Title: Enhancing transient gravitational wave analyses with machine learning

Speakers: Ik Siong Heng

Series: Strong Gravity

Date: May 20, 2021 - 1:00 PM

URL: http://pirsa.org/21050017

Abstract: Gravitational wave observations are beginning to reveal the nature of the dark side of our universe. The Advanced LIGO and Virgo detectors have observed dozens of binary black hole mergers during the recent third observing run and, with planned sensitivity improvements, expect to observe significantly more binary black hole mergers in future observing runs. The combination of the increased number of detections and the sheer volume of data associated with each detection provides a significant data analysis challenge. In recent years, various machine learning approaches such as convolutional neural networks have been explored as a basis for rapid analyses for gravitational wave data. This seminar will give a brief introduction to current transient gravitational wave data analysis methodology and highlight novel applications of machine learning for rapid detection of binary black holes and rapid inference of their astrophysical properties. The use of generative machine learning algorithms for transient gravitational wave signal generation will also be discussed.

Enhancing transient gravitational wave analyses with machine learning

Ik Siong Heng
with plenty of input from Chris Messenger, Jordan McGinn and Hunter Gabbard

Perimeter Institute, 20th May, 2021

IGR

University of Glasgow

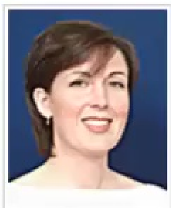
Overview

- Brief introduction
- Neural networks and autoencoders
- Rapid parameter estimation with conditional variational autoencoders (VItamin)
- Generative adversarial networks for Burst waveform generation
- Summary

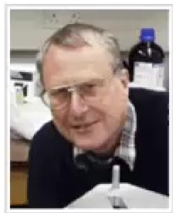
IGR University of Glasgow

Institute for Gravitational Research

Academics

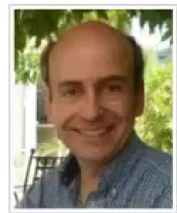


Prof. Sheila Rowan

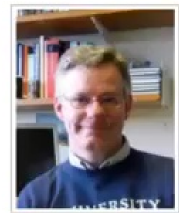


Prof. James Hough

Prof. Ken Strain



Prof. Graham Woan

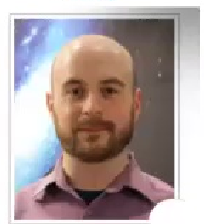


Prof. Martin Hendry

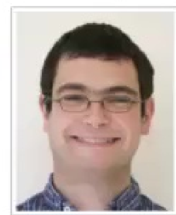
Prof. Giles Hammond

Prof. Ik Siong Heng

Prof. Harry Ward

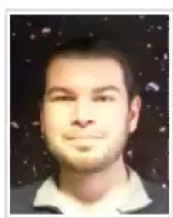


Dr Christopher Berry

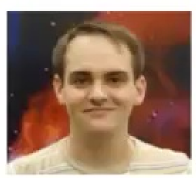


Dr Iain Martin

Dr Chris Messenger



Dr John Veitch



Dr Eric Oelker

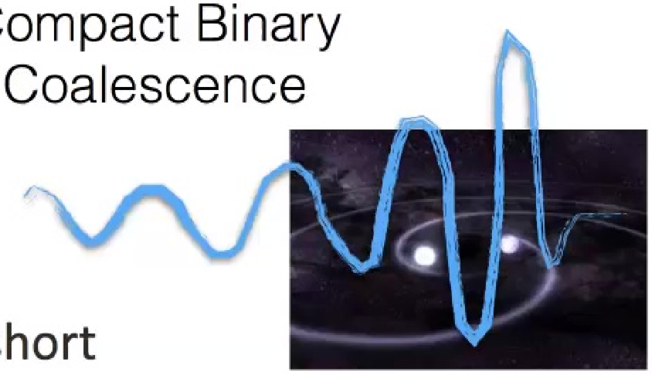
The Institute for Gravitational Research (IGR) consist of over 80 members: 13 academic staff, ~30 postdocs/research fellows, ~40 PhD students plus support staff

