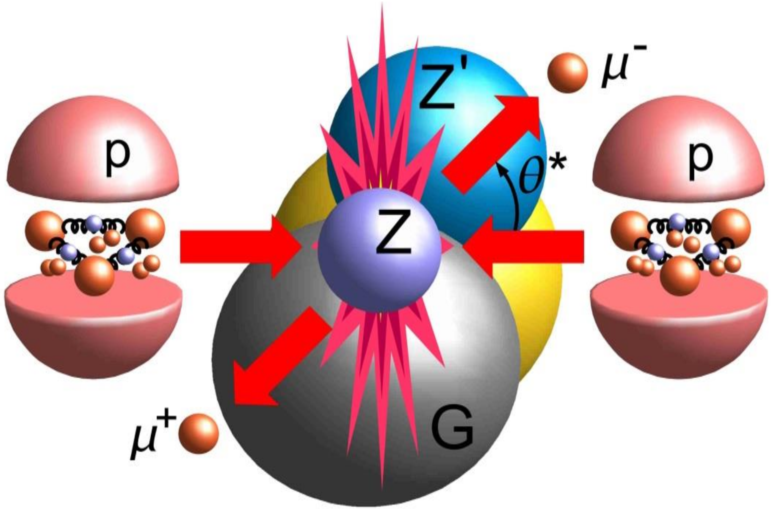
计算物理 第二部分

第8讲

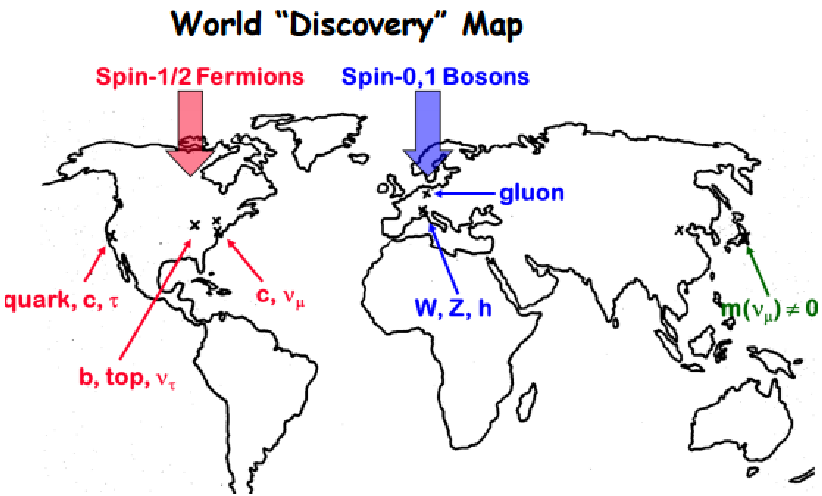
李强　北京大学物理学院中楼411

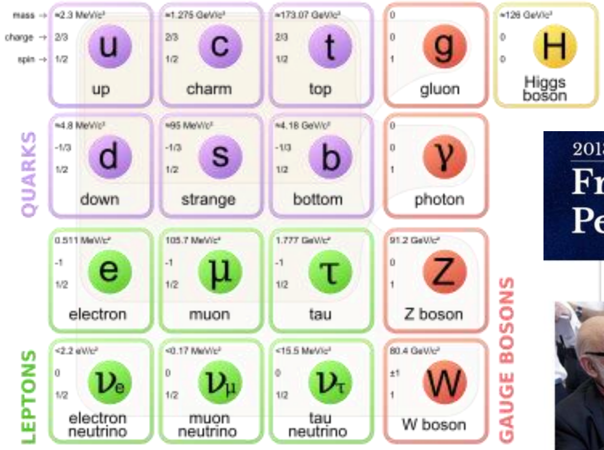
qliphy0@pku.edu.cn， 15210033542

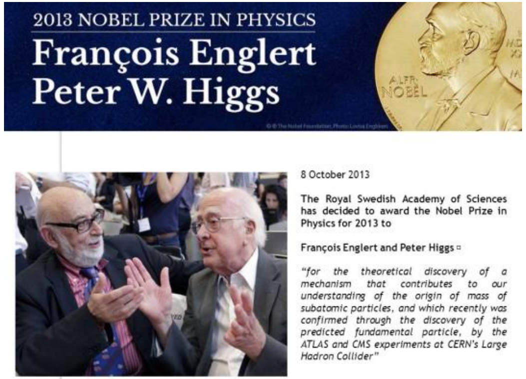
机器学习在高能物理中的应用



高能对撞机： 物理简介

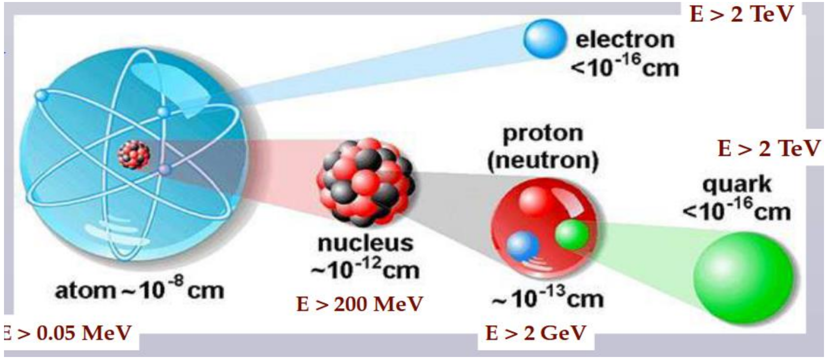






$$\Delta x \Delta p \geq \frac{\hbar}{2}$$

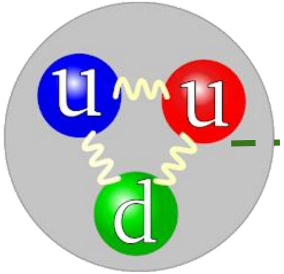
$$(1\ \mathrm{GeV})^{-1} = 0.2\ \mathrm{fm} = 0.2\ 10^{-15}\ \mathrm{m}$$

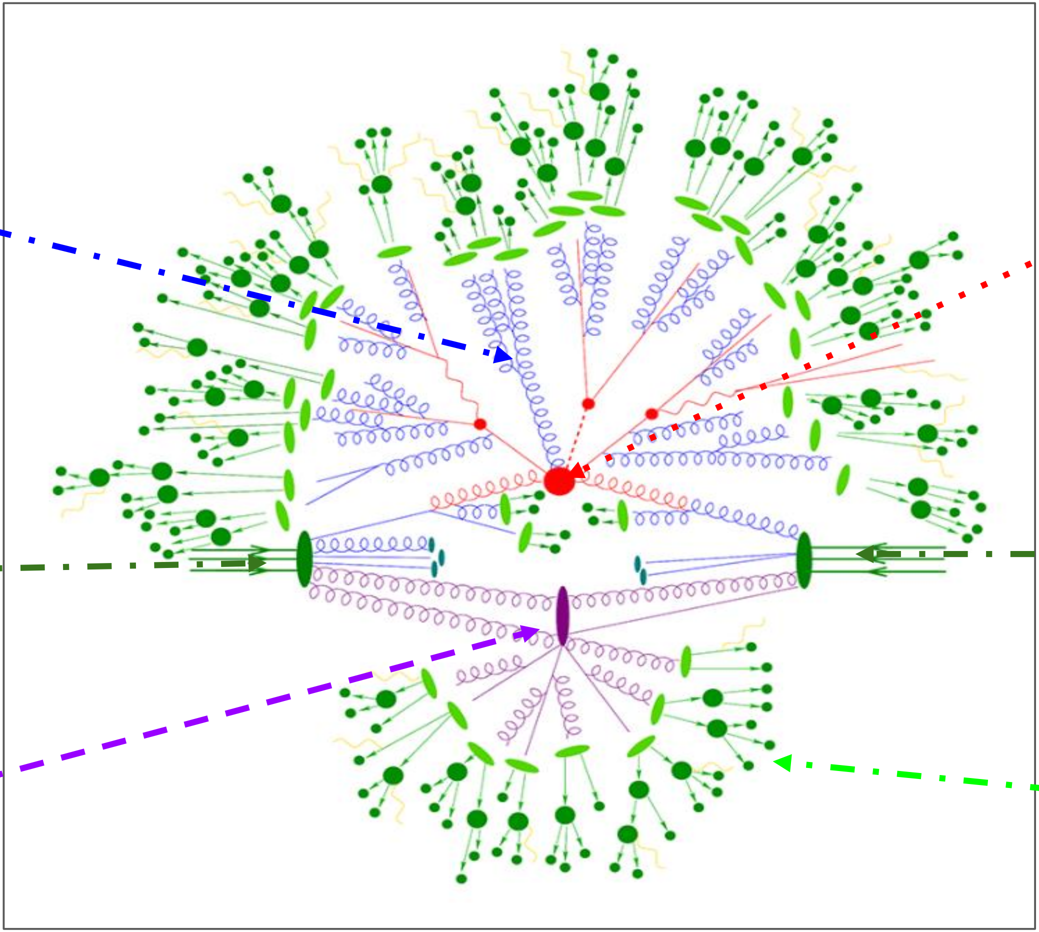


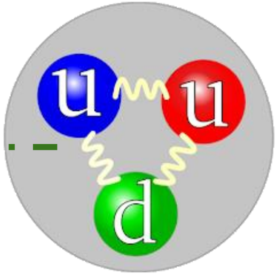
高能对撞

硬散射

QCD演化：
Parton Shower



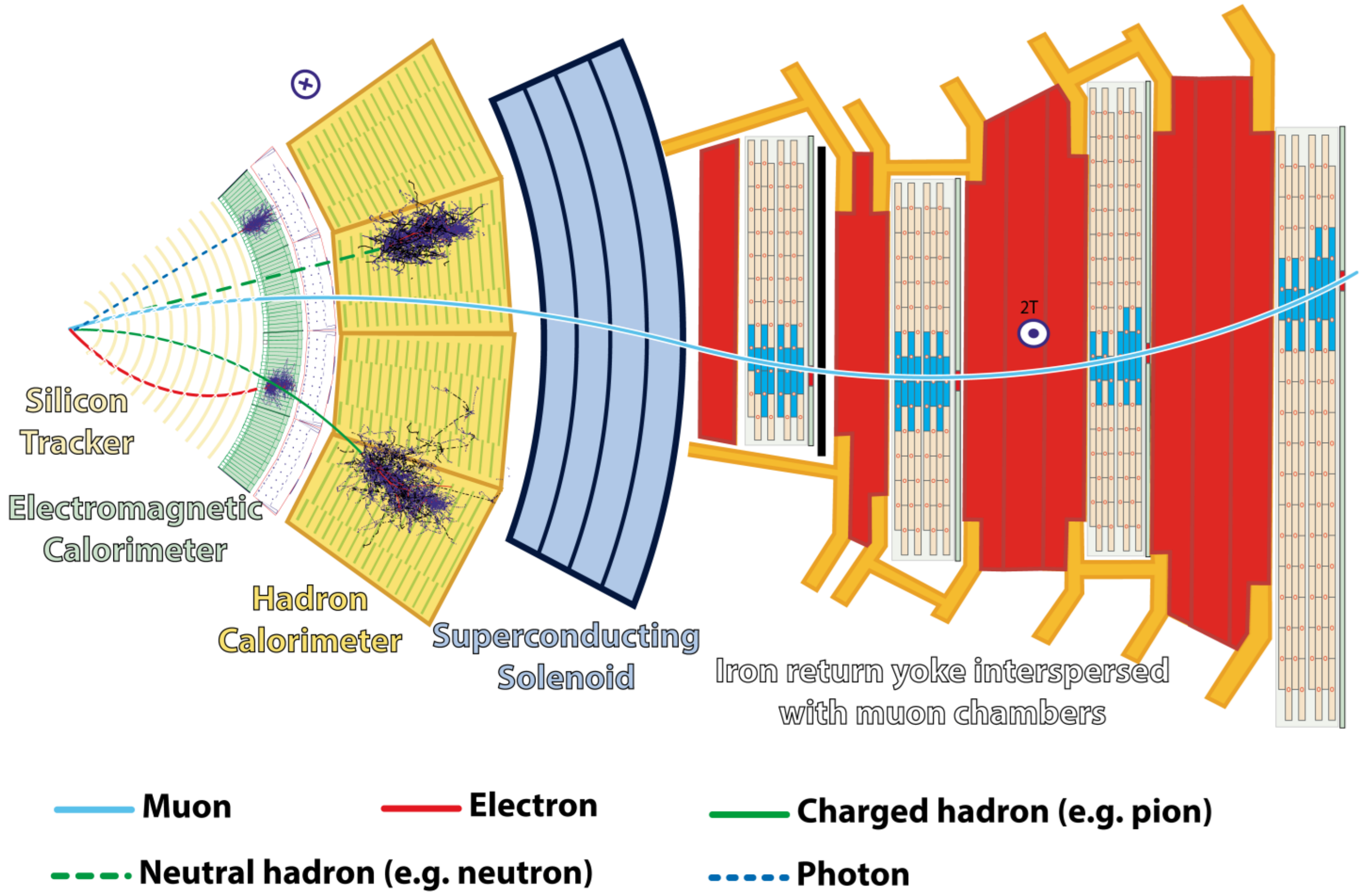


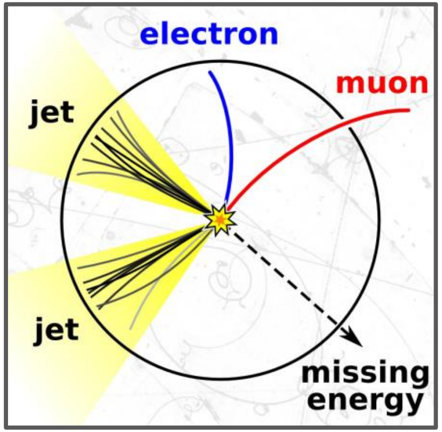


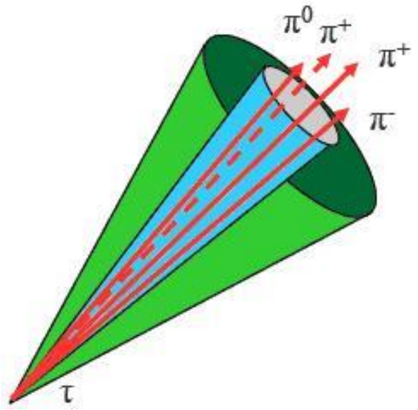
多重散射

强子化

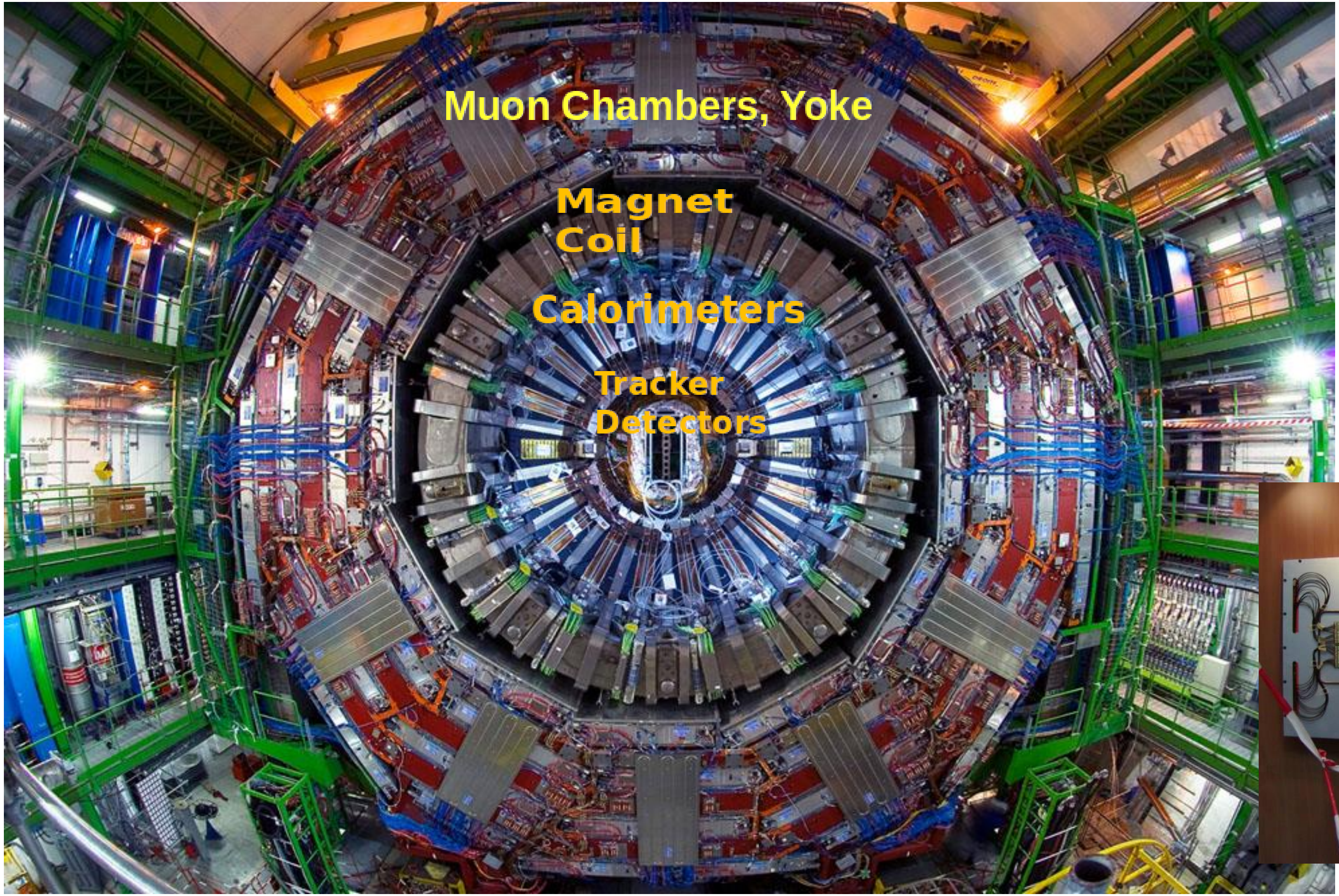
高能对撞机： 探测⟶信息


2T
Silicon
Tracker
Electromagnetic
Calorimeter
Hadron
Calorimeter
Superconducting
Solenoid
Iron return yoke interspersed
with muon chambers
Muon
Electron
Charged hadron (e.g. pion)
Neutral hadron (e.g. neutron)
Photon


