Using Multivariate Statistical Classification to Rank-Order Potential Gravitational-Wave Events

Kari Hodge[1] for the LSC

[1]California Institute of Technology, LIGO

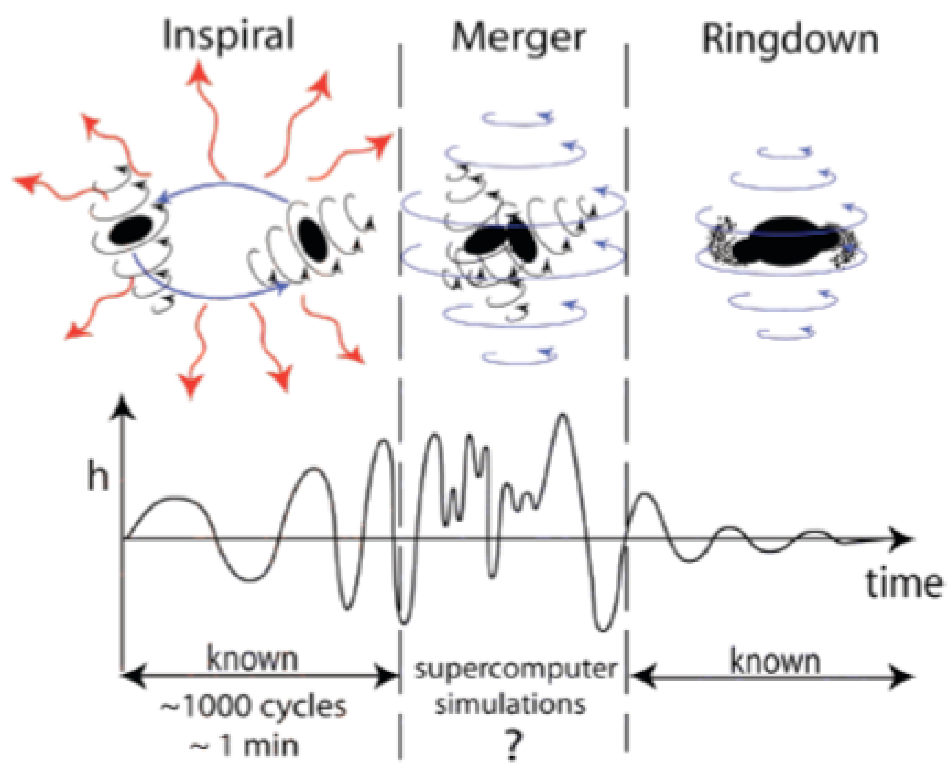
Pacific Coast Gravity Meeting

22 March 2008

LIGO-G080031-00-Z

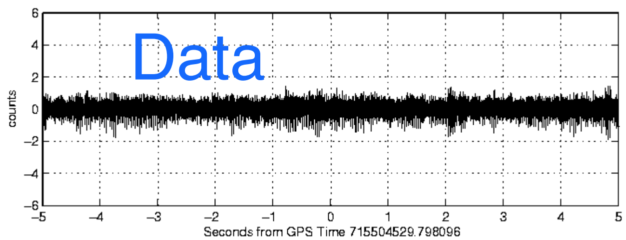
Compact Binary Coalescences

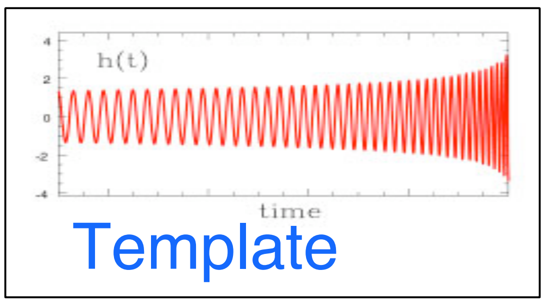
- Compact Binaries:
  - » Two neutron stars
  - » Two black holes
  - » A neutron star and a black hole
- The gravitational waveform emitted by the system during the inspiral phase of the coalescence has been modeled with General Relativity
  - » Second order Post-Newtonian templates