IGR University of Glasgow

Gravitational wave signal types

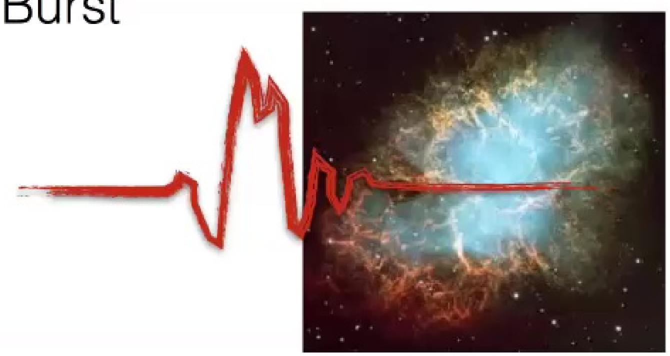
modelled | unmodelled

Compact Binary Coalescence

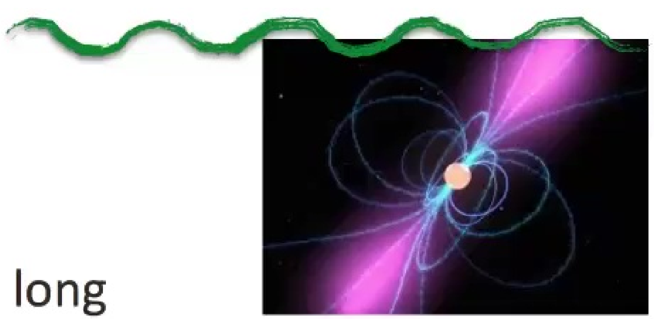


short

Burst

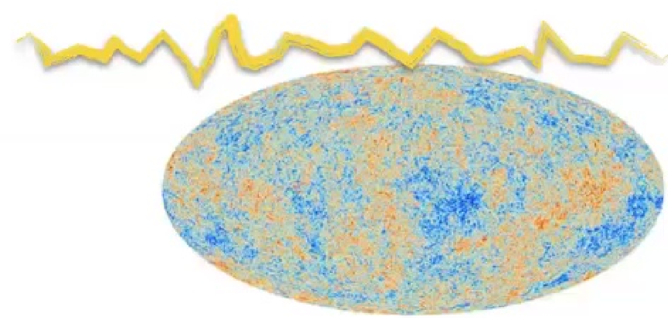


Continuous



long

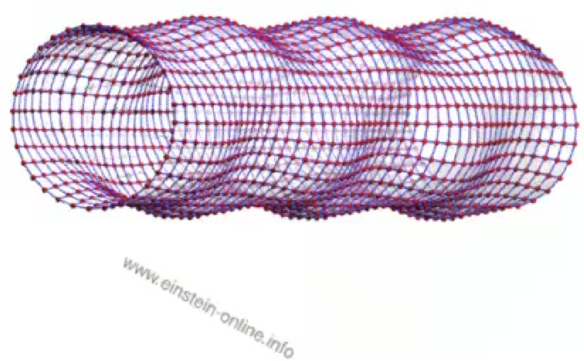
Stochastic

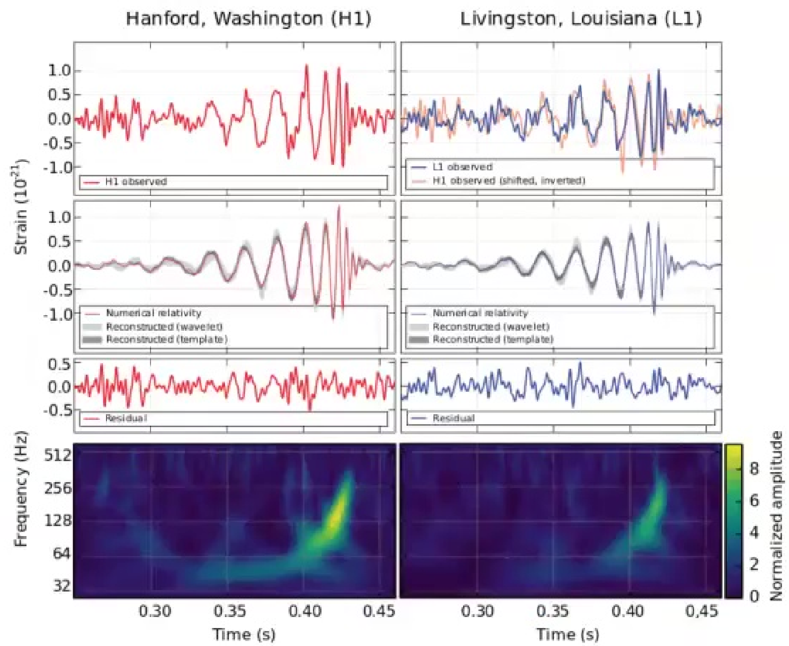


IGR University of Glasgow

Very brief gravitational wave intro

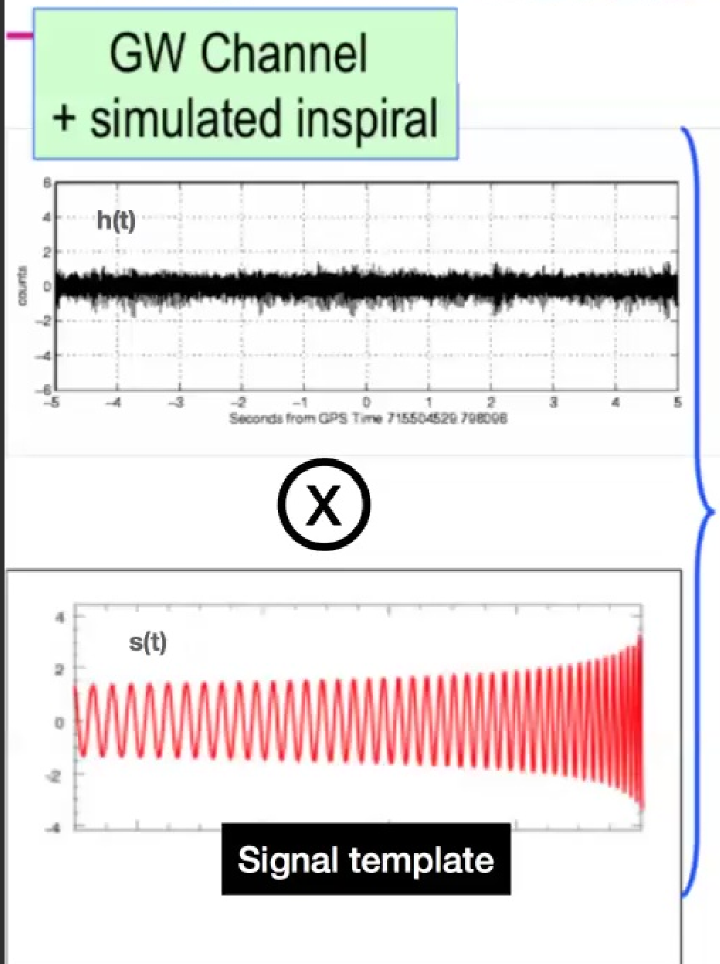
- Gravitational waves are ripples in space-time that travel at the speed of light
- They are generated by time varying mass distributions
- They have 2 polarisation states and affect the relative positions of test particles
- We will focus on signals generated from compact binary coalescences





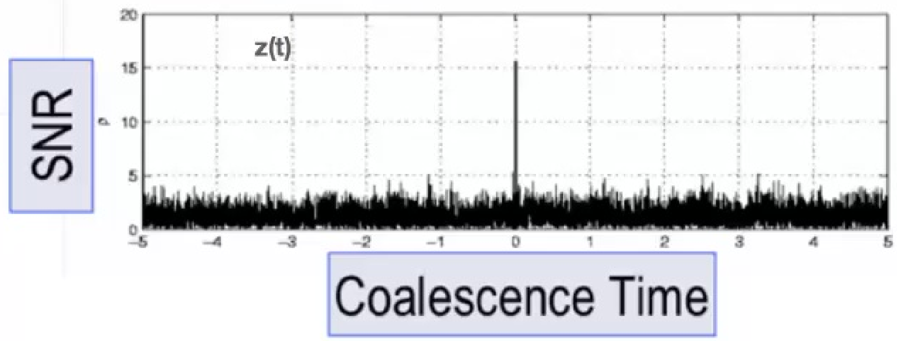
LIGO-Virgo Collaboration, PRL, 116, 6 (2016)

Detecting compact binary coalesences



For matched filtering, we require a lot of signal templates (template bank) to cover the search parameter space for possible signals and minimise loss of SNR.

The search can be performed relatively quickly using a computing cluster.



$$z(t) = 4 \int_0^{\infty} \frac{\tilde{h}(f)^* \tilde{s}(f)}{S_n(f)} e^{2\pi i f t}\, df,$$

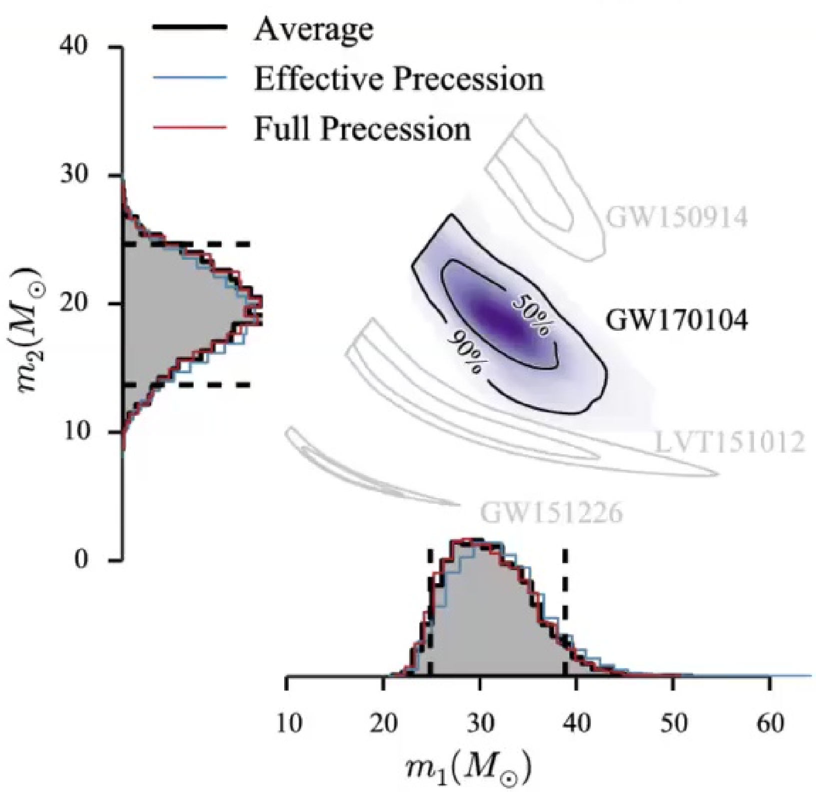
Bayesian inference & parameter estimation

