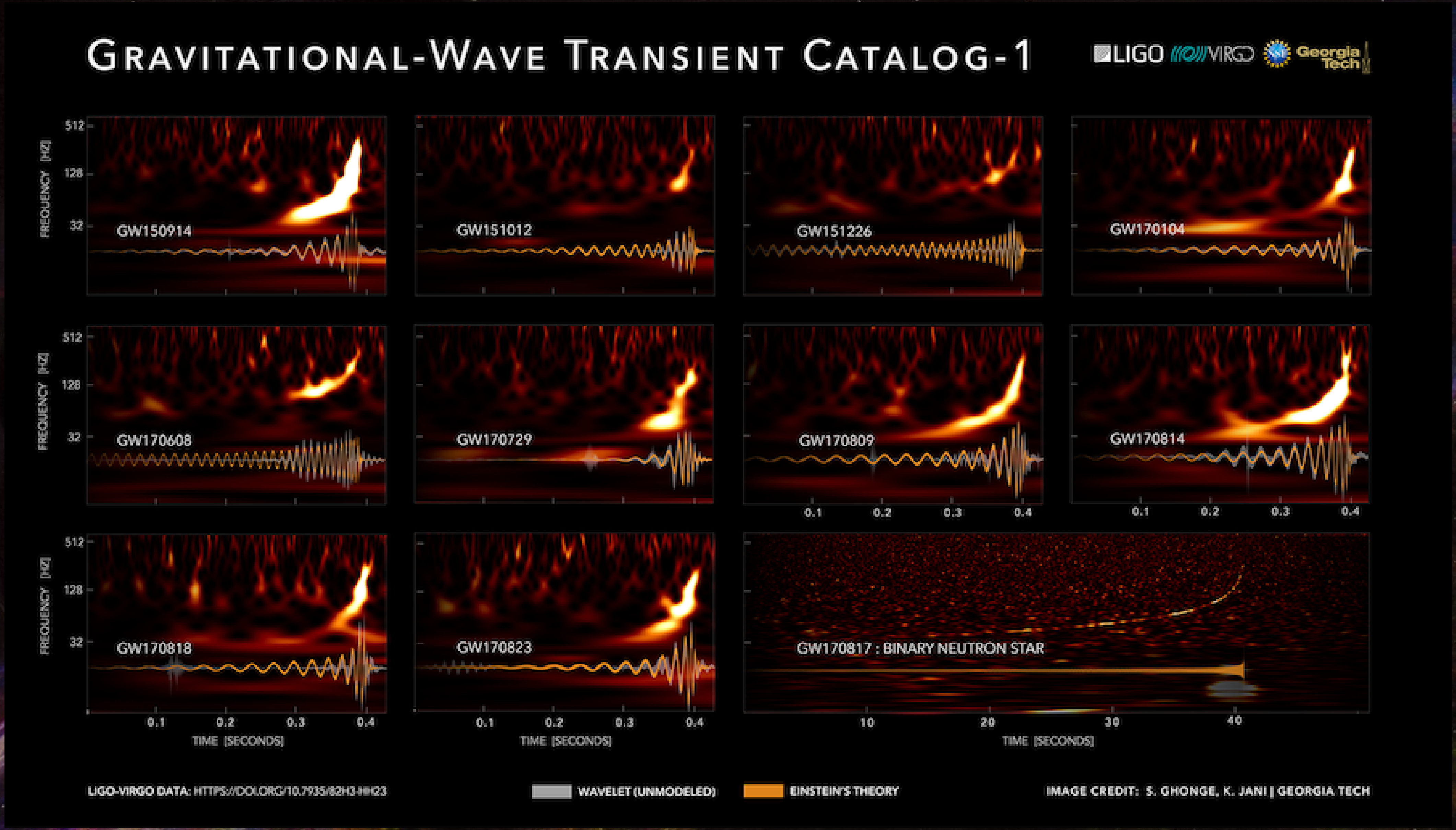
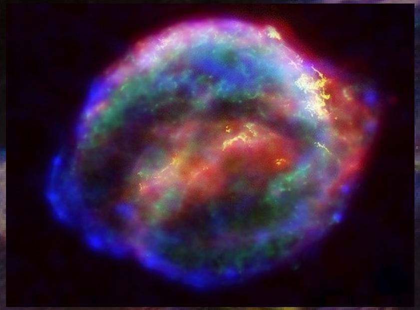
Background reduction in searches for gravitational-wave signals from supernovae: A machine learning approach

Marco Cavaglia (Missouri University of Science and Technology)
with
Sergio Gaudio, Travis Hansen, Marek Szczepanczyk, Michele Zanolin (Embry-Riddle Aeronautical University)
Kai Staats (Northwestern University)


GRAVITATIONAL-WAVE TRANSIENT CATALOG-1
LIGO VIRGO Georgia Tech
FREQUENCY [HZ]
512
128
32
GW150914
GW151012
GW151226
GW170104
GW170608
GW170729
GW170809
GW170814
GW170818
GW170823
GW170817 : BINARY NEUTRON STAR
0.1 0.2 0.3 0.4
10 20 30 40
TIME [SECONDS]
LIGO-VIRGO DATA: HTTPS://DOI.ORG/10.7935/82H3-HH23
WAVELET (UNMODELED)
EINSTEIN'S THEORY
IMAGE CREDIT: S. GHONGE, K. JANI | GEORGIA TECH

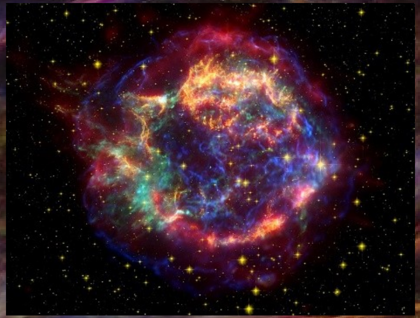
The next big thing could be a core-collapse supernova

- The last supernova seen by human eyes in the Milky Way was in 1604 (Kepler's supernova).



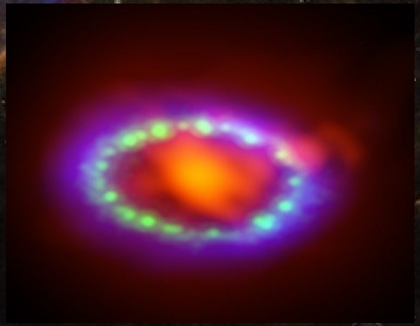
NASA/ESA/JHU/R.Sankrit & W.Blair

- The last supernova known to have occurred in the Milky Way was ~ 300 years ago (Cassiopeia A).



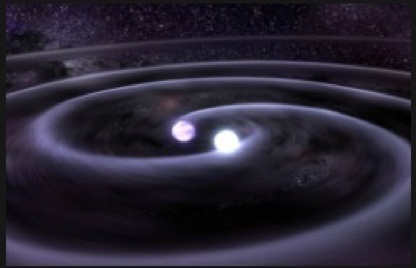
NASA/JPL-Caltech

- Last notable supernova in the vicinity of the Milky Way was in 1987 (SN 1987A)



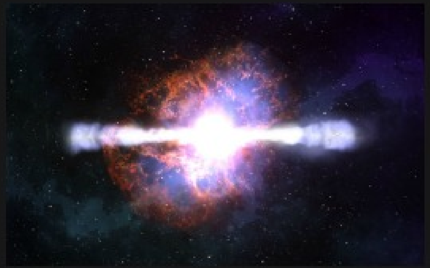
ALMA (ESO/NAOJ/NRAO)/A. Angelich

TYPE I SUPERNOVAE:



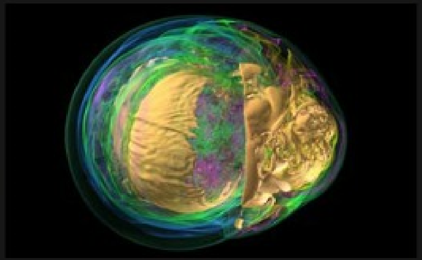
This type of nova takes place in binary star systems, with one of the stars classified as a white dwarf.

The dwarf accretes material from its larger counterpart, accumulating mass as a result. This eventually incites a chain nuclear reaction..



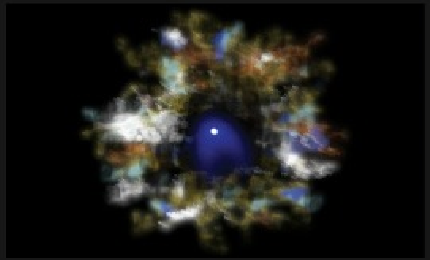
culminating in the star reaching critical density, when it explodes in a supernova. Beams of gamma radiation can also be emitted.

TYPE II SUPERNOVAE:



After losing the abilty to stably fuse heavy elements, the star can no longer retain a gravitational equilibrium, thus the core collapses in on itself.

The core rebounds in quick succession, subsequently releasing the outerlayers of gas off into space — forming a nebula.



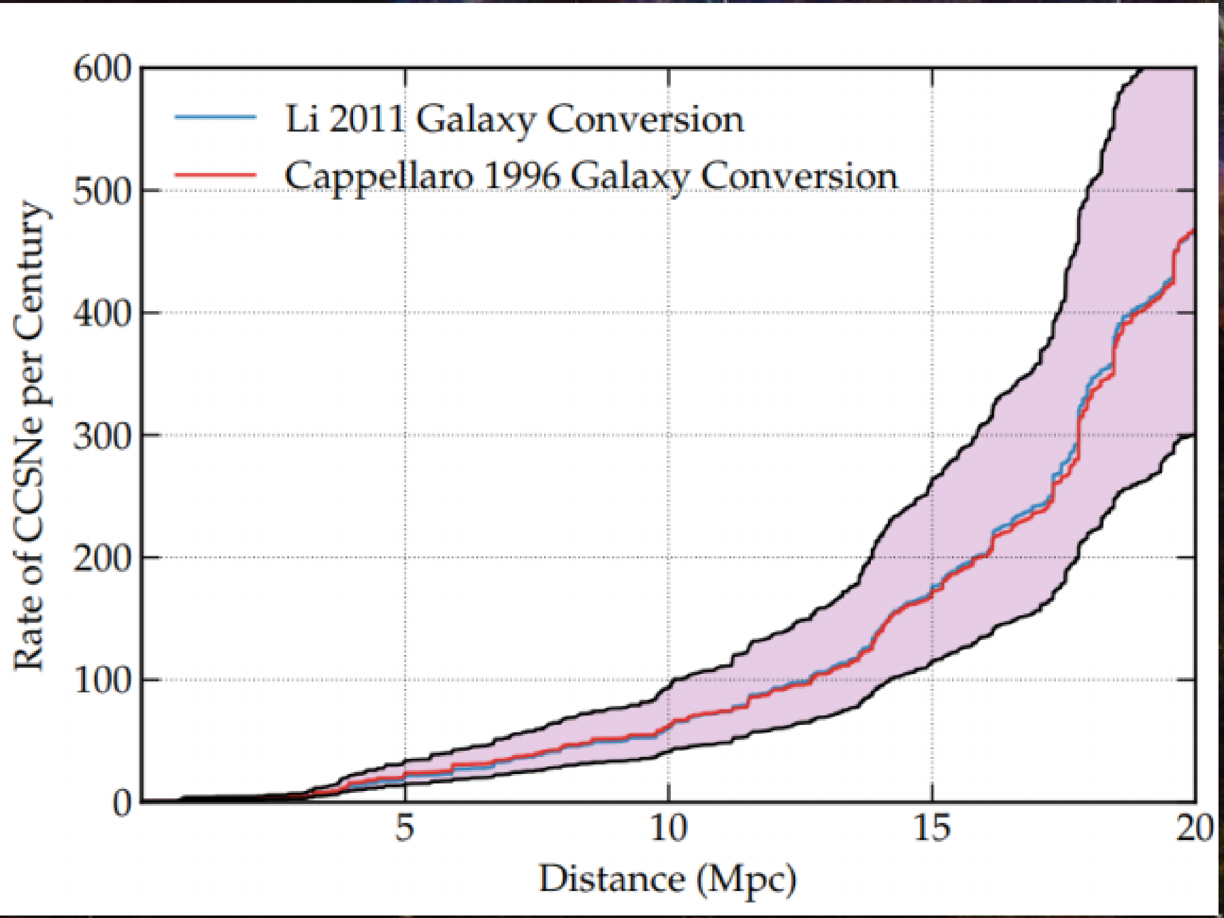
After the dust settles, a neutron star or black hole is left behind (which one will hinge on the star's mass)

Image Credits (Clockwise, from top to bottom): Tod Strohmayer & Dana Berry, Celojums Kosmosa (Vimeo), NASA/Dana Berry/Skyworks Digital, NASA (taken by Chandra)., 'Daftopia' (Deviant Art), Argonne National Laboratory/Hongfeng Yu. (Infographic by' From Quarks to Quasars')