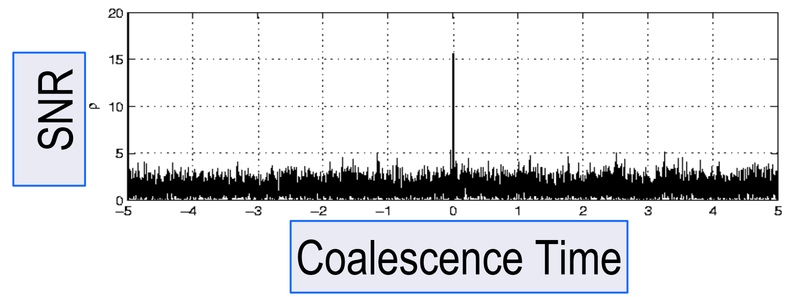


Inspiral Pipeline

1. Matched Filtering







+

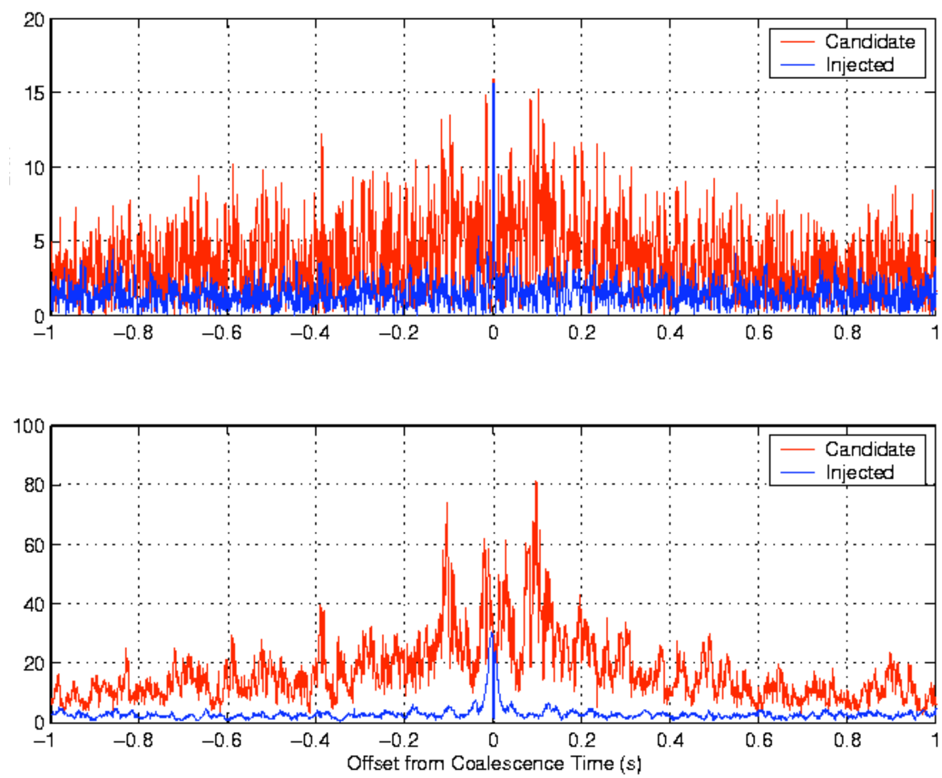
Parameters (masses, t_{coal},…)

2. Coincidence: Parameters from more than one detector are similar in time and mass

3. Follow up

Signal Based Vetoes

- Any large glitch in the data can cause the matched filter to have a large SNR output
  - » Loud glitches hide low-SNR signals = reduction in rate
  - » Reduces the volume of the sky we can see by (reduction in rate)^3
- Signal based vetoes (χ^2 test, r^2 veto duration) check that the matched filter output is consistent with a signal

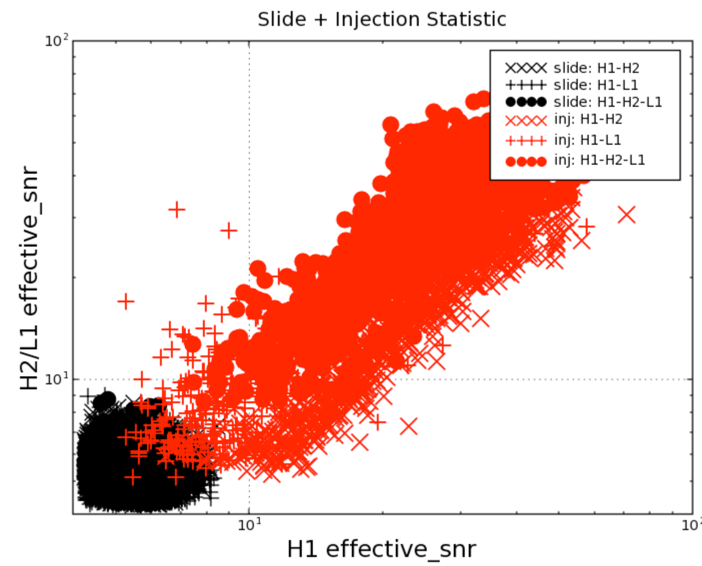


LIGO-G060580-00-Z

Follow-up Candidate Events

- Follow-ups are necessary because of the glitches that ring up triggers and pass the signal based vetoes
- Currently, the candidates are ranked according to the sum of squares of effective SNR for each detector

$$\rho_{eff}^2 = \frac{\rho^2}{\sqrt{\left(\frac{\chi^2}{2p-2}\right)\left(1+\frac{\rho^2}{250}\right)}}$$



- The top-ranking candidates on this list are subjected to rigorous examination

Multivariate Analysis

- Multivariate statistical classification methods can take into account the correlations between the many parameters that describe a candidate event and create a more robust rank-ordering statistic
  - » Random forests are the state of the art in multivariate analysis

