

CMS DETECTOR

Total weight : 14,000 tonnes
Overall diameter : 15.0 m
Overall length : 28.7 m
Magnetic field : 3.8 T

STEEL RETURN YOKE
12,500 tonnes

SILICON TRACKERS
Pixel (100x150 μm) $\sim 16\text{m}^2 \sim 66\text{M}$ channels
Microstrips (80x180 μm) $\sim 200\text{m}^2 \sim 9.6\text{M}$ channels

SUPERCONDUCTING SOLENOID
Niobium titanium coil carrying $\sim 18,000\text{A}$

MUON CHAMBERS
Barrel: 250 Drift Tube, 480 Resistive Plate Chambers
Endcaps: 468 Cathode Strip, 432 Resistive Plate Chambers

PRE-SHOWER
Silicon strip detectors $\sim 2,000$ channels
CALORIMETER
Steel + Quartz fibres $\sim 2,000$ Channels

Reti Neurali Convoluzionali per il seeding nella ricostruzione delle tracce a CMS

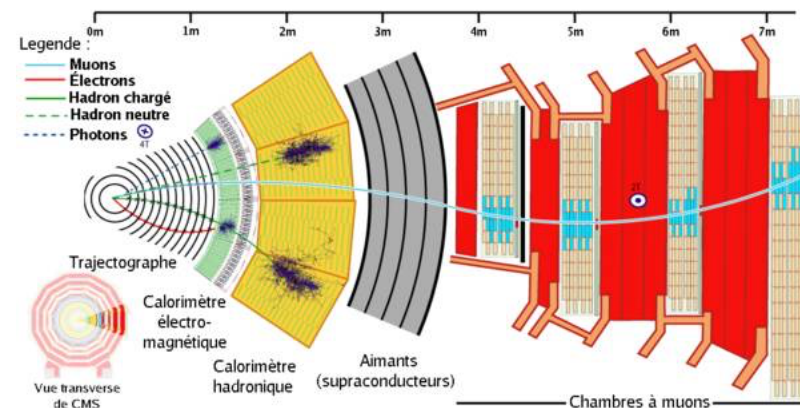
Adriano Di Florio
a nome della collaborazione CMS

CRYSTAL
ELECTROMAGNETIC
CALORIMETER (ECAL)
 $\sim 76,000$ scintillating PbWO_4 crystals

HADRON CALORIMETER (HCAL)
Brass + Plastic scintillator $\sim 7,000$ channels

► Il **Compact Muon Solenoid (CMS)** è uno dei esperimenti “general purpose” installati presso il Large Hadron Collider al CERN per rilevare e misurare l’energia, l’impulso e la direzione di volo delle particelle prodotte nelle collisioni adroniche (*e.s. $e, \mu, K, \pi \dots$*)

SOLO **1000 EVENTI/SECONDO** [1 MB/EVENTO] POSSONO ESSERE REGISTRATI SU DISCO (LIMITI DI I/O)



► SISTEMA DI SELEZIONE DEGLI EVENTI A DUE LIVELLI

TRIGGER L1

Calorimetri e detector di Muoni

- 40 MHz ingresso / 100 KHz uscita
- ~ 500 KB / evento
- Tempo di processamento $O(\mu s)$
- Ricostruzione locale
- Implementazione FPGA/Hardware



HIGH LEVEL TRIGGER (HLT)

Intero detector

- 100 KHz ingresso / 1 KHz uscita
- ~ 500 KB / evento
- Tempo procesamento $O(100 ms)$
- Ricostruzione globale (semplificata)
- Implementazione software su CPU

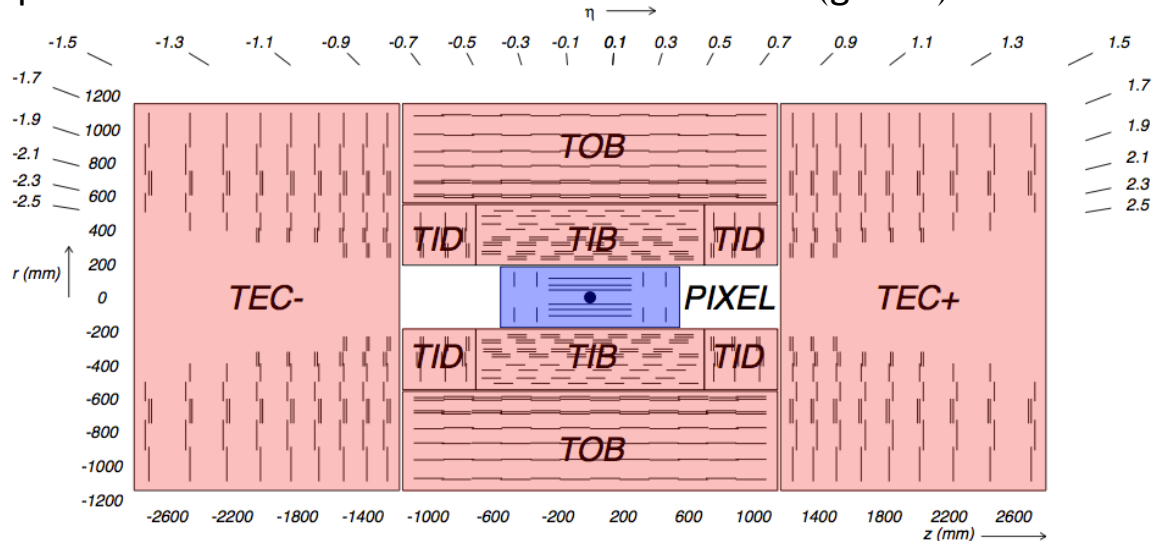


~26,000 XEON CORE

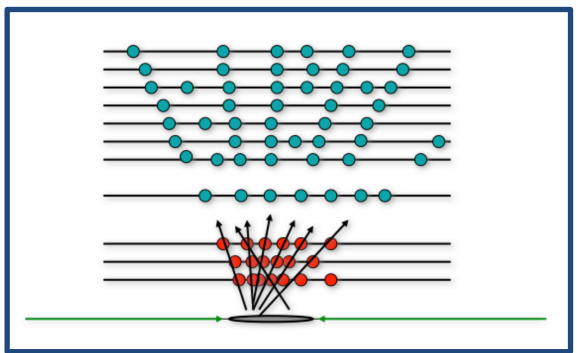
Ricostruzione delle tracce



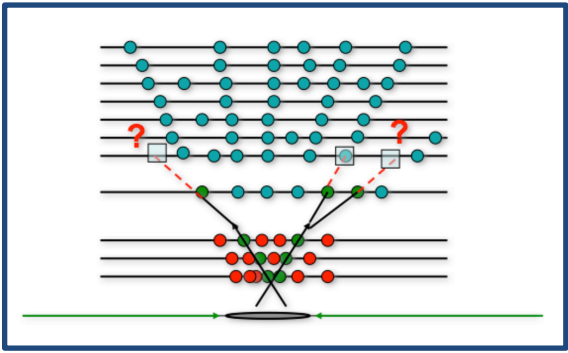
➤ L'algoritmo di tracciamento a CMS consiste in una **procedura iterativa** in cui le tracce sono ricostruite, con criteri di selezione **progressivamente meno stringenti**, a partire dai punti di interazione delle particelle cariche con i rivelatori del tracciatore (gli HIT).