- Astrophysical properties of detected coalescenes are inferred with a Bayesian framework

$$\underbrace{p(X|Y,I)}_{\text{posterior}} = \frac{\overbrace{p(Y|X,I)}^{\text{likelihood}} \times \overbrace{p(X|I)}^{\text{prior}}}{\underbrace{p(Y|I)}_{\text{evidence}}}$$

- The likelihood calculation is costly for a high-dimensional system so (quasi-)stochastic algorithms are used to efficiently evaluate the likelihood
  - Markov Chain Monte Carlo (MCMC)
  - nested sampling

LIGO-Virgo Collaboration, PRL, 118, 22 (2017)

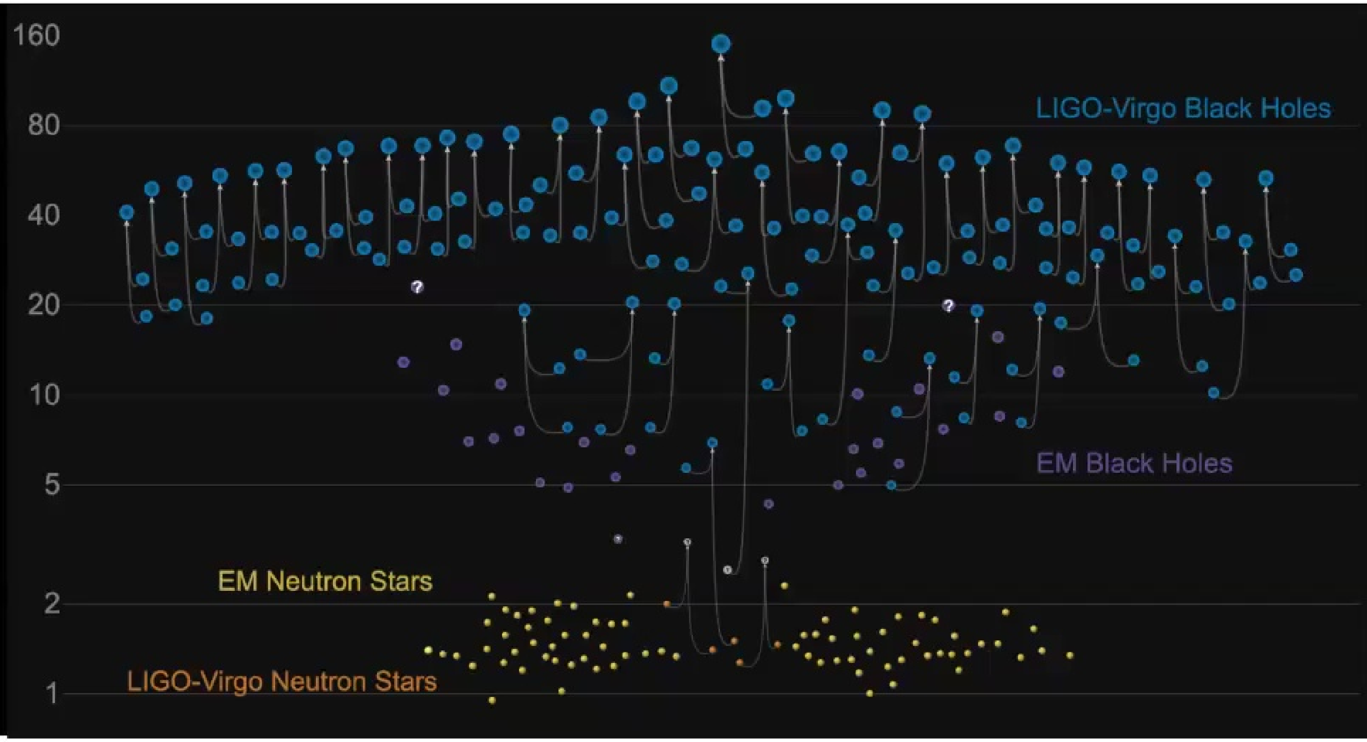


$$p(parameter|data, I) \propto p(data|parameter, I) \times p(parameter|I)$$

IGR University of Glasgow

Example detections

GWTC-2 plot v1.0 LIGO-Virgo, Frank Elavsky, Aaron Geller, Northwestern

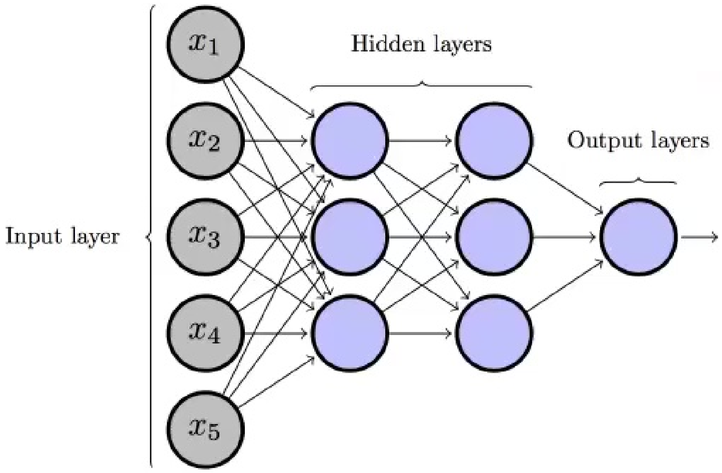


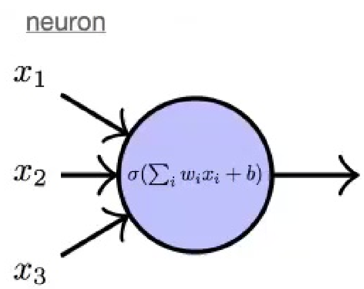
Parameter estimation

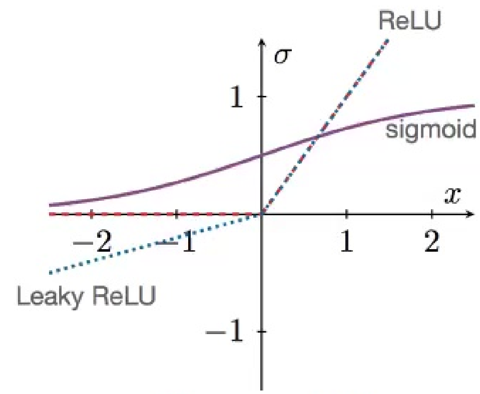
- Gravitational wave parameter estimation is slow
- Typical analyses (for O3) have taken between 6 hours and 5 days
- This is for full Bayesian PE and not to be compared with the rapid sky only tools [Singer & Price PRD, 93, 2 (2016)]
- There are other overheads in getting analyses running
- Rapid parameter estimation is important for multi-messenger astrophysics and also for computational efficiency
- Use machine learning to speed up parameter estimation