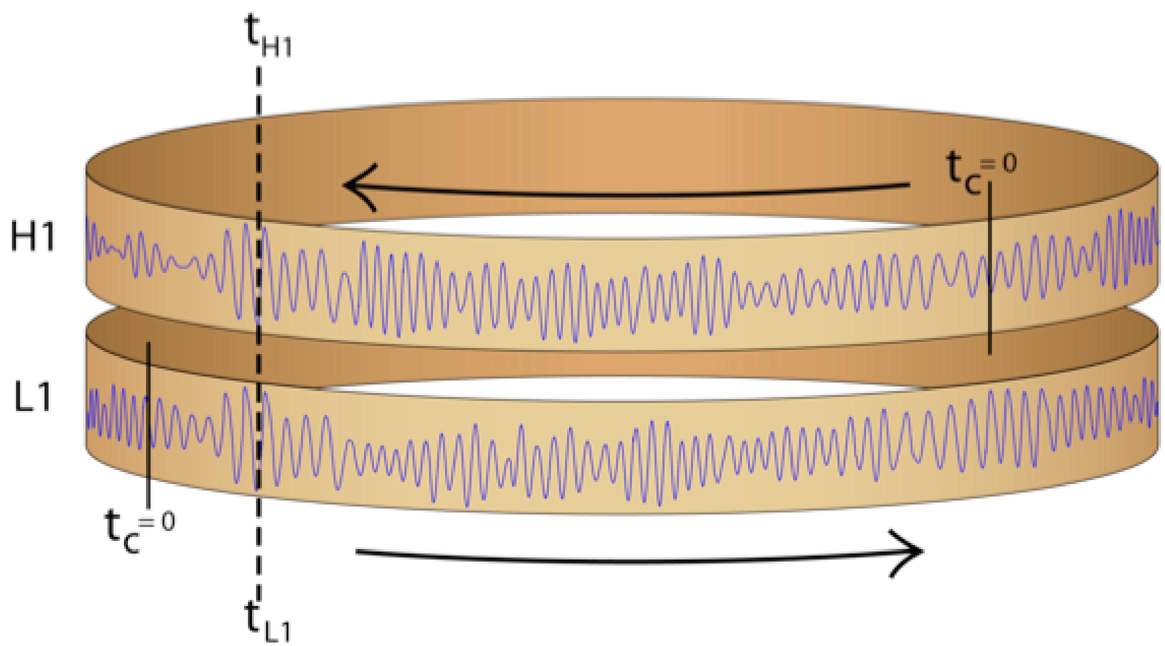
- I use simulated gravitational waves "injected" into the data as the signal for my analysis
  - » From the LSC's 1st year S5 Low-Mass Compact Binary Coalescence Analysis*
  - » I am only using H1-L1 coincidences for the moment since they are harder to classify as signal than triply coincident events

I have 9,569 injections

*LIGO-G070820-00-Z

Background

- Time slides estimate the background:
  - » The data streams from two detectors are slid integer multiples of 5 seconds from each other and run through the Inspiral Pipeline
  - » These accidental coincidences can't be gravitational waves
  - » I use time slides from one month of the S5 playground



I have a total of 267,689 time slides

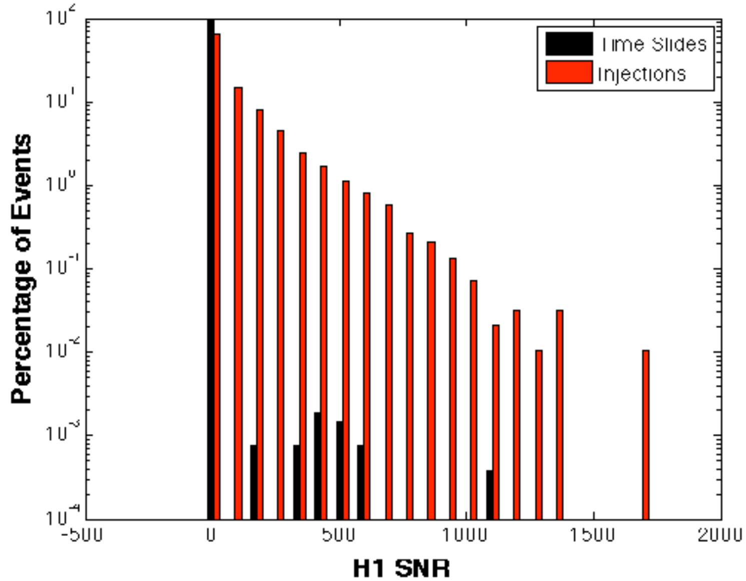
Input Parameters

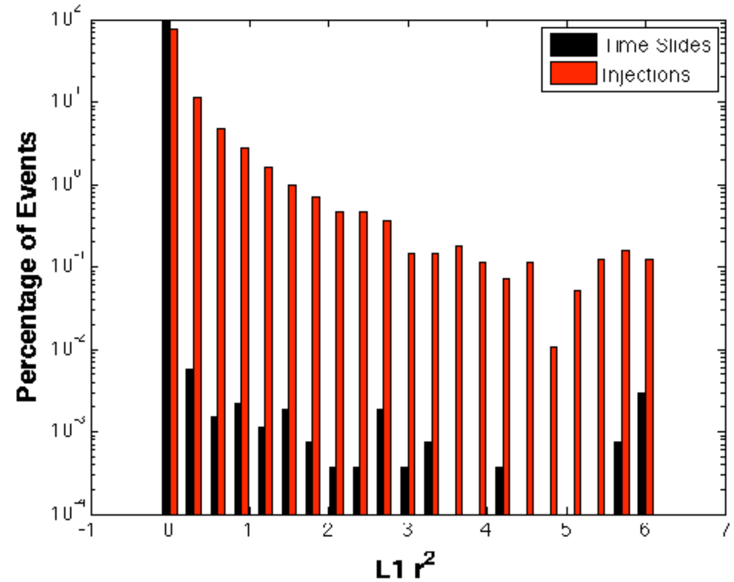
- Single detector parameters
  - SNR
  - chi^2
  - r^2 veto duration
  - $\mathrm{SNR}_{\mathrm{eff}}^2$

