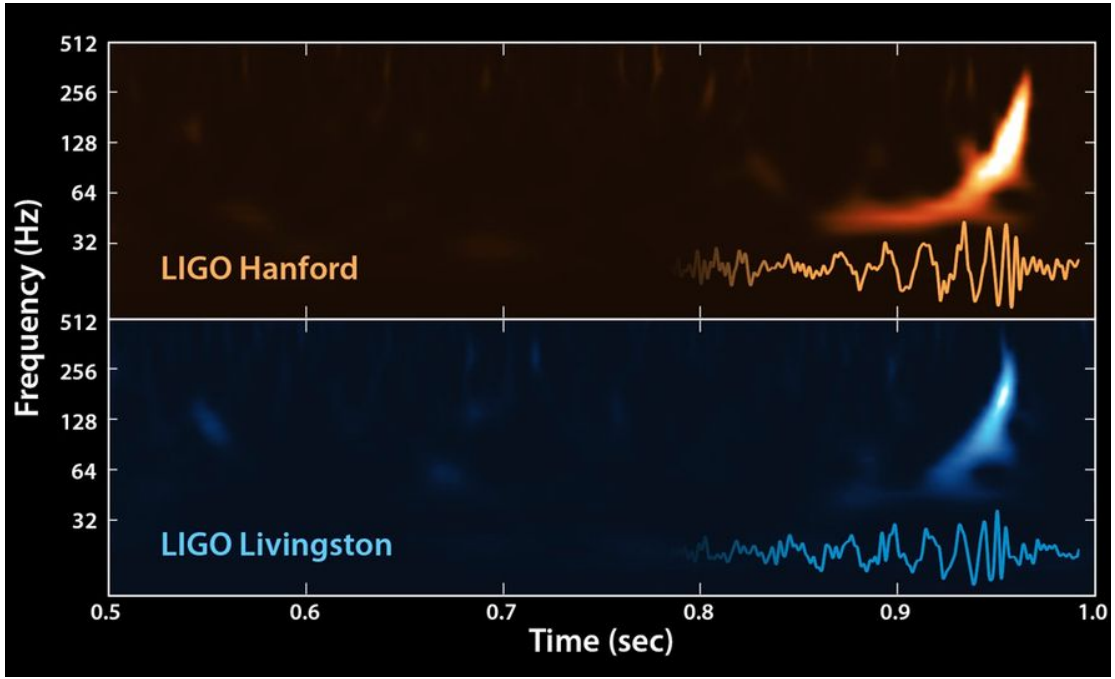
Detection of Gravitational Waves Using Neural Networks

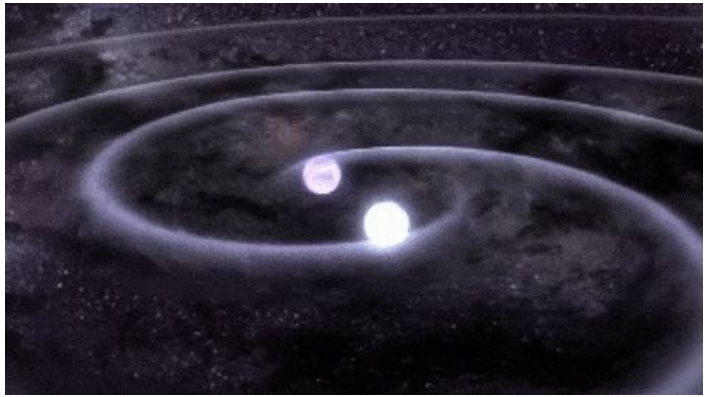
Shikha Bangar

THE UNIVERSITY OF TENNESSEE KNOXVILLE

Motivation:

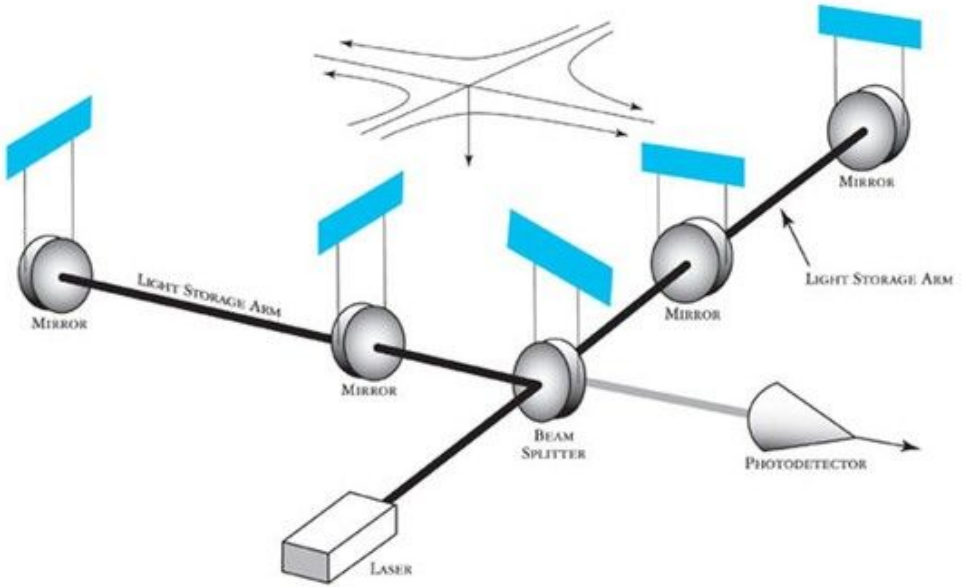
2015: Detection of Gravitational Waves

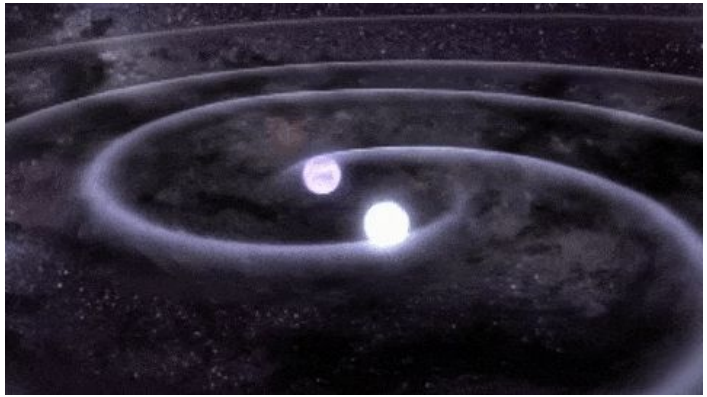




Motivation:

2015: Detection of Gravitational Waves





Goal:

Detection of Gravitational Waves using Neural Networks

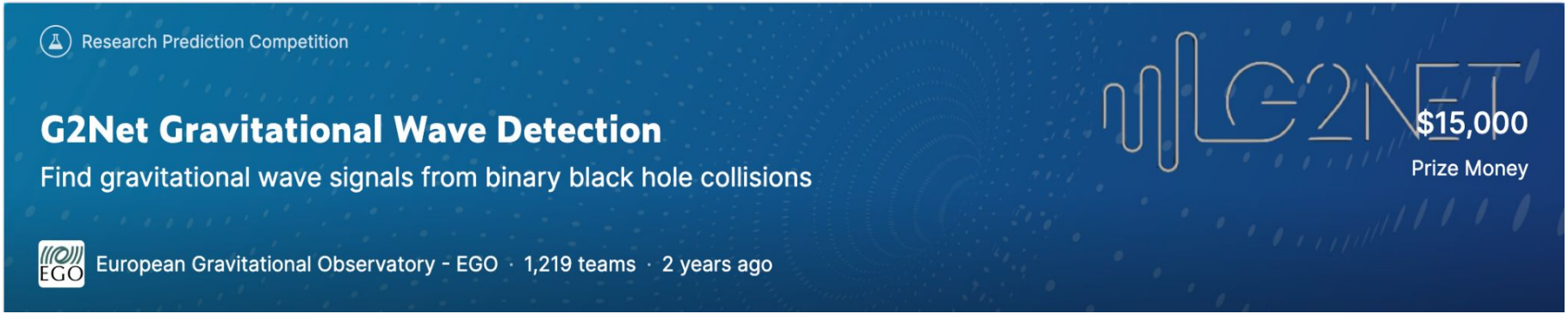
Data

Neural Network Model

Results and Discussion

Data:

kaggle



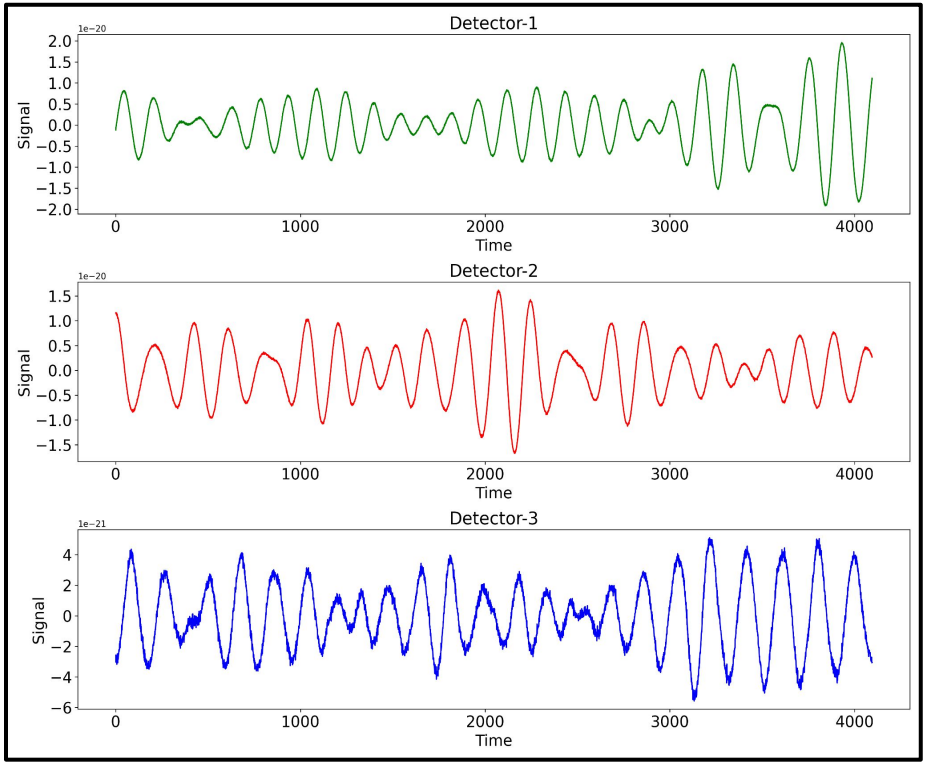
- Detect GW signals from the mergers of binary black holes
- GW time-series data from a network of Earth-based detectors - LIGO Hanford, LIGO Livingston, and Virgo