Neural network basics

- In a neural network, each neuron consists of an activation function where the input to the function is the combined outputs of neurons from the previous layer.
  - eg activation functions: sigmoid, rectified linear unit (ReLU) and leaky ReLU
- A typical neural network data set is split into 3 data sets: training, validation and testing
  - training: weights & biases; validation: hyperparameters (eg. learning rate); testing: performance evaluation (eg. % of testing data labelled correctly)
- A loss function measures how close predicted values are to the truth

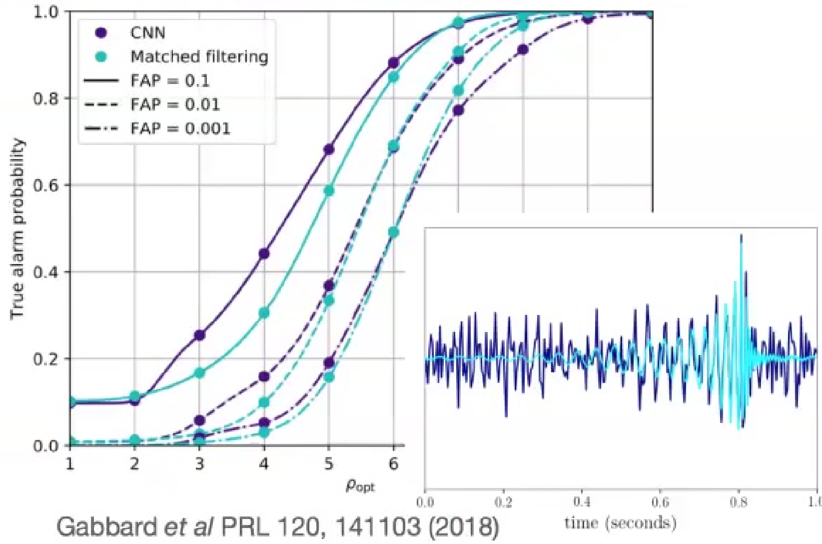




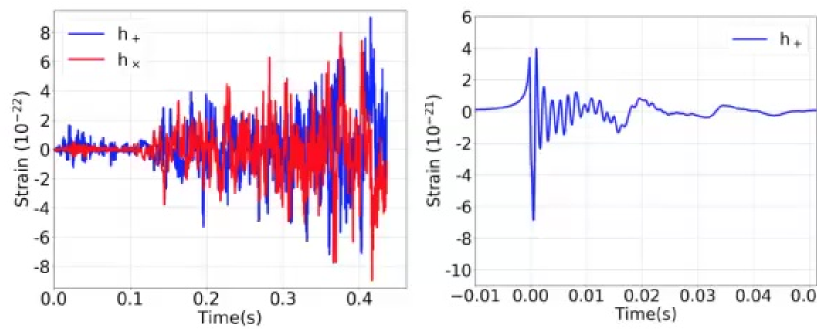


Neural networks for detection

- Convolutional neural networks (CNN) have been shown to match matched filtering for binary black hole signals
  - many papers on this topic including Gabbard *et al.* (2018)
- Convolutional aspect is mainly for data reduction, by combining multiple data points together
  - eg. reduce 16x16 image to 4x4
- In these cases, the CNNs are deployed as classifiers; predicting whether the data contains noise only or noise+signal
- CNNs have also been show to work on supernova waveforms
  - eg. Chan *et al.* (2020)



Gabbard *et al* PRL 120, 141103 (2018)

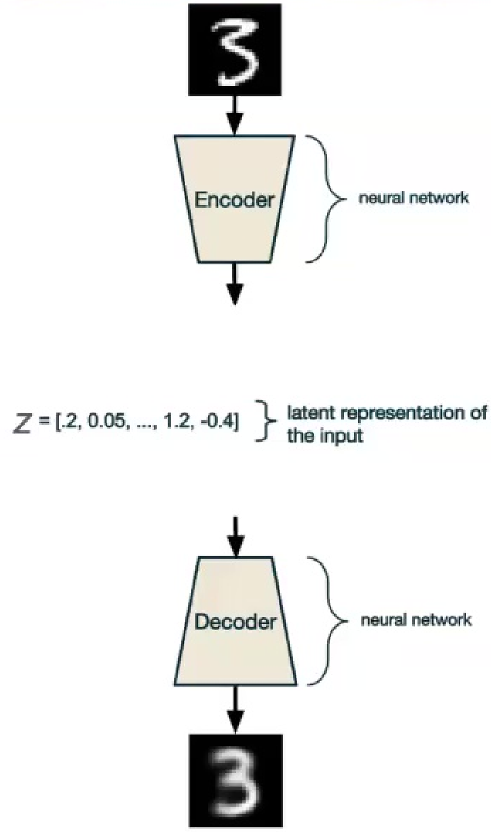


Chan *et al*. Phys. Rev. D 102, 043022 (2020)

IGR University of Glasgow

Basic autoencoder

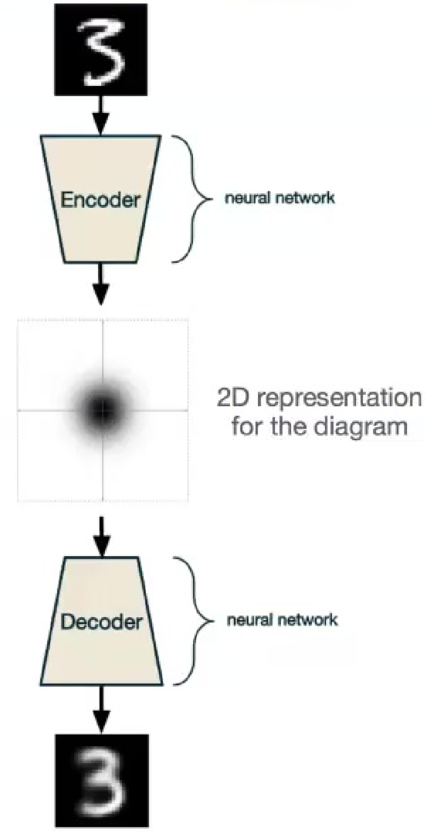
- Autoencoders combine two neuron networks in sequence
- Encoder maps the input into a (reduced) abstract "latent" representation
- The Decoder network converts the latent representation into an output
- The loss function is minimised when the output best matches the input