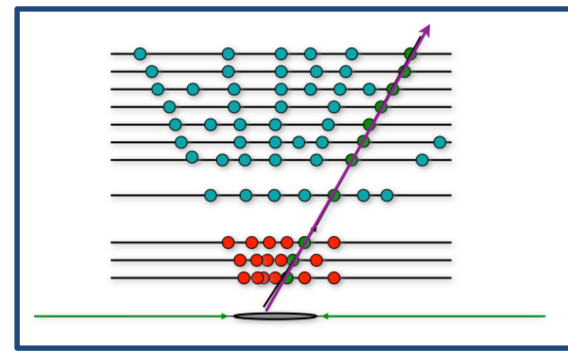
➤ RICOSTRUZIONE DELLE TRACCE A CMS



Seeding

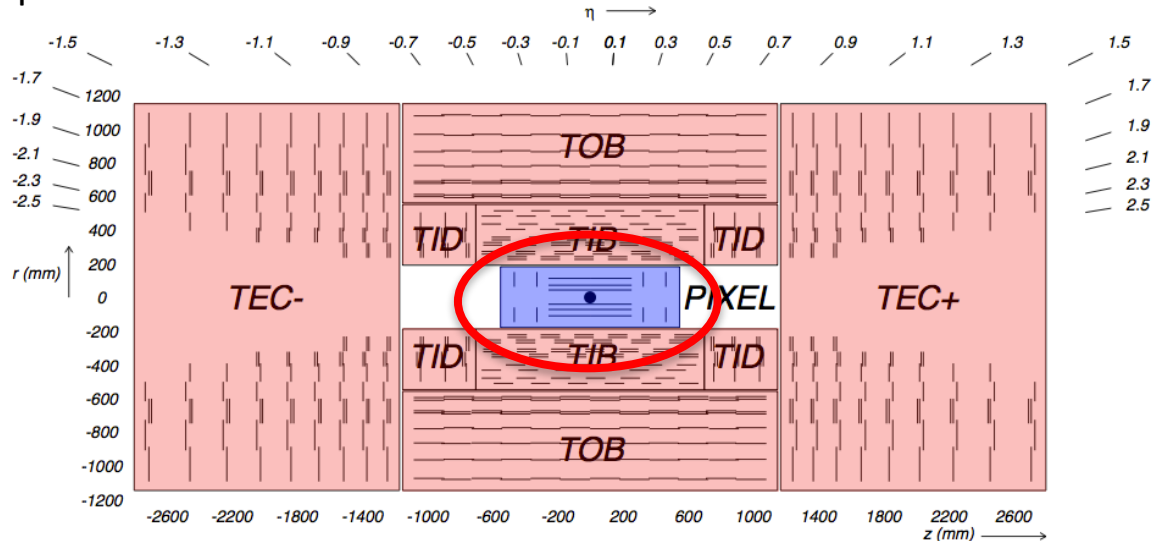


Costruzione delle tracce



Interpolazione tracce

➤ L'algoritmo di tracciamento a CMS consiste in una **procedura iterativa** in cui le tracce sono ricostruite, con criteri di selezione **progressivamente meno stringenti**, a partire dai punti di interazione delle particelle cariche con i rivelatori del tracciatore.



➤ RICOSTRUZIONE ONLINE (HLT)

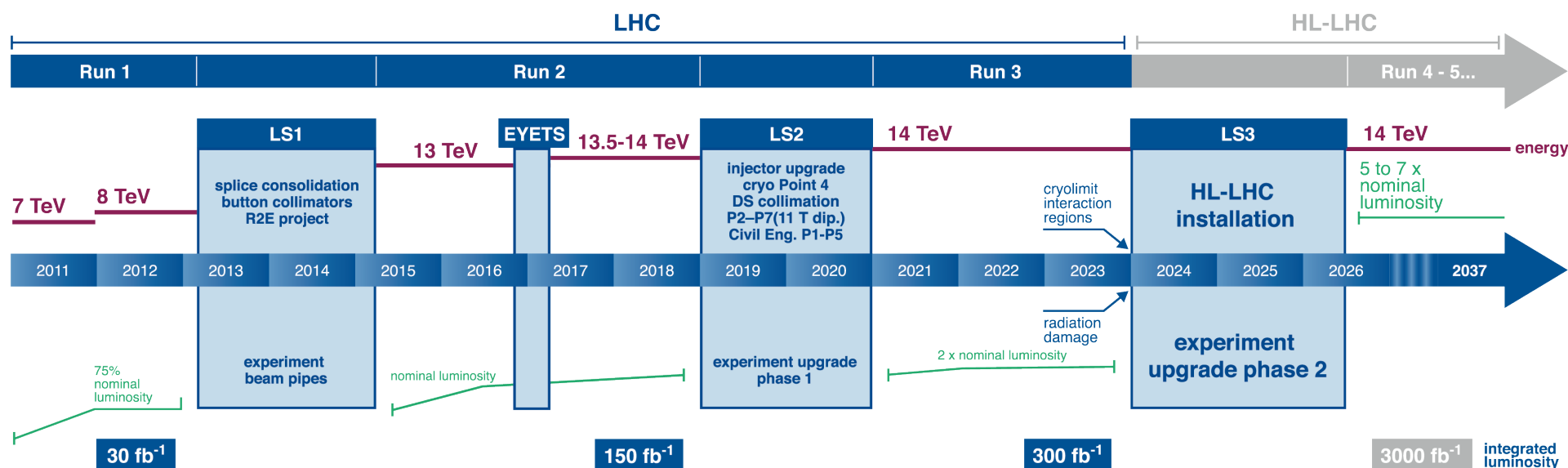
Deve sottostare a limiti di tempo $O(100 \text{ ms})$ ed è basata sulla ricostruzione delle tracce nel solo rivelatore a **pixel di silicio**.



A cosa andiamo incontro?



