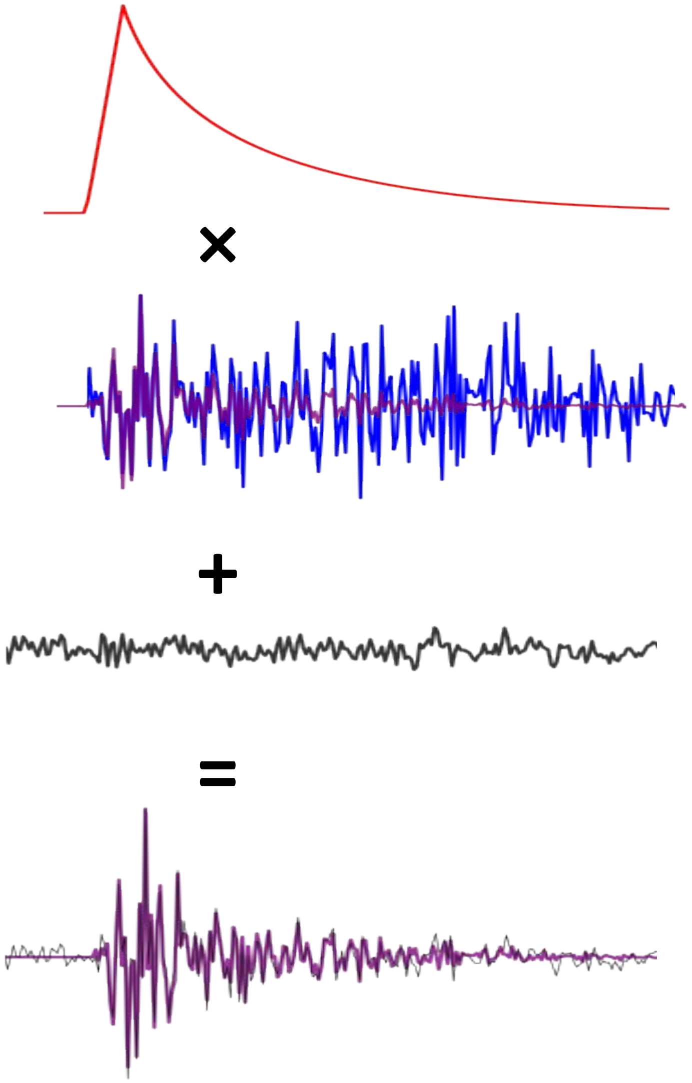


2009 DPRK event detected by SIGVISA @ DZM (New Caledonia), missed by IMS station processing

Correlated signals from nearby events



Signal model: single event/phase



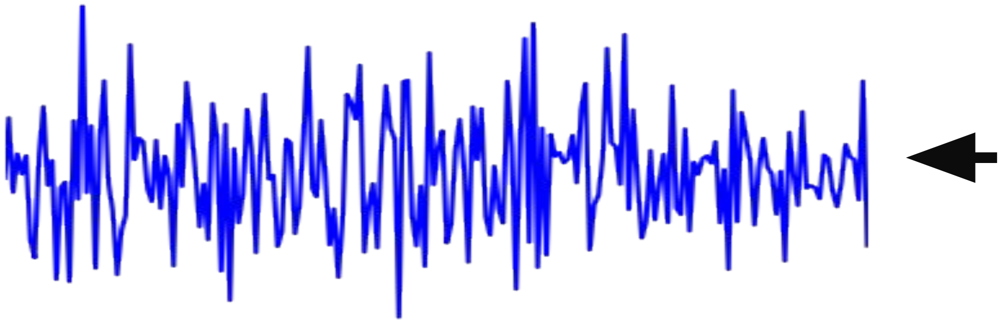
Envelope template: parametric shape depends on event magnitude, depth, location, phase.