https://ijdykeman.github.io/ml/2016/12/21/cvae.html

Variational autoencoder

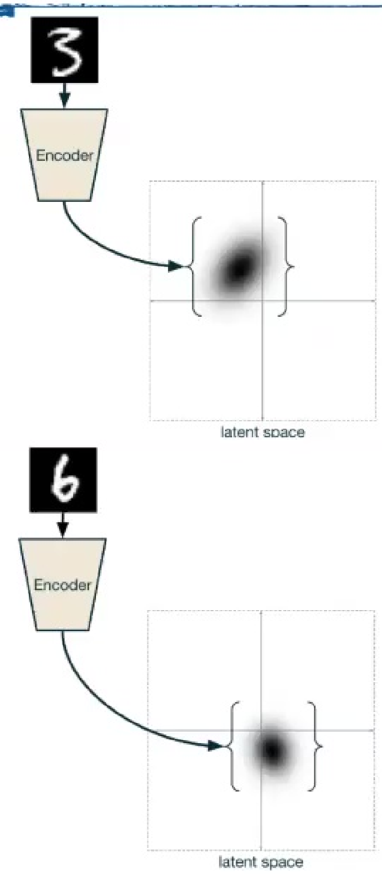
- Encoder predicts the mean and covariance of a multi-dimensional Gaussian in the latent space
- We then randomly sample from that distribution
- The Decoder network converts the (random) latent representation into an output
- The loss function is minimised when the output best matches the input, and…
- there's an extra loss component that keeps the latent space Gaussian averaged over all inputs



https://ijdykeman.github.io/ml/2016/12/21/cvae.html

Variational autoencoder

- So an image of a “3” gets mapped to a particular region of the latent space.
- The inherent spread in the distribution represents the acceptable variation in that ”3”.
- A “6” lives elsewhere in the latent space, probably close to the “8”s, and “5”s since they share similar characteristics.
- Other dimensions of the space may encode things like stroke width, and angle, etc…

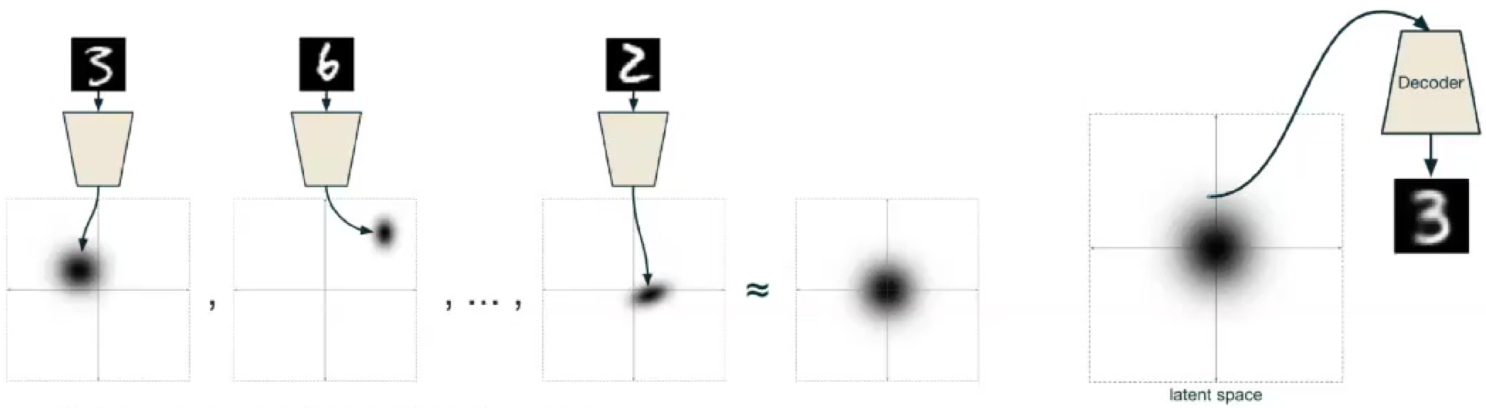


https://ijdykeman.github.io/ml/2016/12/21/cvae.html

IGR University of Glasgow

Variational autoencoder

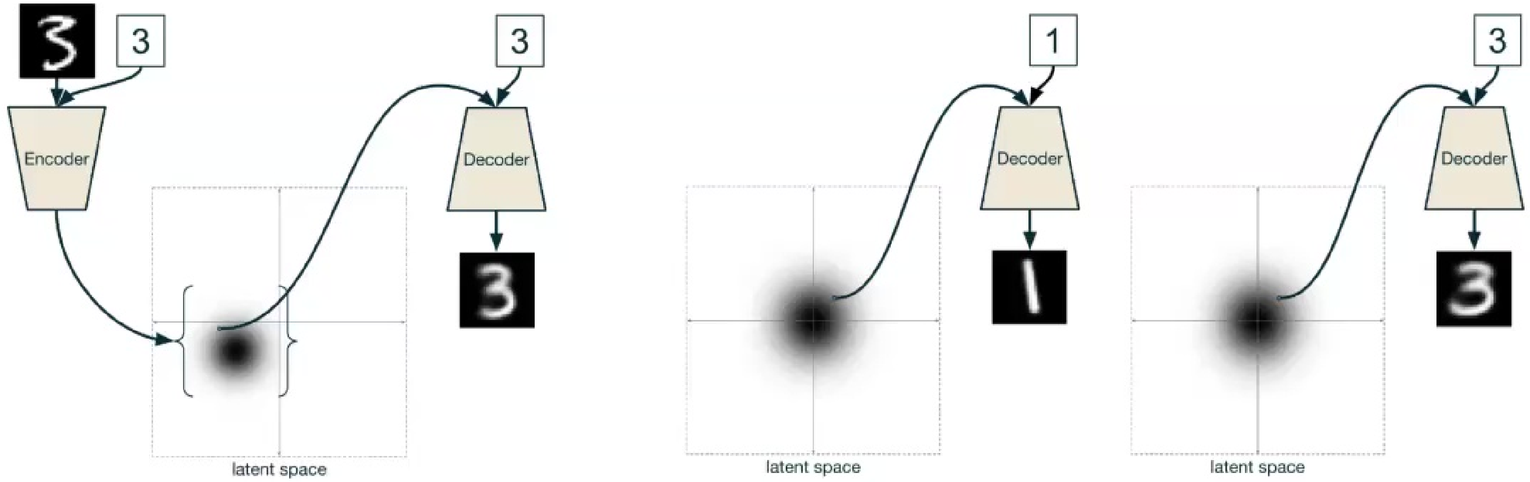
- The Kullback-Leibler divergence (KL) loss keeps the ensemble of training data mapped to a zero-mean, unit-variance Gaussian.
- So you can then sample from it after training to generate new images



https://ijdykeman.github.io/ml/2016/12/21/cvae.html

Conditional Variational Autoencoder (CVAE)

- Passing labels allows you specify properties of the output



https://ijdykeman.github.io/ml/2016/12/21/cvae.html

Conditional Variational Autoencoder (CVAE)

