



TAIJI LABORATORY  
FOR GRAVITATIONAL WAVE UNIVERSE



ICTP-AP  
International Centre  
for Theoretical Physics Asia-Pacific  
国际理论物理中心-亚太地区



中国科学院大学  
University of Chinese Academy of Sciences

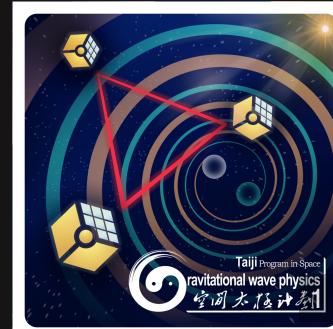
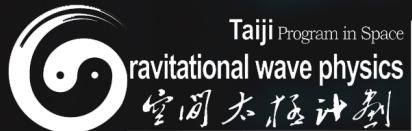
# 引力波数据探索：编程与分析实战训练营

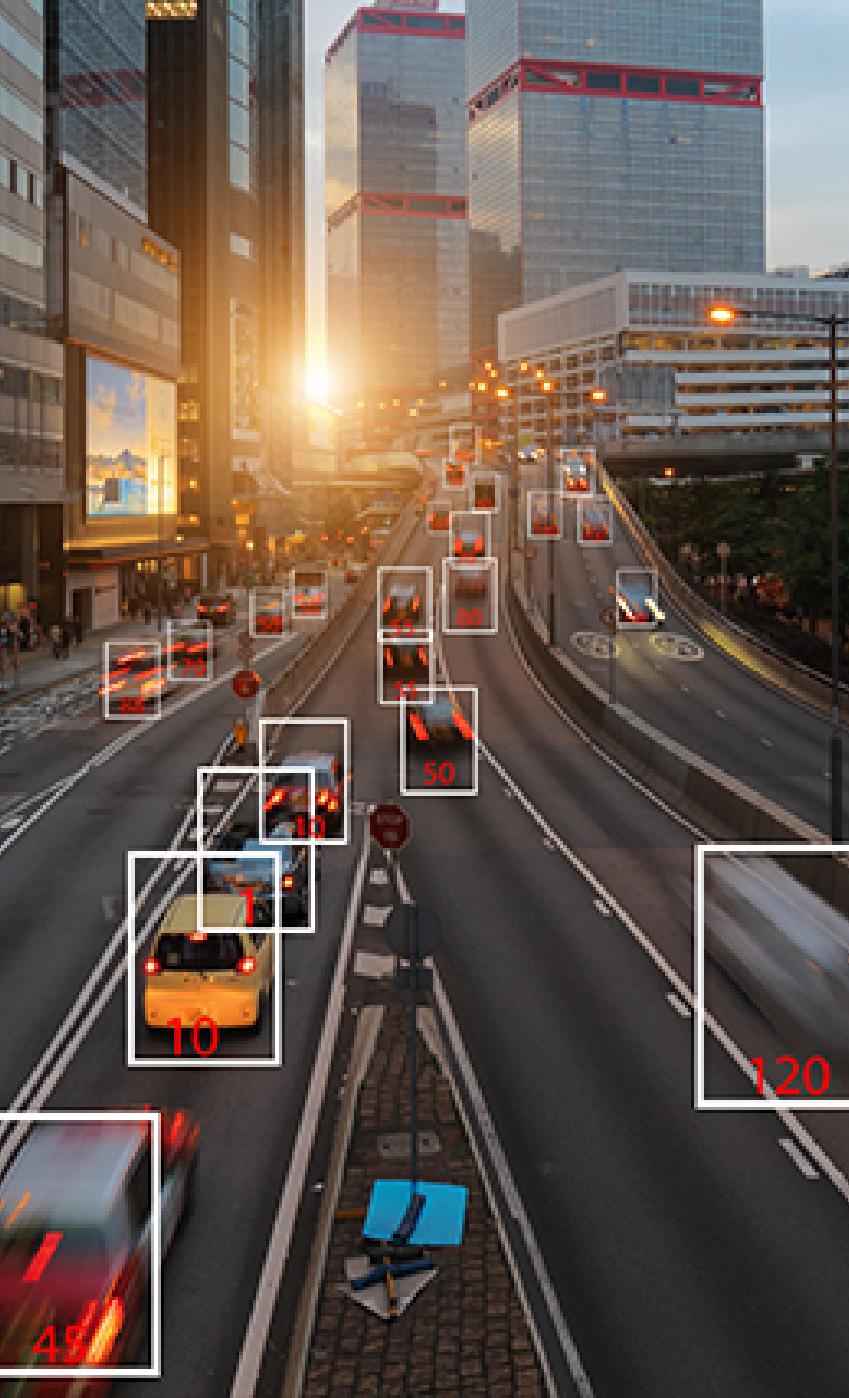
## 第4部分 深度学习基础 卷积神经网络与引力波信号探测

主讲老师：王赫

ICTP-AP, UCAS

2023/12/29





# 深度学习：卷积神经网络

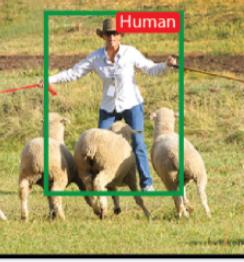
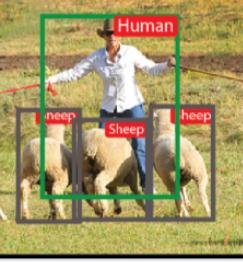
- 卷积（一维）
- 卷积（二维）
- 传统神经网络 vs 卷积神经网络
- Vanilla CNN
- ResNet



# 深度学习：卷积神经网络

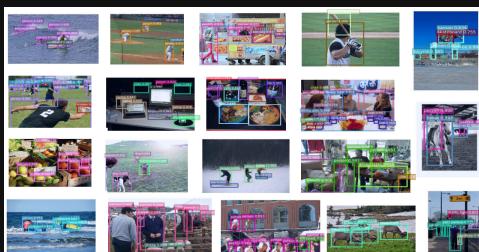
- 无处不在的卷积神经网络

<https://github.com/Nikasa1889/HistoryObjectRecognition>

 <p><b>Image Classification</b> Classify an image based on the dominant object inside it. <b>datasets:</b> MNIST, CIFAR, ImageNet</p>	 <p><b>Object Localization</b> Predict the image region that contains the dominant object. Then image classification can be used to recognize object in the region <b>datasets:</b> ImageNet</p>	 <p><b>Object Recognition</b> Localize and classify all objects appearing in the image. This task typically includes: proposing regions then classify the object inside them. <b>datasets:</b> PASCAL, COCO</p>
 <p><b>Semantic Segmentation</b> Label each pixel of an image by the object class that it belongs to, such as human, sheep, and grass in the example. <b>datasets:</b> PASCAL, COCO</p>	 <p><b>Instance Segmentation</b> Label each pixel of an image by the object class and object instance that it belongs to. <b>datasets:</b> PASCAL, COCO</p>	 <p><b>Keypoint Detection</b> Detect locations of a set of predefined keypoints of an object, such as keypoints in a human body, or a human face. <b>datasets:</b> COCO</p>



Gebru et al. ICCV (2017)



Zhou et al. CVPR (2018)



Shen et al. CVPR (2018)

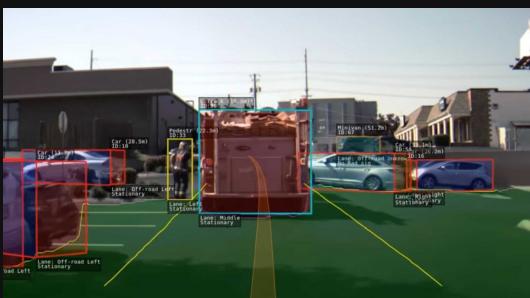
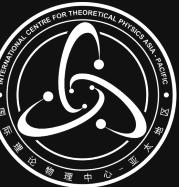


Image courtesy of Tesla (2020)



# 深度学习：卷积神经网络

- 卷积（一维）

- Integral Form

$$s(t) = \int x(\tau)w(t - \tau)d\tau =: (x * w)(t)$$

←

$$(a_1x_1 + a_2x_2) * w = a_1(x_1 * w) + a_2(x_2 * w) \quad (\text{linearity})$$

$$(x * w)(t - T) = x(t - T) * w(t) \quad (\text{time invariance})$$

- Discrete Form

$$s[n] = \sum_{m=\max(0, n-D)}^{\min(n, M)} x[m] \cdot w[n-m], \quad n = 0, 1, \dots, D+M$$

$x[n], n = 0, 1, \dots, D-1;$   
 $w[n], n = 0, 1, \dots, M-1$

$D = 5, M = 4 :$

$$s_0 = w_0 x_0$$

$$s_1 = w_0 x_1 + w_1 x_0$$

$$s_2 = w_0 x_2 + w_1 x_1 + w_2 x_0$$

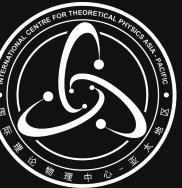
$$s_3 = w_0 x_3 + w_1 x_2 + w_2 x_1 + w_3 x_0$$

$$s_4 = w_0 x_4 + w_1 x_3 + w_2 x_2 + w_3 x_1$$

$$s_5 = w_1 x_4 + w_2 x_3 + w_3 x_2$$

$$s_6 = w_2 x_4 + w_3 x_3$$

$$s_7 = w_3 x_4$$



# 深度学习：卷积神经网络

- 卷积（一维）

$D = 5, M = 4 :$

$$s_0 = w_0 x_0$$

$$s_1 = w_0 x_1 + w_1 x_0$$

$$s_2 = w_0 x_2 + w_1 x_1 + w_2 x_0$$

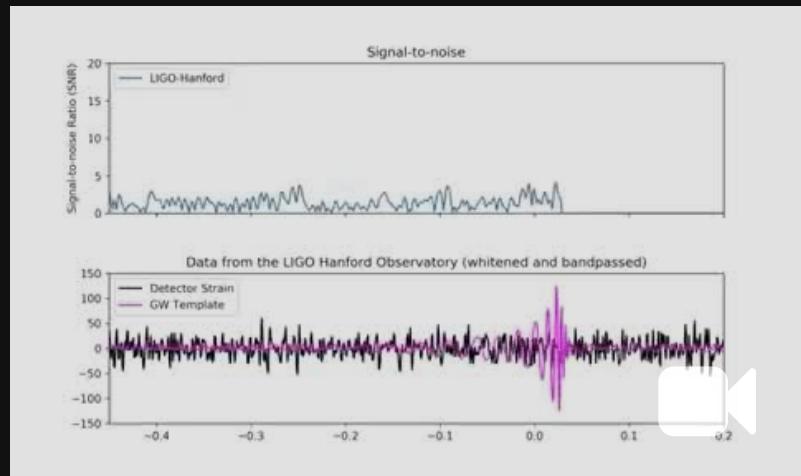
$$s_3 = w_0 x_3 + w_1 x_2 + w_2 x_1 + w_3 x_0$$

$$s_4 = w_0 x_4 + w_1 x_3 + w_2 x_2 + w_3 x_1$$

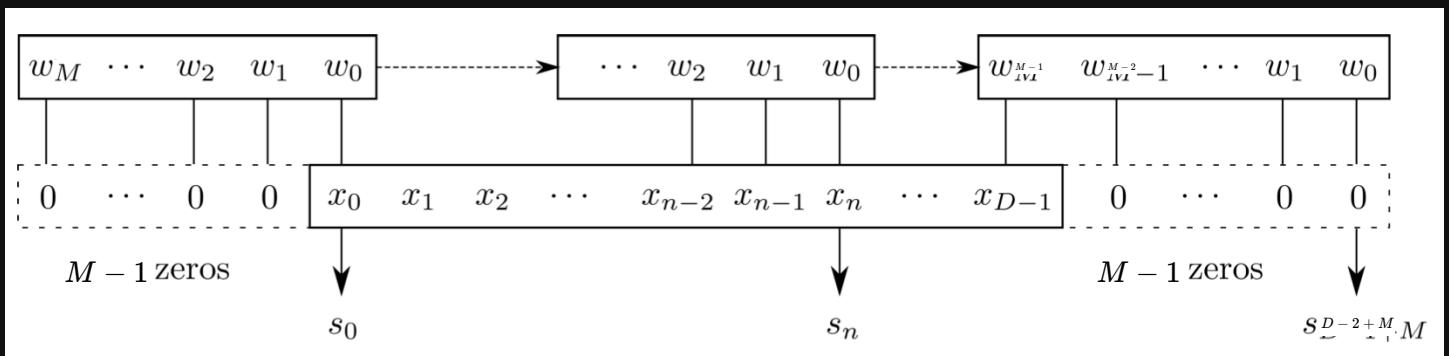
$$s_5 = w_1 x_4 + w_2 x_3 + w_3 x_2$$

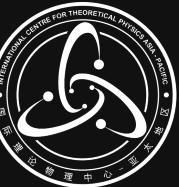
$$s_6 = w_2 x_4 + w_3 x_3$$

$$s_7 = w_3 x_4$$



- Flip-and-slide Form





# 深度学习：卷积神经网络

- 卷积（一维）

$D = 5, M = 4 :$

$$s_0 = w_0 x_0$$

$$s_1 = w_0 x_1 + w_1 x_0$$

$$s_2 = w_0 x_2 + w_1 x_1 + w_2 x_0$$

$$s_3 = w_0 x_3 + w_1 x_2 + w_2 x_1 + w_3 x_0$$

$$s_4 = w_0 x_4 + w_1 x_3 + w_2 x_2 + w_3 x_1$$

$$s_5 = w_1 x_4 + w_2 x_3 + w_3 x_2$$

$$s_6 = w_2 x_4 + w_3 x_3$$

$$s_7 = w_3 x_4$$

- Matrix Form :  $D = 5, M = 4$

$$\mathbf{s} = [ \ s_0 \quad s_1 \quad \cdots \quad s_7 \ ] = [ \ x_0 \quad x_1 \quad \cdots \quad x_4 \ ] \cdot \begin{bmatrix} w_0 & w_1 & w_2 & w_3 & 0 & 0 & 0 & 0 \\ 0 & w_0 & w_1 & w_2 & w_3 & 0 & 0 & 0 \\ 0 & 0 & w_0 & w_1 & w_2 & w_3 & 0 & 0 \\ 0 & 0 & 0 & w_0 & w_1 & w_2 & w_3 & 0 \\ 0 & 0 & 0 & 0 & w_0 & w_1 & w_2 & w_3 \end{bmatrix} = \mathbf{x} \cdot \mathbf{w}$$

- It corresponds to a convolutional layer in deep learning with one kernel (channel);
- kernel size (K) of 4; padding (P) is 3; stride (S) is 1.



# 深度学习：卷积神经网络

- 卷积（一维）

- Matrix Form :  $D = 5, M = 3$

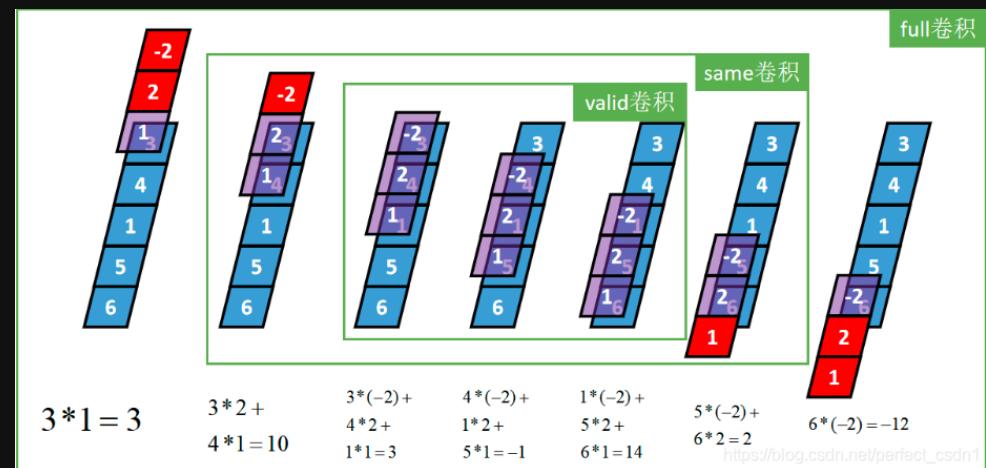
$$\mathbf{s} = [s_0 \ s_1 \ \cdots \ s_6] = [x_0 \ x_1 \ \cdots \ x_4] \cdot \begin{bmatrix} w_0 & w_1 & w_2 & 0 & 0 & 0 & 0 \\ 0 & w_0 & w_1 & w_2 & 0 & 0 & 0 \\ 0 & 0 & w_0 & w_1 & w_2 & 0 & 0 \\ 0 & 0 & 0 & w_0 & w_1 & w_2 & 0 \\ 0 & 0 & 0 & 0 & w_0 & w_1 & w_2 \end{bmatrix} = \mathbf{x} \cdot \mathbf{w}$$

- It corresponds to a convolutional layer in deep learning with one kernel (channel);
- kernel size (K) of 3; padding (P) is 2; stride (S) is 1.

- 卷积层

$$D = 5, M = 3$$

样本个数    卷积核个数  
 $\downarrow$                $\downarrow$   
 $[1, 1, 5] \star [1, 1, 3] \rightarrow [1, 1, 3/5/7]$   
 “深度”维度





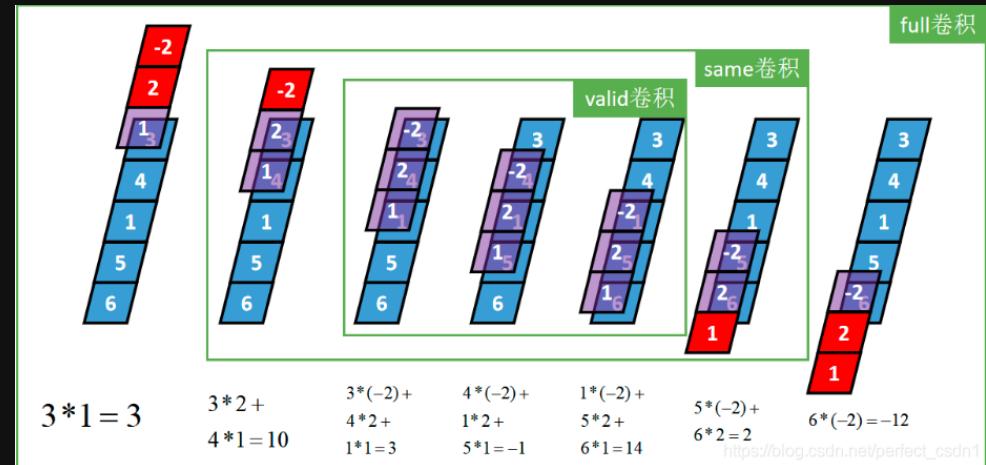
# 深度学习：卷积神经网络

- 卷积（一维）

  - 卷积层

$$D = 5, M = 3$$

样本个数 卷积核个数  
 $\downarrow \quad \downarrow$   
 $[1, 1, 5] * [1, 1, 3] \rightarrow [1, 1, 3/5/7]$   
 $\nwarrow \quad \nearrow$   
 “深度”维度