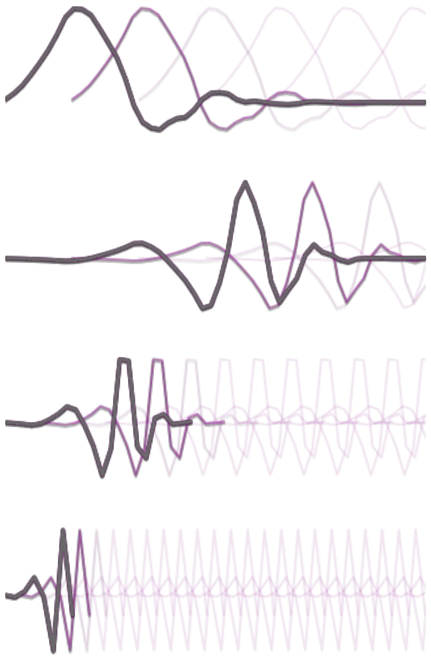
Repeatable modulation: the “wiggles”, depends on event location, depth, phase.

Background noise: autoregressive process at each station.

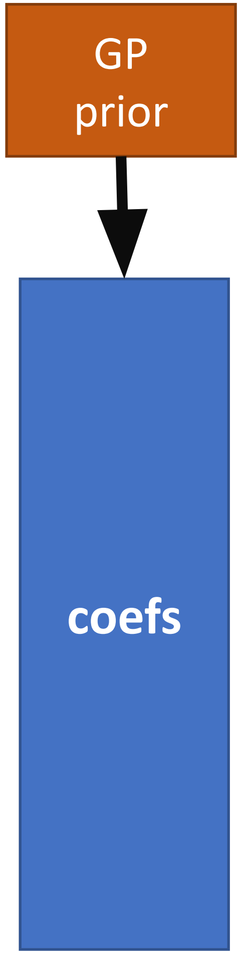
Observed signal: sum of all arriving phases, plus background noise.