- You should think of the encoder network in terms of probability distributions.
- For this basic CVAE the encoder is modelling the distribution

$$p(z|x, y)$$

- The decoder is modelling the function

$$f(y, z)$$

x is the image
y is the label
z is the latent space location

- and the loss function is (something like)

$$L = \langle (f(z, y) - x)^2 \rangle_{p(x,y,z)} + \beta \, \mathrm{KL}\left(p(z|x, y) | G(0, 1)\right)$$

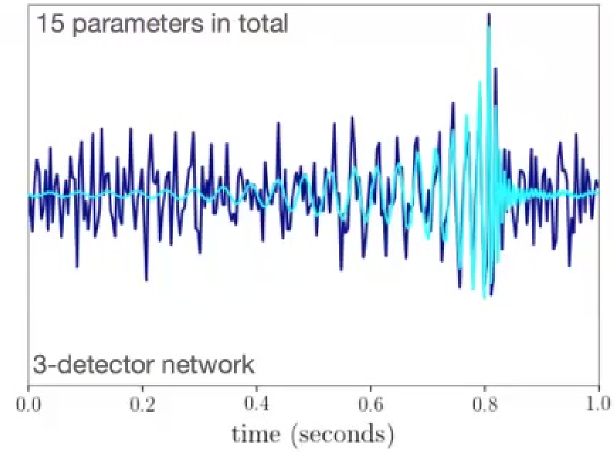
https://ijdykeman.github.io/ml/2016/12/21/cvae.html

IGR University of Glasgow

VItamin

- We develop a rapid inference method (VItamin) based on conditional variational autoencoders
- VItamin is trained on whitened BBH time series in Gaussian noise and the true parameter values with the aim of producing samples from the posterior distributions of the parameters
- Note that **posteriors are NOT used in training**

Parameter name	symbol	min	max	units
mass 1	m_1	35	80	solar masses
mass 2	m_2[a]	35	80	solar masses
luminosity distance	d_L	1	3	Gpc
time of coalescence	t_0	0.65	0.85	seconds
phase at coalescence	ϕ_0	0	2π	radians
right ascension	α	0	2π	radians
declination	δ	$-\pi/2$	$\pi/2$	radians
inclination	ι	0	π	radians
polarisation	ψ	0	π	radians
spins	-	0		-



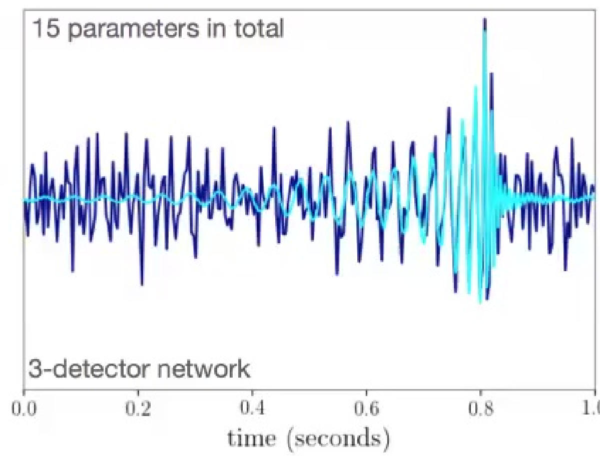
Gabbard et al, arXiv 1909.06296 (2019)

IGR University of Glasgow

VItamin

- We develop a rapid inference method (VItamin) based on conditional variational autoencoders
- VItamin is trained on whitened BBH time series in Gaussian noise and the true parameter values with the aim of producing samples from the posterior distributions of the parameters
- Note that **posteriors are NOT used in training**

Parameter name	symbol	min	max	units
mass 1	m_1	35	80	solar masses
mass 2	m_2[a]	35	80	solar masses
luminosity distance	d_L	1	3	Gpc
time of coalescence	t_0	0.65	0.85	seconds
phase at coalescence	ϕ_0	0	2π	radians
right ascension	α	0	2π	radians
declination	δ	$-\pi/2$	$\pi/2$	radians
inclination	ι	0	π	radians
polarisation	ψ	0	π	radians
spins	-	0		-



Gabbard et al, arXiv 1909.06296 (2019)

IGR University of Glasgow

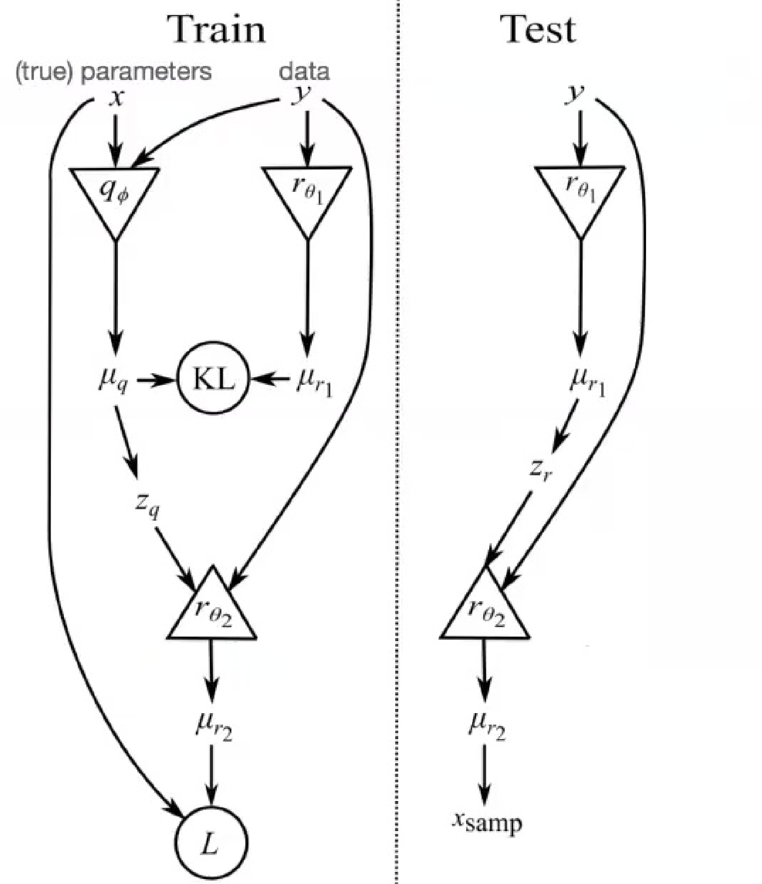
VItamin

- The data we measure is a noisy time series (y) consisting of a deterministic signal plus noise
- The signal is defined by the parameters (x)
- We want to obtain the posterior on the signal parameters

$$p(x|y)$$

- But, instead of calculating the posterior directly, we train an approximate function that can generate samples drawn from the posterior rapidly

$$r(x|y)$$



Gabbard et al, arXiv 1909.06296 (2019)