electron
muon
jet
jet
missing
energy

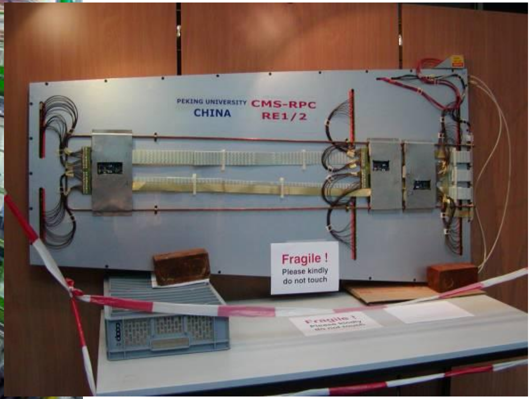

π^0 π^+
π^+
π^-
τ

高能对撞机： CMS探测器

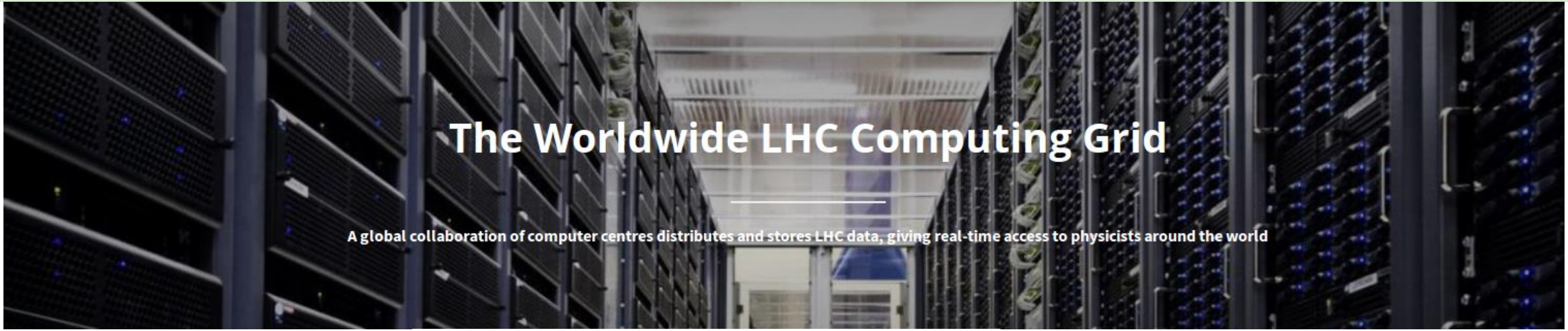


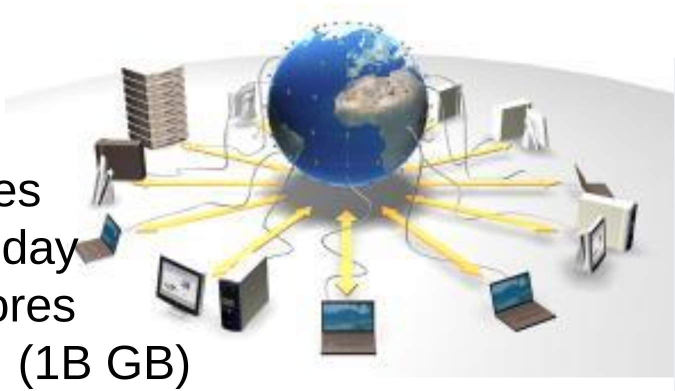
weight: 12500 t
overall diameter: 15 m
overall length: 21.6 m

照相机？
录音机？

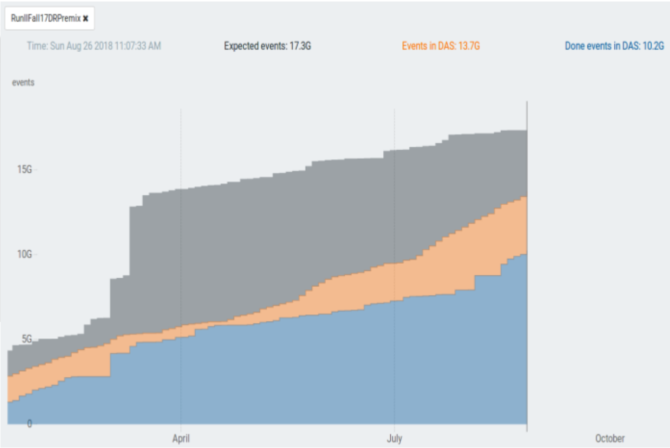


高能对撞机：大数据



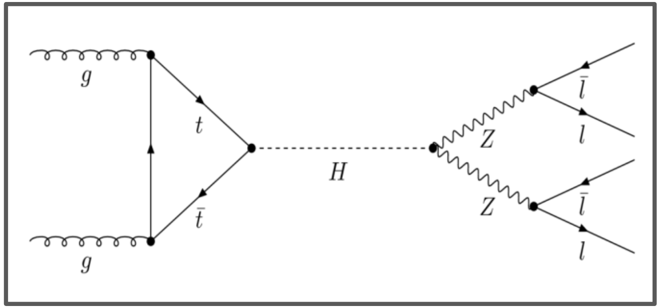


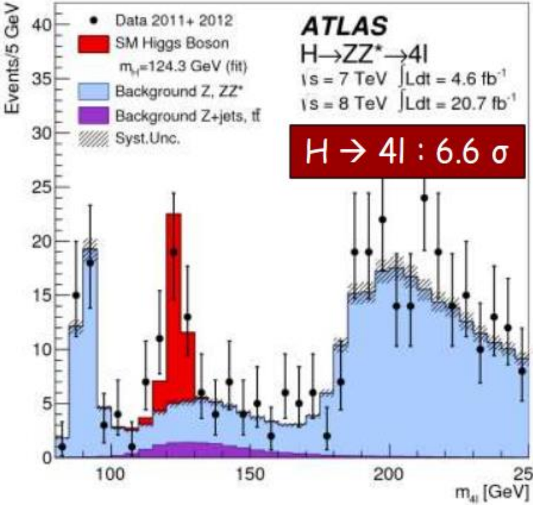
42 countries
170 computing centres
Over 2 million tasks/ day
1 million computer cores
1 exabyte of storage (1B GB)

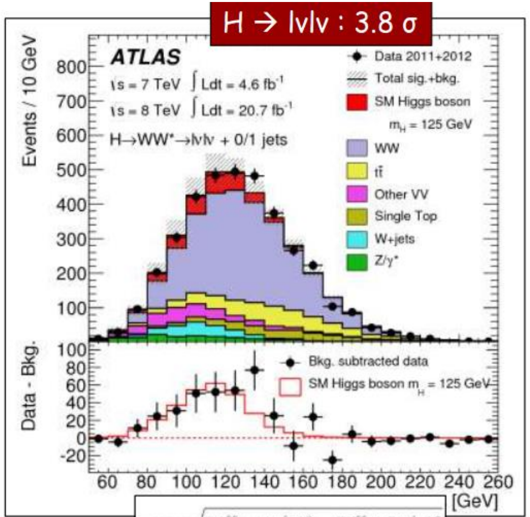


CMS： 15B events in 8 months

高能对撞机：大数据







$$m_T = \sqrt{(E_T^{\ell\ell} + E_T^{\text{miss}})^2 - (\mathbf{P}_T^{\ell\ell} + \mathbf{P}_T^{\text{miss}})^2}$$

Needles in a haystack

In ATLAS, up to July 4, 2012:

A million billion collisions

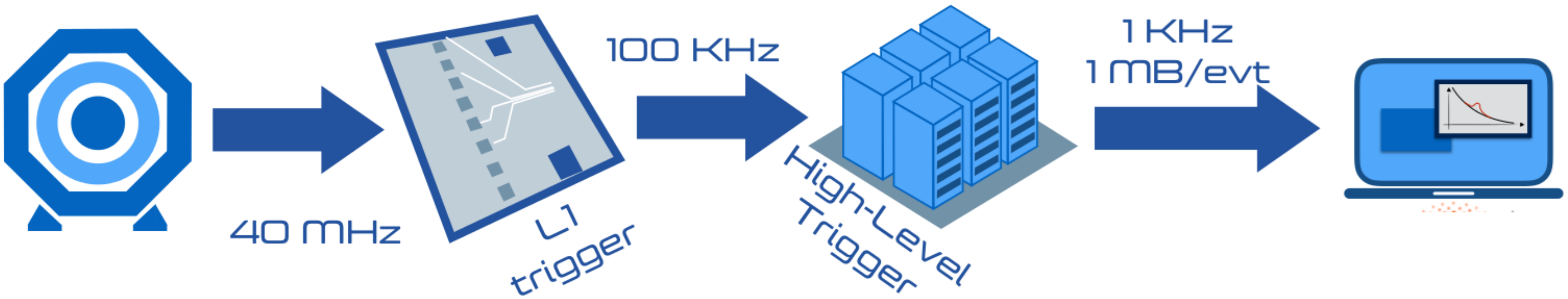
4.2 billion events analyzed

240,000 Higgs particles produced

~**350** diphoton Higgs events detected

~**8** four-lepton Higgs events detected

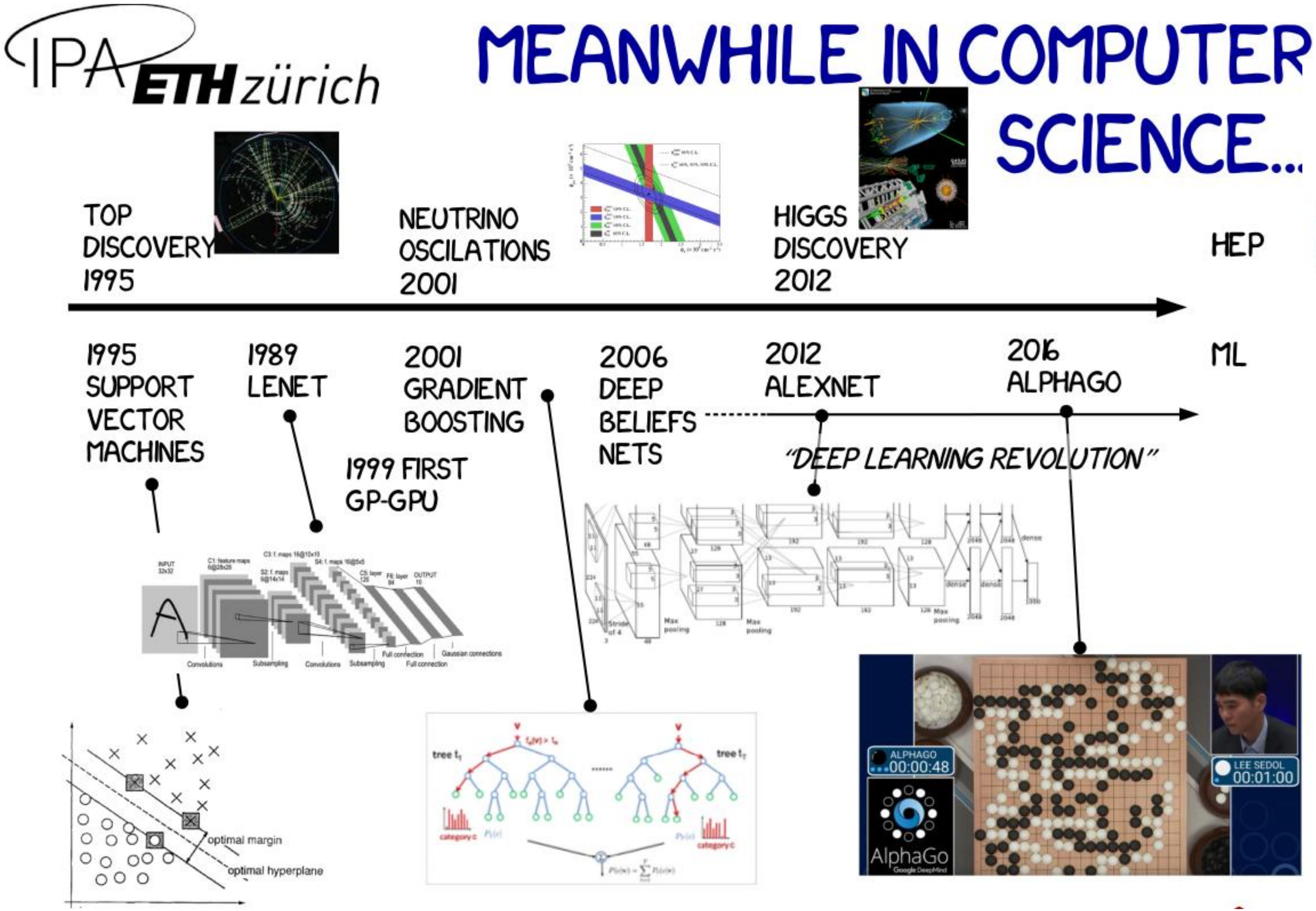
LHC数据流

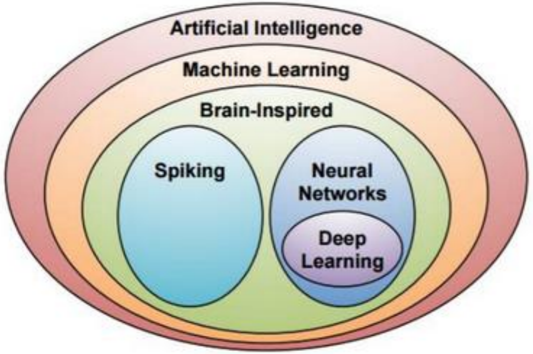


- **L1 trigger:** local, hardware based, on FPGA, @experiment site
- **HLT:** local/global, software based, on CPU, @experiment site
- **Offline:** global, software based, on CPU, @CERN T0
- **Analysis:** user-specific applications running on the grid