LHC / HL-LHC Plan



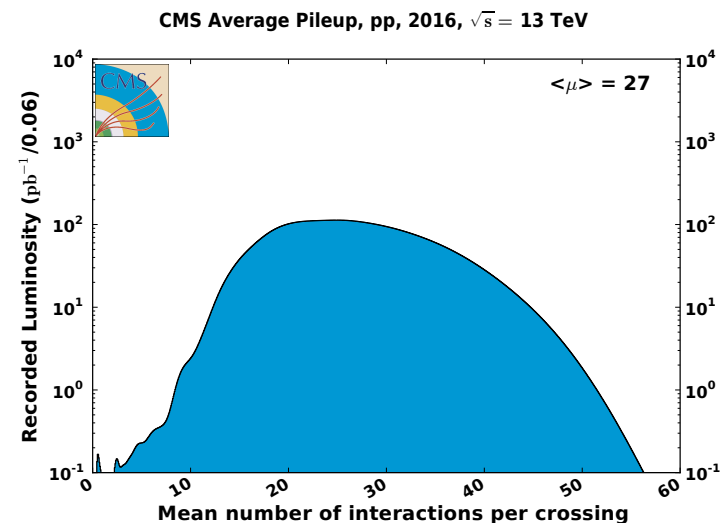
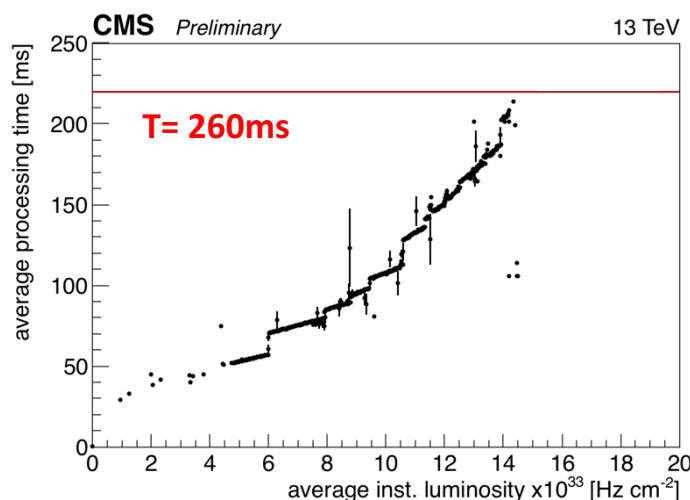
A cosa andiamo incontro?



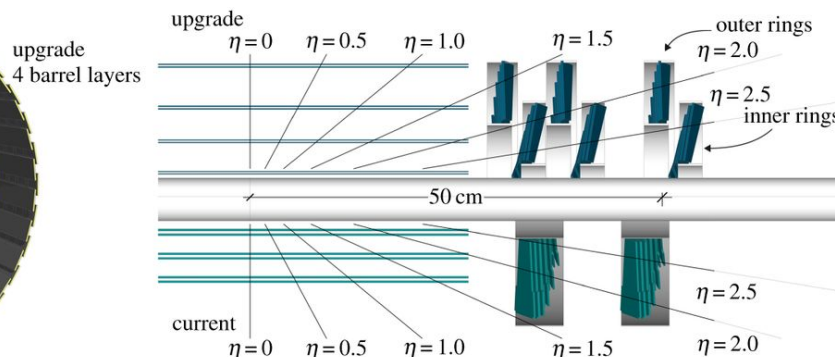
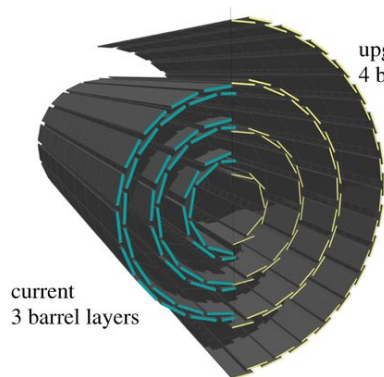
➤ AUMENTO DI LUMINOSITA' ISTANTANEA & PILE-UP

$$\mathcal{L} = 5 \cdot 10^{34} \text{ cm}^2 \text{ s}^{-1}$$

$$\langle PU \rangle \sim 200$$

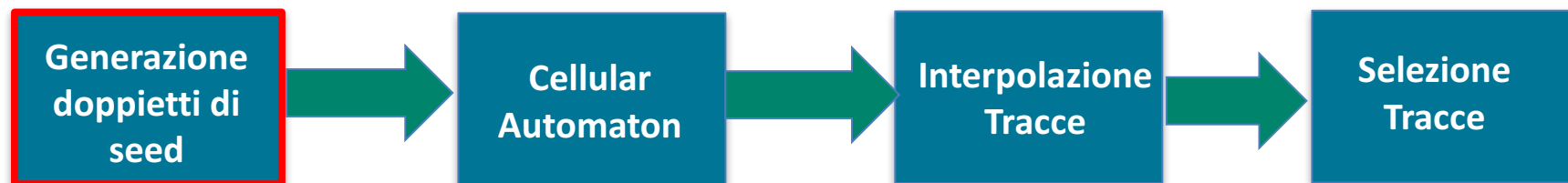


➤ STRUTTURA RIVELATORE PIÙ COMPLESSA (DAL 2017)



10 layers (6 endcap + 4 barrel)

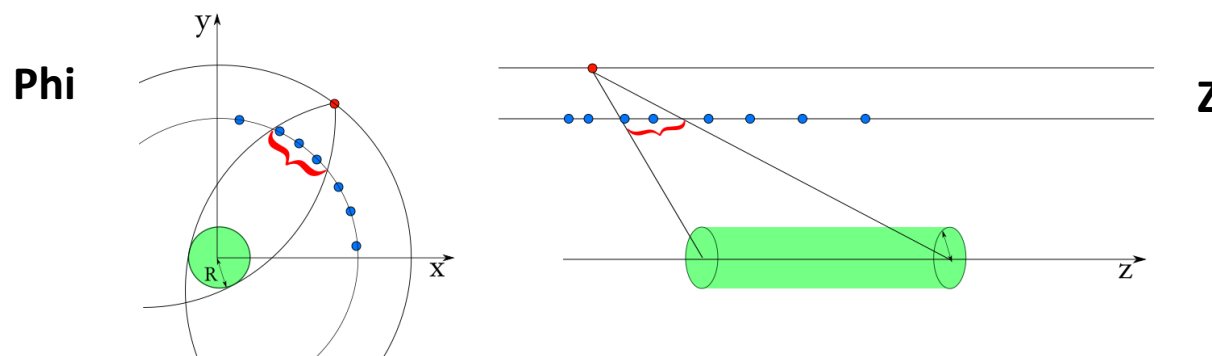
➤ Il **seeding** ad HLT parte dalla generazione delle coppie di hit o doppietti