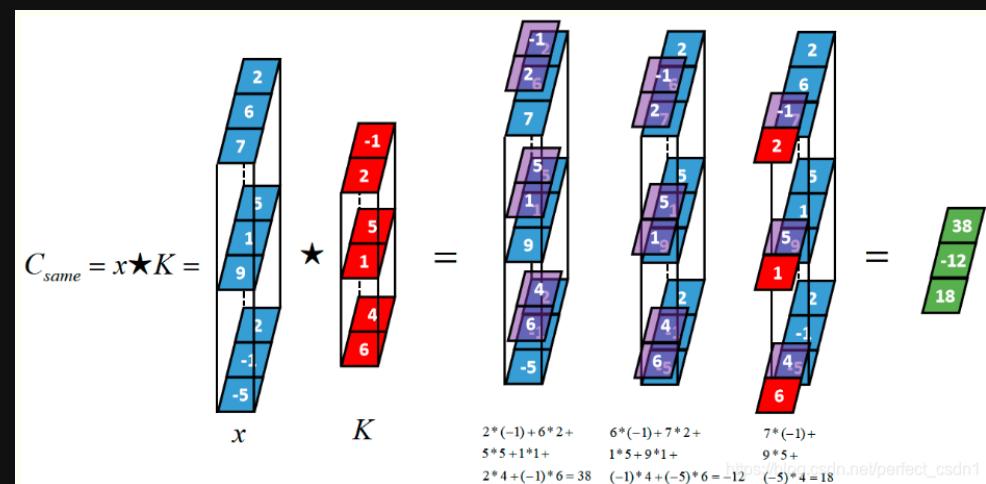


- 具备“深度”维度的输入序列

  - 输入张量的深度和卷积核的深度是相等的

$$[1, 3, 3] * [1, 3, 2] \rightarrow [1, 1, 3]$$

$\nwarrow \quad \nearrow$   
 “深度”维度





# 深度学习：卷积神经网络

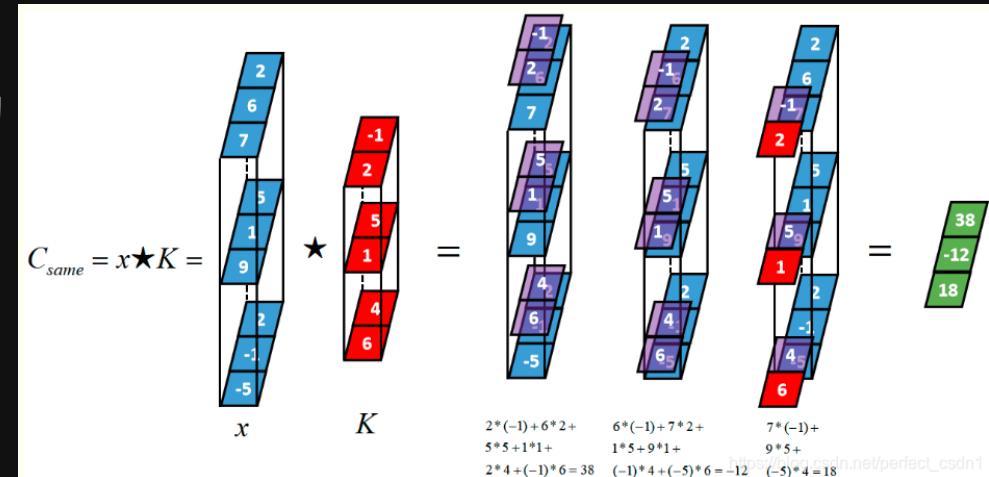
- 卷积（一维）

- 具备“深度”维度的输入序列

- 输入张量的深度和卷积核的深度是相等的

$$[1, 3, 3] \star [1, 3, 2] \rightarrow [1, 1, 3]$$

“深度”维度



- 具备深度的张量与多个卷积核的卷积

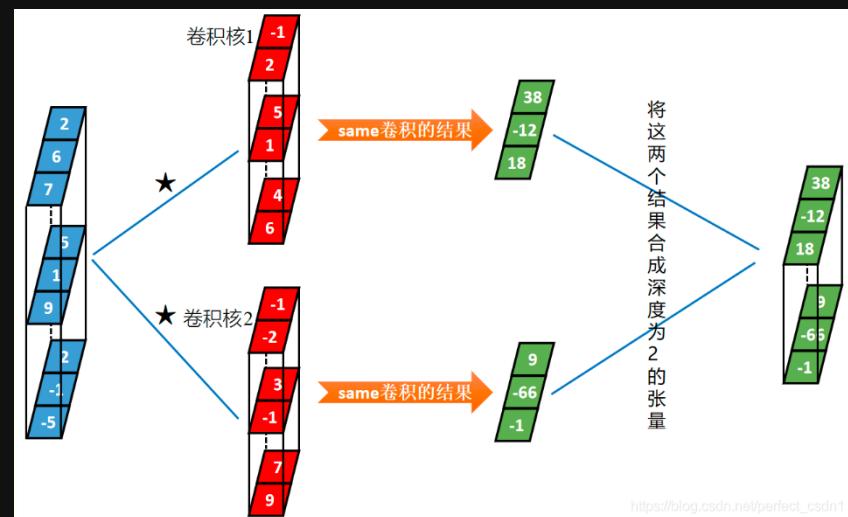
- 同一个张量与多个卷积核本质上是该张量分别于每一个卷积核卷积，然后将每一个卷积结果在深度方向连接起来。

样本个数    卷积核个数

$$\downarrow \quad \downarrow$$

$$[1, 3, 3] \star [2, 3, 2] \rightarrow [1, 2, 3]$$

“深度”维度

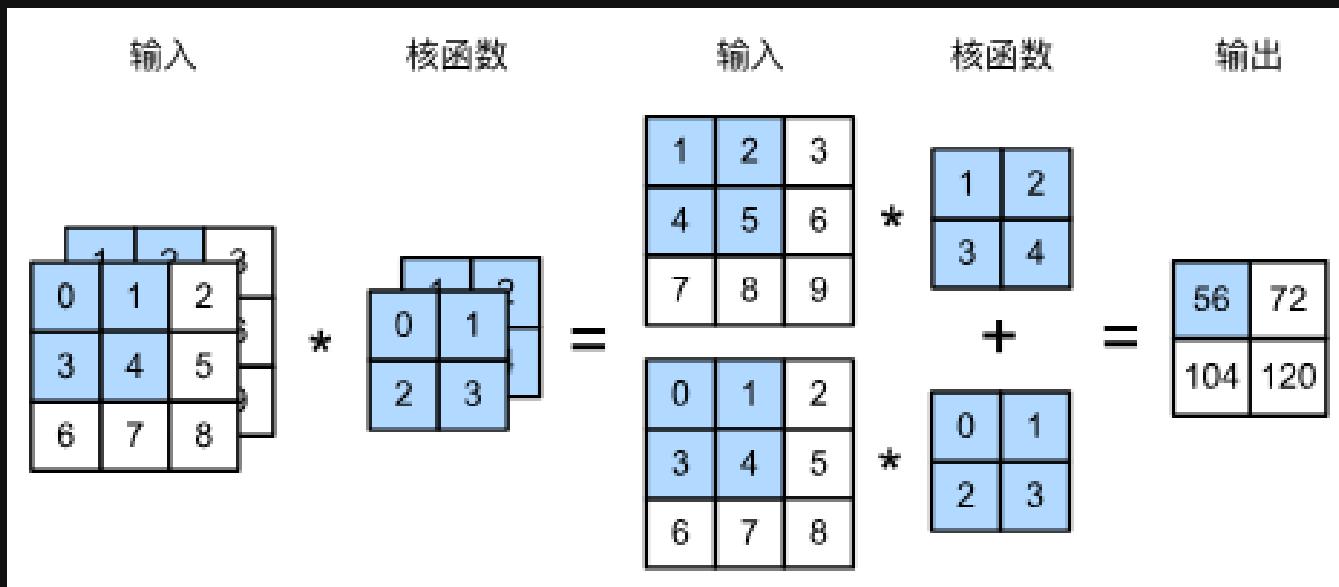




# 深度学习：卷积神经网络

- 卷积（二维）

- 深度  $\rightarrow$  channel
- 卷积的本质
  - 有效提取相邻像素间的相关特征

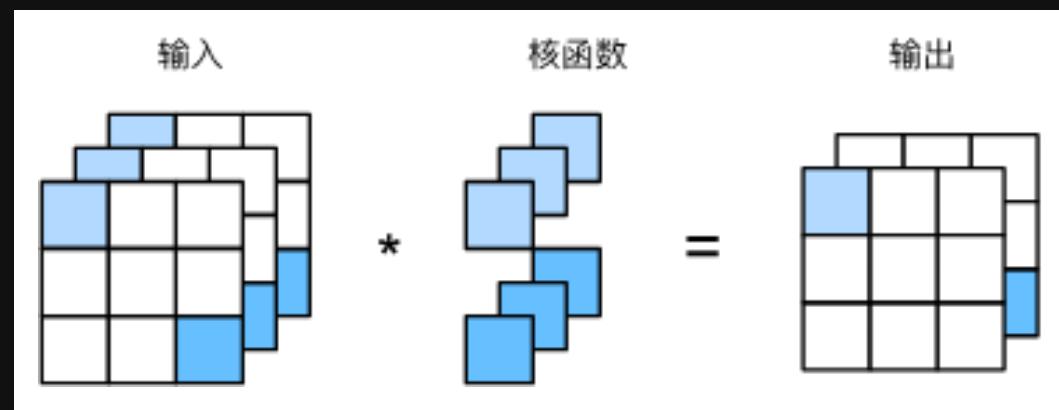


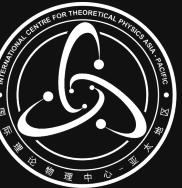


# 深度学习：卷积神经网络

- 卷积（二维）

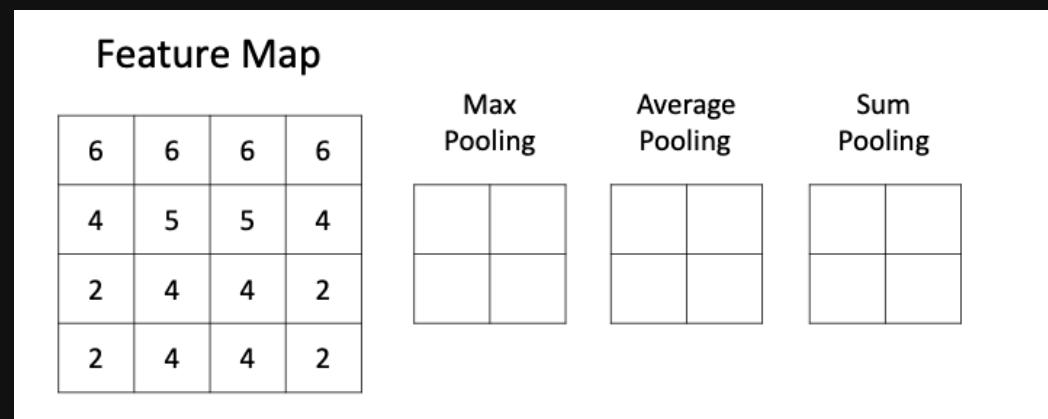
- 深度  $\rightarrow$  channel
- $1 \times 1$  卷积层
  - 唯一计算发生在channel上





# 深度学习：卷积神经网络

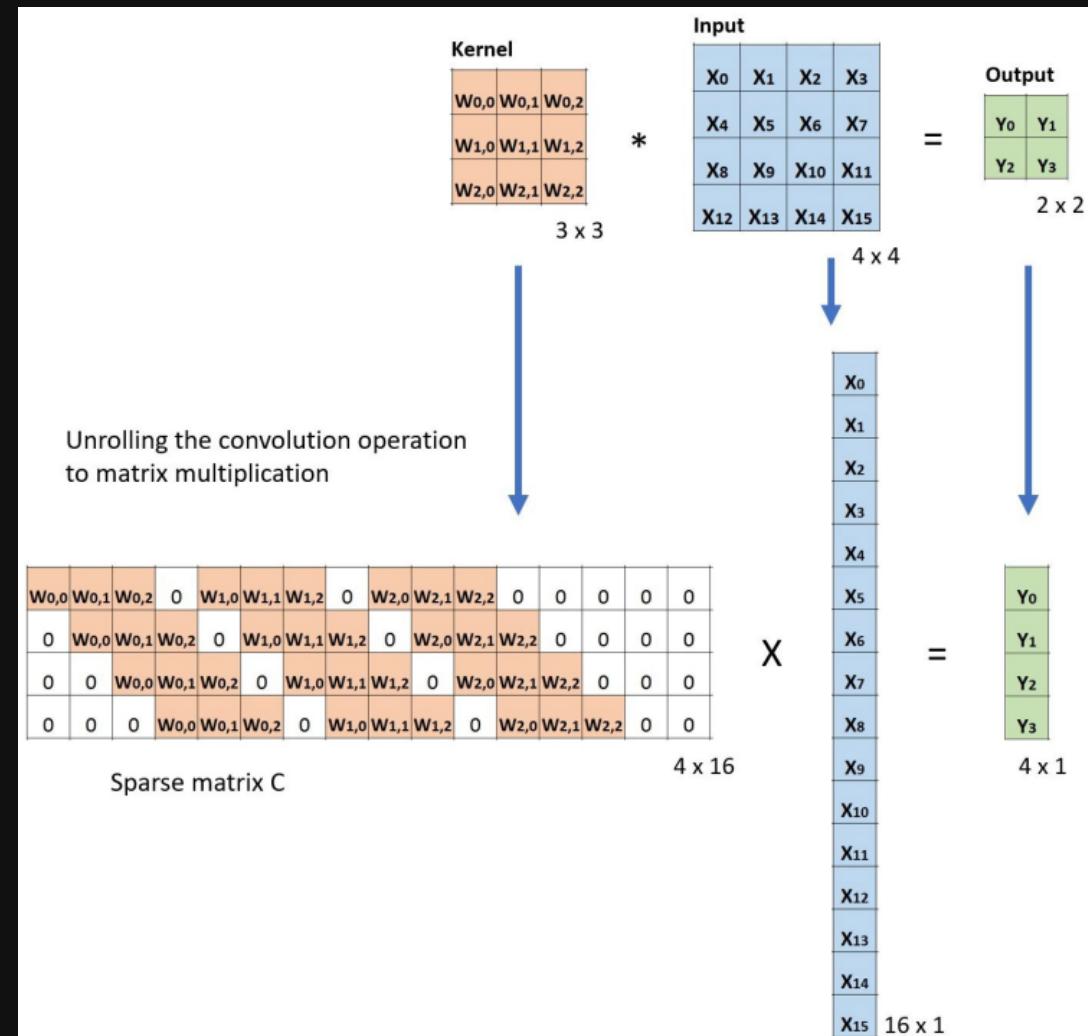
- 卷积（二维）
  - 深度 → channel
  - 池化层 (Pooling layer)
    - 池化层的提出是为了缓解卷积层对位置的过度敏感性。
    - 在处理多通道输入数据时，池化层对每个输入通道分别池化，而不是像卷积层那样将各通道的输入按通道相加。





# 深度学习：卷积神经网络

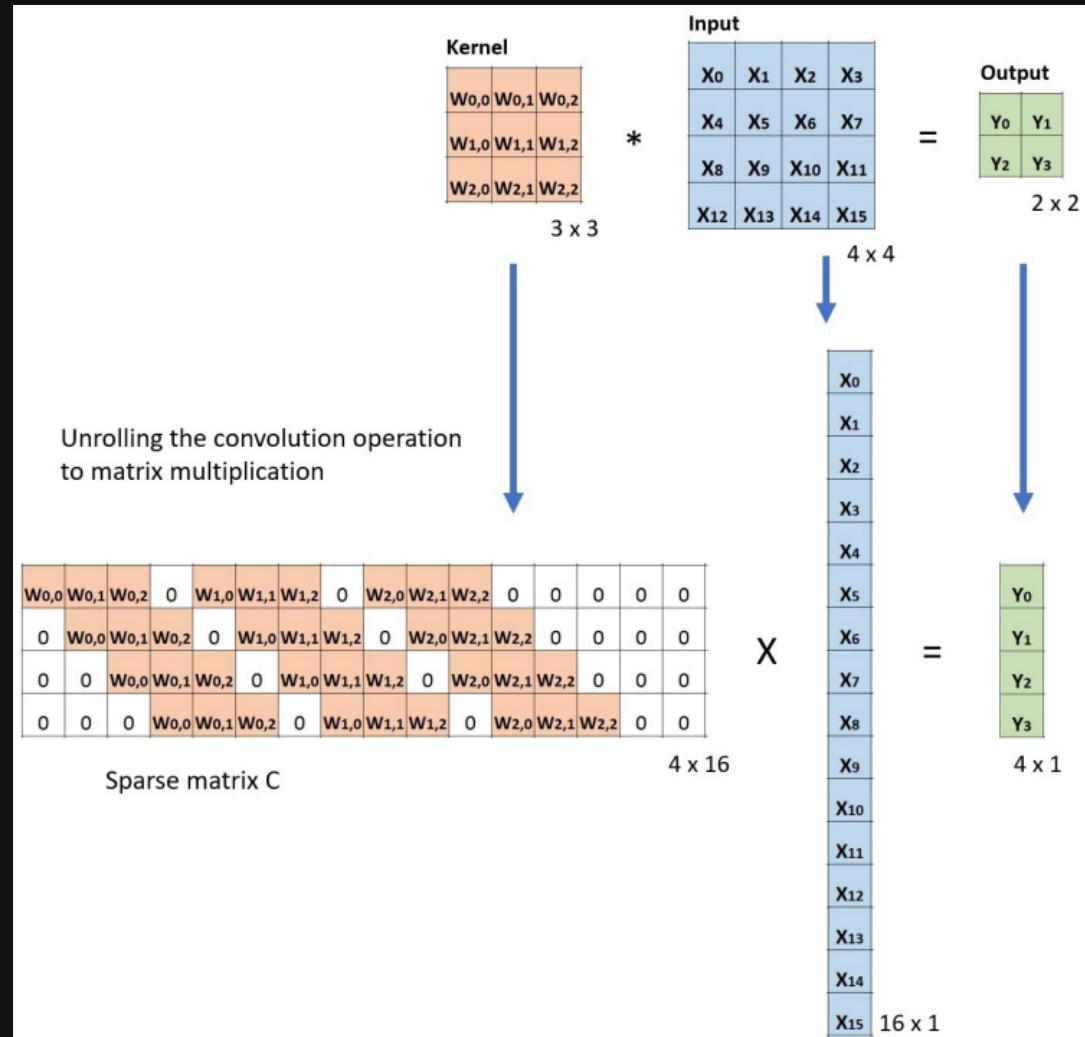
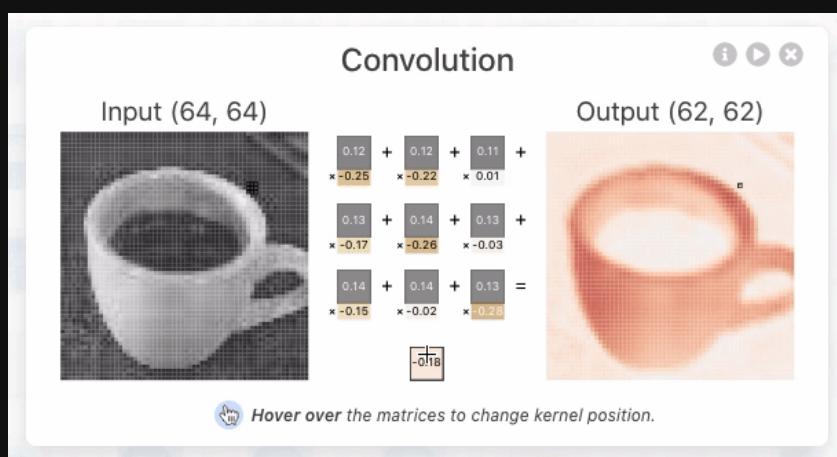
- 传统神经网络 vs 卷积神经网络
  - 卷积层是全连接层的一种特例
    - 全连接层：参数太多，易过拟合
    - 卷积层：
      - 稀疏连接 (sparse connectivity)
      - 权重参数共享 (parameter sharing)





# 深度学习：卷积神经网络

- 传统神经网络 vs 卷积神经网络
  - 卷积层是全连接层的一种特例
    - 全连接层：参数太多，易过拟合
    - 卷积层：
      - 稀疏连接 (sparse connectivity)
      - 权重参数共享 (parameter sharing)