机器学习： 粒子鉴别；信号挖掘；快速判断；自主学习

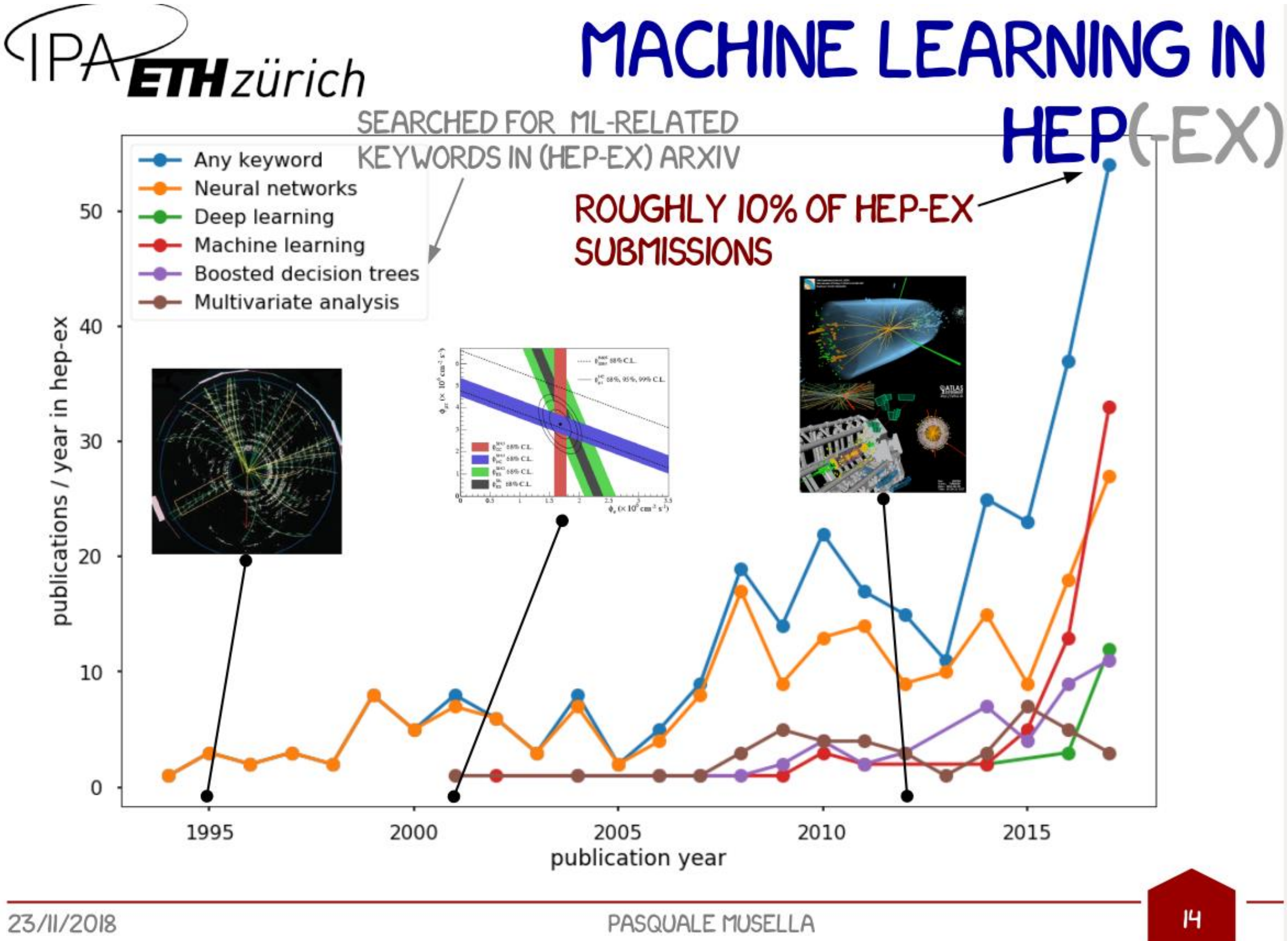
机器学习简史



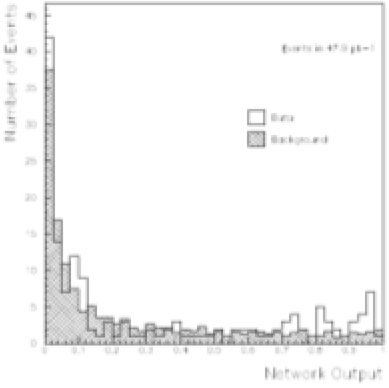


2018年图灵奖

高能物理机器学习应用简史



SEARCH FOR TTBAR USING NN AT D0

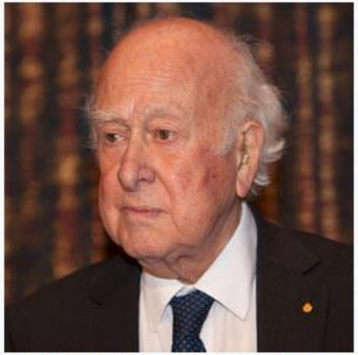


[HEP-EX/9507007]

Tevatron：Top夸克
LHC: Higgs发现
miniBOONE: 粒子鉴别

高能物理及机器学习

Peter Higgs
CH FRS FRSE FInstP



Nobel laureate Peter Higgs at a press conference, Stockholm, December 2013

Born	Peter Ware Higgs 29 May 1929 (age 90) Newcastle upon Tyne, England, UK
Residence	Edinburgh, Scotland, UK
Nationality	British[1]
Alma mater	King's College London (BSc, MSc, PhD)
Known for	Higgs boson Higgs field Higgs mechanism Symmetry breaking

Institutions	University of Edinburgh Imperial College London University College London King's College London
Thesis	*Some problems in the theory of molecular vibrations* (1955)
Doctoral advisor	Charles Coulson[2][3] Christopher Longuet-Higgins[2][4]

Charles Alfred Coulson: 应用数学家，化学家

Christopher Longuet-Higgins，理论化学家，40岁（1970s），改行做人工智能

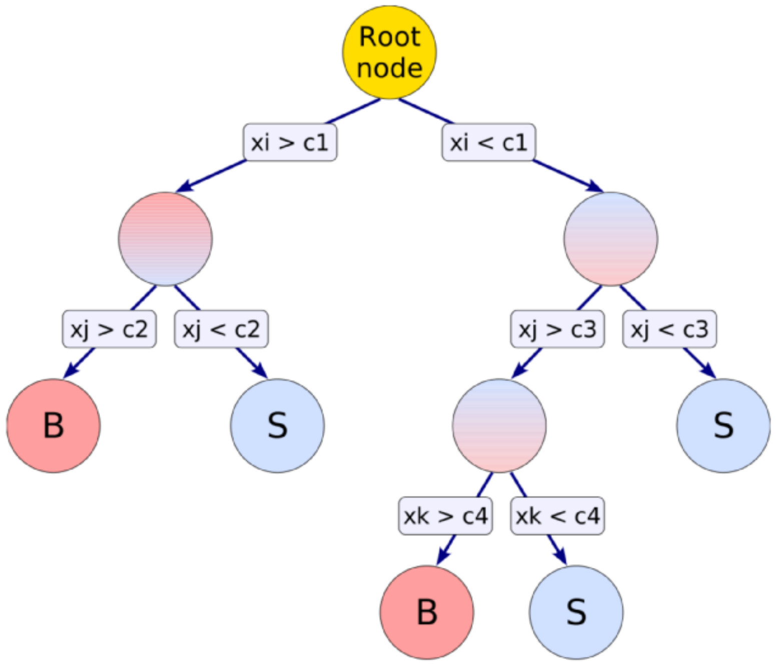
Doctoral advisor	Christopher Longuet-Higgins[3][4][5]
Doctoral students	Richard Zemel[6] Brendan Frey[7] Radford M. Neal[8] Ruslan Salakhutdinov[9] Ilya Sutskever[10]
Other notable students	Yann LeCun (postdoc) Peter Dayan (postdoc) Zoubin Ghahramani (postdoc)

Geoffrey Hinton
FRS FRSC CC

Hinton in 2013

Born	Geoffrey Everest Hinton 6 December 1947 (age 71)[1] Wimbledon, London
Residence	Canada
Alma mater	University of Cambridge (BA) University of Edinburgh (PhD)

BDT简介



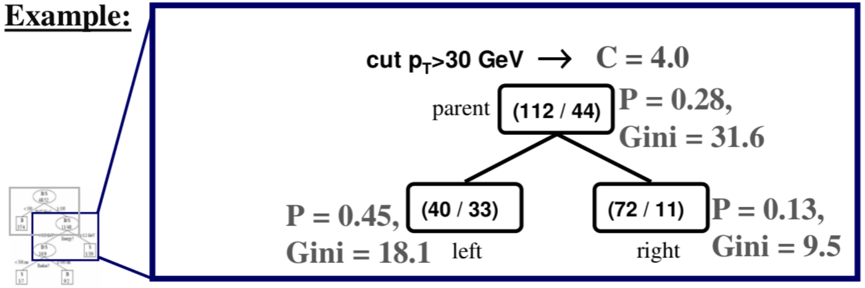
Tree
Node
Leaf

Gini: Note that Gini is 0 for all signal or all background. W_i is the weight of event "i".

$$G_{ini} = \left(\sum_{i=1}^{n} W_i \right) P(1-P)$$

- Pick the branch to maximize the change in gini.
- Criterion C = Gini$_{\text{parent}}$ – Gini$_{\text{right-child}}$ –Gini$_{\text{left-child}}$

Example:



- Optimize each node (e.g. p_T>30 GeV) by maximizing "C".

BDT简介

- Easy to understand/interpret； Training Fast
- Single tree are not stable
  - a small change/fluctuation in the data can make a large difference!
- Solution: e.g. Boosting! → Boosted Decision Trees
  - Each tree is created iteratively
  - The tree's output (*h(x)*) is given a weight (*w*) relative to its accuracy
  - The ensemble output is the weighted sum:

$$\hat{y}(x) = \sum_t w_t h_t(x)$$

  - After each iteration each data sample is given a weight based on its misclassification
    - The more often a data sample is misclassified, the more important it becomes
  - The goal is to minimize an objective function

$$O(x) = \sum_i l(\hat{y}_i, y_i) + \sum_t \Omega(f_t)$$

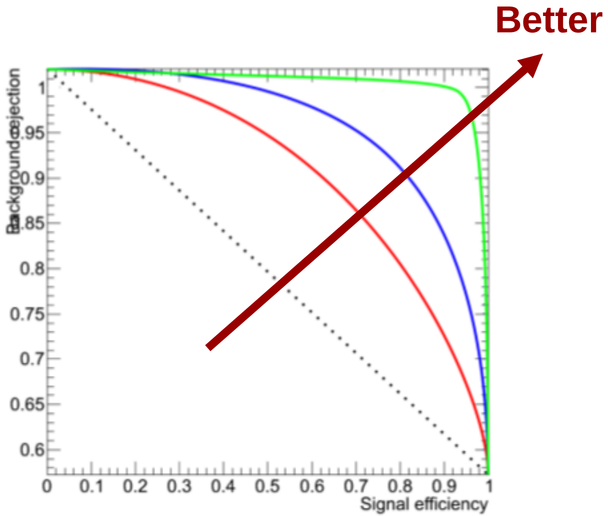
    - $l(\hat{y}_i, y_i)$ is the loss function --- the distance between the truth and the prediction of the *i*th sample
    - $\Omega(f_t)$ is the regularization function --- it penalizes the complexity of the *t*th tree

https://en.wikipedia.org/wiki/Boosting_(machine_learning)

BDT简介

- **AdaBoost “Adaptive Boosting”**
  - One of the originals
  - Freund and Schapire
- **Gradient Boosting**
  - Uses gradient descent to create new learners
  - The loss function is differentiable
  - Friedman: https://statweb.stanford.edu/~jhf/ftp/trebst.pdf
- **XGBoost “eXtreme Gradient Boosting”**
  - Type of gradient boosting
  - Has become very popular in data science competitions
  - Chen and Guestrin: https://arxiv.org/abs/1603.02754

ROC (Receiver Operating Characteristic) Curves

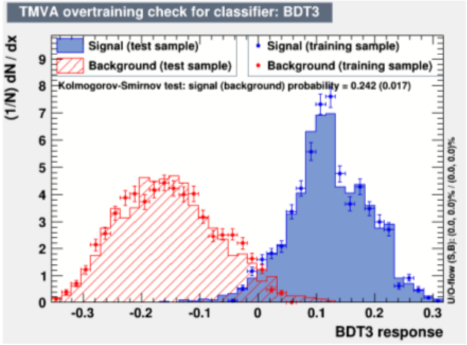


Best classifier can be identified by the largest **AUC (**Area under curve)

Overtraining check:

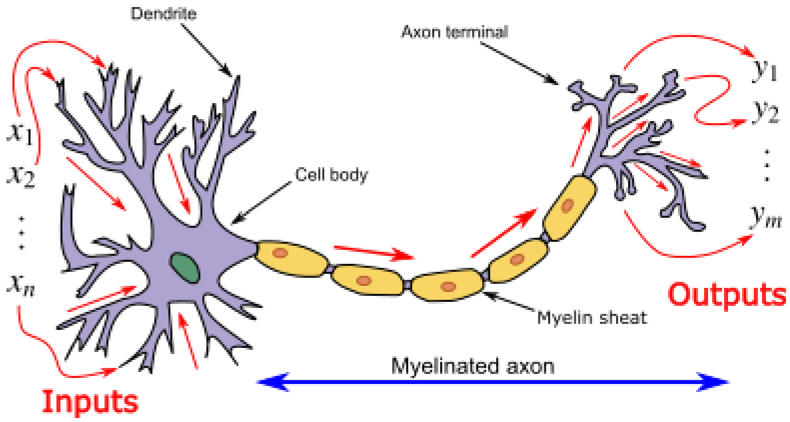
- Split data in **training / test**
- Performance on the training samples should not be better than on the test sample

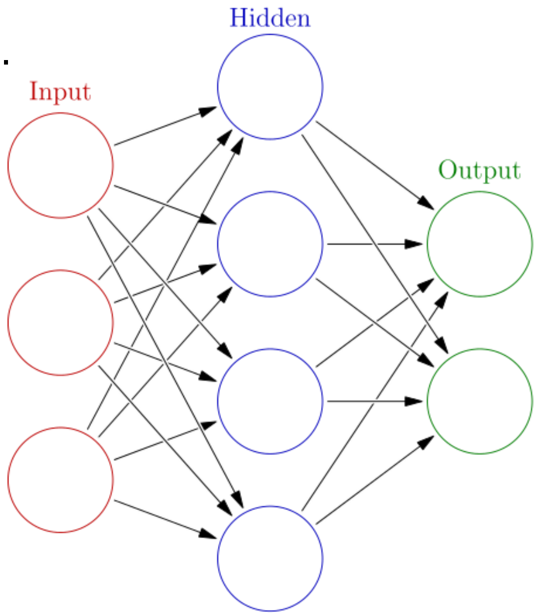
2500 trees



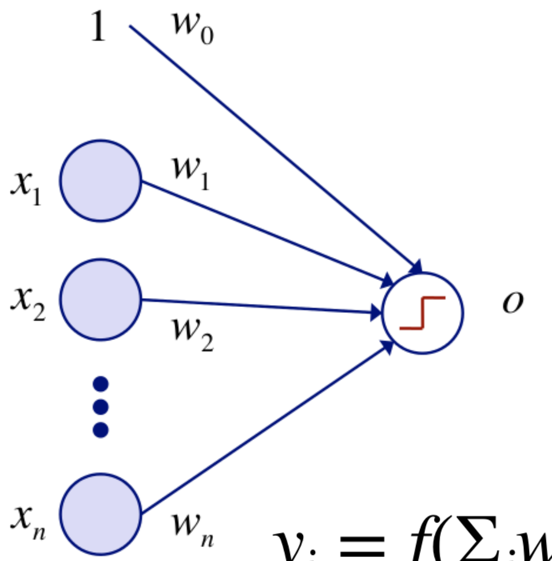
NN简介

- Artificial Neural Networks, connectionist models
- inspired by interconnected neurons in biological systems
    - First Mathematical model of neurons   Pitts & McCulloch (1943)
    - 1986 **Backpropagation** reinvented: Rumelhart, Hinton et al.Nature
    - **1990s： Great success of SVM and graphical models almost kills the ANN**
    - **Yann LeCun (1998) developed deep convolutional neural networks**
        - LeNet-5, a pioneering 7-level convolutional network
    - 2006+： Deep learning is a rebranding of ANN research.
        - Convolutional neural networks running on GPUs





NN简介



$$y_i = f(\Sigma_j w_{ij} x_j + b_i)$$

given: network structure and a training set $D = \{(\boldsymbol{x}^{(1)}, y^{(1)}) \ldots (\boldsymbol{x}^{(m)}, y^{(m)})\}$

initialize all weights in $\boldsymbol{w}$ to small random numbers

until stopping criteria met do

for each $(\boldsymbol{x}^{(d)}, y^{(d)})$ in the training set

input $\boldsymbol{x}^{(d)}$ to the network and compute output $o^{(d)}$

calculate the error $E(\boldsymbol{w}) = \frac{1}{2}\left(y^{(d)} - o^{(d)}\right)^2$

calculate the gradient

$$\nabla E(\boldsymbol{w}) = \left[\frac{\partial E}{\partial w_0}, \frac{\partial E}{\partial w_1}, \cdots, \frac{\partial E}{\partial w_n}\right]$$

update the weights

$$\Delta \boldsymbol{w} = -\eta \, \nabla E(\boldsymbol{w})$$

b： bias term

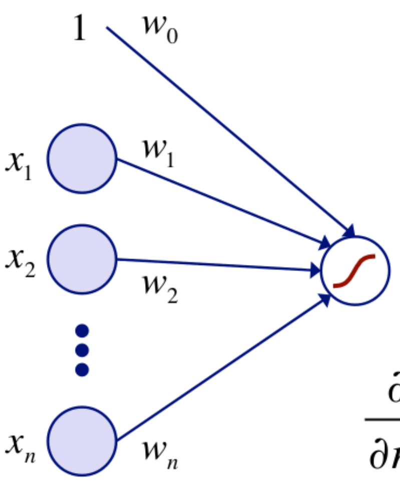
η： learning rate

Standard gradient descent (batch training)

Stochastic gradient descent (online training)