Data:

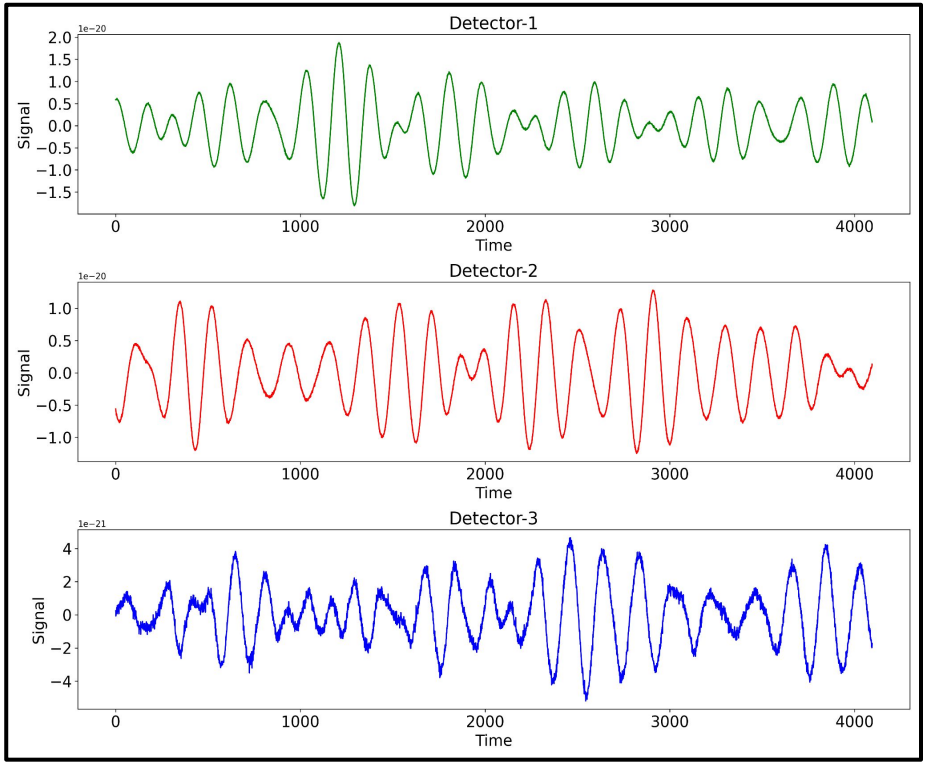
- Files - 786002 files, Size - 77.38 GB
- Each data sample (npy file) contains 3 time series (1 for each detector) and each spans 2 sec and is sampled at 2,048 Hz.

	id	target	path
0	00000e74ad	1	/content/g2net-gw/train/0/0/0/00000e74ad.npy
1	00001f4945	0	/content/g2net-gw/train/0/0/0/00001f4945.npy
2	0000661522	0	/content/g2net-gw/train/0/0/0/0000661522.npy
3	00007a006a	0	/content/g2net-gw/train/0/0/0/00007a006a.npy
4	0000a38978	1	/content/g2net-gw/train/0/0/0/0000a38978.npy