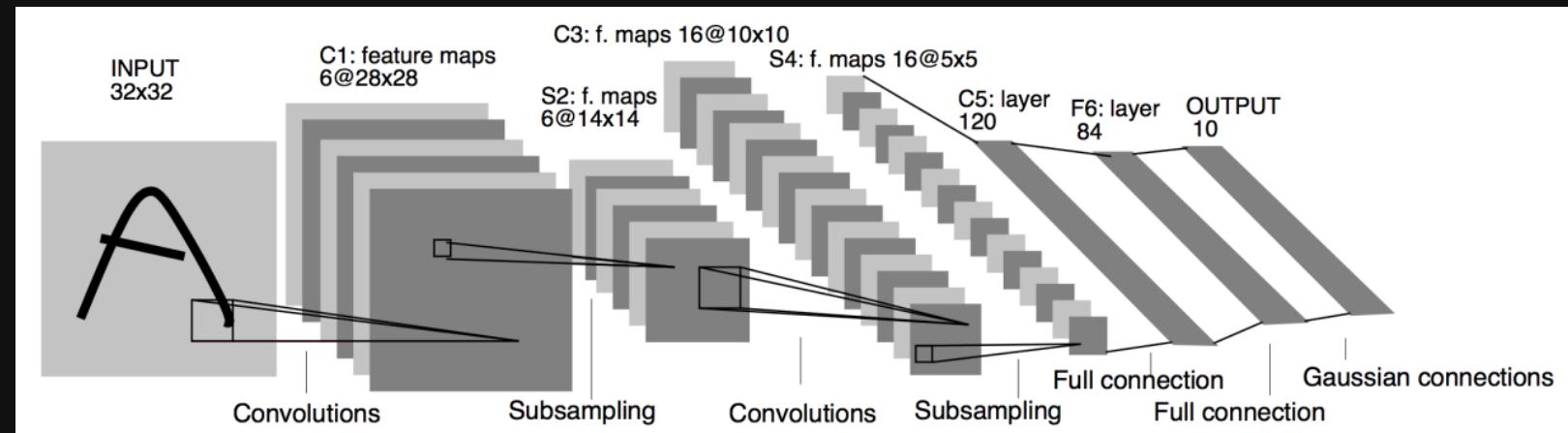




# 深度学习：卷积神经网络

- Vanilla CNN

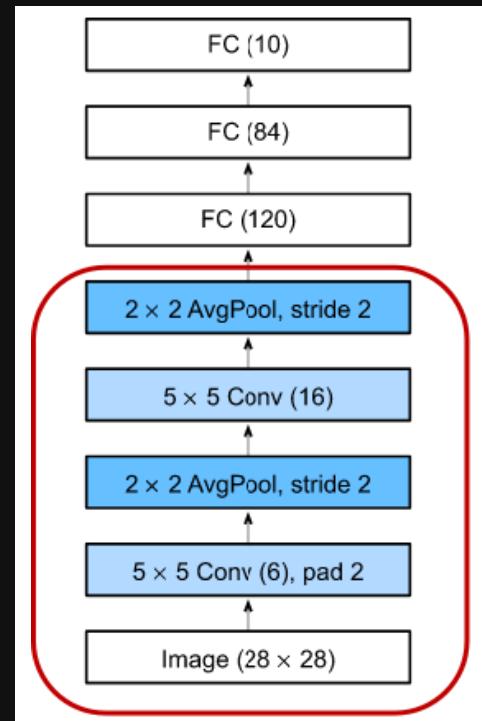
- 卷积神经网络 (Convolutional Neural Network, CNN) 就是含卷积层的网络。这里介绍一个早期用来识别手写数字图像的卷积神经网络：[LeNet](#)。这个名字来源于 LeNet 论文的第一作者 Yann LeCun。LeNet 展示了通过梯度下降训练卷积神经网络可以达到手写数字识别在当时最先进的结果。这个奠基性的工作第一次将卷积神经网络推上舞台，为世人所知。LeNet 的网络结构如下图所示。





# 深度学习：卷积神经网络

- Vanilla CNN
  - LeNet 分为两个部分组成：
    - 卷积层块：由两个卷积层块组成；
    - 全连接层块：由三个全连接层组成。
  - 每个卷积块中的基本单元是一个卷积层、一个 sigmoid 激活函数和平均池化层。
  - 请注意，虽然 ReLU 和最大池化层更有效，但它们在20世纪90年代还没有出现。每个卷积层使用  $5 \times 5$  卷积核，这些层将输入映射到多个二维特征输出，通常同时增加通道的数量。第一卷积层有 6 个输出通道，而第二个卷积层有 16 个输出通道。每个  $2 \times 2$  池化操作通过空间下采样将维数减少 4 倍。卷积的输出形状由批量大小、通道数、高度、宽度决定。
  - 为了将卷积层块的输出传递给全连接层块，我们必须在小批量中展平 (flatten) 每个样本。换言之，我们将这个四维输入转换成全连接层所期望的二维输入。这里的二维表示的第一个维度索引小批量中的样本，第二个维度给出每个样本的平面向量表示。LeNet 有三个全连接层，分别有 120、84 和 10 个输出。因为我们仍在执行分类，所以输出层的 10 维对应于最后输出结果的数量。



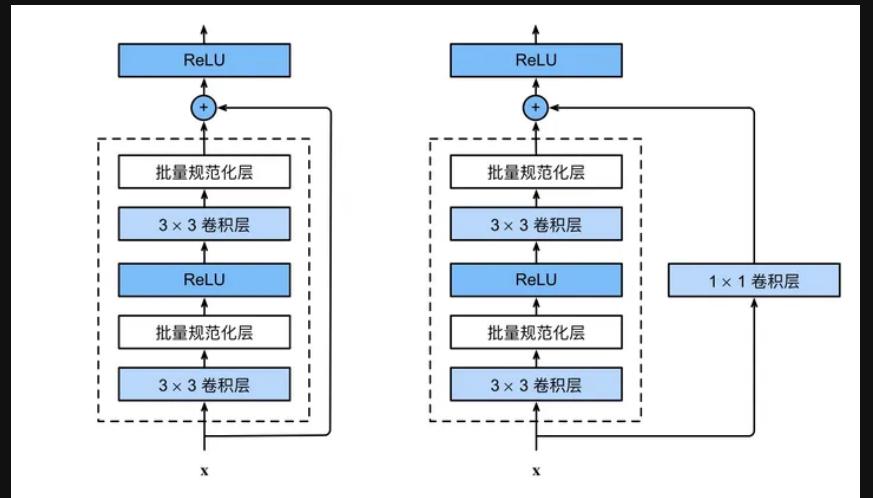
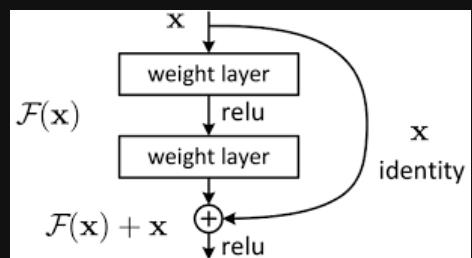


# 深度学习：卷积神经网络

- ResNet
  - 残差学习网络 (deep residual learning network)
  - 残差的思想: 去掉相同的主体部分, 从而突出微小的变化。
  - 可以被用来训练非常深的网络



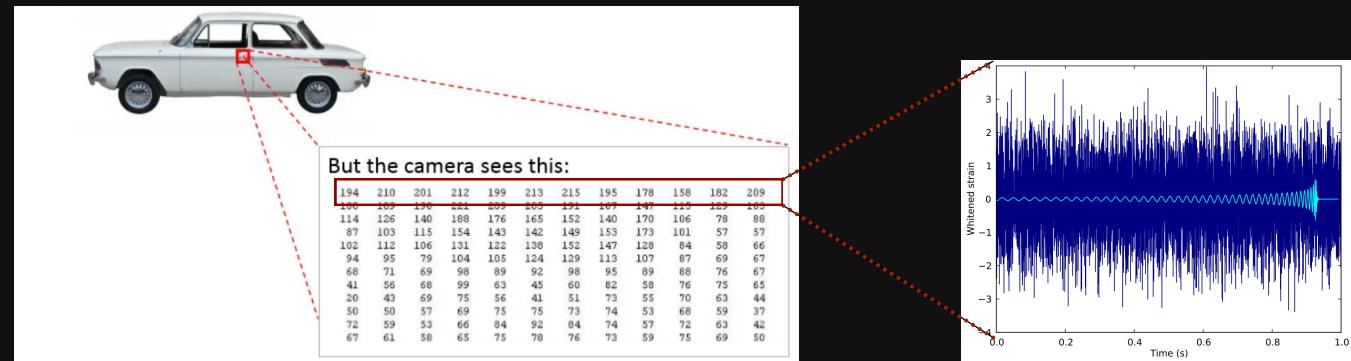
He K, Zhang X, Ren S, et al. Deep residual learning for image recognition. CVPR (2016)





# 深度学习：卷积神经网络

- ResNet
  - 残差模块关于 GW + AI 的应用已经非常广泛
  - 最普遍且直接的理解角度 CV → GWDA:
    - 像素点 → 采样点



DEEP RESIDUAL NETWORKS FOR GRAVITATIONAL WAVE ... PHYS. REV. D 108, 024022 (2023)

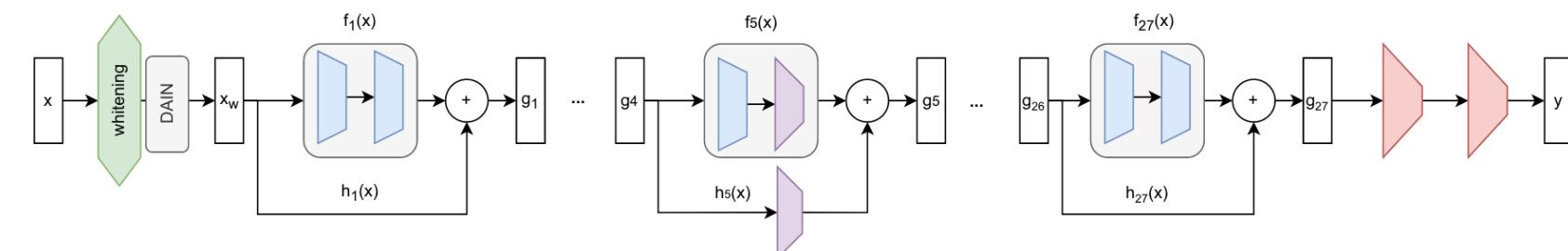


FIG. 3. Description of our residual network architecture. The input  $\mathbf{x}$  is  $2 \times 2048$ -dimensional. There are 27 blocks comprising two convolutional layers. In five of these blocks, the dimensionality is halved (stride 2, shown in purple) and an additional layer is used in the residual connection. Finally, there are two individual convolutional layers which gradually decrease the number of channels down to two, corresponding to the binary classification targets (noise plus injected waveform vs. noise only) in the output.



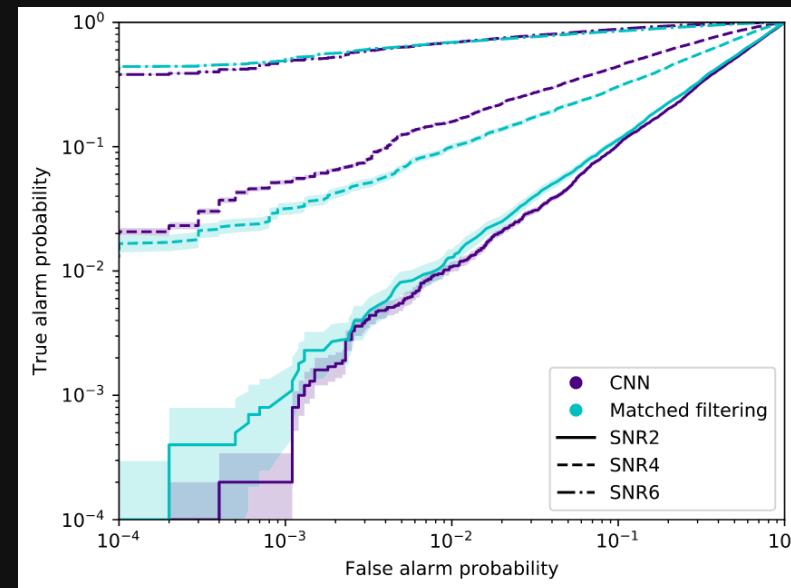
# 引力波数据分析与深度学习

- 信号探测



# 引力波信号搜寻：深度学习技术

- Deep convolutional neural network to search for binary black hole gravitational-wave signals.
- Input is the whitened time series of measured gravitational-wave strain in Gaussian noise.
- Sensitivity comparable to match filtering.

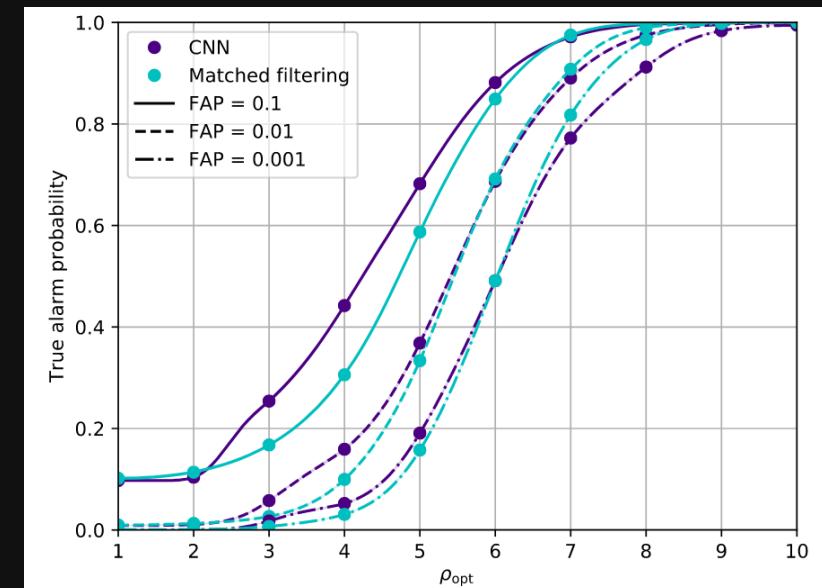


What is ROC?

PHYSICAL REVIEW LETTERS 120, 141103 (2018)  
Editors' Suggestion

**Matching Matched Filtering with Deep Networks for Gravitational-Wave Astronomy**  
Hunter Gabbard,\* Michael Williams, Fergus Hayes, and Chris Messenger  
SUPA, School of Physics and Astronomy, University of Glasgow, Glasgow G12 8QQ, United Kingdom

arXiv: 1712.06041



See also:  
D. George and E.A.  
Huerta Phys. Lett. B 778  
64–70 (2018)



# 引力波信号搜寻：深度学习技术

- 目标：复现这个学习任务，搭建一个 Baseline
- 理解什么是 ROC？
- ELU? Dropout? Softmax?
- ...



TABLE I. The optimized network consisting of six convolutional layers ( $C$ ), followed by three hidden layers ( $H$ ). Max pooling is performed on the first, fifth, and eighth layer, whereas dropout is only performed on the two hidden layers. Each layer uses an exponential linear unit (Elu) activation function (with range  $[-1, \infty]$ ) while the last layer uses a Softmax (SMax) activation function in order to normalize the output values to be between 0 and 1 so as to give a probability value for each class.

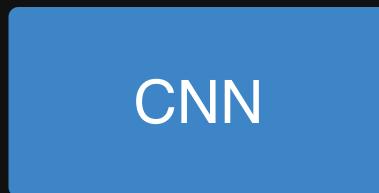
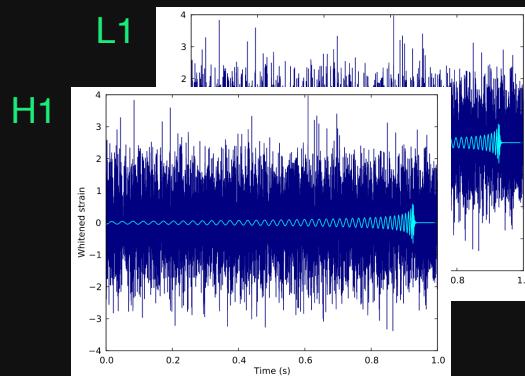
Parameter (Option)	Layer								
	1	2	3	4	5	6	7	8	9
Type	$C$	$C$	$C$	$C$	$C$	$C$	$H$	$H$	$H$
No. Neurons	8	8	16	16	32	32	64	64	2
Filter size	64	32	32	16	16	16	Not applicable	Not applicable	Not applicable
Max pool size	Not applicable	8	Not applicable	6	Not applicable	4	Not applicable	Not applicable	Not applicable
Drop out	0	0	0	0	0	0	0.5	0.5	0
Activation function	Elu	Elu	Elu	Elu	Elu	Elu	Elu	Elu	SMax



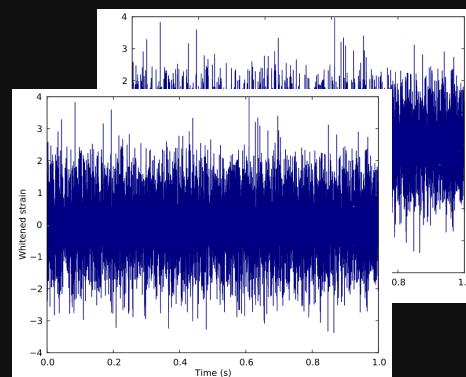
# 引力波信号搜寻：卷积神经网络

- 目标：复现这个学习任务，搭建一个 Baseline
- 任务：判断一段含有引力波信号的时序数据 (H1/L1)，是否含有引力波信号（二分类问题）

期望达到的效果 (Evaluation) :



0.99



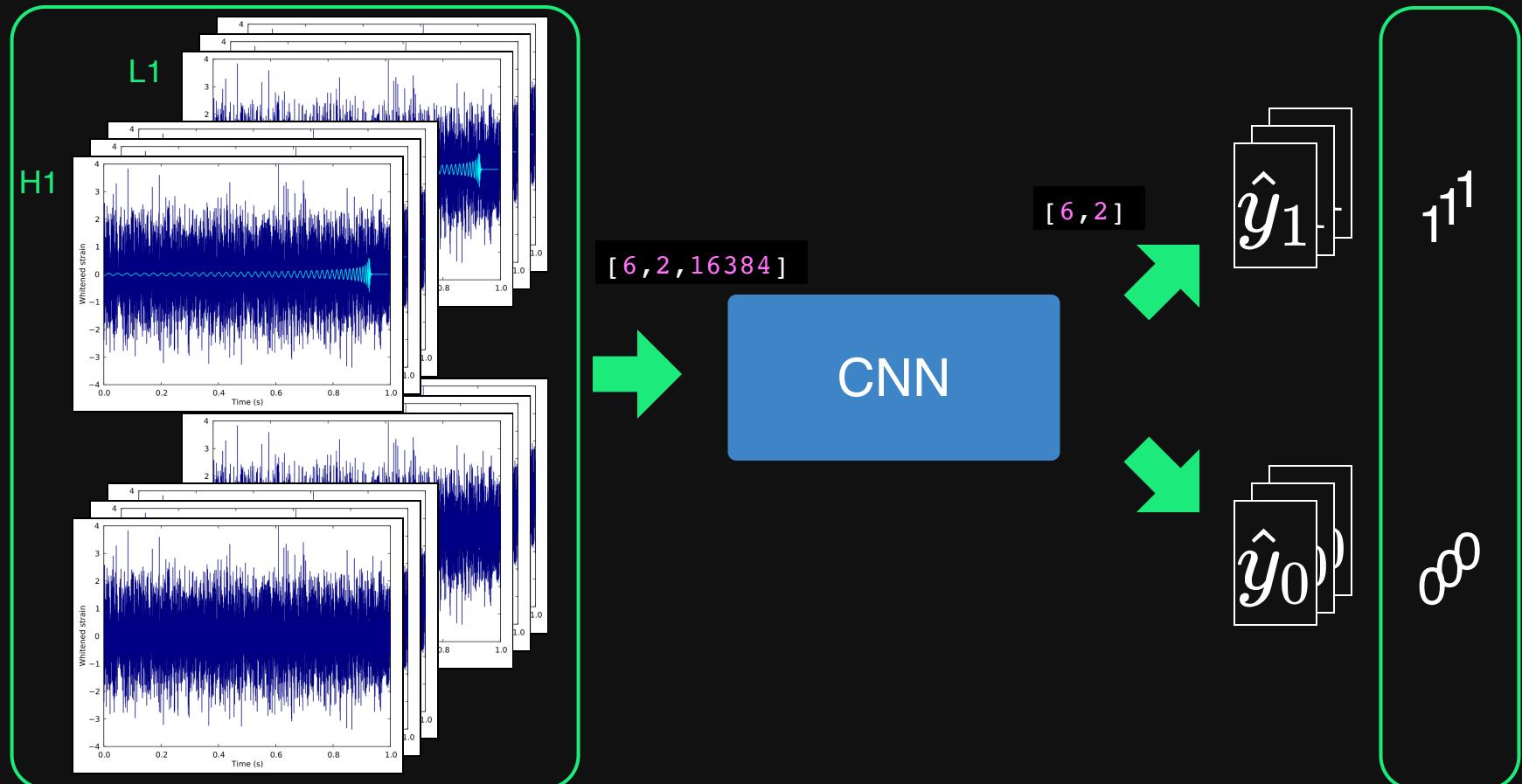
0.05

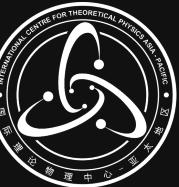


# 引力波信号搜寻：卷积神经网络

- 目标：复现这个学习任务，搭建一个 Baseline
- 任务：判断一段含有引力波信号的时序数据 (H1/L1)，是否含有引力波信号（二分类问题）

训练的过程 (Train) :