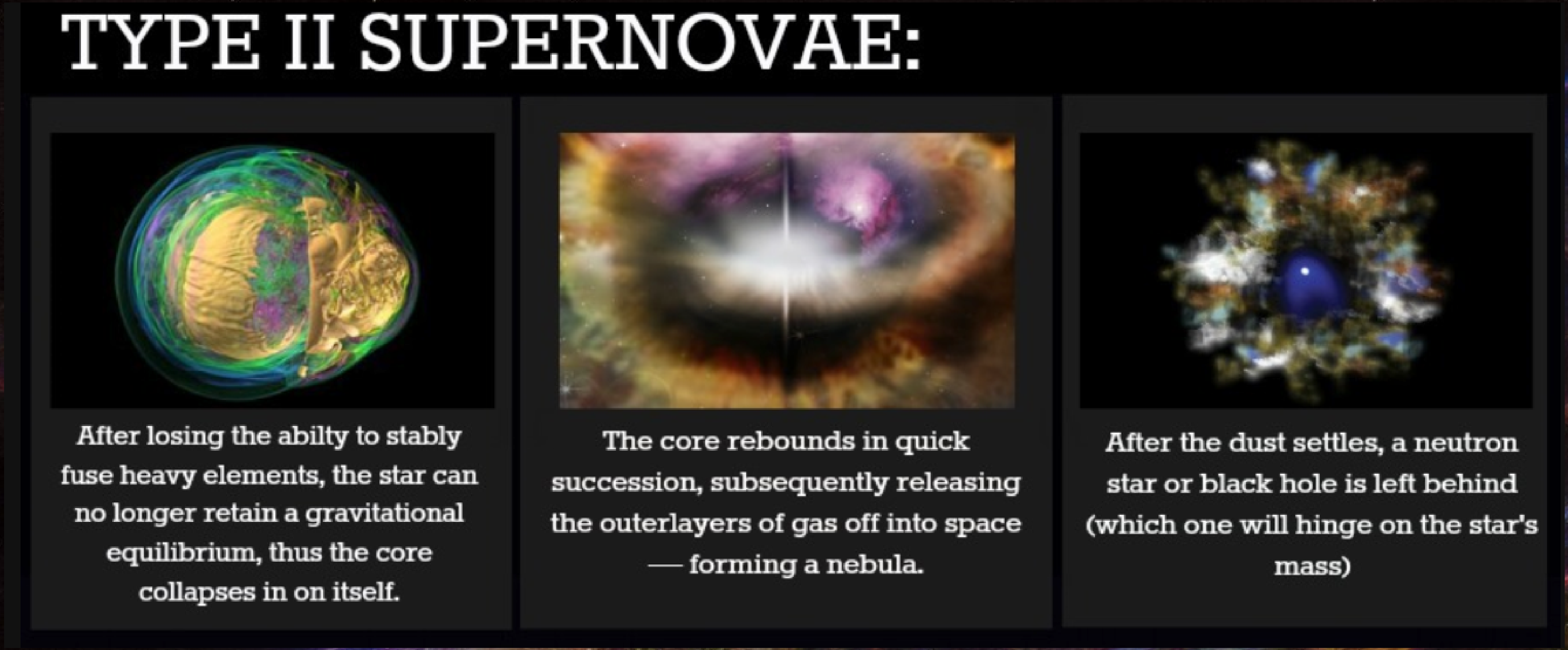
Supernova Rates

- Approx. 1 supernova per day discovered
- Approx. 4 supernovae per year up to 20Mpc
- Approx. 12 SN per century (?) in Milky Way

~ 80% of all supernovae are core collapse supernovae (CCSN)



J. Gill et al, in preparation, courtesy of M. Szczepańczyk.



Before collapse: Iron core of size 1000 ~ 2000 km.

- After collapse: Neutron core of size ~ 50 km.

Energy available: ~ 3×10^{46} J (~ $0.15\ M_{SUN}c^2$).

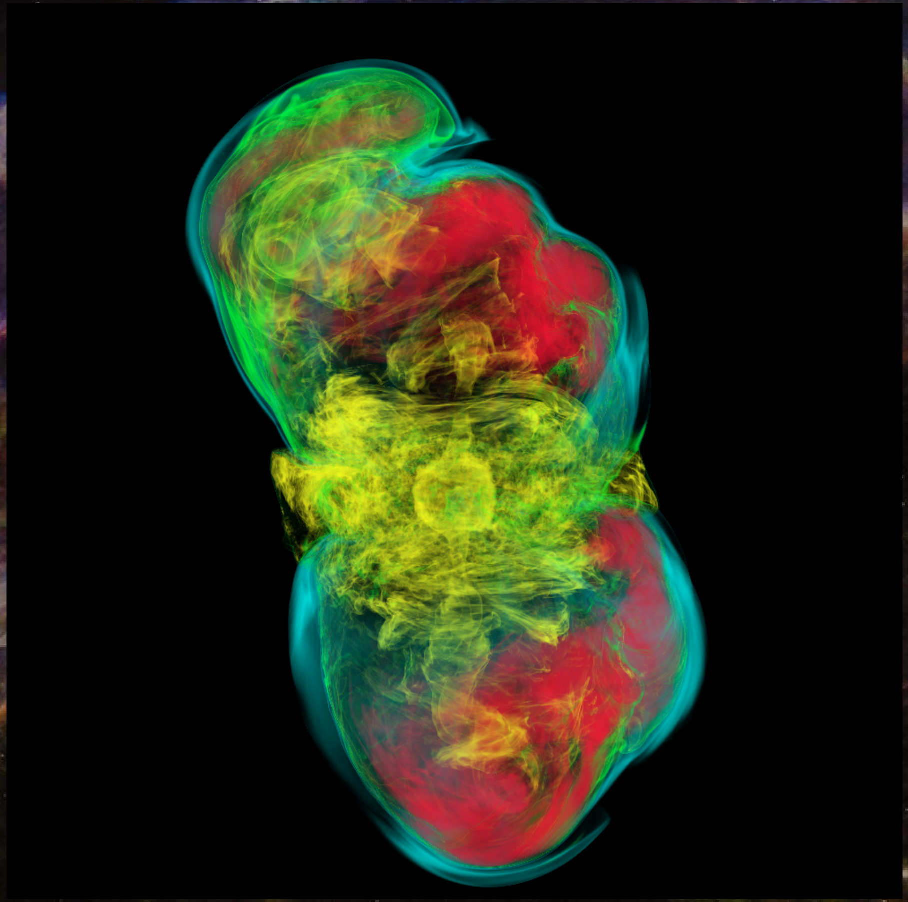
- Energy observed: ~ 3 - 10×10^{44} J. → Neutrinos, GWs!

Gravitational waves from CCSN

$$h_{jk}^{TT}(t,\vec{x}) = \left[\frac{2}{c^4}\frac{G}{|\vec{x}|}\ddot{I}_{jk}\left(t-\frac{|\vec{x}|}{c}\right)\right]^{TT}$$

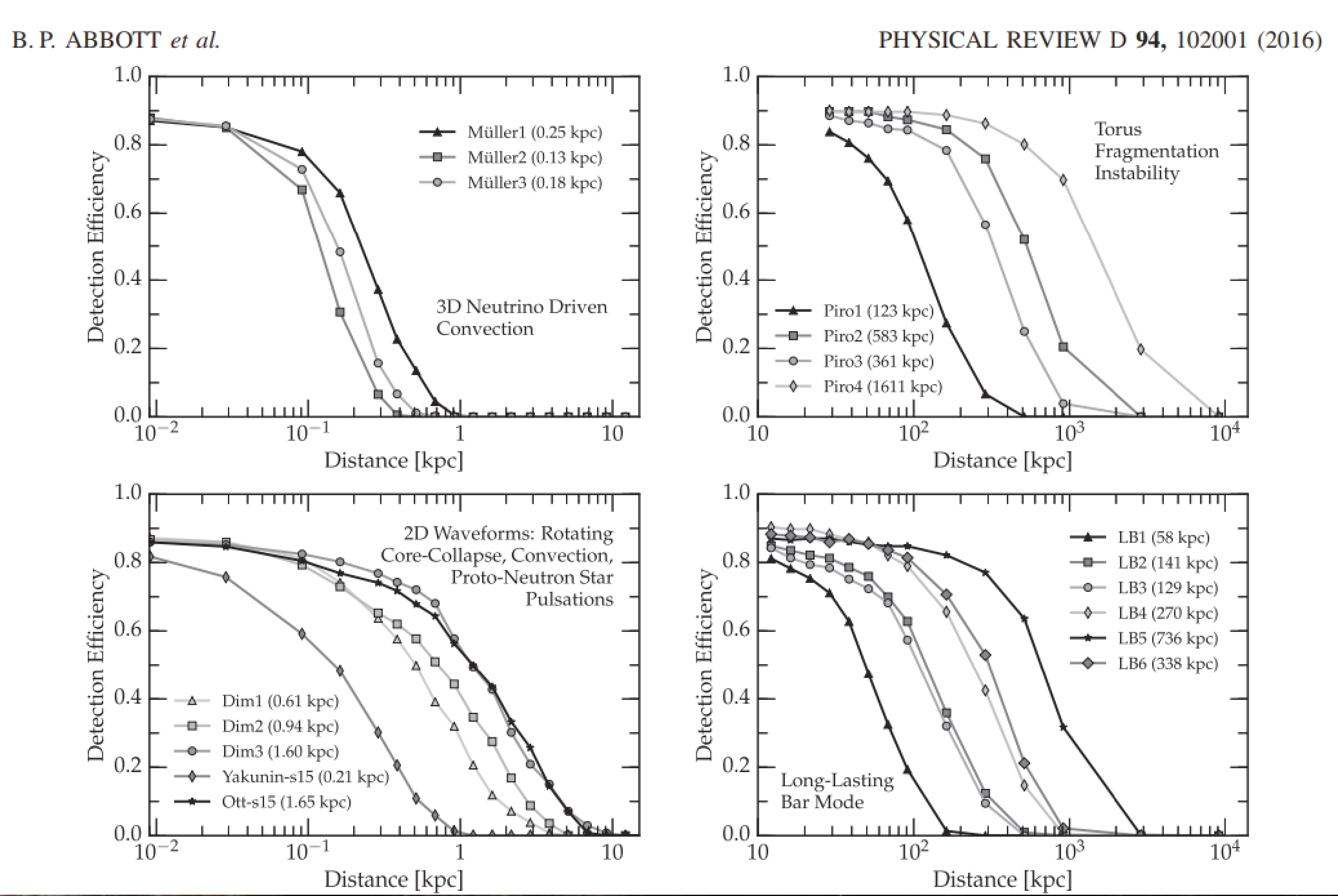
Aspherical mass-energy energy outflow can be produced by:

- Magnetic stresses, 3D MHD instabilities.
- Rotating core collapse and bounce, pulsation of proto-neutron star, convection in proto-neutron star, black hole formation.
- Turbulent convection, anisotropic neutrino emission, Standing Accretion Shock Instability (SASI).



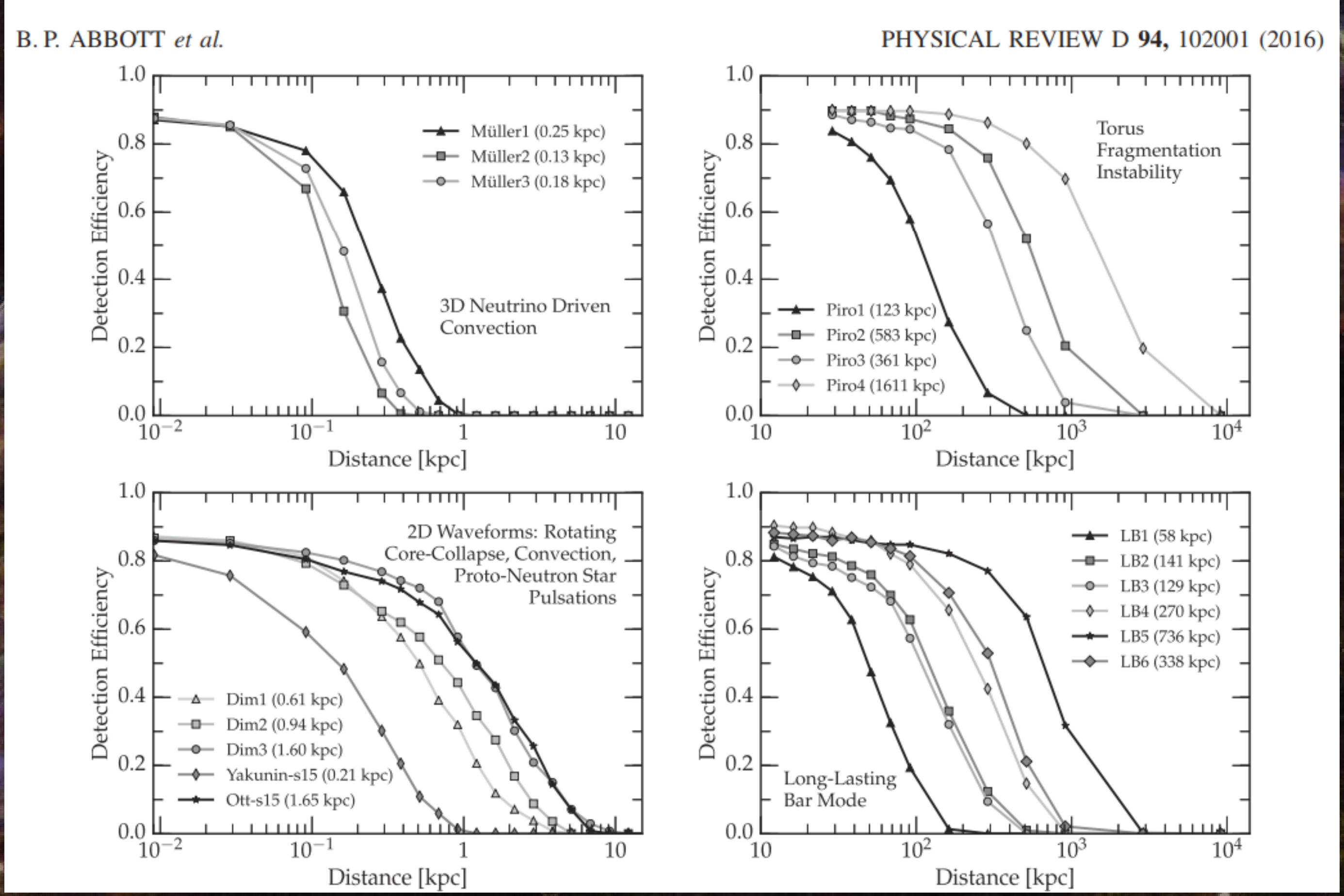
Moesta, TAPIR, California Institute of Technology

