



# UNIVERSITÀ DI SIENA

1240

DIPARTIMENTO DI SCIENZE FISICHE, DELLA TERRA E DELL'AMBIENTE

---

---

Corso di Laurea in Fisica e Tecnologie Avanzate

## APPLICAZIONE DI TECNICHE DI INTELLIGENZA ARTIFICIALE AL TRACCIAMENTO DI RAGGI COSMICI CARICHI NEL RIVELATORE CALET

*Relatore:*  
Prof. Paolo Maestro

*Tesi di laurea di:*  
Giulio Giamello

Anno Accademico 2023/2024



# Indice

<b>Introduzione</b>	<b>1</b>
<b>1 Reti neurali e loro applicazioni</b>	<b>3</b>
1.1 Introduzione alle reti neurali . . . . .	3
1.1.1 Supervised Learning . . . . .	4
1.1.2 Unsupervised Learning . . . . .	5
1.1.3 Self-Supervised Learning . . . . .	5
1.2 Deep learning: rappresentazioni e pesi . . . . .	5
1.2.1 Overfitting e underfitting . . . . .	6
1.3 Convolutional Neural Networks . . . . .	7
1.3.1 L'operazione di convoluzione . . . . .	7
1.3.2 Vantaggi e proprietà delle CNNs . . . . .	9
1.3.3 Funzioni di attivazione . . . . .	13
1.3.4 Rectified Linear Unit . . . . .	13
1.3.5 Pooling . . . . .	14
<b>2 CALorimetric Electron Telescope (CALET)</b>	<b>15</b>
2.1 Sciami di particelle . . . . .	15
2.1.1 Sciami elettromagnetici . . . . .	16
Produzione di coppia . . . . .	17
Bremsstrahlung . . . . .	18
Sviluppo longitudinale . . . . .	19
Sviluppo laterale . . . . .	20
2.1.2 Sciami adronici . . . . .	21
Sviluppo longitudinale . . . . .	22
Sviluppo laterale . . . . .	23
2.1.3 Calorimetria . . . . .	24
2.2 Missione CALET . . . . .	25
2.2.1 Calorimetro CALET . . . . .	27
CHarge Detector (CHD) . . . . .	28
IMaging Calorimeter (IMC) . . . . .	29

Total AbSorption Charge (TASC) . . . . .	30
Sistemi di trigger . . . . .	32
2.3 Risultati di CALET: spettro di nuclei di elio . . . . .	33
<b>3 Struttura e training della rete neurale</b>	<b>37</b>
3.1 Introduzione . . . . .	37
3.2 Descrizione del problema . . . . .	39
3.2.1 Accettanza . . . . .	39
3.2.2 Strategia e procedimento . . . . .	40
3.2.3 Dati simulati MC . . . . .	41
3.3 Coding e sviluppo della rete . . . . .	43
3.3.1 Modello di rete con Keras . . . . .	43
3.3.2 Workstation . . . . .	43
3.3.3 Struttura della rete . . . . .	44
3.4 Training rete . . . . .	45
3.4.1 Preparazione dei dati . . . . .	45
3.4.2 Training e salvataggio predizioni . . . . .	46
3.5 Applicazione della CNN ai dati . . . . .	47
<b>4 Analisi dei dati delle simulazioni MC</b>	<b>49</b>
4.1 Introduzione . . . . .	49
4.2 Analisi dati . . . . .	50
4.2.1 Training, <i>mae</i> ed <i>mse</i> . . . . .	52
4.2.2 Distribuzioni dei residui . . . . .	54
4.2.3 Media dei residui e risoluzione spaziale . . . . .	58
4.2.4 Media dei residui e risoluzione angolare . . . . .	62
4.3 Confronto quantitativo fra CNN e metodo dell'asse dello sciame . . . . .	64
4.4 Analisi dati con Kalman filter . . . . .	65
4.4.1 Distribuzioni dei residui . . . . .	67
4.4.2 Media dei residui e risoluzione spaziale . . . . .	69
4.4.3 Media dei residui e risoluzione angolare . . . . .	71
4.5 Kalman filter: confronto qualitativo . . . . .	72
<b>5 Analisi dei dati di volo</b>	<b>73</b>
5.1 Introduzione . . . . .	73
5.2 Selezione delle particelle . . . . .	73
5.2.1 Misura della carica . . . . .	73
5.2.2 Fit spettrale con legge di potenza . . . . .	75
5.3 Risultati della CNN sui dati di volo . . . . .	76
5.4 Distribuzioni dei residui . . . . .	78
5.5 Media dei residui e risoluzione spaziale . . . . .	80

5.6 Dati di volo: confronto quantitativo . . . . .	83
<b>Conclusioni</b>	<b>87</b>
<b>A Metodi di tracciamento utilizzati in CALET</b>	<b>89</b>
A.1 Kalman filter . . . . .	90
A.2 Metodo dell'asse dello sciamone . . . . .	92
<b>B Grafici</b>	<b>93</b>
B.1 Kalman filter, campione di dati simulati MC . . . . .	95
B.2 Campione di nuclei di He simulati MC . . . . .	104
B.3 Campione di p e nuclei di He simulati MC . . . . .	114
B.4 Campione di nuclei di He dei dati di volo . . . . .	124
B.5 Campione di p e nuclei di He dei dati di volo . . . . .	135
B.6 Selezione della carica dei nuclei . . . . .	146
<b>Bibliografia</b>	<b>150</b>



# Introduzione

Sin dalla loro scoperta nel 1912 da parte del fisico Victor Hess [1] (premio Nobel nel 1936), i *raggi cosmici* (CR) ovvero quella che, all'inizio, venne definita ‘radiazione invisibile e penetrante’ sono stati e sono tutt'oggi oggetto di studio e di ricerca scientifica.

In particolare i maggiori interessi sono rivolti alla comprensione della loro origine e composizione, nonché dei meccanismi accelerazione e propagazione nel mezzo interstellare. L'obiettivo dello studio diretto<sup>1</sup> dei raggi cosmici è dunque quello di compiere misure degli spettri energetici dei singoli elementi, delle loro abbondanze ed individuare nello spazio delle ‘regioni’ di provenienza di questi, cosicché sia possibile indagarne l'origine ed i fenomeni di accelerazione.

La raccolta dati e lo studio diretto degli spettri energetici dei CR sono alla base dello sviluppo delle teorie astrofisiche che si occupano di modellizzare e descrivere tali fenomeni. A questo scopo negli ultimi decenni sono stati sviluppati strumenti calorimetrici in grado di rivelare CR fino ad energie dell'ordine di centinaia di TeV, tramite la misura dell'energia depositata in essi dagli sciami di particelle originati dai CR che li attraversano. Grazie all'utilizzo di altri rivelatori di precisione, questi calorimetri sono in grado di compiere misure precise anche sulla carica e sulla direzione di arrivo nel calorimetro.

Il recente sviluppo della ricerca e delle tecniche di *Intelligenza Artificiale* (AI), in particolare di *Machine* e *Deep Learning* hanno permesso l'impiego di queste ‘nuove’ tecnologie anche nell’analisi dati: l'avere accesso a grandi quantità di dati permette di sviluppare modelli in grado di compiere predizioni accurate su certe quantità di interesse scientifico. Ad esempio, in esperimenti di fisica delle alte energie, l'impiego di tali tecniche permette di studiare ed approcciare il problema del *tracking* dei CR da un'altra prospettiva.

La missione CALET (*CALorimetric Electron Telescope*) è un esperimento di calorimetria spaziale in orbita sulla ISS (*International Space Station*) di cui fa parte anche un gruppo di ricerca del DSFTA (*Dipartimento di Scienze Fisiche e Tecnologie Avanzate* dell'Università di Siena).

---

<sup>1</sup>Tramite strumenti di misura in orbita nella stratosfera o nello spazio.

Lo studio presentato in questa tesi si è occupato di sviluppare un modello di rete neurale convoluzionale (CNN) in grado di predire la traiettoria di attraversamento seguita da nuclei di elio nel range di energia (20,2e4) [GeV]. Partendo quindi da simulazioni Monte Carlo (MC) degli eventi di attraversamento del calorimetro CALET, è stato effettuato il *training* della CNN su tali dati, ne è stata misurata la risoluzione e ne sono state confrontate le prestazioni con altri algoritmi utilizzati in CALET per ricostruire la traiettoria di attraversamento del CR.

Una volta allenato il modello e verificatane l'efficienza, questo è stato poi applicato su un *dataset* di dati di volo<sup>2</sup> (FD). Per valutare la correttezza delle predizioni effettuate, è stato effettuato un confronto con i valori calcolati tramite un *Kalman filter combinatorio* (KF), algoritmo già implementato in CALET per ricostruire le traiettorie dei CR.

Infine, tramite una stima della risoluzione dell'algoritmo KF sullo stesso dataset impiegato nel training del modello, è stato possibile misurare la risoluzione della CNN sui dati di volo.

Di seguito si introducono brevemente i capitoli di questo lavoro di tesi:

- [Capitolo 1](#), viene fornita un'introduzione ai principi di funzionamento delle reti neurali, in particolare al tipo utilizzato nello studio di questa tesi, ovvero alle *convolutional neural network*;
- [Capitolo 2](#), dopo un'introduzione sugli sciami di particelle, vengono descritti l'esperimento CALET, la sua struttura ed i principali risultati ottenuti;
- [Capitolo 3](#), viene illustrata la struttura scelta per la costruzione del modello di CNN utilizzato e viene spiegato come è stato effettuato il *training* di quest'ultimo;
- [Capitolo 4](#), vengono riportati i risultati prodotti dalla CNN su un dataset di dati CALET simulati tramite Monte Carlo;
- [Capitolo 5](#), la CNN viene impiegata per predire i punti di impatto di eventi relativi a dati veri, quindi ne viene poi misurata l'efficienza tramite il confronto con l'algoritmo Kalman filter combinatorio.

---

<sup>2</sup>Dati ‘veri’ di eventi di attraversamento registrati da CALET nel periodo compreso tra gennaio e marzo 2019.

# Capitolo 1

## Reti neurali e loro applicazioni

### 1.1 Introduzione alle reti neurali

L'*artificial intelligence* (AI) costituisce un campo della *computer science* che studia il modo in cui i computer possano “apprendere” dall’esperienza e “ragionare” sulla base di gerarchie, partendo da concetti semplici e definendone di più complessi sulla base di questi [2, 3, 4].

La branca della AI che studia come un computer possa andare oltre ciò che gli viene insegnato ed apprendere *da solo* come risolvere un problema è detta *machine learning* (ML). Tale campo di ricerca si occupa dell’auto-apprendimento: ovvero ricerca algoritmi che permettano di non dover passare in input al computer, oltre ai dati (*dataset*), anche i criteri secondo cui processarli, ma che, a partire da dati, permettano al computer di capire autonomamente quali criteri seguire per processarli. Per fare in modo che il computer sia in grado di apprendere autonomamente i criteri e le regole, il machine learning ricorre ad algoritmi che accettano in input i dati ed, in alcune tipologie di apprendimento, ad es. nel *supervised learning*, anche la risposta desiderata (il valore vero, che la rete deve restituire in output quando si passa in input il dato corrispondente).

Le informazioni in input vengono poi processate tramite algoritmi con lo scopo di produrre un modello statistico in grado di compiere previsioni, ovvero di restituire un output partendo da dati che non sono contenuti nelle informazioni utilizzate per la costruzione del modello stesso.

Un generico modello di machine learning viene costruito tramite i seguenti elementi:

- un *dataset* in input;
- gli esempi degli outputs attesi (*targets*), corrispondenti a ciascun elemento del dataset (ad es. nel *supervised learning*);

- una *funzione di loss*, ovvero una funzione che misuri la distanza tra il valore previsto e quello calcolato dal modello ad una certa iterazione del training.

Il valore della funzione di loss costituisce un'indicazione di come modificare l'algoritmo per andare a minimizzare la distanza tra dati attesi e calcolati.

Un modello di machine learning trasforma i dati in input in un output con un certo “significato”. Segue allora che la scelta della *rappresentazione* dei dati in input gioca un ruolo fondamentale nel:

- produrre output con “significato” utile;
- migliorare le performance degli algoritmi di machine learning (ad es. per costruire un modello più preciso o per ridurre le risorse computazionali richieste).

Per cercare di produrre outputs “significativi” si ricorre ad un sotto-campo del ML: il *deep learning* (DL).

Il DL si occupa di individuare le migliori rappresentazioni dei dati in input per risolvere un problema pratico tramite *layers* di rappresentazioni successive, sempre più significative. Il numero di layers viene detto *depth* del modello e da questo segue il nome deep learning. Le rappresentazioni di ciascun layer sono apprese tramite modelli chiamati *neural networks*.

Ciò che interessa è dunque approfondire come avviene l'apprendimento delle informazioni contenute nel dataset da parte della macchina, ovvero come avviene il *Machine Learning*. È possibile individuare tre principali categorie di algoritmi di machine learning: *Supervised Learning*, *Unsupervised Learning* e *Self-Supervised Learning*.

(Verranno di seguito introdotte notazione e terminologia più specifiche).

### 1.1.1 Supervised Learning

Nel supervised learning il *dataset* è costituito da una collezione di *labeled examples*

$$\{(\mathbf{x}_i, y_i)\}_{i=1}^N$$

Dove ciascun elemento  $\mathbf{x}_i$  è detto *feature vector* ed ha dimensione  $D$ . Ogni elemento di un feature vector ( $j = 1, \dots, D$ ) indica una specifica feature dell'evento  $i$ -esimo che stiamo descrivendo.

Ad esempio, se stiamo descrivendo un evento di rivelazione di particelle possiamo avere che la  $j$ -esima componente di ciascun feature vector  $x^{(j)}$  descrive una specifica feature (ad es. la carica della particella), mentre la  $j+1$ -esima  $x^{(j+1)}$  descrive un'altra

feature (ad es. la direzione) e così via.

L'elemento  $y_i$  è detto *label* e può essere un elemento appartenente a: un insieme finito di *classes*  $\{1, 2, \dots, C\}$ , all'insieme dei numeri reali  $\mathbb{R}$ , o anche a strutture più complesse come matrici, tensori o grafi.

Una *classe* può essere definita come una categoria cui un certo evento appartiene.

L'obiettivo di un algoritmo di Supervised Machine Learning è quello di usare il dataset per produrre un modello statistico che accetti in input un feature vector  $\mathbf{x}$  e restituisca in output informazioni che permettono di risalire al label di tale feature vector.

### 1.1.2 Unsupervised Learning

Si tratta di unsupervised learning quando il dataset è formato da una collezione di *unlabeled examples*

$$\{\mathbf{x}_i\}_{i=1}^N$$

con  $\mathbf{x}_i$  feature vector. In questo caso l'obiettivo consiste nel creare un modello che accetti in input il feature vector e lo trasformi in un altro vettore o in un valore numerico che possa essere sfruttato per risolvere il problema concreto di partenza, tramite procedure di *data visualization*, *data compression* o *data denoising*.

Solitamente il learning unsupervised viene sfruttato come step iniziale per avere una migliore comprensione del dataset di partenza, prima di passarlo cercare di risolvere il problema tramite supervised learning. Alcuni esempi di unsupervised learning sono: *clustering* e la *dimensionality reduction*.

### 1.1.3 Self-Supervised Learning

Si parla di self-supervised learning quando il dataset contiene solo *unlabeled examples*, come nel caso unsupervised, ed è la rete stessa, senza bisogno di passare i labels, ad “estrarli” ed “apprenderli” dai dati in input.

Viene chiamata *self-supervised* perché non richiede l'intervento di un controllo umano esterno (*supervised*), ma è la rete stessa ad individuare i labels.

## 1.2 Deep learning: rappresentazioni e pesi

Nel machine learning viene dunque effettuato un *mapping* tra gli inputs ed i targets a partire dagli esempi di riferimento passati al computer. Nel deep learning il mapping avviene tramite una “profonda” (*deep*) successione di trasformazioni semplici dei dati (*layers*) e delle informazioni acquisite dai layers precedenti.

La trasformazione che ciascun layer effettua sui dati dipende dai “pesi” (*weights*) del layer, che, nella maggior parte dei casi, sono dei valori numerici costanti. Si dice che la trasformazione implementata da un layer è *parametrizzata* dai suoi pesi.

È allora possibile ricondurre il problema del learning al problema della scelta dei pesi ottimali: ciò viene fatto tramite la *funzione di loss*, che va a misurare la distanza tra le predizioni ed i valori target veri, e tramite un *optimizer* che interviene sui pesi per ridurre il valore della distanza misurata. Questa procedura prende il nome di *Backpropagation algorithm*.

Inizialmente per i pesi vengono scelti dei valori random, di conseguenza la rete inizia ad implementare una serie di trasformazioni, anch’esse random: successivamente, con l’aumentare del numero di eventi processati, i pesi vengono aggiustati nella “direzione corretta”, ovvero quella per cui la funzione di loss è minima.

Questa procedura, detta *training loop*, viene ripetuta un numero di volte sufficiente (tipicamente decine di iterazioni su migliaia di eventi) a raggiungere un valore della funzione di loss prossimo al minimo, così da avere valori predetti il più “vicini” possibile ai valori targets veri.

### 1.2.1 Overfitting e underfitting

*Overfitting* e *underfitting* costituiscono due dei principali problemi cui prestare attenzione nel training di un modello per la risoluzione di un problema di machine learning. Si osserva infatti che è necessario trovare un bilanciamento tra due caratteristiche dei modelli di machine learning: l’*ottimizzazione* e la *generalizzazione*. La prima si riferisce al processo di modifica del modello per ottenere performance migliori; mentre la seconda indica l’abilità del modello di compiere predizioni su dati mai visti prima. L’obiettivo è quello di ottenere una buona generalizzazione e l’unico modo per farlo è modificare i dati di training.

All’inizio della fase di training, ottimizzazione e generalizzazione sono legate: al diminuire dell’errore sui dati di training diminuisce anche quello sui dati di test, in questa fase siamo in *underfit*. Il modello non ha ancora completamente appreso i patterns per comprendere i dati in input come richiesto. Succede poi che, dopo un certo numero di iterazioni sui dati di training, la generalizzazione smette di migliorare per poi iniziare a peggiorare, il modello si trova in *overfit*: ovvero la rete sta imparando patterns che sono specifici dei dati di training, ma non generali e quindi non validi anche sui dati di test, causando così un peggioramento delle prestazioni.

La prima soluzione per evitare che si verifichi questo fenomeno è quella di ricorrere a dataset di training con un numero più elevato di dati. Nei casi però in cui ciò non fosse possibile, si deve prestare attenzione ad interrompere le iterazioni (o epoches) del modello sui dati di training prima che questo vada in overfit, così da ottenere le prestazioni migliori.

## 1.3 Convolutional Neural Networks

Le reti convoluzionali (*convolutional networks*) [5], note anche come reti neurali convoluzionali (*convolutional neural networks*) *CNNs* sono uno specifica tipologia di reti neurali impiegate per processare dati con una struttura ‘a griglia’ nota. Alcuni esempi di dataset con questa struttura possono essere: serie di valori in funzione del tempo, che possono essere pensati come una griglia 1D di dati, campionati ad intervalli regolari; oppure le immagini, che possono essere considerate come griglie 2D (matrici) di pixels.

Il nome “reti neurali convoluzionali” indica che le reti di questo tipo compiono sui dati un’operazione matematica detta *convoluzione*. La *convoluzione* è un tipo particolare di operazione lineare: le reti convoluzionali sono semplici reti neurali che, al posto della moltiplicazione matriciale, utilizzano l’operazione di convoluzione in almeno uno dei loro layers.

### 1.3.1 L’operazione di convoluzione

Nella sua definizione più generale la *convoluzione* è un’operazione su due funzioni di argomento reale.

$$s(t) = (x * w)(t) \quad (1.1)$$

$$s(t) = \int x(a)w(t-a) da \quad (1.2)$$

Indichiamo il primo argomento dell’operazione di convoluzione  $x$  con *input* ed il secondo  $w$  con *kernel*.

Ad esempio, se con  $x(t)$  indichiamo la posizione di un oggetto che si muove nel tempo e supponiamo che la misura di tale posizione sia affetta da un qualche rumore, per avere una migliore stima di tale posizione possiamo ricorrere alla convoluzione. Possiamo infatti calcolare la media di molteplici misurazioni, pesando ciascuna con un certo valore  $w(a)$ , dove, in questo caso,  $a$  indica il momento nel tempo in cui è stata compiuta la misura ed è possibile dunque ottenere una funzione  $s(t)$  con cui stimare la posizione  $x(t)$ .

Più in generale la convoluzione è definita per qualsiasi funzione per cui risulti definito l’integrale riportato sopra. In [Figura 1.1](#) si ha una rappresentazione grafica dell’operazione di convoluzione.

Solitamente, quando si lavora con certi dati tramite un computer, questo utilizza intervalli discreti di tempo in cui vengono compiute le misure, anch’esse discrete. Dunque, se assumiamo invece che  $x$  e  $w$  siano discreti, ovvero siano definiti solo su valori interi di  $t$ , possiamo definire la *convoluzione discreta* come

$$s(t) = (x * w)(t) = \sum_{a=-\infty}^{\infty} x(a)w(t-a) \quad (1.3)$$

L'output  $s(t)$  viene solitamente detto *feature map*.

Si noti che, sebbene nella definizione della somma l'indice  $a$  scorre da  $-\infty$  a  $+\infty$ , poiché ogni elemento dell'input e del kernel devono essere salvati separatamente, nella pratica si assume che tali funzioni siano ovunque nulle tranne che nei punti per i quali vengono salvati i valori. Questo permette di implementare l'[Equazione 1.3](#) come sommatoria su di un numero finito di elementi.

Se, ad esempio, applichiamo la convoluzione su più di un asse contemporaneamente, ovvero se utilizziamo un'immagine  $I$  a due dimensioni come input, anche il kernel sarà bidimensionale:

$$S(i, j) = (I * K)(i, j) = \sum_m \sum_n I(m, n)K(i - m, j - n) \quad (1.4)$$

L'operazione di convoluzione gode della proprietà commutativa:

$$S(i, j) = (K * I)(i, j) = \sum_m \sum_n I(i - m, j - n)K(m, n) \quad (1.5)$$

La proprietà commutativa segue dal fatto che abbiamo *flippato* il kernel rispetto all'input: al crescere di  $m$ , cresce l'indice dell'input e decresce quello del kernel o viceversa.

Possiamo definire anche la funzione *cross-correlation*, che compie la stessa operazione della convoluzione, ma senza effettuare il *flip* del kernel.

$$S(i, j) = (K * I)(i, j) = \sum_m \sum_n I(i + m, j + n)K(m, n) \quad (1.6)$$

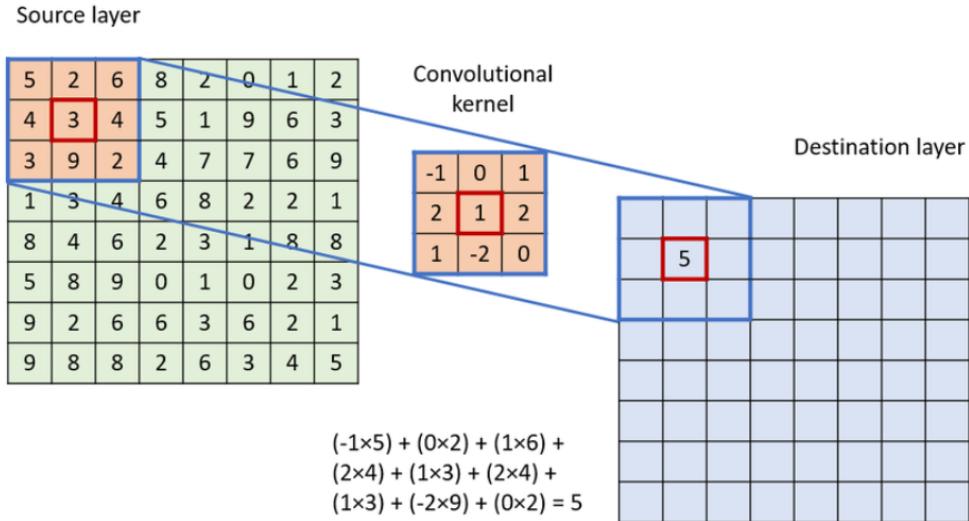


Figura 1.1: nell’immagine viene mostrato un esempio di applicazione dell’operazione di convoluzione discreta tra una matrice di kernel (*Convolutional kernel*) e una matrice di input (*Source layer*).

### 1.3.2 Vantaggi e proprietà delle CNNs

L’operazione di convoluzione è particolarmente vantaggiosa per i sistemi di machine learning poiché gode delle seguenti proprietà:

- *Sparse interactions*;
- *Parameter sharing*;
- *Equivariance to translation*.

Come detto in precedenza, i layers delle reti neurali ‘tradizionali’ utilizzano la moltiplicazione matriciale dei parametri, ovvero utilizzano un parametro distinto per descrivere ogni singola interazione tra un nodo di input e di output. Questo significa che ogni nodo (o neurone) di input interagisce con tutti i nodi di output.

Le reti convoluzionali invece ricorrono a *sparse interactions* andando a ridurre la dimensione del kernel rispetto a quella dell’input. Ad esempio, nel processare un’immagine, l’input (immagine) può avere migliaia o milioni di pixels, ma possiamo rivelare caratteristiche (*features*) ‘piccole’ e ‘significative’, quali i bordi delle figure, anche con un kernel che ricopre solo decine o centinaia di pixels e questo si traduce nella possibilità di dover tenere in memoria un minor numero di parametri, migliorando l’efficienza statistica del modello.

Inoltre, le sparse interactions riducono il numero di operazioni richieste in fase di computazione, andando così ad incrementare l'efficienza del modello: se abbiamo  $m$  inputs ed  $n$  outputs, la moltiplicazione matriciale richiede  $m \times n$  parametri e l'algoritmo ha una *complessità temporale*<sup>1</sup> di  $O(m \times n)$ . Al contrario, se imponiamo un limite massimo  $k$  al numero di connessioni che ciascun output può avere, allora l'approccio con sparse interactions richiede un numero minore di parametri  $k \times n$  con una ridotta complessità temporale  $O(k \times n)$ .

Per una migliore comprensione, il concetto della *sparse connectivity* viene rappresentato di seguito nella Figura 1.2 in funzione delle ‘connessioni’ (frecce) tra uno specifico input e gli outputs; mentre nella Figura 1.3 in funzione degli ‘outputs’ (frecce) tra uno specifico output e gli inputs.

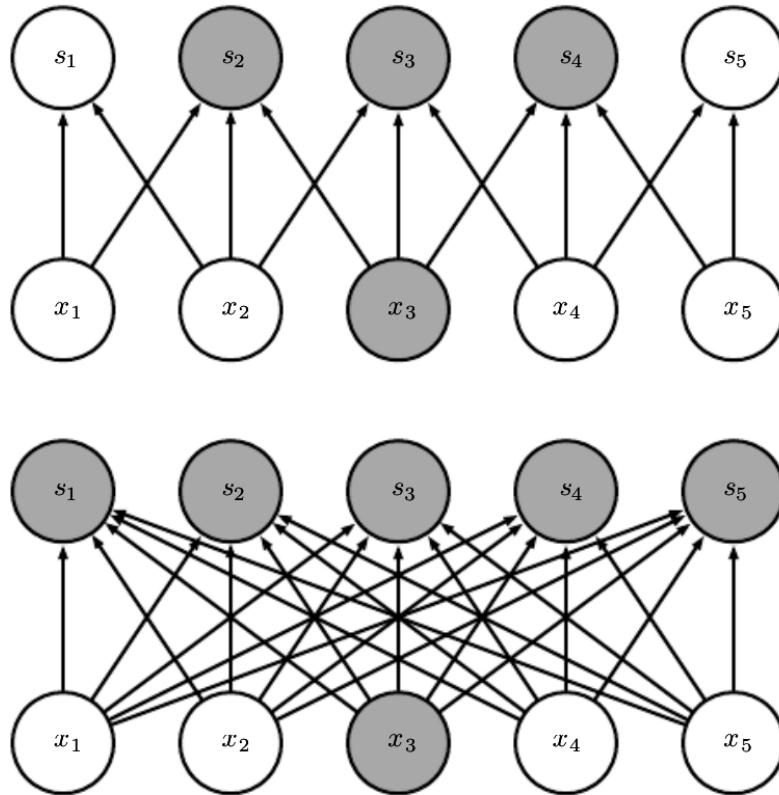


Figura 1.2: nell’immagine sono evidenziati in grigio scuro il nodo di input  $x_3$  ed i nodi di output  $s$  a questo legati. Di sopra, quando l’output è ottenuto con *sparse connectivity*; di sotto quando è ottenuto tramite moltiplicazione matriciale.

<sup>1</sup>La *complessità temporale* di un algoritmo quantifica il tempo necessario per l’esecuzione.

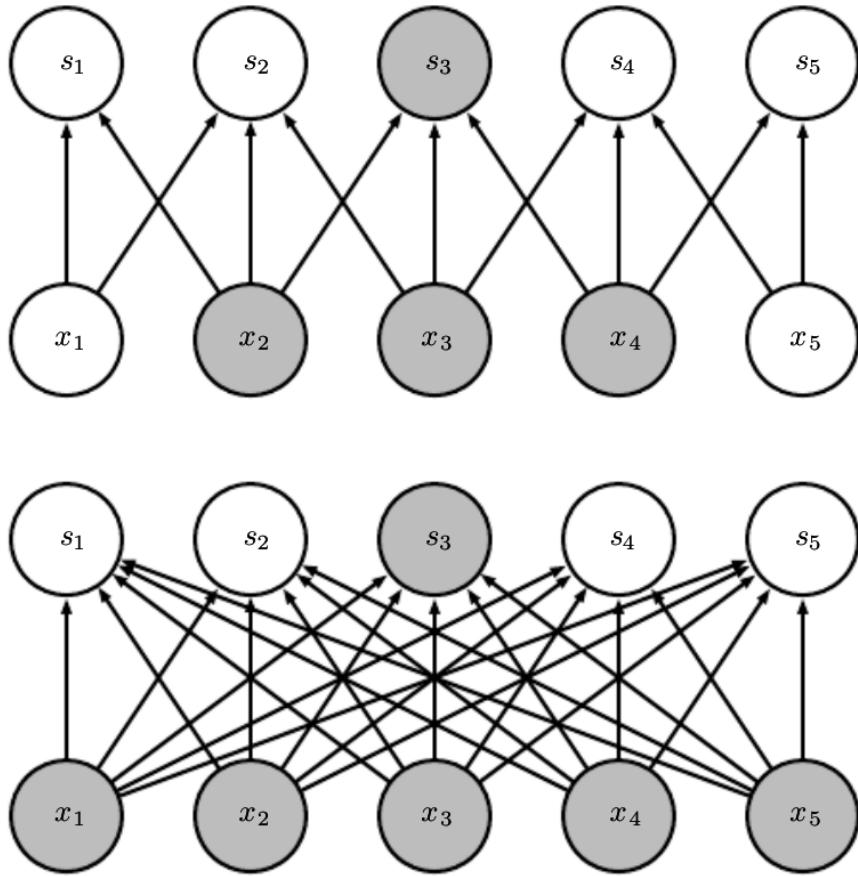


Figura 1.3: nell’immagine sono evidenziati in grigio scuro i nodi di input  $x$  e il nodo di output  $s_3$ . Di sopra, quando l’output è ottenuto con *sparse connectivity*; di sotto quando è ottenuto tramite moltiplicazione matriciale.