Repeatable modulation signal





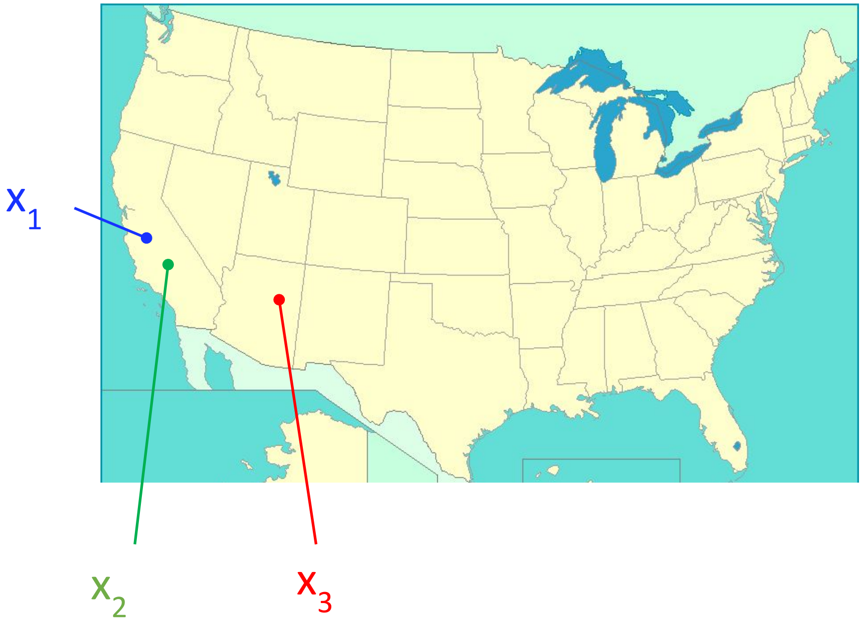
Daubechies db4
wavelet basis

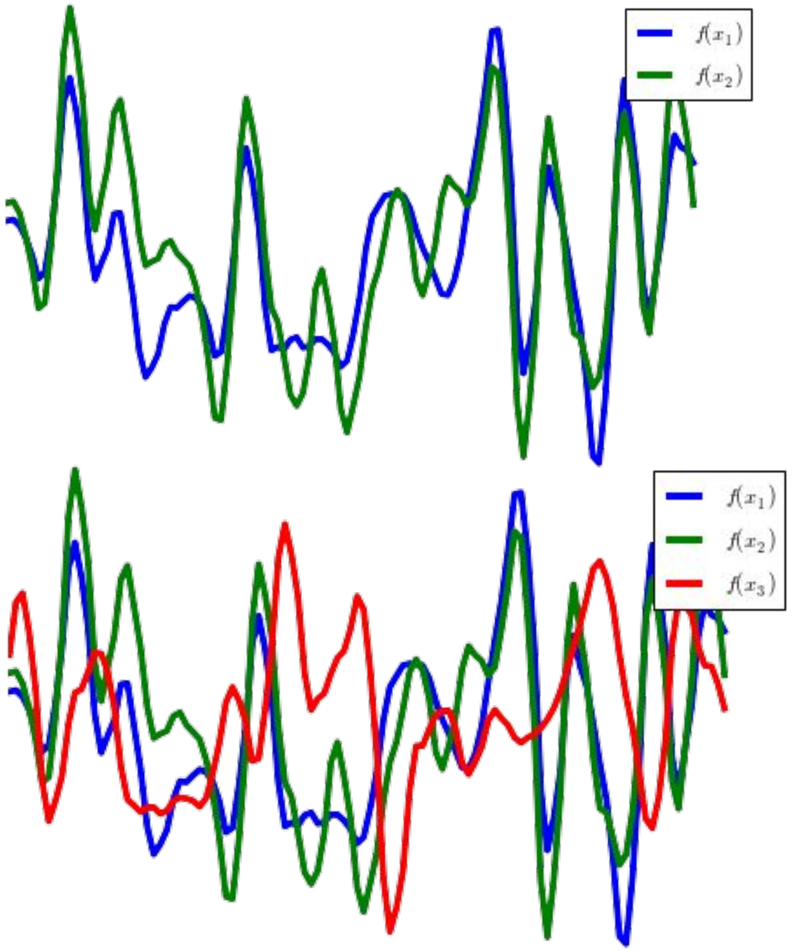


Basis coefficients (wavelets) from a Gaussian process conditioned on event location.

Result: nearby events generate similar signals.