# 引力波信号搜寻：卷积神经网络

CNN

- 目标：复现这个学习任务，搭建一个 Baseline
- 任务：判断一段含有引力波信号的时序数据 (H1/L1)，是否含有引力波信号（二分类问题）

Reshape your input data:

[ 6 , 2 , 16384 ]



↓      ↓  
样本个数    长和宽  
[ 6 , 1 , [ 2 , 16384 ] ]

Conv2D:

↓  
in channel  
[ 8 , 1 , 1 , 32 ]  
↑  
卷积核个数 / out channel  
↓  
[ 6 , 8 , 2 , 16353 ]

in channel

```
Conv2d(1, 8, kernel_size=(1, 32), stride=(1, 1))
```

out channel

Layer (type)	Output Shape	Param #
Conv2d-1	[ -1, 8, 2, 16353 ]	264
ELU-2	[ -1, 8, 2, 16353 ]	0
BatchNorm2d-3	[ -1, 8, 2, 16353 ]	16
MaxPool2d-4	[ -1, 8, 2, 2044 ]	0
Conv2d-5	[ -1, 16, 2, 2029 ]	2,064
ELU-6	[ -1, 16, 2, 2029 ]	0
BatchNorm2d-7	[ -1, 16, 2, 2029 ]	32
Conv2d-8	[ -1, 16, 2, 2014 ]	4,112
ELU-9	[ -1, 16, 2, 2014 ]	0
BatchNorm2d-10	[ -1, 16, 2, 2014 ]	32
Conv2d-11	[ -1, 32, 2, 1999 ]	8,224
ELU-12	[ -1, 32, 2, 1999 ]	0
BatchNorm2d-13	[ -1, 32, 2, 1999 ]	64
Conv2d-14	[ -1, 64, 2, 1992 ]	16,448
ELU-15	[ -1, 64, 2, 1992 ]	0
BatchNorm2d-16	[ -1, 64, 2, 1992 ]	128
MaxPool2d-17	[ -1, 64, 2, 332 ]	0
Conv2d-18	[ -1, 64, 2, 325 ]	32,832
ELU-19	[ -1, 64, 2, 325 ]	0
BatchNorm2d-20	[ -1, 64, 2, 325 ]	128
Conv2d-21	[ -1, 128, 2, 322 ]	32,896
ELU-22	[ -1, 128, 2, 322 ]	0
BatchNorm2d-23	[ -1, 128, 2, 322 ]	256
Conv2d-24	[ -1, 128, 2, 319 ]	65,664
ELU-25	[ -1, 128, 2, 319 ]	0
BatchNorm2d-26	[ -1, 128, 2, 319 ]	256
MaxPool2d-27	[ -1, 128, 2, 79 ]	0
Flatten-28	[ -1, 20224 ]	0
Linear-29	[ -1, 64 ]	1,294,400
ELU-30	[ -1, 64 ]	0
Dropout-31	[ -1, 64 ]	0
Linear-32	[ -1, 2 ]	130



# 引力波信号搜寻：卷积神经网络

CNN

- 目标：复现这个学习任务，搭建一个 Baseline
- 任务：判断一段含有引力波信号的时序数据 (H1/L1)，是否含有引力波信号（二分类问题）

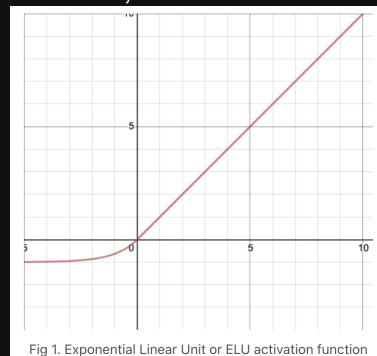
ELU activation

[ 6, 8, 2, 16353 ]



[ 6, 8, 2, 16353 ]

Credit: 1, 2

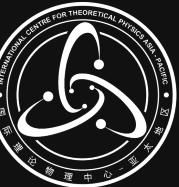


$$\text{ELU}(x) = \begin{cases} x, & \text{if } x > 0 \\ \alpha * (\exp(x) - 1), & \text{if } x \leq 0 \end{cases}$$

arXiv: 1511.07289

ELU(alpha=0.01)

Layer (type)	Output Shape	Param #
Conv2d-1	[ -1, 8, 2, 16353 ]	264
ELU-2	[ -1, 8, 2, 16353 ]	0
BatchNorm2d-3	[ -1, 8, 2, 16353 ]	16
MaxPool2d-4	[ -1, 8, 2, 2044 ]	0
Conv2d-5	[ -1, 16, 2, 2029 ]	2,064
ELU-6	[ -1, 16, 2, 2029 ]	0
BatchNorm2d-7	[ -1, 16, 2, 2029 ]	32
Conv2d-8	[ -1, 16, 2, 2014 ]	4,112
ELU-9	[ -1, 16, 2, 2014 ]	0
BatchNorm2d-10	[ -1, 16, 2, 2014 ]	32
Conv2d-11	[ -1, 32, 2, 1999 ]	8,224
ELU-12	[ -1, 32, 2, 1999 ]	0
BatchNorm2d-13	[ -1, 32, 2, 1999 ]	64
Conv2d-14	[ -1, 64, 2, 1992 ]	16,448
ELU-15	[ -1, 64, 2, 1992 ]	0
BatchNorm2d-16	[ -1, 64, 2, 1992 ]	128
MaxPool2d-17	[ -1, 64, 2, 332 ]	0
Conv2d-18	[ -1, 64, 2, 325 ]	32,832
ELU-19	[ -1, 64, 2, 325 ]	0
BatchNorm2d-20	[ -1, 64, 2, 325 ]	128
Conv2d-21	[ -1, 128, 2, 322 ]	32,896
ELU-22	[ -1, 128, 2, 322 ]	0
BatchNorm2d-23	[ -1, 128, 2, 322 ]	256
Conv2d-24	[ -1, 128, 2, 319 ]	65,664
ELU-25	[ -1, 128, 2, 319 ]	0
BatchNorm2d-26	[ -1, 128, 2, 319 ]	256
MaxPool2d-27	[ -1, 128, 2, 79 ]	0
Flatten-28	[ -1, 20224 ]	0
Linear-29	[ -1, 64 ]	1,294,400
ELU-30	[ -1, 64 ]	0
Dropout-31	[ -1, 64 ]	0
Linear-32	[ -1, 2 ]	130



# 引力波信号搜寻：卷积神经网络

CNN

- 目标：复现这个学习任务，搭建一个 Baseline
- 任务：判断一段含有引力波信号的时序数据 (H1/L1)，是否含有引力波信号（二分类问题）

## Batch Normalization

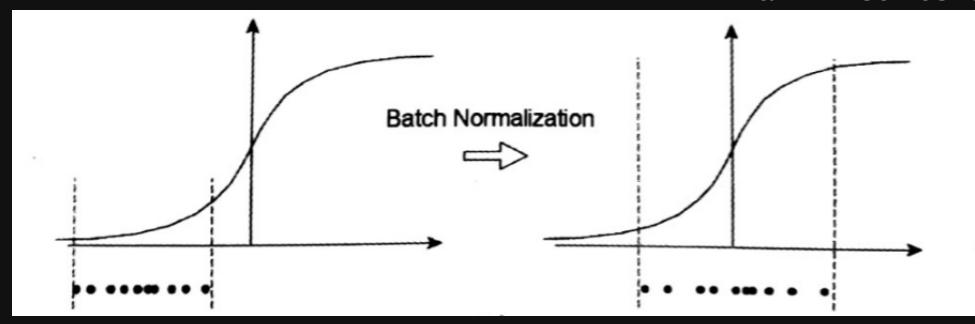
$$y = \frac{x - \text{E}[x]}{\sqrt{\text{Var}[x] + \epsilon}} * \gamma + \beta$$

[ 6, 8, 2, 16353 ]



[ 6, 8, 2, 16353 ]

arXiv: 1502.03167

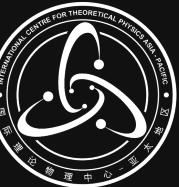


in channel

BatchNorm2d(8, eps=1e-05, momentum=0.1)

- BN 批归一化可以加快模型的收敛速度，缓解“梯度弥散”的问题
- See [more](#)

Layer (type)	Output Shape	Param #
Conv2d-1	[ -1, 8, 2, 16353 ]	264
ELU-2	[ -1, 8, 2, 16353 ]	0
<b>BatchNorm2d-3</b>	[ -1, 8, 2, 16353 ]	16
MaxPool2d-4	[ -1, 8, 2, 2044 ]	0
Conv2d-5	[ -1, 16, 2, 2029 ]	2,064
ELU-6	[ -1, 16, 2, 2029 ]	0
<b>BatchNorm2d-7</b>	[ -1, 16, 2, 2029 ]	32
Conv2d-8	[ -1, 16, 2, 2014 ]	4,112
ELU-9	[ -1, 16, 2, 2014 ]	0
<b>BatchNorm2d-10</b>	[ -1, 16, 2, 2014 ]	32
Conv2d-11	[ -1, 32, 2, 1999 ]	8,224
ELU-12	[ -1, 32, 2, 1999 ]	0
<b>BatchNorm2d-13</b>	[ -1, 32, 2, 1999 ]	64
Conv2d-14	[ -1, 64, 2, 1992 ]	16,448
ELU-15	[ -1, 64, 2, 1992 ]	0
<b>BatchNorm2d-16</b>	[ -1, 64, 2, 1992 ]	128
MaxPool2d-17	[ -1, 64, 2, 332 ]	0
Conv2d-18	[ -1, 64, 2, 325 ]	32,832
ELU-19	[ -1, 64, 2, 325 ]	0
<b>BatchNorm2d-20</b>	[ -1, 64, 2, 325 ]	128
Conv2d-21	[ -1, 128, 2, 322 ]	32,896
ELU-22	[ -1, 128, 2, 322 ]	0
<b>BatchNorm2d-23</b>	[ -1, 128, 2, 322 ]	256
Conv2d-24	[ -1, 128, 2, 319 ]	65,664
ELU-25	[ -1, 128, 2, 319 ]	0
<b>BatchNorm2d-26</b>	[ -1, 128, 2, 319 ]	256
MaxPool2d-27	[ -1, 128, 2, 79 ]	0
Flatten-28	[ -1, 20224 ]	0
Linear-29	[ -1, 64 ]	1,294,400
ELU-30	[ -1, 64 ]	0
<b>Dropout-31</b>	[ -1, 64 ]	0
Linear-32	[ -1, 2 ]	130



# 引力波信号搜寻：卷积神经网络

CNN

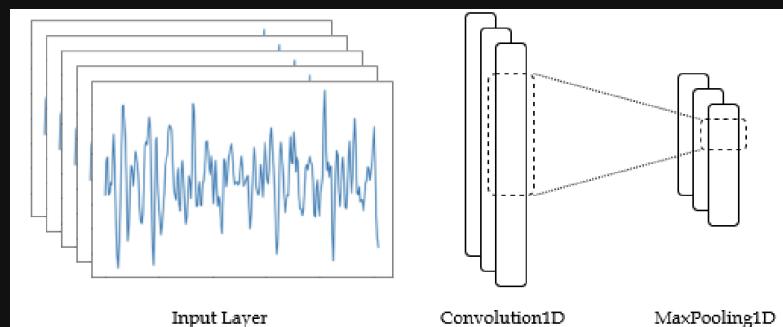
- 目标：复现这个学习任务，搭建一个 Baseline
- 任务：判断一段含有引力波信号的时序数据 (H1/L1)，是否含有引力波信号（二分类问题）

## Max Pooling

[ 6, 8, 2, 16353 ]

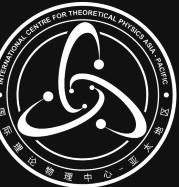


[ 6, 8, 2, 2044 ]



```
MaxPool2d(kernel_size=[ 1, 8 ],
           stride=[ 1, 8 ],
           padding=0)
```

Layer (type)	Output Shape	Param #
Conv2d-1	[ -1, 8, 2, 16353 ]	264
ELU-2	[ -1, 8, 2, 16353 ]	0
BatchNorm2d-3	[ -1, 8, 2, 16353 ]	16
<b>MaxPool2d-4</b>	[ -1, 8, 2, 2044 ]	0
Conv2d-5	[ -1, 16, 2, 2029 ]	2,064
ELU-6	[ -1, 16, 2, 2029 ]	0
BatchNorm2d-7	[ -1, 16, 2, 2029 ]	32
Conv2d-8	[ -1, 16, 2, 2014 ]	4,112
ELU-9	[ -1, 16, 2, 2014 ]	0
BatchNorm2d-10	[ -1, 16, 2, 2014 ]	32
Conv2d-11	[ -1, 32, 2, 1999 ]	8,224
ELU-12	[ -1, 32, 2, 1999 ]	0
BatchNorm2d-13	[ -1, 32, 2, 1999 ]	64
Conv2d-14	[ -1, 64, 2, 1992 ]	16,448
ELU-15	[ -1, 64, 2, 1992 ]	0
BatchNorm2d-16	[ -1, 64, 2, 1992 ]	128
MaxPool2d-17	[ -1, 64, 2, 332 ]	0
Conv2d-18	[ -1, 64, 2, 325 ]	32,832
ELU-19	[ -1, 64, 2, 325 ]	0
BatchNorm2d-20	[ -1, 64, 2, 325 ]	128
Conv2d-21	[ -1, 128, 2, 322 ]	32,896
ELU-22	[ -1, 128, 2, 322 ]	0
BatchNorm2d-23	[ -1, 128, 2, 322 ]	256
Conv2d-24	[ -1, 128, 2, 319 ]	65,664
ELU-25	[ -1, 128, 2, 319 ]	0
BatchNorm2d-26	[ -1, 128, 2, 319 ]	256
MaxPool2d-27	[ -1, 128, 2, 79 ]	0
Flatten-28	[ -1, 20224 ]	0
Linear-29	[ -1, 64 ]	1,294,400
ELU-30	[ -1, 64 ]	0
Dropout-31	[ -1, 64 ]	0
Linear-32	[ -1, 2 ]	130



# 引力波信号搜寻：卷积神经网络

CNN

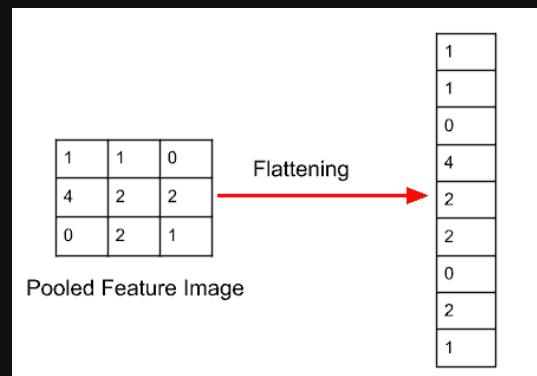
- 目标：复现这个学习任务，搭建一个 Baseline
- 任务：判断一段含有引力波信号的时序数据 (H1/L1)，是否含有引力波信号（二分类问题）

Flatten

[ 6, 128, 2, 79 ]



[ 6, 20224 ]



Linear

[ 20224, 64 ]



[ 6, 20224 ]

[ 6, 64 ]

[ 64, 2 ]



[ 6, 64 ]

[ 6, 2 ]

Layer (type)	Output Shape	Param #
Conv2d-1	[ -1, 8, 2, 16353 ]	264
ELU-2	[ -1, 8, 2, 16353 ]	0
BatchNorm2d-3	[ -1, 8, 2, 16353 ]	16
MaxPool2d-4	[ -1, 8, 2, 2044 ]	0
Conv2d-5	[ -1, 16, 2, 2029 ]	2,064
ELU-6	[ -1, 16, 2, 2029 ]	0
BatchNorm2d-7	[ -1, 16, 2, 2029 ]	32
Conv2d-8	[ -1, 16, 2, 2014 ]	4,112
ELU-9	[ -1, 16, 2, 2014 ]	0
BatchNorm2d-10	[ -1, 16, 2, 2014 ]	32
Conv2d-11	[ -1, 32, 2, 1999 ]	8,224
ELU-12	[ -1, 32, 2, 1999 ]	0
BatchNorm2d-13	[ -1, 32, 2, 1999 ]	64
Conv2d-14	[ -1, 64, 2, 1992 ]	16,448
ELU-15	[ -1, 64, 2, 1992 ]	0
BatchNorm2d-16	[ -1, 64, 2, 1992 ]	128
MaxPool2d-17	[ -1, 64, 2, 332 ]	0
Conv2d-18	[ -1, 64, 2, 325 ]	32,832
ELU-19	[ -1, 64, 2, 325 ]	0
BatchNorm2d-20	[ -1, 64, 2, 325 ]	128
Conv2d-21	[ -1, 128, 2, 322 ]	32,896
ELU-22	[ -1, 128, 2, 322 ]	0
BatchNorm2d-23	[ -1, 128, 2, 322 ]	256
Conv2d-24	[ -1, 128, 2, 319 ]	65,664
ELU-25	[ -1, 128, 2, 319 ]	0
BatchNorm2d-26	[ -1, 128, 2, 319 ]	256
MaxPool2d-27	[ -1, 128, 2, 79 ]	0
Flatten-28	[ -1, 20224 ]	0
Linear-29	[ -1, 64 ]	1,294,400
ELU-30	[ -1, 64 ]	0
Dropout-31	[ -1, 64 ]	0
Linear-32	[ -1, 2 ]	130

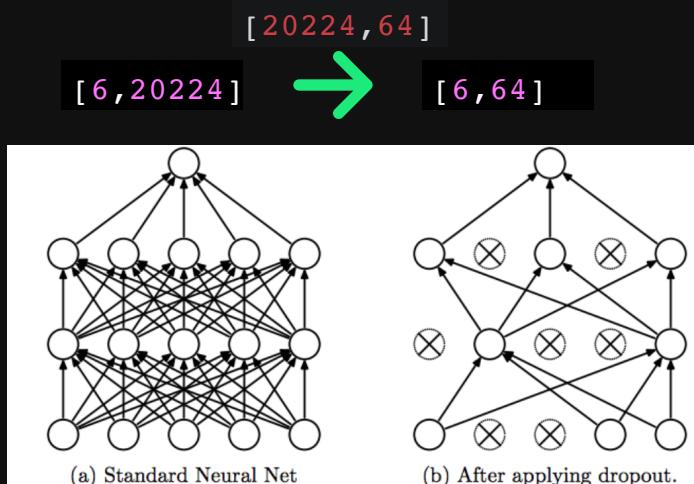


# 引力波信号搜寻：卷积神经网络

CNN

- 目标：复现这个学习任务，搭建一个 Baseline
- 任务：判断一段含有引力波信号的时序数据 (H1/L1)，是否含有引力波信号（二分类问题）

## Dropout



### Dropout ( $p=0.5$ )

- Dropout 随机失活：以一定概率  $p$  要求中间层的部分神经元临时删除。
- 大部分实验表明，Dropout 可以有效地防止过拟合问题的发生。
- 机制：相当于对大量不同网络子模型的性能做了近似平均，不同的子网络会以不同的方式过度拟合，模型集成后就会减轻过度拟合的效果。