target = 0

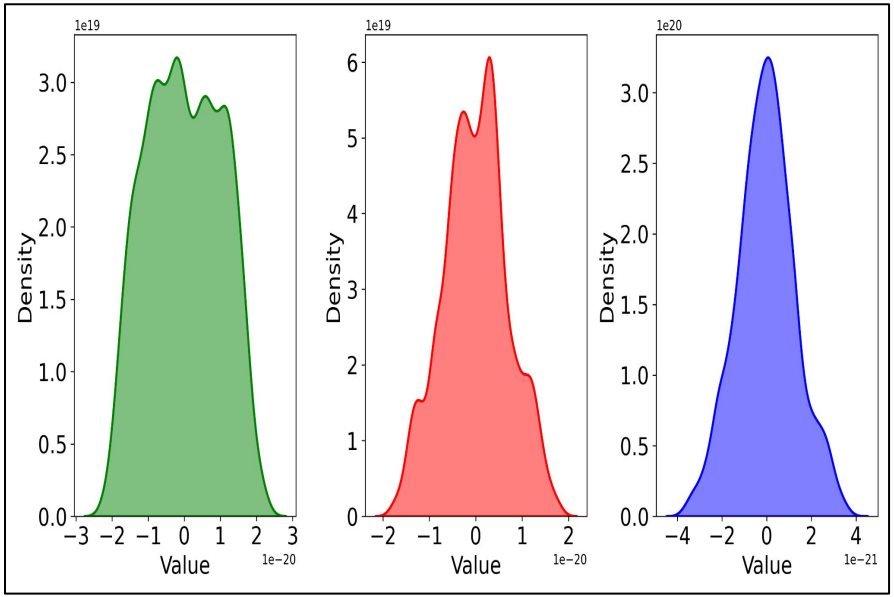


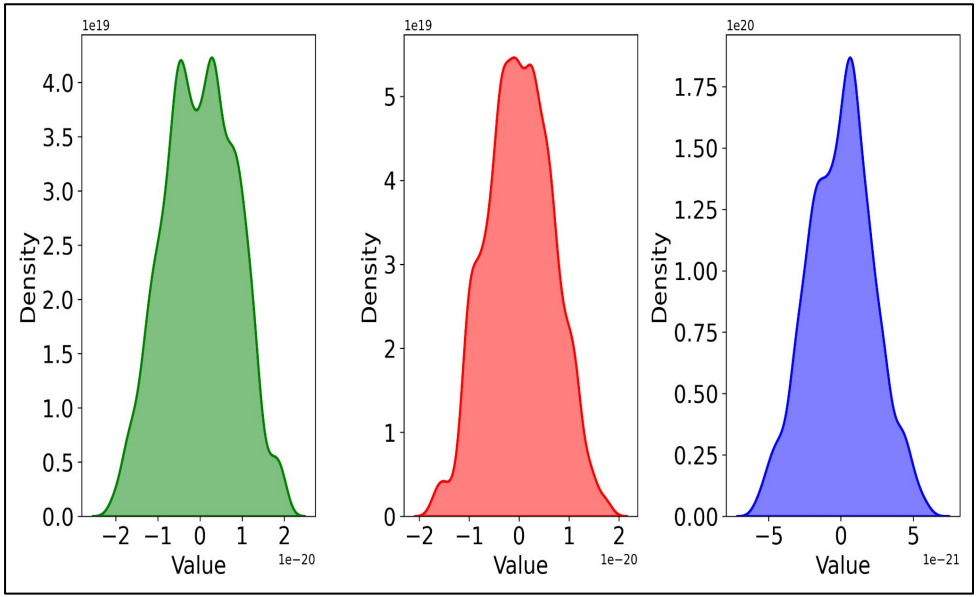
target = 1



target = 0

target = 1

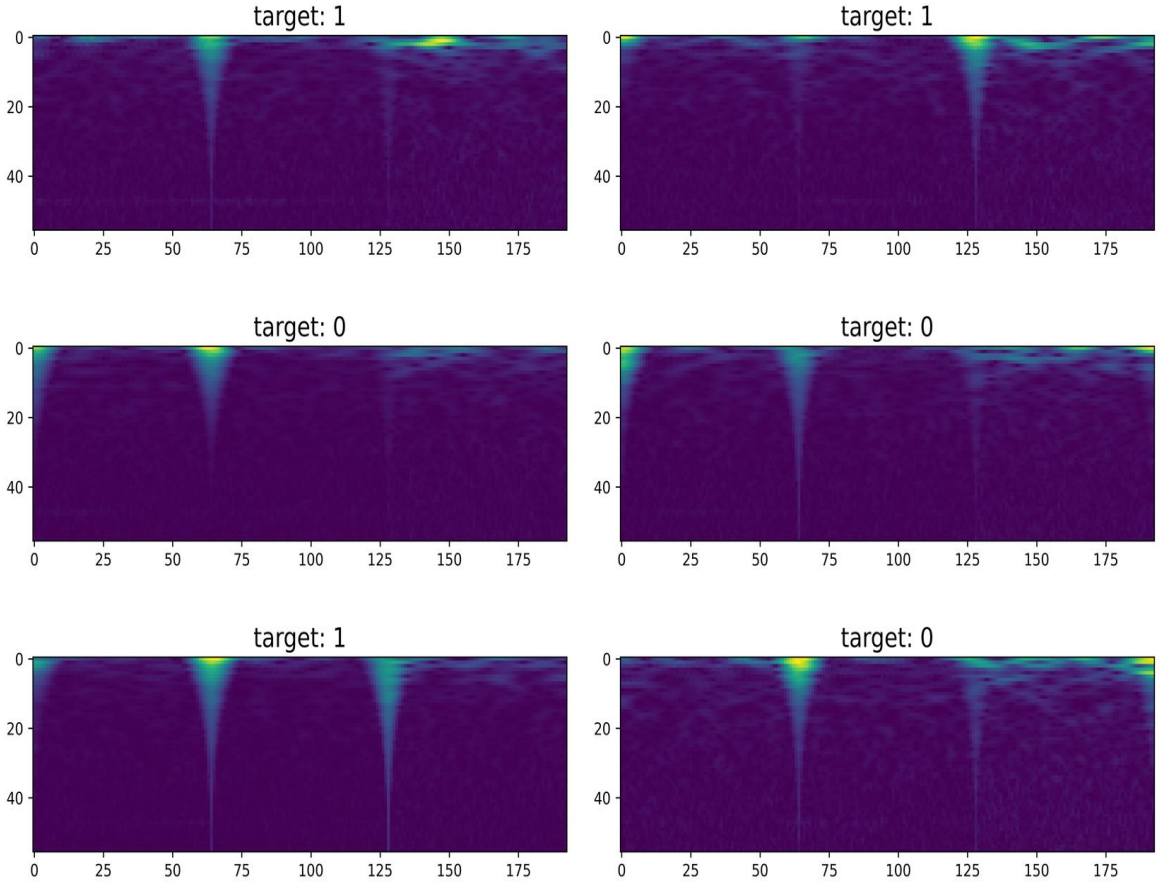




Using machine learning is important

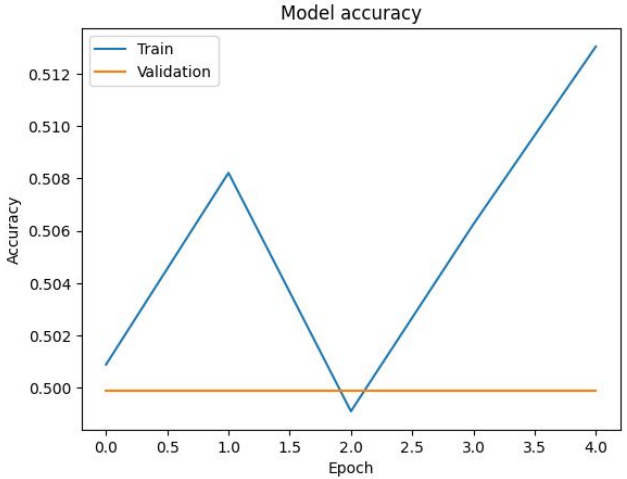