NN简介

如果不用激活函数，每一层输出都是上层输入的线性函数，无论神经网络有多少层，输出都是输入的线性组合，这种情况就是最原始的感知机

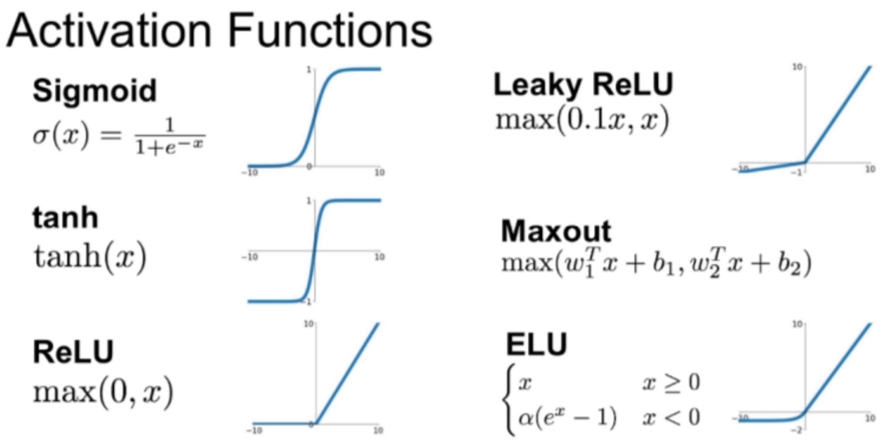


$$net^{(d)} = w_0 + \sum_{i=1}^{n} w_i x_i^{(d)}$$

$$o^{(d)} = \frac{1}{1 + e^{-net^{(d)}}}$$

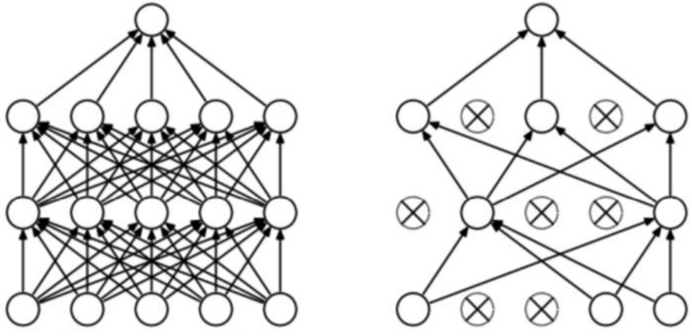
Sigmoid 函数

$$\frac{\partial o^{(d)}}{\partial net^{(d)}} = o^{(d)}(1 - o^{(d)})$$



线性整流函数

$$\frac{\partial E^{(d)}}{\partial w_i} = \frac{\partial}{\partial w_i} \frac{1}{2}\left(y^{(d)} - o^{(d)}\right)^2 = -\left(y^{(d)} - o^{(d)}\right) o^{(d)}(1 - o^{(d)}) x_i^{(d)}$$

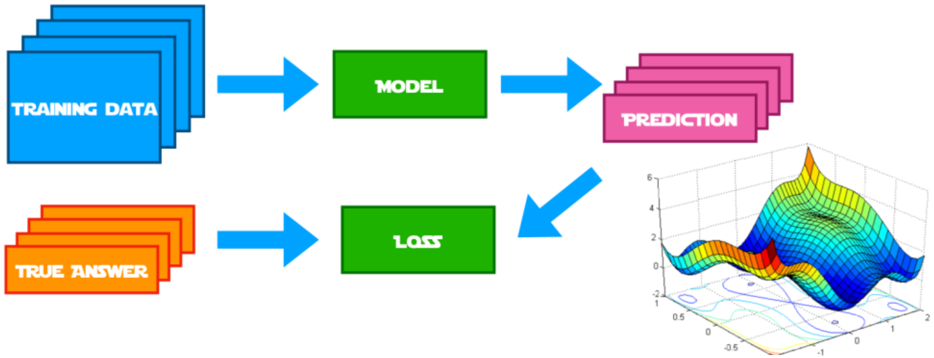


(a) Standard Neural Net (b) After applying dropout.

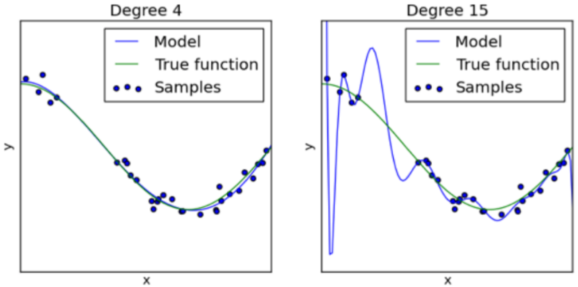
Dropout layer:

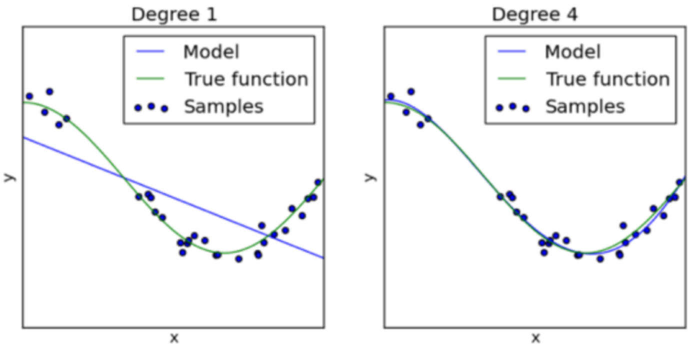
Randomly drop links between neurons, with probability p

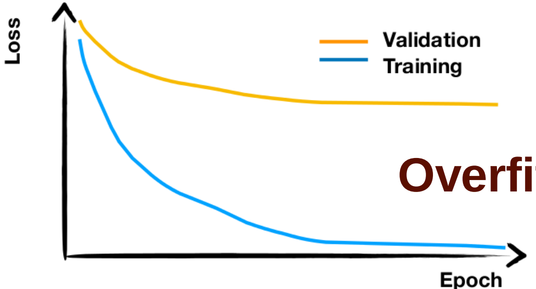
Supervised Learning



Training | Validation | Test

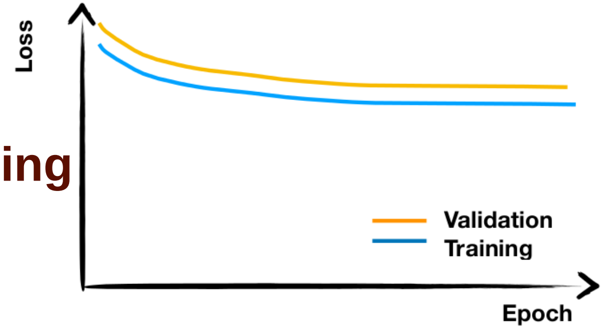






Overfitting

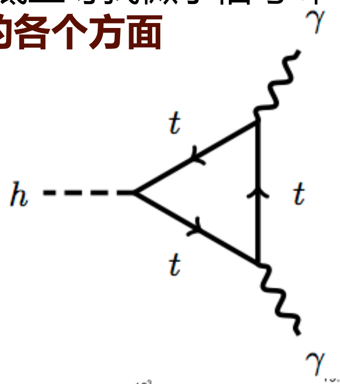
underfitting

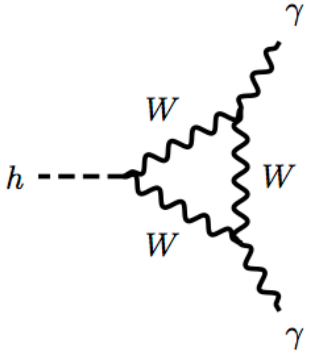


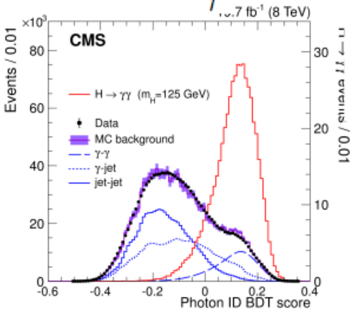
机器学习应用： Higgs粒子寻找

CMS实验中Higgs双光子道的寻找：

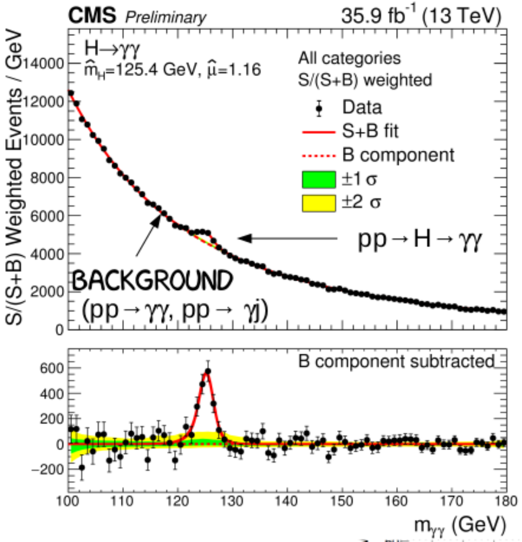
- 分支比10^{-3}: 在本底上寻找微小信号峰；
- **BDT应用于分析的各个方面**
  - 光子鉴别
  - 事例分类
  - 光子能量
  - 双光子顶点



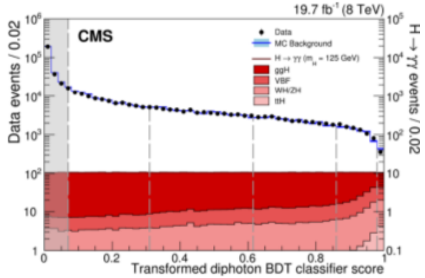




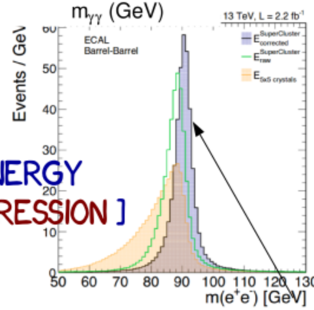
[PARTICLE-ID: SEPARATE PROMPT γ FROM HADRONIC JETS]



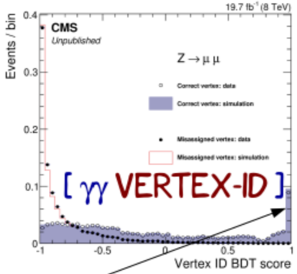
[EVENT CLASSIFICATION]



[INVARIANT MASS ESTIMATION]



[γ ENERGY REGRESSION]



[γγ VERTEX-ID]

$m^2 = E_1E_2(1-\cos\alpha)$

机器学习应用： NNPDF

ANNs provide **universal unbiased interpolants** to parametrize the non-perturbative dynamics that determines the **size and shape of the PDFs** from experimental data

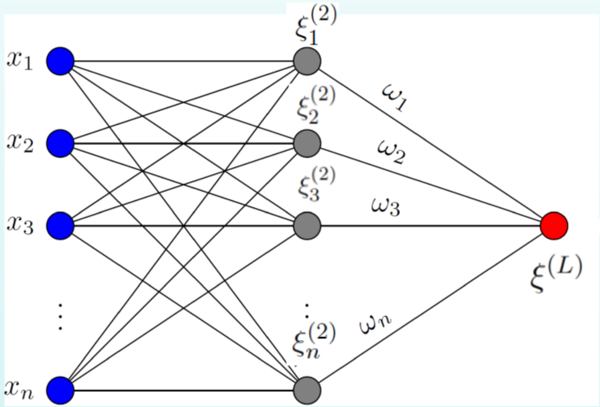
not from QCD!

Traditional approach

$$g(x, Q_0) = A_g(1-x)^{a_g} x^{-b_g} \left(1 + c_g\sqrt{s} + d_g x + \ldots\right)$$

NNPDF approach

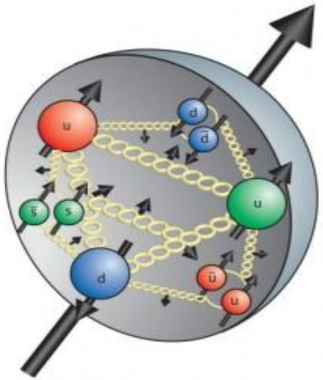
$$g(x, Q_0) = A_g \mathrm{ANN}_g(x)$$

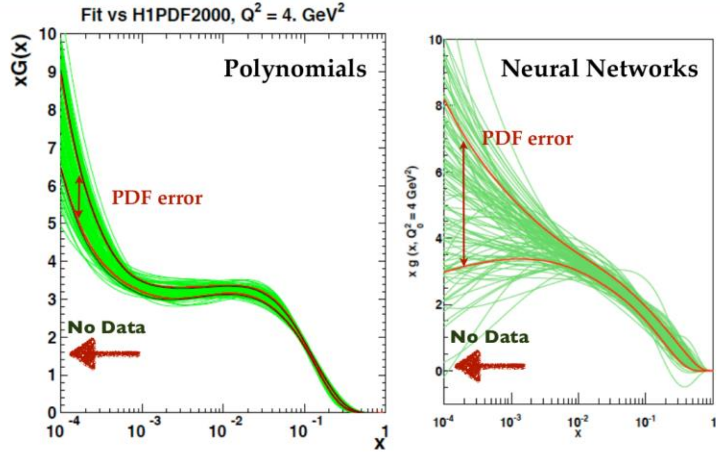


$$\mathrm{ANN}_g(x) = \xi^{(L)} = \mathcal{F}\left[\xi^{(1)}, \{\omega_{ij}^{(l)}\}, \{\theta_i^{(l)}\}\right]$$

$$\xi_i^{(l)} = g\left(\sum_{j=1}^{n_{l-1}} \omega_{ij}^{(l-1)} \xi_j^{(l-1)} - \theta_i^{(l)}\right)$$

- ANNs eliminate **theory bias** introduced in PDF fits from choice of *ad-hoc* functional forms
- NNPDF fits used **O(400) free parameters**, to be compared with O(10-20) in traditional PDFs. Results stable if **O(4000) parameters used!**

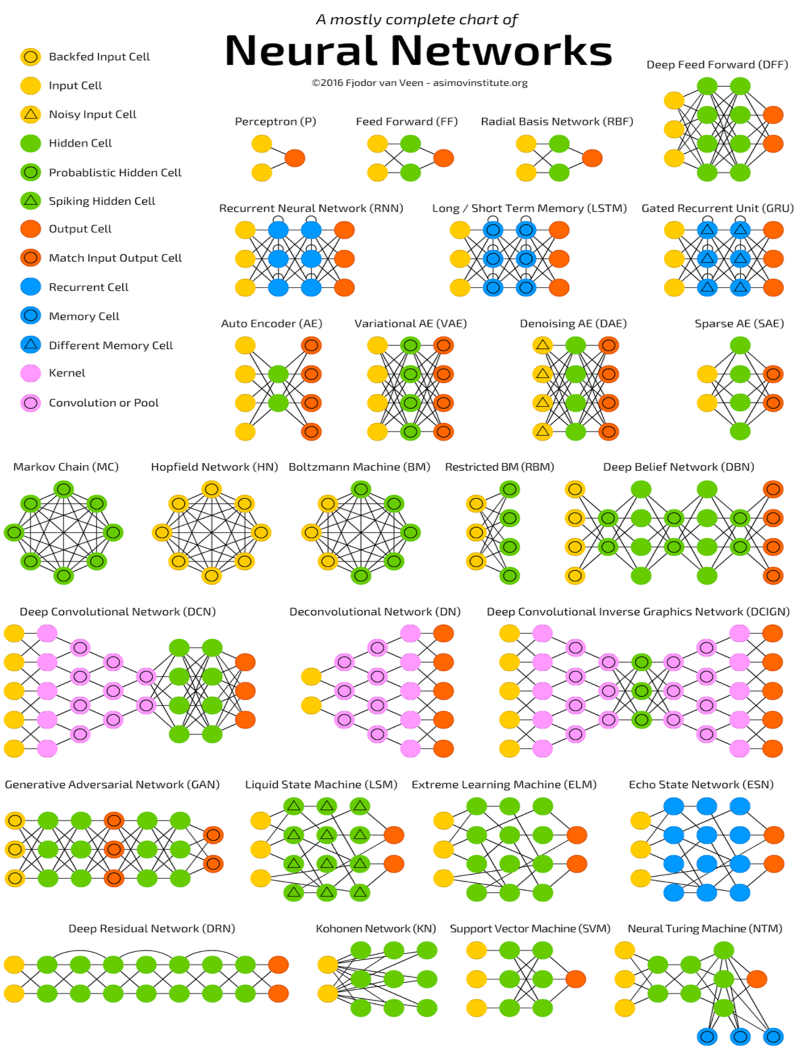




ANNs avoid biasing the PDFs, faithful extrapolation at small-x (very few data, thus error blow up)

深度神经网络DNN

- Deep neural networks are those with >1 inner layer
- Thanks to GPUs, it is now possible to train them efficiently, which boosted the revival of neural networks in the 2000s
- In addition, **new architectures** emerged, which better exploit the new computing power



Universal approximation theorem:

The standard multilayer feed-forward networks with a single hidden layer that contains finite number of hidden neurons, and with arbitrary **activation** functionare universal approximators in C(R^m).