Nelle reti convoluzionali particolarmente ‘profonde’, ovvero con un numero elevato di layers, i nodi dei layers più profondi possono interagire indirettamente con una porzione maggiore dei nodi di input, come viene graficamente mostrato nella [Figura 1.4](#). Questo permette alla rete di descrivere in maniera efficiente complicate interazioni tra molteplici variabili, tramite la costruzione di interazioni di questo tipo a partire da nodi semplici, in cui ciascuno descrive sparse interactions.

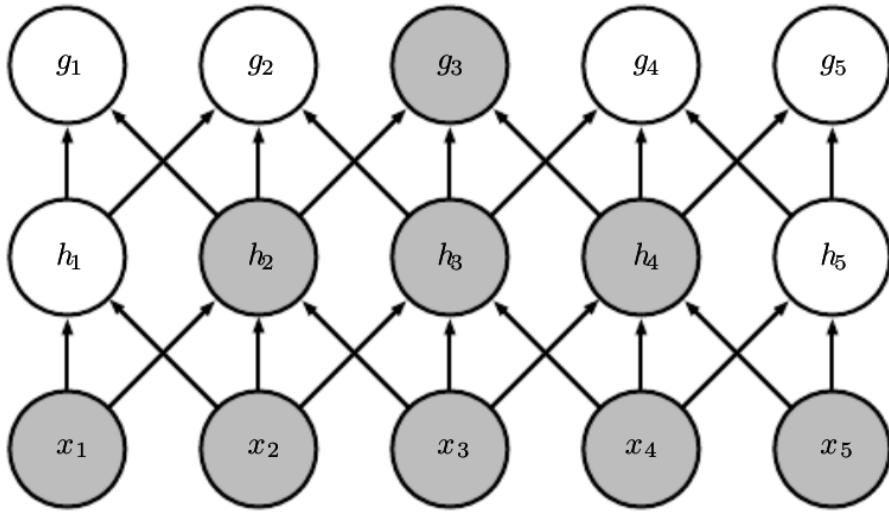


Figura 1.4: nell’immagine viene mostrato come il ‘campo ricettivo’ dei nodi dei layers più profondi sia maggiore di quello dei layers più superficiali. Ciò significa che, sebbene in una rete convoluzionale le connessioni siano *sparse*, i nodi dei layers più profondi possono essere comunque connessi *indirettamente con tutti o quasi gli inputs*.

In una rete tradizionale ciascun elemento della matrice dei pesi viene usato esattamente una volta nel calcolare l’output di un certo layer: viene moltiplicato per un elemento dell’input e mai più impiegato nella computazione dell’output.

In una rete convoluzionale invece si utilizzano pesi ‘legati’ (*tied weights*); infatti il valore di un peso applicato ad un certo input è legato al valore di un altro peso applicato ad un altro input. Questa operazione è detta *parameter sharing* ed evita che la rete debba imparare set di parametri diversi per ogni possibile elemento, facendo in modo che utilizzi sempre lo stesso set di parametri (condivisi) per ogni elemento.

Questa caratteristica non modifica il tempo di esecuzione, che rimane  $O(k \times n)$ , ma riduce lo spazio richiesto in memoria, dato che limita il numero di parametri della matrice dei pesi a  $k$ . Generalmente  $k$  è diversi ordini di grandezza più piccolo di  $m$  e, poiché  $m$  ed  $n$  sono approssimativamente dello stesso ordine di grandezza,  $k$  è trascurabile rispetto a  $m \times n$ , andando così a ridurre notevolmente la memoria richiesta.

Dalla caratteristica del *parameter sharing* segue inoltre che le reti convoluzionali godono anche della proprietà di ‘equivarianza’ (*equivariance*) rispetto a traslazioni.

Una funzione viene definita ‘equivariante’ se ad una variazione dell’input corrisponde una variazione dell’output *nello stesso modo*. Più nel dettaglio, si intende che: data una funzione  $f(x)$ , questa è *equivariante* rispetto ad una funzione  $g(x)$  se

$$f(g(x)) = g(f(x))$$

Se come funzione  $g$  prendiamo una qualsiasi funzione di traslazione dell'input, allora l'operazione di convoluzione è equivariante con tale  $g$ . Ad esempio, sia  $I$  una funzione che descrive la luminosità di un'immagine in punti con coordinate intere e sia  $g$  una funzione che mappa un'immagine con un'altra,  $I' = g(I)$  con  $I'(x, y) = I(x - 1, y)$ . La funzione  $g$  in questo esempio sposta tutti i pixel dell'immagine  $I$  di una unità verso destra; se applichiamo questa trasformazione ad  $I$  e poi applichiamo la convoluzione sull'immagine  $I'$ , il risultato è lo stesso che se avessimo applicato le due operazioni in ordine inverso, ovvero prima la convoluzione sull'immagine  $I$  e poi la funzione  $g$ :

$$w * g(I) = g(w * I)$$

Nel processare serie di dati dipendenti dal tempo questo si traduce nella possibilità di avere una *timeline* che mostra 'quando' una certa caratteristica si manifesta negli inputs. In questo caso si ha equivarianza: infatti se spostiamo un evento degli input nel tempo, la rappresentazione di questo apparirà negli output spostata allo stesso modo. Anche per le immagini, la convoluzione produce una mappa 2D, che indica negli inputs 'dove' appaiono certe features (o caratteristiche) e se muoviamo l'immagine in input, la sua rappresentazione si muove della stessa quantità.

Si presta infine attenzione al fatto che l'operazione di convoluzione non è equivariante con ogni possibile funzione  $g$ , ma esistono funzioni, quali variazioni di scala o rotazioni dell'immagine in input, che non lo sono.

### 1.3.3 Funzioni di attivazione

Con *funzione di attivazione*  $g$  di un nodo  $i$  viene indicata una funzione vettoriale che calcola l'output di un nodo basandosi sui suoi inputs  $z$  e pesi  $W$ :

$$f_i(z) = g_i(W_i z + b_i) \quad (1.7)$$

I pesi  $W$  sono rappresentati sotto forma di matrice, mentre i parametri  $b$  (reali) sotto forma di vettore. Con  $f_i(z)$  si indica l'output prodotto dal layer  $i$ -esimo in risposta al vettore di input  $z$ .

Tali funzioni possono essere lineari e non; tra le più utilizzate vi è la funzione di attivazione *Rectified Linear Unit* (impiegata anche nello studio oggetto di questa tesi, nel [Capitolo 3](#)).

### 1.3.4 Rectified Linear Unit

La funzione lineare di attivazione *Rectified Linear Unit* (ReLU) è definita come segue:

$$g(z) = z^+ = \max(0, z) = \frac{z + |z|}{2} = \begin{cases} z & \text{se } z > 0, \\ 0 & \text{altrimenti,} \end{cases} \quad (1.8)$$

come è possibile notare l'equazione accetta in input valori numerici reali (scalari). Da questa è possibile definire allora la funzione vettoriale  $\mathbf{g}(\mathbf{z})$  come:

$$\mathbf{g}(\mathbf{z}) = [g(z_1), g(z_2), \dots, g(z_j), \dots, g(z_N)] \quad (1.9)$$

con  $N$  pari alla dimensione del vettore di input  $\mathbf{z} = [z_1, z_2, \dots, z_j, \dots, z_N]$ .

### 1.3.5 Pooling

Un layer di una rete convoluzionale consiste solitamente di tre livelli. Nel primo il layer esegue una serie di operazioni di convoluzione in parallelo per produrre un insieme di attivazioni lineari. Nel secondo, ciascuna attivazione lineare viene eseguita tramite una funzione di attivazione non lineare, quale ad esempio la funzione di attivazione *Rectified Linear Unit*.

Nel terzo livello il layer esegue la funzione di *pooling*. Una funzione di pooling sostituisce un nodo di output di una rete con una media statistica di nodi vicini. Ad esempio, la funzione di *max pooling* seleziona come output il nodo dal valore massimo all'interno di un rettangolo di dimensioni prestabilite. Altri esempi di funzioni di pooling includono la media su regioni rettangolari, la norma  $L^2$  o la media pesata in funzione della distanza dal pixel centrale della regione selezionata.

In ogni situazione, l'operazione di pooling migliora l'*invarianza* (approssimativa) rispetto a 'piccole' traslazioni degli inputs. Con invarianza per traslazioni degli inputs si intende che se trasliamo gli inputs di una quantità 'piccola', il valore dell'output sotto operazione di pooling non cambia.

L'invarianza locale per traslazione risulta essere una proprietà particolarmente utile nel caso in cui si voglia verificare la presenza o meno di certe caratteristiche negli inputs, piuttosto che la loro posizione specifica. Un esempio di applicazione di questa tecnica è la *face detection*: quando si vuole determinare se un'immagine contenga o meno una faccia (o, in generale, un certo oggetto) e non si vuole determinare la posizione di esso all'interno dell'immagine.

# Capitolo 2

## CALorimetric Electron Telescope (CALET)

In questo capitolo viene fornita una descrizione degli sciami elettromagnetici ed adronici, con alcune considerazioni sullo sviluppo laterale e longitudinale. Poi si passa alla descrizione del calorimetro CALET e dei risultati da questo ottenuti, in particolare riguardo nuclei di elio nei raggi cosmici.

### 2.1 Sciami di particelle

In fisica delle alte energie uno *sciame* di particelle è una “cascata” di particelle prodotta a catena dall’interazione di una particella *primaria* con la materia [6, 7, 8]. A partire dall’interazione della particella primaria vengono create altre particelle *secondarie*, che a loro volta interagiscono con la materia e danno origine ad altre particelle: questo processo si sviluppa creando sempre più particelle, di energia sempre inferiore, che infine verranno assorbite dalla materia.

Si possono avere sciami *elettromagnetici* e sciami *adronici*, a seconda che il fenomeno di interazione sia mediato dall’*interazione elettromagnetica* o dall’*interazione forte*.

### 2.1.1 Sciami elettromagnetici

Uno sciame elettromagnetico, come quello riportato in [Figura 2.1](#), ha origine quando un elettrone  $e^-$ , un positrone  $e^+$  o un fotone  $\gamma$  di alta energia interagisce con la materia. Ad alte energie (maggiori di qualche decina di MeV) i principali meccanismi di perdita di energia sono la *produzione di coppia* per i fotoni<sup>1</sup> e la *bremsstrahlung* per elettroni e positroni.

Questi due processi continuano fino a che i fotoni scendono sotto l'energia *di soglia* per la produzione di coppia e cominciano a dominare perdite di energia degli elettroni, non dovute a bremsstrahlung.

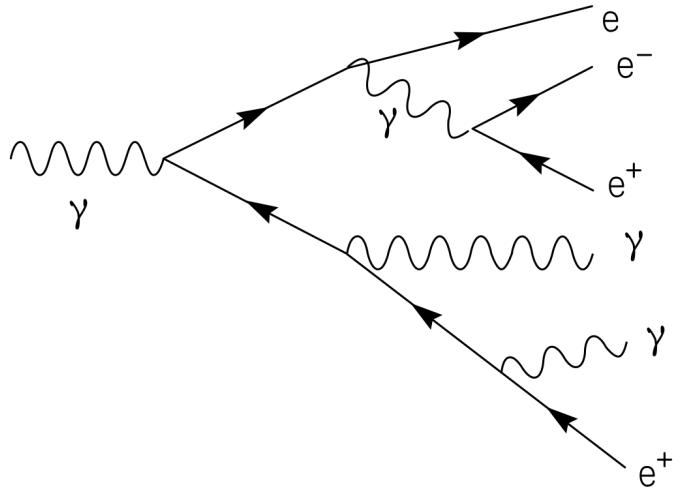


Figura 2.1: inizio di uno sciame elettromagnetico rappresentato con un diagramma di Feynman.

La scala caratteristica delle lunghezze su cui questi fenomeni si sviluppano viene indicata con la *lunghezza di radiazione*  $X_0$ . Nel caso di elettroni tale grandezza rappresenta lo spessore dopo cui l'energia dell'elettrone è  $1/e$  di quella iniziale:

$$E(x) = E_0 e^{-\frac{x}{X_0}}$$

$$E(X_0) = \frac{E_0}{e}$$

mentre nel caso di fotoni è legata al *cammino libero medio*  $\lambda_{\text{pair}}$  per produzione di coppia da:

$$\lambda_{\text{pair}} \approx \frac{9}{7} X_0$$

---

<sup>1</sup>Ad energie dell'ordine del MeV e superiori la *produzione di coppia* risulta dominante su altri processi di interazione quali l'*effetto fotoelettrico* e l'*effetto Compton*.

Per quanto segue risulterà evidente che  $X_0$  è anche la lunghezza che caratterizza lo sviluppo degli sciami elettromagnetici.

Quando un fotone o un elettrone di alta energia attraversano la materia, danno origine ad uno sciame di fotoni ed elettroni *secondari*, prodotti tramite *produzione di coppia* e *bremsstrahlung* rispettivamente. Ad ogni interazione successiva questi processi si ripetono per un numero sempre maggiore di particelle e continuano fin quando non viene raggiunta l'energia critica  $E_c$ , al di sotto della quale si interrompe il fenomeno di moltiplicazione delle particelle e diventano predominanti altri meccanismi di perdita di energia, quali l'*effetto fotoelettrico* e l'*effetto Compton* per i fotoni e la perdita di energia *per collisione* per gli elettroni.

### Produzione di coppia

Il processo di produzione di coppia, come riportato in [Figura 2.2](#), è una reazione in cui un fotone interagisce con la materia, convertendo la sua energia in materia ed antimateria, ovvero producendo una coppia elettrone/positrone ( $e^-$ ,  $e^+$ ).

La produzione di coppia può avvenire solo nel campo coulombiano del nucleo o di elettroni atomici, secondo le reazioni:

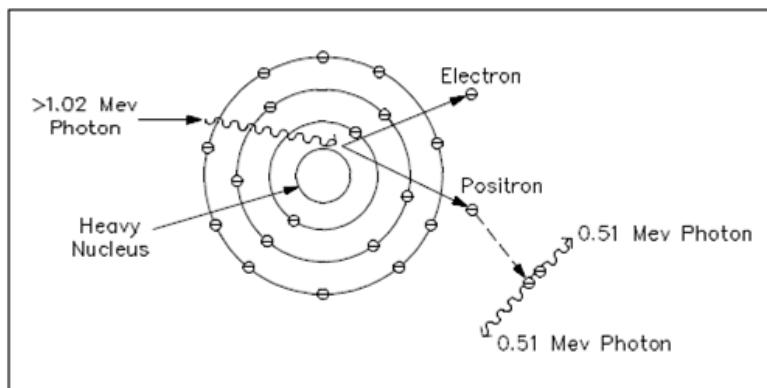
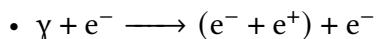
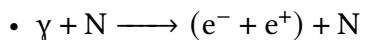


Figura 2.2: descrizione grafica del processo di produzione di coppia: si ha interazione di un fotone con energia pari all'energia di soglia  $E_\gamma^{\min}$  con il campo coulombiano di un nucleo e produzione di una coppia elettrone/positrone con energia equamente ripartita.

Nel processo si ha conservazione dell'energia e quindi, in accordo con l'equazione di equivalenza tra massa ed energia di Einstein  $E = mc^2$  dove  $E$  indica l'energia totale relativistica,  $m$  la massa relativistica e  $c$  la velocità della luce nel vuoto.

L'energia del raggio gamma incidente viene equamente ripartita nella particella e nella sua antiparticella corrispettiva, dato che queste hanno stessa massa. Inoltre, il processo di produzione è *a soglia*, ovvero è necessario che l'energia della particella incidente sia maggiore di un certo valore perché la reazione si verifichi. In particolare il fotone deve avere energia maggiore della somma dell'energia delle masse a riposo di un elettrone e di un positrone, quindi risulta che:

$$E_{\gamma}^{\min} = 2m_e c^2 \left(1 + \frac{m_e}{M}\right)$$

- $E_{\gamma}^{\min} \approx 2m_e c^2 = 1.022 \text{ MeV}$  (su nucleo con  $M \gg m_e$ )
- $E_{\gamma}^{\min} \approx 4m_e c^2 = 2.044 \text{ MeV}$  (su elettrone  $M = m_e$ )

### Bremsstrahlung

La bremsstrahlung (dal tedesco, “radiazione di frenamento”) è un processo di perdita di energia per irraggiamento nel campo elettrico del nucleo ed è particolarmente rilevante per gli elettroni. In [Figura 2.3](#) vi è una rappresentazione grafica del fenomeno. Essi possono perdere anche notevoli quantità della loro energia cinetica per emissione di fotoni e venire fortemente deviati.

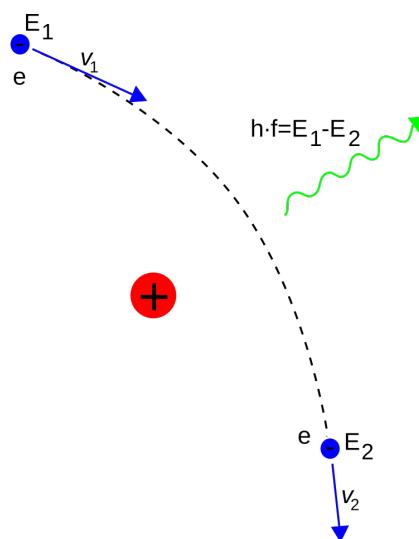


Figura 2.3: bremsstrahlung prodotta da un elettrone altamente energetico deflesso dal campo elettrico di un nucleo atomico.

L'elettrone deviato perde energia cinetica e, per il principio di conservazione dell'energia, emette quindi un fotone (radiazione gamma).

I fotoni emessi per bremsstrahlung sono caratterizzati da una distribuzione continua che diviene più intensa con l'aumentare dell'energia degli elettroni ed hanno quindi energia  $E_\gamma$ :

$$0 < E_\gamma \leq E_e = h\nu_{\max}$$

nell'ultima uguaglianza viene esplicitata la relazione tra l'energia cinetica dell'elettrone  $E_e$  e la frequenza massima della radiazione gamma emessa  $\nu_{\max}$ .

La probabilità di emissione di un fotone con energia  $E_\gamma$  ha una distribuzione:

$$P(E_\gamma) \propto \frac{1}{E_\gamma}$$

### Sviluppo longitudinale

Con sviluppo longitudinale si indica il percorso dei secondari nella direzione della particella primaria. Questo può essere descritto tramite la *lunghezza di radiazione*  $X_0$ , infatti la particella primaria (ad es. un fotone) produrrà una coppia elettrone/positrone, ciascuno con metà della sua energia, dopo aver attraversato circa uno spessore  $\approx X_0$  di materia, più precisamente  $\frac{9}{7}X_0$ . La coppia secondaria percorrerà un altro spessore  $\approx X_0$  e ciascuna particella produrrà un fotone per bremsstrahlung.

Si esprime dunque lo spessore attraversato  $x$  in lunghezze di radiazione,  $k = \frac{x}{X_0}$ . Questa descrizione dello sviluppo permette di stimare il numero di particelle  $N$  ad una profondità di  $k$  lunghezze di radiazione come

$$N(k) \approx 2^k$$

e l'energia delle particelle secondarie in funzione di quella della primaria come

$$E(k) \approx \frac{E_0}{N(k)} = \frac{E_0}{2^k}$$

Inoltre, sapendo che il processo di moltiplicazione è limitato dall'energia *critica*  $E_c$ , è possibile individuare a quale profondità  $k_{\max}$  lo sciame raggiunge il massimo sviluppo:

$$E(k_{\max}) = E_c \Rightarrow k_{\max} = \frac{\log(E_0/E_c)}{\log(2)}$$

in corrispondenza del quale il numero di secondari è massimo

$$N(k_{\max}) = \frac{E_0}{E_c}$$

Pertanto è possibile definire la *total track length* come la somma delle tracce di ionizzazione:

$$TL \propto X_0 \frac{E_0}{E_c}$$

mentre il numero di secondari prodotti fino a  $k_{\max}$  è

$$N_{\text{tot}} = \sum_{k=0}^{k_{\max}} 2^k = 2^{(k_{\max}+1)} - 1 \approx 2 \frac{E_0}{E_c}$$

Un metodo analitico più complesso per la descrizione dello sviluppo longitudinale degli sciami elettromagnetici, che prende in considerazione le diverse caratteristiche delle interazioni ed introduce quella che si chiama “Approssimazione B”, si può trovare nell’articolo di Bruno Rossi [9].

### Sviluppo laterale

Con sviluppo laterale si indica l’effetto di quei processi di interazione che causano un allargamento dello sciame nella direzione perpendicolare a quella di propagazione.

Il fenomeno ha simmetria cilindrica attorno all’asse di propagazione della particella primaria ed è determinato da processi di scattering multiplo nella materia, che dominano sul trascurabile allargamento dovuto a produzione di coppia e bremsstrahlung caratterizzato da un angolo  $\theta \sim \frac{m_e c^2}{E_s}$ , con  $E_s$  energia dei secondari.

Il cilindro di allargamento viene diviso in due componenti: una centrale stretta ed una esterna più larga.

La struttura centrale è dovuta ai processi di scattering coulombiano multiplo da parte di quegli elettroni che hanno energia sufficiente per allontanarsi dall’asse, ma non per irradiare fotoni. Questa componente ha dimensione laterale tipica data dal *raggio di Molière*:

$$R_{\text{Mol}} = \frac{E_{\text{Mol}}}{E_c} X_0$$

con  $E_{\text{Mol}} = m_e c^2 \sqrt{\frac{4\pi}{\alpha}} = 21.2 \text{ MeV}$

In media, il 90% dell’energia dello sciame è contenuta in un cilindro con raggio pari al raggio di Molière  $R(90\%) = R_{\text{Mol}}$ ; mentre il 95% è contenuta in un cilindro con raggio pari al doppio del raggio di Molière  $R(95\%) = 2R_{\text{Mol}}$ .

La struttura esterna è dovuta alla propagazione di fotoni di bassa energia e che quindi producono elettroni secondari disallineati dall’asse del primario tramite effetto fotoelettrico e Compton.

Spesso, per descrivere l'allargamento di uno sciame elettromagnetico su un piano perpendicolare alla direzione di propagazione del primario ed alla profondità  $k$ , viene effettuata una *parametrizzazione a due esponenziali*, uno per la struttura interna e l'altro per l'esterna:

$$f(x, k) = Ae^{-\frac{x}{b_1}} + Be^{-\frac{x}{b_2}}$$

con  $x$  distanza dall'asse del primario e  $b_1$  e  $b_2$  lunghezze di attenuazione caratteristiche interna ed esterna, rispettivamente.

### 2.1.2 Sciami adronici

Si parla di sciami adronici quando i meccanismi di produzione di particelle secondarie sono mediati dall'*interazione forte*. Uno sciame adronico è generato da una particella adronica (ovvero nucleoni e altre particelle costituite da quark).

Differentemente dagli sciami elettromagnetici, la descrizione di uno sciame adronico costituisce un problema estremamente complesso: non esistono infatti descrizioni analitiche di tale fenomeno e si ricorre sd approcci quantitativi basati su metodi numerici e su simulazioni Monte Carlo.

L'interazione di adroni energetici (carichi o neutri, p, n,  $\pi$ , K) con i nuclei di un materiale è determinata da processi nucleari anelastici che danno luogo a grandi fluttuazioni nell'energia depositata da ogni sciame.

L'interazione avviene in due fasi: inizialmente si ha una veloce *cascata internucleare* seguita poi da una lenta fase di *evaporazione*. Nella prima fase si ha l'interazione tra adrone e un nucleone, che può considerarsi quasi libero nel nucleo. Il nucleone acquista energia e collide su altri nucleoni: si forma quindi una cascata di nucleoni veloci, pioni e altri adroni. Delle particelle presenti nella cascata, alcune fuoriescono dal nucleo, mentre altre vi rimangono e ridistribuiscono l'energia acquistata. In una tipica interazione adronica circa metà dell'energia incidente è convertita nella produzione di particelle secondarie come ad esempio pioni carichi e neutri.

Un'importante frazione dei secondari prodotti in uno sciame adronico è costituita da pioni neutri  $\pi^0$ , i quali decadono elettromagneticamente in due fotoni e producono quindi uno sciame elettromagnetico. Da questo segue che uno sciame adronico è sempre seguito da uno sciame elettromagnetico.

Più nel dettaglio, se l'energia del primario è  $E$  (espresso in GeV) il numero medio di pioni neutri prodotti è  $n_0 \approx 5 \ln(E) - 4.6$ . La frazione di energia adronica depositata sotto forma di cascata elettromagnetica è in media  $f_{em} \approx 1 - E^{-0.15}$ .

La *cascata internucleare di spallazione* è fatta di leading particles (p, n,  $\pi$ , K) con direzione circa quella della particella incidente e di elevata energia. Il nucleo intermedio risultante si dissecchia e un certo numero di particelle sono *evaporate* (p, n, a) cioè emesse isotropicamente nel sistema di riferimento del laboratorio. La restante energia

viene depositata per ionizzazione (tra 40% e 60%), generazione di neutroni (tra 10% e 15%), rottura nucleare e rinculo di frammenti nucleari (tra 30% e 45%) e produzione di fotoni ( $\sim 3\%$ ) di qualche MeV.

## Sviluppo longitudinale

Ad energie maggiori di qualche GeV (alte energie) la sezione d’urto anelastica dipende poco dall’energia e dal tipo di particella incidente, ma dipende soprattutto dal bersaglio:

$$\begin{aligned}\sigma_{\text{anel}} &= 35A^{0.7} \text{ [mb]} \\ \sigma_{\text{tot}} &= \sigma_{\text{el}} + \sigma_{\text{anel}}\end{aligned}$$

con  $\sigma_{\text{el}}$  e  $\sigma_{\text{anel}}$  sezione d’urto elastica ed anelastica su nucleo di numero di massa  $A$ .

Lo sviluppo longitudinale dello sciame adronico è proporzionale alla *lunghezza di interazione nucleare*  $\lambda_I$ :

$$\lambda_I = \frac{A}{N_A \rho \sigma_{\text{anel}}} \approx 35 \frac{A^{1/3}}{\rho} \text{ [cm]}$$

dove  $\rho$  è la densità del materiale e  $N_A$  il numero di Avogadro. Lo sviluppo longitudinale segue inizialmente una crescita rapida, dovuta anche al contributo elettromagnetico, per poi raggiungere un massimo e decrescere lentamente tramite interazioni adroniche.

Il massimo sviluppo dello sciame si ha alla profondità (misurata in *lunghezze di interazione nucleare*  $\lambda_I$ ):

$$k_{\max} \approx 0.2 \ln(E) + 0.7$$

ed il 95% del contenimento longitudinale di uno sciame adronico si ha, mediamente, alla profondità:

$$L_h(95\%) \approx k_{\max} + 2.5\lambda_{\text{att}}$$

dove  $\lambda_{\text{att}} \propto E^{0.13}$  è la *lunghezza di attenuazione* che descrive il decadimento dello sciame una volta superato il massimo  $k_{\max}$ .

Tipicamente i valori di  $X_0$  degli sciami elettromagnetici sono inferiori dei valori di  $\lambda_I$  per gli sciami adronici, ovvero quest’ultimi si sviluppano su lunghezze maggiori.

Materiale	$X_0$ [cm]	$E_c$ [MeV]	$R_M$ [cm]	$R_M/X_0$	$\lambda_I$ [cm]
Be	35.28	113.70	6.60	0.19	42.10
Si	9.37	40.19	4.94	0.53	46.52
Al	8.90	42.70	4.42	0.50	39.70
Fe	1.76	21.68	1.72	0.98	16.77
BGO	1.12	10.50	2.26	2.02	22.32
Cu	1.44	19.42	1.57	1.09	15.32
W	0.35	7.97	0.93	2.66	9.95
Pb	0.56	7.43	1.60	2.86	17.59
U	0.32	6.65	1.00	3.13	11.03
PWO	0.89	9.64	1.96	2.20	20.27
PVT	42.54	94.11	9.59	0.23	78.80
PS	41.31	93.11	9.41	0.23	77.07

### Sviluppo laterale

Anche lo sviluppo laterale è maggiore in uno sciame adronico rispetto ad uno elettromagnetico, questo è dovuto agli ampi angoli a cui vengono prodotte le particelle secondarie. Si ha quindi che un'importante frazione di energia del primario è depositata lontana dall'asse dello sciame.

In modo similare al caso elettromagnetico, anche in questo caso è possibile individuare due strutture, una interna ed una esterna, descrivibili tramite la somma di due esponenziali. Il contenimento radiale del 95% dell'energia depositata dallo sciame si ha in un cilindro di raggio pari a:

$$R_h(95\%) \approx \lambda_I$$

Infine, come visto per lo sviluppo longitudinale per le differenze tra  $X_0$  e  $\lambda_I$ , lo sviluppo laterale di uno sciame adronico risulta maggiore di quello di uno sciame elettromagnetico, ovvero sono più ‘larghi’.

### 2.1.3 Calorimetria

Per compiere misure sull'energia depositata da fenomeni di attraversamento di porzioni di materia da parte di sciami di particelle si utilizzano strumenti chiamati *calorimetri*. Un calorimetro è un rivelatore di particelle che misura l'energia di una particella: l'energia può essere misurata internamente richiedendo il totale contenimento dello sciamo di particelle, o solo in parte.

Solitamente i calorimetri vengono divisi sia trasversalmente al fine di acquisire informazioni, oltre che sull'energia della particella, anche sulla sua direzione di attraversamento, sia longitudinalmente al fine di compiere misure volte all'identificazione della particella.

L'energia totale depositata è misurata come somma dei segnali (ad esempio luce di scintillazione) prodotti dai secondari dello sciamo negli elementi attivi del calorimetro, e risulta proporzionale all'energia della particella incidente.

Per permettere alla particella di depositare tutta la sua energia, il calorimetro deve essere costituito di materiale di elevata densità e spessore; siccome la profondità necessaria per fermare completamente la particella cresce logaritmicamente al crescere dell'energia, si ha la possibilità di assemblare strumenti molto compatti anche per analizzare le alte energie.

Particolarmente importante in fase di progettazione è anche il contenimento laterale, ovvero il calorimetro deve essere correttamente dimensionato, in modo che sia in grado di contenere la moltitudine di particelle secondarie generate nello sciamo che si disperdono ad angoli grandi rispetto all'asse del primario. Avere un buon dimensionamento permette di incrementare la risoluzione energetica dello strumento.

I calorimetri possono essere *elettromagnetici* o *adronici*, a seconda della natura dello sciamo che si vuole rivelare. Nel primo caso, particelle, quali elettroni, positroni e fotoni che hanno interazioni elettromagnetiche con la materia; mentre nel secondo protoni, pioni e nuclei, che interagiscono con la materia tramite interazione forte.