Data:

- Spectrogram
- CQT - constant quality factor transform



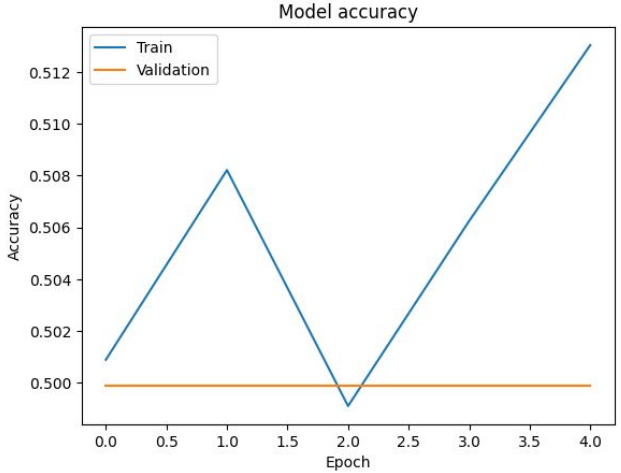
Selecting a model - optimization

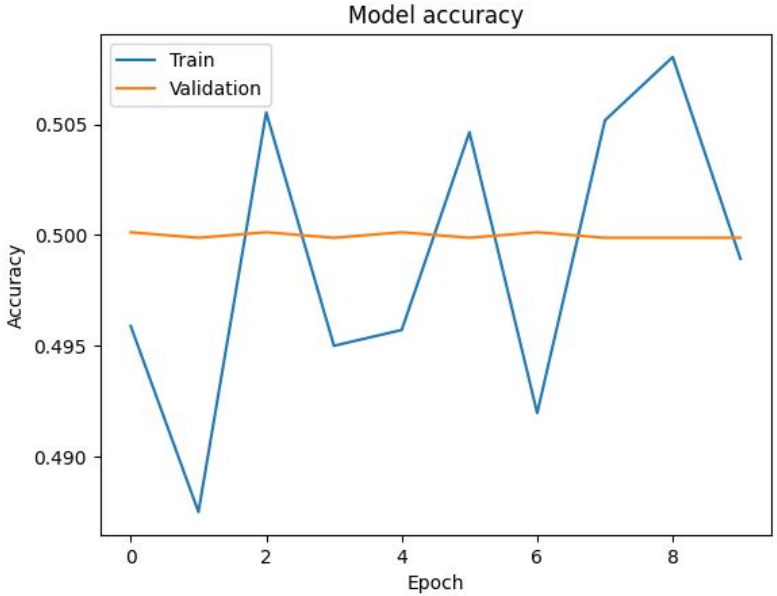
Simple Deep Neural Network (DNN)