Detector range: Initial LIGO



Detector range: Current and future detectors

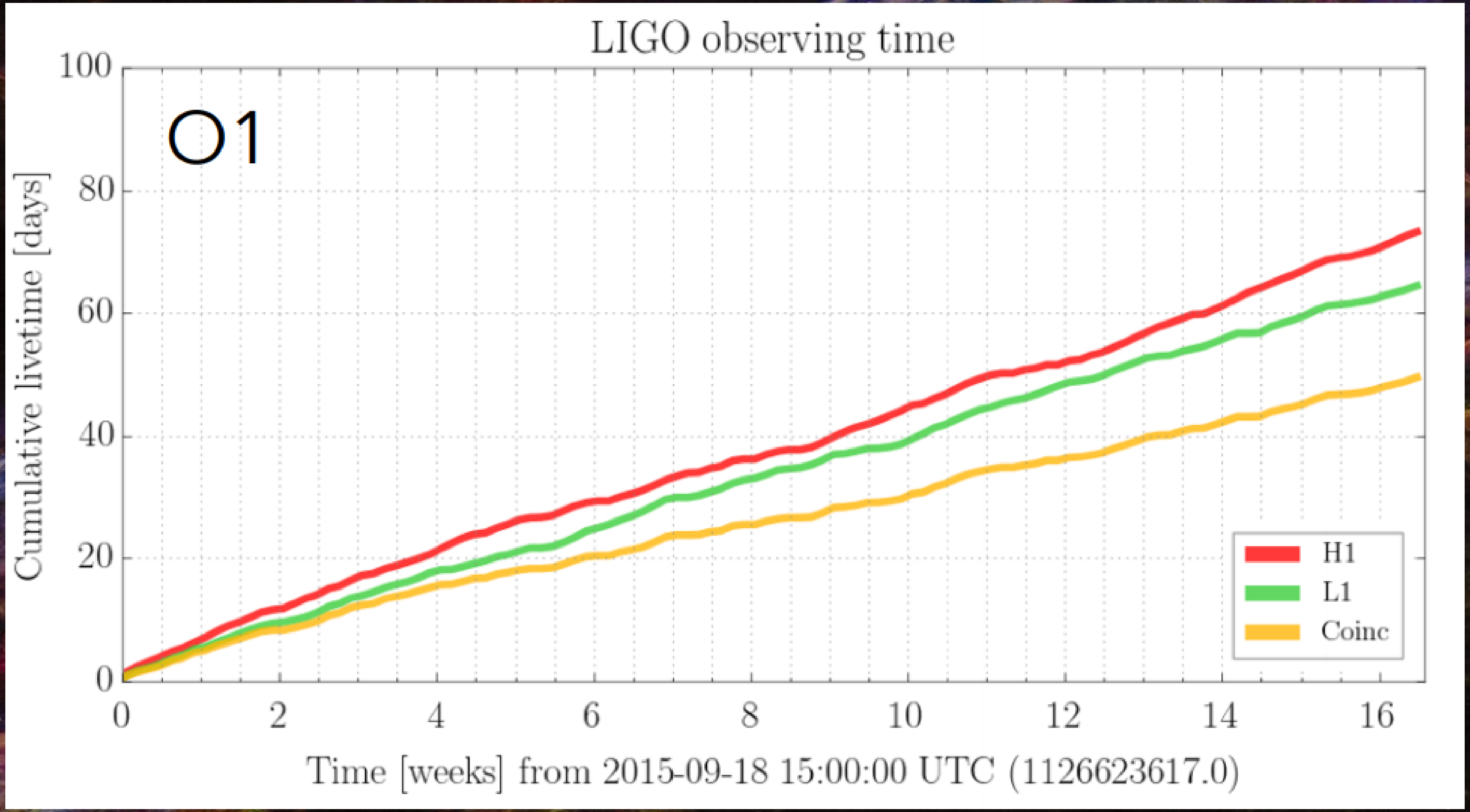
- Center of Milky Way: 8.5 kpc.
- Large Magellanic Cloud: 50 kpc.
- Andromeda: 780 kpc.
- Virgo Cluster: ~ 15 - 25 Mpc.
- 2G detectors: ~ 10x detector sensitivity.
- 3G detectors: ~ 100x detector sensitivity.



It is only a matter of time until we detect the first GW signal from a CCSE. We should be prepared to confidently claim detection and extract as much information as possible.

Maximizing detection chances

- In the first Advanced LIGO run (O1), about 40% of the data were collected in single interferometer (single-IFO) mode.



Courtesy of LIGO Scientific Collaboration. Detector characterization summary pages.

Maximizing detection chances

- In the first Advanced LIGO run (O1), about 40% of the data were collected in single interferometer (single-IFO) mode.
- In the next few years we expect ~ 20% of the data to be single-IFO data.
- We would like to be able to search data and claim a galactic CCSN detection even with only one instrument.
- However: GW searches of CCSN are (essentially) unmodelled. Single-IFO data are espected to have a large population of loud glitches, which affect detection confidence.
- Optical CCSN counterpart and clever analysis techniques may allow us to do so.

Improving single-IFO detection

- We want to develop a method that integrates with current LIGO-Virgo search pipelines. We focus on LIGO because of the higher detector sensitivity.

- Perform a two interferometer strategy search, where data from a LIGO interferometer is an exact copy of the data from the second. This allows for an immediate deployment in existing pipelines [coherent WaveBurst – cWB, S. Klimenko, et. al, Class. Quantum Grav. 25 (2008) 114029].

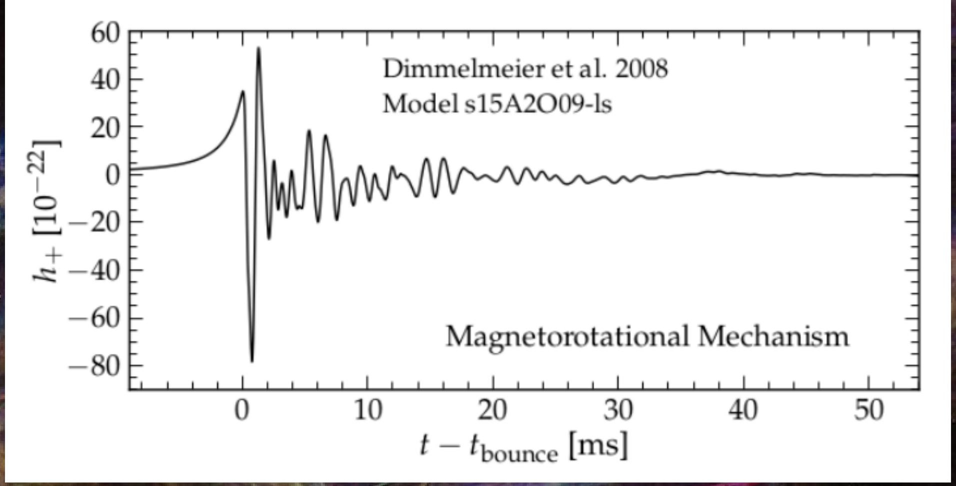
- Even for the most realistic emission models in the currently adopted pool for LIGO-Virgo searches, a galactic center detection is out of reach at 3σ c.l. with the current implementation of cWB.

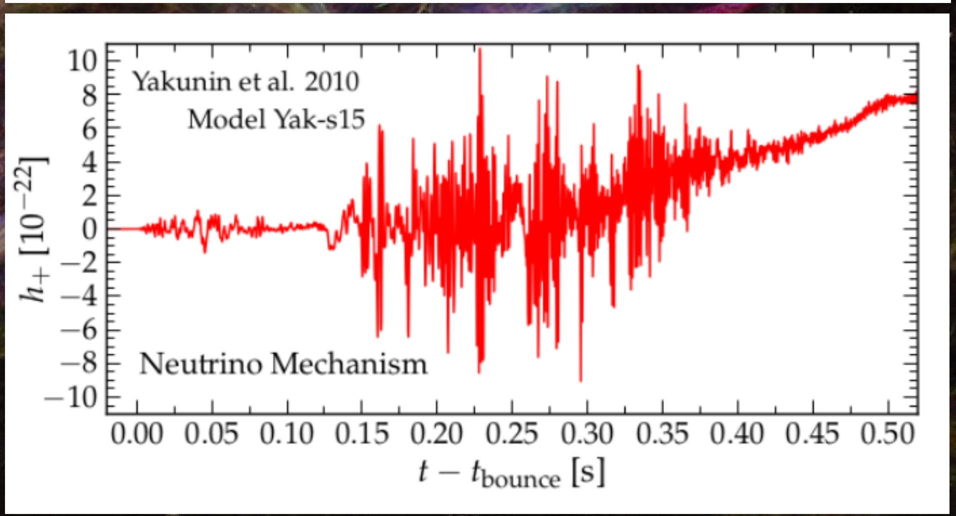
- Explore the possibility of reducing the population of background glitches with machine learning. Goals: (1) 3σ c.l. detection; (2) Remove background events without decreasing efficiency curves.

CCSN waveforms: Basic properties

Waveforms generally exhibit:

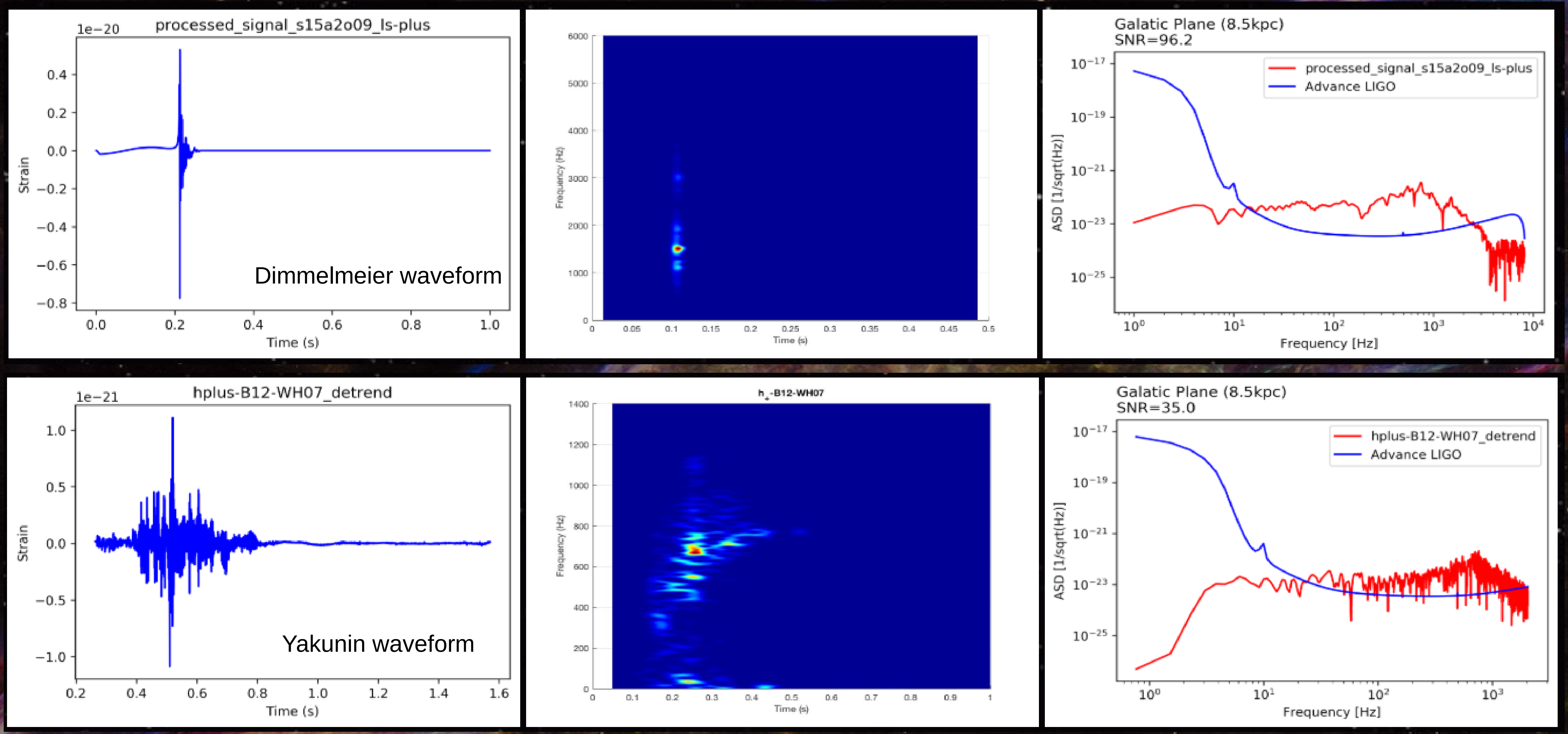
- Broadband and long duration signal
- Strong high-frequency component
- Non deterministic shape





Very complex problem!