深度神经网络：框架

ROOT

- Data analysis framework for HEP, developed mainly at CERN
- Written in C++ (fully interpreted)

TMVA

- **T**oolkit for **M**ulti**v**ariate **A**nalysis
- Includes several machine learning algorithms such as :
  - *Likelihood, KNN, Fisher, MLP, SVN, Neural Networks, BDT, etc...*

TensorFlow

PyTorch

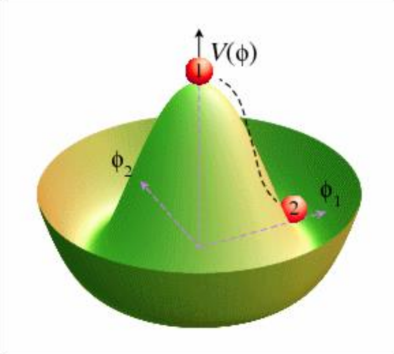
mxnet

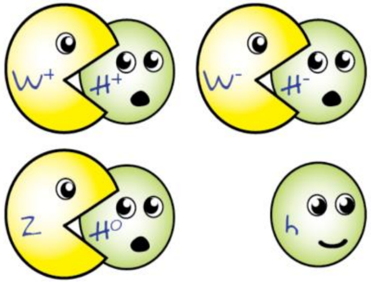
NumPy

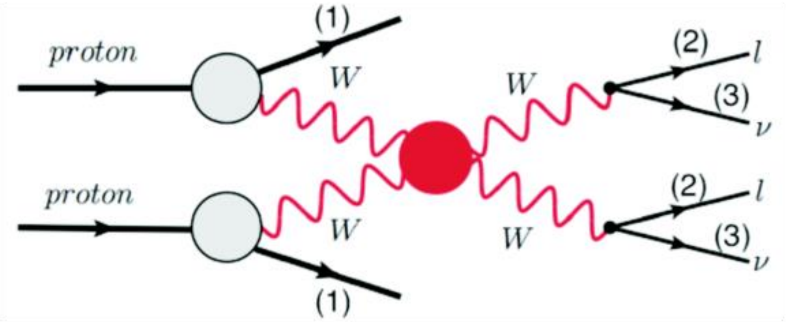
scikit learn

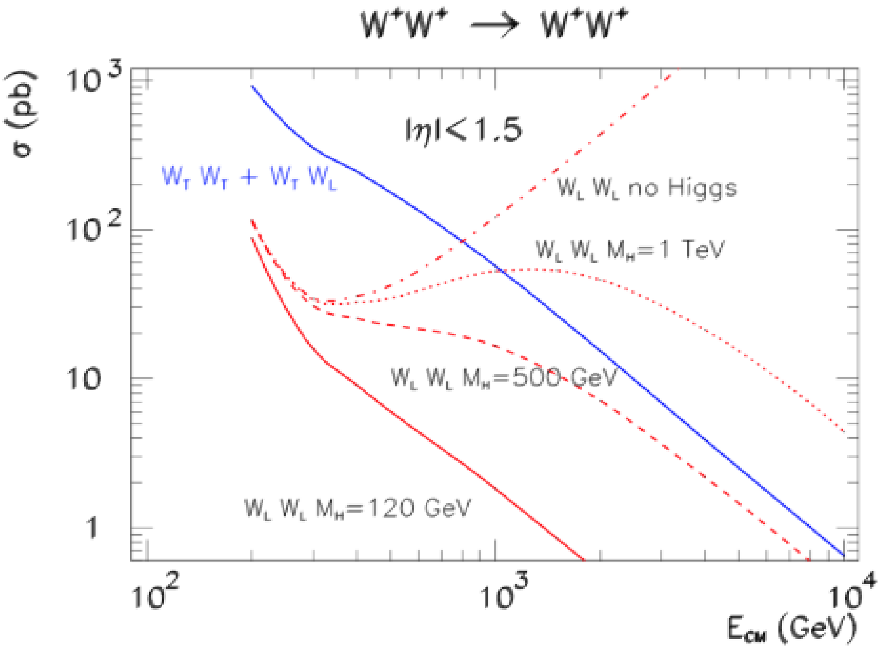
Keras

深度神经网络： 矢量玻色子散射







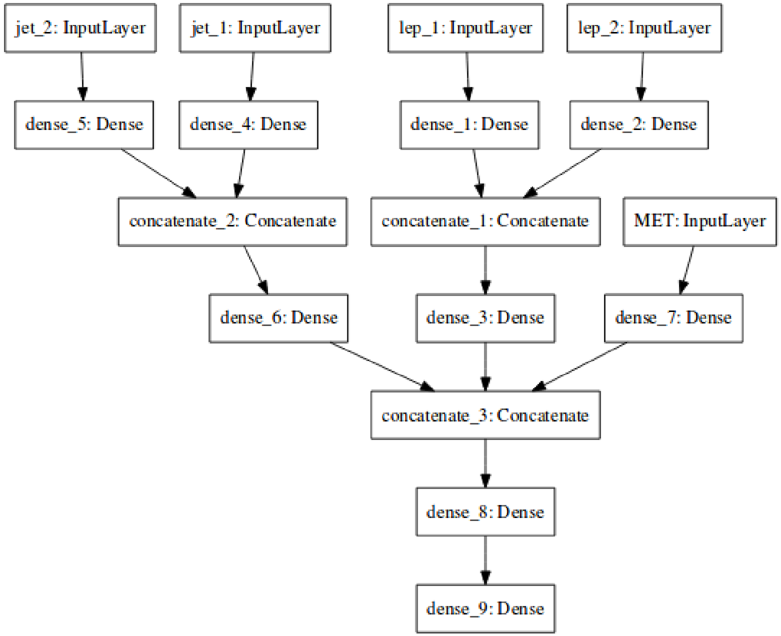


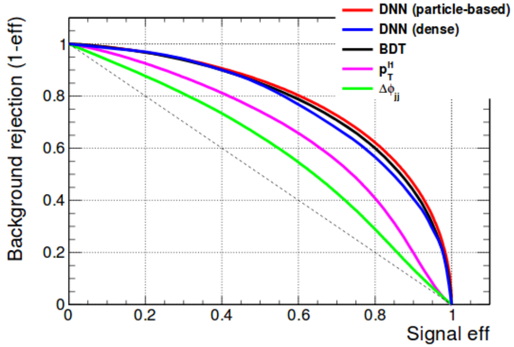
矢量玻色子散射在当前LHC实验上是十分有意义的，因为它对Higgs幺正性很敏感。

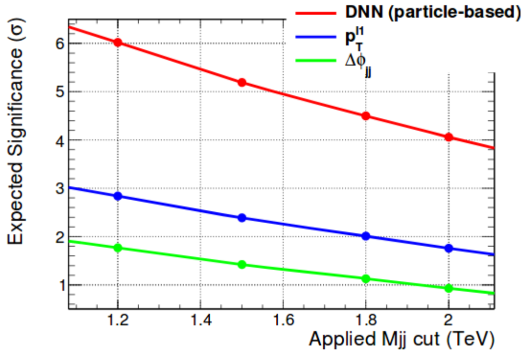
Higgs幺正性意味着VBS的反应截面将被限制，而不会发散。

其中纵向散射对于高能新物理极为敏感，但是相对横向散射占比小，不易观测到

深度神经网络：矢量玻色子散射



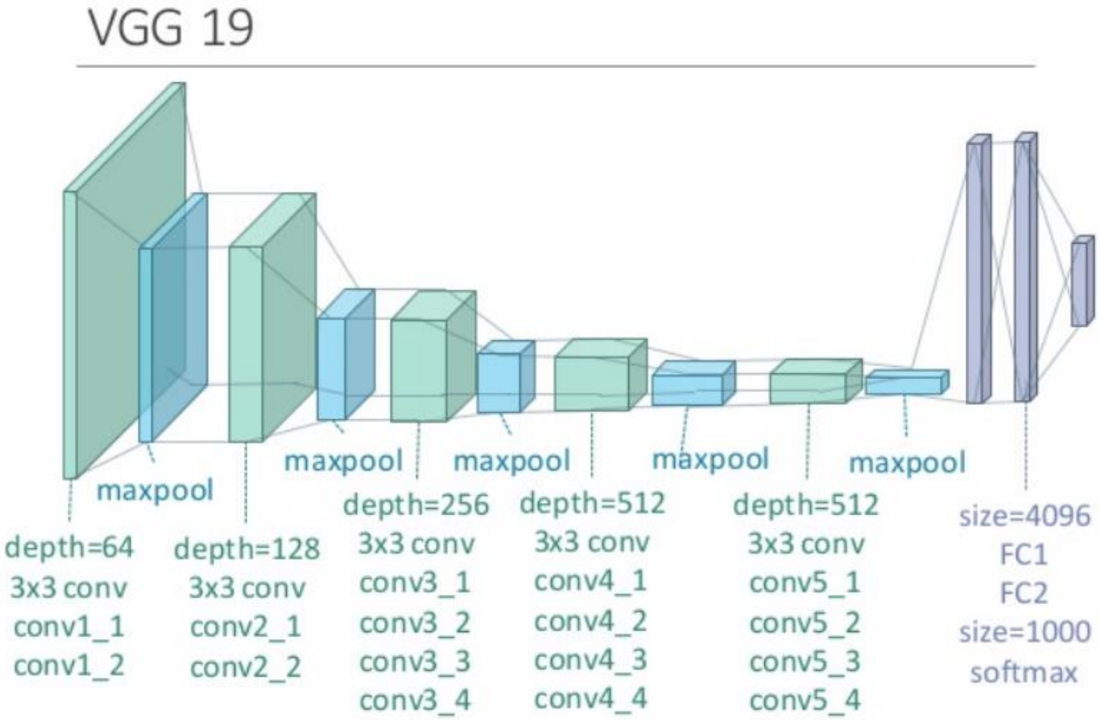




W、Z玻色子的纵向分量与Higgs密切相关；WW散射纵向分量的探测可以验证、揭示Higgs幺正机制。北大李俊昊利用深度学习，首次表明HL-LHC可以在5sigma水平观测到纵向散射。文章发表于Phys. Rev. D 99, 033004 (2019)

卷积神经网络CNN-概观

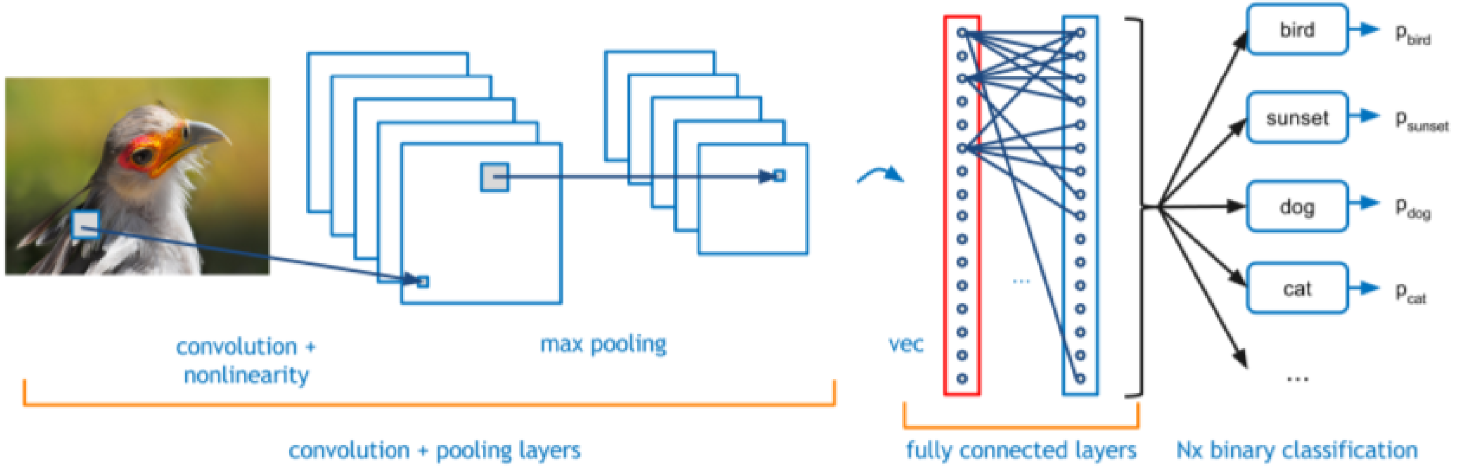
- *A full ConvNN is a sequence of Con2D+Pooling (+BatchNormalization+ Dropout) layers*
- *The Conv+Pooling layer reduces the 2D image representation*
- *The use of multiple filters on the image make the output grow on a third dimension*
- *Eventually, flattening occurs and the result is given to a dense layer*



卷积神经网络CNN-卷积层

- *Special architectures read the raw information (e.g., images) and convert them into "smart variables" (high-level features) to accomplish the task*

- *Typical example: convolutional neural networks for image processing & computing vision*



卷积神经网络CNN- 卷积

- *The main ingredient of ConvNN is a filter, a k x k’ matrix of weights*
- *The filter scans the image and performs a scalar product of each image patch*
- *This results into a new matrix of values, with different dimensionality*

0	3	5	6	2	4	5
7	4	7	3	6	3	4
9	1	2	1	9	6	0
9	2	1	1	7	3	5
8	0	4	7	6	8	0
8	3	4	5	5	3	4
7	9	4	6	5	2	6

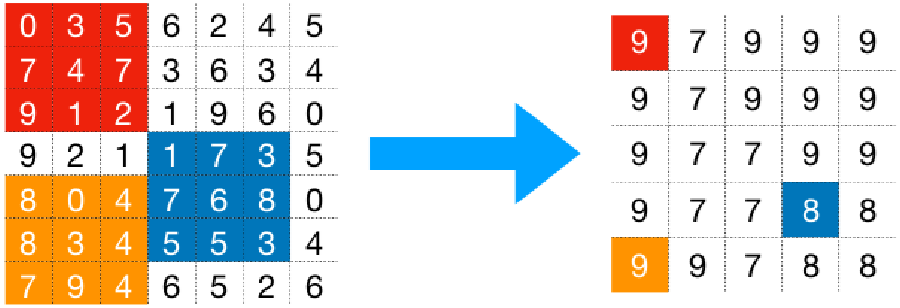
4	-1	4
-2	2	-5
3	1	-6

0x4 - 3x1 + 5x4 +
-7x2 + 4x2 - 7x5 +
9x3 + 1x1 - 2x6 = -8

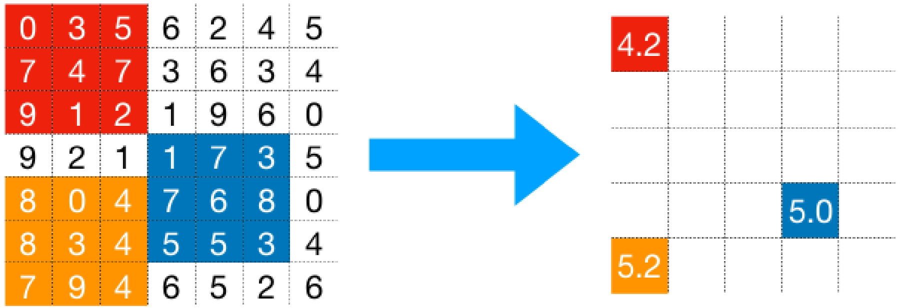
-8				

卷积神经网络CNN- 池化

- *MaxPooling: Given an image and a filter of size k x k', scans the image and replaces each k x k' patch with its maximum*



- *AveragePooling: Given an image and a filter of size k x k', scans the image and replaces each k x k' patch with its average*



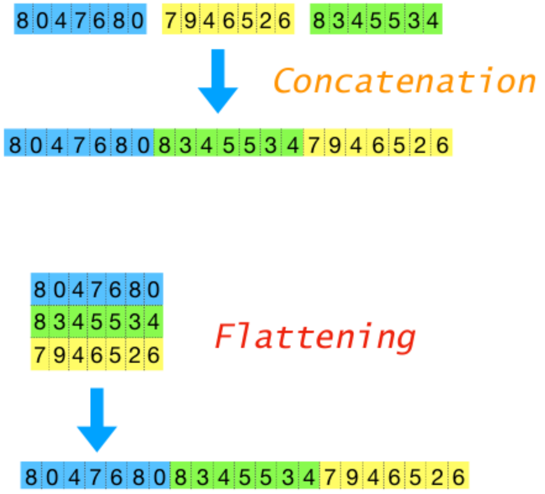
- …

卷积神经网络CNN- Padding，Flattening, Inception

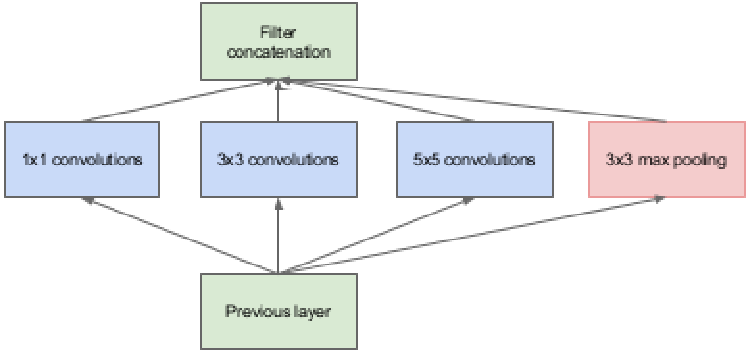
- *When the filter arrived at the edge, it might exceeds it (if n/k is not an integer)*
- *In this case, a padding rule needs to be specified*
  - *Same: repeat the values at the boundary*
  - *Zero: fill the extra columns with zeros*