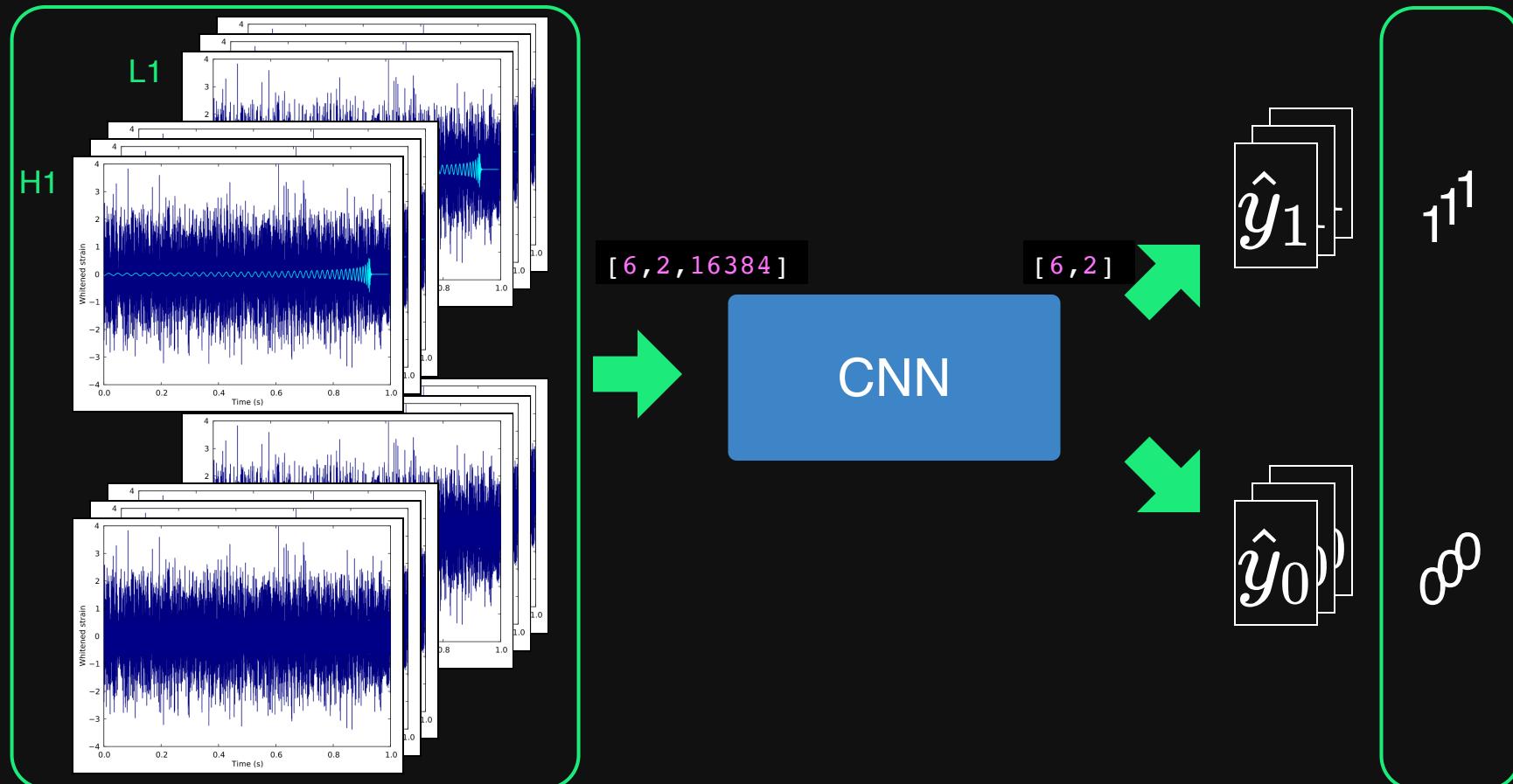
Layer (type)	Output Shape	Param #
Conv2d-1	[ -1, 8, 2, 16353 ]	264
ELU-2	[ -1, 8, 2, 16353 ]	0
BatchNorm2d-3	[ -1, 8, 2, 16353 ]	16
MaxPool2d-4	[ -1, 8, 2, 2044 ]	0
Conv2d-5	[ -1, 16, 2, 2029 ]	2,064
ELU-6	[ -1, 16, 2, 2029 ]	0
BatchNorm2d-7	[ -1, 16, 2, 2029 ]	32
Conv2d-8	[ -1, 16, 2, 2014 ]	4,112
ELU-9	[ -1, 16, 2, 2014 ]	0
BatchNorm2d-10	[ -1, 16, 2, 2014 ]	32
Conv2d-11	[ -1, 32, 2, 1999 ]	8,224
ELU-12	[ -1, 32, 2, 1999 ]	0
BatchNorm2d-13	[ -1, 32, 2, 1999 ]	64
Conv2d-14	[ -1, 64, 2, 1992 ]	16,448
ELU-15	[ -1, 64, 2, 1992 ]	0
BatchNorm2d-16	[ -1, 64, 2, 1992 ]	128
MaxPool2d-17	[ -1, 64, 2, 332 ]	0
Conv2d-18	[ -1, 64, 2, 325 ]	32,832
ELU-19	[ -1, 64, 2, 325 ]	0
BatchNorm2d-20	[ -1, 64, 2, 325 ]	128
Conv2d-21	[ -1, 128, 2, 322 ]	32,896
ELU-22	[ -1, 128, 2, 322 ]	0
BatchNorm2d-23	[ -1, 128, 2, 322 ]	256
Conv2d-24	[ -1, 128, 2, 319 ]	65,664
ELU-25	[ -1, 128, 2, 319 ]	0
BatchNorm2d-26	[ -1, 128, 2, 319 ]	256
MaxPool2d-27	[ -1, 128, 2, 79 ]	0
Flatten-28	[ -1, 20224 ]	0
Linear-29	[ -1, 64 ]	1,294,400
ELU-30	[ -1, 64 ]	0
Dropout-31	[ -1, 64 ]	0
Linear-32	[ -1, 2 ]	130



# 引力波信号搜寻：卷积神经网络

- 目标：复现这个学习任务，搭建一个 Baseline
- 任务：判断一段含有引力波信号的时序数据 (H1/L1)，是否含有引力波信号（二分类问题）

训练的过程 (Train) :





# 引力波信号搜寻：卷积神经网络

**Dataset** containing  $N$  examples sampling from true but unknown data generating distribution  $p_{\text{data}}(\mathbf{x})$ :

$$\mathbf{X} = \{\mathbf{x}^{(i)} \mid i = 1, 2, \dots, N\}, \quad \mathbf{x} \in \mathbb{R}^n$$

with corresponding **ground-truth labels**:

$$\mathbf{Y} = \{\mathbf{y}^{(i)} \mid i = 1, 2, \dots, N\}, \quad \mathbf{y} \in \{0, 1\}$$

Machine learning **model** is nothing but a **map  $f$**  from samples to labels:

$$\begin{aligned}\mathbf{X} &\longrightarrow \hat{\mathbf{Y}} = f(\mathbf{X}; \Theta) \\ \mathbf{x} &\longmapsto \hat{\mathbf{y}} = f(\mathbf{x}; \Theta) \in \mathbb{R}\end{aligned}$$

where  $\Theta$  is parameters of the model and the outputs are **predicted labels**:

$$\hat{\mathbf{Y}} = \{\hat{\mathbf{y}}^{(i)} \mid i = 1, 2, \dots, N\}, \quad 0 \leq \hat{\mathbf{y}} \leq 1$$

described by  $p_{\text{model}}(\mathbf{y}|\mathbf{x}; \Theta)$ , a parametric family of probability distributions over the same space indexed by  $\Theta$ .



# 引力波信号搜寻：卷积神经网络

**Dataset** containing  $N$  examples sampling from true but unknown data generating distribution  $p_{\text{data}}(\mathbf{x})$ :

$$\mathbf{X} = \{\mathbf{x}^{(i)} \mid i = 1, 2, \dots, N\}, \quad \mathbf{x} \in \mathbb{R}^n$$

with corresponding **ground-truth labels**:

$$\mathbf{Y} = \{\mathbf{y}^{(i)} \mid i = 1, 2, \dots, N\}, \quad \mathbf{y} \in \{0, 1\}$$

Machine learning **model** is nothing but a **map  $f$**  from samples to labels:

$$\mathbf{X} \rightarrow \hat{\mathbf{Y}} = f(\mathbf{X}; \Theta)$$

$$\mathbf{x} \mapsto \hat{\mathbf{y}} = f(\mathbf{x}; \Theta) \in \mathbb{R}$$

where  $\Theta$  is parameters of the model and the outputs are **predicted labels**:

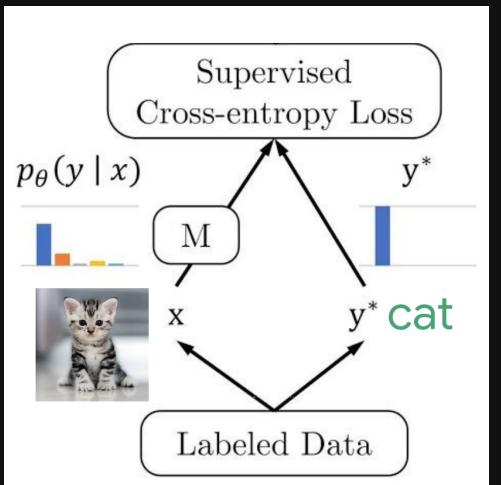
$$\hat{\mathbf{Y}} = \{\hat{\mathbf{y}}^{(i)} \mid i = 1, 2, \dots, N\}, \quad 0 \leq \hat{\mathbf{y}} \leq 1$$

described by  $p_{\text{model}}(\mathbf{y}|\mathbf{x}; \Theta)$ , a parametric family of probability distributions over the same space indexed by  $\Theta$ .

## Objective:

- For each sample,  $\hat{\mathbf{y}}^{(i)} \rightarrow \mathbf{y}^{(i)}$
- Find the best  $\Theta$  that

$$p_{\text{model}}(\mathbf{x}; \Theta) \rightarrow p_{\text{data}}(\mathbf{x})$$





# 引力波信号搜寻：卷积神经网络

For classification problem, we always use **maximum likelihood** estimator for  $\Theta$

$$\begin{aligned}\boldsymbol{\theta}_{\text{ML}} &= \arg \max_{\theta} p_{\text{model}}(\hat{\mathbf{Y}} | \mathbf{X}; \boldsymbol{\theta}) \\ &= \arg \max_{\theta} \prod_{i=1}^N p_{\text{model}}(\hat{\mathbf{y}}^{(i)} | \mathbf{x}^{(i)}; \boldsymbol{\theta}) \\ &= \arg \max_{\theta} \sum_{i=1}^N \log p_{\text{model}}(\hat{\mathbf{y}}^{(i)} | \mathbf{x}^{(i)}; \boldsymbol{\theta}) \\ &= \arg \max_{\theta} \mathbb{E}_{\mathbf{x}, \mathbf{y} \sim \hat{p}_{\text{data}}} \log p_{\text{model}}(\mathbf{y} | \mathbf{x}; \boldsymbol{\theta})\end{aligned}$$

to construct **cost function  $J(\boldsymbol{\theta})$**  (also called loss func. or error func.)

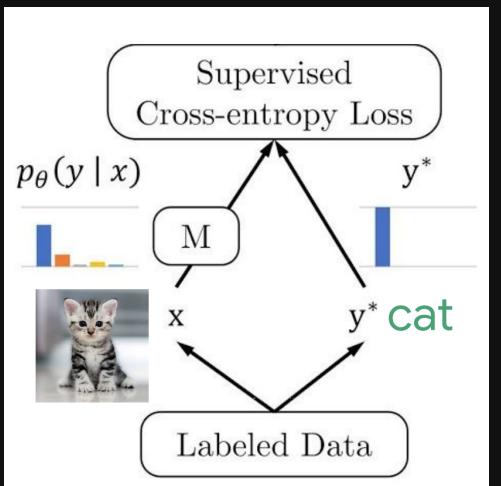
$$J(\boldsymbol{\theta}) = -\mathbb{E}_{\mathbf{x}, \mathbf{y} \sim \hat{p}_{\text{data}}} \log p_{\text{model}}(\mathbf{y} | \mathbf{x}; \boldsymbol{\theta})$$

$$\boldsymbol{\theta}_{\text{ML}} = \arg \min_{\theta} J(\boldsymbol{\theta})$$

## Objective:

- For each sample,  $\hat{\mathbf{y}}^{(i)} \rightarrow \mathbf{y}^{(i)}$
- Find the best  $\Theta$  that

$$p_{\text{model}}(\mathbf{x}; \boldsymbol{\Theta}) \rightarrow p_{\text{data}}(\mathbf{x})$$



- Minimizing the KL divergence corresponds exactly to minimizing the **cross-entropy** (negative log-likelihood of a Bernoulli/Softmax distribution) between the distributions.



# 引力波信号搜寻：卷积神经网络

在信息论中，可以通过某概率分布函数  $p(x), x \in X$  作为变量，  
定义一个关于  $p(x)$  的单调函数  $h(x)$ ，称其为概率分布  $p(x)$  的  
**信息量(measure of information)**:  $h(x) \equiv -\log p(x)$

定义所有信息量的期望为随机变量  $x$  的 **熵(entropy)**:

$$H(x) \equiv -\sum_x p(x) \log p(x) = -\mathbb{E}_{x \sim p} \log p(x)$$

若同一个随机变量  $x$  有两个独立的概率分布  $p(x)$  和  $q(x)$ ，则可以  
定义这两个分布的相对熵 (relative entropy)，也常称为 **KL 散度**  
(Kullback-Leibler divergence)，来衡量两个分布之间的差异:

$$\begin{aligned} \text{KL}(p\|q) &\equiv -\sum_x p(x) \log q(x) - \left( -\sum_x p(x) \log p(x) \right) \\ &= -\sum_x p(x) \log \left\{ \frac{q(x)}{p(x)} \right\} \\ &= -H_p(x) + H_{(p,q)}(x) \end{aligned}$$

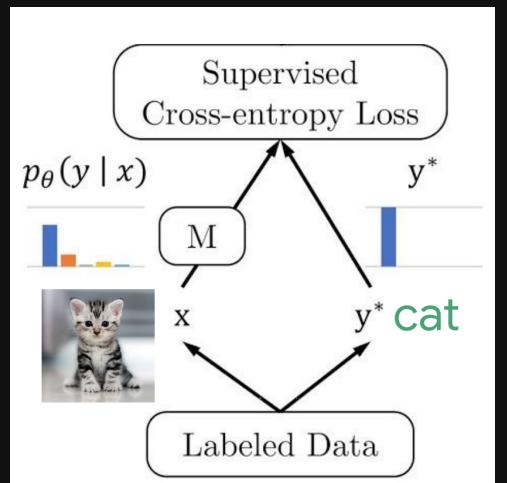
可见 KL 越小，表示  $p(x)$  和  $q(x)$  两个分布越接近。上式中，我们  
已经定义了**交叉熵(cross entropy)** 为

$$H_{(p,q)}(x) \equiv -\sum_x p(x) \log q(x) = -\mathbb{E}_{x \sim p} \log q(x)$$

## Objective:

- For each sample,  $\hat{\mathbf{y}}^{(i)} \rightarrow \mathbf{y}^{(i)}$
- Find the best  $\Theta$  that

$$p_{\text{model}}(\mathbf{x}; \Theta) \rightarrow p_{\text{data}}(\mathbf{x})$$





# 引力波信号搜寻：卷积神经网络

当对应到机器学习中最大似然估计方法时，训练集上的经验分布  $\hat{p}_{\text{data}}$  和模型分布之间的差异程度可以用 KL 散度度量为：

$$\text{KL}(\hat{p}_{\text{data}} \| p_{\text{model}}) = -H_{p_{\text{data}}}(\mathbf{X}) - \mathbb{E}_{\mathbf{x} \sim \hat{p}_{\text{data}}} \log p_{\text{model}}(\mathbf{x}; \theta)$$

由上式可知，等号右边第一项仅涉及数据的生成过程，和机器学习模型无关。这意味着当我们训练机器学习模型最小化 KL 散度时，我们只需要等价优化地最小化等号右边的第二项，即有

$$\arg \min_{\theta} \text{KL}(\hat{p}_{\text{data}} \| p_{\text{model}}) = -\arg \min_{\theta} \mathbb{E}_{\mathbf{x} \sim \hat{p}_{\text{data}}} \log p_{\text{model}}(\mathbf{x}; \theta)$$

由此可知，对于任何一个由负对数似然组成的代价函数都是定义在训练集上的经验分布和定义在模型上的概率分布之间的交叉熵。

**Recall:**

$$\text{KL}(p \| q) = -H_p(x) + H_{(p,q)}(x)$$

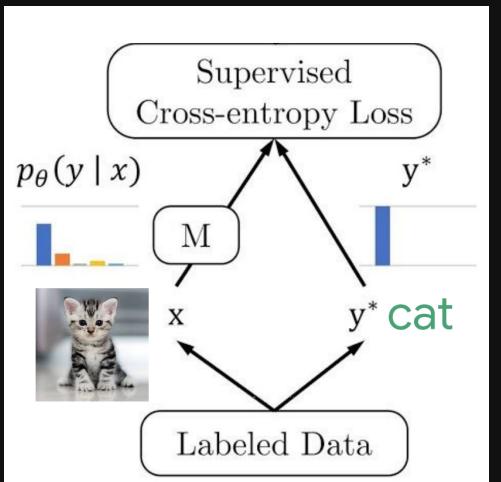
$$H_{(p,q)}(x) \equiv -\mathbb{E}_{x \sim p} \log q(x)$$

$$J(\theta) = -\mathbb{E}_{\mathbf{x}, \mathbf{y} \sim \hat{p}_{\text{data}}} \log p_{\text{model}}(\mathbf{y} \mid \mathbf{x}; \theta)$$