VItamin loss function

- We train the network with the aim of minimising the cross entropy between the true posterior and the approximation

$$H(y) = -\int dx\, p(x|y) \log r(x|y).$$

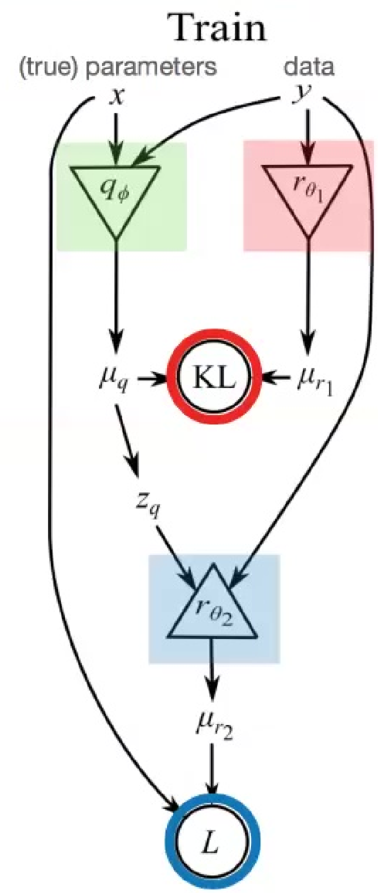
- It can be shown that

$$r(x|y) \geq \int q(z|x,y) \log\left(\frac{r(x|z,y)r(z|y)}{q(z|x,y)}\right) dz$$

- Combining these two, and changing the integral to a sum (since we have discrete data samples) we find

KL L

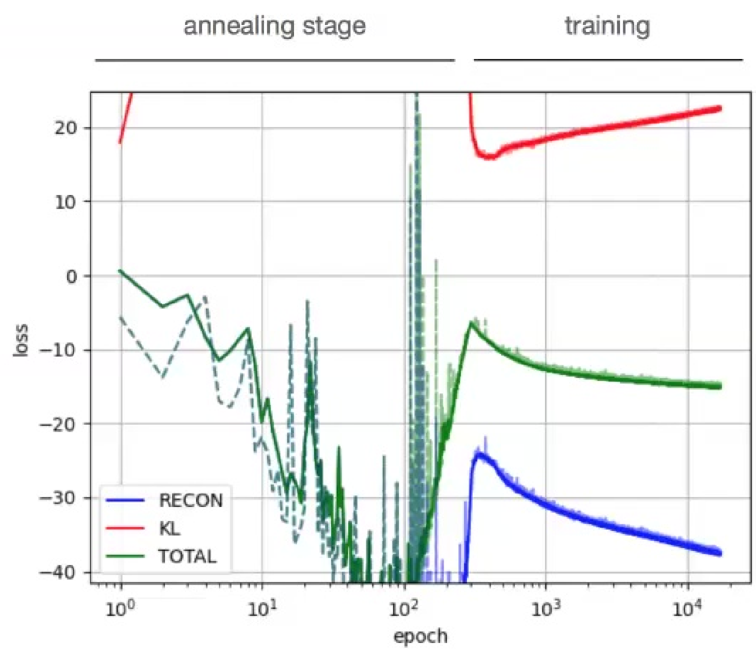
$$H \leq -\frac{1}{N}\sum_j \left(\log\left(\frac{r(z_j|y_j)}{q(z_j|x_j,y_j)}\right) + \log r(x_j|y_j,z_j)\right)\Bigg|_{\substack{x\sim p(x)\\ y\sim p(y|x)\\ z\sim q(z|x,y)}}$$



Gabbard et al, arXiv 1909.06296 (2019)

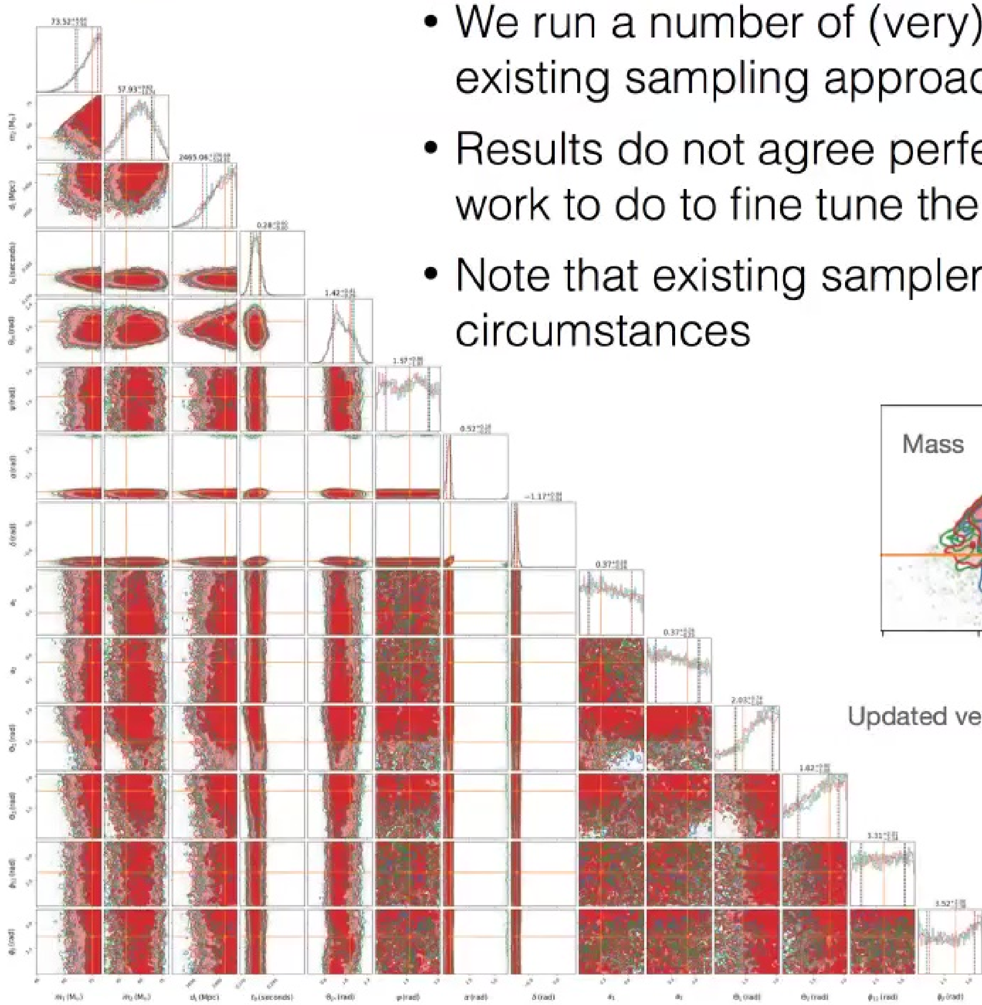
Vitamin results

- Needs lots of training data in the form of examples of noisy signals (y) plus the true signal parameters (x)
- Need a GPU and still takes ~days
- We do not need any costly pre-computed posteriors for training
- The total cost/loss is minimised at the expense of increased KL

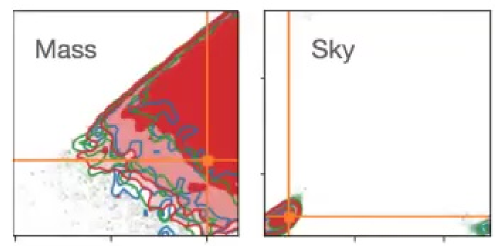


Updated version of Gabbard et al, arXiv 1909.06296 (2019)