Questi due tipi di calorimetro possono essere distinti a loro volta in altre due sottocategorie, a seconda che il calorimetro sia *omogeneo* o *a campionamento*. I primi sono costituiti da un blocco unico e omogeneo di materiale sensibile, il cui vantaggio è quello di avere un'ottima risoluzione in energia dato che tutto il volume è materiale sensibile; i secondi sono invece costituiti da strati di materiale denso in cui viene prodotto lo sciamo e da strati di materiale sensibile in cui viene misurata l'energia depositata. Il vantaggio è di poter essere resi più compatti rispetto ad un calorimetro omogeneo, con il difetto però di una peggiore risoluzione in energia.

## 2.2 Missione CALET

La missione CALET è un esperimento spaziale finanziato dalla *Japan Aerospace Exploration Agency* (JAXA), dall'*Italian Space Agency* (ASI) e dalla *National Aeronautics and Space Administration* (NASA). Partecipano alla collaborazione internazionale le università italiane degli Studi di Siena (capofila della partecipazione italiana alla missione), di Pisa, Firenze, e Roma Tor Vergata.

CALET è stato portato in orbita il 19 agosto 2015 dal razzo giapponese *H-II Transfer Vehicle* (HTV-5) ed installato sul *Japanese Experiment Module Exposure Facility* (JEM-EF), la piattaforma esterna del modulo giapponese (KIBO) [10, 11] della *International Space Station* (ISS) il 24 agosto, come è possibile vedere in [Figura 2.4](#). La JEM-EF è una piattaforma di dimensioni  $4 \times 6 \text{ m}^2$  direttamente esposta allo spazio esterno che può ospitare fino a 10 moduli da esperimenti allo stesso tempo e fornire loro energia elettrica, scambiatori termici e sistemi per trasmissione dati; CALET è posizionato alla porta numero 9.

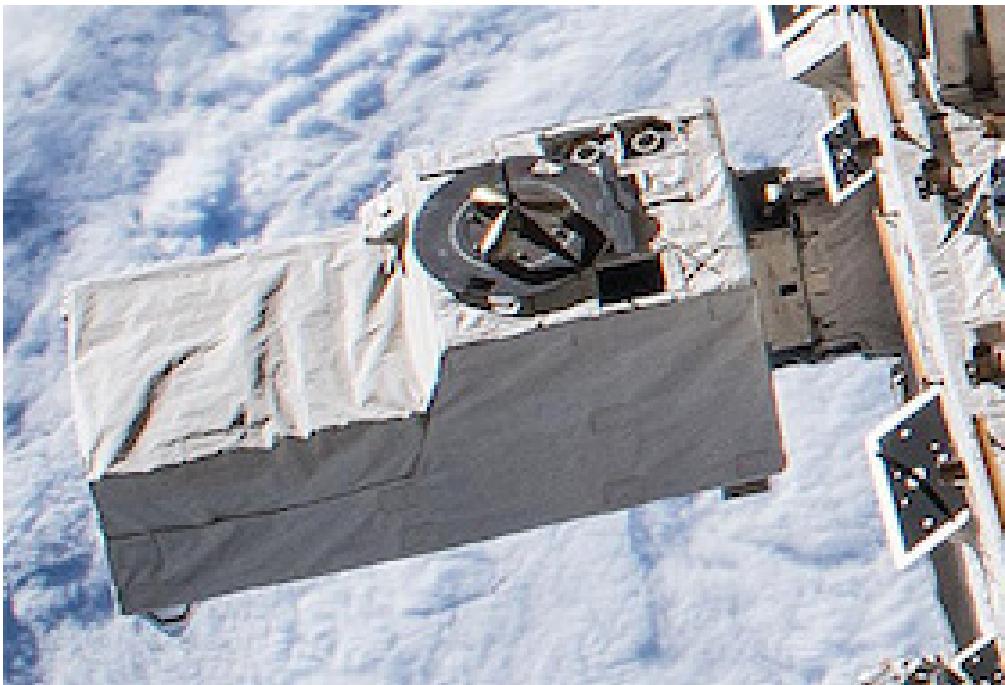


Figura 2.4: il calorimetro CALET alla porta numero 9 della piattaforma JEM-EF della ISS.

L'obiettivo principale della missione è la ricerca di indizi della presenza di sorgenti astrofisiche di elettroni di alta energia in prossimità della Terra o di indizi di materia oscura, tramite la misura dello spettro di  $e^- + e^+$  nel range di energia da 1 GeV fino a circa

20 TeV. CALET inoltre studia l'accelerazione e propagazione dei raggi cosmici (CR) nella Galassia, effettuando misure dirette degli spettri energetici e della composizione elementare dei nuclei contenuti nei CR. Le principali caratteristiche dello strumento sono una eccellente risoluzione in energia e un'eccellente capacità di separazione tra adroni ed elettroni.

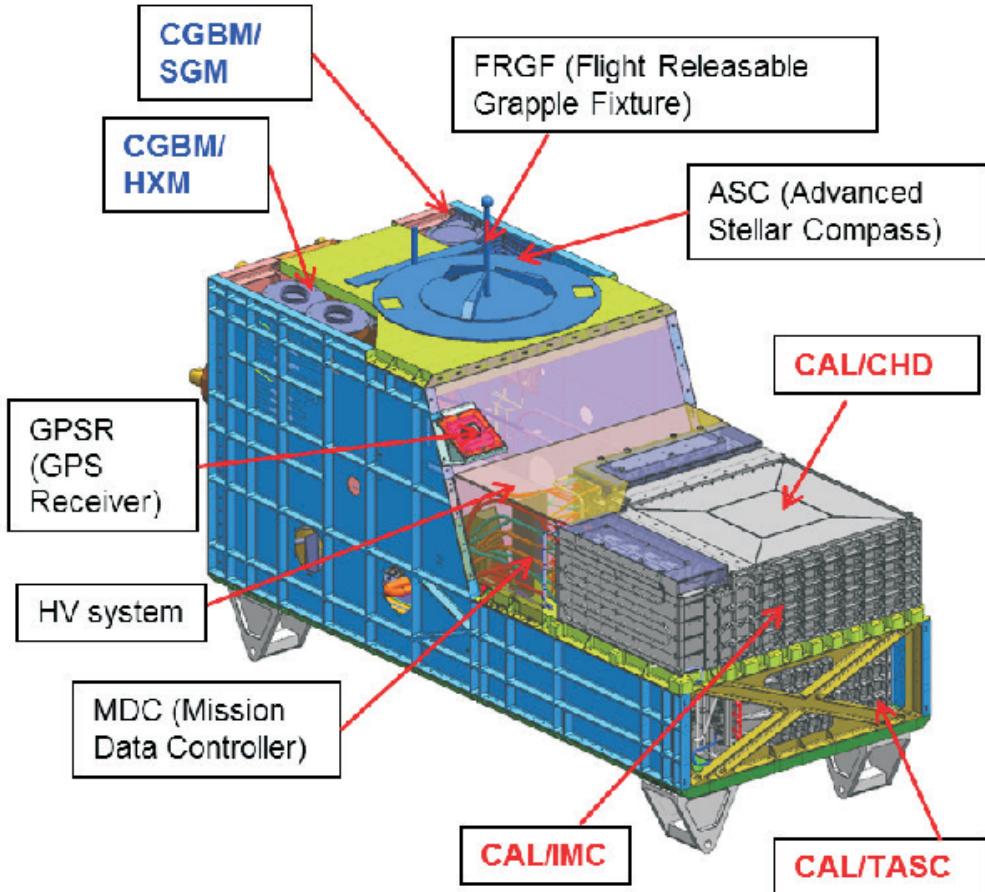


Figura 2.5: schema dello strumento CALET.

Lo strumento è composto dal detector principale, il calorimetro, e da ulteriori sistemi, la cui posizione è indicata in [Figura 2.5](#). I principali sono riportati di seguito:

- *Flight Releasable Grapple Fixture* (FRGF), punto di ancoraggio per il braccio meccanico e astronauti;
- *Mission Data Controller* (MDC), computer per la gestione del rivelatore e dei dati;
- *Advance Stellar Compass* (ASC) e ricevitore GPS (GPSR) per il controllo del puntamento e della posizione;

- *CALET Gamma ray Burst Monitor* (CGBM), sub-detector per misurare i raggi X e raggi gamma in un range di energia fuori sensibilità del telescopio principale.

### 2.2.1 Calorimetro CALET

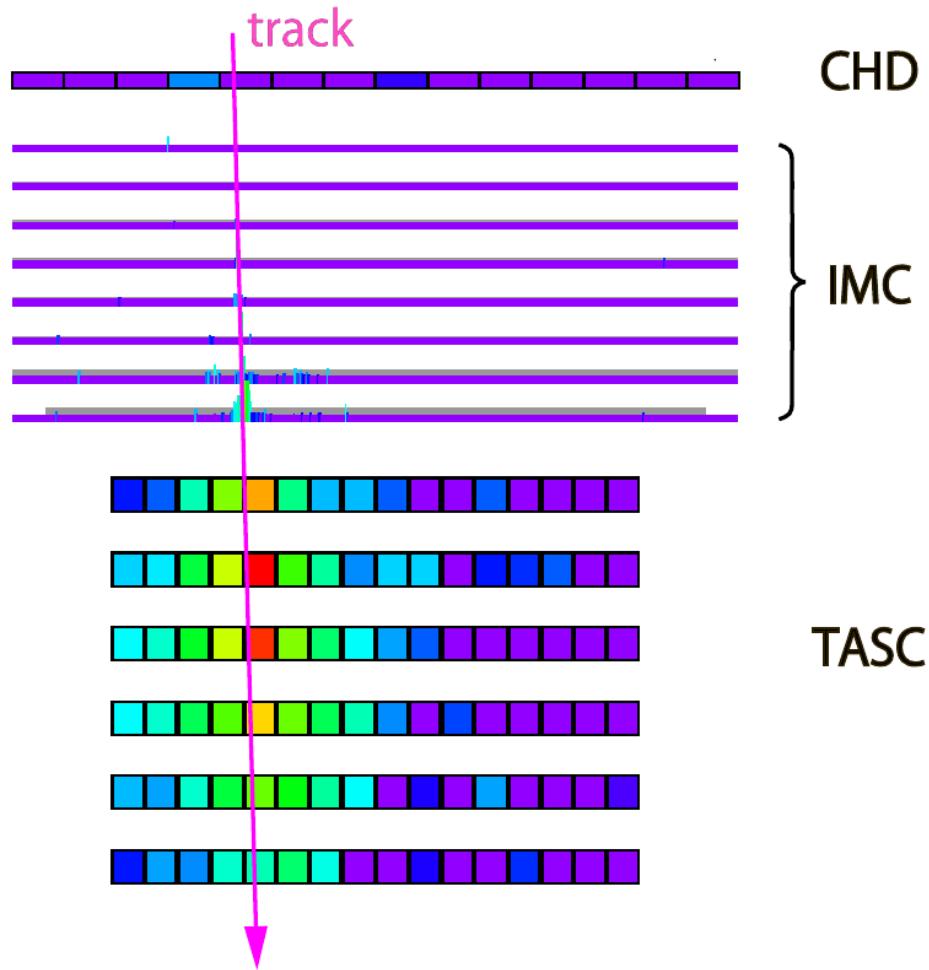


Figura 2.6: schema di una vista bidimensionale del calorimetro.

Lo strumento principale di CALET è un rilevatore calorimetrico con uno spessore di 30 lunghezze di radiazione. Il campo visivo è di circa 45 gradi dallo zenith e il fattore geometrico è  $1200 \text{ cm}^2 \text{ sr}$ . Il calorimetro, come mostrato in Figura 2.6 ed in maggiore dettaglio in Figura 2.7, è suddiviso in tre sotto-sistemi di rivelatori disposti uno sopra l'altro [12]:

- *CHARGE Detector* (CHD), posizionato in cima a CALET, misura la carica elettrica  $Z$  della particella incidente, ovvero compie l'identificazione degli elementi chimici presenti nei raggi cosmici;
- *IMaging Calorimeter* (IMC), posto sotto al CHD e sopra al TASC, fornisce un'immagine del profilo iniziale dello sciamone e, tramite 16 piani di fibre scintillanti, ricostruisce la traiettoria della particella incidente. Inoltre è anche in grado di misurare la carica della particella, contribuendo alla sua identificazione;
- *Total AbSorption Calorimeter* (TASC), misura l'energia totale dello sciamone generato dalla particella primaria e permette di distinguere tra sciami elettromagnetici ed adronici.

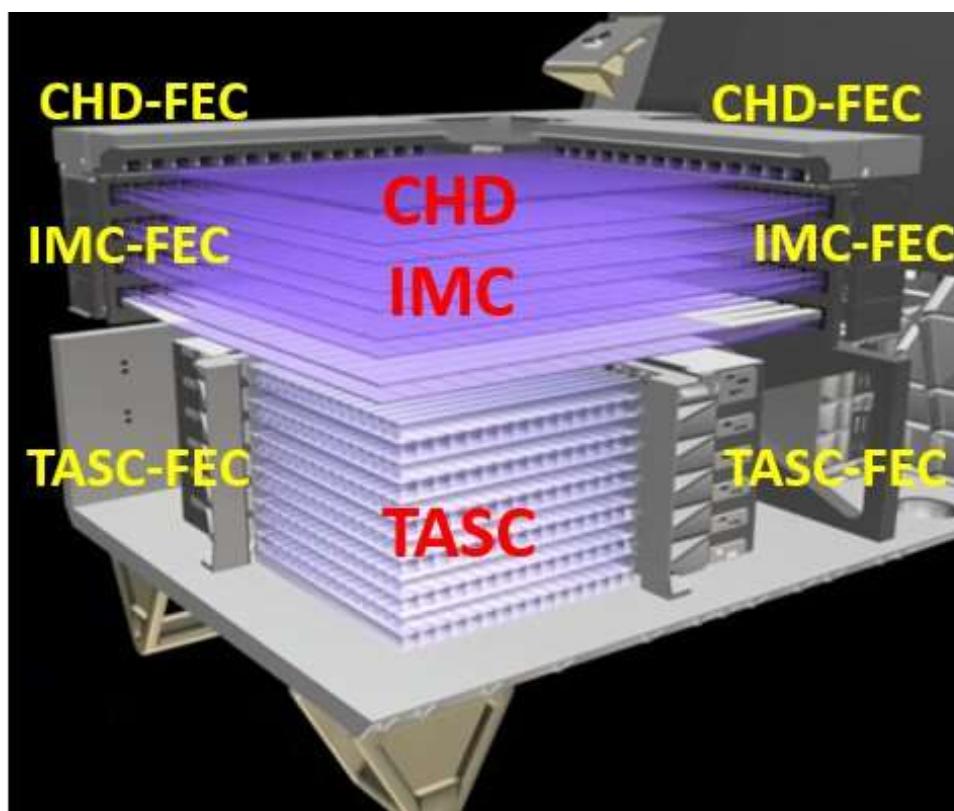


Figura 2.7: schema tridimensionale del calorimetro e dei sotto-sistemi di rivelazione.

### **CHARGE Detector (CHD)**

Il *CHARGE Detector* (CHD) è il primo rivelatore a venire attraversato dalla particella incidente. Ha la funzione di misurare il valore assoluto della carica della particella primaria, al fine di poterla identificare. La misura della carica si basa sulla dipendenza da

$Z^2$  del valore misurato della ionizzazione specifica di ioni ultra-relativistici di numero atomico  $Z$  (si veda la *formula di Bethe-Bloch*).

Quindi, poiché tramite la *formula di Bethe-Bloch* è possibile risalire al valore di  $Z^2$ , ciò che si ricava è il valore assoluto della carica della particella, ma non il suo segno: il CHD non permette dunque di distinguere tra particelle con carica uguale in modulo, ma di segno opposto, ad esempio elettroni e positroni.

La risoluzione in carica viene espressa in termini dell'unità elementare di carica  $e$ . I valori di risoluzione nel CHD sono compresi tra  $0.15e$  e  $0.3e$  per nuclei con  $2 < Z < 26$ .

Il CHD è formato da due strati (*layers*) ortogonali di 14 strisce (*paddle*) di scintillatori plastici ( $32 \text{ mm} \times 450 \text{ mm} \times 10 \text{ mm}$  ciascuno). L'ortogonalità dei due piani, ovvero il fatto che siano ruotati di  $90^\circ$  uno rispetto all'altro, permette di stimare la posizione di incidenza dei raggi cosmici, le coordinate  $(x, y)$  sul piano del CHD. La segmentazione in strisce permette di ridurre il contributo al segnale dovuto ai depositi di energia delle particelle di *backscattering* prodotte dallo sciame che si sviluppa nell'IMC e nel TASC.

Il *backscattering* è una problematica comune a tutti gli strumenti calorimetrici alle alte energie: in CALET si cerca di risolvere questo problema sfruttando la ricostruzione della traccia effettuata nell'IMC in modo da selezionare nel CHD solo la striscia effettivamente colpita dalla particella primaria, evitando di confonderla con quelle colpite invece dalla particelle secondarie di backscattering [13].

## IMaging Calorimeter (IMC)

L'*IMaging Calorimeter* (IMC) è un calorimetro *a campionamento*, ovvero diviso in strati di materiale denso in cui viene fatto sviluppare lo sciame e da strati di materiale sensibile in cui viene misurata l'energia depositata; è formato da 8 piani (*layers*) di fibre scintillanti (*SciFi*) in polistirene (PS) alternati con 7 piani assorbitori di tungsteno (W).

In ciascuno degli 8 piani si hanno 2 strati di fibre scintillanti, ruotate l'una rispetto all'altra di  $90^\circ$ , ovvero orientate secondo gli assi ortogonali  $x$  e  $y$ , fornendo così due viste trasversali dei piani  $xz$  e  $yz$  rispettivamente.

In ciascuno strato si hanno 448 fibre, di lunghezza 448 mm e sezione di diametro 1 mm. Dunque le dimensioni dei piani di fibre sono  $448 \text{ mm} \times 448 \text{ mm} \times 1 \text{ mm}$  ciascuno e due piani successivi si trovano ad una distanza di circa 2 cm. Ad eccezione del primo piano di fibre, ognuno degli altri piani si trova immediatamente sotto a uno dei 7 piani di tungsteno che hanno spessori diversi a seconda della profondità. Spostandosi infatti dall'alto verso il basso i primi cinque piani hanno spessore 0.07 cm (corrispondente a  $0.2 X_0$ ), mentre gli ultimi due hanno spessore 0.35 cm (corrispondente a  $1 X_0$ ). Lo spessore totale dell'IMC è quindi di  $\sim 3X_0$  lunghezze di radiazione.

Grazie a questa struttura il calorimetro IMC permette di far sì che il raggio cosmico inizi e sviluppi lo sciame nei piani di tungsteno, mentre tramite i piani SciFi fornisce

immagini dello sciame ad alta risoluzione spaziale, utili per effettuare l'identificazione ed il tracciamento della particella primaria.

L'IMC viene infatti impiegato come *preshower*: al suo interno fotoni ed elettroni primari sono indotti a sciamare dai piani di tungsteno, mentre le fibre SciFi producono un'immagine della parte iniziale di sviluppo dello sciame; al contrario, se le particelle primarie incidenti sono protoni, solo una frazione ridotta di questi interagisce e sciamma nei piani dell'IMC, per via della ridotta profondità dello strumento in termini di lunghezza di interazione.

Le caratteristiche sopra esposte possono essere riassunte dicendo che l'IMC permette di:

- ricostruire la traiettoria della particella incidente;
- determinare il punto di inizio dello sciame elettromagnetico;
- discriminare la particella incidente primaria da quelle di backscattering.

L'abilità dell'IMC di poter ricostruire la traiettoria della particella incidente, quindi il punto di inizio dello sciame, è di notevole importanza per poter compiere misure volte ad identificare gli elettroni rispetto al 'fondo' di fotoni e protoni. Infatti nel calorimetro gli sciami originati da elettroni (o positroni) e fotoni sono simili sotto molti aspetti e si differenziano solo per il fatto che negli eventi di sviluppo dello sciame originati dai fotoni le fibre che precedono il punto di inizio dello sviluppo non registrano un rilascio di energia compatibile con quello di una particella carica al minimo di ionizzazione. Questo permette di identificare gli sciami originati dai fotoni.

Nel caso dei protoni invece, l'inizio dello sciame si verifica a profondità maggiori dello spessore totale dell'IMC [14, 15].

### Total AbSorption Charge (TASC)

Il *Total AbSorption Calorimeter* (TASC) è un calorimetro *omogeneo* progettato per contenere completamente lo sviluppo degli sciami gli sciami, ovvero è costituito da un blocco unico e omogeneo di materiale sensibile, il cui vantaggio è quello di avere un'ottima risoluzione in energia, dato che tutto il volume è materiale sensibile. Grazie a tale struttura il TASC misura quindi l'energia totale dello sciame generata dalla particella primaria e permette di distinguere tra sciami elettromagnetici ed adronici, separando gli eventi dovuti agli elettroni dal fondo adronico, formato quasi esclusivamente da protoni.

Al suo interno si trovano 12 piani di barre di cristalli di tungstato di piombo ( $\text{PbWO}_4$ ) a forma di parallelepipedo a base quadrata ( $20 \text{ mm} \times 20 \text{ mm} \times 320 \text{ mm}$  ciascuna). I piani sono ruotati alternativamente l'uno rispetto all'altro di  $90^\circ$ , per fornire le due viste  $xz$  e  $yz$  dello sviluppo dello sciame.

Ognuno dei 12 piani contiene 16 barre, per un totale di 192 cristalli di tungstato di piombo all'interno del calorimetro TASC.

Il TASC ha una profondità di 24 cm, che corrispondono a  $27 X_0$  lunghezze di radiazione e  $1.2 \lambda_I$  lunghezze di interazione nucleare.

Ciò implica un contenimento quasi totale per gli sciami elettromagnetici e al contrario una notevole perdita di energia dal fondo del calorimetro per gli sciami adronici perché il loro contenimento longitudinale è parziale. Questo fatto viene sfruttato per ottenere un'alta reiezione del fondo di protoni nelle misure dei flussi elettronici.

Per poter misurare elettroni fino a decine di TeV il range dinamico richiesto è particolarmente elevato: il sistema deve infatti essere in grado di rivelare depositi di energia da 0.5 MIP (Minimum Ionizing Particle) a circa  $10^5$  MIP per elettroni da 10 TeV. Tale range dinamico è quindi circa di 6 ordini di grandezza e viene coperto utilizzando due fotodiodi di area sensibile diversa per la lettura di ciascun cristallo, e una elettronica di readout a doppio guadagno per ogni fotorivelatore [14]. In [Figura 2.8](#) viene mostrato l'interno del calorimetro specificando le dimensioni dei sotto-sistemi che lo compongono.

Si presta attenzione al fatto che CALET è uno strumento puramente calorimetrico e non è dotato di uno spettrometro: sia la misura dell'energia depositata che la reiezione del fondo di protoni vengono effettuate dal TASC, che, per questo motivo, costituisce il 'cuore' di tutto lo strumento.

CALET è in grado di misurare lo spettro elettronico con una risoluzione in energia dell'ordine del  $2 \sim 3\%$  e con un potere di reiezione del fondo dell'ordine di  $10^5$  [16].

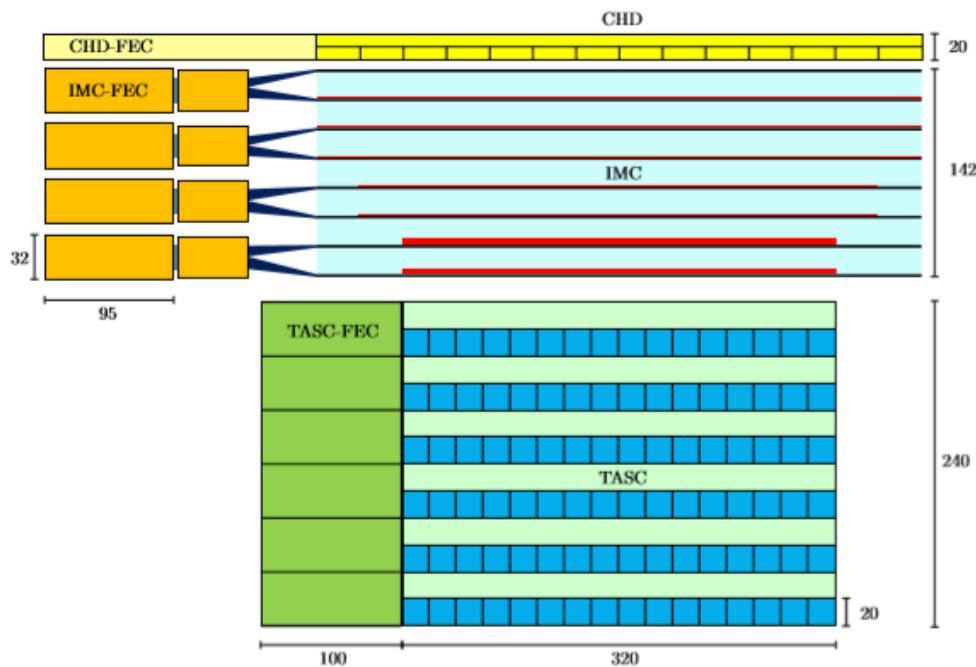


Figura 2.8: dimensioni e posizioni dei sotto-rivelatori nel calorimetro di CALET, misure in mm e vista  $yz$ .

### Sistemi di trigger

In CALET, per le operazioni di presa e salvataggio dei dati vi sono diversi trigger hardware, configurabili tramite discriminatori a soglia (*thresholds*) nei tre rivelatori precedentemente descritti (CHD, IMC e TASC).

Le principali modalità di trigger utilizzate per effettuare raccolta dati in CALET sono:

- *High Energy shower trigger (HE)*, per rivelare con efficienza  $\epsilon > 95\%$  elettroni di energia  $E > 10$  GeV. Questo trigger permette di selezionare quegli eventi che danno origine a sciami all'interno del calorimetro;
- *Low Energy shower trigger (LE)*, per rivelare elettroni di energia ( $1 < E < 10$ ) GeV nelle regioni polari e raggi gamma di bassa energia ( $E > 1$ ) GeV a basse latitudini;
- *Single trigger*, utilizzato per calibrare lo strumento tramite l'acquisizione di dati di particelle non interagenti.

## 2.3 Risultati di CALET: spettro di nuclei di elio

Da “Direct Measurement of the Cosmic-Ray Helium Spectrum from 40 GeV to 250 TeV with the Calorimetric Electron Telescope on the International Space Station” [17] ed il *supplemental material*. Si riportano i principali risultati esposti nell’articolo.

I dati di volo oggetto d’analisi nello studio sopra citato sono stati raccolti dal 13 ottobre 2015 al 30 aprile 2022, (2392 giorni). Il tempo totale di osservazione diretta ammonta a 48459.7 ore e la frazione di presa diretta rispetto al tempo totale è pari all’84.4%. L’analisi dei dati è simile alla procedura di analisi di protoni [18, 19], carbonio C ed ossigeno O [20], ferro Fe [21], e nichel Ni [22].

Una ricostruzione ad alta efficienza della direzione delle tracce adroniche è di primaria importanza per la misura del flusso. Per questo motivo, come per l’analisi dello spettro dei protoni, anche per quello dei nuclei di elio si ricorre all’algoritmo Kalman filter combinatorio (KF, in [sezione A.1](#)), che fornisce buoni risultati per la ricostruzione della direzione delle tracce delle particelle incidenti.

L’energia dello sciamo generato da ciascun evento è calcolata a partire dalla somma dei depositi di energia nel TASC ( $E_{TASC}$ ).

Sono state effettuate simulazioni Monte Carlo, basate sul pacchetto di simulazione EPICS [23], per la stima delle efficienze di ricostruzione. La calibrazione dello strumento è stata effettuata in una serie di *beam tests* al CERN-SPS [24, 25].

L’efficienza del trigger e la risposta in energia ottenute dalle simulazioni MC sono state regolate utilizzando i risultati del beam test ottenuti nel 2015 con fasci di ioni di energia pari a 13, 19, e 150 GeV/n. Per sciami di nuclei di elio è stata applicata una correzione di 10.4% a 13 GeV/n e di 8% a 19 GeV/n, mentre di 3.2% ad energie  $E > 150$  GeV/n. I fattori correttivi per valori energetici intermedi (non misurati al CERN) sono stati calcolati eseguendo una interpolazione logaritmico-lineare. Non sono invece state applicate correzioni all’efficienza del trigger poiché le misure del beam test sono risultate consistenti con le simulazioni MC.

Nell’analisi degli adroni, in particolare alle alte energie, dove non è possibile effettuare un beam test di calibrazione dello strumento, è necessario effettuare un confronto tra differenti modelli MC. A questo scopo, sono state eseguite altre simulazioni MC, utilizzando FLUKA [26] e confrontate con quelle effettuate tramite EPICS.

Viene poi applicata una preselezione di eventi ‘ben ricostruiti’ e ‘ben contenuti’ all’interno del calorimetro, prima ancora di procedere all’identificazione della carica, così da ridurre al minimo la contaminazione di fondo dei nuclei di elio selezionati.

Di seguito vengono riportati i principali criteri di selezione (“tagli”) applicati sui dati:

- *Trigger*: vengono selezionati solo quegli eventi che abbiano ‘acceso’ l’High Energy shower trigger (HE). La coerenza tra i dati di volo (FD) ed i dati simulati MC per gli eventi selezionati con il trigger HE si ottiene applicando un trigger offline che richiede condizioni più severe più severe rispetto al trigger onboard. Questo permette di rimuovere quegli errori residui dovuti alle variazioni posizionali e temporali del guadagno del rivelatore.
- *Qualità della traccia*: si richiede che gli eventi selezionati abbiano una ‘buona’ traccia primaria ricostruita in entrambe le viste con l’algoritmo KF. In dettaglio, si richiede un certo numero minimo di punti ricostruiti per ogni segmento della traccia ed inoltre si applica un taglio anche sul valore del  $\chi^2$  del fit. In questo modo si raggiunge una risoluzione angolare per i nuclei di elio di  $\sim 0.1^\circ$  ed una risoluzione spaziale sul punto di impatto (IP) di  $\sim 400 \mu\text{m}$  sul piano superiore del CHD.
- *Condizioni geometriche*: si richiede che gli eventi selezionati siano di “accettanza A”, ovvero attraversino l’intero calorimetro dal piano superiore del CHD al fondo del TASC, ed entro 2 cm dalle pareti laterali del TASC.
- *Ricerca degli elettroni*: viene applicato un taglio di scarto degli elettroni basato sulla quantità frazionaria nota come ‘concentrazione di Molière lungo la traccia’ e calcolata, prima comando tutti i depositi di energia all’interno di un raggio di Molière attorno a ciascuna fibra dell’IMC attraversata dalla traccia, poi normalizzando rispetto alla somma dei depositi totali di energia nell’IMC. Imponendo che questa quantità sia  $< 0.75$  e la frazione di energia del TASC depositata nell’ultimo layer sia  $> 0.01$ , risulta che più del 90% degli elettroni sono scartati, pur mantenendo un’efficienza molto alta per nuclei di elio ( $> 99.9\%$  per  $E > 50 \text{ GeV}$ ).