## Objective:

- For each sample,  $\hat{\mathbf{y}}^{(i)} \rightarrow \mathbf{y}^{(i)}$
- Find the best  $\Theta$  that

$$p_{\text{model}}(\mathbf{x}; \Theta) \rightarrow p_{\text{data}}(\mathbf{x})$$

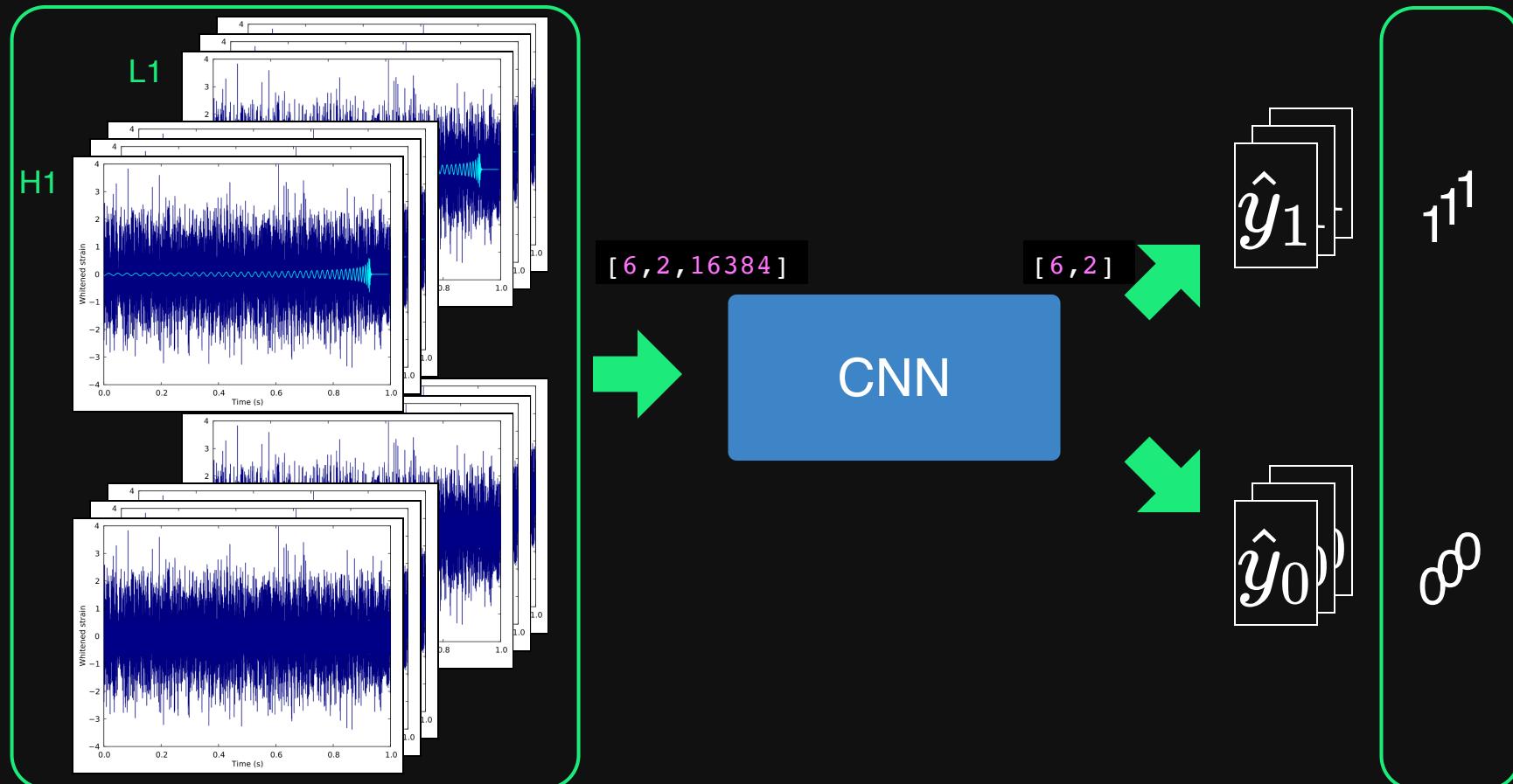




# 引力波信号搜寻：卷积神经网络

- 目标：复现这个学习任务，搭建一个 Baseline
- 任务：判断一段含有引力波信号的时序数据 (H1/L1)，是否含有引力波信号（二分类问题）

训练的过程 (Train) :

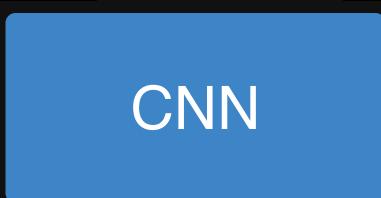
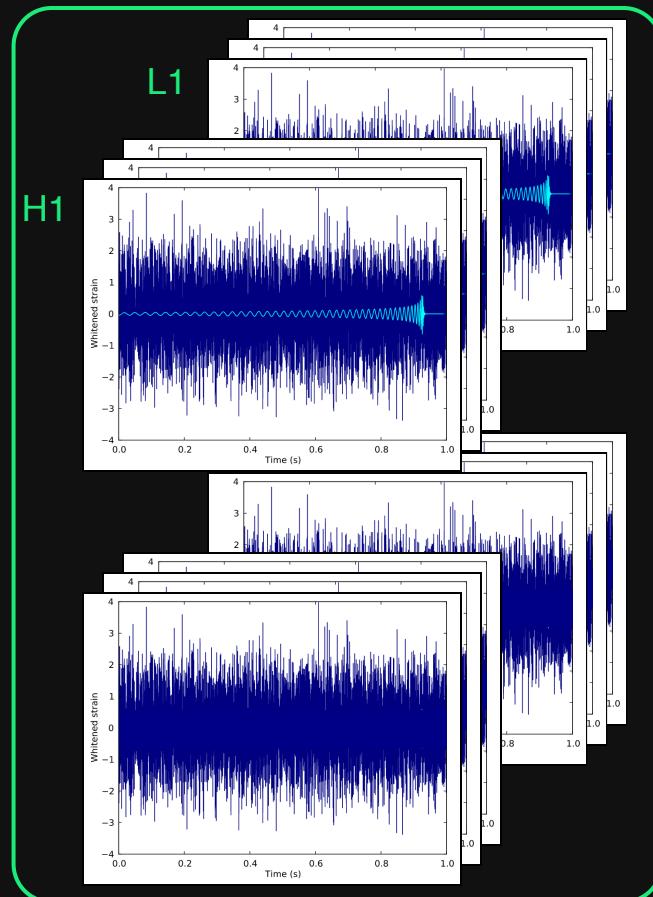




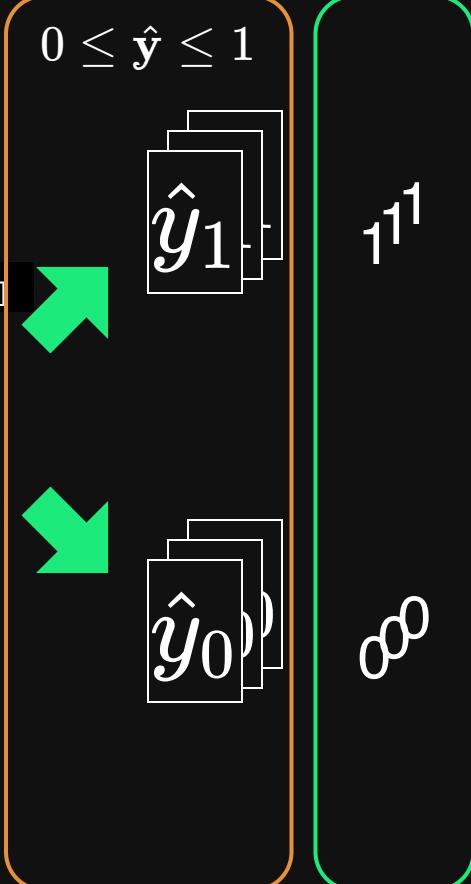
# 引力波信号搜寻：卷积神经网络

- 目标：复现这个学习任务，搭建一个 Baseline
- 任务：判断一段含有引力波信号的时序数据 (H1/L1)，是否含有引力波信号（二分类问题）

训练的过程 (Train) :



[ 6 , 2 ]





# 引力波信号搜寻：卷积神经网络

基于分类问题的网络模型里，我们都是用 softmax 函数作为模型最终输出数据时的非线性计算单元，其函数形式为：

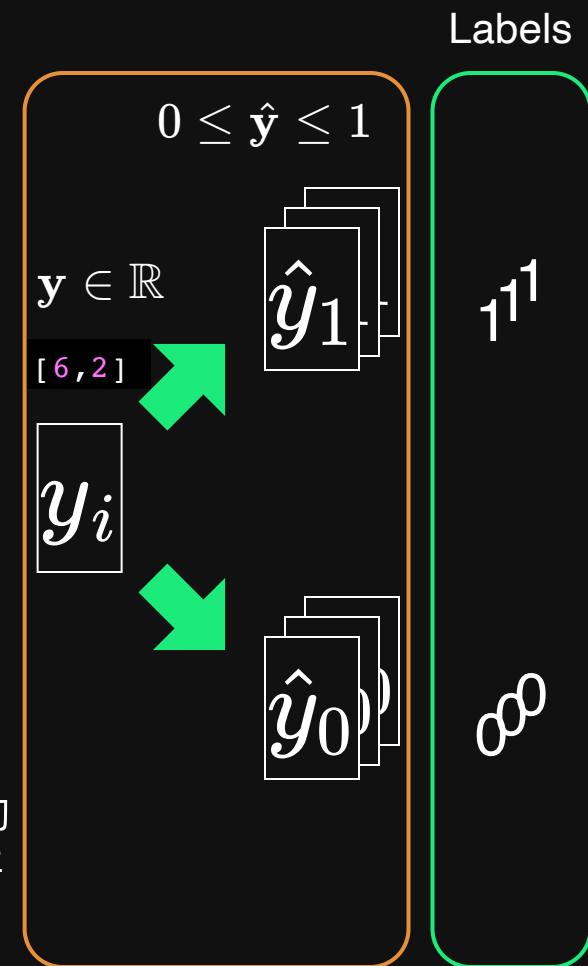
$$\text{softmax}(\mathbf{y})_i = \frac{\exp(y_i)}{\sum_j \exp(y_j)} = \hat{y}_i$$

由于该函数的输出元素都介于 0 和 1 之间，且向量之和为 1，这使得其可以作为一个有效的“概率分布”  $p_{\text{model}}(y = k | \mathbf{x}^{(i)})$ 。

由此，我们使用最大化条件对数似然输出某样本的目标分类  $y$  时，即等价于对下式最大化，

$$\begin{aligned} \log p_{\text{model}}(\hat{y}_{k=\text{label}} | \mathbf{y}) &= \log \text{softmax}(\mathbf{y})_k \\ &= y_k - \log \sum_j^N \exp(y_j) \end{aligned}$$

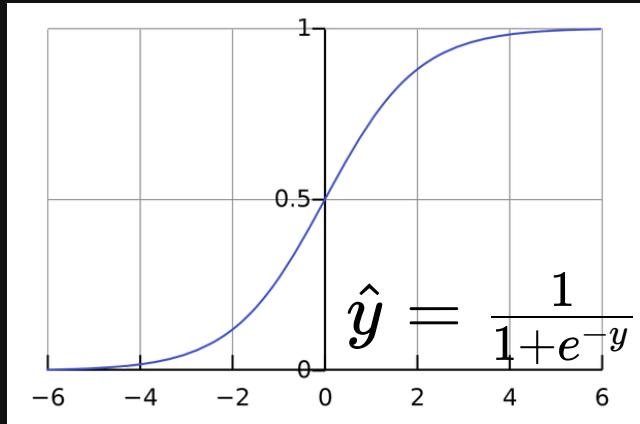
上式中的第一项表示模型的直接输出结果  $\hat{y}_k$ ，对优化目标有着直接的贡献。在最大化对数似然时，当然是第一项越大越好，而第二项是鼓励越小越好。根据  $\log \sum_j^N \exp(\hat{y}_j) \sim \max_j \hat{y}_j$  近似关系，可以发现负对数似然代价函数总是强烈的想要惩罚最活跃的不正确预测。如果某样本的正确 label 对应了 softmax 的最大输入，那么  $-y_k$  项和  $\log \sum_j^N \exp(y_j) \sim \max_j y_j = y_k$  项将大致抵消。





# 引力波信号搜寻：卷积神经网络

对于二值型的分类学习任务，softmax 函数会退化到 sigmoid 函数：

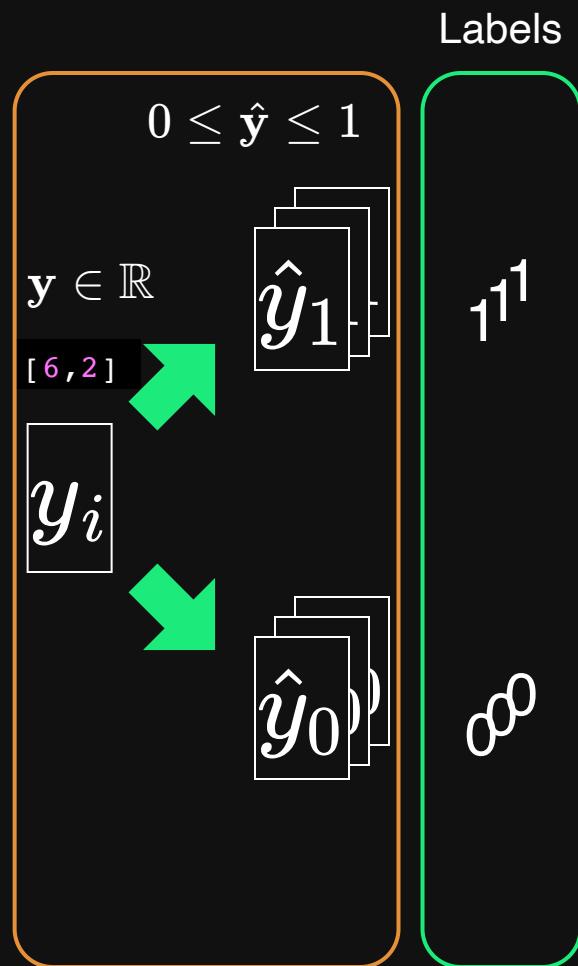


由此可以证明，cost function 可以表示为

$$J(\theta) = -\sum_{i=1}^N y^{(i)} \log \hat{y}^{(i)} + (1 - y^{(i)}) \log(1 - \hat{y}^{(i)})$$

$N = 2$  :

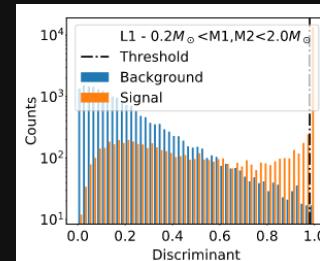
$$J(\theta) = \begin{cases} -\log(1 - y) & \text{if label} = 0 \\ -\log(y) & \text{if label} = 1 \end{cases}$$





# 引力波信号搜寻：卷积神经网络

- 机器学习算法输出的信号存在与否的“概率” (p-value)，不能作为探测统计量，其无法对疑似信号的统计显著性进行排序
- "*The negative log-likelihood cost function always strongly penalizes the most active incorrect prediction. And the correctly classified examples will contribute little to the overall training cost.*"  
—— I. Goodfellow, Y. Bengio, A. Courville. Deep Learning. 2016.
- 因此，单纯用 Softmax 函数代表 p-value 表示引力波信号是否存在，会造成在正负样本上 p-value 概率密度分布的不单调性。



Menéndez-Vázquez  
A, et al. PRD 2021

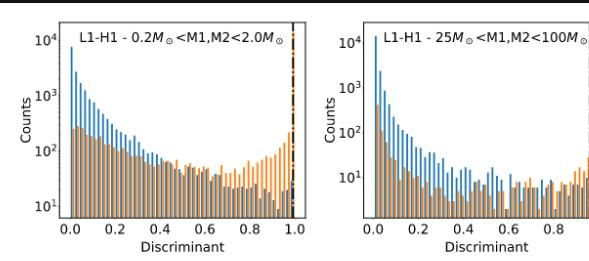


FIG. 5. CNN discriminating outputs corresponding to the H1-L1 case for background and signal images in the case of (left) low mass and (right) high mass ranges. The dashed-dotted lines indicate the threshold used to identify the signal events.

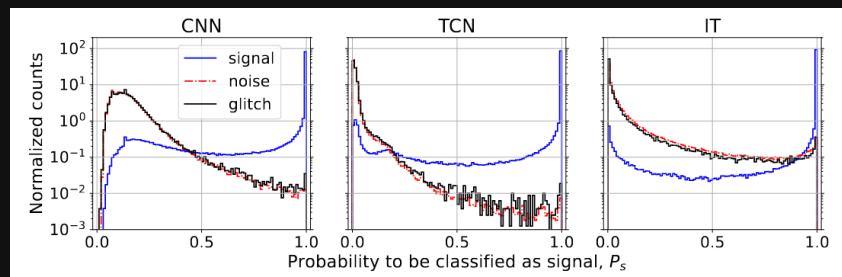


Figure 2. Distributions of  $P_s$  (the class membership probability assigned to the signal class), conditioned on the class of the input segment from the testing set: signal (blue), noise (dot-dashed red), or glitch (black). These distributions were computed for the CNN (left), TCN (middle), and IT (right) architectures. The histograms are normalized to have a unit sum. The classifiers do not distinguish between samples from the noise and glitch classes, thus resulting in practically identical probability distributions (see Sec. 4 for a discussion on this point).

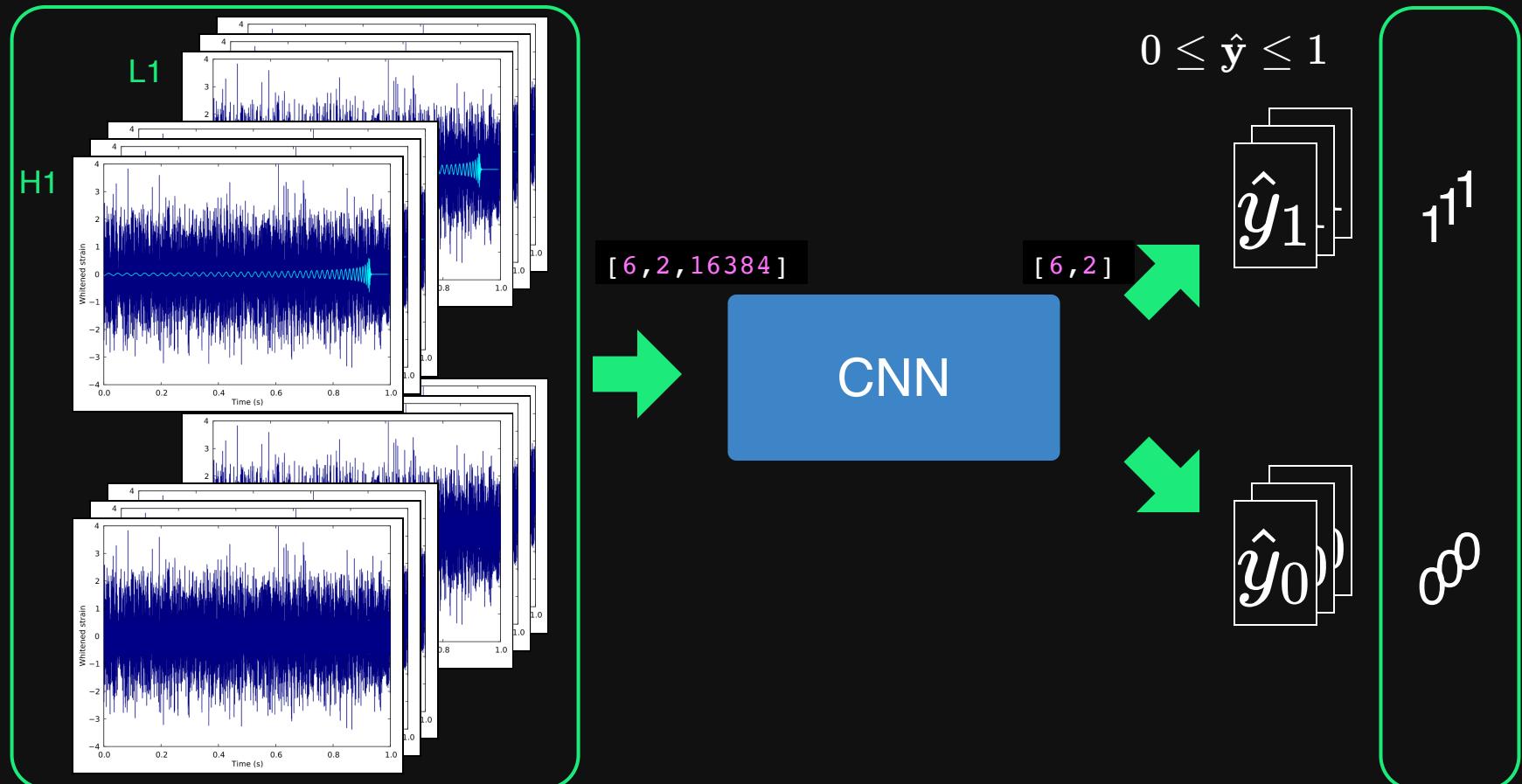
arXiv: 2307.09268



# 引力波信号搜寻：卷积神经网络

- 目标：复现这个学习任务，搭建一个 Baseline
- 任务：判断一段含有引力波信号的时序数据 (H1/L1)，是否含有引力波信号（二分类问题）

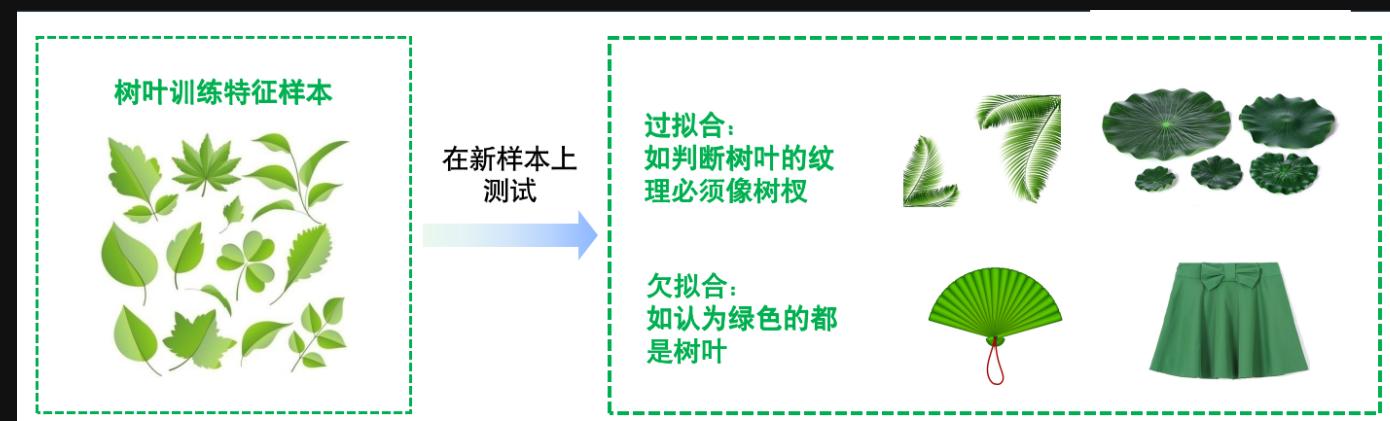
训练的过程 (Train) :