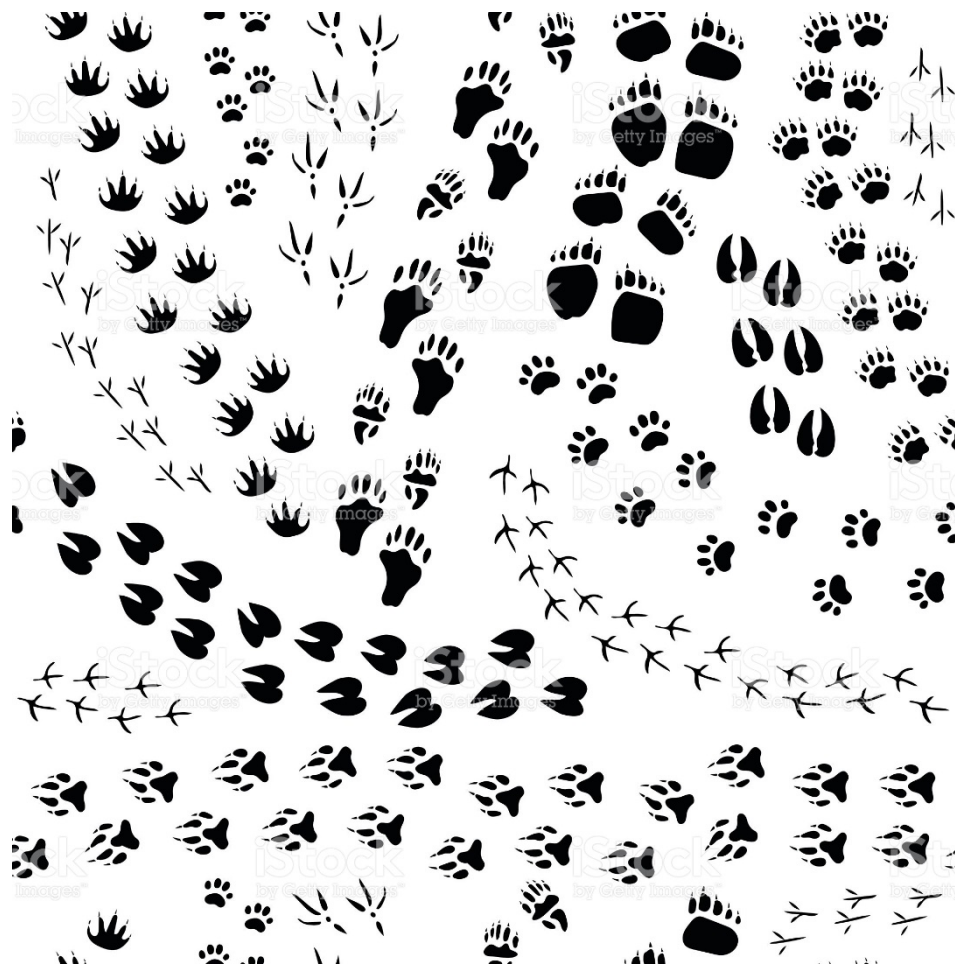
➤ Il numero di **doppietti** generati, a causa del fondo combinatorio, rappresenta un importante collo di bottiglia per i passaggi successivi della ricostruzione.

ES. Per un singolo evento simulato $t\bar{t}$ a $\sqrt{s} = 13\text{TeV}$ e con $\langle PU \rangle = 35$ vengono **$O(10^5)$ doppietti** con un rapporto **fondo/segnale** $\sim O(100)$ corrispondente a **$O(1000)$ doppietti veri**.

➤ Selezione dei doppietti basata **solo su considerazioni di compatibilità geometrica**.

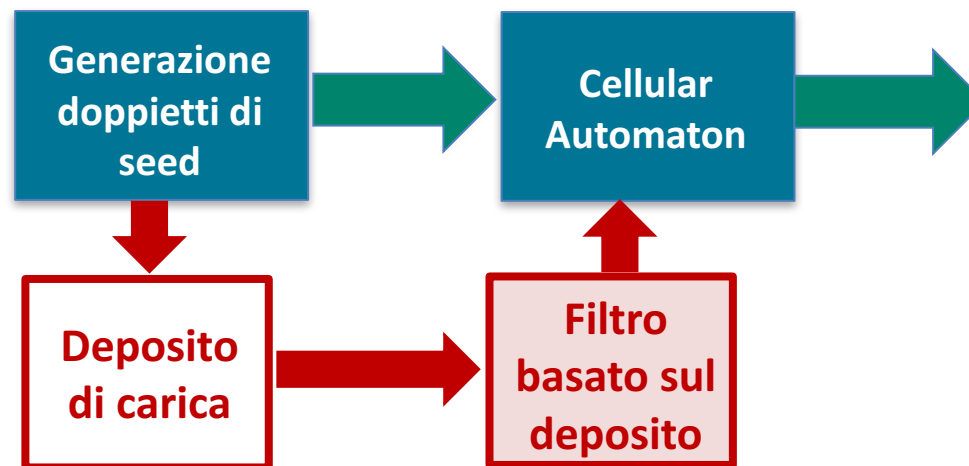
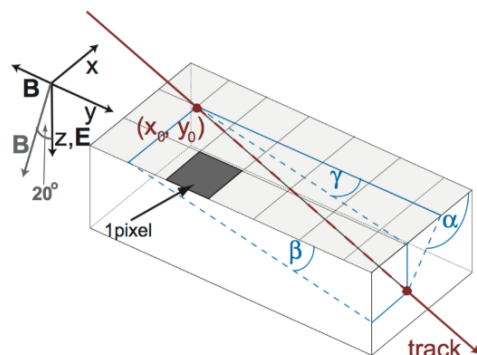


Ma ...

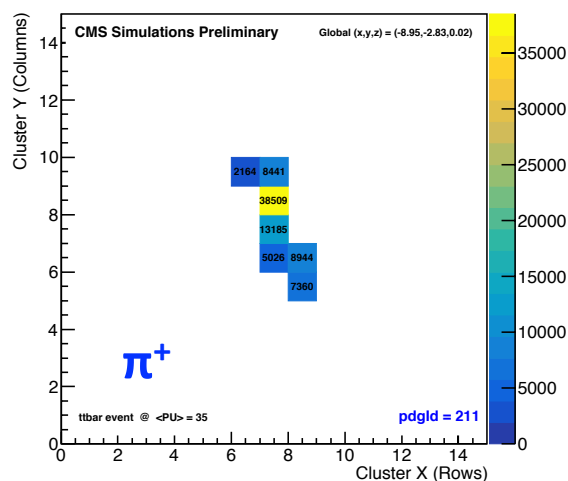


... ci sono altre impronte da seguire.