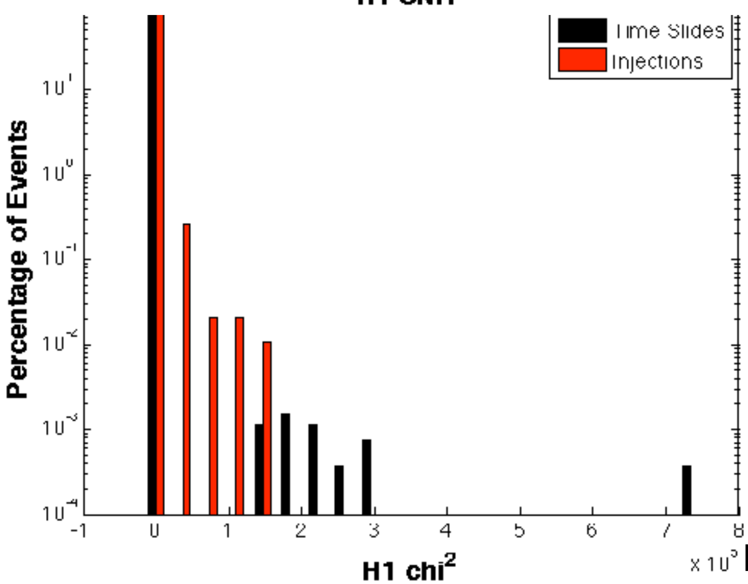
$$\rho_{eff}^2 = \frac{\rho^2}{\sqrt{\left(\frac{\chi^2}{2p-2}\right)\left(1+\frac{\rho^2}{250}\right)}}$$

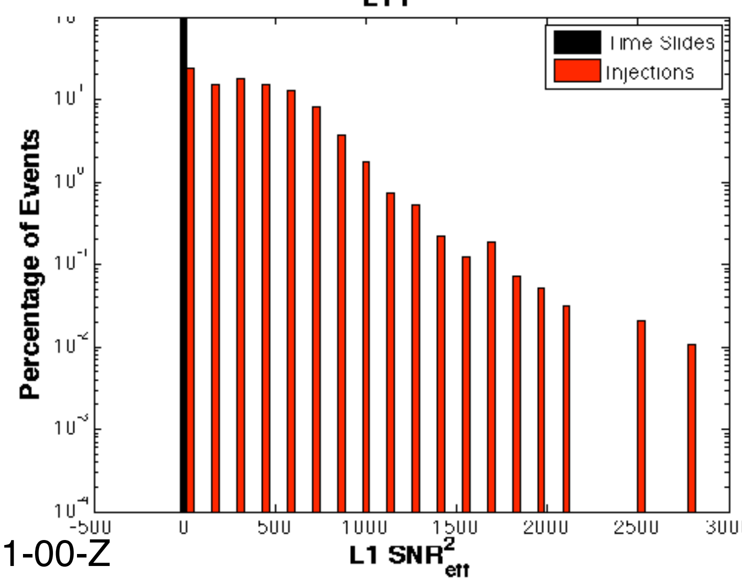
- Coincidence parameters
  - Difference in coalescence time
  - Relative difference in chirp mass