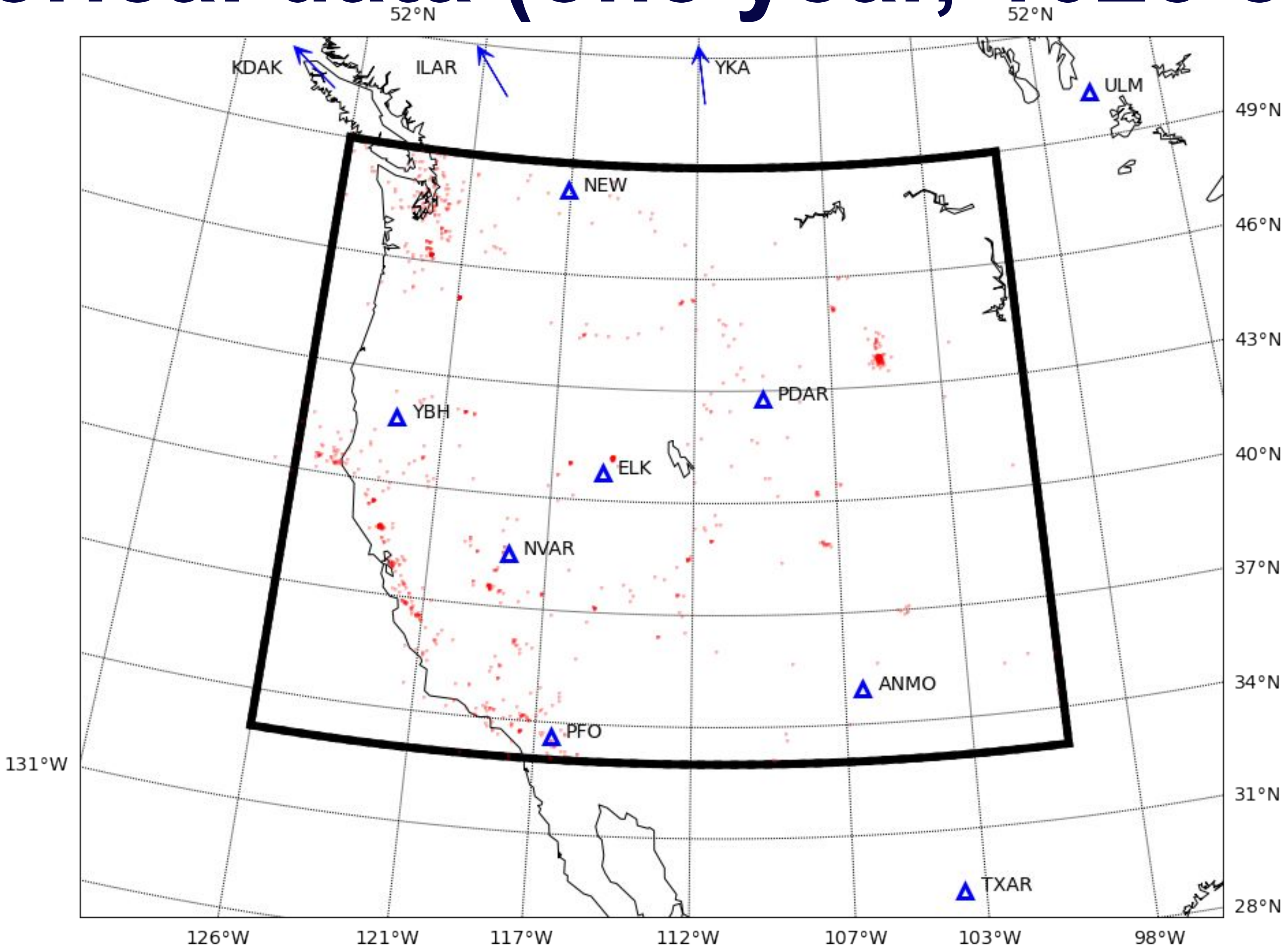
Repeatable modulation signal



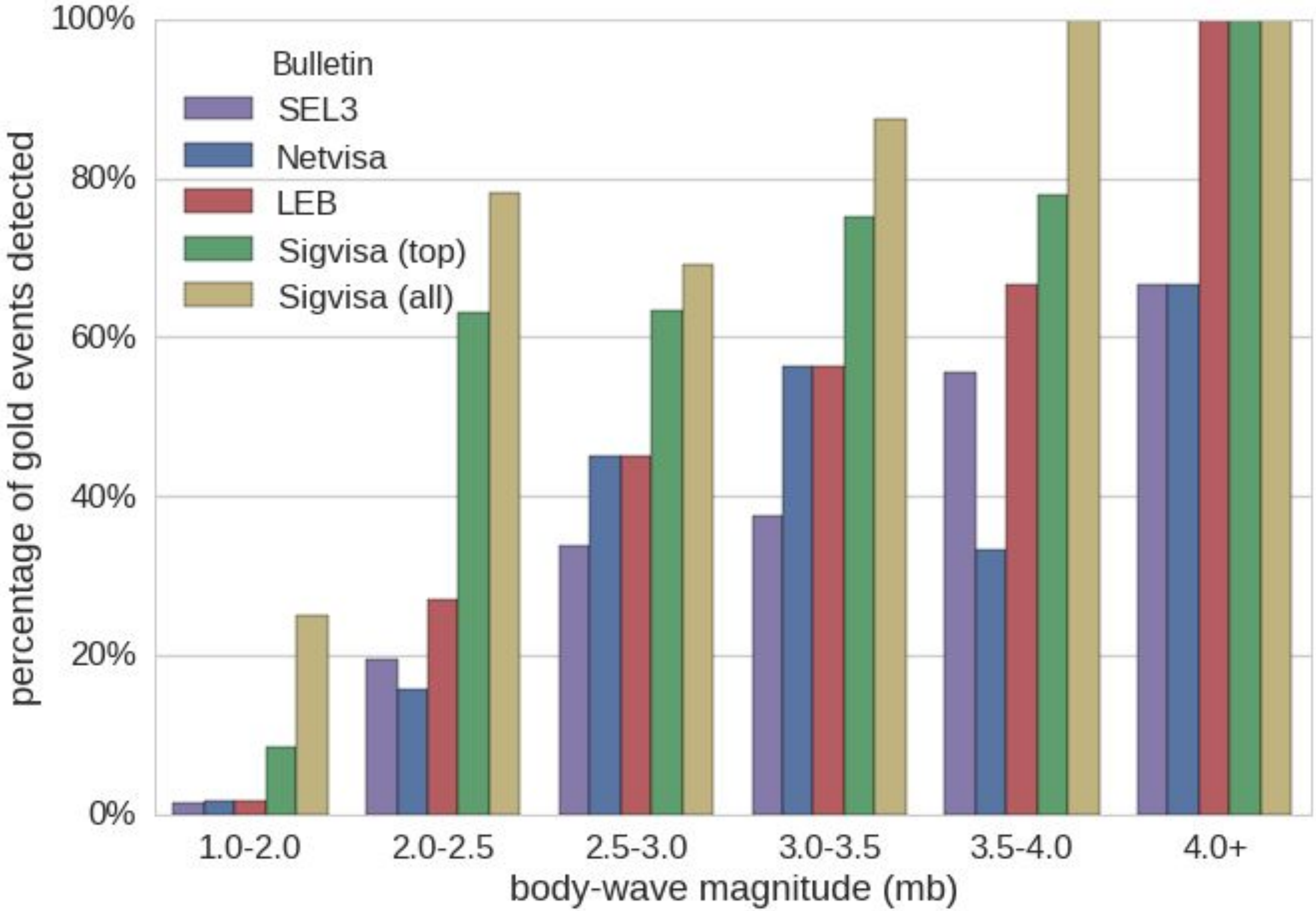