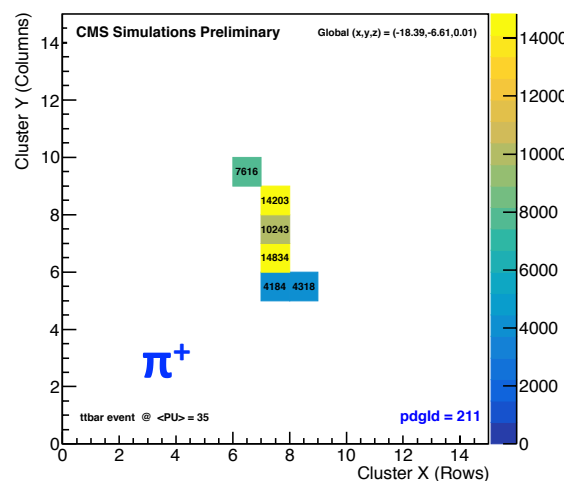
➤ Ogni doppietto è costituito da una coppia di hit sul rivelatore a **pixel di silicio** che non sono altro che una collezione di **pixel** (in 2D) **accesi o spenti**. Ad ogni pixel è associato un livello **ADC** proporzionale al deposito di carica rilasciato dal passaggio della particella carica.



BPix1 - Inner Hit



BPix2 - Outer Hit

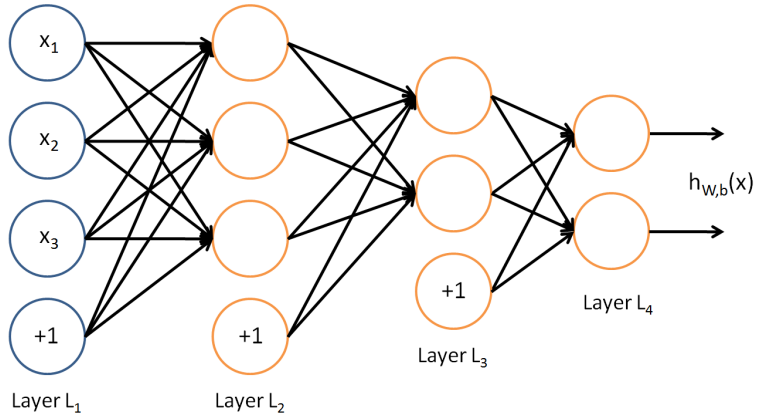


➤ A ogni hit possiamo associare un'immagine/collezione di pixel (es.15x15) centrata sul centro di carica.

Problema di classificazione binaria (SEGNALE vs FONDO) di immagini: adatto ad essere trattato con reti neurali.

► Una rete neurale **“tradizionale”** è un algoritmo volto ad approssimare una generica funzione f componendo funzioni più semplici organizzate in **livelli o layers**).

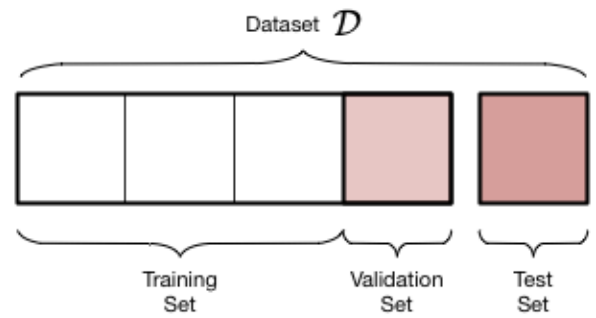
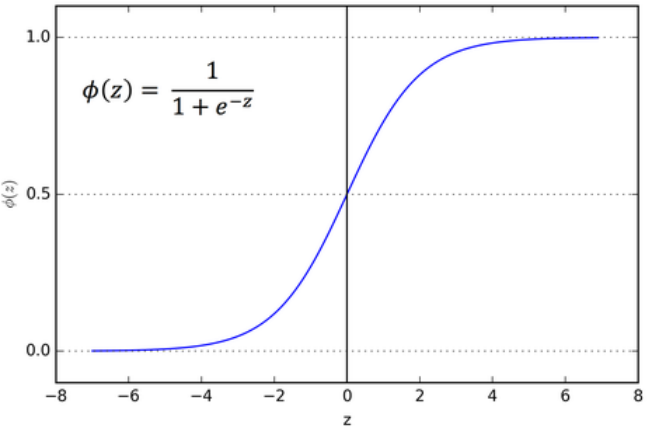
Ogni livello è costituito da più nodi (detti **neuroni**) che ricevono in ingresso una combinazione di segnali dal livello precedente e la trasformano secondo una funzione (detta di **attivazione**)



TRAINING SUPERVISIONATO ◀

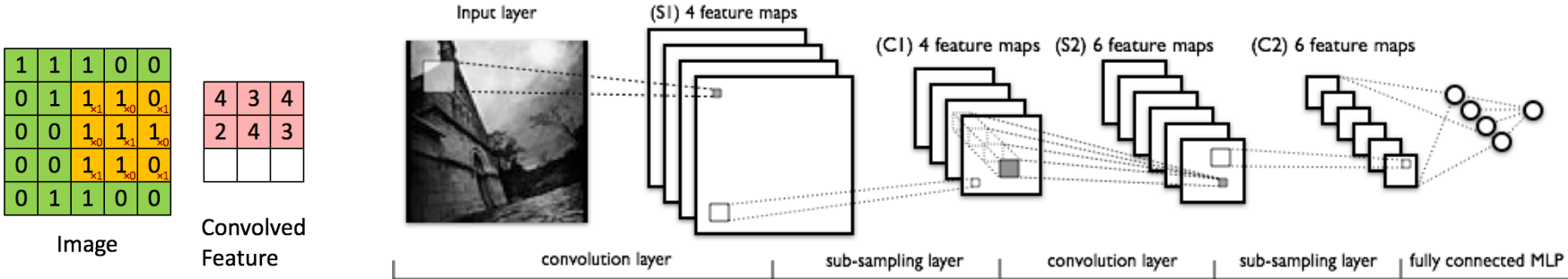
$$X_j^l = f(\sum_i^n \underbrace{w_i}_{\text{pesi}} \cdot X_i^{l-1} + \underbrace{b_i}_{\text{bias}})$$

Come si determinano i valori ottimali per \mathbf{b} e \mathbf{w} ?
 Se i valori di output sono noti si addestra la rete su un una porzione dei dati.