CCSN waveforms: Basic properties

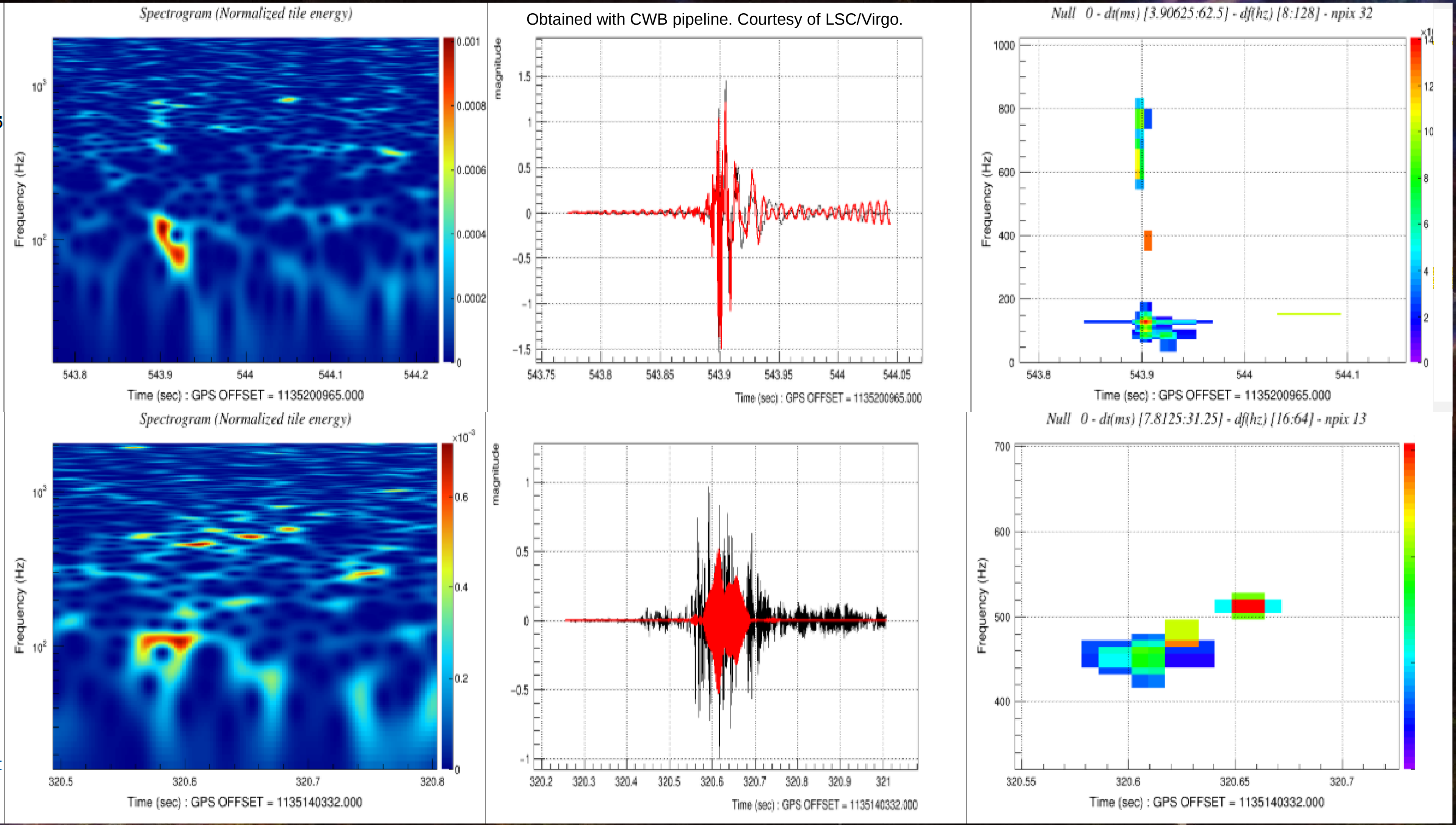


Courtesy of http://mercury.pr.erau.edu/~quinonep/yakunin.html

CCSN waveforms in real data

GPS TIME:
1135201508.922485
WAVEFORM: **dim1**
DISTANCE: **1 Kpc**
SNR: **11.9**
DURATION: **0.020**
PEAK FREQUENCY: **337.9**

GPS TIME:
1135140652.627
WAVEFORM: **yak3**
DISTANCE: **1.78 Kpc**
SNR: **8.1**
DURATION: **0.021**
PEAK FREQUENCY: **478.3**

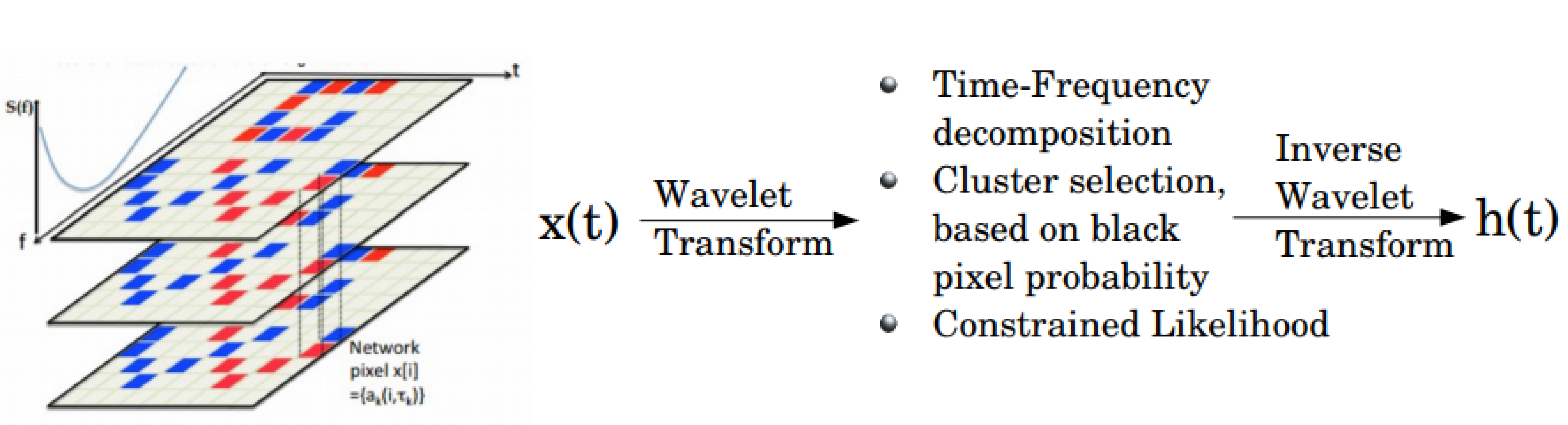


cWB pipeline

- Wavelet-based burst-detection method that combines all data streams into one coherent statistic [S. Klimenko et. al, Class. Quantum Grav. 25 (2008) 114029].

- Detection is based on the maximum likelihood ratio statistic which represents the total signal-to-noise ratio of the GW signal detected in the network.

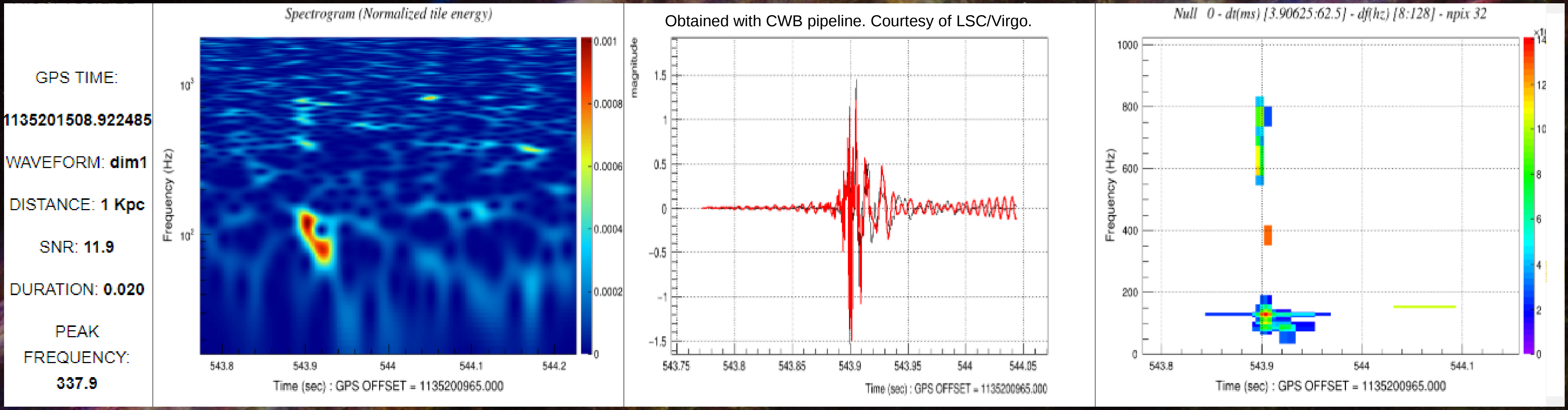
- It allows <u>waveform reconstruction</u> and <u>parameter extraction</u>.



Courtesy of M. Szczepanczyk

cWB pipeline

- Wavelet-based burst-detection method that combines all data streams into one coherent statistic [S. Klimenko et. al, Class. Quantum Grav. 25 (2008) 114029].
- Detection is based on the maximum likelihood ratio statistic which represents the total signal-to-noise ratio of the GW signal detected in the network.
- It allows waveform reconstruction and parameter extraction.



Implementation

- We use cWB with two identical LIGO interferometers.
- Background triggers and CCSN simulation triggers in real LIGO-Virgo data.
- Two days of background data (second observing run).
- Inject different waveform models (Dimmelmeier, Yakunin,…) at specific galactic distances (*)
- Rather than using waveforms and/or scalograms (time consuming) we use only cWB extracted parameters.

(*) From the CCSN optical counterpart we should know distance and localization.

cWB parameters

- cWB ranking statistics (effectively SNR).
- Number of wavelet time-frequency pixels composing the events.
- Energy-weighted duration estimated in time-frequency domain.
- Difference between event stop and start.
- Central frequency of the event computed from the reconstructed waveform.
- Energy-weighted central frequency estimated in time-frequency domain.
- Minimum frequency associated to the time-frequency map pixels.
- Maximum frequency associated to the time-frequency map pixels.
- Energy-weighted bandwidth (frequency resolution) estimated in time-frequency domain.
- Difference between high and low bandwidth values.
- Norm Factor or ellipticity.

Injections

Waveforms:

- Mueller-2012: arXiv:1210.6984 [astro-ph.SR].
- Yakunin-2015: arXiv:1505.05824 [astro-ph.HE].
- Ott-2013: arXiv:1210.6674 [astro-ph.HE].
- Scheidegger-2010: arXiv:1001.1570 [astro-ph.HE].
- Dimmelmeier-2008: arXiv:0806.4953 [astro-ph].