# 引力波信号搜寻：卷积神经网络

- 从假设空间中 学习/选择 最优模型的准则?
- “最合适”的模型：机器学习从有限的观测数据中学习出规律 [训练误差 (training error)], 并将总结的规律推广应用到未观测样本上 → 追求泛化性能 [泛化误差 (generalization error)]。
- 所以，我们认定机器学习算法效果是否很好，两个学习目标：
  1. 降低训练误差。
  2. 缩小训练误差和测试误差的差距。
- 这两个目标分别对应了机器学习的两个重要挑战：
  - 欠拟合 (underfitting)：模型不能在训练集上获得足够低的误差。
  - 过拟合 (overfitting)：训练误差与测试误差之间的差距过大。

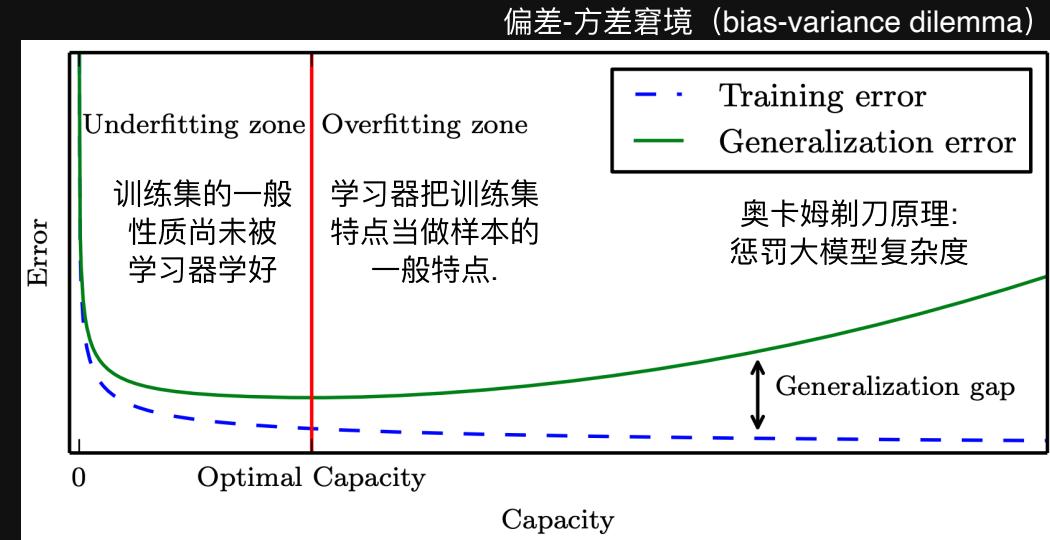




# 引力波信号搜寻：卷积神经网络

- 从假设空间中 学习/选择 最优模型的准则?
- “最合适”的模型：机器学习从有限的观测数据中学习出规律 [训练误差 (training error)], 并将总结的规律推广应用到未观测样本上 → 追求泛化性能 [泛化误差 (generalization error)]。
- 所以，我们认定机器学习算法效果是否很好，两个学习目标：
  1. 降低训练误差。
  2. 缩小训练误差和测试误差的差距。
- 这两个目标分别对应了机器学习的两个重要挑战：
  - 欠拟合 (underfitting)：模型不能在训练集上获得足够低的误差。
  - 过拟合 (overfitting)：训练误差与测试误差之间的差距过大。
- 泛化性能 是由
  1. 学习算法的能力、
  2. 数据的充分性以及
  3. 学习任务本身的难度共同决定。

模型的容量 Capacity 是指其拟合各种函数的能力，一般也可以代表模型的复杂程度。 (“计算学习理论”)





# 引力波信号搜寻：卷积神经网络

- 模型调优时，如何判断是否过拟合？



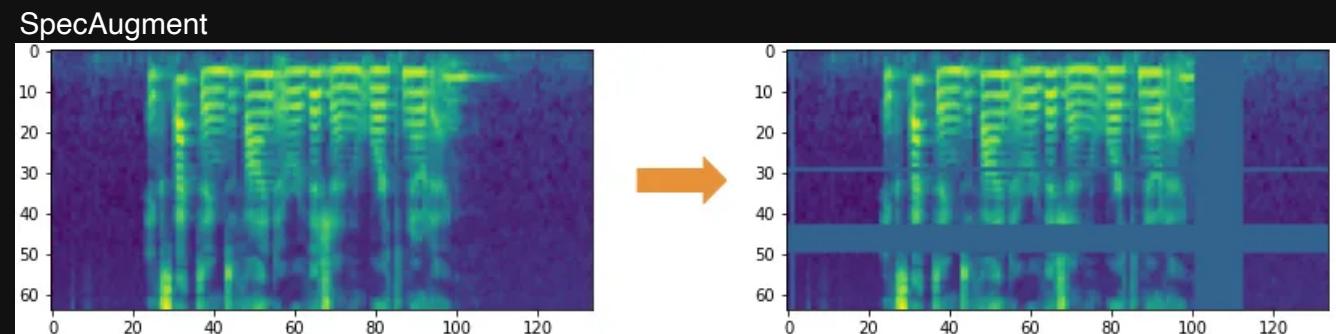
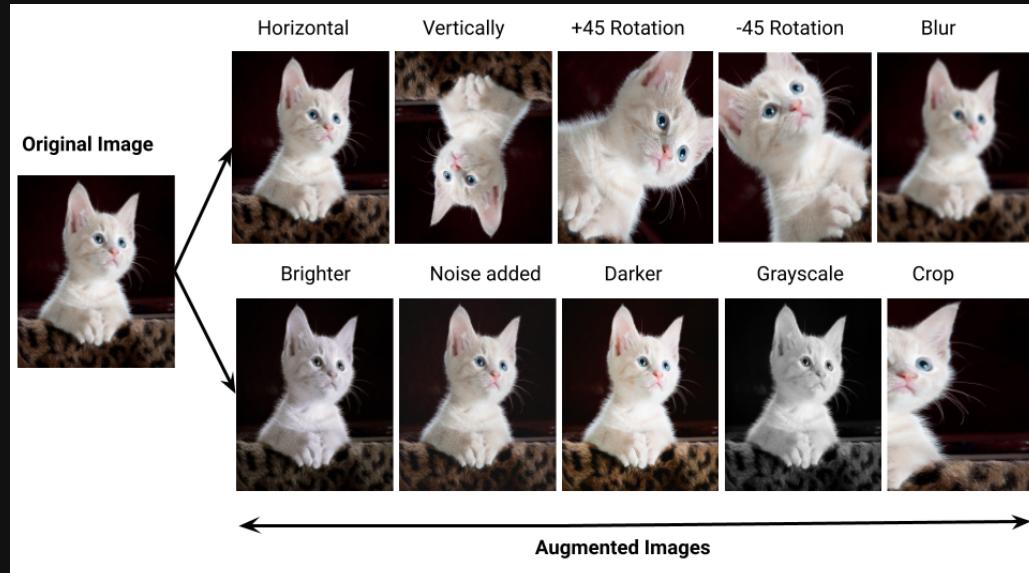
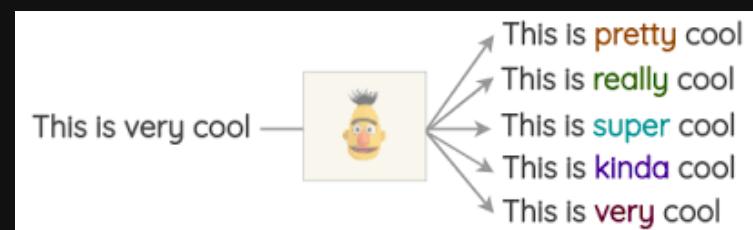
参数调得好不好，往往对最终性能有关键影响。

	欠拟合	恰到好处！	过拟合
特点	<ul style="list-style-type: none"> <li>高训练误差</li> <li>训练误差与测试误差接近</li> <li>高偏差</li> </ul>	训练误差略微低于测试误差	<ul style="list-style-type: none"> <li>低训练误差</li> <li>训练误差远低于测试误差</li> <li>高方差</li> </ul>
案例			
深度学习建模过程			
可选的解决办法	<ul style="list-style-type: none"> <li>模型复杂化</li> <li>引入更多特征</li> <li>延长训练时间</li> </ul>		<ul style="list-style-type: none"> <li>模型“惩罚”（正则化）</li> <li>获取更多数据</li> </ul> <p>early stop、dropout</p>



# 引力波信号搜寻：卷积神经网络

- 模型调优时，如何判断是否过拟合？
- **数据增广** (训练集越大，越不容易过拟合)
  - 计算机视觉：图像旋转、缩放、剪切
  - 自然语言处理：同义词替换
  - 语音识别：添加随机噪声





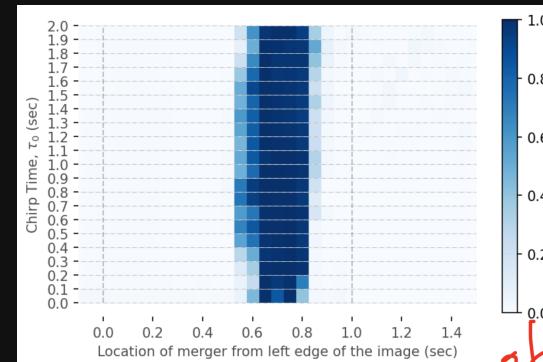
# 引力波信号搜寻：卷积神经网络

- 模型调优时，如何判断是否过拟合？
- **数据增广** (训练集越大，越不容易过拟合)
  - 计算机视觉：图像旋转、缩放、剪切
  - 自然语言处理：同义词替换
  - 语音识别：添加随机噪声
  - **引力波数据处理**：

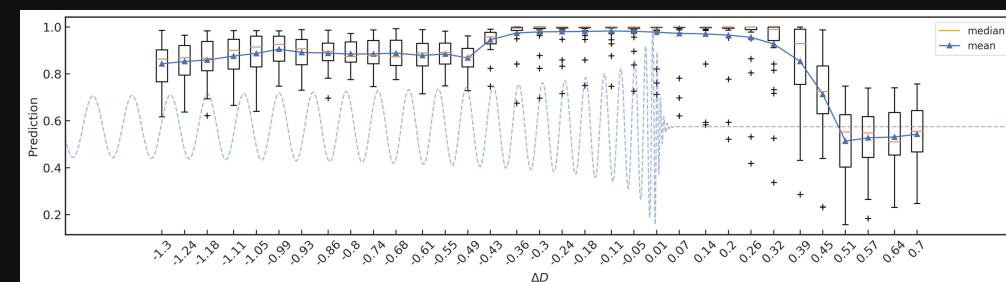
1. such that the cosine of inclination is uniform on the range  $[-1, 1]$ . The waveforms are then randomly placed within the time series such that the peak amplitude of each waveform is randomly positioned within the fractional range  $[0.75, 0.95]$  of the time series.

The waveform amplitude is scaled to achieve a prede...  
arXiv: 1712.06041

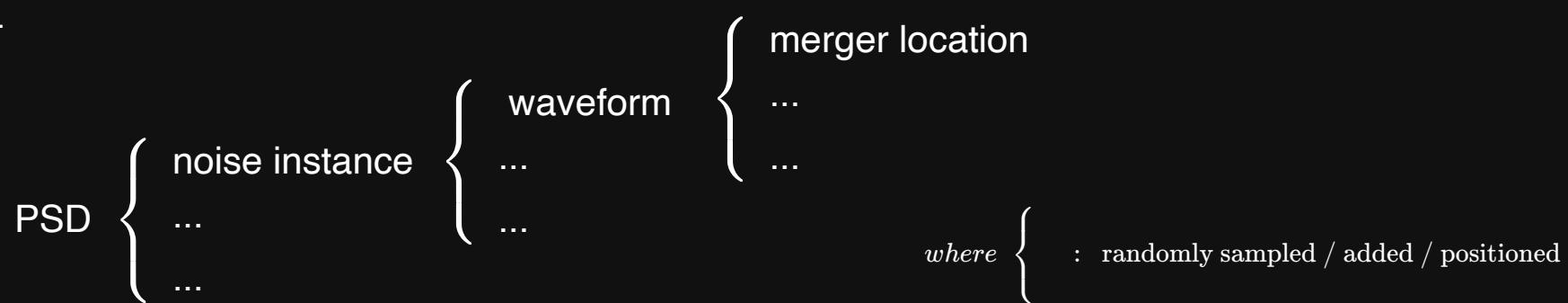
Jadhav et al. 2306.11797



Source: X



2.

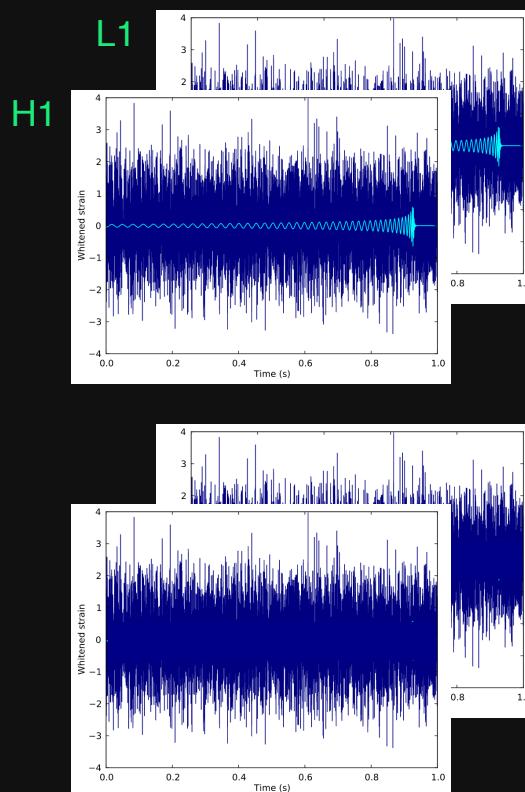




# 引力波信号搜寻：卷积神经网络

- 目标：复现这个学习任务，搭建一个 Baseline
- 任务：判断一段含有引力波信号的时序数据 (H1/L1)，是否含有引力波信号（二分类问题）

期望达到的效果 (Evaluation) :



$$0 \leq \hat{y} \leq 1$$

0.99

CNN

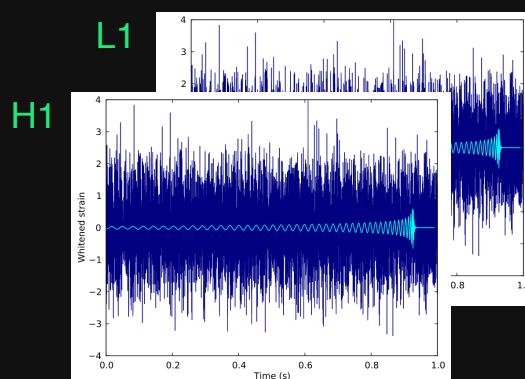
0.05



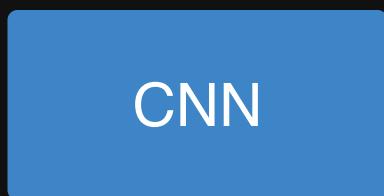
# 引力波信号搜寻：卷积神经网络

- 目标：复现这个学习任务，搭建一个 Baseline
- 任务：判断一段含有引力波信号的时序数据 (H1/L1)，是否含有引力波信号（二分类问题）

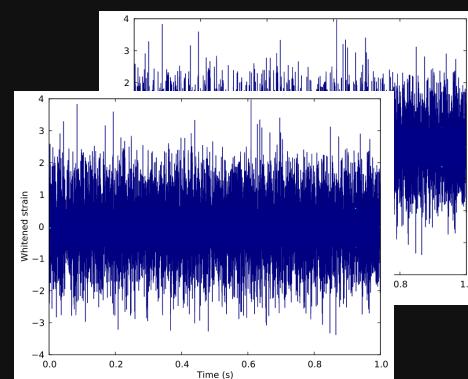
期望达到的效果 (Evaluation) :



$$0 \leq \hat{y} \leq 1$$



0.54 有信号？无信号？



- 分类模型泛化性能评估：混淆矩阵/ROC/AUC



# 引力波信号搜寻：卷积神经网络

- 确定阈值 (threshold) , 才能把分类模型真的确定下来。
  - 0.5 阈值仅代表模型最大梯度更新的极值点，并没有理论上明确的理想分类阈值取法。
  - 不过对于一个训练好的分类模型而言，分类阈值的选取会直接影响模型的泛化考量。



threshold = 0.5, 这里有引力波信号!  
threshold = 0.6, 这里没有引力波信号!





# 引力波信号搜寻：卷积神经网络

- **性能度量** (performance measure):
- 在模型的训练过程中，一般会用**错误率** (error rate) 和**准确率** (accuracy) 来定量衡量分类模型的性能。准确率是指该模型输出分类正确的样本数占样本总数的比例，错误率是其补集。
- 对于引力波信号搜寻，我们更加关心是：
  - “引力波信号中有多少是可以被探测到”
  - “探测到的信号中有多少其实是假的引力波信号”
- 确定**阈值** (threshold)，才能把分类模型真的确定下来。
  - 0.5 阈值仅代表模型最大梯度更新的极值点，并没有理论上明确的理想分类阈值取法。
  - 不过对于一个训练好的分类模型而言，分类阈值的选取会直接影响模型的泛化考量。



threshold = 0.5, 这里有引力波信号!  
threshold = 0.6, 这里没有引力波信号!