L’ampiezza degli intervalli di energia è stata scelta in modo da essere proporzionale alla risoluzione del TASC. In questo modo lo spettro energetico, riportato in [Figura 2.9](#), si ottiene dalla distribuzione energetica come segue:

$$\Phi(E) = \frac{N(E)}{\Delta E \times \epsilon(E) \times S\Omega \times T}$$

$$N(E) = U[N_{\text{obs}}(E_{\text{TASC}}) - N_{\text{bg}}(E_{\text{TASC}})]$$

dove  $\Delta E$  indica l’ampiezza degli intervalli di energia;  $E$  è l’energia cinetica della particella, calcolata come la media geometrica dell’estremo inferiore e superiore dell’intervallo;  $N(E)$  è il numero di conteggi nell’intervallo relativo ad una certa  $E$ ;  $\epsilon(E)$  il parametro globale di efficienza;  $T$  è il livetime dello strumento;  $S\Omega$  l’accettanza geometrica;  $U$  è la procedura di *unfolding*;  $N_{\text{obs}}(E_{\text{TASC}})$  è il contenuto dell’intervallo della

distribuzione osservata di energia (incluso il fondo); e  $N_{\text{bg}}(E_{\text{TASC}})$  gli eventi di fondo nello stesso intervallo.

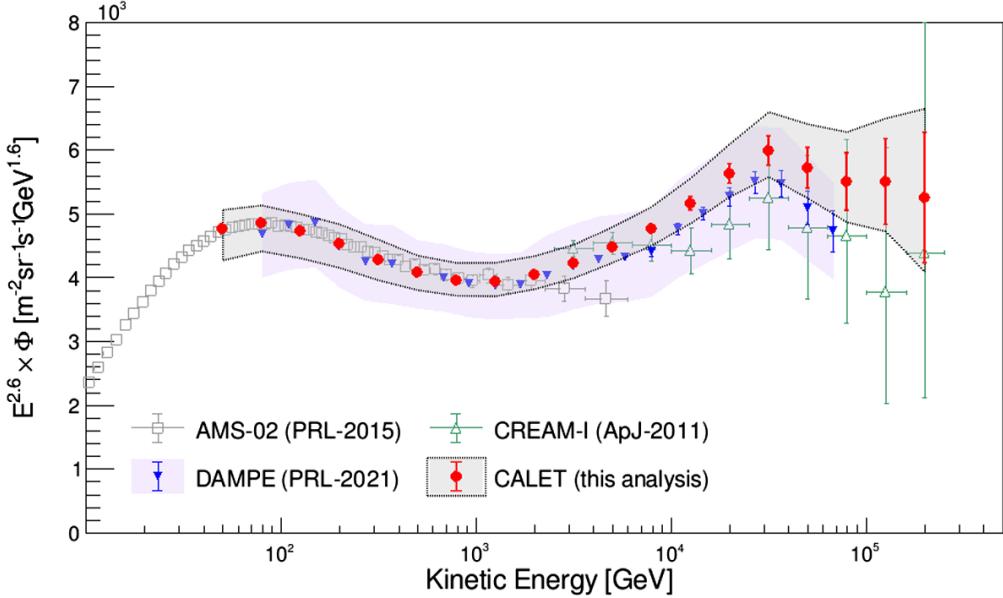


Figura 2.9: spettro di raggi cosmici di nuclei di elio misurato da CALET in rosso, confrontato con precedenti osservazioni dirette [27, 28, 29]. Le barre di errore rappresentano solo l'errore statistico; la banda grigia indica la somma quadratica degli errori sistematici e statistici, mentre la banda colorata di viola mostra le incertezze sistematiche.

I dati misurati da CALET sono stati fittati nel range da 60 GeV fino a 250 GeV tramite una funzione *double broken power law* (DBPL):

$$\Phi(E) = C \left( \frac{E}{\text{GeV}} \right)^{\gamma} \left[ 1 + \left( \frac{E}{E_0} \right)^S \right]^{\frac{\Delta\gamma}{S}} \left[ 1 + \left( \frac{E}{E_1} \right)^{S_1} \right]^{\frac{\Delta\gamma_1}{S_1}}$$

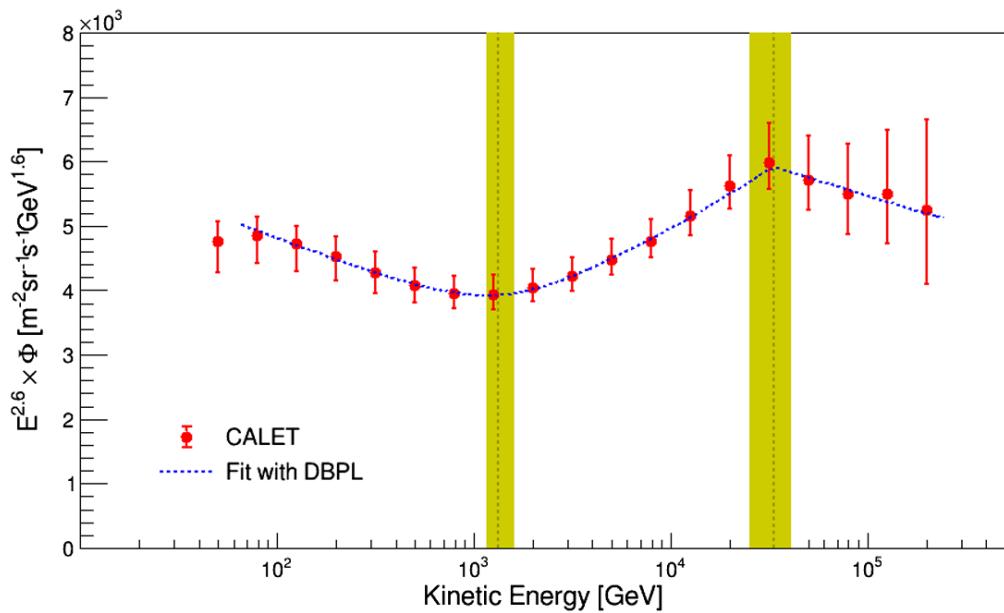


Figura 2.10: fit dei dati registrati da CALET con una funzione DBPL. I risultati sono in accordo con altre misurazioni recenti [27], entro il margine di errore. Sia gli errori sistematici che statistici sono stati presi in considerazione.

Nello spettro si notano infine due significative variazioni dell'indice spettrale che dimostrano la presenza di *hardening* ad un'energia di circa 1319 GeV e di *softening* a circa 33.2 TeV, evidenziati dalle barre in giallo nella Figura 2.10. (Per valori numerici del fit, [17]).

Nel prossimo capitolo (Capitolo 3) viene spiegata la procedura seguita per la costruzione ed il training del modello di CNN utilizzato successivamente per analizzare i dati di volo di CALET.

# Capitolo 3

## Struttura e training della rete neurale

### 3.1 Introduzione

Nel [Capitolo 1](#) sono state introdotte le reti neurali. Per una generica rete possiamo individuarne la struttura come una gerarchia che mette in relazione i seguenti elementi:

- *layers*;
- *input data* e *targets* corrispondenti;
- *loss function*;
- *optimizer*.

I *layers* sono gli ‘strati’ che formano la struttura della rete e cui vengono passati gli *input data* ed i corrispondenti *targets* con cui effettuare il training. Durante quest’ultima fase, viene utilizzata una *loss function* per avere un segnale di feedback, ovvero una misura di successo della rete: minore è il valore di questa funzione, migliore è l’accordo tra le predizioni ed i targets; si tratta quindi della quantità da minimizzare durante l’apprendimento. Infine, si ricorre ad un *optimizer* per determinare il processo di apprendimento da seguire e quindi come modificare i pesi dei layers allo scopo di minimizzare la loss function. Tale processo è schematicamente riportato in [Figura 3.1](#).

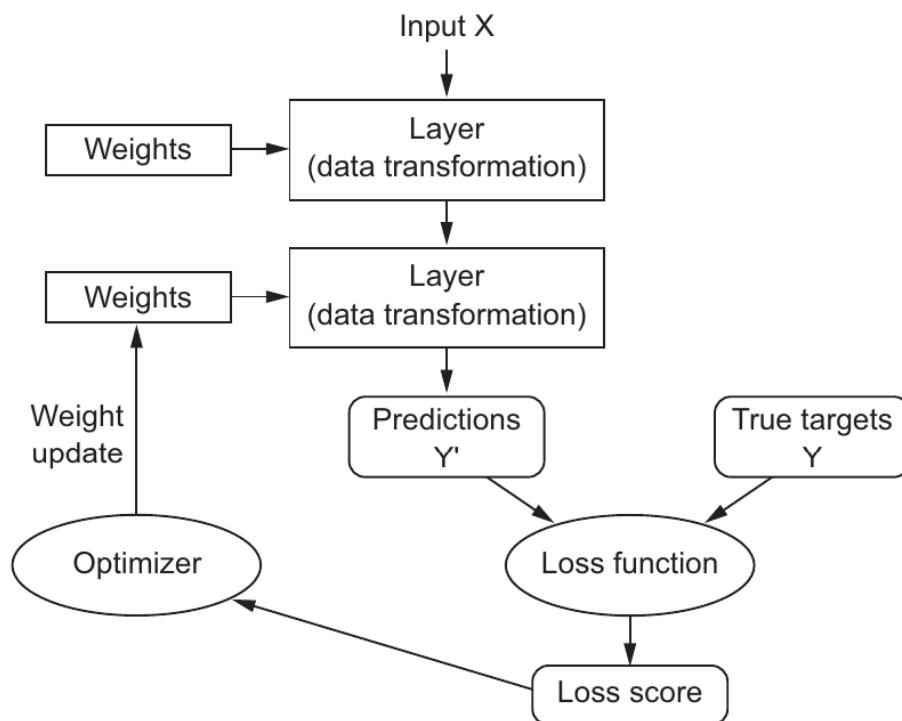


Figura 3.1: nel diagramma di flusso è possibile apprezzare in maniera schematica la relazione tra questi elementi: si ha una successione di layers, in cui, ciascuno con i propri pesi, esegue una serie di operazioni sui dati in input e li mappa con le predizioni; poi viene misurata la distanza di queste dai targets (valori veri) tramite la loss function e quindi viene effettuato l'update dei pesi tramite l'optimizer.

Il layer costituisce la struttura fondamentale di una rete neurale; i dati che riceve in input sono rappresentati tramite tensori e vengono trasformati secondo una serie di operazioni. Queste sono regolate dai pesi del layer, che svolgono la funzione di restituire in output un nuovo tensore (o più di uno), modificato in una direzione che incrementi la ‘conoscenza’ della rete e ne migliori quindi l’abilità di effettuare predizioni accurate.

## 3.2 Descrizione del problema

Il lavoro svolto in questa tesi si è concentrato sul misurare le prestazioni di una rete neurale CNN applicata al problema di ricostruzione della traiettoria dei raggi cosmici carichi nell'esperimento CALET. Di seguito vengono spiegati in maggiore dettaglio alcune caratteristiche degli eventi selezionati e la strategia adottata per eseguire lo studio.

### 3.2.1 Accettanza

All'interno del calorimetro possiamo individuare 3 sottosistemi principali: *Charge Detector* (CHD), *IMaging Calorimeter* (IMC), *Total AbSorption Calorimeter* (TASC). Ogni volta che viene registrato un evento è possibile ricostruire la traiettoria seguita dalla particella primaria utilizzando algoritmi (ad es. *Kalman filter* in [sezione A.1](#), *metodo dell'asse dello sciame* in [sezione A.2](#)) che analizzano i segnali di ionizzazione che essa rilascia negli scintillatori del CHD, nelle fibre dell'IMC e nei cristalli del TASC attraversati. Gli eventi vengono classificati in 4 categorie di accettanza geometrica (A, B, C, D) come mostrato in [Figura 3.2](#).

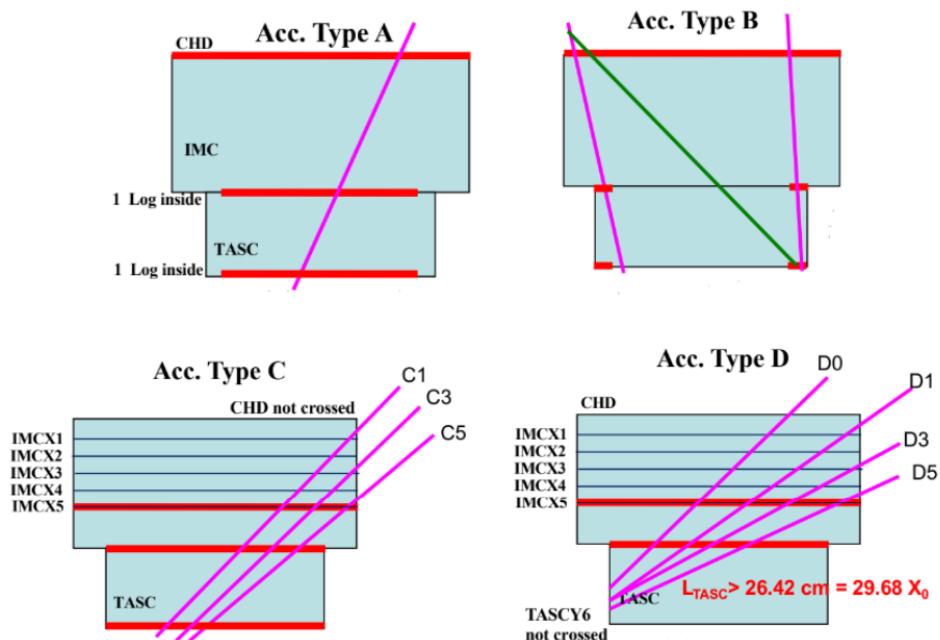


Figura 3.2: schematizzazione del calorimetro CALET nel quale è riportata in viola la direzione di particelle classificate nelle diverse categorie di accettanza.

Gli eventi di accettanza A sono quelli ‘meglio’ contenuti nel rivelatore, ovvero quelli che entrano nel piano superiore del CHD, attraversano il primo piano del TASC e ne raggiungono il fondo rimanendo entro 2 cm dai bordi laterali.

Invece, se la particella segue la stessa traiettoria, ma raggiunge il fondo del TASC ad una distanza minore di 2 cm dai bordi laterali, si parla di eventi con accettanza B.

Hanno poi accettanza C quegli eventi la cui traiettoria non attraversa il CHD ed entrano nel calorimetro da un lato dell’IMC per poi proseguire ben contenuti nel TASC.

Infine gli eventi di accettanza D sono quelli che sviluppano nel TASC uno sciamone di lunghezza superiore ai 27 cm (circa 30 lunghezze di radiazione), ma non ne raggiungono l’ultimo piano uscendo da uno dei bordi laterali.

In questo studio sono stati processati ed analizzati eventi di accettanza A.  
(`tree.fAccType[0] == 1` nel file `.root`)

### 3.2.2 Strategia e procedimento

Si illustra di seguito in maniera schematica il procedimento seguito per arrivare allo scopo ultimo di questo lavoro, ovvero ricostruire la direzione dei raggi cosmici con una rete neurale applicata ai dati di CALET e valutare la risoluzione spaziale e angolare del metodo:

- simulazione eventi con il software Monte Carlo (MC) di CALET;
- scrittura del codice [Python](#) della CNN;
- training della CNN sui dati della simulazione MC;
- analisi della risoluzione ottenuta sui dati MC simulati e confronto con il metodo dell’asse dello sciamone ([Capitolo 4](#));
- ricostruzione della traiettoria dei dati di volo tramite CNN;
- analisi della risoluzione ottenuta con la CNN sui dati di volo e confronto con quella del Kalman filter ([Capitolo 5](#)).

Le variabili oggetto dello studio sono quindi le coppie di coordinate  $(x_0, y_0)$ ,  $(x_1, y_1)$  dei punti di impatto rispettivamente sul primo piano del CHD e sul fondo del TASC e le *variabili angolari*  $(\theta_x, \theta_y)$ . Le variabili angolari indicano l’angolo formato dalla direzione della traiettoria della particella ([Figura 3.2](#)) con i piani del calorimetro, rispettivamente nelle viste  $x$  ed  $y$ .

Il valore di tali variabili può essere ricostruito a partire dai punti di impatto nel seguente modo:

$$\theta_x = \arctan \left[ \frac{x_1 - x_0}{z_1 - z_0} \right] \quad \theta_y = \arctan \left[ \frac{y_1 - y_0}{z_1 - z_0} \right]$$

Queste variabili permettono di ricostruire la traiettoria di attraversamento del calorimetro.

Si noti che nello studio della risoluzione della CNN sui dati simulati è possibile fare un confronto tra i valori predetti e quelli veri (in quanto la verità è fornita dalla simulazione); mentre nello studio della risoluzione della CNN sui dati di volo questo non è possibile (anche perché, se così non fosse, significherebbe avere già la conoscenza del punto di impatto e non sarebbe quindi necessario ricostruirlo). In questo caso si è dunque proceduto a misurare la risoluzione della CNN confrontando le predizioni con le ricostruzioni dei punti di impatto effettuate tramite Kalman filter. Si è poi misurata la risoluzione del KF sui dati MC ed, infine, si è stimata la risoluzione effettiva della CNN sui FD sottraendo in quadratura quella del KF.

### 3.2.3 Dati simulati MC

Le simulazioni MC degli eventi di attraversamento di CALET sono effettuate tramite il software Monte Carlo EPICS [23] che permette di riprodurre in ambiente virtuale la geometria del calorimetro, i processi di interazione radiazione-materia ed i segnali che vengono processati nei vari rivelatori.

Le interazioni degli sciami adronici nel calorimetro sono modellizzate tramite il software DPMJET-III [30].

Nel programma MC le particelle sono generate isotropicamente su una sfera di raggio 1 m con centro posto nel mezzo del piano superiore del CHD. Quegli eventi uscenti dalla sfera che abbiano traiettoria passante attraverso il calorimetro sono propagati ed i depositi di energia nei rivelatori sono scritti in un TTree all'interno di un file .root [31].

L'energia cinetica della particella primaria è scelta randomicamente da una distribuzione con legge di potenza  $E^{-1}$  e successivamente i dati sono ripesati per simulare gli spettri osservati che hanno leggi di potenza con indice spettrale compreso tra  $(-2.7, -2.6)$ .

Infine, i dati simulati vengono processati da CREC, il programma ufficiale della collaborazione italiana di CALET, che produce il file L2.root all'interno che quale vi è un TTree contenente le informazioni ricostruite per ciascun evento.

Per questo lavoro di tesi sono state utilizzate le seguenti variabili dei dati simulati MC:

- fTASCElayer [12], l'energia depositata in ciascun piano del TASC;

- `fAccType`, il tipo di accettanza calcolato utilizzando i parametri di generazione dell'evento;
- `fIP` ( $x_0, y_0, x_1, y_1$ ), i valori veri dei punti di impatto simulati;
- `fTASCparx` e `fTASCpary`, i parametri della direzione della particella ricostruiti col metodo dell'asse dello sciame;
- `fTASCimg`, immagini degli sciami adronici nel TASC, come quelle in [Figura 3.3](#), i cui pixel sono colorati con una scala di grigio a 8 bit e proporzionali all'energia depositata (dati in input alla CNN).

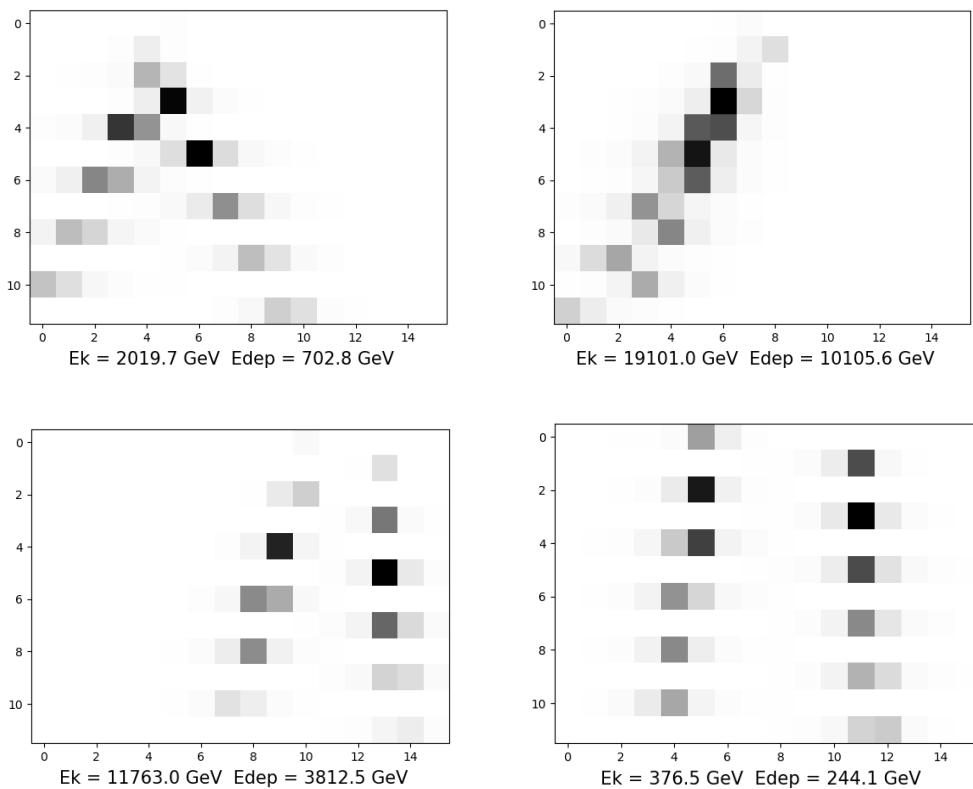


Figura 3.3: immagini di 4 eventi nel TASC, visualizzati in entrambe le viste. Partendo dall'alto, le immagini di numero dispari rappresentano la vista  $xz$ , quelle pari la vista  $yz$ . Per ogni evento è riportata l'energia cinetica della particella incidente e l'energia depositata nel TASC.

### 3.3 Coding e sviluppo della rete

L’ambiente utilizzato per lo sviluppo e l’allenamento della rete è stato il deep-learning framework per Python, [Keras](#) [32].

Si è scelto di utilizzare Keras perché questo permette di eseguire il codice sia su CPU che su GPU, inoltre fornisce una user friendly API che semplifica le operazioni di progettazione di un modello di rete ed ha anche un supporto built-in per le reti convoluzionali.

Keras è distribuito sotto licenza MIT, ciò significa che può essere liberamente utilizzato in progetti commerciali e di ricerca; è compatibile con ogni versione di Python a partire dalla 2.7.

#### 3.3.1 Modello di rete con Keras

Il framework Keras è stato utilizzato per definire la struttura della rete neurale ed effettuare il training.

Di seguito sono riportate le operazioni di coding in Python effettuate per produrre il codice con cui allenare il modello:

1. definizione dei dati di training: modificati in modo da avere *input* e *target* sotto forma di tensori;
2. definizione della struttura della rete;
3. impostazione del processo di apprendimento, scelta di una loss function ed un optimizer per la misura e correzione dell’errore sulle predizioni;
4. iterazione della procedura sui dati di training tramite il metodo Keras `fit()`.

#### 3.3.2 Workstation

Il modello di rete neurale è stato allenato sfruttando la potenza di calcolo fornita da una workstation con

CPU: INTEL CORE I7 da 3.60 GHz

GPU: [Asus NVIDIA GeForce RTX3050 8 GB](#).

Infatti è altamente raccomandato, sebbene non strettamente necessario, eseguire il codice su una scheda video NVIDIA: alcune applicazioni del deep learning, in particolare l’image processing tramite reti convoluzionali, risultano particolarmente lente se eseguite su CPU. La velocità di esecuzione su GPU è di un fattore 5 o 10 volte superiore a quella su CPU.

### 3.3.3 Struttura della rete

La rete neurale utilizzata in questo studio è stata allenata con le immagini delle viste  $xz$  e  $yz$  degli sciami adronici nel TASC di dimensioni  $12 \times 16$  pixels, dove questi sono colorati secondo una scala di grigio a 8 bit e proporzionali all'energia depositata ( $fTASCimg$ ).

In output la rete restituisce dei tensori di dimensione  $(4 \times nev)$ , ovvero si ha una ‘riga’ di quattro variabili indicanti le coordinate  $(x_0, y_0), (x_1, y_1)$  dei punti di impatto ricostruiti dalla rete, rispettivamente sul primo piano del CHD e l’ultimo del TASC. Il solitamente tensore 4D che contiene le immagini degli eventi èvengono processatoi tramite 2D convolutional layers (Conv2D). I punti di impatto ricostruiti sono poi stati utilizzati per valutare la risoluzione spaziale ed angolare della rete, confrontandoli con i valori ‘veri’ del MC, ovvero con i *targets* di riferimento. Di seguito si riporta la struttura della rete utilizzata in questo studio.

```

1 inputs=Input(shape=(12, 16, 1))
2
3 x=layers.Conv2D(256, (3, 3), activation='relu')(inputs)
4 x=layers.Conv2D(64, (3, 3), activation='relu')(x)
5 x=layers.Conv2D(32, (3, 3), activation='relu')(x)
6 x=layers.Flatten()(x)
7 x=layers.Dense(100, activation='relu')(x)
8 x=layers.Dense(50, activation='relu')(x)
9
10 outputs = layers.Dense(4)(x)

```

La struttura è da leggersi dall’alto verso il basso. Si hanno quindi tre convolutional layers Conv2D che restituiscono output di dimensioni 256, 64, 32 rispettivamente e ricevono input con dimensione pari all’output del layer precedente, o dei dati stessi in `inputs`. Segue un layer `Flatten` che rende l’output monodimensionale e tre layer `Dense` che forniscono una diversa rappresentazione del tensore in input tramite la funzione di attivazione *Rectified Linear Unit* (`relu`).

Questa struttura è stata scelta tenendo in considerazione i risultati ottenuti in [33]. Nello ‘Studio 1: confronto fra CNN con strutture differenti’ viene ottimizzato il numero di epoche di training per evitare fenomeni di over ed underfitting e vengono confrontate le prestazioni di reti con strutture (numero di layers) differenti.

Il numero di epoche ottimale risulta essere 40, mentre la struttura migliore quella e 3 layers.

Layer (type)	Output Shape	Param #
input_1 (InputLayer)	[(None, 12, 16, 1)]	0
conv2d (Conv2D)	(None, 10, 14, 256)	2560
conv2d_1 (Conv2D)	(None, 8, 12, 64)	147520
conv2d (Conv2D)	(None, 6, 10, 32)	18464
flatten (Flatten)	(None, 1920)	0
dense (Dense)	(None, 100)	192100
dense_1 (Dense)	(None, 50)	5050
dense_2 (Dense)	(None, 4)	204

Una volta definita la struttura della rete si procede specificando quale loss function ed optimizer utilizzare, ovvero dichiarando optimizer e metrica con cui supervisionare la fase di training.

```
1 model.compile(optimizer=optimizers.RMSprop(learning_rate=1e-4),
    loss='mse', metrics=['mae'])
```

Come loss function è stata scelta la *mean squared error* (`mse`), cioè la radice del valore medio della distanza al quadrato tra i target e le predizioni della CNN.

Come metrica è stata scelta la *mean absolute error* (`mae`) che rappresenta il valore assoluto della differenza tra target e predizioni.

Come optimizer è stato scelto l'algoritmo *Root Mean Squared propagation* (`RMSprop`) con *learning rate* pari a  $10^{-4}$ .

## 3.4 Training rete

In questa sezione viene descritta la procedura seguita per effettuare il training della rete neurale, concentrandosi in particolare sul modello e sui dati analizzati in questo studio.

### 3.4.1 Preparazione dei dati

I dati delle simulazioni MC sono contenuti in file `.root`, all'interno di un `TTree`.

I file sono stati suddivisi in range di energia: per i protoni sono prodotti quattro file ( $20\text{-}200$ ,  $200\text{-}2\times 10^3$ ,  $2\times 10^3\text{-}2\times 10^4$ ,  $2\times 10^4\text{-}10^6$ ) [GeV]; mentre per i nuclei di elio ne sono prodotti tre ( $20\text{-}200$ ,  $200\text{-}2\times 10^3$ ,  $2\times 10^3\text{-}2\times 10^4$ ) [GeV].

Le variabili estratte da ciascun file sono salvate in tensori `np.array()` della libreria `NumPy` [34] di Python riportati di seguito:

- `nev`, numero di eventi selezionati nel `TTree`;

- data, input data selezionati;
- targets, targets corrispondenti agli input data;
- TASCip, impact points nel TASC;
- TASCeDep, energia depositata nel TASC;
- AccType, parametro di accettanza;
- prohe, parametro per distinguere se l'evento è relativo ad un protone o ad un nucleo di elio.

Il processo di selezione degli eventi avviene in maniera sequenziale, analizzando uno dopo l'altro ciascun file. Ad ogni file viene applicato un filtro di selezione degli eventi: solo quelli che soddisfano la seguente condizione vengono estratti e salvati nei tensori corrispondenti.

```

1 if (tree.fAccType[0]==1
2   and tree.fTASCElayer[12]<EDEPTHR
3   and tree.fTASCElayer[12]>20
4   and emaxlay[0]<0.45
5   and emaxlay[1]<0.45):

```

Dove, con `tree.fAccType[0]==1` si selezionano gli eventi di ‘accettanza A’; con `tree.fTASCElayer[12]<EDEPTHR` e `tree.fTASCElayer[12]>20` si richiede che il valore dell’energia depositata nel TASC, ovvero ( $E_{TASC}$ ), sia nel range (20,  $2 \times 10^4$ ) [GeV].

La variabile `data` è un tensore di dimensione 4 in cui vengono salvati i valori degli input selezionati tramite due cicli `for` sulle variabili `j` e `k`.

```

1 data[nev, j, k, 0]=tree.fTASCimg[j*16+k]

```

Questo permette di creare (visivamente) delle ‘pile’ di matrici di ‘spessore’ unitario, in cui ciascuna matrice coincide con una immagine (dei punti di impatto) contenuta nel file `.root` ed ogni elemento di una matrice coincide con il pixel in posizione ( $j, k$ ) dell’immagine. L’altezza di ciascuna ‘pila’ dipende dal numero di matrici impilate, ovvero dal numero di immagini salvate nel tensore e quindi dal numero di eventi selezionati in ciascun file (`nev`).

### 3.4.2 Training e salvataggio predizioni

Una volta estratti i dati di interesse dal file `.root` si prosegue con la suddivisione di questi in tre categorie: *train*, *validation*, *test*.

Nello studio in questione si è scelto di ripartirli secondo le percentuali:

```
1 train=0.8  
2 val=0.1  
3 test=0.1
```

dove la variabile `train` indica la percentuale di dati da utilizzare come dati per effettuare il training della rete; mentre la variabile `val` indica la percentuale di dati da utilizzare per effettuare la validazione del modello, ovvero per misurare l'accuratezza dell'operazione di training; infine la variabile `test` indica la percentuale di dati su cui non allenare la rete per testarne l'abilità su un set di dati 'nuovi', ovvero di cui la rete non conosce la verità.

Si prosegue dunque con la fase di training, richiamando la funzione `model.fit()` e passando in input i dati `test` e `val`:

```
1 history=model.fit(train_data, train_targets, epochs=epoche,  
2 batch_size=32, validation_data=(val_data, val_targets),  
3 callbacks=None)
```

Questa stringa lancia ed effettua il training del modello per il numero di epoche passato in input (40, nel nostro studio). Successivamente si testa il modello appena allenato sulla porzione di dati dedicata al test, per misurarne l'accuratezza anche su dati non provenienti dal dataset di training:

```
1 test_pred = model.predict(test_data)  
2 model.save(path_model_save, overwrite=True)
```

Quindi, tramite il comando `model.save()` riportato alla seconda riga, viene salvato il modello ottenuto, così che sia possibile utilizzarlo per produrre predizioni su altri dataset senza dover effettuare nuovamente il training ogni volta.