Distances:

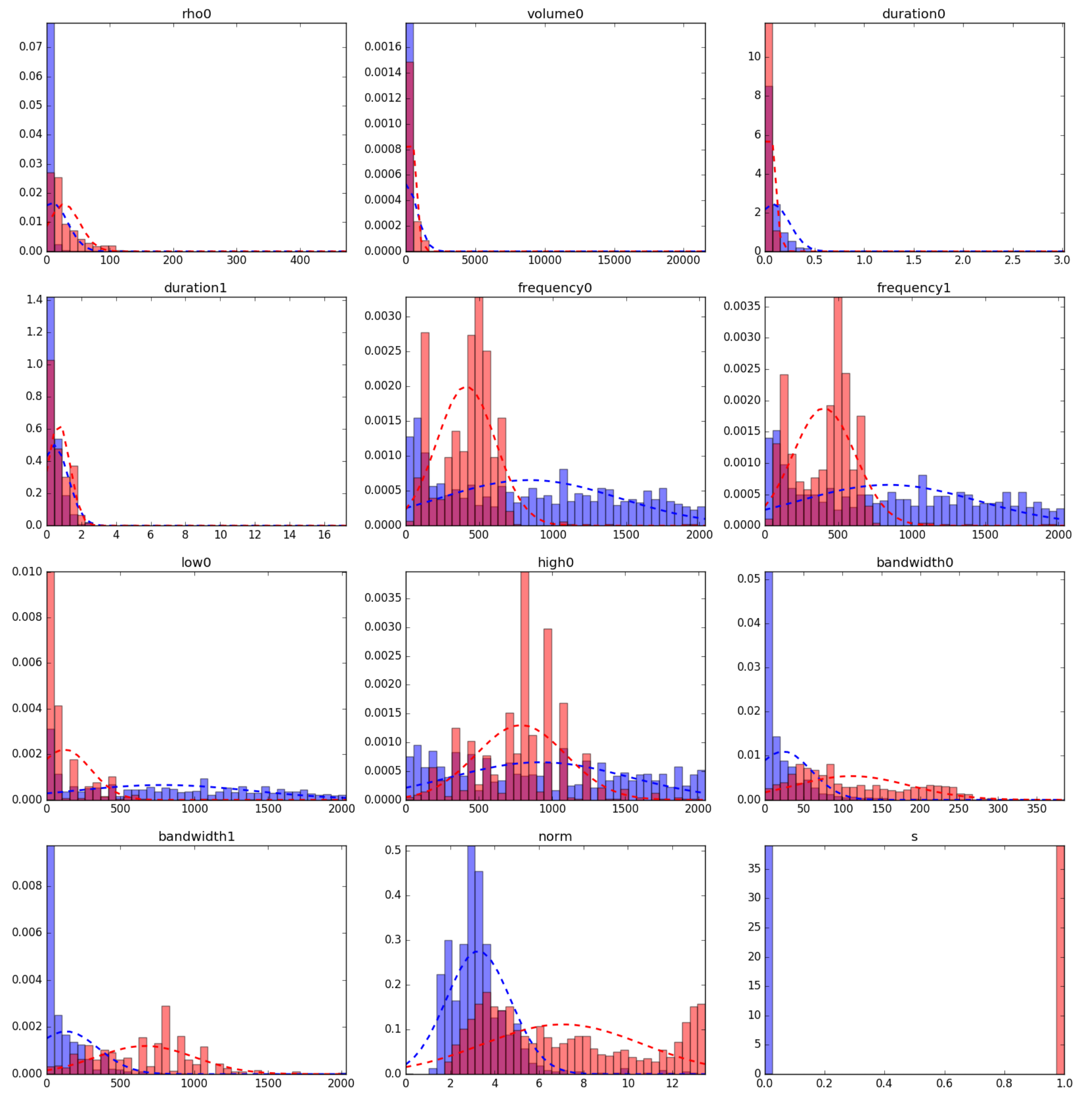
- 1.0 kpc, 1.78 kpc, 3.16 kpc, 4.22 kpc, 5.62 kpc.

Number of injections:

- From ~ few hundreds to ~ one thousand each.
- Comparable number of background triggers.

Example:

- All waveform models combined.
- Distance: 4.22 kpc



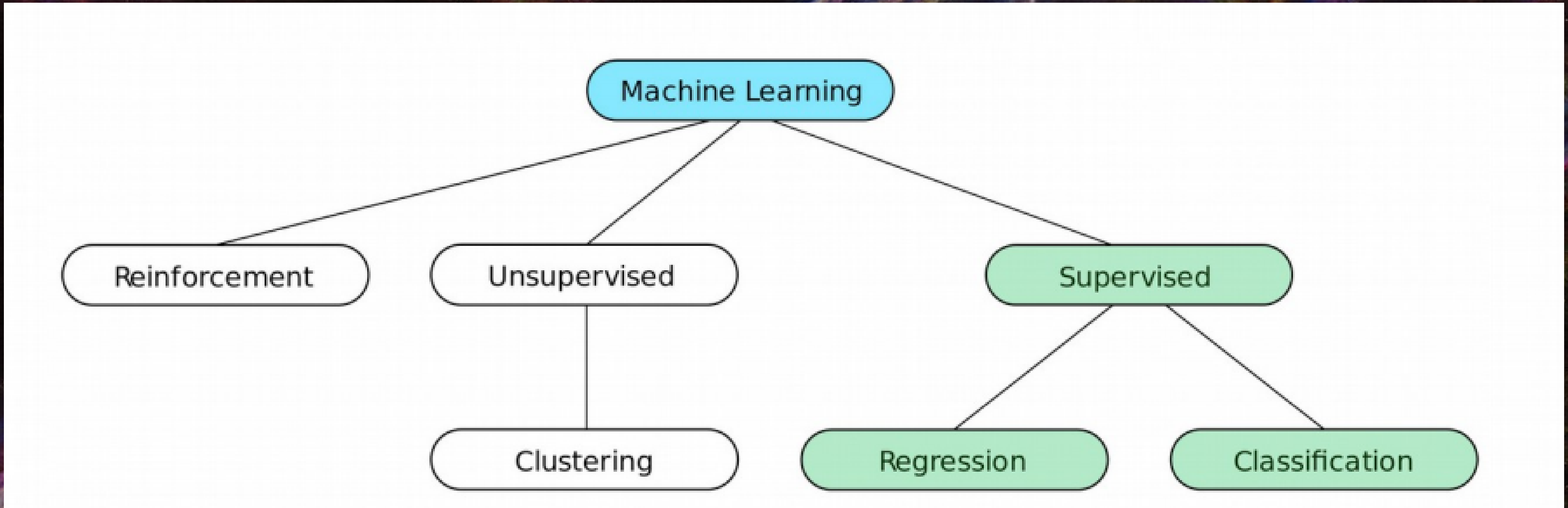
M. Cavaglia et al, in preparation.

Background reduction

- We use a machine learning (ML) approach for background reduction.
- We want to apply it to the cWB output. No fancy new method. Easily deployable.
- Dataset is numerical (not images). No need for deep learning, convolutional neural networks,… Maybe in the future, for more refined results.
- Genetic Programming.
- Method can be straightforwardly extended to include information from auxiliary channels or to the multi-interferometer case.

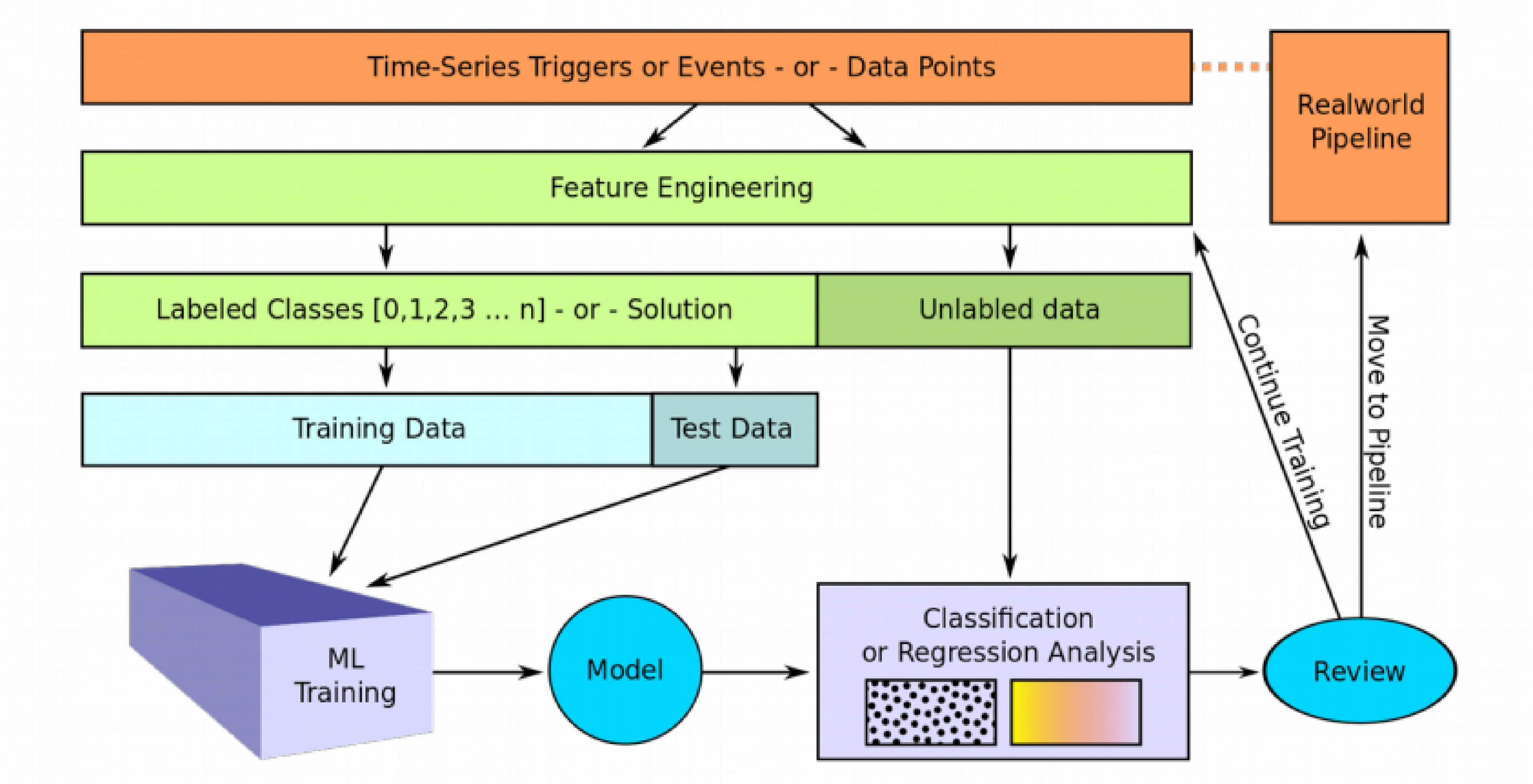
Brief introduction to Machine Learning

When the dimensionality and/or volume of data is too great for humans, we need computer algorithms to help us discover and understand the data trends.



Courtesy of K. Staats.

Training vs. testing vs. real world



Courtesy of K. Staats.

Genetic Programming

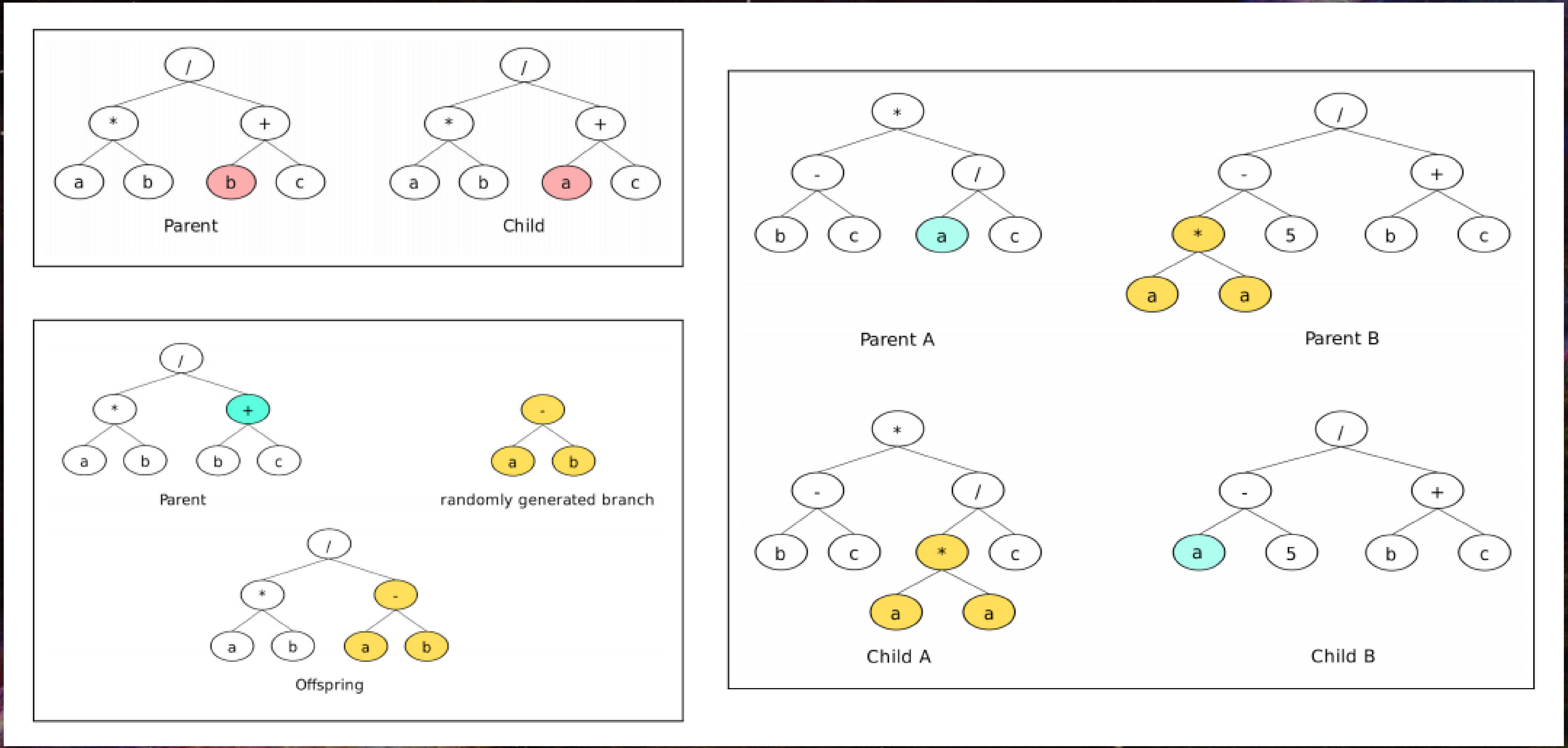
- Idea dates back to Alan Turing. Developed by John Koza et al. in the 1980s.
- A supervised, evolutionary algorithm.
- Uses random mutation, selection, a fitness function, and multiple generations of evolution to resolve a user-defined task.
- Produces human-readable correlations between features (variables) in datasets.
- Can perform classification and regression.
- Applications: Predictive modeling, data mining, financial modeling, image processing,...