0	3	5	6	2	4	4
7	4	7	3	6	3	3
9	1	2	1	9	6	6
9	2	1	1	7	3	3
8	0	4	7	6	8	8
8	3	4	5	5	3	3
8	3	4	5	5	3	3

0	3	5	6	2	4	0
7	4	7	3	6	3	0
9	1	2	1	9	6	0
9	2	1	1	7	3	0
8	0	4	7	6	8	0
8	3	4	5	5		
0	0	0	0	0		



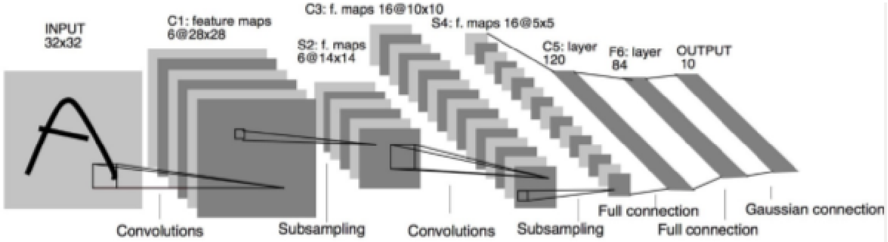
Inception:
Several conv layers, with different filter size, process the same inputs



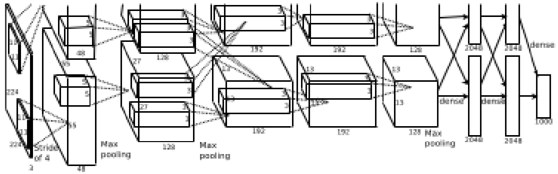
(a) Inception module, naïve version

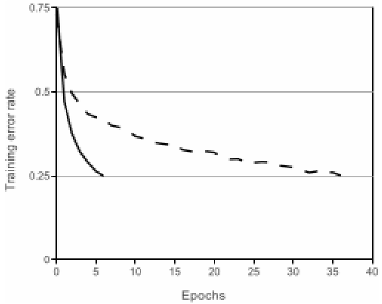
卷积神经网络CNN-历史

- *LeNet (1990s): the very first ConvNN, designer for digit recognition (ZIP codes)*

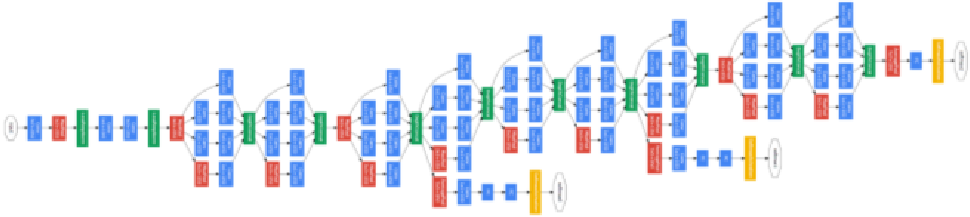


- *AlexNet (2012): the first big ConvNN (60M parameters, 650K neurtons), setting the state of the art: trained on GPUs, using ReLU and Dropout*

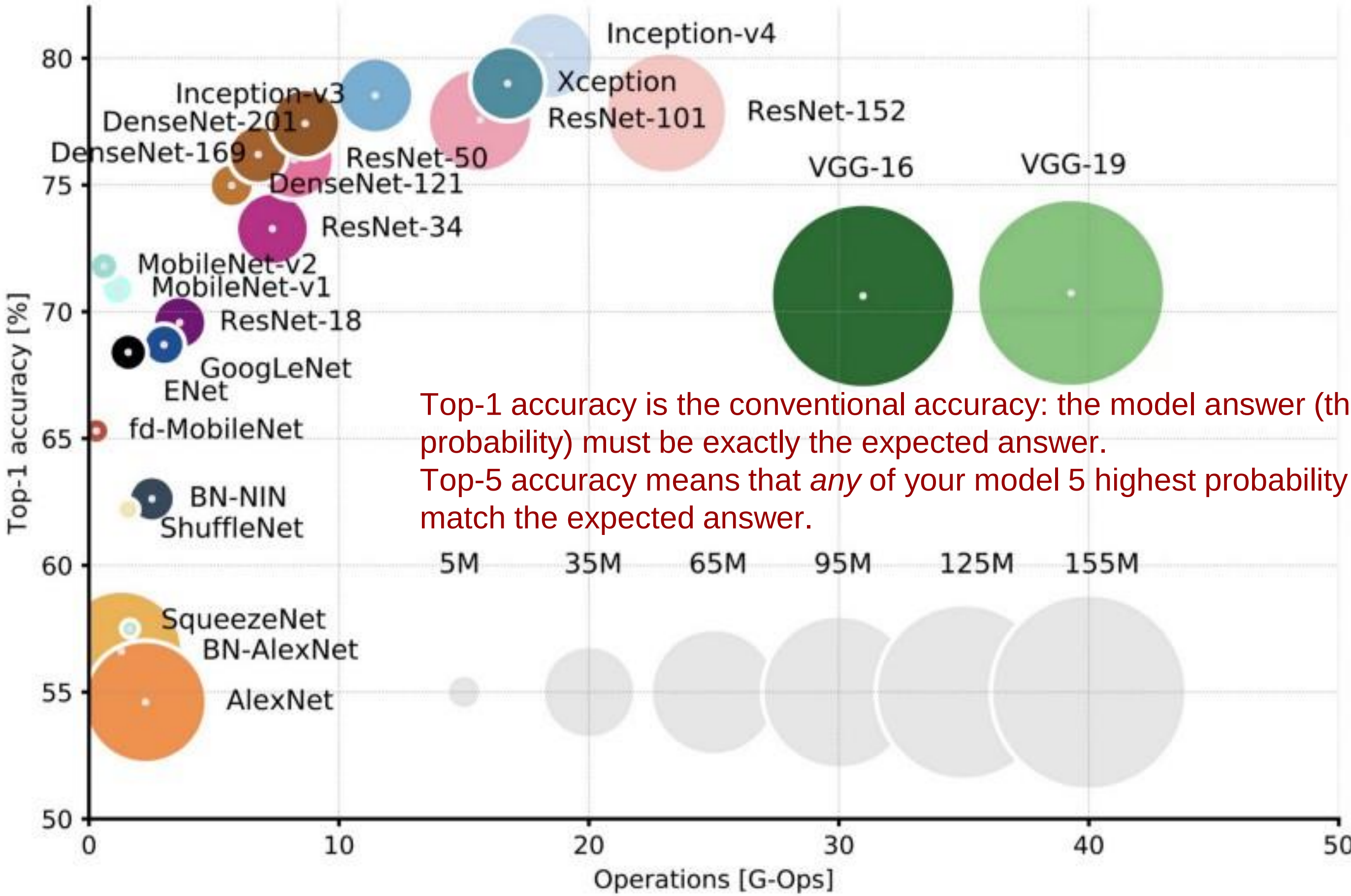




- *GoogleNet (2014): built on AlexNet, introduced an inception model to reduce e the number of parameters*

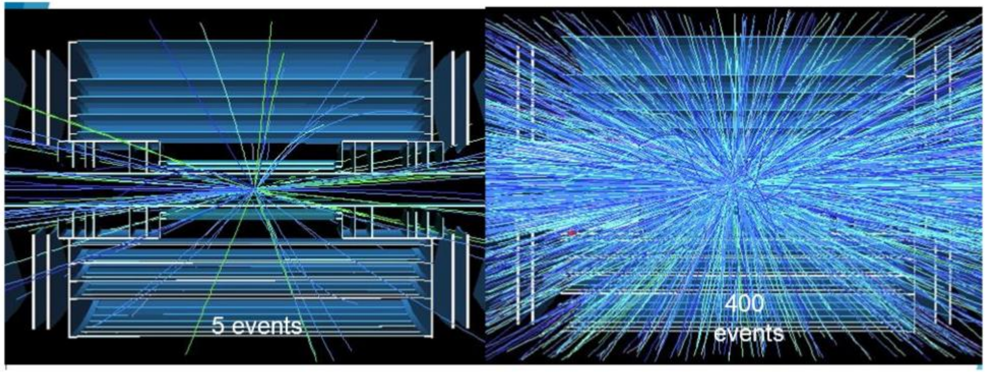


卷积神经网络CNN-历史

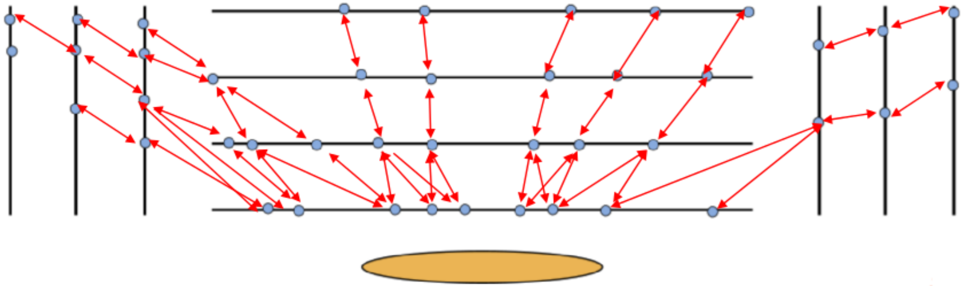


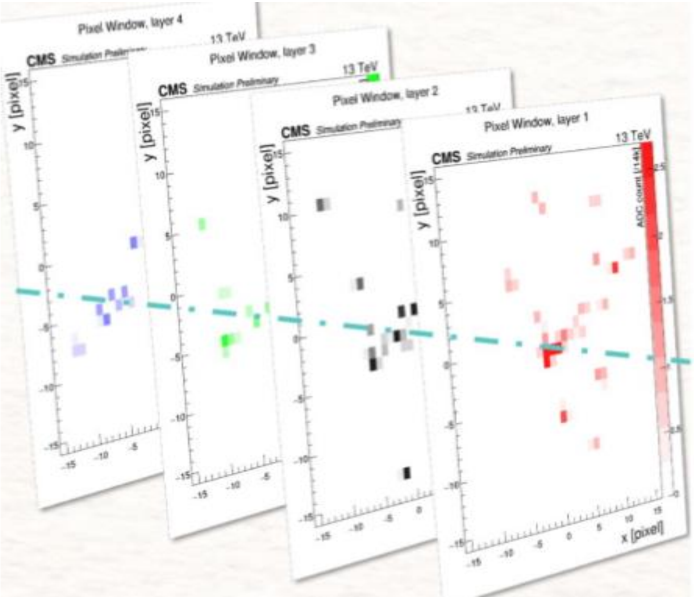
Top-1 accuracy is the conventional accuracy: the model answer (the one with highest probability) must be exactly the expected answer.
Top-5 accuracy means that *any* of your model 5 highest probability answers must match the expected answer.

CNN应用： 径迹重建



嘈杂环境中重建带电粒子径迹，具有挑战性：误组合，假种子

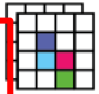


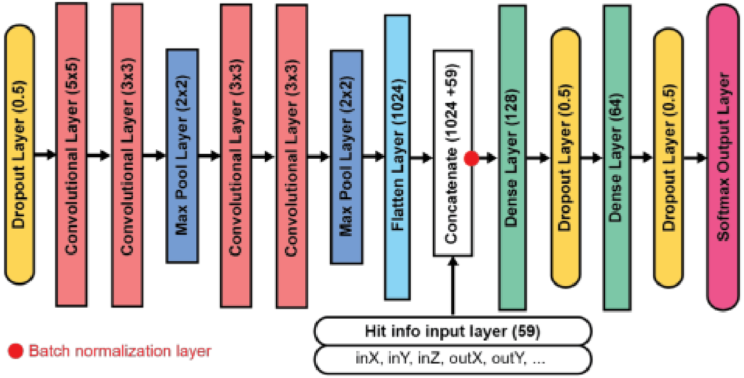


CNN应用： 径迹重建

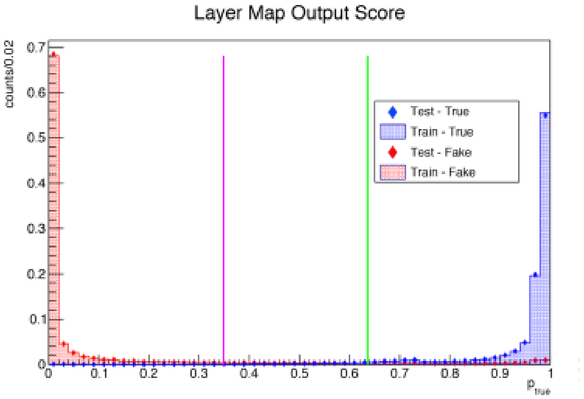
PixelSeed ConvNN

- *The final model uses two sets of inputs:*
  - *the hit images*
  - *a set of expert features (e.g., position of the hits in the detector) to help the learning process*





- *The trained model shows a good separation of true vs fake seeds*



- ***One can reduce the fake rate by one order of magnitude with a few % loss in efficiency***

```
Efficiency (tpr) @ fake rejection
tpr @ rej 50%: 0.998996700259
tpr @ rej 75%: 0.990524391331
tpr @ rej 90%: 0.922210826719
tpr @ rej 99%: 0.338669401587
```

卷积神经网络CNN： 喷注标记

https://indico.cern.ch/event/783781/contributions/3389493/attachments/1832744/3001915/Deep_Heavy_Resonance_Tagging_HEP2019_Kontaxakis.pdf

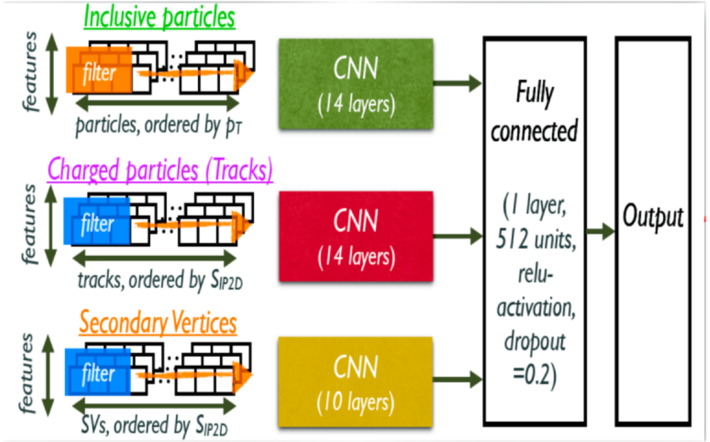
CMS实验的DeepAK8标记技术：

北大组贡献于Scale-Factor, Mass Decorrelation Tagger

DeepAK8

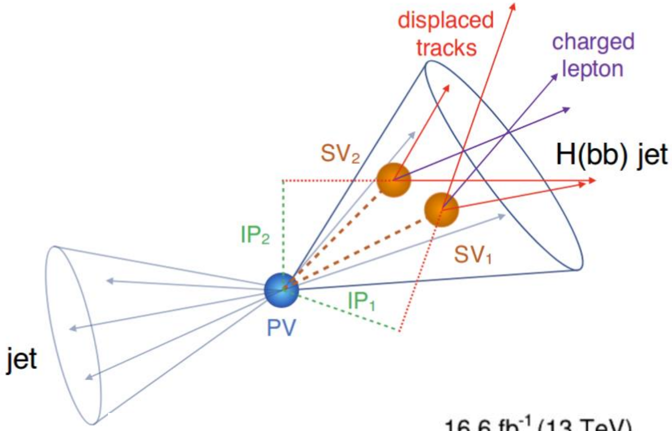
CMS-DP-2017-049
10.1007/JHEP10(2017)005

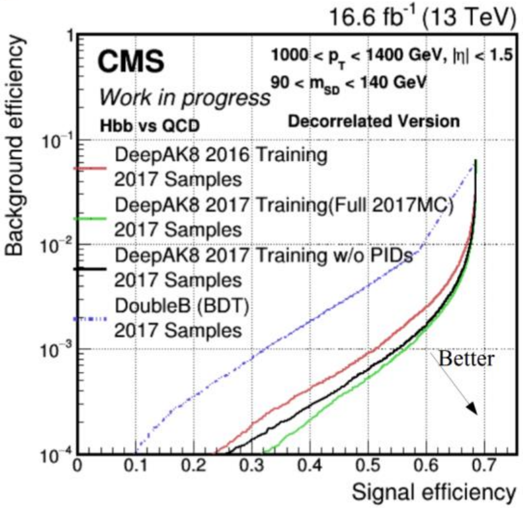
- Deep AK8 takes advantage of this additional information
  - Includes particle and detector-level quantities (tracking, vertex formation)
  - Individual jet constituents as inputs
- Uses convolutional NNs to take advantage of nearby correlations



Many output categories!