Infine vengono salvate le predizioni prodotte dalla rete ed i valori di `mae` ed `mse` misurati durante la fase di training in un file `.root` in output. Tale file viene utilizzato successivamente per effettuare l'analisi dei dati ottenuti e produrne i relativi grafici. ([Appendice B](#)).

## 3.5 Applicazione della CNN ai dati

In questo lavoro di tesi si è effettuato il training del modello su due dataset: dati MC simulati di nuclei elio e dati combinati di protoni e nuclei di elio.

I due modelli sono stati poi impiegati per compiere predizioni sui dati MC simulati di nuclei di elio e valutare quale dei due fosse il più preciso, ovvero avesse risoluzione spaziale migliore.

Di seguito, entrambi i modelli sono stati utilizzati per ricostruire i punti di impatto di un dataset di dati ‘veri’, di cui non si conosce quindi la verità: per questa ragione, al fine di stimare la risoluzione del metodo in questo caso, è stato necessario confrontare le predizioni di quest’ultimo con quelle di un altro metodo, l’algoritmo *Kalman filter combinatorio* (di cui maggiori dettagli sono in [sezione A.1](#)).

Nei prossimi capitoli, [Capitolo 4](#) e [Capitolo 5](#), vengono effettuate le analisi dei risultati ottenuti dalla CNN sui dati del MC e del FD rispettivamente.

# Capitolo 4

## Analisi dei dati delle simulazioni MC

Il modello di rete neurale precedentemente descritto è stato testato su dati delle simulazioni di CALET, in modo da misurarne le prestazioni ed, in particolare, la risoluzione spaziale ed angolare sulla direzione ricostruita delle particelle. Questo ha permesso di effettuare un confronto con i risultati ottenuti tramite il metodo di ricostruzione dell’asse dello sciame (in [sezione A.2](#)).

In particolare sono stati allenati due modelli, entrambi con la stessa struttura, ma su dataset differenti: uno su dati MC simulati di nuclei elio e l’altro su dati combinati di protoni e nuclei di elio.

### 4.1 Introduzione

Di seguito vengono riportati i risultati ottenuti dall’analisi dati effettuata sui dati prodotti in output fornendo i due dataset alla CNN descritta nella [sottosezione 3.3.3](#).

Operativamente, per analizzare tali dati è stata scritta una ‘macro’ ROOT che svolge le seguenti operazioni:

- accetta in input i file processati dal modello di CNN;
- calcola le distribuzioni dei residui delle variabili spaziali, quindi la differenza tra il valore predetto dalla rete ed il valore vero della simulazione Monte Carlo per ciascuna variabile del tensore del punto di impatto  $(x_0, y_0), (x_1, y_1)$ ;
- calcola le distribuzioni dei residui delle variabili angolari<sup>1</sup>  $(\theta_x, \theta_y)$ ;
- calcola la risoluzione spaziale;
- calcola la risoluzione angolare.

---

<sup>1</sup>Definite nella [sottosezione 3.2.2](#).

Le distribuzioni sono state analizzate in 14 intervalli di energia depositata nel TASC ( $E_{TASC}$ ). Per l'analisi dati dei risultati prodotti dalla CNN allenata su dati combinati di p e He, sono state selezionate solo le predizioni relative a nuclei di elio.

Range [GeV]	Eventi	411	↔	632	3244
20	↔ 31	1827	632	↔ 974	3286
31	↔ 47	2215	974	↔ 1500	3353
47	↔ 73	2621	1500	↔ 2310	3297
73	↔ 112	2722	2310	↔ 3557	3120
112	↔ 173	2961	3557	↔ 5477	2481
173	↔ 267	3041	5477	↔ $2 \times 10^4$	932
267	↔ 411	3128			

Il criterio adottato nella suddivisione del range di energia  $(20, 2 \times 10^4)$  [GeV] è il seguente: sono stati utilizzati intervalli di ampiezza costante in scala logaritmica, ovvero tali per cui  $\log_{10}(E_{i+1}/E_i) = \text{const}$  per tutti gli intervalli tranne che per l'ultimo. Da 5477 a  $2 \times 10^4$  [GeV] si è scelto di estendere l'intervallo fino a raccogliere tutti gli eventi rimasti.

## 4.2 Analisi dati

Gli eventi prodotti dalla simulazione MC di CALET sono stati salvati in tre file, a seconda del range di energia di appartenenza  $(20\text{-}200, 200\text{-}2 \times 10^3, 2 \times 10^3\text{-}2 \times 10^4)$  [GeV] e, come spiegato nel [Capitolo 3](#), la CNN è stata prima *allenata* e *validata* sul 90% dei dati presenti nel dataset, mentre è stata poi *testata* 10% di questi.

Per valutare l'efficienza della rete nel predire i punti di impatto (variabili spaziali) e le variabili angolari, quindi la risoluzione spaziale ed angolare, sono state calcolate le distribuzioni dei residui tra questi valori e la verità del Monte Carlo.

Come spiegato con maggiore dettaglio a [Pag. 37](#), i dati presenti nel dataset sono immagini di sciami adronici nel TASC, dai quali sono stati selezionati p ed He ad alte e basse energie. Alcuni esempi di tali immagini sono riportati nella [Figura 3.3](#).

Per poter valutare concretamente ‘quanto bene’ la rete performi è necessario conoscere il valore ‘vero’ e misurare quanto questo e quello predetto siano lontani. Non sarebbe stato infatti possibile impiegare direttamente la CNN per effettuare previsioni su dati di volo (FD) CALET poiché di questi non si ha a disposizione il valore ‘vero’, che è proprio ciò che vogliamo stimare, e quindi non sarebbe stato possibile avere una stima dell’errore e della risoluzione della rete nell’effettuare tale previsione.

Dunque, è stato prima di tutto necessario valutare la risoluzione della CNN su dati simulati e solo successivamente applicarla ai dati di volo.

Di seguito vengono riportati alcuni grafici delle distribuzioni dei residui e della risoluzione sia della CNN allenata su nuclei di elio che della CNN allenata su dati combinati di p e He. Per ciascun grafico vengono messe in evidenza le caratteristiche di maggiore importanza.

(In questo capitolo sono stati inseriti solo i grafici relativi alla variabile  $x_0$  spaziale ed  $\theta_x$  angolare. Per gli altri consultare l'[Appendice B](#)).

#### 4.2.1 Training, *mae* ed *mse*

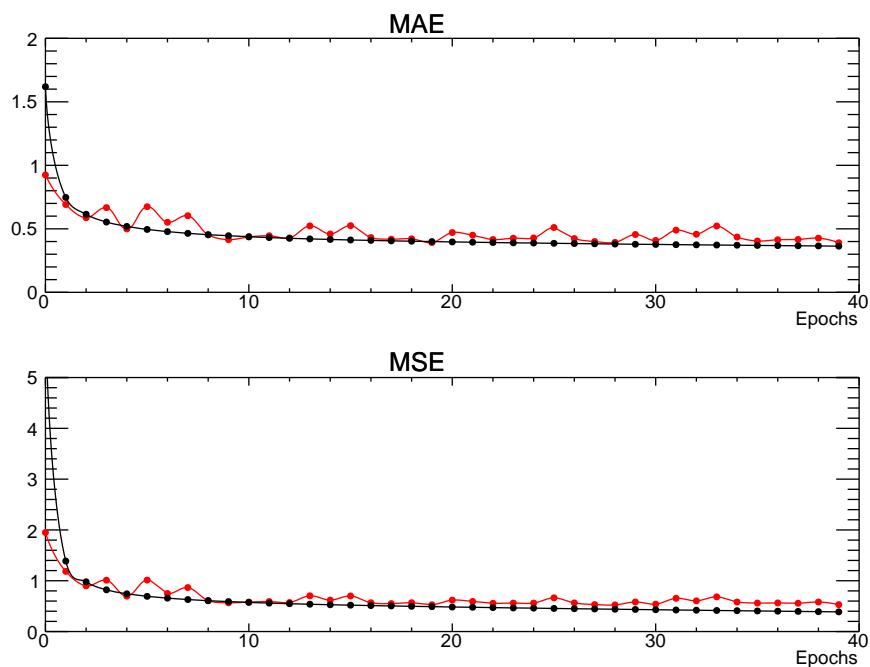


Figura 4.1: grafici del *mean absolute error* (*mae*) e *mean square error* (*mse*) relativi alla CNN allenata sui nuclei di elio. In nero la curva del training, in rosso quella della validazione.

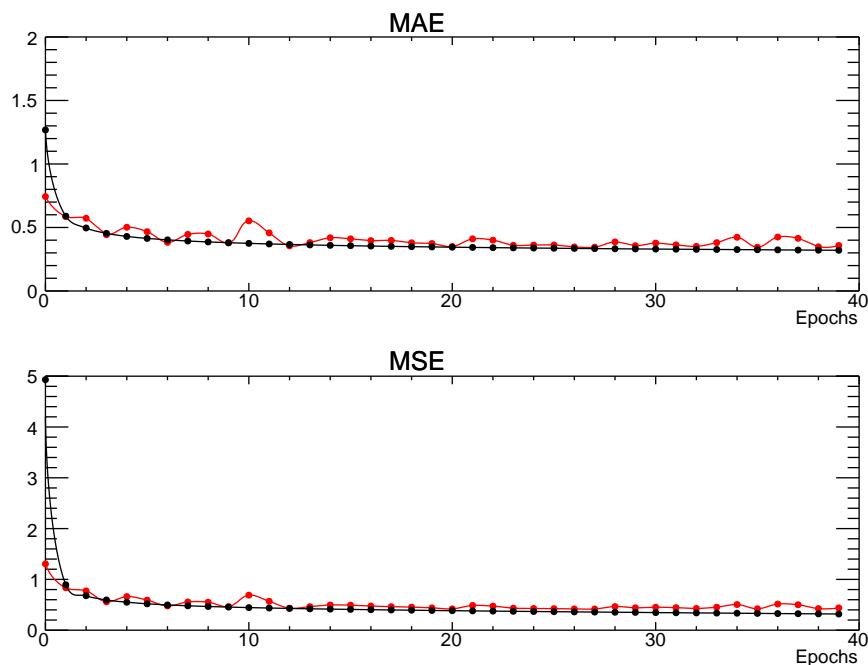


Figura 4.2: grafici del *mean absolute error* (mae) e *mean square error* (mse) relativi alla CNN allenata sul dataset combinato di p e He. In *nero* la curva del training, in *rosso* quella della validazione.

Si osserva che effettuare il training per 40 epoche stabilizza sia il mae che il mse su valori prossimi allo 0.5 ed evita problemi di overfitting.

### 4.2.2 Distribuzioni dei residui

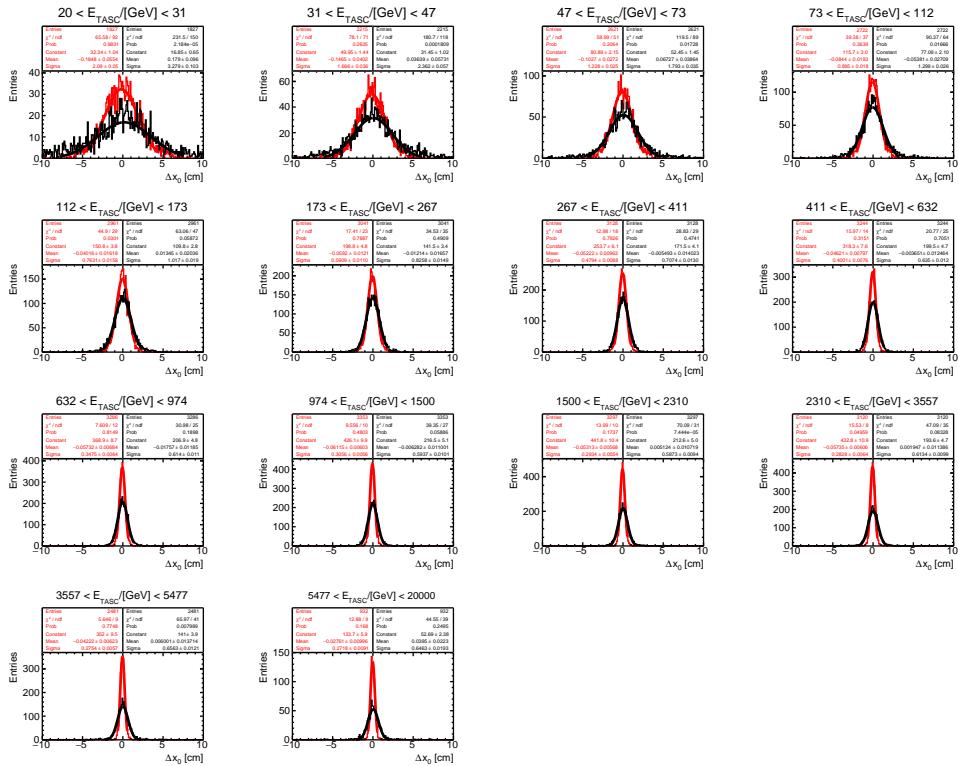


Figura 4.3: distribuzione dei residui della variabile di impact point  $x_0$  predetta dalla CNN allenata sui nuclei di elio (istogramma in rosso) e dal metodo dell'asse dello sciamo (istogramma in nero). Le curve sovrapposte agli istogrammi costituiscono i fit gaussiani agli istogrammi del medesimo colore.

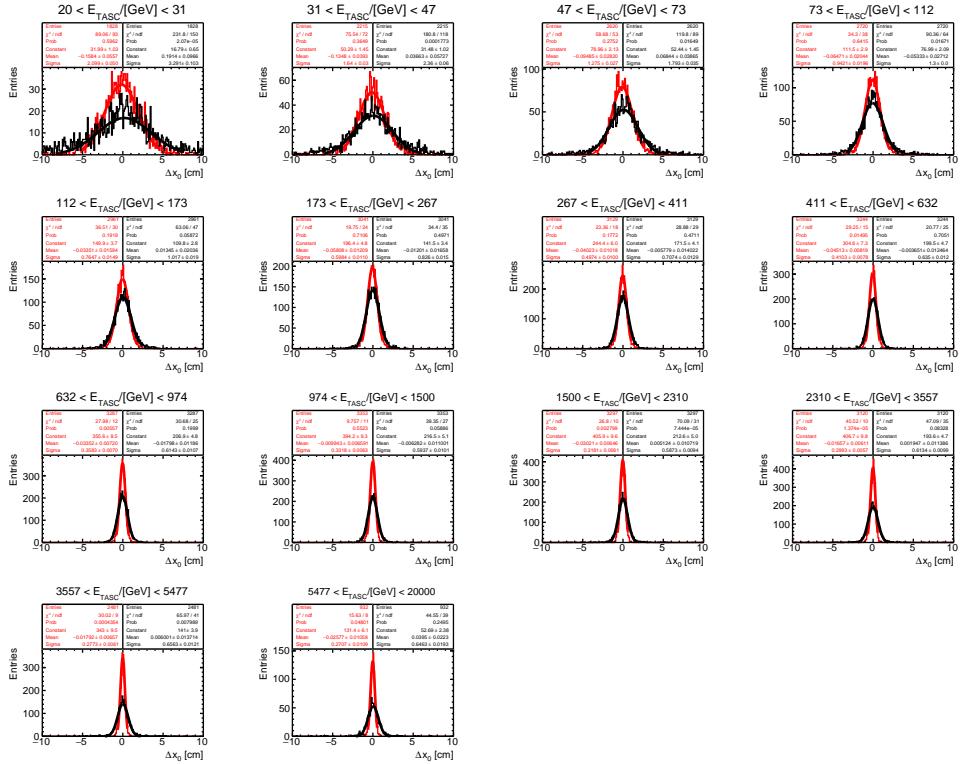


Figura 4.4: distribuzione dei residui della variabile di impact point  $x_0$  predetta dalla CNN allenata con il dataset combinato di p e He (istogramma in rosso) e dal metodo dell'asse dello sciame (istogramma in nero). Le curve sovrapposte agli istogrammi costituiscono i fit gaussiani agli istogrammi del medesimo colore.

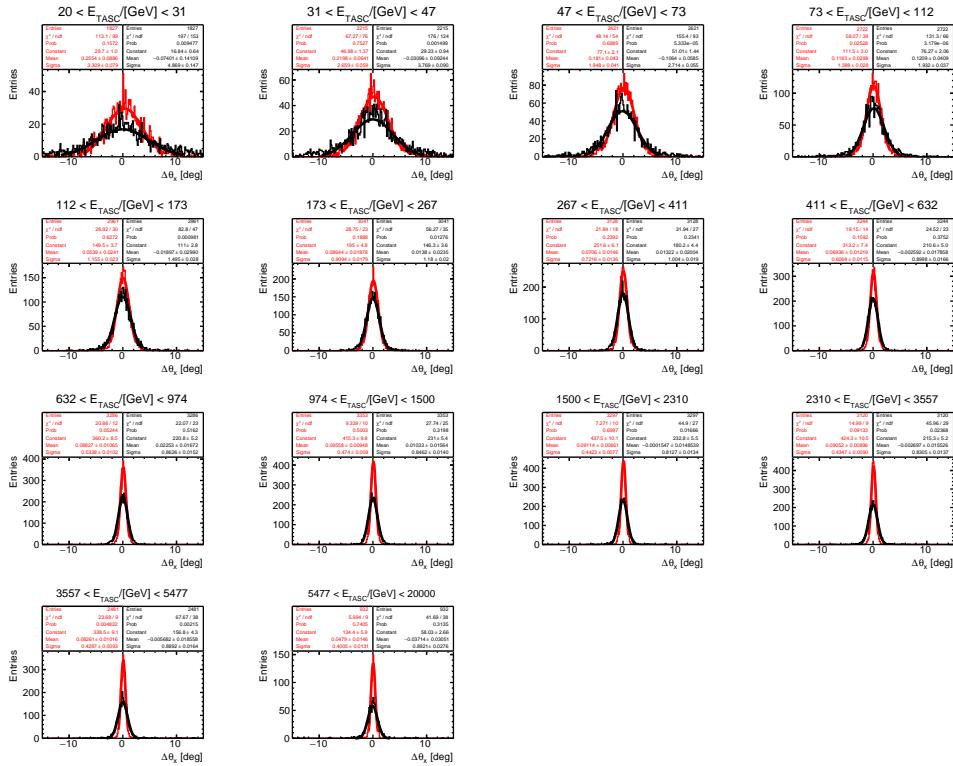


Figura 4.5: distribuzione dei residui della variabile angolare  $\theta_x$  predetta dalla CNN allenata sui nuclei di elio (istogramma in rosso) e dal metodo dell’asse dello sciame (istogramma in nero). Le curve sovrapposte agli istogrammi costituiscono i fit gaussiani agli istogrammi del medesimo colore.

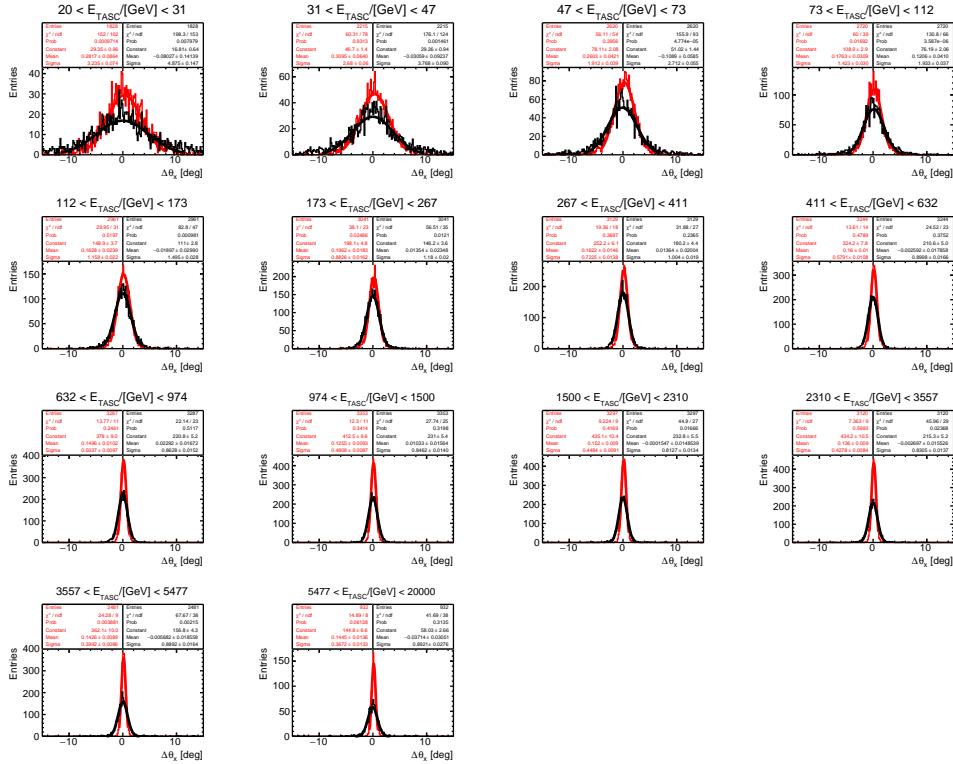


Figura 4.6: distribuzione dei residui della variabile angolare  $\theta_x$  predetta dalla CNN allenata con il dataset combinato di p e He (istogramma in *rosso*) e dal metodo dell’asse dello sciame (istogramma in *nero*). Le curve sovrapposte agli istogrammi costituiscono i fit gaussiani agli istogrammi del medesimo colore.

### 4.2.3 Media dei residui e risoluzione spaziale

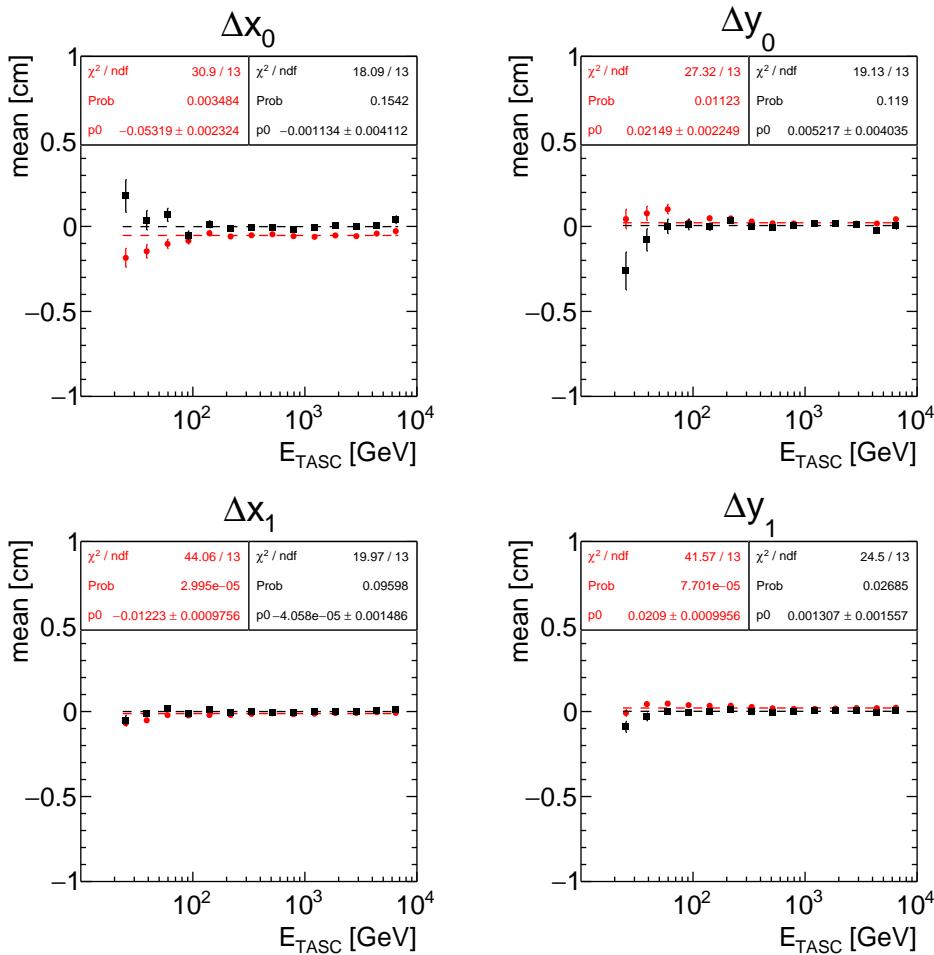


Figura 4.7: valore medio della distribuzione dei residui dei punti di impatto della CNN allenata sui nuclei di elio (in rosso) e del metodo dell'asse dello sciame (in nero), in funzione dell'energia media depositata in ciascun range.

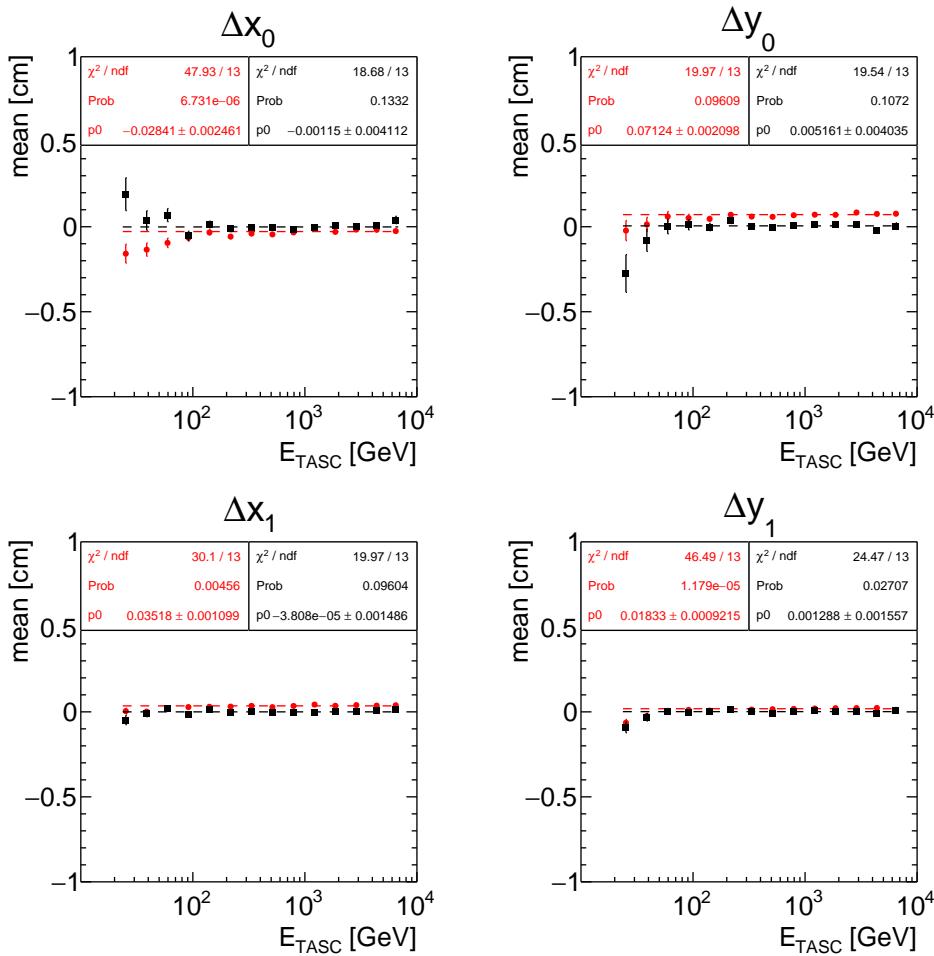


Figura 4.8: valore medio della distribuzione dei residui dei punti di impatto della CNN allenata con il dataset combinato di p e He (in rosso) e del metodo dell'asse dello sciame (in nero), in funzione dell'energia media depositata in ciascun range.

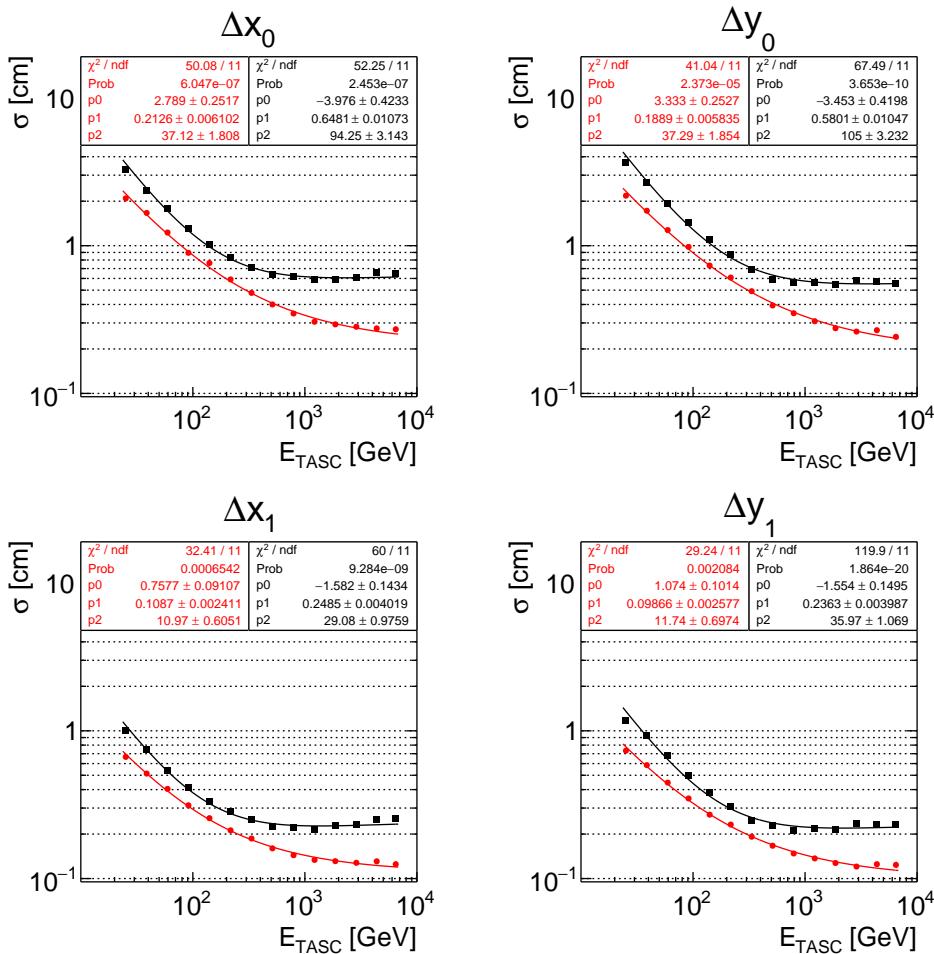


Figura 4.9: risoluzione spaziale della distribuzione dei residui dei punti di impatto della CNN allenata sui nuclei di elio (in rosso) e del metodo dell’asse dello sciame (in nero), in funzione dell’energia media depositata in ciascun range.