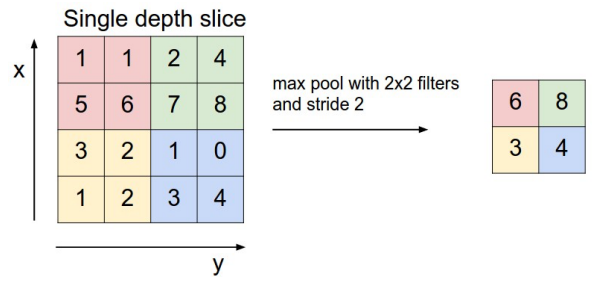


$$H(p, q) = - \sum_x p(x) \log q(x).$$

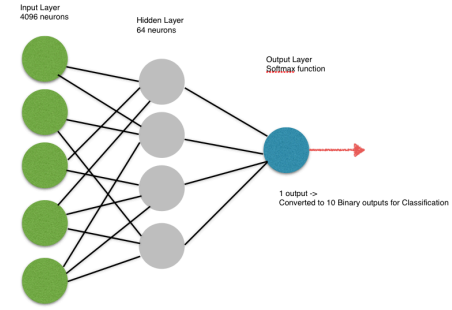
Una **rete neurale convoluzionale** (o CNN) è un tipo di rete neurale specializzata per l'analisi di dati con una struttura bidimensionale, come un'immagine composta da pixel. Il **processo di convoluzione** consiste nel **dividere l'immagine in vari frammenti sovrapposti ed analizzarli per individuarne le caratteristiche peculiari**.



Pooling: riduce la dimensione dei dati in modo da rendere più rapida l'analisi (es. operazioni di media o massimo).



Fully connected: livelli di una rete neurale "classica" che agiscono sull'immagine linearizzata e la riducono ad un unico score finale mediante una funzione **esponenziale normalizzata (o softmax)**

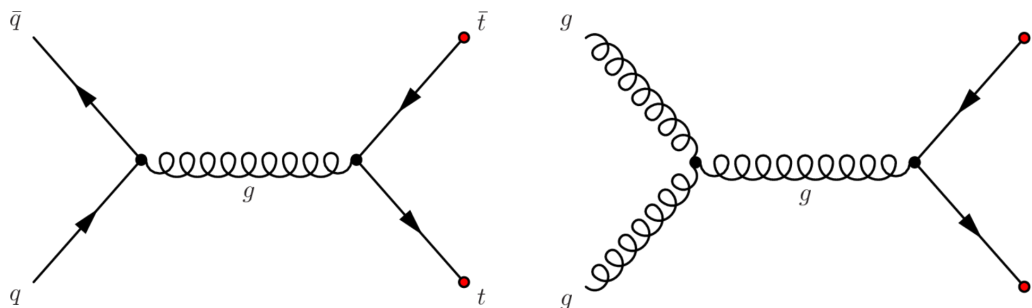


$$\sigma(\mathbf{z})_j = \frac{e^{z_j}}{\sum_{k=1}^K e^{z_k}}$$

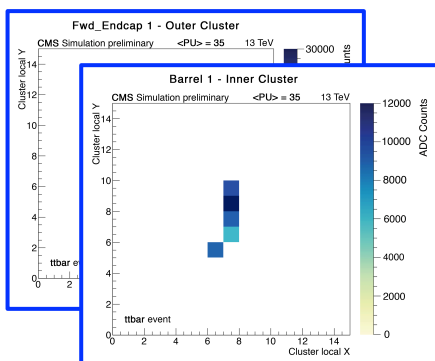
per $j = 1, \dots, K$.

Nel nostro caso la rete neurale agisce come un classificatore **binario (segnale o fondo)** e riduce quindi l'immagine ad un singolo punteggio $p(x)$ corrispondente alla $\mathbf{p}_{\text{segnale}}(\mathbf{x})$.

- Generazione di eventi simulati (mediante PYTHIA integrato framework per la ricostruzione CMSSW) di eventi $t\bar{t}$ a $\sqrt{s} = 13\text{TeV}$ e con $\langle PU \rangle = 35$ vengono prodotti $O(10^5)$ doppietti con un rapporto **fondo/segnale** $\sim O(100)$ corrispondente a **$O(1000)$ doppietti veri.**



- Ogni doppietto viene assegnato alla categoria **segnale** o **fondo** con la seguente procedura:
1. **raccolta** di tutti gli *hit delle tracce* ricostruite ed *associate ad una traccia simulata*;
 2. **raccolta** di tutti i *doppietti* prodotti;
 3. **doppietti di segnale** = doppietti composti da hit provenienti dalla stessa traccia (1)



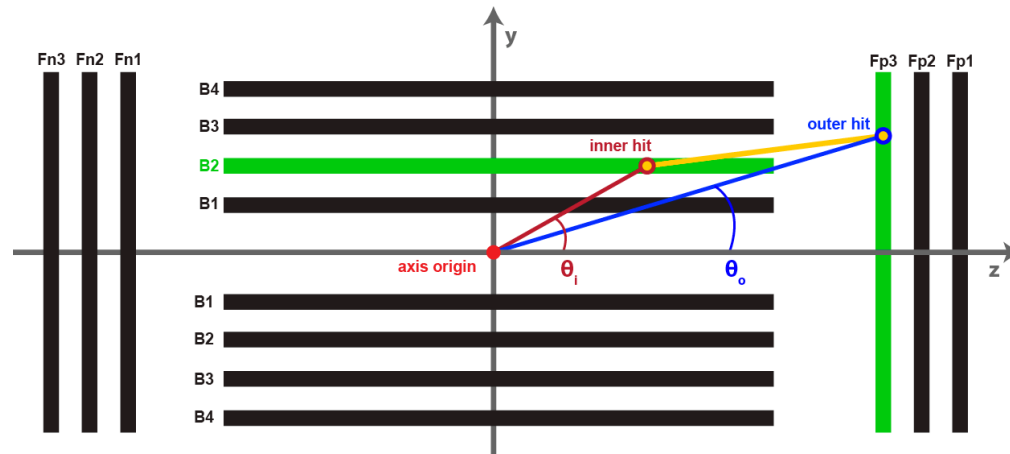
Ad ogni doppietto sono associati

- **due immagini 15x15** (una per l'hit più interno e una per l'hit più esterno)
- set di **informazioni locali** (x,y,z, carica,, ...)

➤ **Generati 2.5 milioni di doppietti (1000 eventi ca.)**

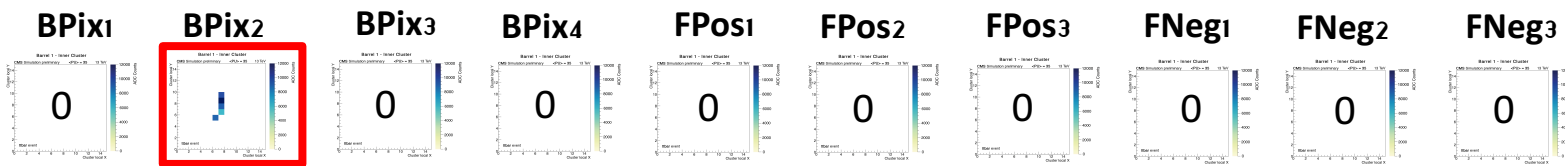
➤ Sfruttiamo la struttura a livelli del rivelatore per estendere ogni singolo doppietto da **due immagini** a **venti immagini** (o **canali**) una per **ogni livello** (*barrel o endcap*)

➤ **Canali:** un immagine può essere la sovrapposizione di più componenti. Es. livelli colore RGB per una foto JPG.