Single Detector Parameters

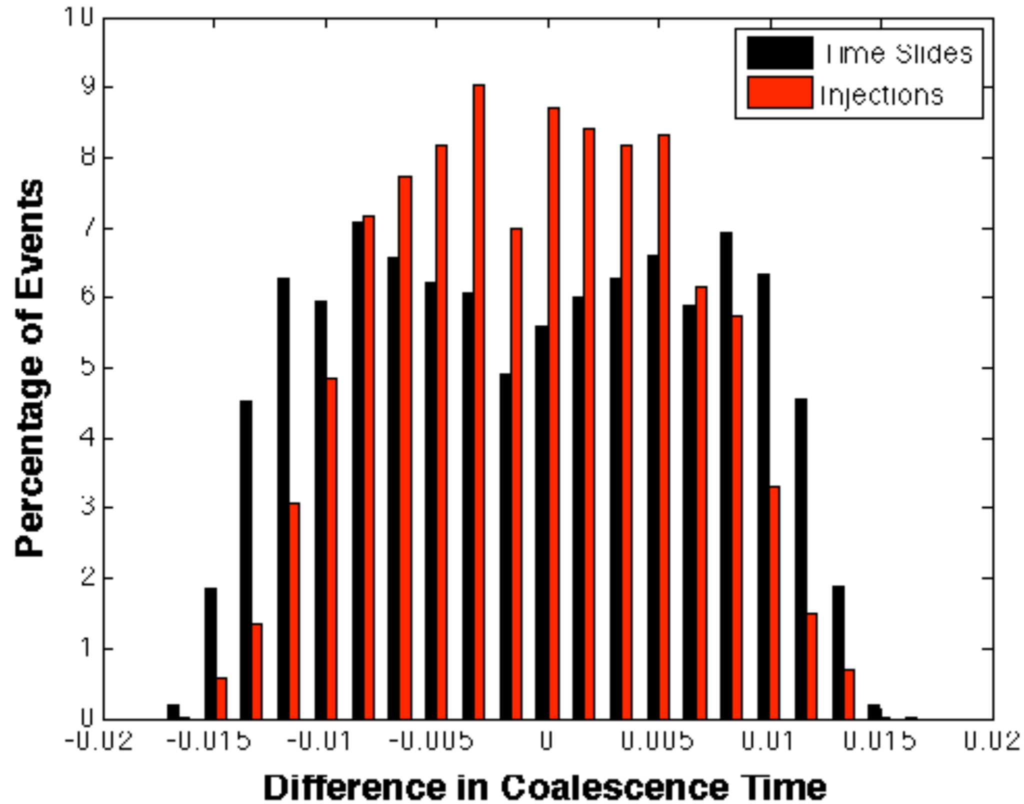


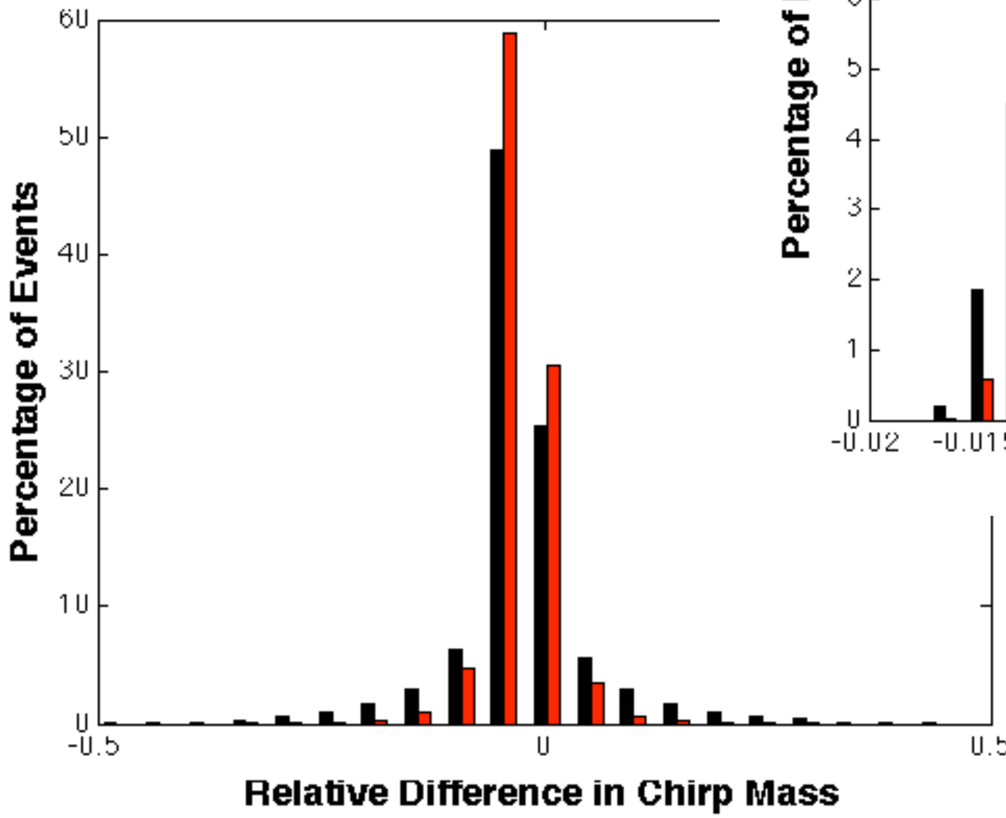




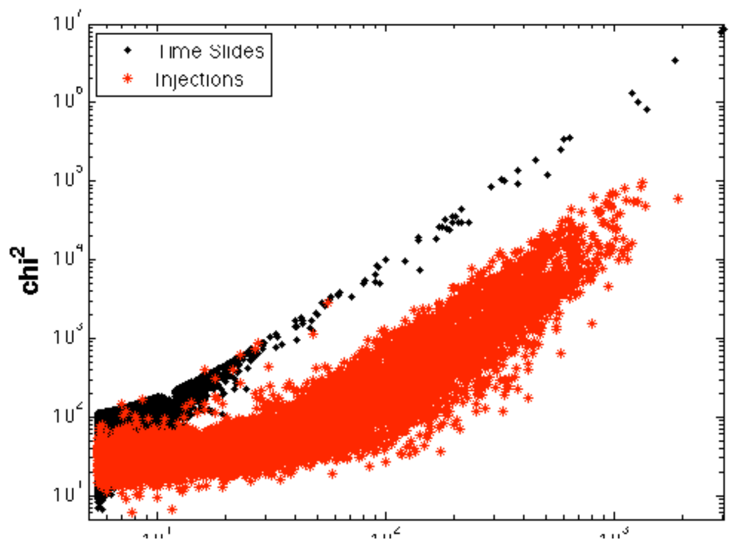


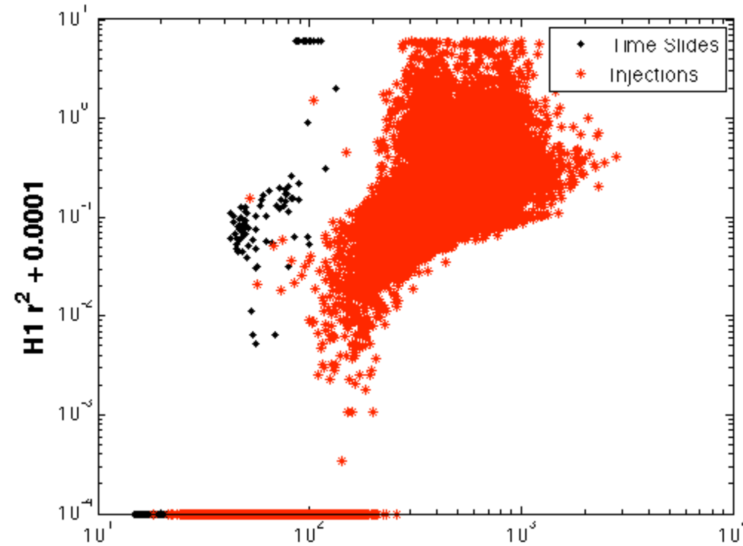
Coincidence Parameters

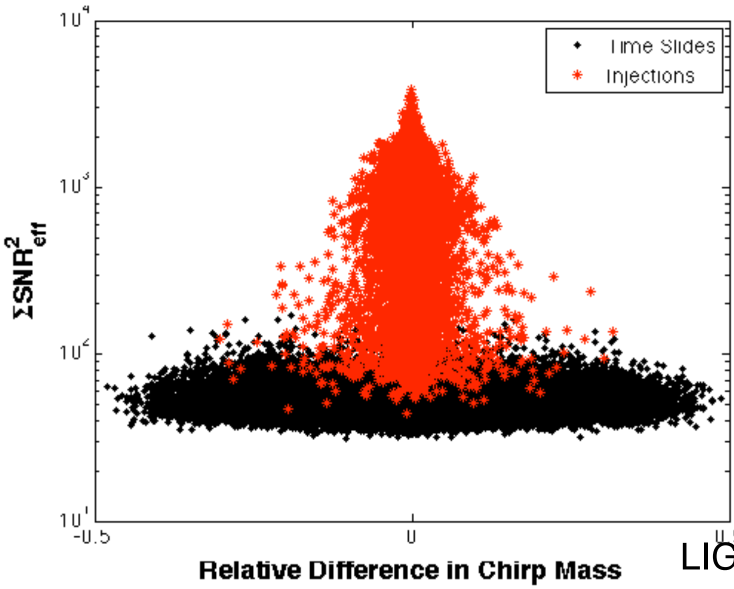


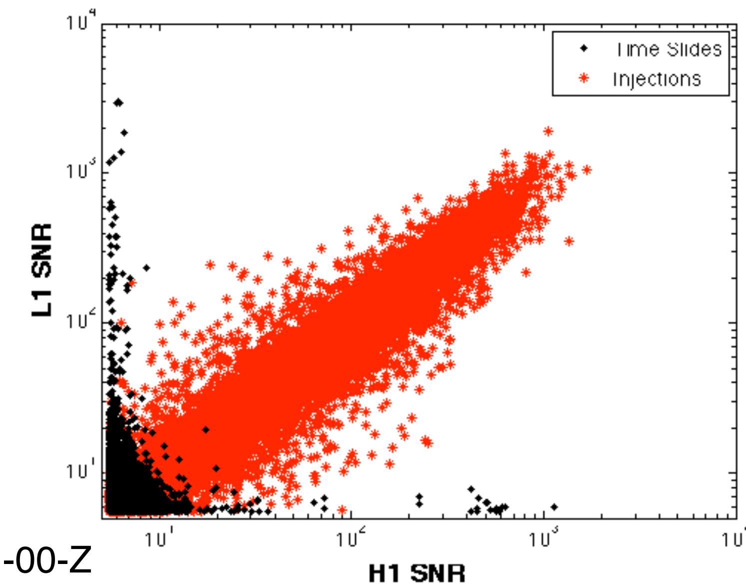