Basics of evolutionary algorithm

- Generate an initial, stochastic population of individuals.
- Evolve the individuals:
  - Evaluate each individual in the current population against the dataset and evaluate its fitness.
  - Randomly select a number of individuals and compare their fitness scores.
  - Apply genetic operators to the leading individual (reproduction, mutation, crossover).
- Evaluate all individuals in each new generation.
- Repeat until the user-defined termination criteria are met.

In GP, each individual is a program. Generations are composed of a population of individual programs. Each program is a mathematical function that when executed against the given data, produces a value.

Genetic operators



M. Cavaglia, K. Staats and T. Gill, Commun. Comput. Phys., 25 (2019), 963-987.

Our problem

For our analysis we implement the “Karoo GP” code

- (K. Staats: https://github.com/kstaats/karoo_gp).

```
Tree 11 yields (sym): a305 + a306*a78*r119 + a43*f49 + a62 - f112 + f12 + f217
Tree 12 yields (sym): -a18 + f190*r284 - q151*r284*r75/r188 - q155
Tree 13 yields (sym): -a18 - a234*r12*r284 + f190*r284 - q155 + q31
Tree 14 yields (sym): -a18 - a234*q151*r284 + f190*q266 - q155 + q31
Tree 15 yields (sym): a306*a78*r119 + a43*f49 + a62 + f102*f82 + f12 - q187 + r20
Tree 16 yields (sym): a154*a41*f82 - a198*q291*r111*r206/q76
Tree 17 yields (sym): a43*f49 + a62 - f112 + f12 + f217 + f57 + q193*r119 + r20
Tree 18 yields (sym): a258 + f131*f272*q41 - q251 + a229/(a125*q121)
Tree 19 yields (sym): -a18 - f124*q151*r284 + f19 + f190*r284 - q155
Tree 20 yields (sym): -a234*q151*r284 + f190*r284 - q155 + q31 - r90
Tree 21 yields (sym): -a18 - a234*r115*r284 + f190*r284 - q155 + q31
Tree 22 yields (sym): -a18 - a234*q151*r284 + f190*r284 + q31 - r289
Tree 23 yields (sym): a151/f28 + f134 + f245 + f5 + q270 - r164 - r318
Tree 24 yields (sym): -a172 + f131*f272*q41 + f68*f96*q169/f182
Tree 25 yields (sym): a154*a41*f36
Tree 26 yields (sym): a306*a78*r119 + a43*f49 + a62 - f112 + f217 + r20 + r57/a99
Tree 27 yields (sym): a161 + a162/(f225*r245) - a287*f146*q9*r0 + f258*f337*q117 - f324 + q282 + r274 + f32/f318
Tree 28 yields (sym): -a18 - a234*q151*r284 + f190*r284 - q155 + q162
Tree 29 yields (sym): -a18 - a234*q151*r284 + a65*r284 - q155 + q31
Tree 30 yields (sym): -a266/a76 + a274 + a85 + f206 - q160*r57 - q295
Tree 31 yields (sym): f245 + q270 - r164 + r284 - r318
Tree 32 yields (sym): a151*f190/a338 - a18 - a234*q151*r284 + f134 + f5 - q155 + q31
Tree 33 yields (sym): a306*a78*r119 + a43*f49 + a62 - f112 + f12 + f217 + r318
Tree 34 yields (sym): a109*r115 + a277*a306*q125*r87/q29 - a40*r20/q175 + f18*f241*q268 - f282 + q75 + r247
Tree 35 yields (sym): a161 - a287*f146*f82*q9 + f258*f337*q117 + f32/r138 - f324 + q282 + r274 -
Tree 36 yields (sym): a102*a162*q291*r206/q76 + a154*a41*r0 - f123
Tree 37 yields (sym): a234 + a306*a78*r119 + a62 - f112 + f12 + f217 + q31 + r20
Tree 38 yields (sym): -a18 - a43*f49*q151*r284 + f190*r284 - q155
Tree 39 yields (sym): a274 + a85 - f131*f272*r57 + f206 - q295 - f267/a76
Tree 40 yields (sym): q160*q41 - q251 + f68*f96*q169/f182
Tree 41 yields (sym): a154*a41*f82 - f190*q291*r111*r206*r284/q76
Tree 42 yields (sym): a102 - a18 - a234*q151*r284 - f123 - q155 + q31
Tree 43 yields (sym): -a18 - a234*q151*r284 + a78*f190 - q155 + q31
Tree 44 yields (sym): a306*r119*r284 + a43*f49 + a62 - f112 + f12 + f217 + r20
Tree 45 yields (sym): f272*q41*r284 - q251 + f68*f96*q169/f182
Tree 46 yields (sym): -a18 - a234*f131*q151 + f190*r284 - q155 + q31
Tree 47 yields (sym): -a102 + a274 + a85 - f123 + f206 - q295 - f267/a76
Tree 48 yields (sym): a154*a41*f82 - q160*q291*r111*r206*r57/q76
Tree 49 yields (sym): f131*f272*q41 + q151 - q251
Tree 50 yields (sym): -a18 - a234*f68*f96*q169*r284/f182 + f190*r284 - q155 + q31
Tree 51 yields (sym): -a18 - a234*q151*r284 + f272 - q155 + q31
Tree 52 yields (sym): f131*f190*q41*r284 - q251 + f68*f96*q169/f182
Tree 53 yields (sym): -a102*q291*r111*r206/q76 + a154*a41*f82 - f123
```

A Classification run

- Depth: 10
- Operators: +, -, *, /
- Population: 300
- Generations: 100-150
- Selection: Tournament (10%)
- Total runs: 160

Courtesy of K. Staats.

Tree: -a1129 + 19*a1258 + a1262 + a1263 + a1272 + 4*a1279 + 5*a1359 - 5*a1757 - a2089 - 6*a2197 + a2200 - 2*a2233 + 2*a2235 + 6*a2359 + a2757 - a2856 - a299 + 5*a3041 - 3*a3492 + a483

Classification report:

	precision	recall	f1-score	support
0.0	0.93	0.98	0.96	223
1.0	0.98	0.93	0.96	234
avg / total	0.96	0.96	0.96	457

Confusion matrix:

	other	glitch
0 other	219	4
1 glitch	16	218

We run the code multiple times for better statistics.

Machine learning lingo

<table>
<tr><td></td><td></td><td colspan="2">True condition</td><td></td><td colspan="2"></td></tr>
<tr><td></td><td>Total population</td><td>Condition positive</td><td>Condition negative</td><td>Prevalence $= \frac{\Sigma \text{ Condition positive}}{\Sigma \text{ Total population}}$</td><td colspan="2">Accuracy (ACC) = $\frac{\Sigma \text{ True positive} + \Sigma \text{ True negative}}{\Sigma \text{ Total population}}$</td></tr>
<tr><td rowspan="2">Predicted condition</td><td>Predicted condition positive</td><td>True positive, Power</td><td>False positive, Type I error</td><td>Positive predictive value (PPV), Precision = $\frac{\Sigma \text{ True positive}}{\Sigma \text{ Predicted condition positive}}$</td><td colspan="2">False discovery rate (FDR) = $\frac{\Sigma \text{ False positive}}{\Sigma \text{ Predicted condition positive}}$</td></tr>
<tr><td>Predicted condition negative</td><td>False negative, Type II error</td><td>True negative</td><td>False omission rate (FOR) = $\frac{\Sigma \text{ False negative}}{\Sigma \text{ Predicted condition negative}}$</td><td colspan="2">Negative predictive value (NPV) = $\frac{\Sigma \text{ True negative}}{\Sigma \text{ Predicted condition negative}}$</td></tr>
<tr><td></td><td></td><td>True positive rate (TPR), Recall, Sensitivity, probability of detection $= \frac{\Sigma \text{ True positive}}{\Sigma \text{ Condition positive}}$</td><td>False positive rate (FPR), Fall-out, probability of false alarm $= \frac{\Sigma \text{ False positive}}{\Sigma \text{ Condition negative}}$</td><td>Positive likelihood ratio (LR+) $= \frac{\text{TPR}}{\text{FPR}}$</td><td rowspan="2">Diagnostic odds ratio (DOR) = $\frac{\text{LR+}}{\text{LR−}}$</td><td rowspan="2">$F_1$ score = $\frac{2}{\frac{1}{\text{Recall}} + \frac{1}{\text{Precision}}}$</td></tr>
<tr><td></td><td></td><td>False negative rate (FNR), Miss rate $= \frac{\Sigma \text{ False negative}}{\Sigma \text{ Condition positive}}$</td><td>True negative rate (TNR), Specificity (SPC) $= \frac{\Sigma \text{ True negative}}{\Sigma \text{ Condition negative}}$</td><td>Negative likelihood ratio (LR−) = $\frac{\text{FNR}}{\text{TNR}}$</td></tr>
</table>

Credit: wikipedia

Condition Positive (P): the number of real positive cases in the data (signals)

- Condition Negative (N): the number of real negative cases in the data (glitches)
- True Positive (TP): hit, i.e., signal correctly identified
- True Negative (TN): correct rejection, i.e., glitch correctly identified
- False Positive (FP): false alarm or Type I error, i.e., glitch misidentified as a signal
- False negative (FN): miss or Type II error, i.e., signal misidentified as a glitch

Machine learning lingo

- For best background recognition we want to maximize Specificity (or TNR), i.e., recognize the maximum number of glitches and minimize FNR (or Miss Rate), i.e., the false negative alarm rate.

- For best signal recognition we want to maximize Recall (or TPR or Sensitivity), i.e., recognize the maximum number of signals and minimize Fall-out (or FPR), i.e., the false positive alarm rate.

sensitivity, recall, hit rate, or true positive rate (TPR)

$$\mathrm{TPR} = \frac{\mathrm{TP}}{P} = \frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FN}} = 1 - \mathrm{FNR}$$

specificity, selectivity or true negative rate (TNR)

$$\mathrm{TNR} = \frac{\mathrm{TN}}{N} = \frac{\mathrm{TN}}{\mathrm{TN}+\mathrm{FP}} = 1 - \mathrm{FPR}$$

precision or positive predictive value (PPV)

$$\mathrm{PPV} = \frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FP}}$$

negative predictive value (NPV)

$$\mathrm{NPV} = \frac{\mathrm{TN}}{\mathrm{TN}+\mathrm{FN}}$$

miss rate or false negative rate (FNR)

$$\mathrm{FNR} = \frac{\mathrm{FN}}{P} = \frac{\mathrm{FN}}{\mathrm{FN}+\mathrm{TP}} = 1 - \mathrm{TPR}$$

fall-out or false positive rate (FPR)

$$\mathrm{FPR} = \frac{\mathrm{FP}}{N} = \frac{\mathrm{FP}}{\mathrm{FP}+\mathrm{TN}} = 1 - \mathrm{TNR}$$

false discovery rate (FDR)

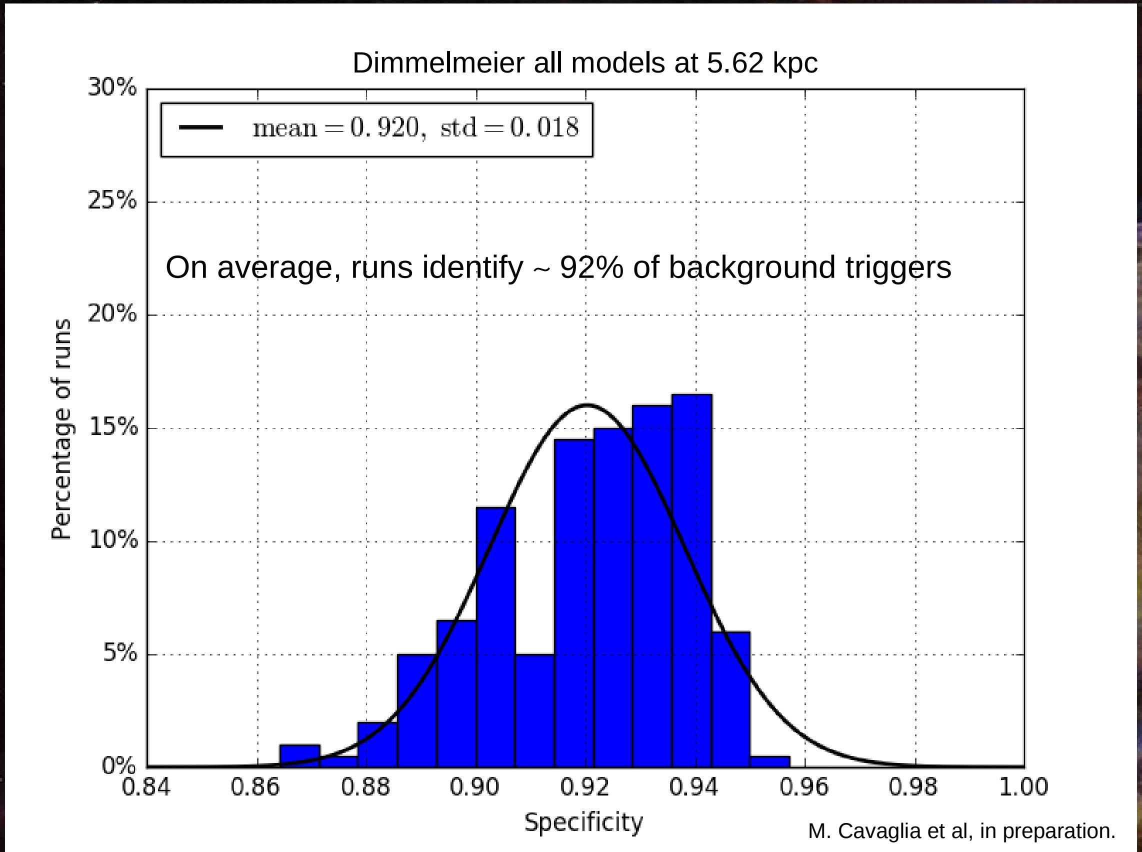
$$\mathrm{FDR} = \frac{\mathrm{FP}}{\mathrm{FP}+\mathrm{TP}} = 1 - \mathrm{PPV}$$