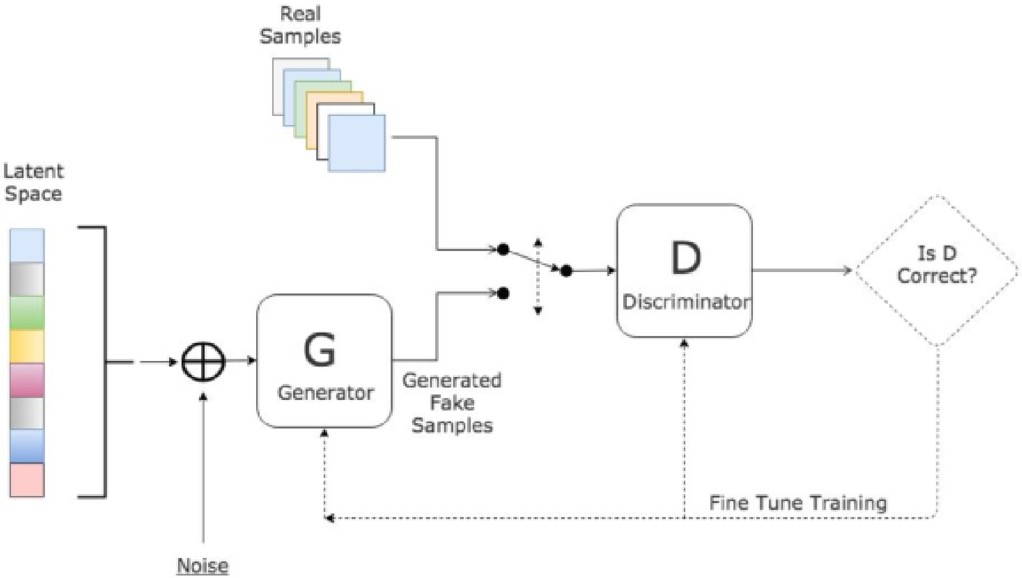
Category	Label
Higgs	H (bb)
	H (cc)
	H (VV*→qqqq)
Top	top (bcq)
	top (bqq)
	top (bc)
	top (bq)
W	W (cq)
	W (qq)
Z	Z (bb)
	Z (cc)
	Z (qq)
QCD	QCD (bb)
	QCD (cc)
	QCD (b)
	QCD (c)
	QCD (others)





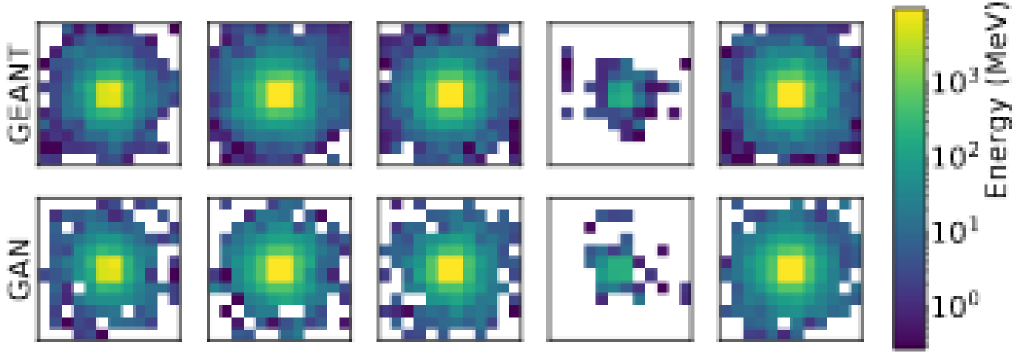
Generative Adversarial Network (GAN)

- *Two networks trained against each other*
  - *Generator: create images (from noise, other images, etc)*
  - *Discriminator: tries to spot which image comes from the generator and which is genuine*

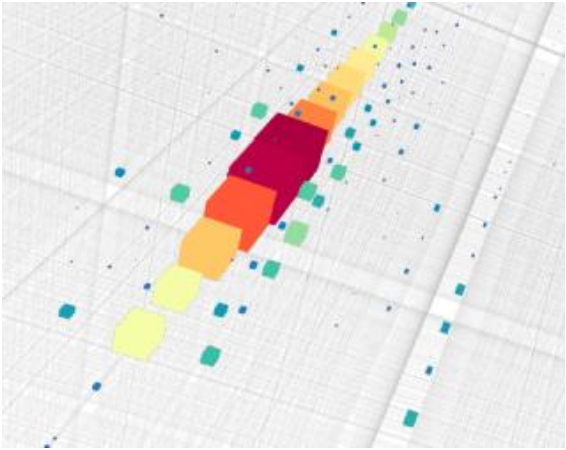


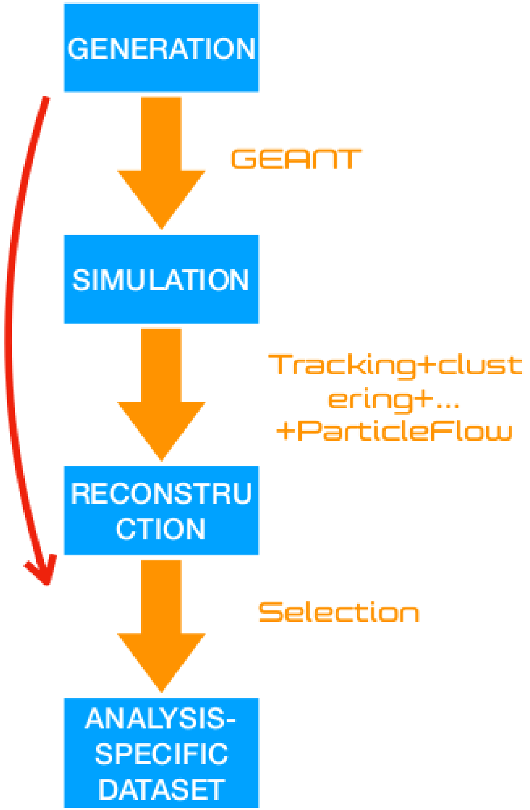
- ***Loss function to minimise: Loss(Gen)-Loss(Disc)***
  - ***Better discriminator -> bigger loss***
  - ***Better generator -> smaller loss***
- ***Trying to full the discriminatore, generatore learns how to create more realistic images***

GAN: 快速模拟、 重建



GEANT
GAN
10^3
10^2
10^1
10^0
Energy (MeV)

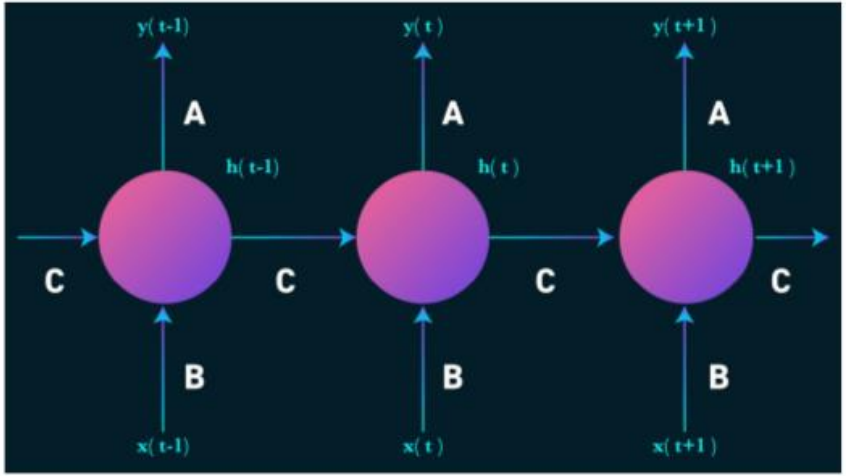




GENERATION
GEANT
SIMULATION
Tracking+clustering+…
+ParticleFlow
RECONSTRUCTION
Selection
ANALYSIS-SPECIFIC DATASET

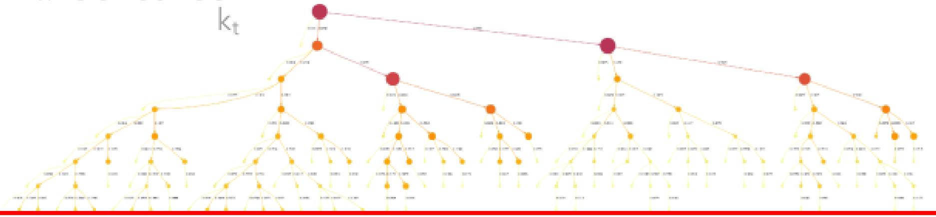
循环神经网络RNN

- *Recurrent architectures are designed to process sequences of data*
- *Then idea is to have information flowing in the network while the sequence is sequentially processed*
- *Through this idea, recurrent networks mimic memory persistence*
- *Advantages*
  - *the input is not fixed-sized*

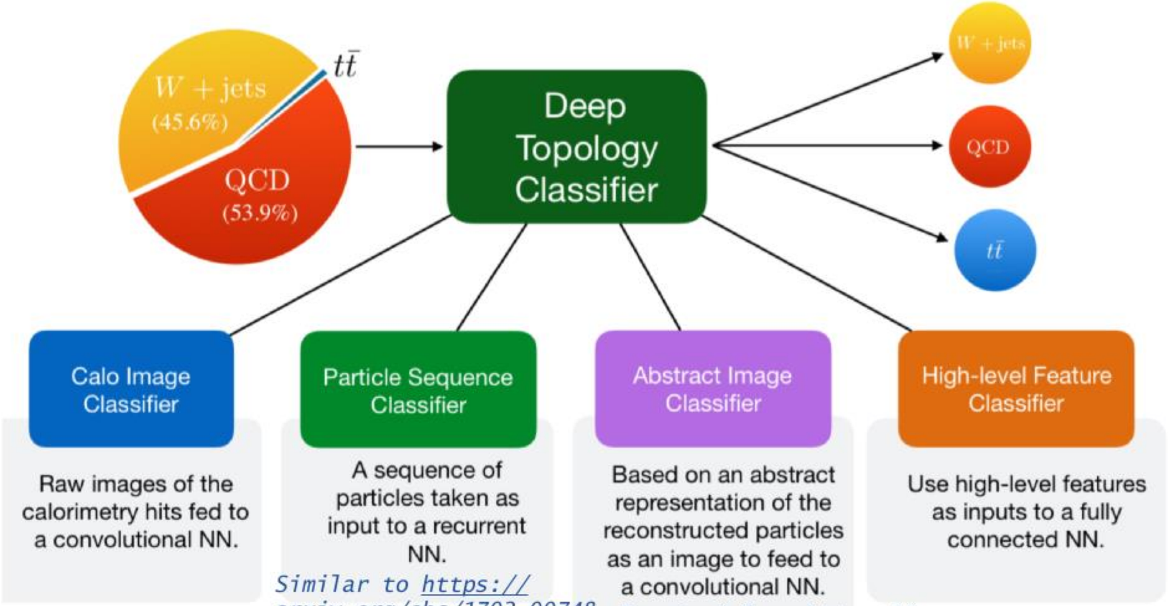


循环神经网络RNN：事例分类

- *particles as words in a sentence*
- *QCD is the grammar*



four-momenta are like words and the clustering history of sequential recombination jet algorithms is like the parsing of a sentence.



Similar to https://arxiv.org/abs/1702.00748

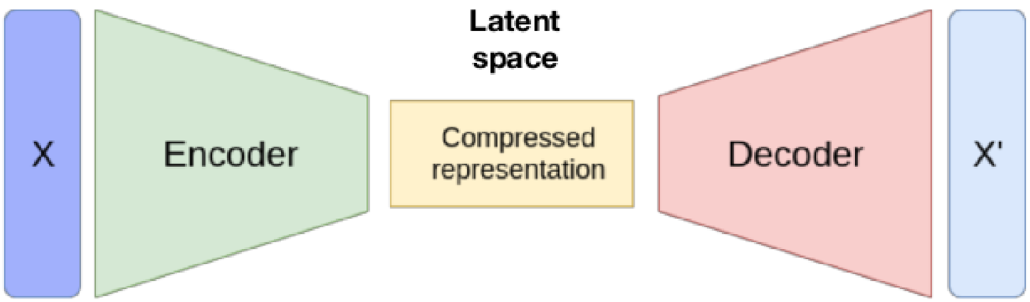
Inspired from https://arxiv.org/abs/1708.07034

自编码Autoencoders

- *Autoencoders are networks with a typical "bottleneck" structure, with a symmetric structure around it*
  - *They go from $\mathbb{R}^n \rightarrow \mathbb{R}^n$*
  - *They are used to learn the identity function as $f^{-1}(f(x))$*

where $f: \mathbb{R}^n \rightarrow \mathbb{R}^k$ and $f^{-1}: \mathbb{R}^k \rightarrow \mathbb{R}^n$

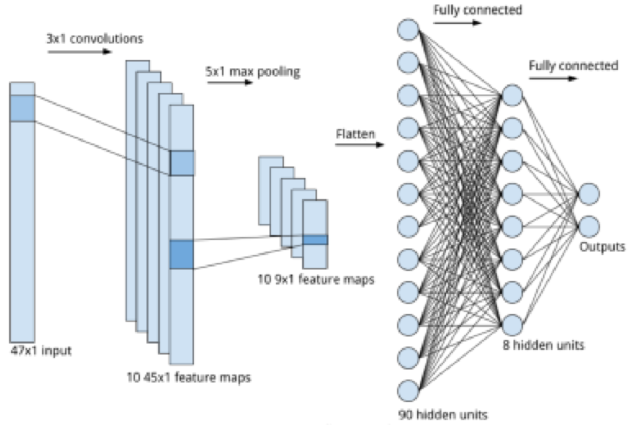
- *Autoencoders are essential tools for unsupervised studies*

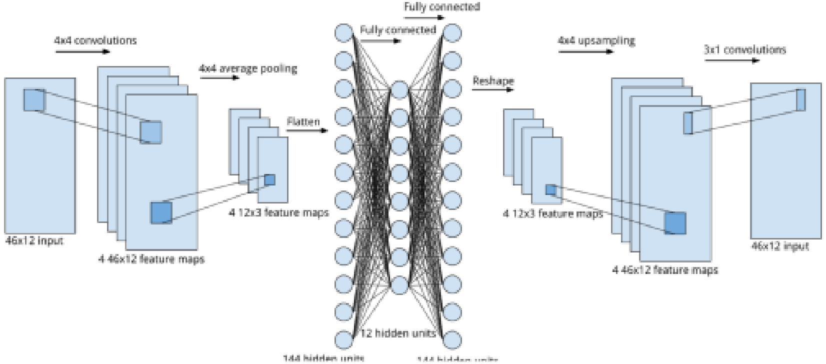


自编码： 数据监控

- *Given the nature of these data, ConvNN are a natural analysis tool. Two approaches pursued*
  - *Classify good vs bad data. Works if failure mode is known*
  - *Use autoencoders to assess data "typicality". Generalises to unknown failure modes*