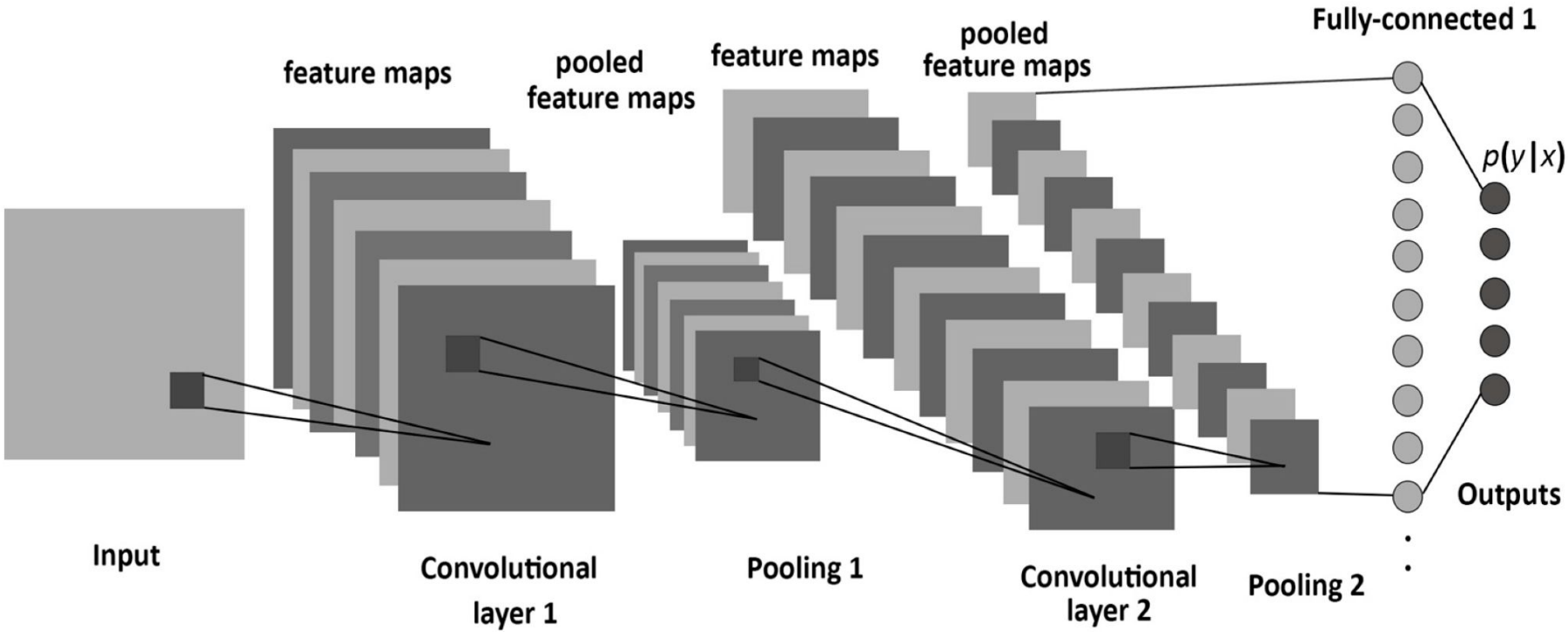
Selecting a model - optimization

Simple Deep Neural Network (DNN) → More complicated DNN





Convolution Neural Network (CNN):

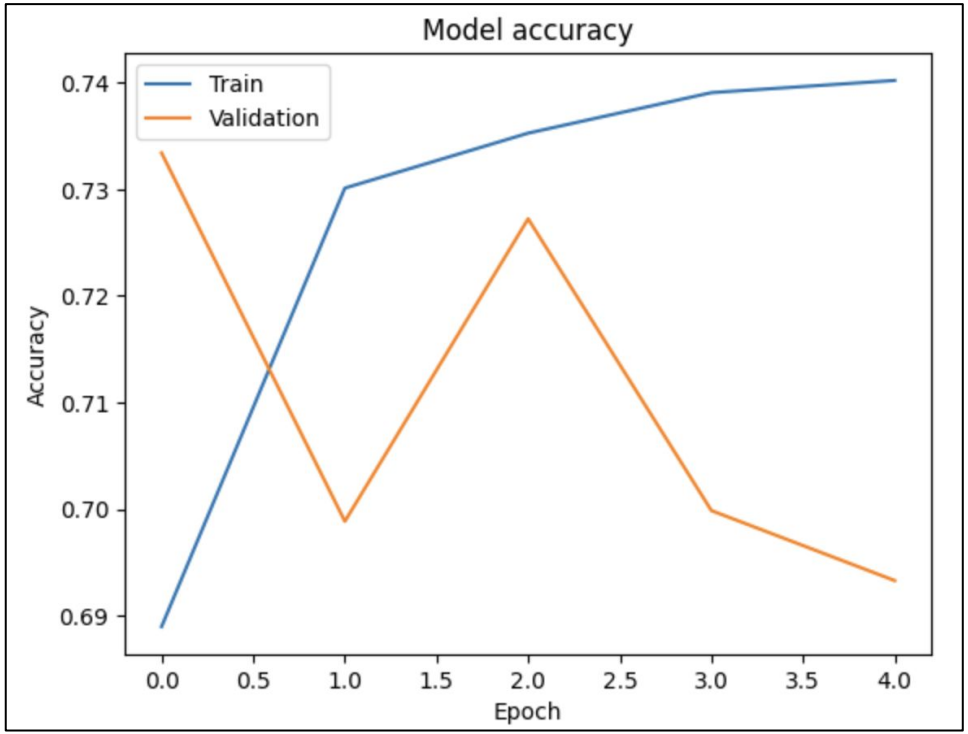


Convolution Neural Network (CNN):