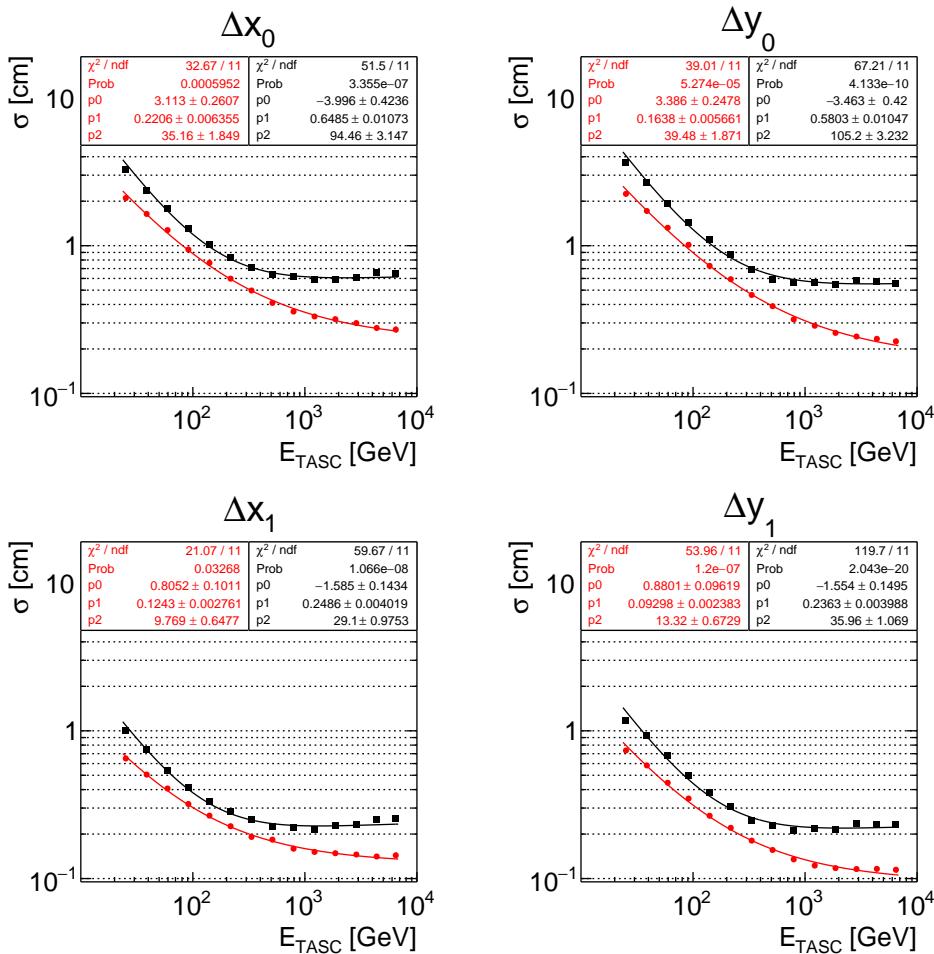


Figura 4.10: risoluzione spaziale della distribuzione dei residui dei punti di impatto della CNN allenata con il dataset combinato di p e He (in rosso) e del metodo dell'asse dello sciame (in nero), in funzione dell'energia media depositata in ciascun range.

#### 4.2.4 Media dei residui e risoluzione angolare

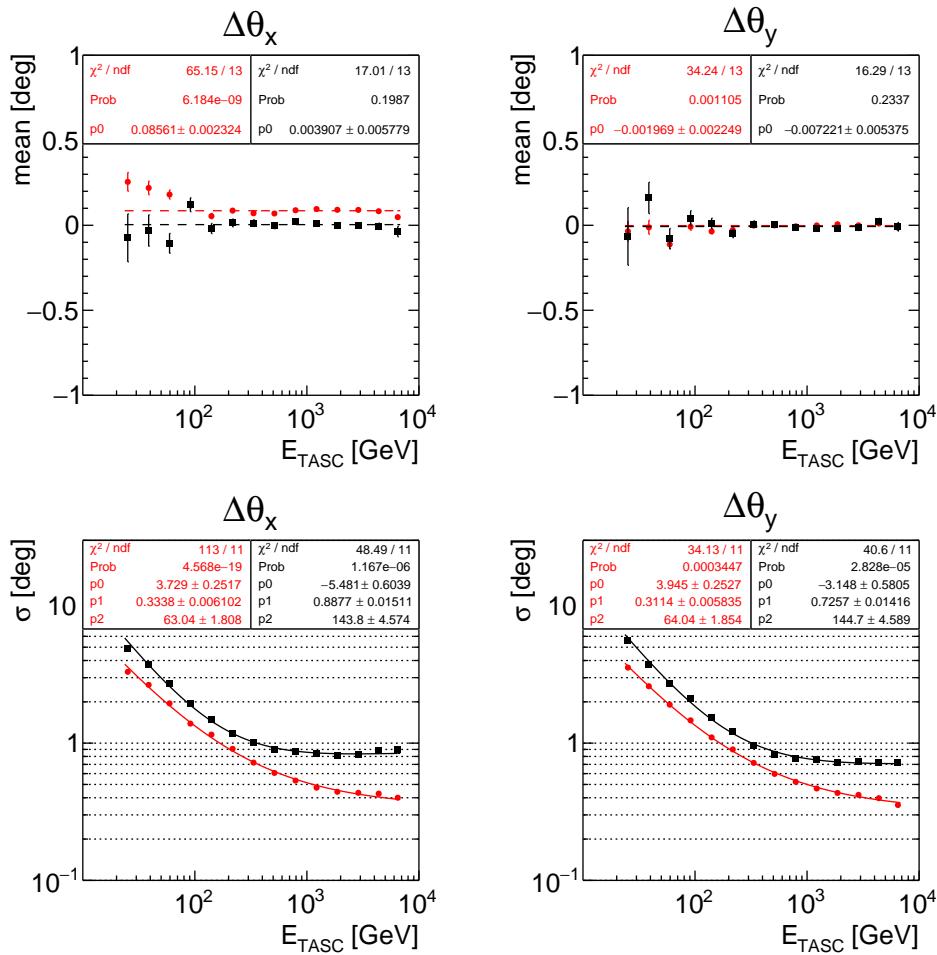


Figura 4.11: nei grafici *sopra* il valore medio della distribuzione dei residui; *sotto* la risoluzione angolare della distribuzione dei residui delle variabili angolari della CNN allenata sui nuclei di elio (in *rosso*) e del metodo dell'asse dello sciame (in *nero*), in funzione dell'energia media depositata in ciascun range.

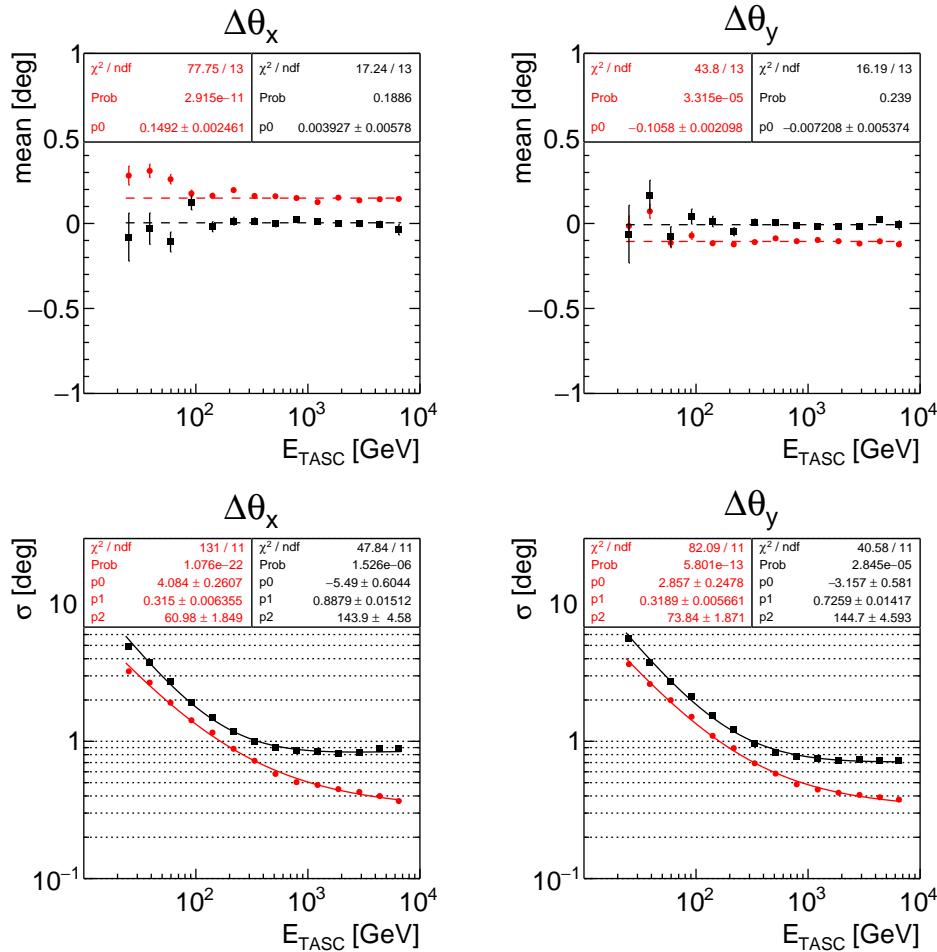


Figura 4.12: nei grafici *sopra* il valore medio della distribuzione dei residui; *sotto* la risoluzione angolare della distribuzione dei residui delle variabili angolari della CNN allenata con il dataset combinato di p e He (in rosso) e del metodo dell'asse dello sciame (in nero), in funzione dell'energia media depositata in ciascun range.

### 4.3 Confronto quantitativo fra CNN e metodo dell’asse dello sciame

Nei grafici [Figura 4.3](#) e [Figura 4.4](#) sono riportate le distribuzioni dei residui relative alle predizioni sulla variabile di impatto  $x_0$  compiute dalla rete allenata su nuclei di elio e da quella allenata su campione di p e He.

Ci si aspetta che l’errore compiuto dalla rete nell’effettuare la previsione sia *casuale*, ovvero è una *variabile casuale*. Dalla teoria statistica sui *teoremi del limite centrale* [35] sappiamo che la somma (normalizzata) di un grande numero di variabili casuali è distribuita approssimativamente come una variabile casuale normale, ovvero segue distribuzione *gaussiana*.

Per questo motivo i grafici dei residui sono stati fittati con curve gaussiane: la media  $\mu$  rappresenta da distanza delle previsioni dal valore vero della simulazione MC e, quindi, tanto più è vicina a zero, migliore è la predizione; la  $\sigma$  indica la “risoluzione” del metodo, ovvero quanto si disperde l’errore attorno alla media e, quindi la larghezza della ‘campana’ sotto cui ricade una certa percentuale dei residui, ad esempio, il 68.3% entro una sigma ( $\mu - 1\sigma, \mu + 1\sigma$ ). Avere una sigma minore, ovvero una ‘campana’ stretta e alta, significa avere una migliore risoluzione.

Detto ciò risulta chiaro come dal fit delle distribuzioni dei residui dei punti di impatto ([Figura 4.3](#), [Figura 4.4](#)) e delle variabili angolari ([Figura 4.5](#), [Figura 4.6](#)) sia possibile calcolare il valor medio della distribuzione dei residui ([Figura 4.7](#), [Figura 4.8](#), [Figura 4.11](#)), e la risoluzione spaziale ([Figura 4.9](#), [Figura 4.10](#)) e angolare ([Figura 4.12](#)) e quindi la distribuzione dei residui e la risoluzione angolare ([Figura 4.11](#), [Figura 4.12](#)) dei due modelli di rete neurale in funzione dell’energia depositata nel TASC e confrontarla con i risultati prodotti dal metodo dell’asse dello sciame. Si è proceduto infatti ad estrarre da ciascun range di energia il valor medio dell’energia depositata nel TASC (asse  $x$  nei grafici) e poi a rappresentare nei grafici (asse  $y$ ) il valore medio e la risoluzione, estratti dal fit gaussiano, in funzione dell’energia media di ciascun range.

Dall’analisi dei grafici, possiamo dunque compiere le seguenti osservazioni:

Non si apprezzano differenze tra i risultati ottenuti con il dataset di soli nuclei di elio rispetto a quello con il dataset combinato di p e He, in nessuno dei parametri considerati. Dunque si conclude che l’impiego del dataset combinato di p e He non introduca né miglioramenti, né peggioramenti nella risoluzione del modello rispetto a quello allenato su dataset di soli nuclei di elio.

Dunque, poiché non vi sono differenze apprezzabili tra i due dataset, nel seguito è stato considerato soltanto quello con soli nuclei di He.

Differente è invece la situazione per quanto riguarda il confronto tra CNN e metodo dell'asse dello sciame: innanzitutto si osserva che in ogni range di energia la risoluzione spaziale ed angolare della CNN è migliore di quella del metodo dell'asse dello sciame. Inoltre si osserva anche che la risoluzione aumenta con l'aumentare dell'energia.

Infatti, se a basse energie (20, 31) GeV, la coordinata  $x_0$  della CNN ha risoluzione  $\sigma_{x_0} = (2.09 \pm 0.05)$  cm e la coordinata  $\theta_x$  ha risoluzione  $\sigma_{\theta_x} = (3.31 \pm 0.08)^\circ$ , si nota che, all'aumentare dell'energia, il valore di  $\sigma_{x_0}$  diminuisce progressivamente, ovvero la larghezza della campana gaussiana si riduce, fino ad arrivare ad alte energie (5477,  $2 \times 10^4$ ) GeV dove  $\sigma_{x_0} = (0.27 \pm 0.01)$  cm e la coordinata  $\theta_x$  ha risoluzione  $\sigma_{\theta_x} = (0.40 \pm 0.01)^\circ$ . Dunque, all'aumentare dell'energia si ha un incremento nella risoluzione di un fattore  $\approx 10$  sulla risoluzione spaziale e di un fattore  $\approx 8$  sulla risoluzione angolare.

Si ha che invece il metodo dell'asse dello sciame ha una risoluzione alle basse energie pari a  $\sigma_{x_0} = (3.28 \pm 0.11)$  cm e  $\sigma_{\theta_x} = (4.87 \pm 0.15)^\circ$ , mentre alle alte pari a  $\sigma_{x_0} = (0.65 \pm 0.02)$  cm e  $\sigma_{\theta_x} = (0.89 \pm 0.03)^\circ$ .

Dunque la CNN ha una risoluzione migliore rispetto al metodo dell'asse dello sciame di  $\Delta\sigma_{x_0} = (1.19 \pm 0.16)$  cm e  $\Delta\sigma_{\theta_x} = (1.56 \pm 0.23)^\circ$  alle basse energie; mentre di  $\Delta\sigma_{x_0} = (0.38 \pm 0.03)$  cm e  $\Delta\sigma_{\theta_x} = (0.49 \pm 0.04)^\circ$  alle alte energie.

Si è osservato che, in alcuni casi, si ha un errore sistematico nella ricostruzione dei punti di impatto che porta il valore medio dei residui ad essere ‘spostato’ in positivo o in negativo rispetto allo zero. Questo errore è probabilmente dovuto ad una scelta non ottimale dei pesi durante il training da parte della CNN e, per essere risolto, si potrebbe provare ad effettuare nuovamente il training su un dataset con un numero più elevato di eventi.

Per effettuare il fit del valore medio delle distribuzioni dei residui è stata utilizzata una funzione costante, mentre per il fit della risoluzione spaziale ed angolare la funzione usata è:

$$y = \frac{a}{\sqrt{x} + b + \frac{c}{x}}$$

con  $a, b, c$  parametri del fit. "[0] /TMath::Sqrt(x)+[1]+[2]/x" nella macro di ROOT.

## 4.4 Analisi dati con Kalman filter

Oltre alla CNN ed al metodo dell'asse dello sciame, il dataset di nuclei di elio prodotto dalla simulazione MC di CALET è stato passato in input anche all'algoritmo Kalman filter combinatorio, riportato in appendice ([sezione A.1](#)). Questo algoritmo, impiegato in CALET principalmente per effettuare la distinzione tra sciami elettromagnetici ed adronici, permette anche di stimare la direzione ed il punto di impatto dello sciame

primario, costituendo così un metodo affidabile di analisi con cui confrontare le prestazioni della CNN.

L'algoritmo Kalman filter combinatorio di CALET è implementato nel programma di simulazione MC. Dunque, operativamente, non è stato necessario implementarlo nuovamente o richiamarlo successivamente sul dataset ottenuto in output dalla simulazione: già durante la simulazione stessa il programma di CALET effettua la ricostruzione dei dati simulati tramite KF. Quindi i valori ricostruiti tramite questo algoritmo vengono forniti direttamente in output dalla simulazione MC.

La scelta effettuata per la divisione del range di energia verrà giustificata nel [Capitolo 5](#), dove le previsioni del KF saranno impiegate per valutare la risoluzione della CNN sui dati di volo.

Di seguito vengono riportati alcuni grafici delle distribuzioni dei residui tra il valore predetto dal KF ed il valore vero MC e della risoluzione spaziale ed angolare di tale metodo di ricostruzione. In questo capitolo sono stati inseriti solo i grafici relativi alla variabile  $x_0$  spaziale ed  $\theta_x$  angolare. Per gli altri consultare l'[Appendice B](#).

#### 4.4.1 Distribuzioni dei residui

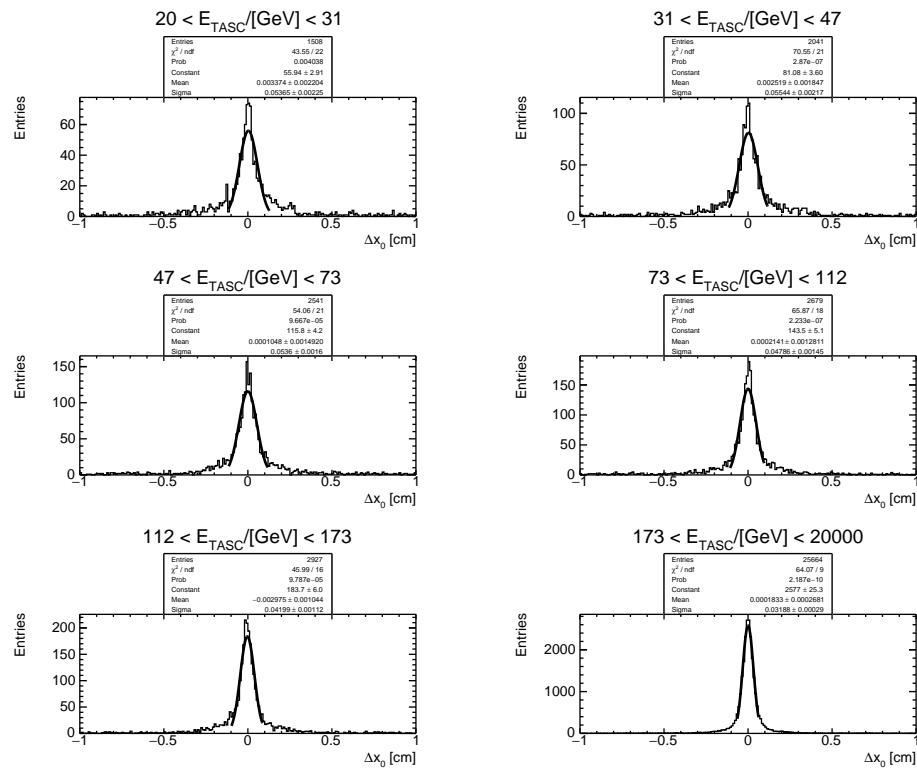


Figura 4.13: distribuzione dei residui della variabile di impact point  $x_0$  predetta dal Kalman filter (istogramma in nero) sui dati simulati MC di elio. La curva sovrapposta agli istogrammi costituisce il fit gaussiano agli istogrammi.

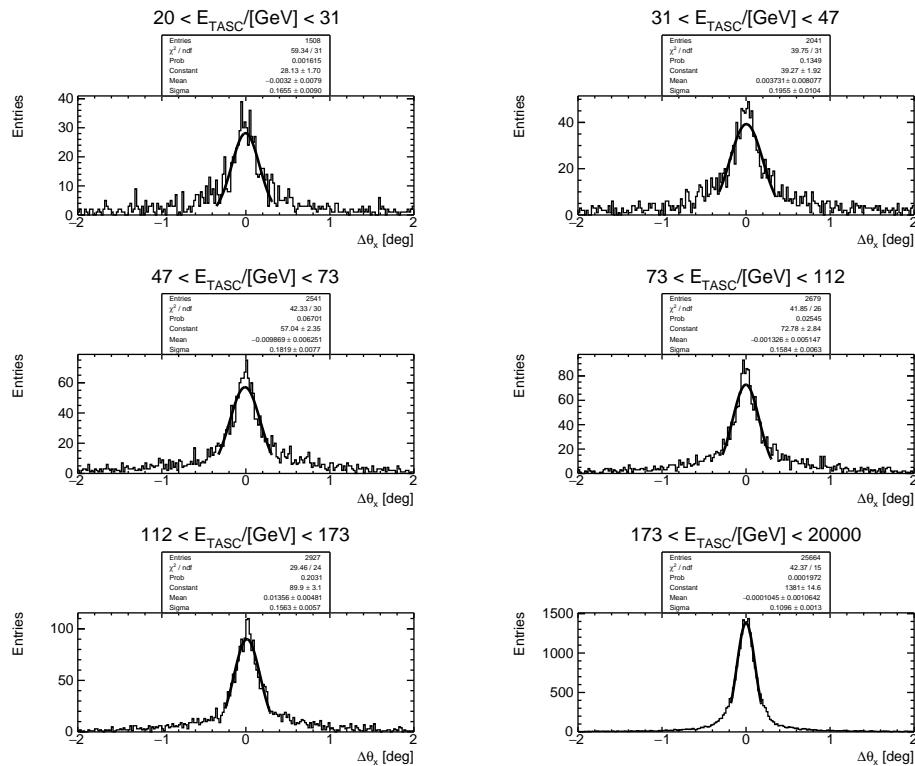


Figura 4.14: distribuzione dei residui della variabile angolare  $\theta_x$  predetta dal Kalman filter (istogramma in nero) sui dati simulati MC di elio. La curva sovrapposta agli istogrammi costituisce il fit gaussiano agli istogrammi.

#### 4.4.2 Media dei residui e risoluzione spaziale

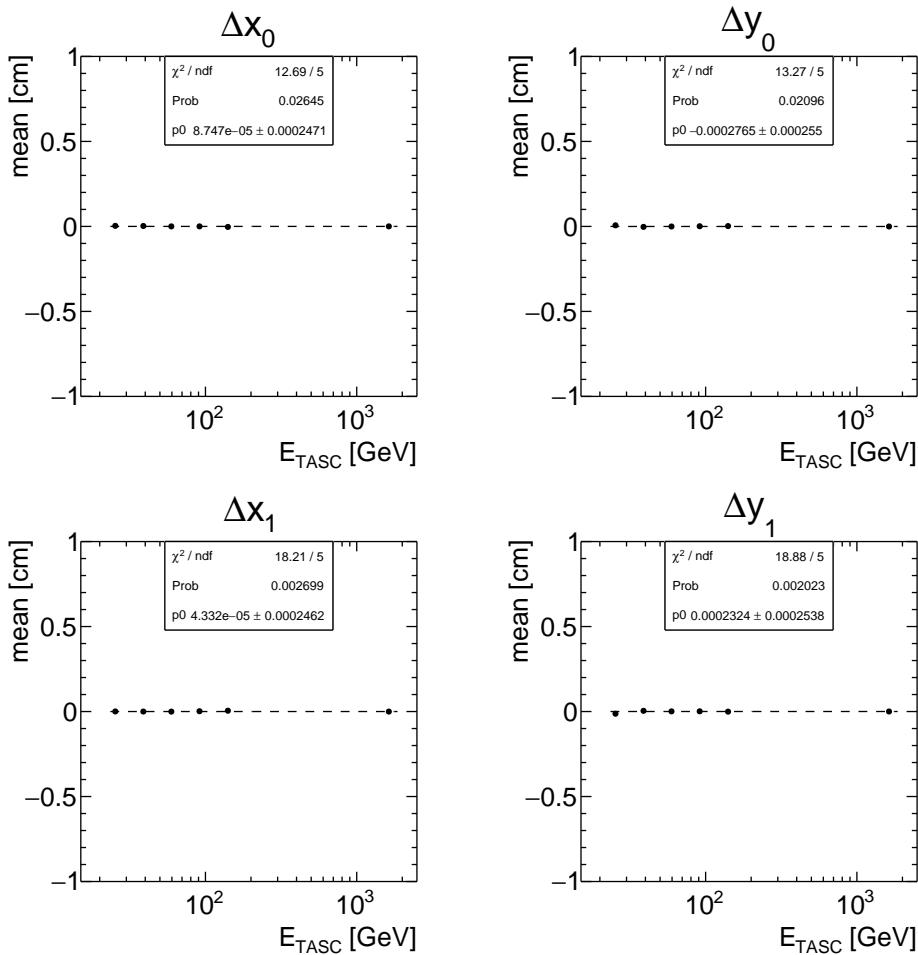


Figura 4.15: valore medio della distribuzione dei residui dei punti di impatto predetti dal Kalman filter (in nero) sui dati simulati MC di elio, in funzione dell'energia media depositata in ciascun range.

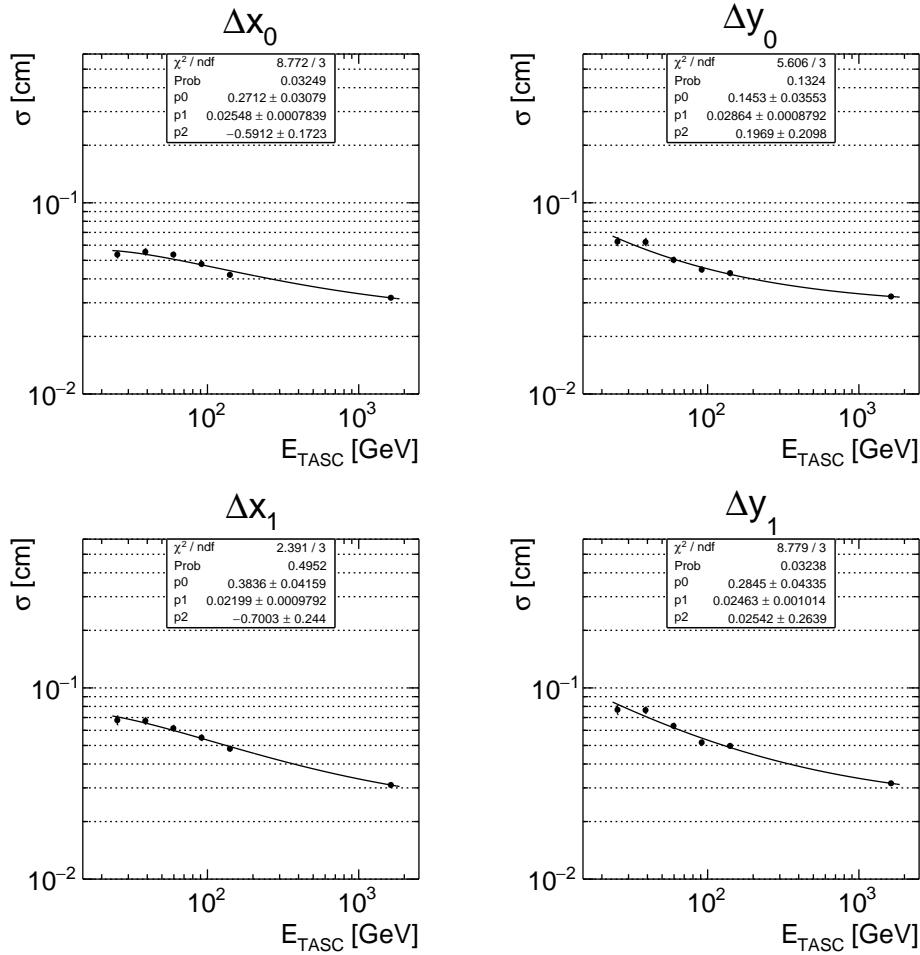


Figura 4.16: risoluzione spaziale del Kalman filter (in nero) sui dati simulati MC di elio, in funzione dell'energia media depositata in ciascun range.

### 4.4.3 Media dei residui e risoluzione angolare

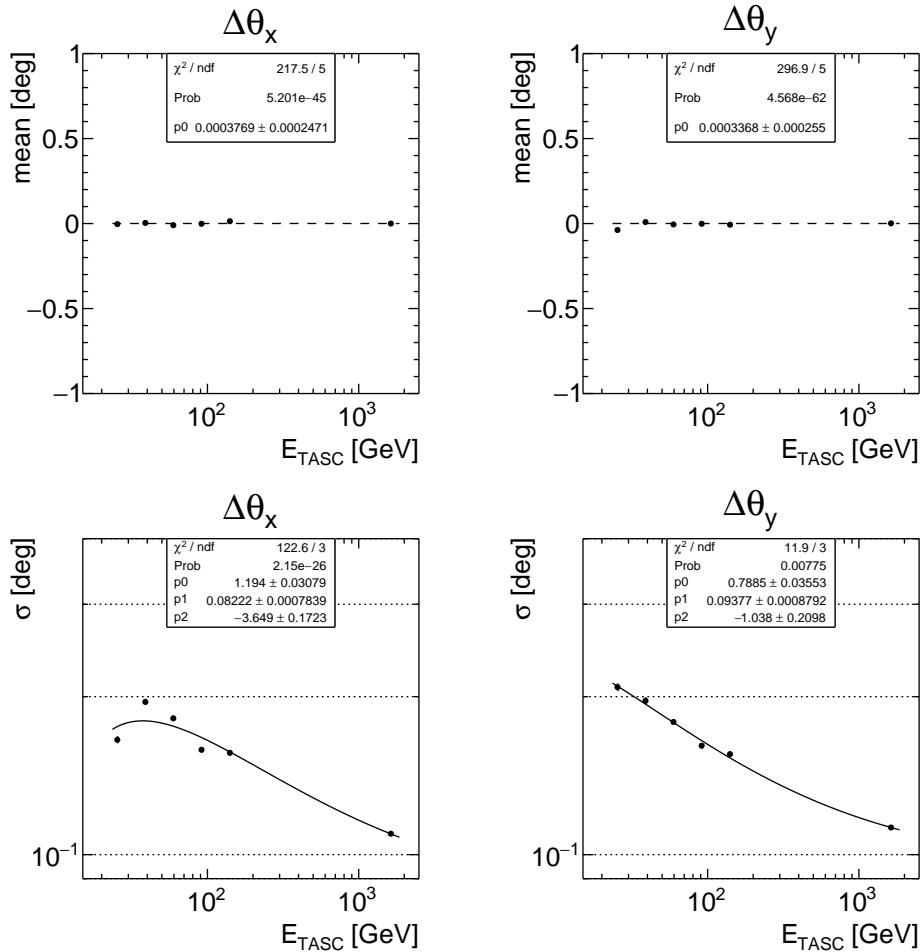


Figura 4.17: nei grafici *sopra* il valore medio della distribuzione dei residui; *sotto* la risoluzione angolare del Kalman filter (in *nero*) sui dati simulati MC di elio, in funzione dell'energia media depositata in ciascun range.