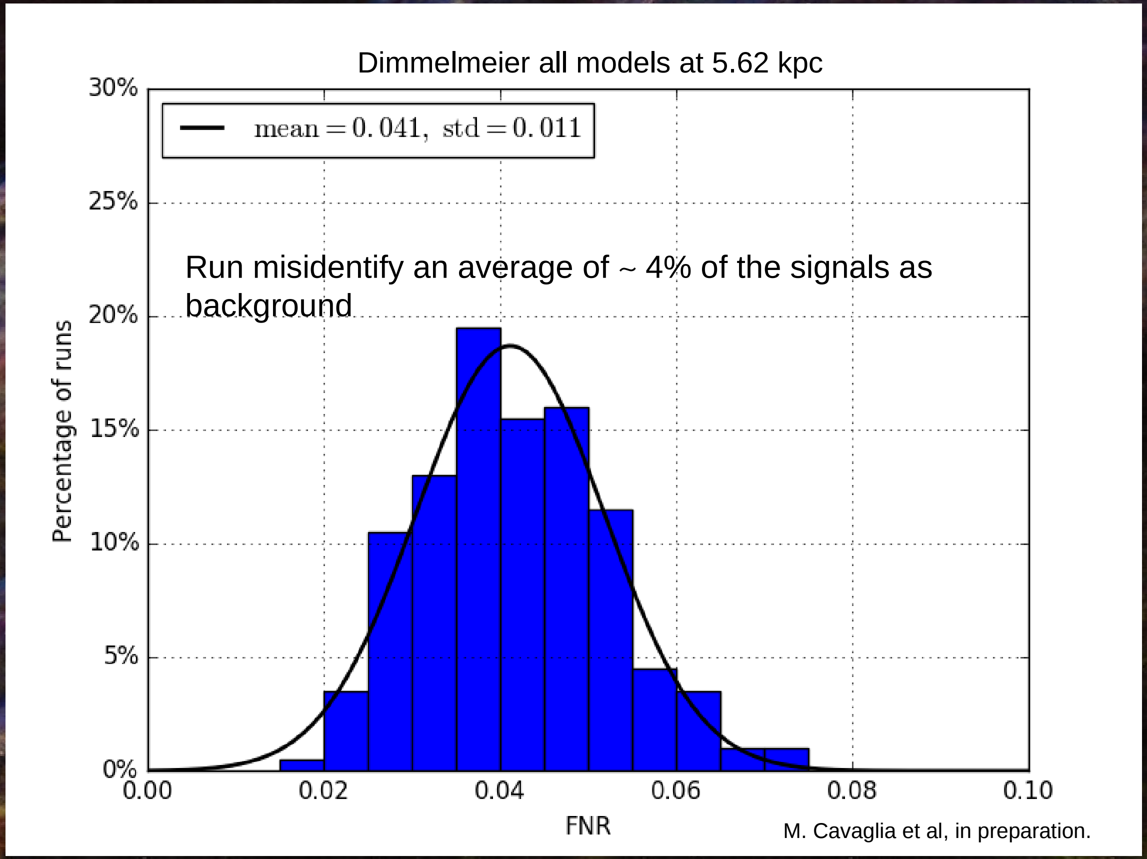
false omission rate (FOR)

$$\mathrm{FOR} = \frac{\mathrm{FN}}{\mathrm{FN}+\mathrm{TN}} = 1 - \mathrm{NPV}$$

Credit: wikipedia

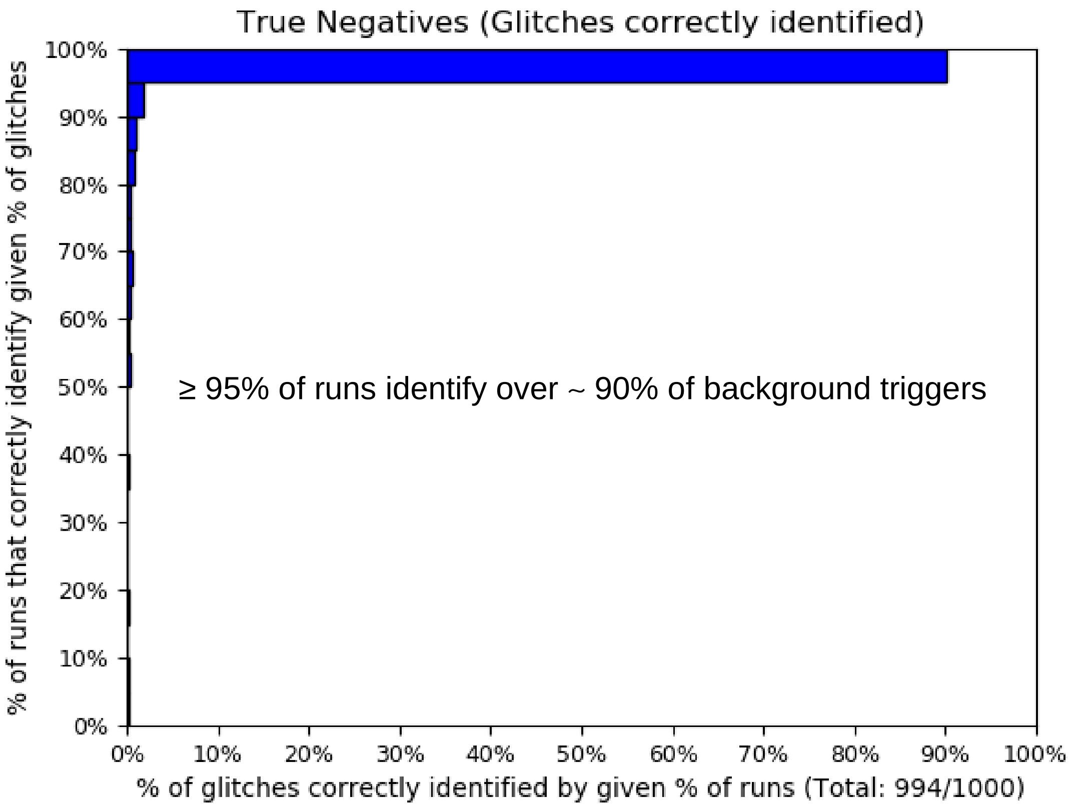
Dimmelmeyer waveforms at 5.62 kpc



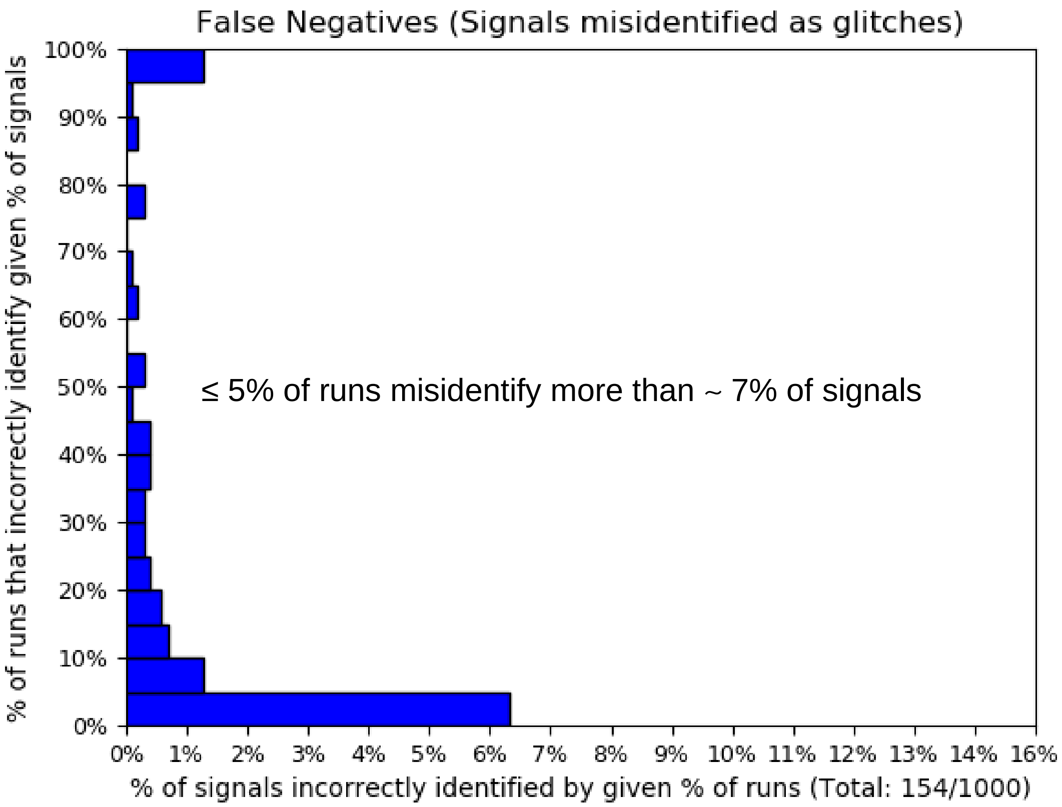


Code statistic performance

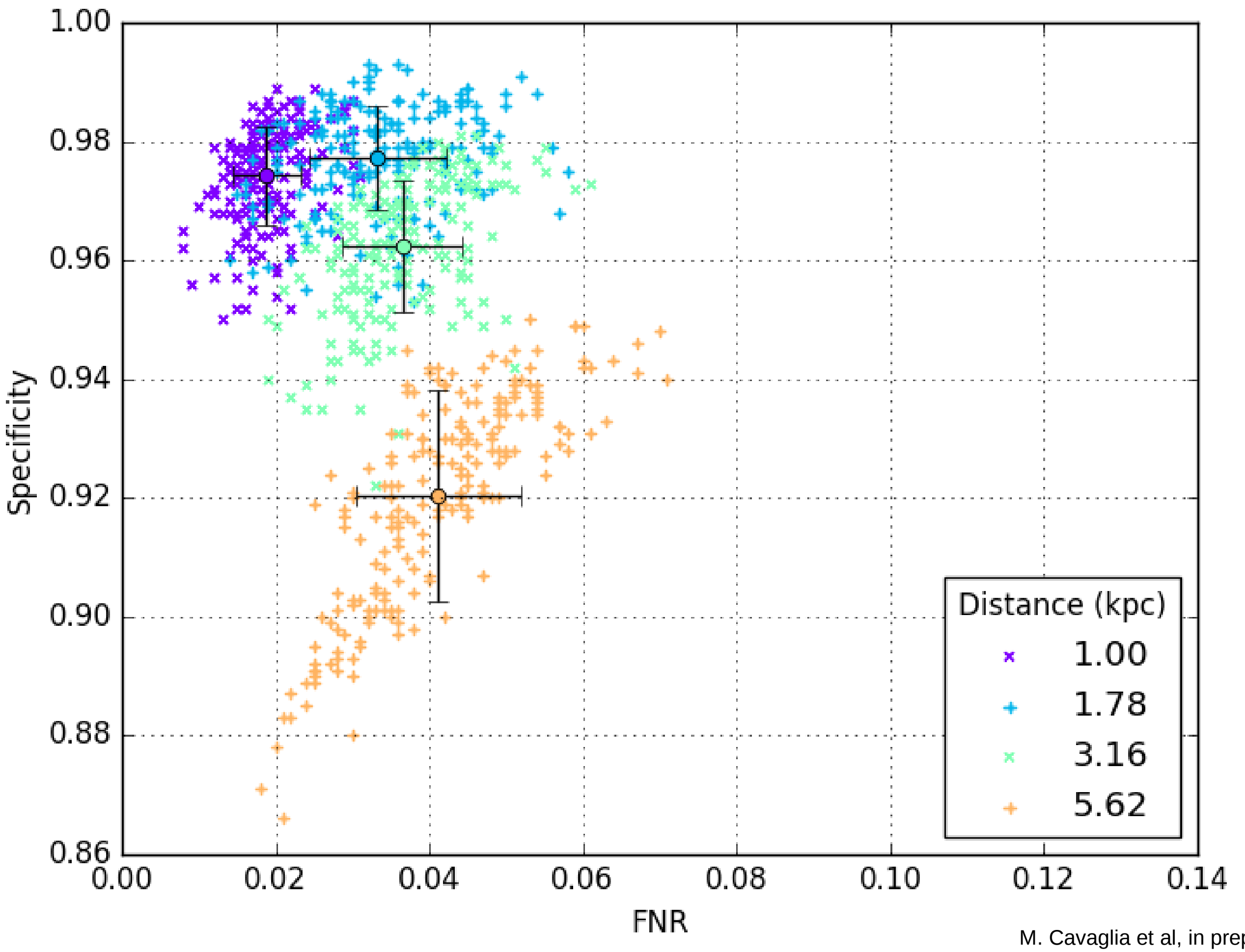
M. Cavaglia et al, in preparation.