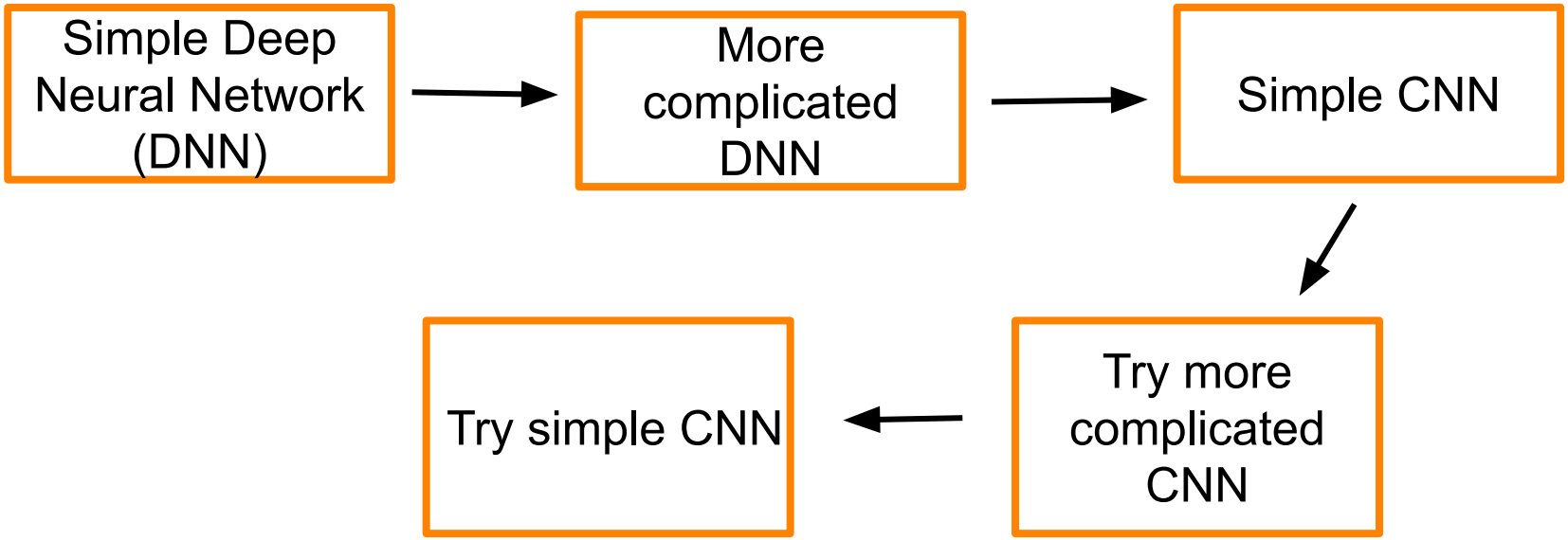
Model: "CNN_model2"

Layer (type)	Output Shape	Param #
Conv_01 (Conv2D)	(None, 54, 191, 32)	320
Pool_01 (MaxPooling2D)	(None, 27, 95, 32)	0
Dropout_01 (Dropout)	(None, 27, 95, 32)	0
Conv_02 (Conv2D)	(None, 25, 93, 32)	9248
Pool_02 (MaxPooling2D)	(None, 12, 46, 32)	0
Dropout_02 (Dropout)	(None, 12, 46, 32)	0
Conv_03 (Conv2D)	(None, 10, 44, 32)	9248
Pool_03 (MaxPooling2D)	(None, 5, 22, 32)	0
Flatten (Flatten)	(None, 3520)	0
Dense_01 (Dense)	(None, 64)	225344
Dense_02 (Dense)	(None, 1)	65
Output (Dense)	(None, 1)	2