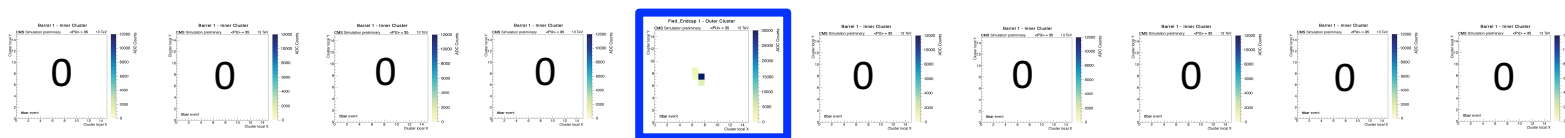


➤ Es. Doppietto tra i due livelli *barrel2* e *forward endcap3*

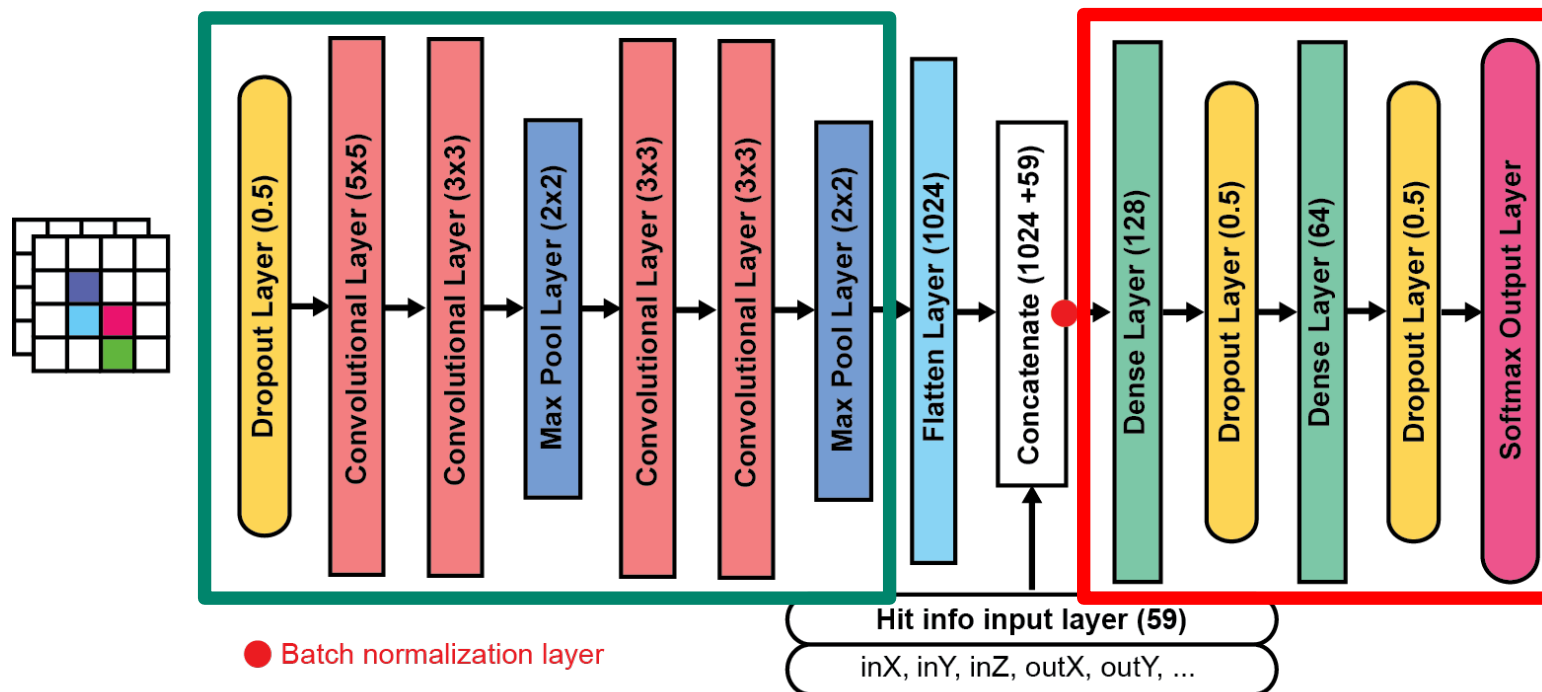
Hit Interno



Hit Esterno



Dati in ingresso per la rete



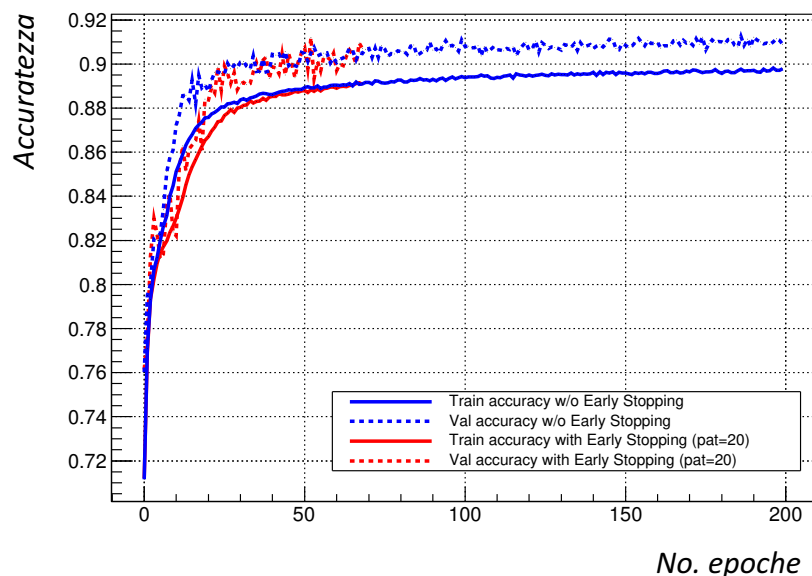
- Un singolo doppietto è costituito da un'immagine a **20 livelli**. Il modello sviluppato è la concatenazione di due strutture:
 - **Architettura CNN** successione di livelli convoluzionali (4) e livelli di max pooling (2)
 - **Architettura "DENSE"** : livelli "fully connected" (2) che ricevono in input le immagini ridotte a un vettore monodimensionale doublets parametri dei doppietti (inX, inY, inZ ...)
- Oltre ai 2.5 milioni di doppietti di "training" sono stati generati 20000 doppietti su cui validare la procedura.