Signals from 45 wavelet coefficients sampled from GP

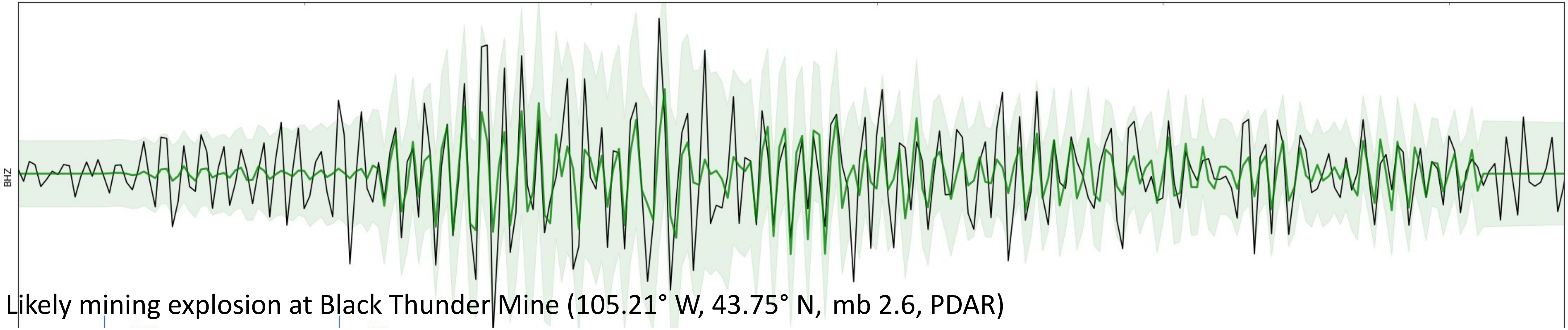
Historical data (one year, 1025 events)