Total params: 244,227
Trainable params: 244,227
Non-trainable params: 0

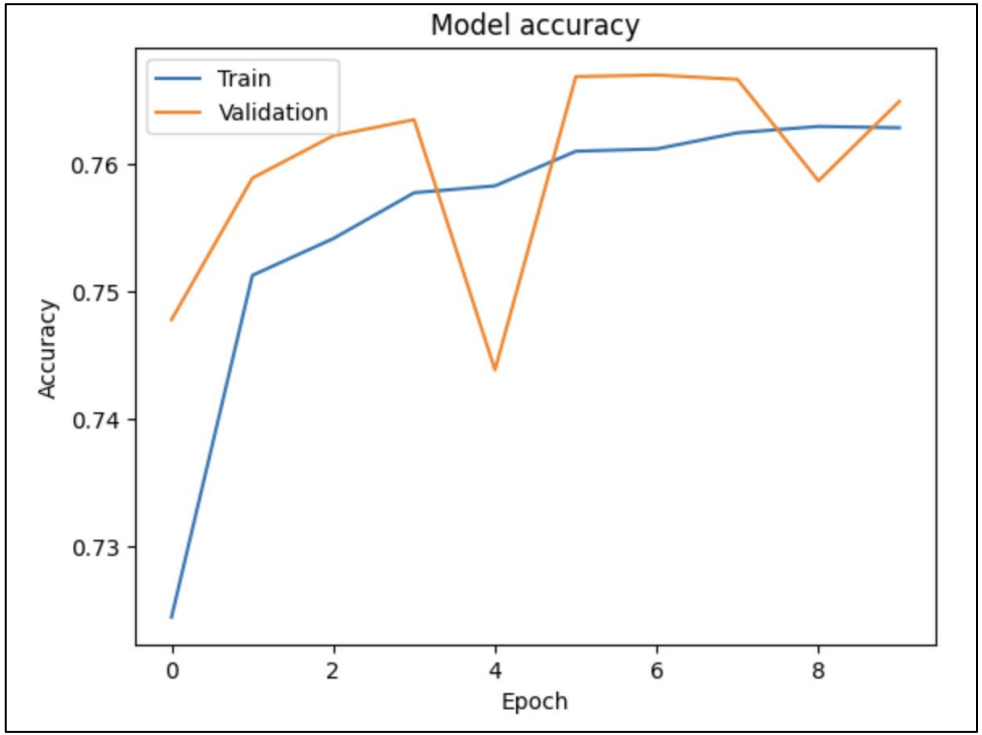


Optimization:



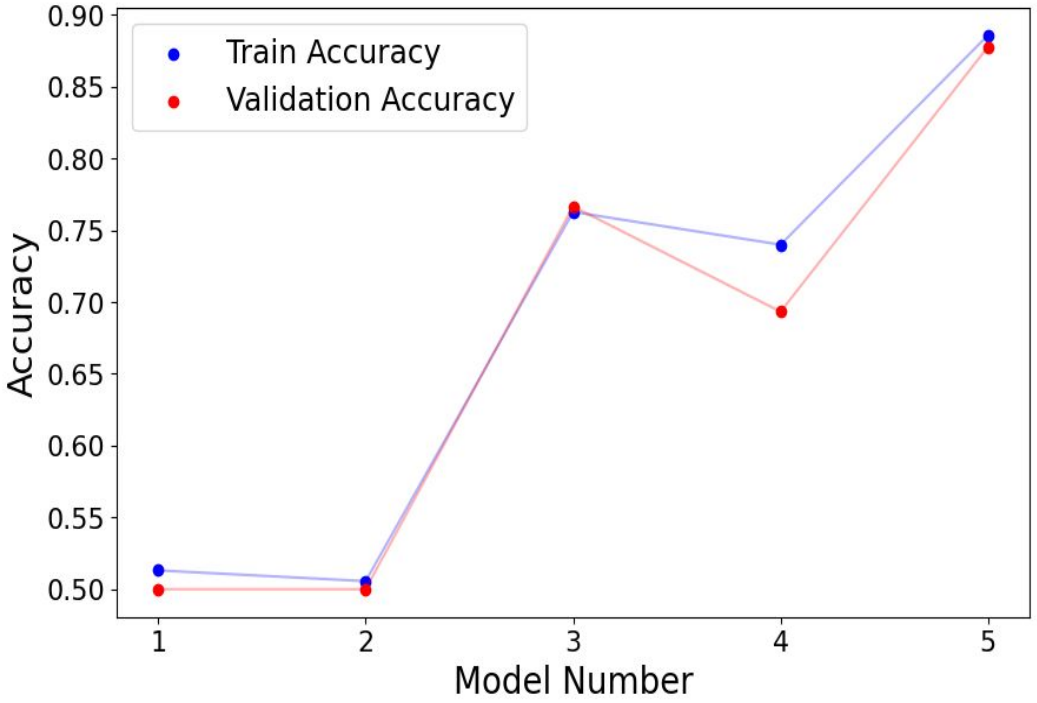
Convolution Neural Network (CNN):

```
Model: "sequential_1"
_________________________________________________________________
 Layer (type)                 Output Shape              Param #
=================================================================
 conv2d_2 (Conv2D)            (None, 54, 191, 64)       640

 max_pooling2d_2 (MaxPooling  (None, 27, 95, 64)        0
 2D)

 conv2d_3 (Conv2D)            (None, 25, 93, 32)        18464

 max_pooling2d_3 (MaxPooling  (None, 12, 46, 32)        0
 2D)

 flatten_1 (Flatten)          (None, 17664)             0

 dense_1 (Dense)              (None, 1)                 17665

=================================================================
Total params: 36,769
Trainable params: 36,769
Non-trainable params: 0
_________________________________________________________________
```



Optimization:

Model No.	Type
1	DNN (simple)
2	DNN (complicated)
3	CNN (simple)
4	CNN (complicated)
5	EffNet



Discussion - *Limitations*

- Computational Resources
- Management of large data
- Noisy data

Conclusion:

- GW data is very noisy and hence using machine learning techniques
- Data - time series is converted into the images via CQT
- CNN works better than DNN
- More advanced techniques - transfer learning (Efficient Net)
- Future work - Use the quantum version of neural networks

Thank you!