► Risultati del **training** su 2.5 milioni di doppietti simulati

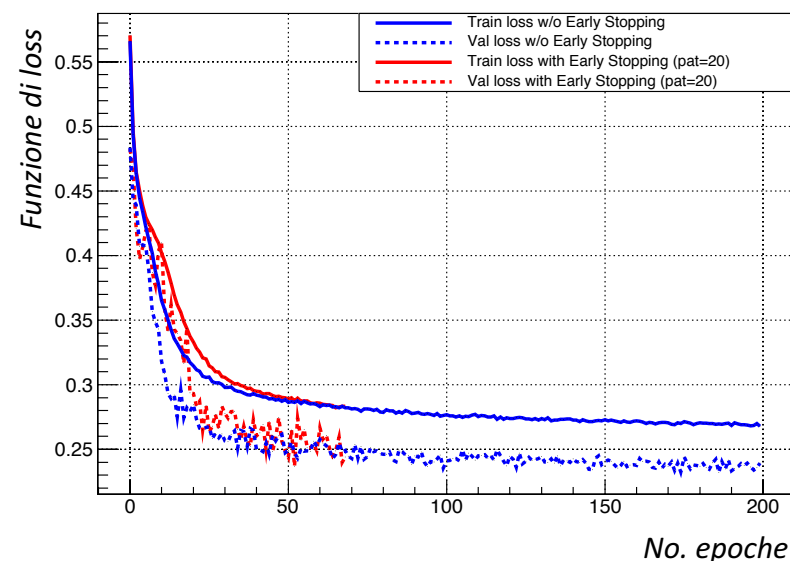
$$ACC = \frac{VP+VN}{P+N} > 0.92$$

$$H(p, q) = - \sum_x p(x) \log q(x).$$

Accuratezza (training su dataset 250k)



Entropia incrociata (training su dataset 250k)



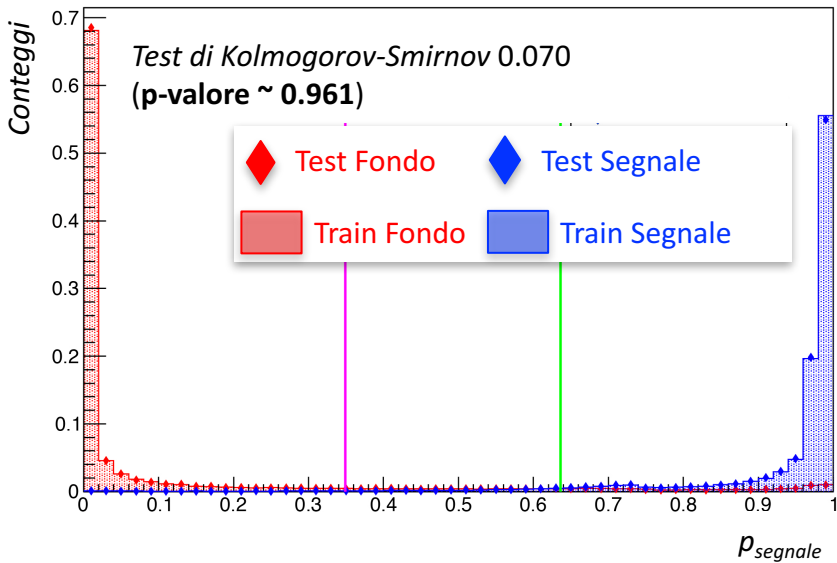
Training complessivo	@ Acc Max				Reiez @ Eff		Eff @ Reiez	
	AUC	ACC	Efficienza	Reiezione	0.99	0.999	0.99	0.5
Test	0.968	0.90	0.96	0.91	0.81	0.65	0.54	0.99
Val	0.966	0.90	0.95	0.91	0.81	0.66	0.54	0.99
Train	0.969	0.91	0.96	0.91	0.81	0.65	0.55	0.99

➤ Risultati del **training** su 2.5 milioni di doppietti simulati

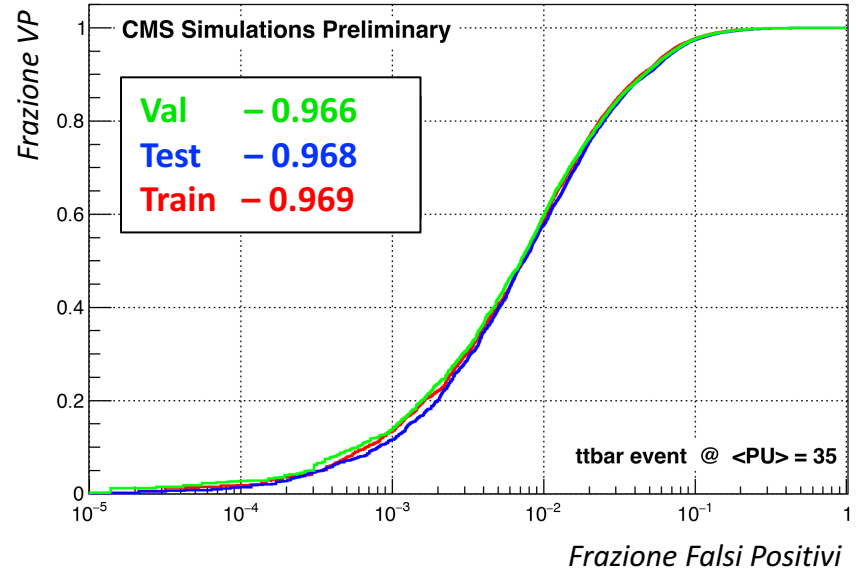
$$FVP = \frac{VP}{P} \quad \text{Sensibilità}$$

$$FFP = \frac{FP}{P} \quad \text{1-Specificità}$$

Punteggio del classificatore



Curva ROC



	Training complessivo		@ Acc Max		Reiez @ Eff		Eff @ Reiez	
	AUC	ACC	Efficienza	Reiezione	0.99	0.999	0.99	0.5
Test	0.968	0.90	0.96	0.91	0.81	0.65	0.54	0.99
Val	0.966	0.90	0.95	0.91	0.81	0.66	0.54	0.99
Train	0.969	0.91	0.96	0.91	0.81	0.65	0.55	0.99

- In definitiva, un approccio con reti neurali per la riduzione del fondo combinatorio sembra dare **risultati molto promettenti**.

- **Integrazione** nel software di ricostruzione di CMS

- Verifica dell'**effetto della riduzione** dei doppietti di seed sui successivi passaggi della ricostruzione

- **Visualizzazione** dei dati



GRAZIE PER L'ATTENZIONE

"I am putting myself to the fullest possible use, which is all I think that any conscious entity can ever hope to do"

HAL9000