Vitamin posteriors



- We run a number of (very) costly analyses with existing sampling approaches for comparison
- Results do not agree perfectly and there is still work to do to fine tune the networks
- Note that existing samplers also disagree in some circumstances



Updated version of Gabbard et al, arXiv 1909.06296 (2019)

Vitamin speed

- The primary difference is that the CVAE is pre-trained so that all cost is up-front
- We get a ~6 order of magnitude speed up in our test cases
- Can now generate 104 posterior samples in O(1) sec
- Training still takes O(days) but needs to only be done once*

Gabbard et al, arXiv 1909.06296 (2019)

TABLE I. Durations required to produce samples from each of the different posterior sampling approaches.

sampler	run time (seconds)			ratio $\frac{\tau_{\text{VItamin}}}{\tau_X}$
	min	max	median	
Dynesty[a] [15]	11795	29838	19400 [b]	5.2×10^{-6}
emcee [16]	18838	69272	32070	3.1×10^{-6}
ptemcee [17]	17124	37446	24372	4.1×10^{-6}
CPNest [14]	9943	53315	26202	3.8×10^{-6}
VItamin[c]	$\mathbf{1 \times 10^{-1}}$			1

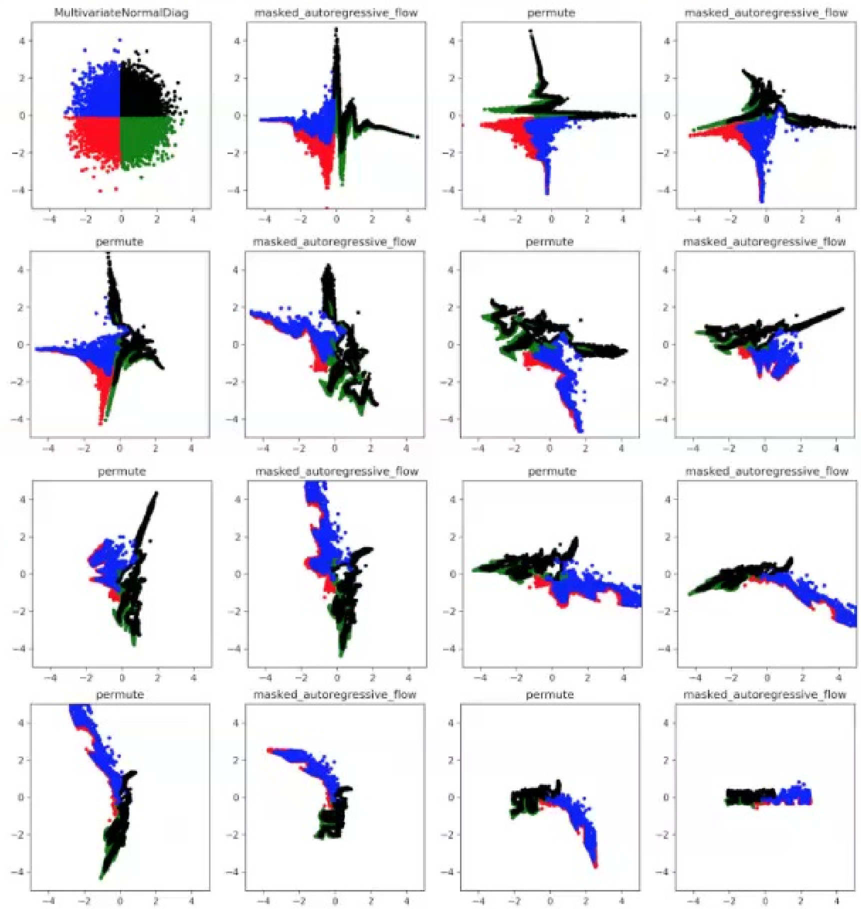
[a] The benchmark samplers all produced $\mathcal{O}(10000)$ samples dependent on the default sampling parameters used.
[b] We note that there are a growing number of specialised techniques [31–33] designed to speed up traditional sampling algorithms that could be used to reduce the runtimes quoted here by $\mathcal{O}(1-2)$ orders of magnitude.
[c] For the VItamin sampler 10000 samples are produced as representative of a typical posterior. The run time is independent of the signal content in the data and is therefore constant for all test cases.

Representative but now outdated

Normalising Flows

- Another Likelihood-free approach that can also obtain Bayesian posteriors is Normalising Flows [Green et al, PRD 102, 10 (2020)]
- These are generative models which produce tractable distributions where both sampling and density evaluation can be efficient and exact

$$p_X(x) = p_Z(f(x)) \left| \det \left(\frac{\partial f(x)}{\partial x^T} \right) \right|$$



https://blog.evjang.com/2018/01/nf2.html

Image generation

- Recent works have been incredibly successful in image generation.

- With conditioning, it is also possible to control the combination of features from each class.
  - eg. If GAN is trained on images of cats and pizza, it can create a pizza-cat.

https://thispersondoesnotexist.com

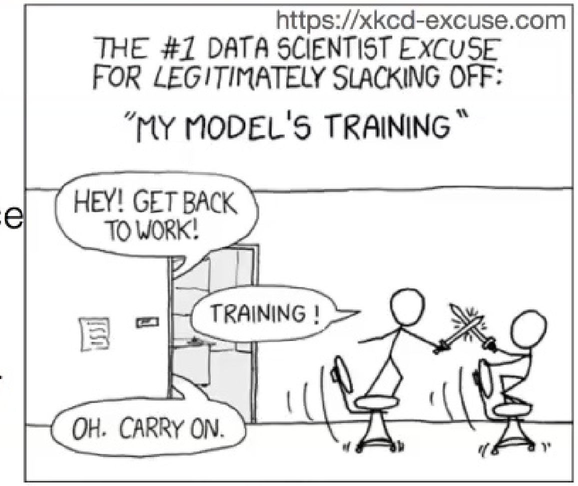
Characterising waveform generations

- A basic search pipeline using a CNN in order to compare the sensitivity of such a search using different GAN generated waveforms in Gaussian noise
  - 3 different CNN networks; one trained on Vertex generations, another on Simplex generations and the last on Uniform generations
- We are interested in the relative sensitivity as a function of the types of waveforms used for training the network.
- Set a threshold corresponding to a false alarm probability of 10^{-3}
- Reminder: Vertex generations correspond to the standard set of waveforms used in Burst searches

https://arxiv.org/abs/2103.01641

Summary

- Machine learning can provide a direct replacement for existing Bayesian parameter estimation
  - VItamin is applicable to general Bayesian inference problems
- This will enable realtime multi messenger astronomy and also the scope for pre-merger detections.
- There are still many challenges, e.g., real detector noise, longer duration signals, etc…
- Machine learning can also be used to generate complex waveform morphologies, beyond what is easily described analytically.
- Generalised Burst signals will hopefully improve search efficiency across the broad generic transient parameter space.
- Other applications: optimising searches, waveform reconstruction, enhancing gravitational wave detector stationarity and robustness…



IGR University of Glasgow