Two Dimensions



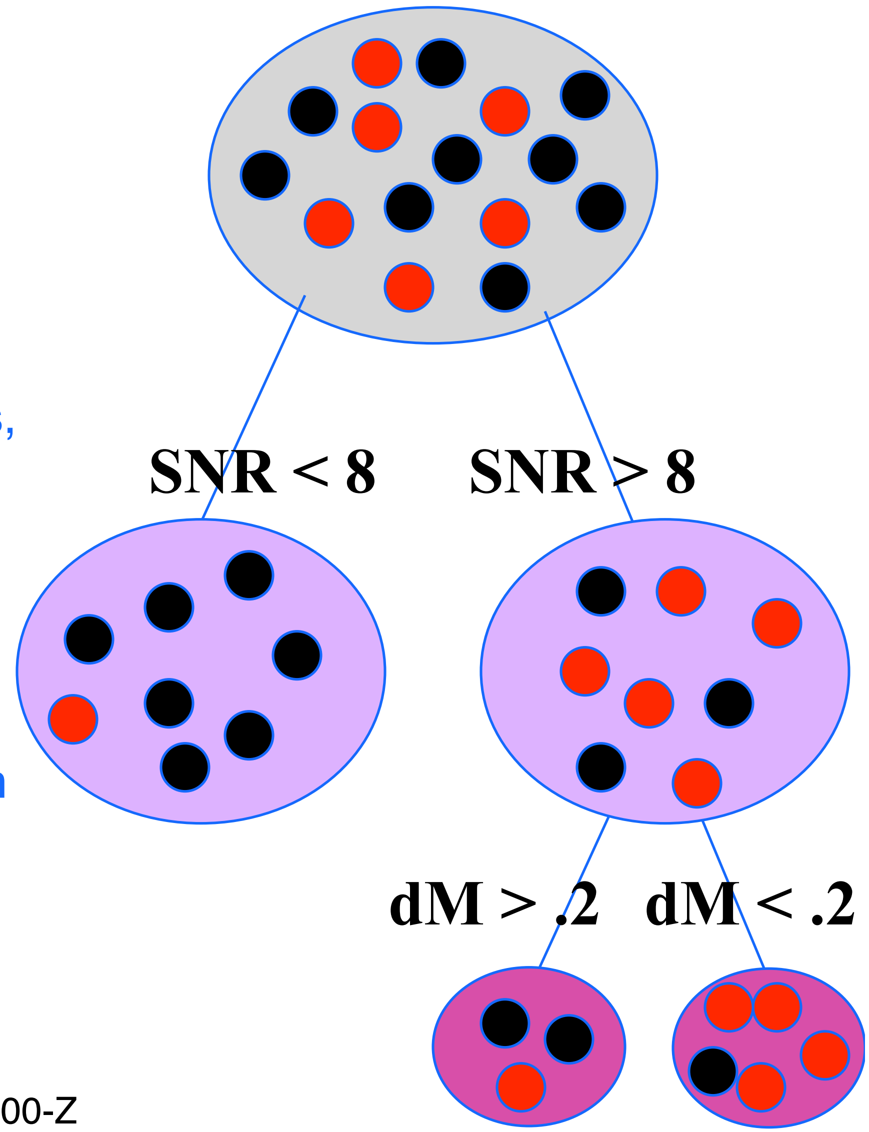






What is a Random Forest?

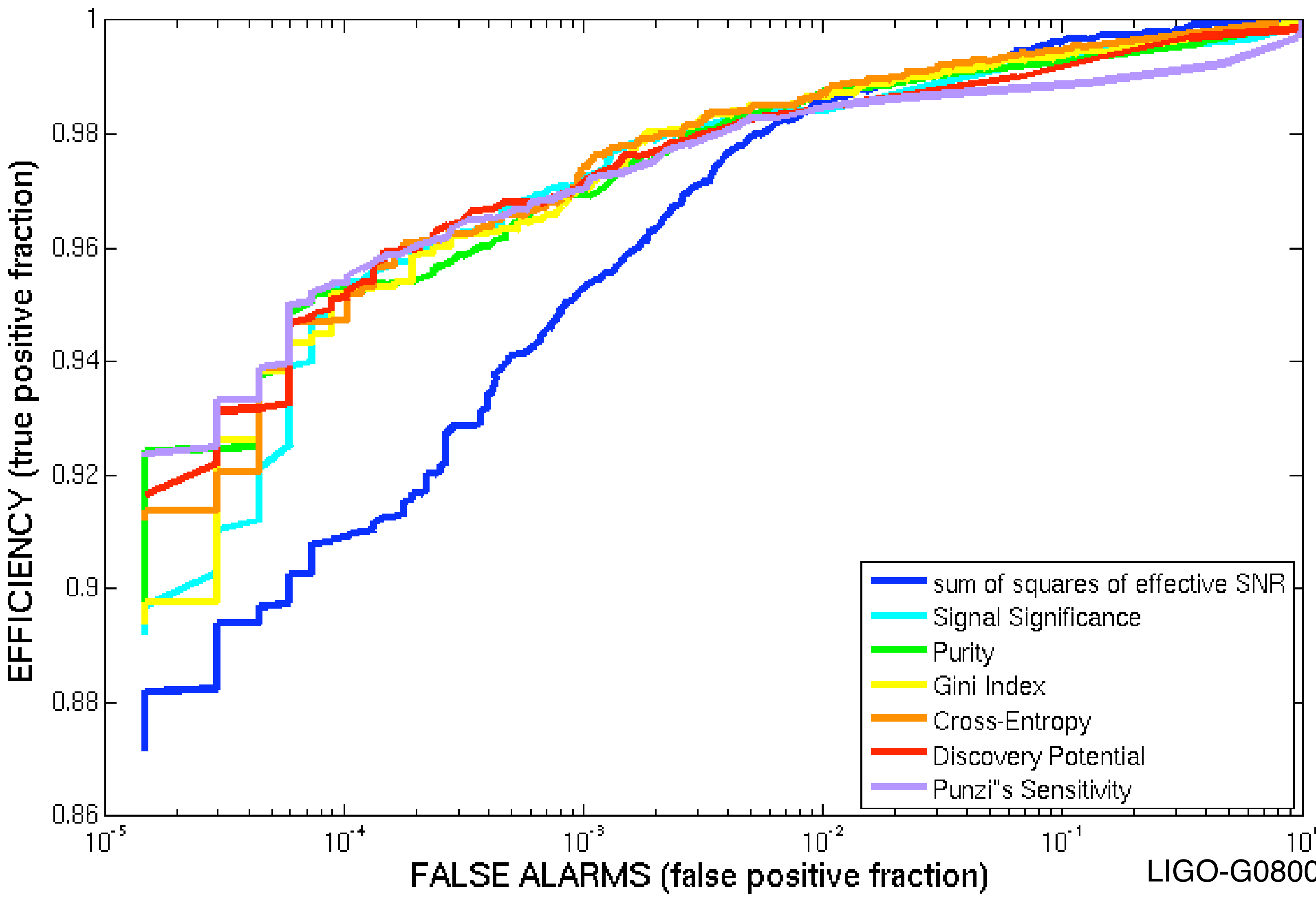
- Use bagging (bootstrap aggregating) to create many decision trees on bootstrap replicas of your training set
  - » Start with N training events
  - » Create many trees, each having N events, chosen with replacement
- Random Forests bootstrap input dimensions
  - » M input parameters
  - » Randomly choose m < M of these at each split on each tree
    - – Choose the cut that optimizes a certain criterion
- Average over all trees in the end