A. Pol et al., to appear soon



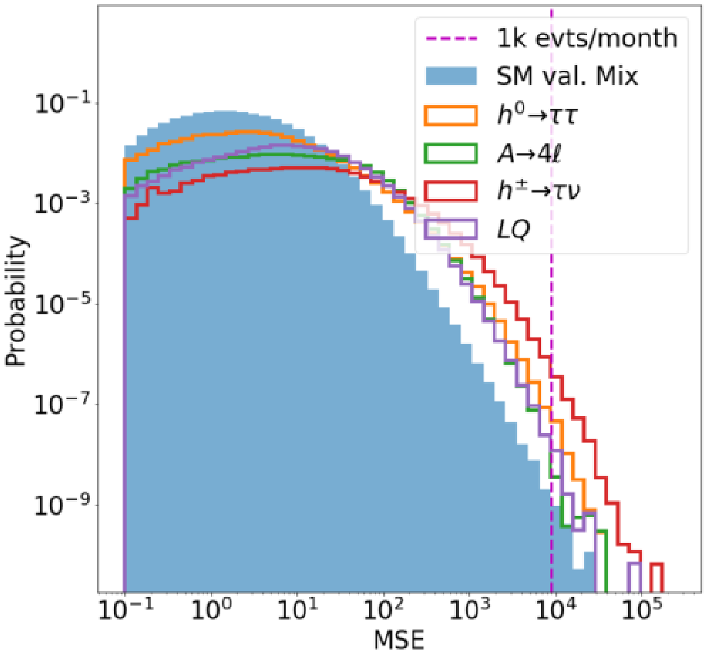


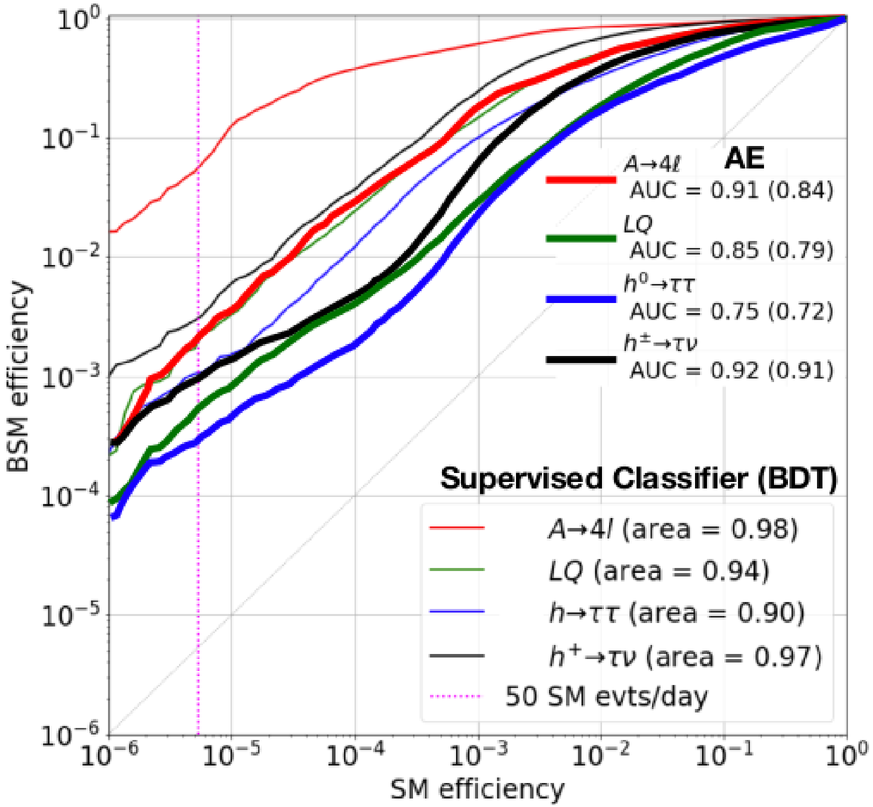
[Pol, G. Cerminara, C. Germain, MP and A. Seth arXiv:1808.00911]

自编码：自动寻找新物理

- Train on standard events
- Run autoencoder on new events
- Consider as anomalous
  all events with loss > threshold

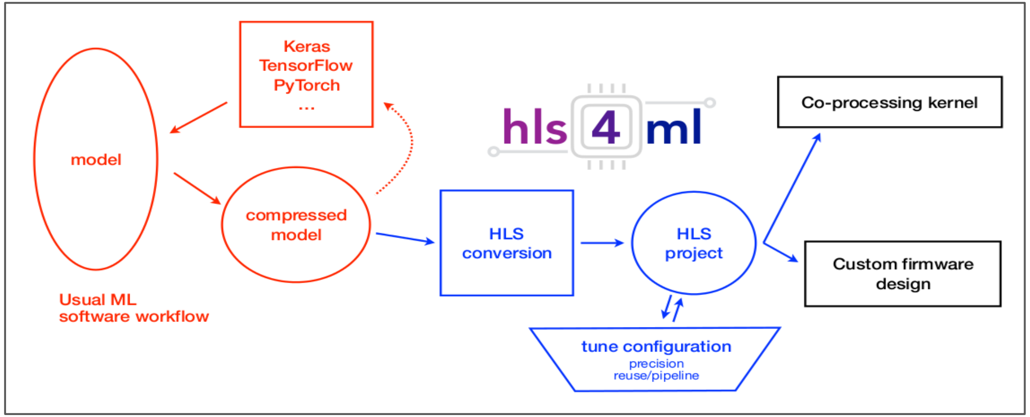
Worse than Supervised but results encouraging





硬件： FPGA

Map DNN nicely into an FPGA



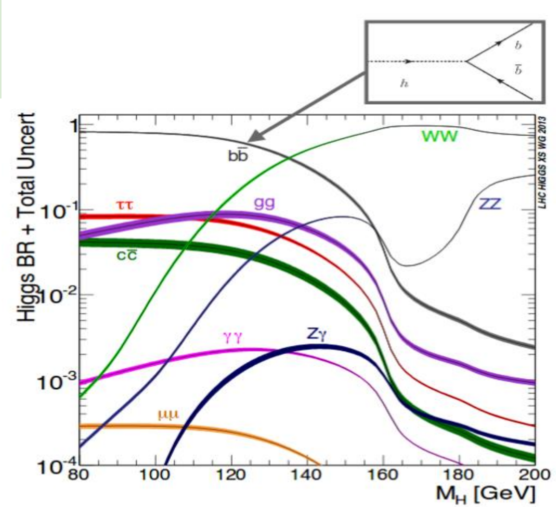
hls4ml

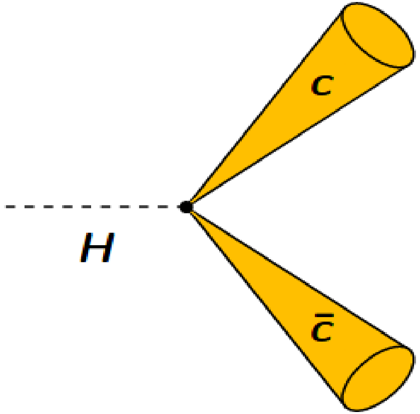
https://github.com/hls-fpga-machine-learning/hls4ml

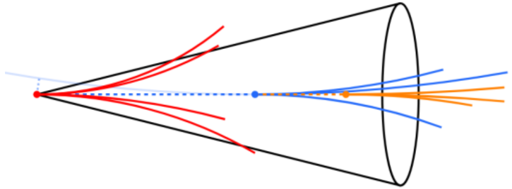
DOI 10.5281/zenodo.1204445

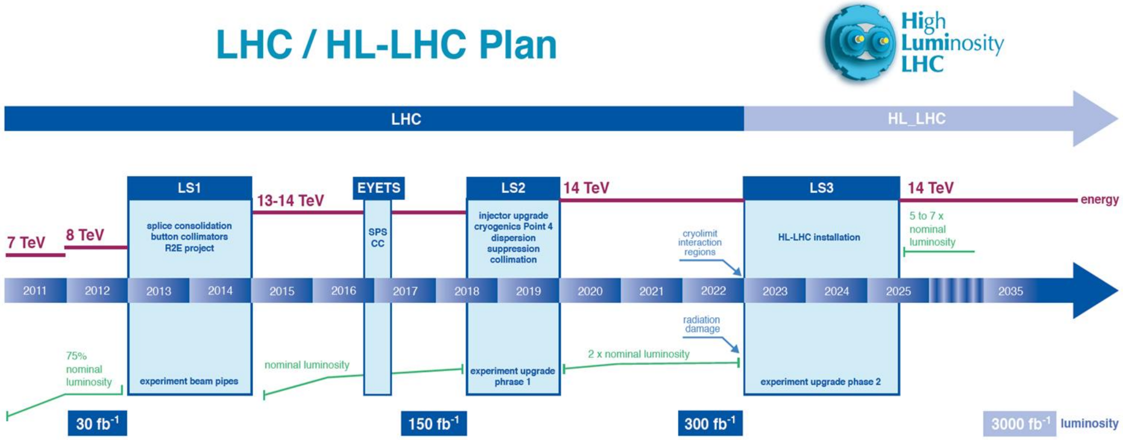
A package for machine learning inference in FPGAs. We create firmware implementations of machine learning algorithms using high level synthesis language (HLS). We translate traditional open-source machine learning package models into HLS that can be configured for your use-case!

未来：　Hcc



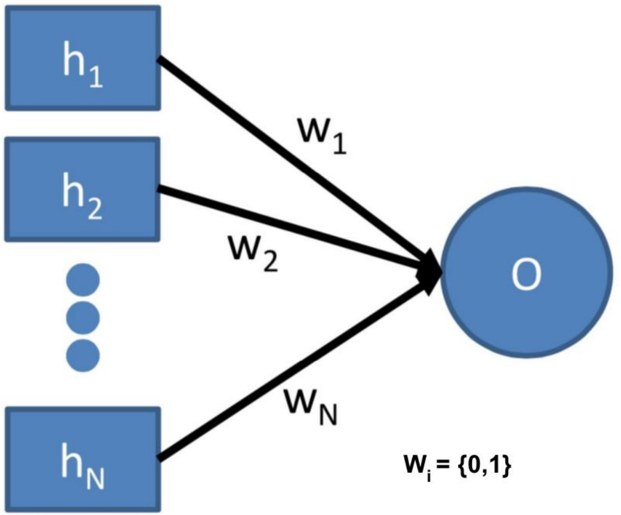






Higgs coupling with 2nd gen. fermion
Very challenging!
ATLAS gives upper limit ~ 100*prediction
Needs HL-LHC 3000fb-1, by ~2035!
Deep Learning will definitely help a lot!

未来： 量子计算



quantum annealing

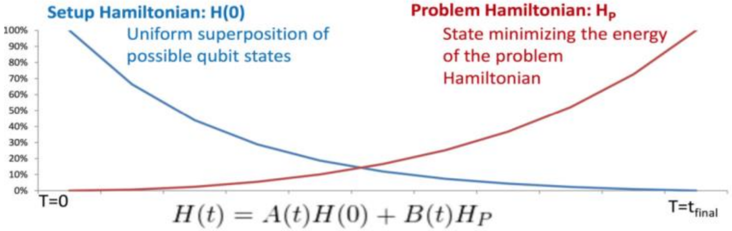
$H(0) = \sum_i \sigma_i^x$

$H_p = \sum_{i,j} J_{ij} s_i s_j + \sum_i h_i s_i$

H_p is effectively $\delta(\vec{w}) \propto \sum_{i,j} C_{ij} w_i w_j + \sum_i (\lambda - 2C_{iy}) w_i$

σ_i^x has a ground state of proportional to $|0\rangle + |1\rangle$

H(0) has no interactions, so cools to ground state quickly, and the total ground state is an equal superposition over all bitstrings



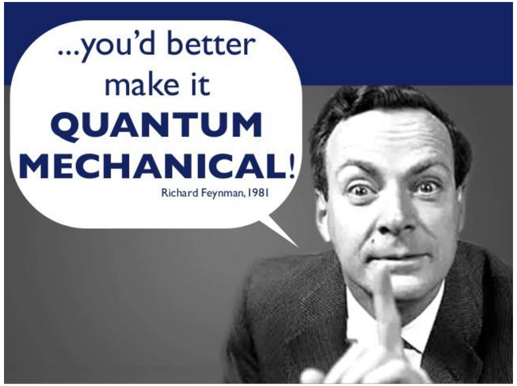

nature
International journal of science

Letter | Published: 18 October 2017

Solving a Higgs optimization problem with quantum annealing for machine learning

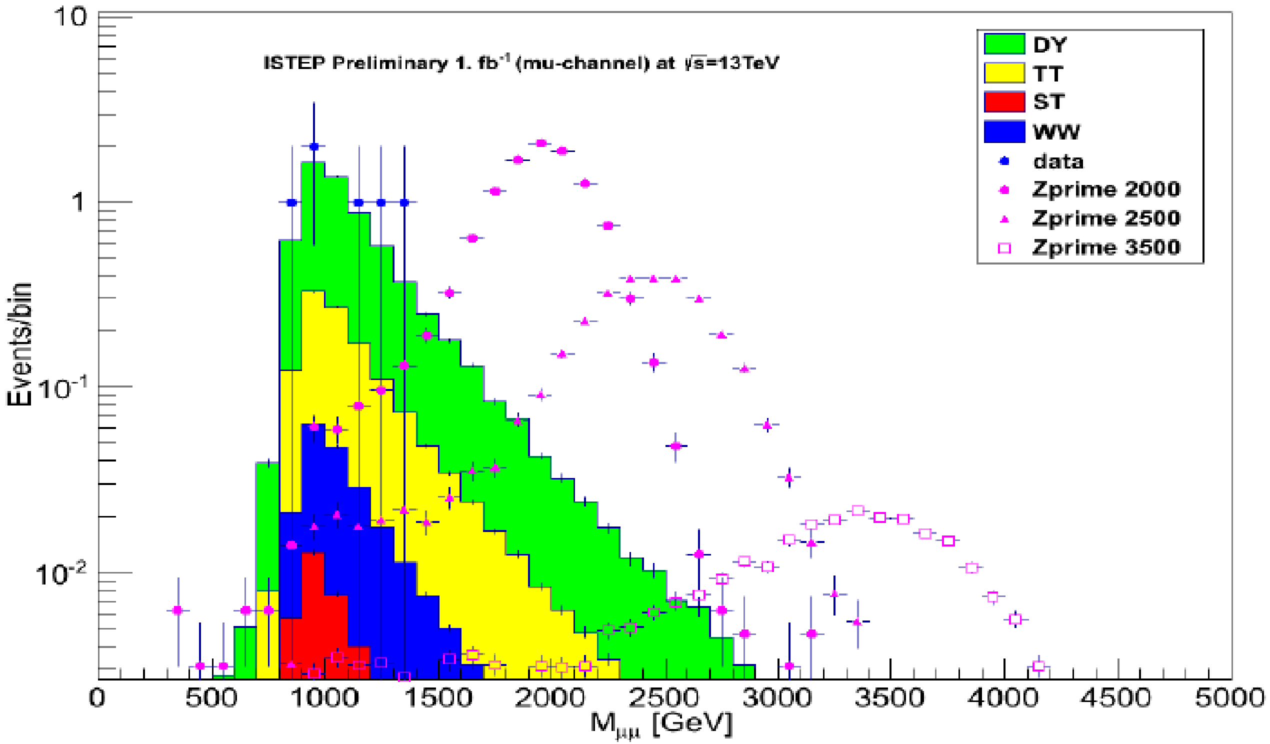
Alex Mott, Joshua Job, Jean-Roch Vlimant, Daniel Lidar & Maria Spiropulu

Nature **550**, 375–379 (19 October 2017) | Download Citation

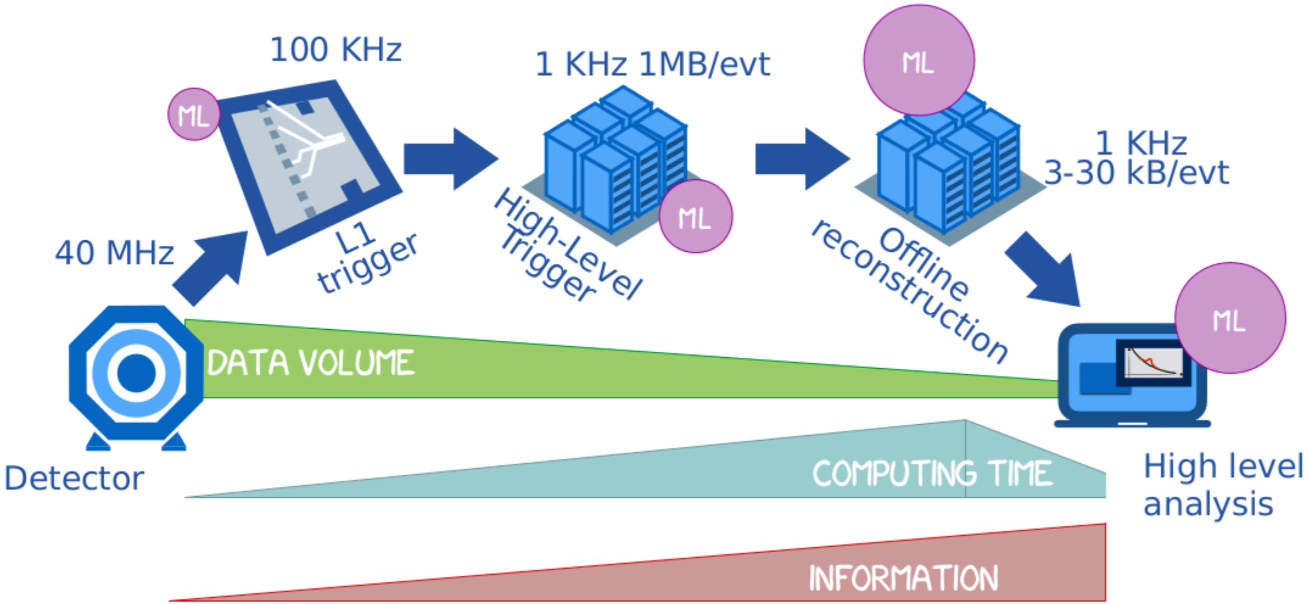


Tutorial： BDT

Z’ search: https://pan.baidu.com/s/1b54D2m



Summary



Deeper and Deeper in HEP