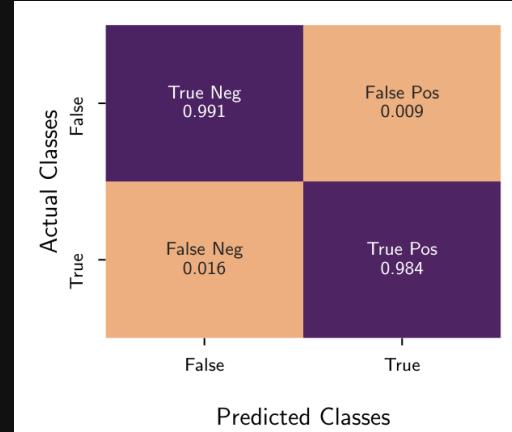


# 引力波信号搜寻：卷积神经网络

arXiv: 2302.00666

- 构建分类结果的混淆矩阵 (confusion matrix)
  - 对于某测试数据集中 + 取定某 threshold 的情况下

真实情况	正例 (预测结果)	反例 (预测结果)
正例	TP (真正例)	FN (假反例)
反例	FP (假正例)	TN (真反例)



- 可以计算出真正例率 (True Positive Rate, TPR) 和假正例率 (False Positive Rate, FPR):

$$\text{TPR} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

$$\text{FPR} = \frac{\text{TP}}{\text{TN} + \text{FP}}$$

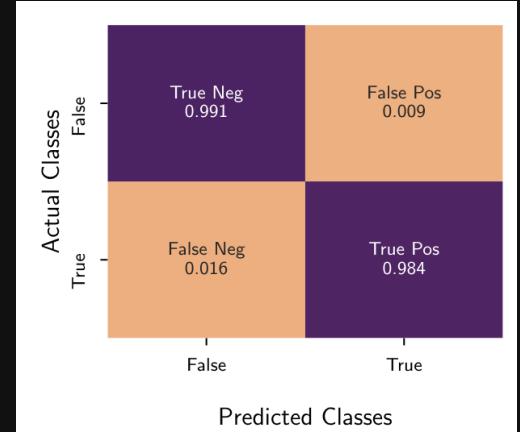


# 引力波信号搜寻：卷积神经网络

arXiv: 2302.00666

- 构建分类结果的混淆矩阵 (confusion matrix)
  - 对于某测试数据集中 + 取定某 threshold 的情况下

真实情况	正例 (预测结果)	反例 (预测结果)
正例	TP (真正例)	FN (假反例)
反例	FP (假正例)	TN (真反例)



- 可以计算出真正例率 (True Positive Rate, TPR) 和假正例率 (False Positive Rate, FPR):

Predicted condition			Sources: [17][18][19][20][21][22][23][24][25] view · talk · edit		
Total population $= P + N$	Positive (PP)	Negative (PN)	Informedness, bookmaker informedness (BM) $= TPR + TNR - 1$		Prevalence threshold (PT) $= \frac{\sqrt{TPR \times FPR} - FPR}{TPR - FPR}$
Actual condition	Positive (P)	True positive (TP), hit	False negative (FN), type II error, miss, underestimation	True positive rate (TPR), recall, sensitivity (SEN), probability of detection, hit rate, power $= \frac{TP}{P} = 1 - FNR$	False negative rate (FNR), miss rate $= \frac{FN}{P} = 1 - TPR$
	Negative (N)	False positive (FP), type I error, false alarm, overestimation	True negative (TN), correct rejection	False positive rate (FPR), probability of false alarm, fall-out $= \frac{FP}{N} = 1 - TNR$	True negative rate (TNR), specificity (SPC), selectivity $= \frac{TN}{N} = 1 - FPR$
Prevalence $= \frac{P}{P + N}$	Positive predictive value (PPV), precision $= \frac{TP}{PP} = 1 - FDR$	False omission rate (FOR) $= \frac{FN}{PN} = 1 - NPV$	Positive likelihood ratio (LR+) $= \frac{TPR}{FPR}$	Negative likelihood ratio (LR-) $= \frac{FNR}{TNR}$	
Accuracy (ACC) $= \frac{TP + TN}{P + N}$	False discovery rate (FDR) $= \frac{FP}{PP} = 1 - PPV$	Negative predictive value (NPV) $= \frac{TN}{PN} = 1 - FOR$	Markedness (MK), deltaP ( $\Delta p$ ) $= PPV + NPV - 1$	Diagnostic odds ratio (DOR) $= \frac{LR+}{LR-}$	
Balanced accuracy (BA) $= \frac{TPR + TNR}{2}$	$F_1$ score $= \frac{2PPV \times TPR}{PPV + TPR} = \frac{2TP}{2TP + FP + FN}$	Fowlkes—Mallows index (FM) $= \sqrt{PPV \times TPR}$	Matthews correlation coefficient (MCC) $= \sqrt{TPR \times TNR \times PPV \times NPV} - \sqrt{FNR \times FPR \times FOR \times DOR}$	Threat score (TS), critical success index (CSI), Jaccard index $= \frac{TP}{TP + FN + FP}$	

Source: Wiki-en



# 引力波信号搜寻：卷积神经网络

- 真正例率 (True Positive Rate, TPR) 和假正例率 (False Positive Rate, FPR)
- 在引力波数据处理中, TPR 和 FPR 可以分别与引力波信号预警模型在引力波数据上的探测灵敏度 (sensitivity) [或  $P_{astro}$ ] 和误报率 (false alarm rate, FAR) 相对应。

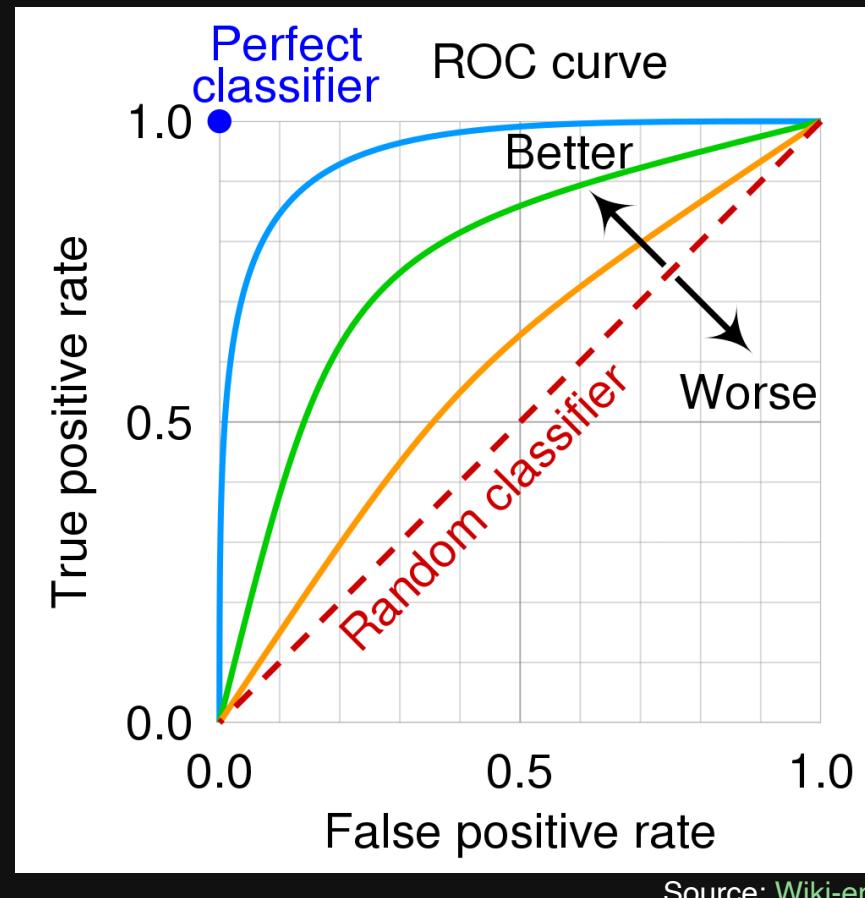
		Predicted condition		Sources: [17][18][19][20][21][22][23][24][25] view · talk · edit	
		Positive (PP)	Negative (PN)	Informedness, bookmaker informedness (BM) $= TPR + TNR - 1$	Prevalence threshold (PT) $= \frac{\sqrt{TPR \times FPR} - FPR}{TPR - FPR}$
Actual condition	Positive (P)	True positive (TP), hit	False negative (FN), type II error, miss, underestimation	True positive rate (TPR), recall, sensitivity (SEN), probability of detection, hit rate, power $= \frac{TP}{P} = 1 - FNR$	False negative rate (FNR), miss rate $= \frac{FN}{P} = 1 - TPR$
	Negative (N)	False positive (FP), type I error, false alarm, overestimation	True negative (TN), correct rejection	False positive rate (FPR), probability of false alarm, fall-out $= \frac{FP}{N} = 1 - TNR$	True negative rate (TNR), specificity (SPC), selectivity $= \frac{TN}{N} = 1 - FPR$
Prevalence $= \frac{P}{P+N}$	Positive predictive value (PPV), precision $= \frac{TP}{PP} = 1 - FDR$	False omission rate (FOR) $= \frac{FN}{PN} = 1 - NPV$	Positive likelihood ratio (LR+) $= \frac{TPR}{FPR}$	Negative likelihood ratio (LR-) $= \frac{FNR}{TNR}$	
Accuracy (ACC) $= \frac{TP + TN}{P + N}$	False discovery rate (FDR) $= \frac{FP}{PP} = 1 - PPV$	Negative predictive value (NPV) $= \frac{TN}{PN} = 1 - FOR$	Markedness (MK), deltaP ( $\Delta p$ ) $= PPV + NPV - 1$	Diagnostic odds ratio (DOR) $= \frac{LR+}{LR-}$	
Balanced accuracy (BA) $= \frac{TPR + TNR}{2}$	$F_1$ score $= \frac{2PPV \times TPR}{PPV + TPR} = \frac{2TP}{2TP + FP + FN}$	Fowlkes-Mallows index (FM) $= \sqrt{PPV \times TPR}$	Matthews correlation coefficient (MCC) $= \sqrt{TPR \times TNR \times PPV \times NPV} - \sqrt{FNR \times FPR \times FOR \times DOR}$	Threat score (TS), critical success index (CSI), Jaccard index $= \frac{TP}{TP + FN + FP}$	

Source: Wiki-en

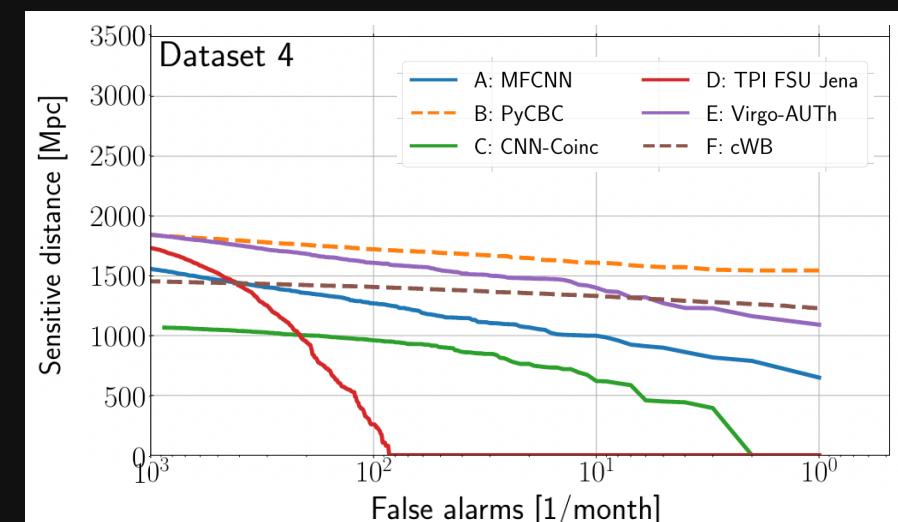


# 引力波信号搜寻：卷积神经网络

- 在不同的阈值选取下，我们分别以 TPR 和 FPR 为纵、横轴作图，就得到了 ROC (Receiver Operating Characteristic, 受试者工作特征) 曲线。



- 图上的每一点，都对应于一种阈值选取可能。
- 对角线对应于“随机猜测”模型，而过左上点(0,1)的曲线就是预测最佳的“理想模型”。
  - 阈值越大，对应于 ROC 的左下角方向，
  - 阈值越小，对应于 ROC 右上角的方向。

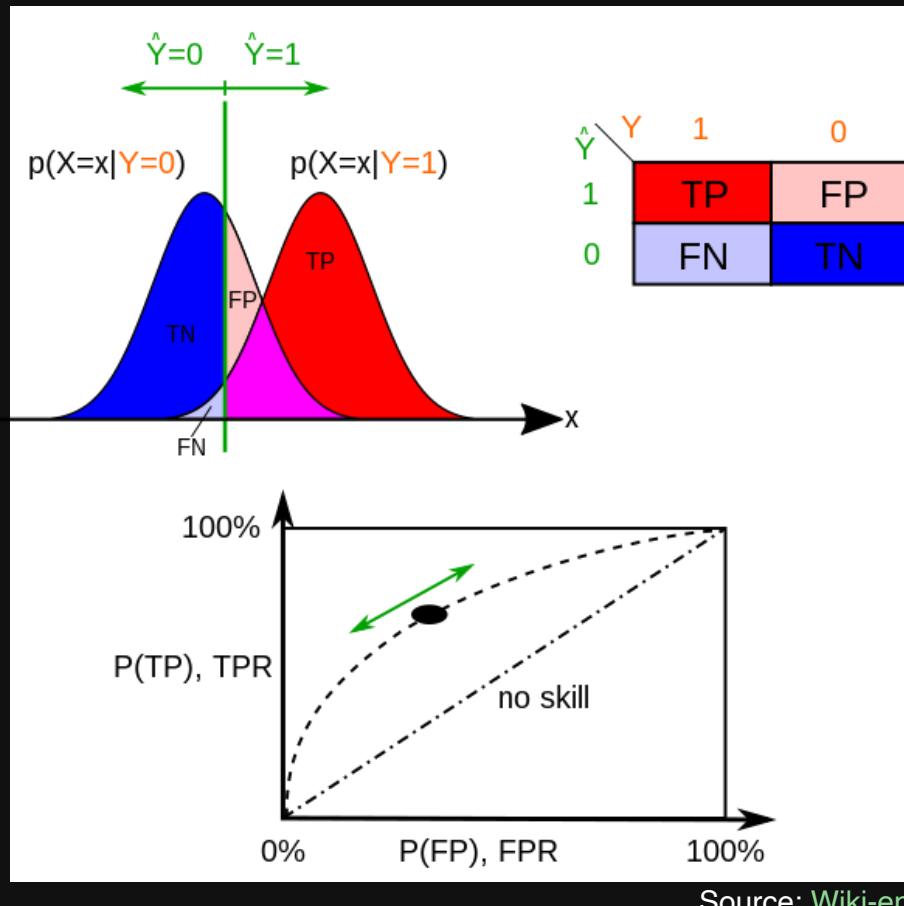


Schäfer, Marlin B., et al. PRD 107, no. 2 (2023): 023021

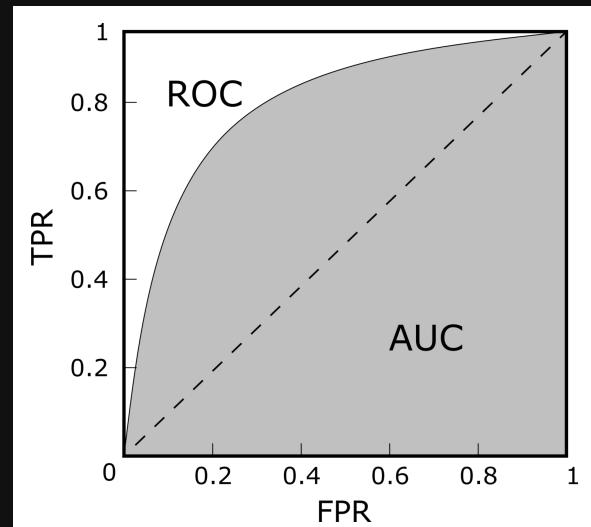


# 引力波信号搜寻：卷积神经网络

- 在不同的阈值选取下，我们分别以 TPR 和 FPR 为纵、横轴作图，就得到了 ROC (Receiver Operating Characteristic, 受试者工作特征) 曲线。



- 对不同的机器学习模型进行比较时，若某模型的 ROC 曲线可以“包住”另一个模型的曲线，则可断言前者的性能优于后者。
- 为了能避免曲线交叉会带来的含糊，一个定量的合理判据就是比较 ROC 曲线下的面积，即 AUC (Area Under ROC Curve)





# 引力波信号搜寻：卷积神经网络

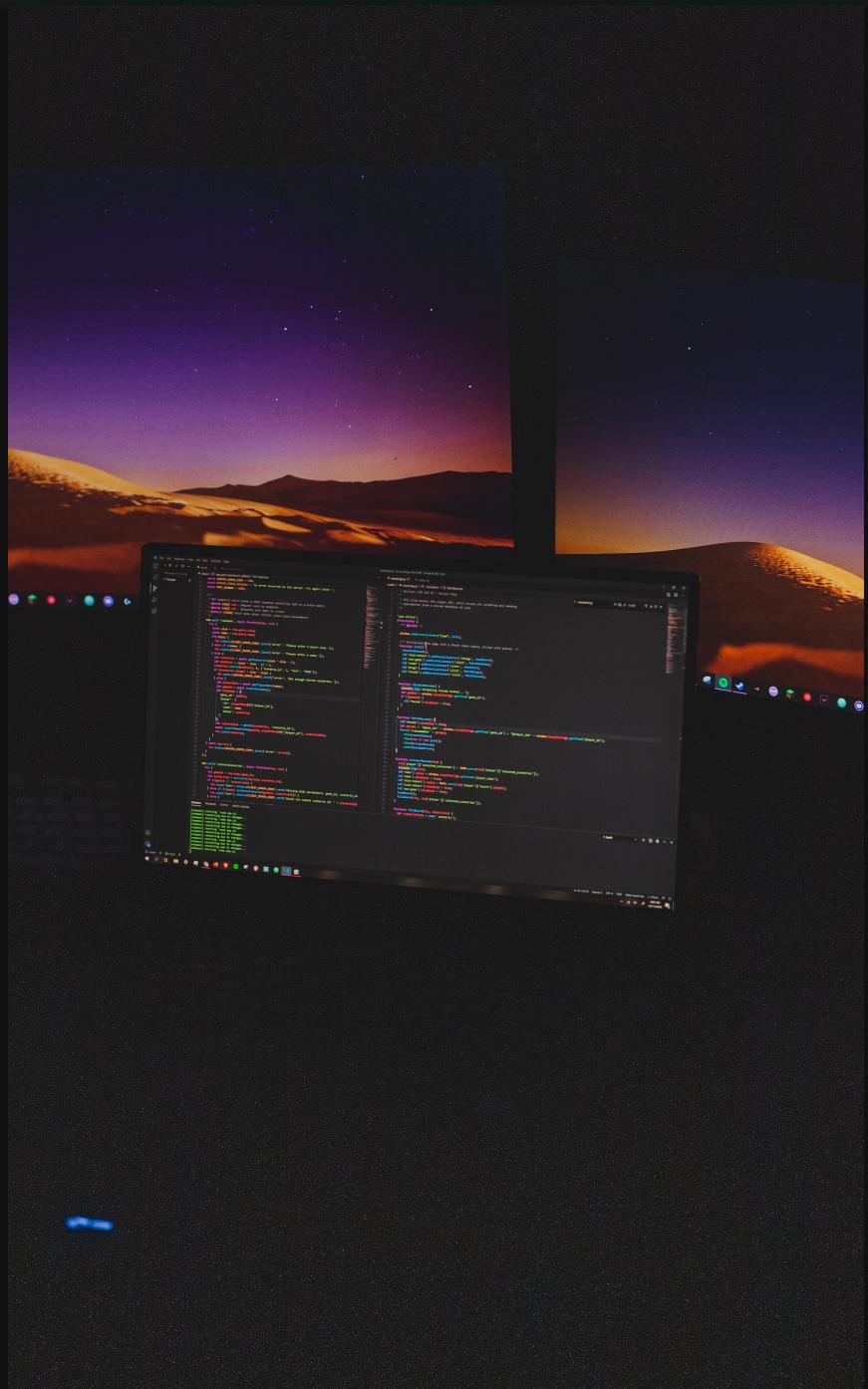


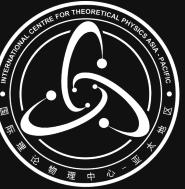
“Talk is  
cheap. Show  
me the code.”

Linus Torvalds

很多其实很重要但没能讲到的内容：

- 网络权重初始化
- 输入数据的标准化
- 感受野
- 正则化
- 类别不均衡
- K 折交叉验证
- 模型的集成与融合
- Fine-tune (迁移学习)
- ...





# Hackathon

Can you find the GW signals?

kaggle™



Hackathon: Can you find the GW signals?

数据科学挑战：引力波信号搜寻



## Can you find the GW signals?

Gravitational Wave Data Exploration: A Practical Training in Programming and Analysis (2023)

Home: <https://www.kaggle.com/competitions/2023-gwdata-bootcamp>

本竞赛将于北京时间 2023年12月29日22:00 开始，并于北京时间 2024年1月5日23:59 结束。

请确保在截止日期前提交你的解决方案。