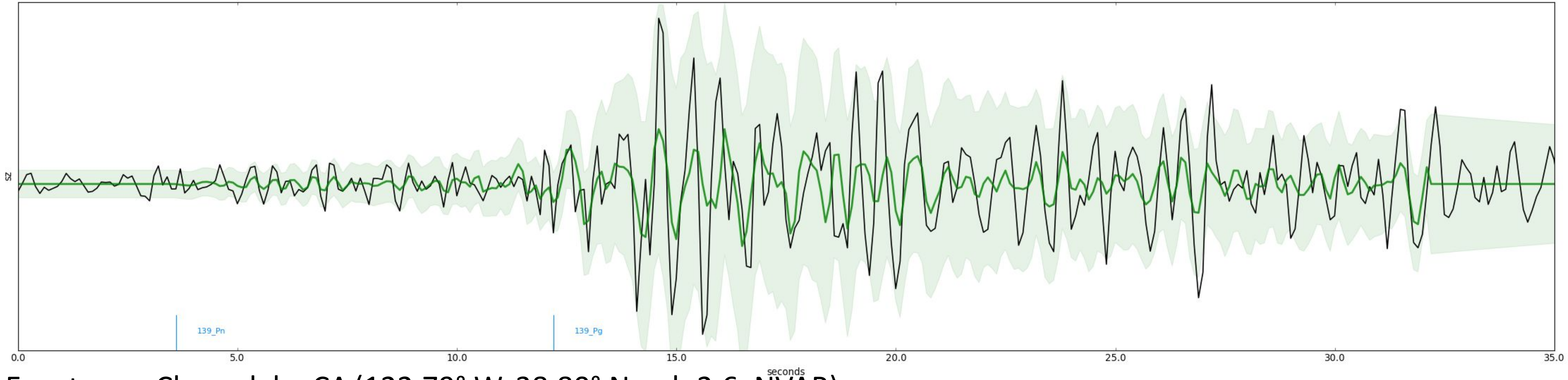
SIGVISA recall results (Western US)



Events not in reference bulletin



Likely mining explosion at Black Thunder Mine (105.21° W, 43.75° N, mb 2.6, PDAR)



Event near Cloverdale, CA (122.79° W, 38.80° N, mb 2.6, NVAR)

Looking forward…

Human oversight still important to handle

- Unmodelled phenomena (e.g., meteorite/ocean, other phases)
- Picking, alignment, and association errors
- Unusual waveforms

Should output multiple hypotheses and probabilities;
Should continuously re-estimate models;
Should treat all catalogue event locations as latent variables

Flagging an event is a ***decision***, based on relative costs of false positives and false negatives (hence probability thresholds)

#SeismicEvents ~ Poisson[T*λ_e];
Time(e) **~** Uniform(0,T)
IsEarthQuake(e) ~ **Bernoulli(.999);**
Location-Depth(e) ~ if IsEarthQuake(e) then SpatialPrior() else **UniformSurfaceDistribution();**
Depth(e) ~ if IsEarthQuake(e) then Uniform[0,700] else 0;
Magnitude(e) ~ Exponential(log(10));
IsDetected(e,p,s) ~ Logistic[weights(s,p)](Magnitude(e), Depth(e), Distance(e,s))**;**
#Detections(site = s) ~ Poisson[T*λ_f(s)];
#Detections(event=e, phase=p, station=s) **=** if IsDetected(e,p,s) then 1 else 0;
OnsetTime(a,s) **~** if (event(a) = null) then Uniform[0,T] else
Time(event(a)) + GeoTravelTime(Distance(event(a),s),Depth(event(a)),phase(a)) +
Laplace(μ_t(s), σ_t(s))
Amplitude(a,s) **~** If (event(a) = null) then NoiseAmplitudeDistribution(s)
else AmplitudeModel(Magnitude(event(a)), Distance(event(a),s),Depth(event(a)),phase(a))
Azimuth(a,s) **~** If (event(a) = null) then Uniform(0, 360)
else GeoAzimuth(Location(event(a)),Depth(event(a)),phase(a),Site(s)) + Laplace(0,σ_a(s))
Slowness(a,s) **~** If (event(a) = null) then Uniform(0,20)
else GeoSlowness(Location(event(a)),Depth(event(a)),phase(a),Site(s)) + Laplace(0,σ_a(s))
ObservedPhase(a,s) **~** CategoricalPhaseModel(phase(a))

Summary

Bayesian model-based approaches allow principled combination of knowledge and data

Probabilistic programming languages make it easy to write models and do inference

NETVISA is in regular daily use at CTBTO; SIGVISA would be much better!

Summary

- Learning is ***essential*** in unknown environments, ***useful*** in many others
- The nature of the learning process depends on
  - agent design
  - what part of the agent you want to improve
  - what prior knowledge and new experiences are available
- Supervised learning: learning a function from labeled examples
  - Concise hypotheses generalize better; trade off conciseness and accuracy
  - Classification: discrete-valued function; example: decision trees
  - Regression: real-valued function; example: linear regression
- Next: Bayesian learning