## 4.5 Kalman filter: confronto qualitativo

Dai grafici sopra riportati e dagli altri in appendice ([sezione A.1](#)) è possibile compiere le seguenti osservazioni:

- la risoluzione del KF ha un miglioramento all'aumentare dell'energia, con un incremento  $\simeq 2$  tra le basse (20, 31) GeV e le alte energie (173,  $2 \times 10^4$ ) GeV per le variabili  $x_0$  spaziale ed  $\theta_x$  angolare.
- la risoluzione dell'algoritmo Kalman filter è migliore di quella del metodo di ricostruzione tramite CNN, il rapporto tra le due risoluzioni è circa di un fattore  $\simeq 10$  sia alle basse che alle alte energie. Ciò conferma l'affidabilità dell'algoritmo KF. Bisogna però anche far presente che il Kalman filter, a differenza della CNN, ricostruisce la traiettoria a partire dalle misure effettuate non nel TASC, bensì nell'IMC che ha una segmentazione molto più fine e quindi anche una maggiore precisione, come spiegato nel [Capitolo 2](#).

Nel prossimo capitolo ([Capitolo 5](#)) i risultati del metodo KF sui dati della simulazione MC di CALET vengono sfruttati per calcolare la risoluzione della CNN su dati *di volo*, ovvero dati veri di eventi registrati dal calorimetro CALET.

# Capitolo 5

## Analisi dei dati di volo

### 5.1 Introduzione

In questo capitolo viene effettuata l'analisi dei dati di volo degli eventi registrati da CALET nel periodo compreso tra gennaio e marzo 2019.

Poiché nei dati di volo sono registrati eventi associati a particelle diverse, è stato necessario selezionare solo i dati relativi a quelle da analizzare: nel nostro caso i nuclei di elio.

Successivamente, sono state confrontate le prestazioni di ricostruzione della traiettoria con il metodo dell'asse dello sciame e con la rete neurale in due situazioni distinte: rete allenata solo su nuclei di elio e rete allenata sul dataset combinato di p e He.

(Dato che nel capitolo [Capitolo 4](#) non sono state osservate differenze significative tra le predizioni effettuate dalla CNN allenata su nuclei di elio e quella allenata sul dataset combinato di p e He, in questo capitolo vengono esposti solo i risultati ottenuti dalla CNN allenata su nuclei di elio. Per i risultati sui dati di volo della CNN allenata con il dataset combinato di p e He si rimanda all'[Appendice B](#)).

### 5.2 Selezione delle particelle

#### 5.2.1 Misura della carica

Viene adesso spiegato il processo di selezione dei dati, quindi i criteri con cui vengono registrati gli eventi e le misure con cui viene ricostruita la carica delle particelle.

Un raggio cosmico che attraversa il calorimetro è registrato come evento nei dati di volo solo se soddisfa una serie di condizioni di controllo. Deve infatti ‘accendere’ il *trigger high energy* (HET), al quale viene poi applicata la conferma offline del trigger,

ovvero la seconda condizione di controllo: l'energia depositata nel primo piano del TASC deve essere maggiore di 100 MIP equivalenti (*Minimum Ionizing Power*).

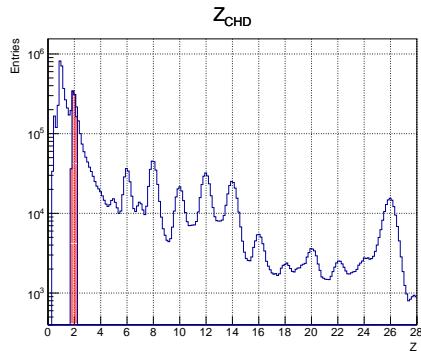
Di questi eventi vengono poi scartati tutti quelli in cui:

o l'algoritmo KF non è riuscito a ricostruire la direzione della particella perché il numero di punti traccia nell'IMC in una o in entrambe le viste è  $< 4$ , o la direzione trovata da KF è incompatibile con l'asse dello sciame ricostruito nel TASC.

Infine, viene applicata la condizione di ‘accettanza A’, ovvero vengono selezionati solo quegli eventi in cui la particella attraversa la faccia superiore del CHD e raggiunge quella inferiore del TASC restando a 2 cm dai bordi (questa condizione permette di selezionare gli eventi con il miglior contenimento dello sciame adronico).

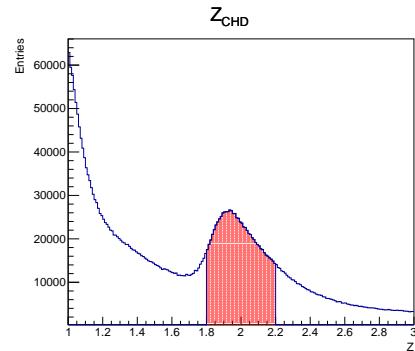
Per ricostruire la carica della particella primaria ne viene innanzitutto estrapolata la traccia nell'IMC e all'indietro nei due piani del CHD, in modo da individuare le strisce attraversate dal raggio cosmico e misurare la perdita di energia per unità di lunghezza ( $dE/dx$ ).

Sfruttando le due misure di perdita di energia per unità di lunghezza è possibile ricostruire la carica della particella primaria, come viene mostrato in [Figura 5.1](#) e [Figura 5.2](#), dove è possibile distinguere tutti i nuclei dal protone al ferro.



[Figura 5.1:](#) in *blu* la carica ricostruita nei due piani del CHD per valori compresi tra 0 e 28; in *rosso* i nuclei di elio selezionati nel range (1.8, 2.2).

Si osservano i picchi in corrispondenza dei nuclei (p, He, C, O, Ne, Mg, Si, S, Ar, Ca, Ti, Cr, Fe).



[Figura 5.2:](#) in *blu* lo zoom della carica ricostruita nei due piani del CHD per valori compresi tra 1 e 3; in *rosso* i nuclei di elio selezionati nel range (1.8, 2.2).

(I grafici della carica misurata dal CHD nelle viste  $xz$  e  $yz$  sono riportati in [sezione B.5: Campione di p e nuclei di He dei dati di volo](#))

L'analisi dei dati di volo è stata effettuata su nuclei di elio, selezionati richiedendo che la carica  $Z$  ricostruita nei due piani del CHD, tramite la procedura sopra descritta, sia  $1.8 < Z < 2.2$ .

### 5.2.2 Fit spettrale con legge di potenza

Sono stati inoltre prodotti anche i grafici della distribuzione dell'energia depositata nel TASC, sia nel caso con che senza selezione sui nuclei di elio.

Entrambi i grafici sono stati disegnati scegliendo come valori per l'asse delle ascisse  $\log_{10}(E_{\text{TASC}})$ ; ciò permette di osservare che, a partire da  $\log_{10}(E_{\text{TASC}}) \simeq 1.77$ , ovvero  $E_{\text{TASC}} \simeq 58.89$  [GeV] la distribuzione di energia segue un andamento *a legge di potenza*. Questo evidenzia una proporzionalità tra l'energia cinetica delle particelle cosmiche che attraversano il rivelatore e l'energia depositata.

Tale proporzionalità è stata verificata effettuando il fit delle distribuzioni di entrambi i grafici tramite la funzione

$$N(E_{\text{TASC}}) = N_0 10^{(1-\gamma)E_{\text{TASC}}}$$

con  $N_0$  fattore di normalizzazione e  $\gamma$  indice spettrale. "[0]\*pow(10, (1-[1])\*x)" nella macro di ROOT. I grafici ottenuti dal fit sono riportati in [Figura 5.3](#) e [Figura 5.4](#).

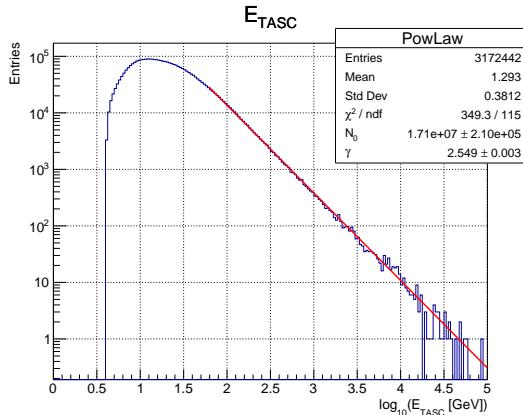


Figura 5.3: in blu l'istogramma della distribuzione di energia depositata nel TASC da tutte le particelle; in rosso il fit con legge di potenza.

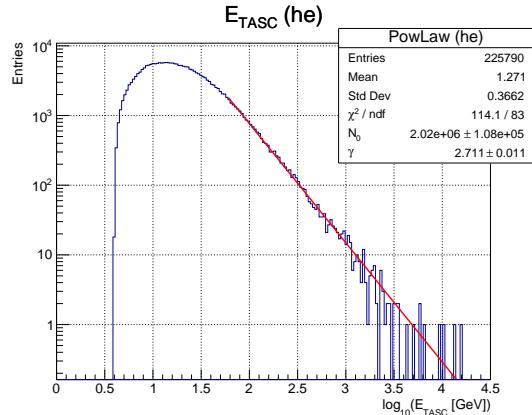


Figura 5.4: in blu l'istogramma della distribuzione di energia depositata nel TASC dai nuclei di elio; in rosso il fit con legge di potenza.

Si noti che i grafici non forniscono una misura dello spettro (flusso rispetto ad energia cinetica,  $dN/dE$ ) di energia dei raggi cosmici, ma riportano il conteggio  $dN$

degli eventi registrati in ciascun range di energia  $dE$  depositata nel TASC, in pratica l'integrale del flusso.

Dal fit della distribuzione di energia depositata da tutte le particelle (a sinistra) e solo dai nuclei di elio (a destra) si ricavano valori di  $\gamma = 2.549 \pm 0.003$ ,  $\gamma = 2.711 \pm 0.011$  rispettivamente, in accordo con i valori osservati in CALET.

### 5.3 Risultati della CNN sui dati di volo

I dati selezionati sono stati dunque passati in input alle due CNN precedentemente descritte ([Capitolo 3](#)) ed i valori predetti da queste sono stati confrontati con quelli ricostruiti tramite il metodo dell'asse dello sciame.

Per entrambe le CNN, quindi sia per quella con training solo su nuclei di elio sia per quella con training su protoni e nuclei di elio, sono state calcolate e graficate le distribuzioni dei residui di ciascuna coordinata dei punti di impatto ( $x_0, y_0, x_1, y_1$ ) e delle variabili angolari ( $\theta_x, \theta_y$ ).

Le distribuzioni sono state analizzate in 6 intervalli di energia depositata  $E_{TASC}$ , come descritto nella [sezione 5.2](#), selezionando particelle con carica  $1.8 < Z < 2.2$  (nuclei di he).

	Range [GeV]	$\langle E_{TASC} \rangle$ [GeV]	Eventi
20	$\longleftrightarrow$	31	24.79
31	$\longleftrightarrow$	47	37.83
47	$\longleftrightarrow$	73	57.66
73	$\longleftrightarrow$	112	88.79
112	$\longleftrightarrow$	173	136.19
173	$\longleftrightarrow$	$2 \times 10^4$	378.17
			3487

I valori riportati nella tabella relativi al numero di eventi ed al valore medio dell'energia depositata dai nuclei di elio in ciascun range sono stati calcolati tramite il software ROOT. Sono stati prodotti istogrammi con numero di bins pari a 200 e da questi sono stati estratti i valori riportati sopra.

Il criterio adottato nella suddivisione del range di energia  $(20, 2 \times 10^4)$  [GeV] è il seguente: sono stati utilizzati intervalli di ampiezza costante in scala logaritmica, ovvero tali per cui  $\log_{10}(E_{i+1}/E_i) = \text{const}$  per tutti gli intervalli tranne che per l'ultimo. Da 173 a  $2 \times 10^4$  [GeV] infatti, per avere una statistica confrontabile con quella degli intervalli precedenti, si è scelto di estendere l'intervallo fino a raccogliere tutti gli eventi rimasti e raggiungere quindi un numero di entries confrontabile con quello dell'intervallo da 112 a 173 [GeV].

L’analisi è stata inferiormente limitata ad eventi con  $E_{\text{TASC}} > 20$  [GeV] per evitare di considerare nuclei di elio non interagenti, che costituiscono particelle al minimo di ionizzazione nell’attraversare il calorimetro.

Il valore  $\langle E_{\text{TASC}} \rangle$  rappresenta l’energia media depositata nei piani del TASC per ciascun range in cui è stato suddiviso il campione da analizzare. Tale valore medio è stato estratto direttamente dalla distribuzione della variabile  $E_{\text{TASC}}$  negli intervalli sopra specificati.

Nelle prossime due sezioni, ([sezione 5.4](#)) e ([sezione 5.5](#)), vengono presentati i grafici delle distribuzioni dei residui delle variabili di impact point e delle variabili angolari. I grafici sono stati prodotti tramite il software ROOT.

(In questo capitolo sono stati inseriti solo i grafici relativi alla variabile  $x_0$  spaziale ed  $\theta_x$  angolare. Per gli altri consultare l’[Appendice B](#)).

Si osservi che, a differenza dell’analisi svolta nel [Capitolo 4](#), in cui è stato possibile confrontare i valori predetti dalla rete con quelli *veri* della simulazione MC, in questa seconda analisi ciò non è più possibile. Infatti, i dati analizzati non sono dati simulati, bensì sono dati di volo registrati da CALET, per cui non disponiamo di un valore *vero* che, anzi, è proprio ciò che si cerca di ricostruire.

Per avere quindi una stima della risoluzione delle CNN e del metodo dell’asse dello sciame sui dati di volo si è dunque effettuato il confronto con i valori ricostruiti dal metodo *Kalman Filter* (KF), applicato sui medesimi eventi.

Segue allora che la risoluzione della CNN misurata (*mis*) in questa analisi, sarà pari alla radice della somma in quadratura della risoluzione del KF e della risoluzione *effettiva* della CNN sui dati di volo; ovvero

$$\sigma_{\text{mis}} = \sqrt{\sigma_{\text{CNN}}^2 + \sigma_{\text{KF}}^2}$$

Un confronto tra la risoluzione effettiva della CNN sui dati di volo e sui dati simulati MC viene eseguito nella [sezione 5.6](#).

## 5.4 Distribuzioni dei residui

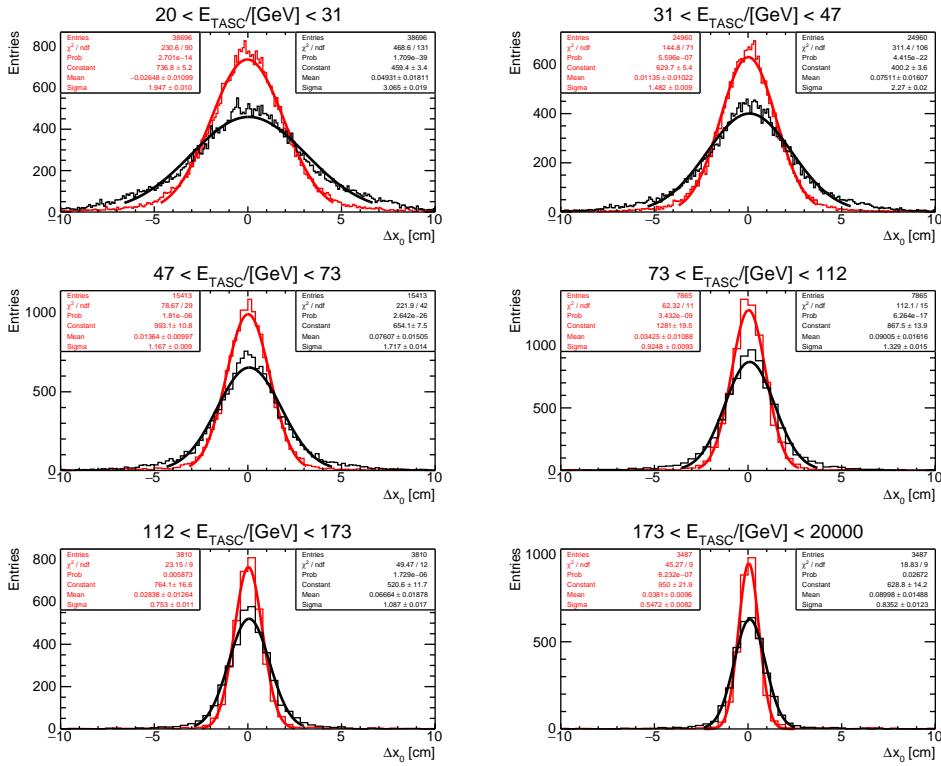


Figura 5.5: distribuzione dei residui della variabile di impact point  $x_0$  predetta dalla CNN allenata sui nuclei di elio (istogramma in rosso) e dal metodo dell'asse dello sciamo (istogramma in nero). Le curve sovrapposte agli istogrammi costituiscono i fit gaussiani agli istogrammi del medesimo colore.

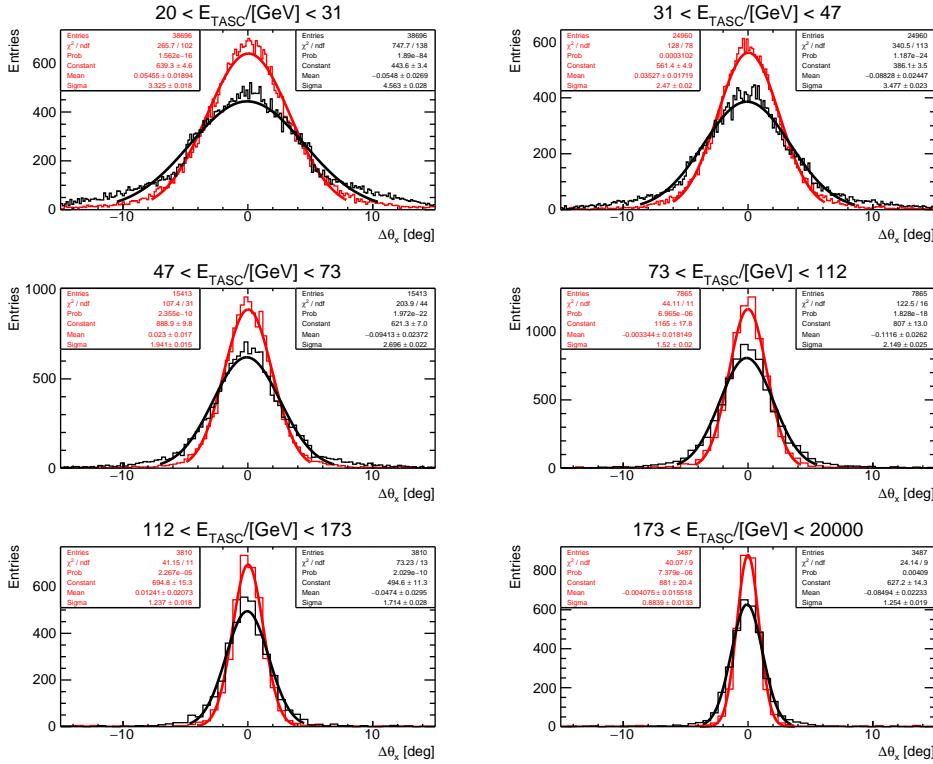


Figura 5.6: distribuzione dei residui della variabile angolare  $\theta_x$  predetta dalla CNN allenata sui nuclei di elio (istogramma in rosso) e dal metodo dell'asse dello sciame (istogramma in nero). Le curve sovrapposte agli istogrammi costituiscono i fit gaussiani agli istogrammi del medesimo colore.

## 5.5 Media dei residui e risoluzione spaziale

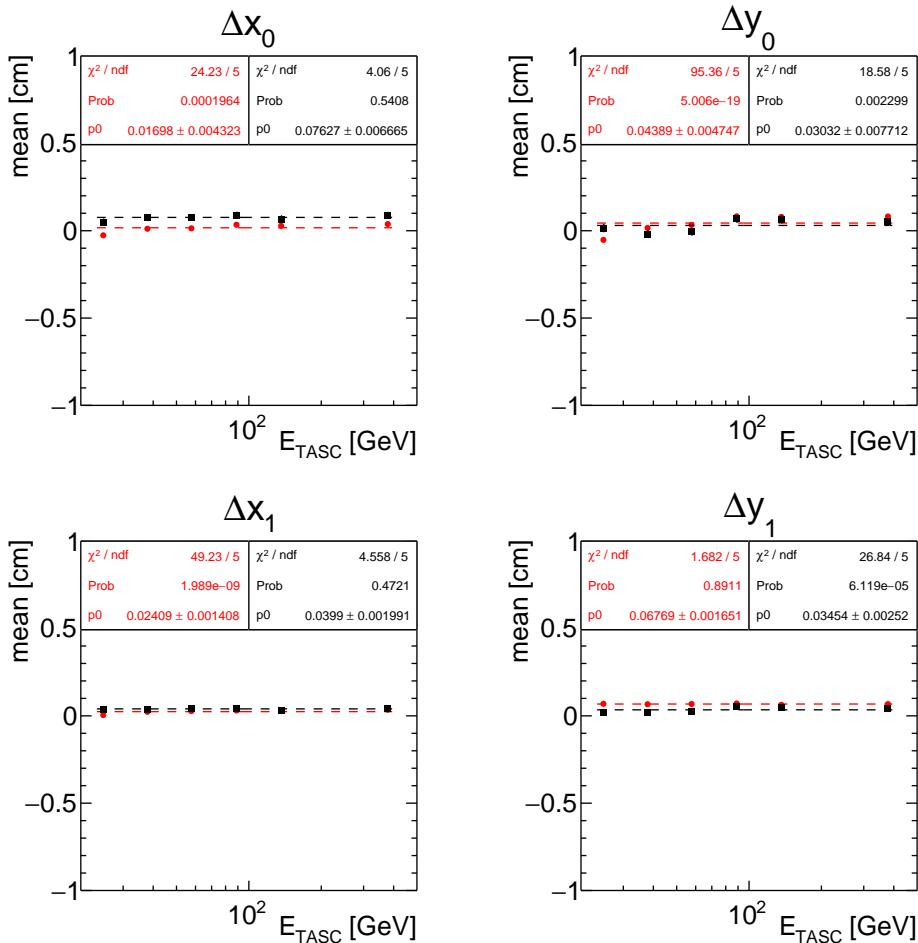


Figura 5.7: valore medio della distribuzione dei residui dei punti di impatto della CNN allenata sui nuclei di elio (in rosso) e del metodo dell'asse dello sciame (in nero), in funzione dell'energia media depositata in ciascun range.

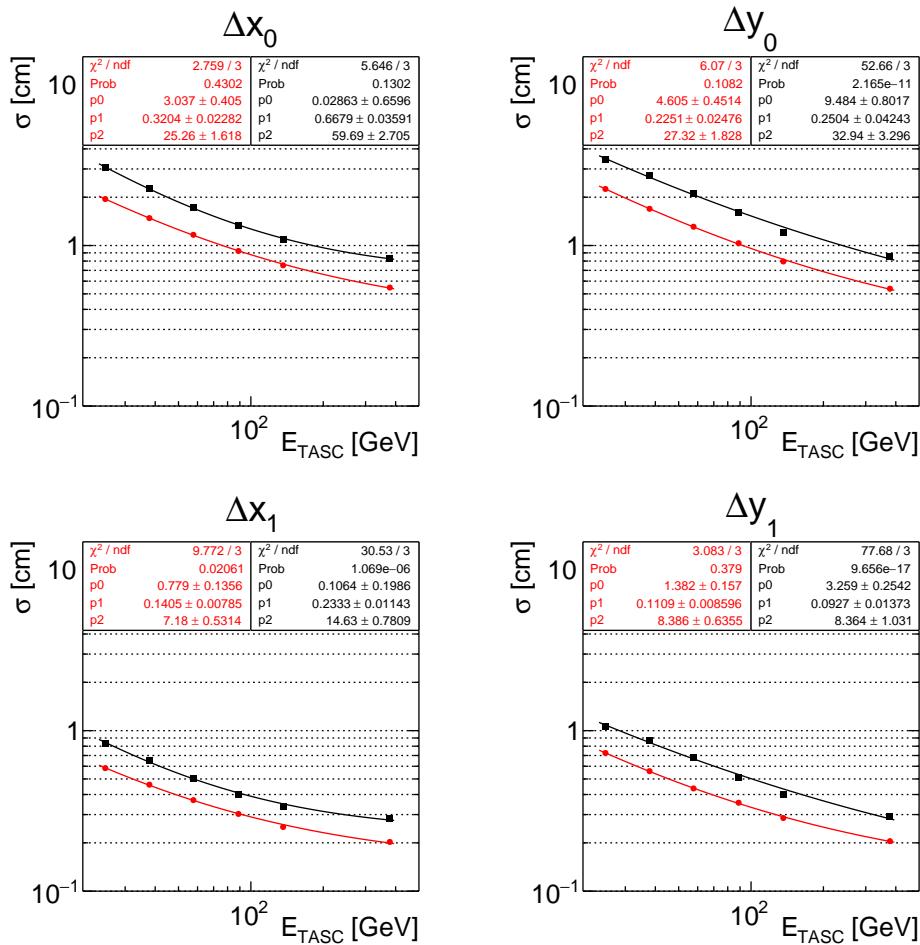


Figura 5.8: risoluzione spaziale della CNN allenata sui nuclei di elio (in rosso) e del metodo dell'asse dello sciame (in nero), in funzione dell'energia media depositata in ciascun range.

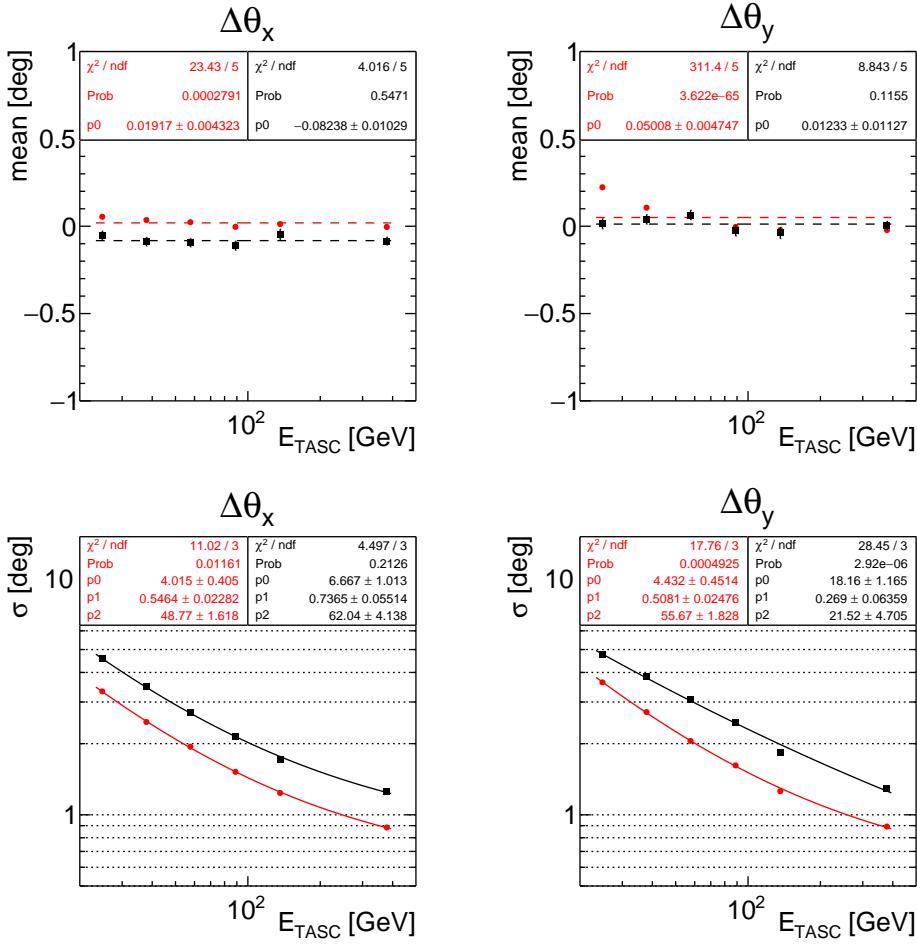


Figura 5.9: nei grafici *sopra* il valore medio della distribuzione dei residui; *sotto* la risoluzione angolare della CNN allenata sui nuclei di elio (in rosso) e del metodo dell’asse dello sciame (in nero), in funzione dell’energia media depositata in ciascun range.

Si osserva che, come per i dati simulati MC, anche per i dati di volo (FD) i residui della CNN sono minori rispetto a quelli dell’asse dello sciame, quindi si ha una migliore risoluzione, che aumenta all’aumentare dell’energia media del range.

Va d’altronde anche notato che per i FD la risoluzione della CNN è confrontabile con quella ottenuta sui dati MC: ciò è indice di ‘stabilità’ della rete e di una buona capacità predittiva, dato che i FD non appartengono al dataset di training e quindi sarebbe stato lecito attendersi un peggioramento della risoluzione.

## 5.6 Dati di volo: confronto quantitativo

Dai grafici sopra riportati possiamo compiere le seguenti osservazioni:

Confronto tra CNN e metodo dell'asse dello sciame: innanzitutto si osserva che in ogni range di energia la risoluzione spaziale ed angolare della CNN è migliore di quella del metodo dell'asse dello sciame. Inoltre si osserva anche che la risoluzione aumenta con l'aumentare dell'energia.

Infatti, se a basse energie (20, 31) GeV, la coordinata  $x_0$  e la variabile angolare  $\theta_x$  della CNN hanno risoluzione  $\sigma_{x_0} = (1.95 \pm 0.01)$  cm e  $\sigma_{\theta_x} = (3.33 \pm 0.02)^\circ$ , si nota che, all'aumentare dell'energia, i valori di  $\sigma_{x_0}$  e  $\theta_x$  diminuiscono progressivamente, ovvero la larghezza della campana gaussiana si riduce, fino ad arrivare ad alte energie (5477,  $2 \times 10^4$ ) GeV dove  $\sigma_{x_0} = (0.55 \pm 0.01)$  cm e  $\sigma_{\theta_x} = (0.88 \pm 0.01)^\circ$ . Dunque, all'aumentare dell'energia si ha un incremento nella risoluzione spaziale di un fattore  $\approx 4$  e nella risoluzione angolare di circa un fattore  $\approx 3.7$ .

Come nell'analisi dati effettuata per il training sui dati simulati, anche in questo caso si è osservato che si ha un errore sistematico nella ricostruzione dei punti di impatto che porta il valore medio dei residui ad essere 'spostato' in positivo o in negativo rispetto allo zero. Questo errore è probabilmente dovuto ad una scelta non ottimale dei pesi durante il training da parte della CNN e, per essere risolto, si potrebbe provare ad effettuare nuovamente il training su un dataset con un numero più elevato di eventi.

Per effettuare il fit del valore medio delle distribuzioni dei residui si è utilizzata una funzione costante  $y = mx + q$ , "pol0" nella macro di ROOT; mentre la funzione utilizzata per effettuare il fit della risoluzione spaziale ed angolare è:

$$y = \frac{a}{\sqrt{x} + b + \frac{c}{x}}$$

con  $a, b, c$  parametri del fit. "[0] /TMath::Sqrt(x)+[1]+[2]/x" nella macro di ROOT.