M. Cavaglia et al, in preparation.

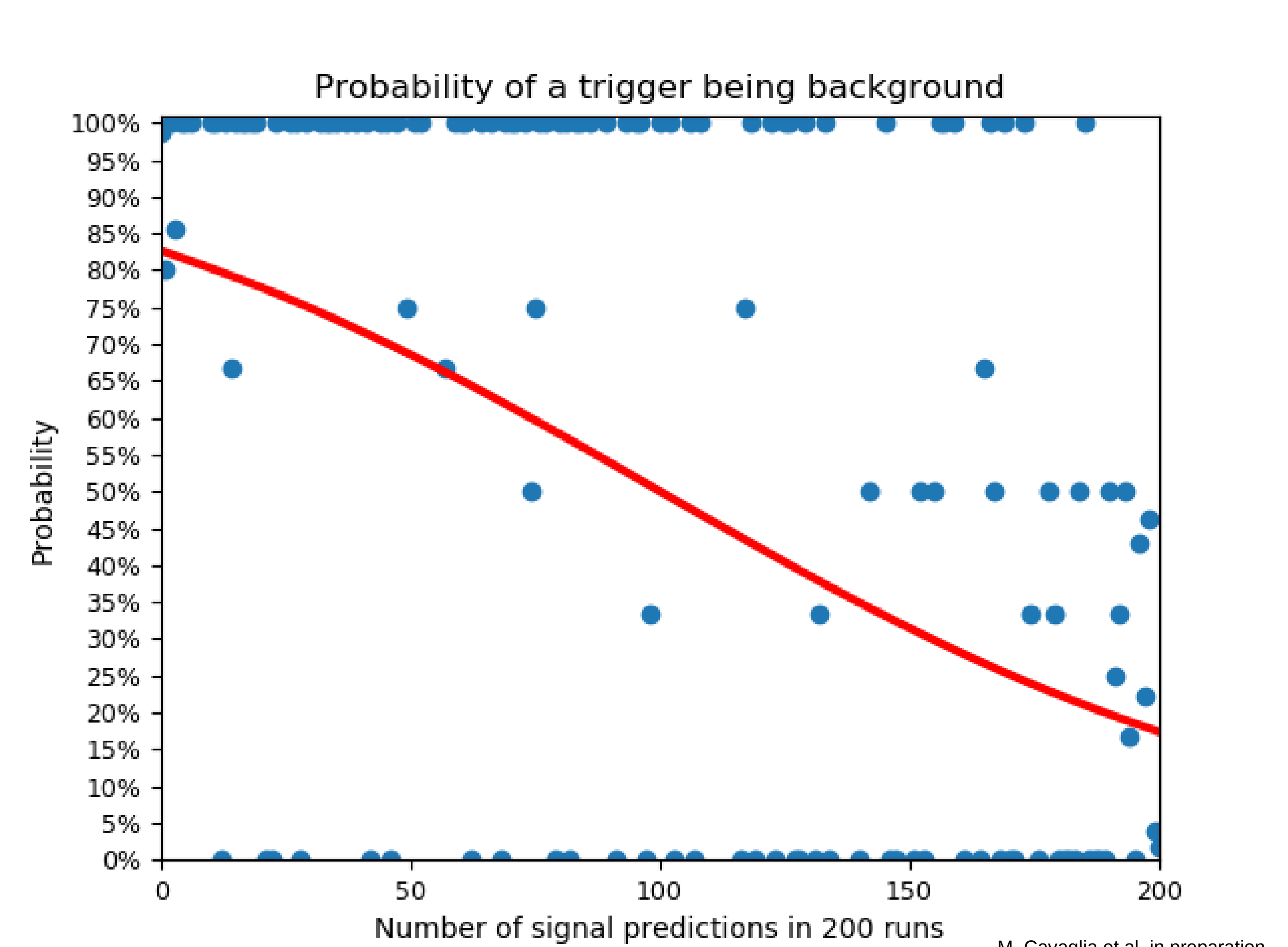


Comparison at different distances



M. Cavaglia et al, in preparation.

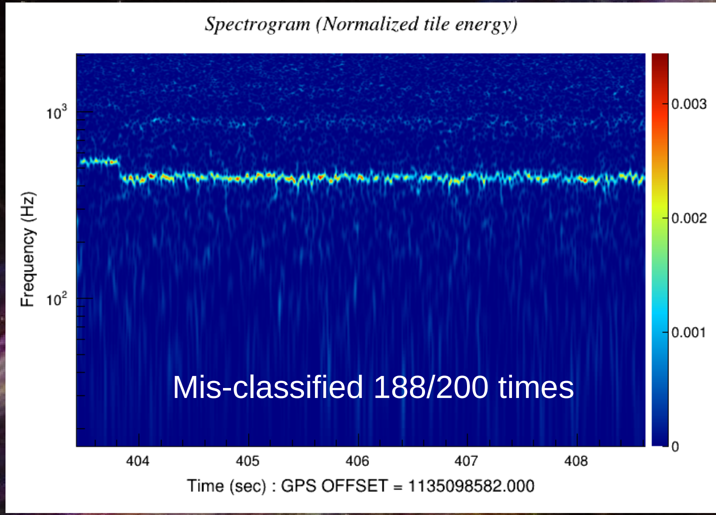
Probability of a trigger to be background



M. Cavaglia et al, in preparation.

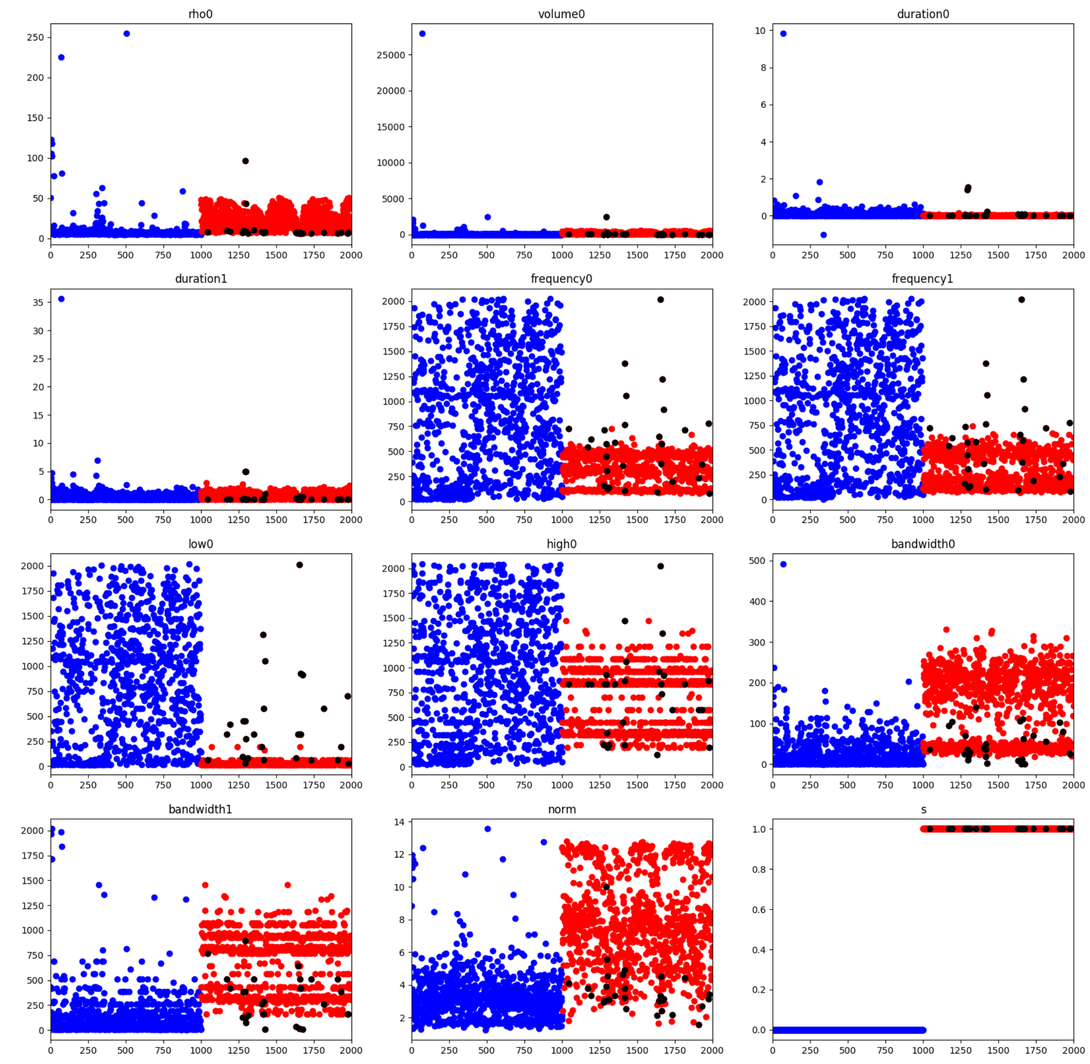
Follow-up

- <u>Pre-selection cuts</u> may further improve efficiency.
- Contaminations in training set.



M. Cavaglia et al, in preparation.

False negatives ≥ 80/200 runs



M. Cavaglia et al, in preparation.

Blind analysis

Random blind set	Actual	Analysis all models, 1kpc			Analysis all models, 1.78kpc			Analysis all models, 3.16kpc			Analysis all models, 4.22kpc		
		SIG	BKG	Prediction	SIG	BKG	Prediction	SIG	BKG	Prediction	SIG	BKG	Prediction
1137221362.849899_1	BKG	0	200	BKG	0	200	BKG	0	200	BKG	0	200	BKG
1137221296.450439_2	BKG	0	200	BKG	9	191	BKG	12	188	BKG	32	168	BKG
1137221270.478584_3	BKG	0	200	BKG	5	195	BKG	7	193	BKG	43	157	BKG
1137221270.315765_4	BKG	0	200	BKG	0	200	BKG	0	200	BKG	0	200	BKG
1137221256.461151_5	BKG	0	200	BKG	0	200	BKG	0	200	BKG	0	200	BKG
1137221254.992889_6	BKG	0	200	BKG	0	200	BKG	0	200	BKG	0	200	BKG
1137221206.790939_7	BKG	0	200	BKG	0	200	BKG	0	200	BKG	2	198	BKG
1137221187.891924_8	BKG	0	200	BKG	0	200	BKG	0	200	BKG	0	200	BKG
1137088411.819580_9	BKG	0	200	BKG	0	200	BKG	0	200	BKG	0	200	BKG
1137088400.326843_10	BKG	39	161	BKG	66	134	BKG	91	109	BKG	85	115	BKG
1137248070.587524_11	SIG: Yak 10kpc	1	199	BKG	2	198	BKG	26	174	BKG	73	127	BKG
1137123606.447540_12	SIG: Yak 3.16kpc	3	197	BKG	27	173	BKG	146	54	SIG	180	20	SIG
1137234559.739685_13	SIG: Yak 3.16kpc	61	139	BKG	155	45	SIG	188	12	SIG	192	8	SIG
1137250081.748009_14	SIG: Yak 3.16kpc	1	199	BKG	36	164	BKG	167	33	SIG	183	17	SIG
1137215815.308205_15	SIG: Yak 1kpc	173	27	SIG	187	13	SIG	188	12	SIG	191	9	SIG
1137240747.519287_16	SIG: Yak 1kpc	173	27	SIG	188	12	SIG	188	12	SIG	190	10	SIG
1137251495.131439_17	SIG: Yak 1kpc	173	27	SIG	189	11	SIG	188	12	SIG	192	8	SIG
1137232392.167053_18	SIG: Yak 1kpc	171	29	SIG	188	12	SIG	188	12	SIG	191	9	SIG
1137237558.365189_19	SIG: Yak 1kpc	172	28	SIG	187	13	SIG	186	14	SIG	187	13	SIG

Analysis with “all-model training” multivariate functions at different distances

Red: Classified incorrectly
Orange: Classified correcly, but would need a follow-up
Black: Classified correctly

M. Cavaglia et al, in preparation.

Conclusions and future directions

- The method seems to be effective at reducing the background up to ~ 90%.
- Rapid, easy ML implementation to current cWB pipeline.
- If implemented, this method could help detection of galactic CCSN at 3σ c.l. even with current detectors.
- Not only single-IFO: The method can be used for different network configurations.
- Addition of non-astrophysical (instrumental/environmental) information can further improve results.
- To do:
  - Explore method at larger distances.
  - Signal recovery/distance estimation.
  - Can we distinguish between CCSN models?

Thank you!

The authors thankfully acknowledge the use of LIGO Scientific Collaboration and Virgo Collaboration resources that have made possible some of the results presented in this talk. This work has been partially supported by NSF grant PHY-1707668.

Background Image Credit: NASA/ESA/Hubble Heritage Team