SprBaggerDecisionTree

- I use SprBaggerDecisionTree, from the C++ package StatPatternRecognition to train a random forest of bagged trees
  - » Written by Caltech particle physicist Ilya Narsky
- The random forest technology will sample up to 4 out of 10 of the variables for each split on the tree
- I build 100 trees
  - » Specify each has a minimum of 5 events per leaf

Criterion Performance

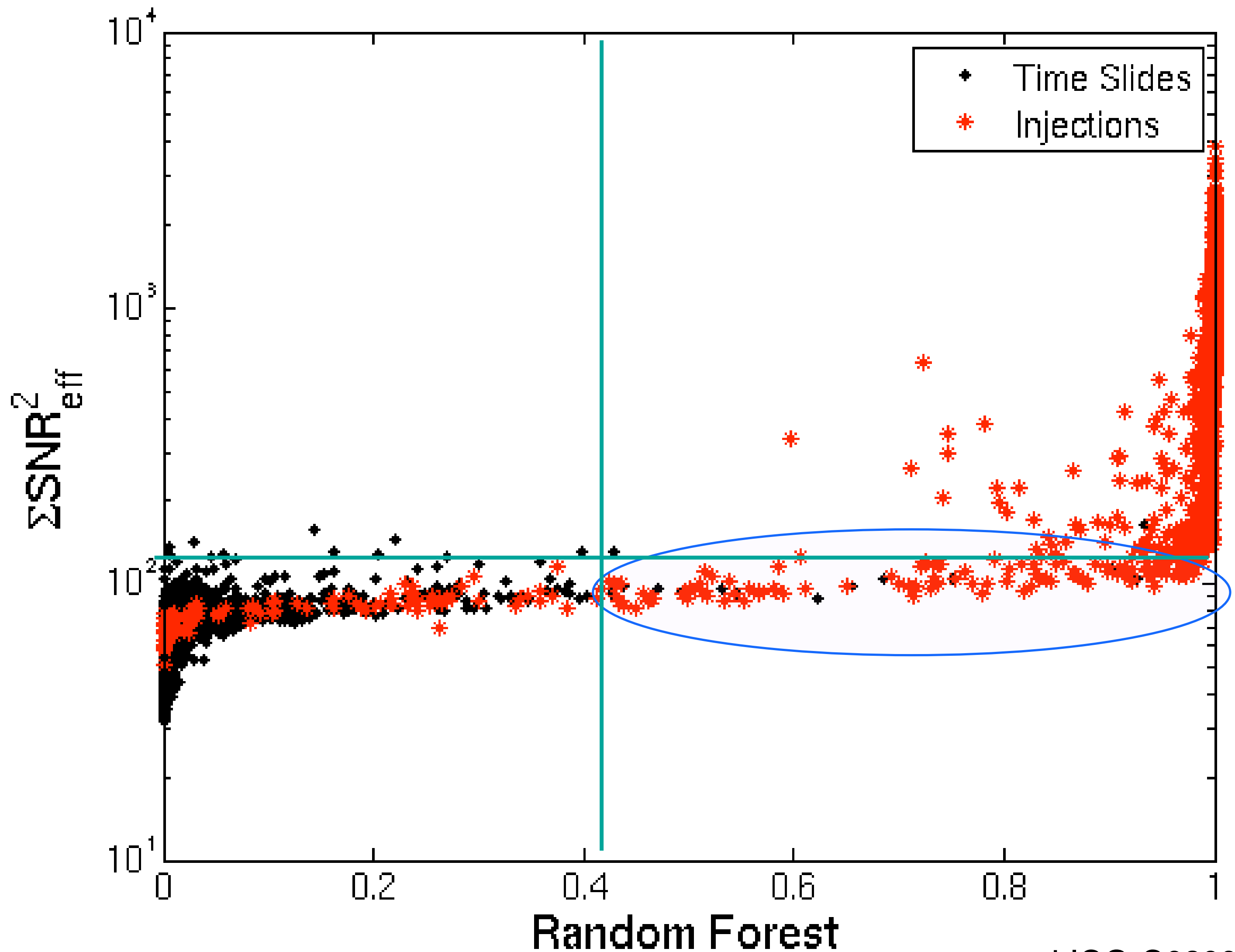


Cross-Entropy

- If you want to live in the region where your false alarm fraction is around 1/1000, then Cross-Entropy gives the best results

Variable	Splits	Delta FOM
dt	1680	193.15895
H1 SNR	2287	255.79869
L1 SNR	2185	254.86952
H1 chi^2	2159	206.18350
L1 chi^2	2499	289.57461
H1 r^2	41	5.58936
L1 r^2	51	8.59637
$(dM)_{rel}$	2360	216.16415
H1 SNR_{eff}^2	2625	264.50807
L1 SNR_{eff}^2	2594	270.51128

Improvement in Region of Weak Signal



Conclusion

- The application of multivariate statistical classification allows us to better separate our signal and background
- The random forest, in particular, separates injected signals from accidental coincidences more effectively than the current ranking statistic
  - » Allows you to see signals with a lower SNR, where the rate is expected to be highest (more space out there!)
- More optimization of the leaf size, number of sampled parameters, etc. will lead to improved results