Nel [sezione 5.3](#) è stato osservato che la risoluzione stimata per la CNN è in realtà una sovrastima della risoluzione *effettiva*, infatti

$$\sigma_{\text{mis}} = \sqrt{\sigma_{\text{CNN}}^2 + \sigma_{\text{KF}}^2}$$

Questo segue dal fatto che tale risoluzione è stata calcolata confrontando i punti di impatto ricostruiti dalla CNN e dal Kalman filter: ovvero il valore con cui è stato effettuato il confronto, differentemente dall'analisi dei dati MC, non è un valore 'vero', ma è a sua volta un valore ricostruito e quindi anche a questo va associata un'incertezza di ricostruzione, ovvero la risoluzione dell'algoritmo Kalman filter combinatorio sui dati della simulazione MC.

$$\sigma_{\text{CNN}} = \sqrt{\sigma_{\text{mis}}^2 - \sigma_{\text{KF}}^2}$$

Segue allora che, per poter risalire alla risoluzione effettiva della CNN sui FD, è necessario stimare la risoluzione del Kalman filter sui dati MC.

Si è dunque proceduto a stimare tali valori, i cui grafici sono riportati nell'[Appendice B](#), e sottrarli in quadratura dalla  $\sigma_{\text{mis}}$ .

In [Figura 5.10](#) viene riportata la risoluzione spaziale della CNN sui dati simulati MC (in rosso) e della CNN sui dati di volo FD (in nero). In [Figura 5.11](#) la risoluzione spaziale, con la medesima colorazione per la risoluzione sui dati simulati MC e sui dati di volo FD.

Si nota che il numero di misure della risoluzione è differente (14 per i dati simulati MC e 6 per i dati di volo FD): questo dipende da come è stato scelto di dividere il range di energia ( $20, 2 \times 10^4$ ) GeV. Segue allora che i valori dell'energia media depositata in ciascun range (ovvero le coordinate  $x$  nei grafici della [Figura 5.10](#) e [Figura 5.11](#)) non coincidono. La risoluzione della CNN sui dati di volo FD quindi risulta essere stimata fino a valori di energia  $E_{\text{TASC}} \simeq 378$  GeV e non oltre. Per poter stimare la risoluzione sui dati di volo FD a valori di energia maggiori (*alte energie*) servirebbe avere un numero maggiore di dati misurati da CALET rispetto a quelli impiegati in questo studio, così da migliorare la statistica anche a valori più elevati di energia. (Si vedano il [Capitolo 4](#) ed il [Capitolo 5](#) per maggiori dettagli sulla scelta di suddivisione del range di energia).

Si nota che i valori della risoluzione  $\sigma_{\text{CNN}}$  della CNN sui dati di volo sono dello stesso ordine di quelli della risoluzione  $\sigma_{\text{MC}}$  sui dati della simulazione MC. Questa considerazione è di notevole importanza: il fatto che la risoluzione non peggiori su un dataset di dati veri è indice di stabilità ed affidabilità della rete neurale. Significa che questa ha imparato a riconoscere le features utili al tracciamento, ovvero ha sviluppato una buona capacità nel compiere predizioni affidabili anche su eventi ‘nuovi’, ovvero non appartenenti al dataset di training.

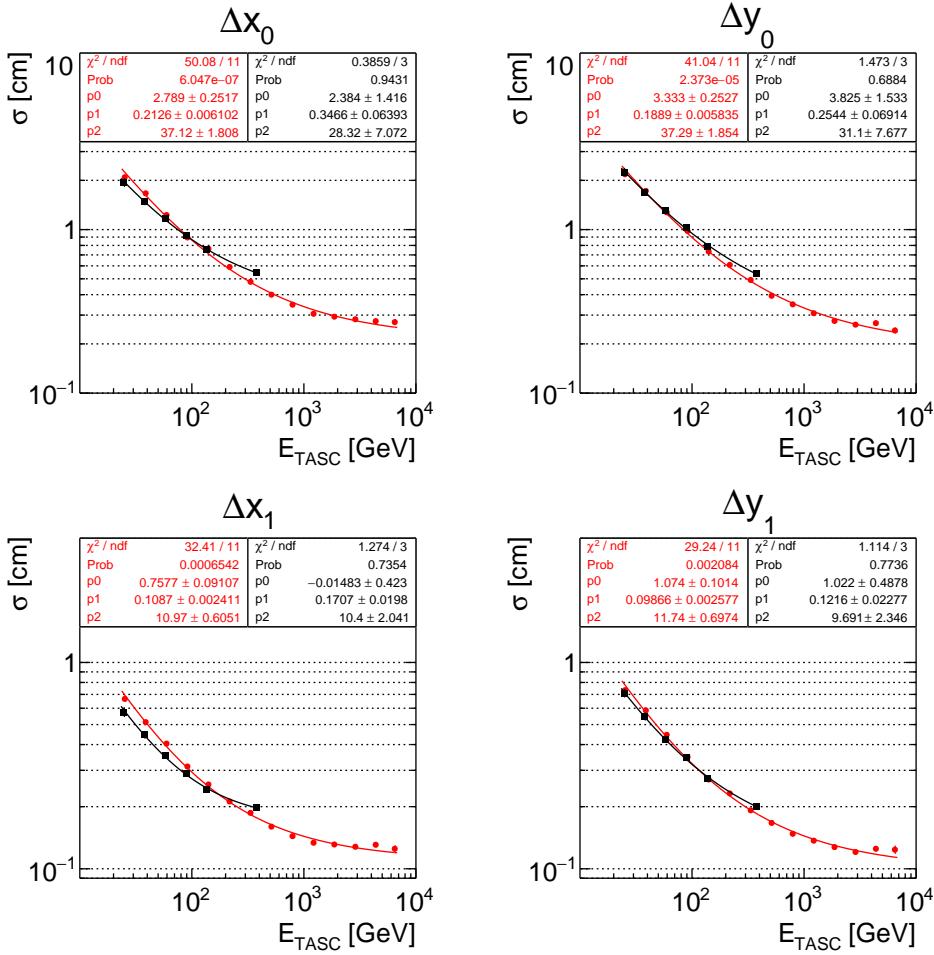


Figura 5.10: confronto tra la risoluzione spaziale della CNN sui dati simulati MC (in rosso) e sui dati di volo FD (in nero), in funzione dell’energia media depositata in ciascun intervallo.

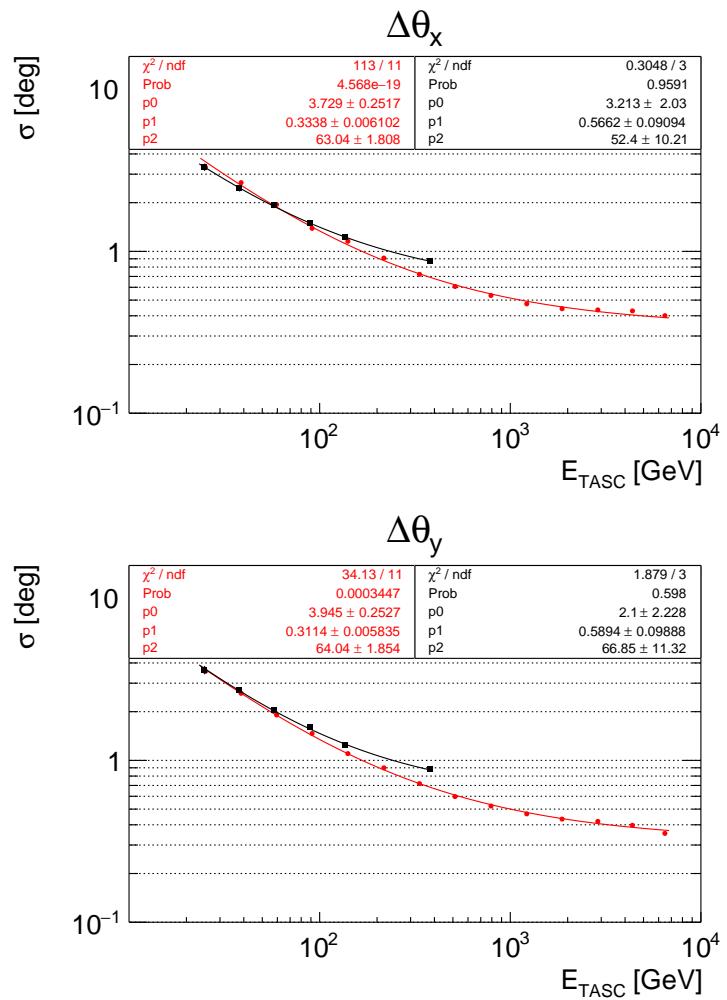


Figura 5.11: confronto tra la risoluzione angolare della CNN sui dati simulati MC (in rosso) e sui dati di volo FD (in nero), in funzione dell'energia media depositata in ciascun intervallo.

# Conclusioni

Il modello di rete neurale convoluzionale (CNN) sviluppato il questo lavoro di tesi è stato dunque applicato ai dati di volo di CALET e l'analisi della risoluzione delle predizioni effettuate ha fornito risultati promettenti. Le prestazioni ottenute con la CNN sono state quindi confrontate con il metodo dell'asse dello sciame e con l'algoritmo Kalman filter combinatorio (KF), implementato nel software di ricostruzione dell'esperimento CALET.

La CNN è stata allenata su due dataset distinti di eventi simulati MC: uno contenente eventi relativi a soli nuclei di elio e l'altro contenente eventi relativi sia a protoni che a nuclei di elio. La differenza nelle previsioni dei due modelli è risultata essere trascurabile, mostrando così come le CNN permettano di sviluppare ed allenare modelli più ‘flessibili’: ovvero è possibile produrre un unico modello di rete neurale in grado di effettuare il tracciamento sia di protoni che di nuclei di elio contemporaneamente, senza quindi perdere in risoluzione rispetto al caso di due modelli separati ed allenati uno per effettuare il tracciamento di protoni e l'altro per il tracciamento di nuclei di elio.

Inoltre, dal confronto dei risultati ottenuti con i diversi metodi impiegati nello studio, è risultato che il modello di rete neurale sviluppato costituisce un metodo di ricostruzione della traiettoria più preciso del metodo dell'asse dello sciame, ma, come previsto, peggiore rispetto al KF. Infatti, come spiegato nella [sezione 4.5](#), il KF effettua il tracciamento a partire dalle misure compiute nell'IMC e non nel TASC.

Sono state ottenute risoluzioni spaziali di circa  $\sim 2$  cm sul punto di impatto nel piano superiore del CHD e di circa  $\sim 0.6$  cm nel piano superiore del TASC alle basse energie, mentre di circa  $\sim 0.5$  cm sul punto di impatto nel piano superiore del CHD e di circa  $\sim 0.2$  cm nel piano superiore del TASC alle alte energie. Per quanto riguarda la risoluzione angolare sulla direzione di incidenza dei raggi cosmici nelle due proiezioni del rivelatore ( $xz$  e  $yz$ ) con la CNN si ha circa  $\sim 3^\circ$  alle basse energie e circa  $\sim 0.8^\circ$  alle alte energie.

Infine, non si è osservato un considerevole peggioramento della risoluzione spaziale ed angolare nell'applicare la CNN sui dati di volo FD. Infatti la risoluzione della CNN sui dati di volo FD risulta essere dello stesso ordine di grandezza di quella stimata sui

dati simulati MC, sebbene il range di energie su cui è stata misurata sia ridotto rispetto a quello dei dati simulati MC, come spiegato nella [sezione 5.6](#).

La risoluzione misurata sui dati di volo costituisce dunque una conferma dell'affidabilità del modello nel compiere previsioni attendibili su dati veri registrati dal calorimetro CALET.

## **Appendice A**

### **Metodi di tracciamento utilizzati in CALET**

## A.1 Kalman filter

L’identificazione e la ricostruzione della traccia in CALET è di massima importanza, infatti effettuare un tracciamento accurato è necessario per:

- identificare quali *paddle* (strisce) del CHD e quali fibre dell’IMC sono state attraversate dalla particella del raggio cosmico entrante, allo scopo di misurarne la carica ed individuare il punto di inizio dello sciame;
- definire l’accettanza geometrica e calcolare le variabili topologiche del calorimetro, come ad esempio la larghezza dello sciame rispetto alla traiettoria della particella, che vengono usate per effettuare la distinzione elettrone-adrone.

Queste operazioni possono risultare particolarmente dispendiose a livello computazionale per via dell’elevato numero di tracce di particelle secondarie nel TASC e dei molti ‘hits’ registrati nell’IMC. Gli hits devono essere suddivisi e raggruppati in gruppi di tracce ‘candidate’, per poi effettuare un fit. L’elevato numero di hits spuri da fissare, sommato a possibili inefficienze del rivelatore, costituiscono una difficoltà nell’identificazione della traccia reale della particella primaria.

In CALET è utilizzato un algoritmo di ricostruzione della traccia basato su un Kalman filter combinatorio [36]: si tratta di una tecnica ibrida di analisi dei dati per la ricostruzione simultanea di tracce (*track fitting*) e il fit dei vertici (*vertex fitting*) ampiamente usata negli esperimenti di alta energia agli acceleratori, ma mai usata in un esperimento calorimetrico come CALET.

L’algoritmo riesce a gestire un elevato numero di hits, come accade nell’IMC per sciami di alta energia, e ricostruire candidati di tracce multiple con un tempo di computazione ridotto, anche nel caso di inefficienze del rivelatore.

Il problema della ricostruzione della traiettoria in CALET può essere ricondotto a due problemi bidimensionali indipendenti, nelle viste *xz* ed *yz* rispettivamente, dal momento che i layers dell’IMC sono orientati nella direzione *x* o nella direzione *y* e posizionati a differenti altezze *z*.

Il processo inizia dal layer superiore dell’IMC: gli hits prossimi tra loro vengono raggruppati in un cluster per ciascun layer dell’IMC, il cui baricentro è passato al KF come possibile punto di traccia. Viene poi creato un candidato della traccia per ogni possibile combinazione di cluster nei primi due layer, includendo anche quelle tracce con hits mancanti. Ciascuna traccia candidata viene fissata con una linea retta, che viene fatta evolvere separatamente dalle altre, ovvero l’angolo di incidenza ed il punto di impatto (che formano il *vettore di stato*) sul layer successivo ( $k+1$ ) vengono calcolati ricorrendo solo alle informazioni (vettore di stato e matrice di covarianza) del layer precedente ( $k$ ).

Questo stato predetto può essere sfruttato per identificare quale cluster assegnare alla traccia nel layer  $(k + 1)$ : un possibile criterio è quello di associare il cluster più vicino alla misura di predizione effettuata. Ad ogni modo, ciò non costituisce un algoritmo affidabile, ad esempio un algoritmo che è resiliente ad ipotesi iniziali errate. Se la traccia di partenza è stimata in modo errato, c'è il rischio che alla traccia venga associata una misura errata, come un cluster di rumore o un punto di un'altra traccia.

La formulazione combinatoria del Kalman filter risolve questo problema tramite la creazione di una nuova traccia candidata per ciascun cluster sul layer  $(k + 1)$ , il quale si trova sufficientemente vicino allo stato previsto. La traccia del candidato viene quindi ‘ramificata’ in molti candidati e ciascuno di essi può essere fatto evolvere separatamente.

Per ogni nuovo ramo, l'informazione riguardo il punto della traccia nel layer  $(k + 1)$  è combinata con lo stato predetto, per produrre una stima ‘filtrata’ del vettore di stato e della matrice di covarianza del livello  $(k + 1)$ ; tale stato filtrato è utilizzato per effettuare nuovamente il fit della traccia e aggiornare il valore del  $\chi^2$ .

La procedura è allora iterata sul layer  $(k + 2)$  e così via, fino all'ultimo layer.

Ovviamente un algoritmo combinatorio di questo tipo può produrre un numero elevato di tracce candidate, le quali possono notevolmente rallentare la computazione. Per evitare di incappare in problematiche di questo genere, una traccia candidata può essere scartata se, in un punto qualsiasi della sua evoluzione, essa ha un  $\chi^2$  maggiore di un certo valore predefinito (tipicamente 10) o un numero troppo grande di punti mancanti (tipicamente  $> 2$ ).

Per eventi che generano sciami di alta energia nel TASC, con conseguente numero elevato di hits nel IMC dovuti al *backscattering*, al fine di ridurre le combinazioni di tracce ed il tempo necessario a processarle, vengono passati all'algoritmo solo i cluster IMC all'interno una ROI (*Region Of Interest*) definita dall'insieme degli hits in ciascun layer che si trovano entro  $\pm 5$  cm dal punto di intersezione con l'estrapolazione dell'asse dello sciame ricostruito nel TASC.

Infine, tra le tracce ricostruite dall'algoritmo, viene scelta come traccia più probabile (reale) del raggio cosmico primario quella che passa più vicina al *core* dello sciame nel TASC, ovvero la zona centrale con il massimo rilascio di energia [37].

## A.2 Metodo dell'asse dello sciame

Il *metodo dell'asse dello sciame* [38] è un algoritmo originariamente sviluppato per separare sciami elettronici ed adronici nei calorimetri.

In CALET viene principalmente impiegato per identificare gli elettroni all'interno del TASC; inoltre anche per effettuare la ricostruzione della direzione di attraversamento e dei punti di impatto sui piani del rivelatore.

L'algoritmo segue lo schema di procedimento, ripetuto evento per evento:

1. si ‘pesano’ le coordinate  $(x_{ij}, z_{ij})$  del piano  $xz$  ( $yz$ ) del TASC con energia depositata  $E_{ij}$ ;
2. con i valori ottenuti si costruiscono le matrici di *covarianza* per la vista  $x$  e per la vista  $y$ ;
3. si diagonalizzano le matrici e, una volta ottenuti gli autovalori, si risale ai corrispondenti autovettori, che forniscono una stima della direzione della particella.

Di seguito le coordinate  $(x_{ij}, z_{ij})$  ‘pesate’

$$\begin{aligned}\bar{x} &= \frac{\sum_{ij} x_{ij} E_{ij}}{\sum_{ij} E_{ij}} & \bar{z} &= \frac{\sum_{ij} z_{ij} E_{ij}}{\sum_{ij} E_{ij}} & \bar{xz} &= \frac{\sum_{ij} x_{ij} z_{ij} E_{ij}}{\sum_{ij} E_{ij}} \\ \bar{x^2} &= \frac{\sum_{ij} x_{ij}^2 E_{ij}}{\sum_{ij} E_{ij}} & \bar{z^2} &= \frac{\sum_{ij} z_{ij}^2 E_{ij}}{\sum_{ij} E_{ij}}\end{aligned}$$

con cui calcolare le matrici di covarianza dei baricentri  $x$  ( $y$ ) dell'evento considerato

$$\text{Cov}(x, z) = \begin{pmatrix} \bar{x^2} - \bar{x}^2 & \bar{xz} - \bar{x}\bar{z} \\ \bar{xz} - \bar{x}\bar{z} & \bar{z^2} - \bar{z}^2 \end{pmatrix}$$

Dove  $E_{ij}$  sono le energie depositate nei cristalli del TASC, l'indice  $i$  rappresenta il numero della barra (0, 15) e  $j$  il layer (0, 5) della vista  $xz$  ( $yz$ ).

Dalla matrice di covarianza si calcolano gli autovalori

$$\begin{pmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{pmatrix}$$

ed i corrispondenti autovettori  $\vec{u}$  e  $\vec{v}$ . L'autovettore  $\vec{v}$  costituisce una stima dell'asse dello sciame e fornisce quindi una ricostruzione della direzione della particella all'interno del calorimetro.

Il punto di impatto si ricava infine estrapolando a  $z = 0$  (piano superiore del CHD) la retta che ha direzione parallela a  $\vec{v}$  e passa per il baricentro  $(\bar{x}, \bar{z})$ .

# Appendice B

## Grafici

B.1	Kalman filter, campione di dati simulati MC	95
B.2	Campione di nuclei di He simulati MC	104
B.3	Campione di p e nuclei di He simulati MC	114
B.4	Campione di nuclei di He dei dati di volo	124
B.5	Campione di p e nuclei di He dei dati di volo	135
B.6	Selezione della carica dei nuclei	146

- **sezione B.1: Kalman filter, campione di dati simulati MC**, grafici dei residui delle variabili di impact point e delle variabili angolari, valore medio della distribuzione dei residui delle variabili spaziali e risoluzione spaziale, valore medio della distribuzione dei residui delle variabili angolari e risoluzione angolare calcolati su dati simulati MC di elio dal metodo Kalman filter;
- **sezione B.2: Campione di nuclei di He simulati MC**, grafici del *mean absolute error* (mae) e *mean square error* (mse), residui delle variabili di impact point e delle variabili angolari, valore medio della distribuzione dei residui delle variabili spaziali e risoluzione spaziale, valore medio della distribuzione dei residui delle variabili angolari e risoluzione angolare calcolati su dati simulati MC dalla CNN allenata su nuclei di elio;
- **sezione B.3: Campione di p e nuclei di He simulati MC**, grafici del *mean absolute error* (mae) e *mean square error* (mse), residui delle variabili di impact point e delle variabili angolari, valore medio della distribuzione dei residui delle variabili spaziali e risoluzione spaziale, valore medio della distribuzione dei residui delle variabili angolari e risoluzione angolare calcolati su dati simulati MC dalla CNN allenata su dati combinati di p e He;

- **sezione B.4: Campione di nuclei di He dei dati di volo**, grafici dei residui delle variabili di impact point e delle variabili angolari, valore medio della distribuzione dei residui delle variabili spaziali e risoluzione spaziale, valore medio della distribuzione dei residui delle variabili angolari e risoluzione angolare calcolati su dati di volo FD dalla CNN allenata su nuclei di elio;
- **sezione B.5: Campione di p e nuclei di He dei dati di volo**, grafici dei residui delle variabili di impact point e delle variabili angolari, valore medio della distribuzione dei residui delle variabili spaziali e risoluzione spaziale, valore medio della distribuzione dei residui delle variabili angolari e risoluzione angolare calcolati su dati di volo FD dalla CNN allenata su dati combinati di p e He;
- **sezione B.6: Selezione della carica dei nuclei**, grafici di ricostruzione della carica nel CHD, con somma delle due viste e viste singole  $xz$  e  $yz$ .

## B.1 Kalman filter, campione di dati simulati MC

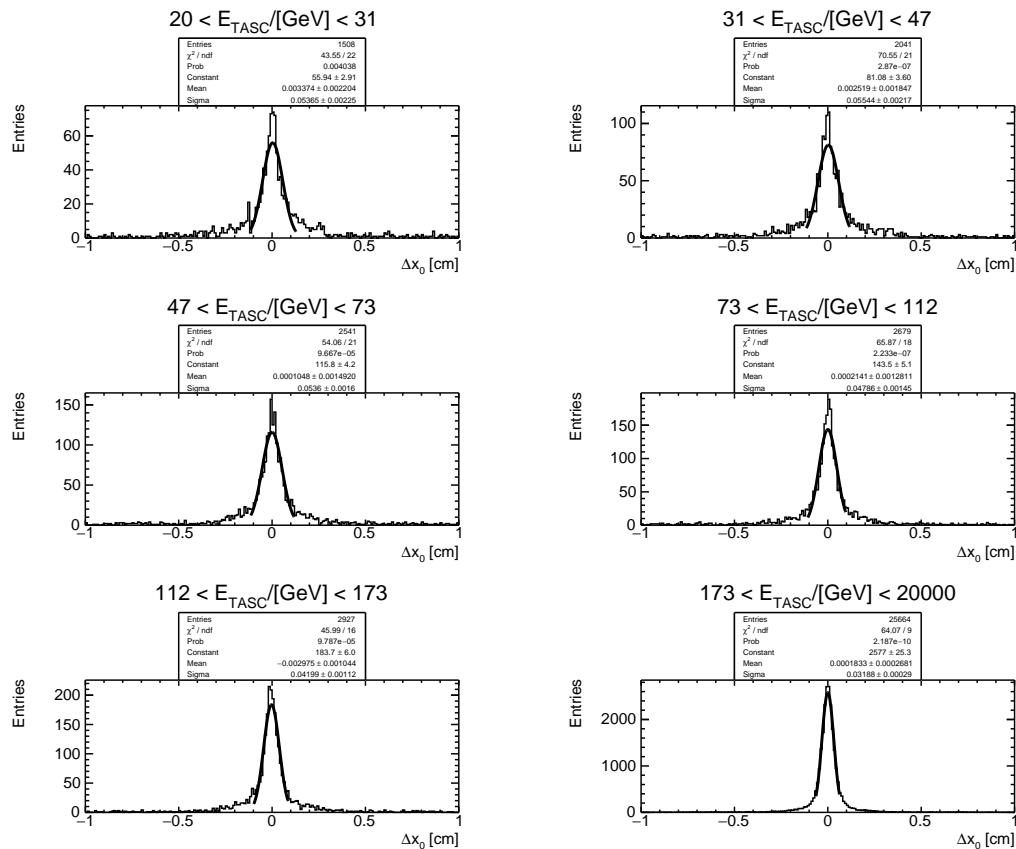


Figura B.1: distribuzione dei residui della variabile di impact point  $x_0$  pre detta dal Kalman filter (istogramma in nero) sui dati simulati MC di elio. La curva sovrapposta agli istogrammi costituisce il fit gaussiano agli istogrammi.

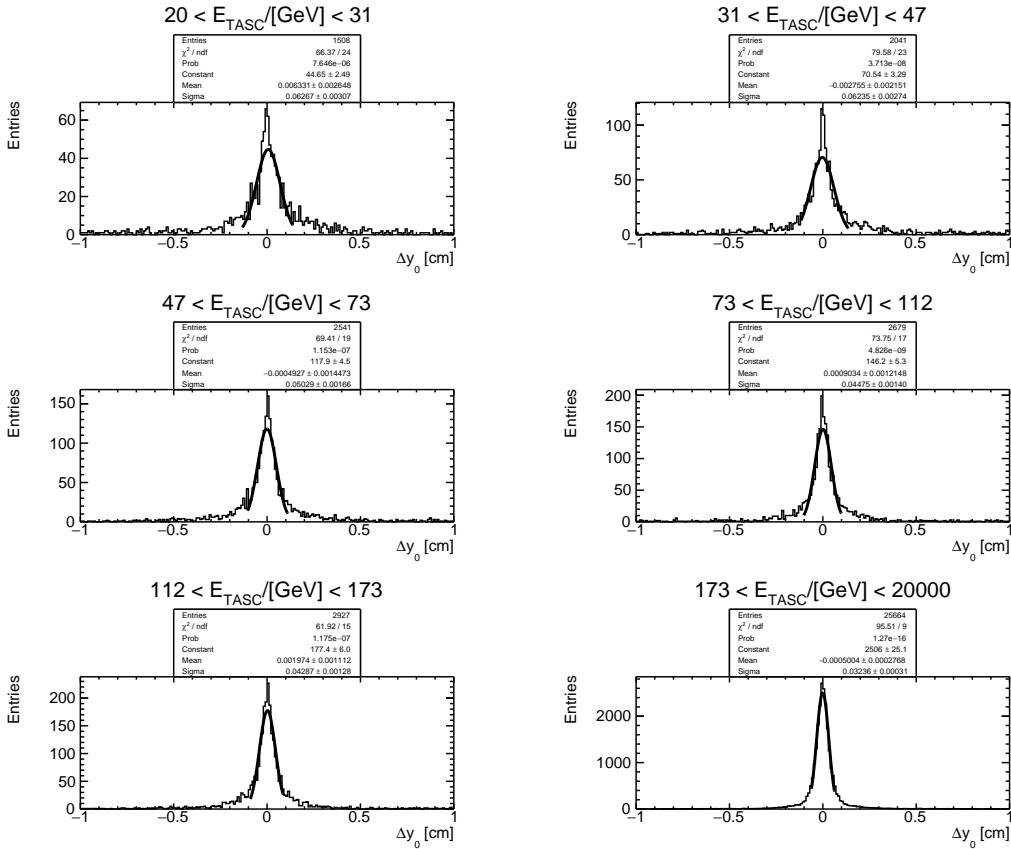


Figura B.2: distribuzione dei residui della variabile di impact point  $y_0$  predetta dal Kalman filter (istogramma in nero) sui dati simulati MC di elio. La curva sovrapposta agli istogrammi costituisce il fit gaussiano agli istogrammi.

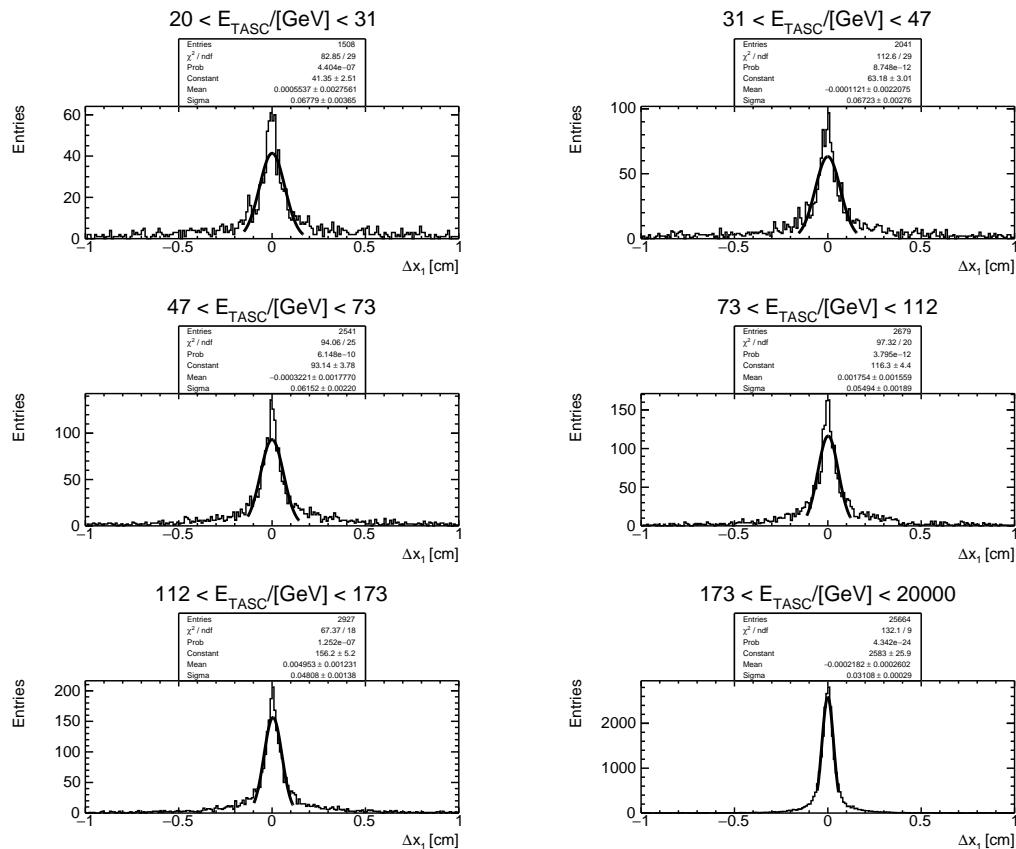


Figura B.3: distribuzione dei residui della variabile di impact point  $x_1$  predetta dal Kalman filter (istogramma in nero) sui dati simulati MC di elio. La curva sovrapposta agli istogrammi costituisce il fit gaussiano agli istogrammi.

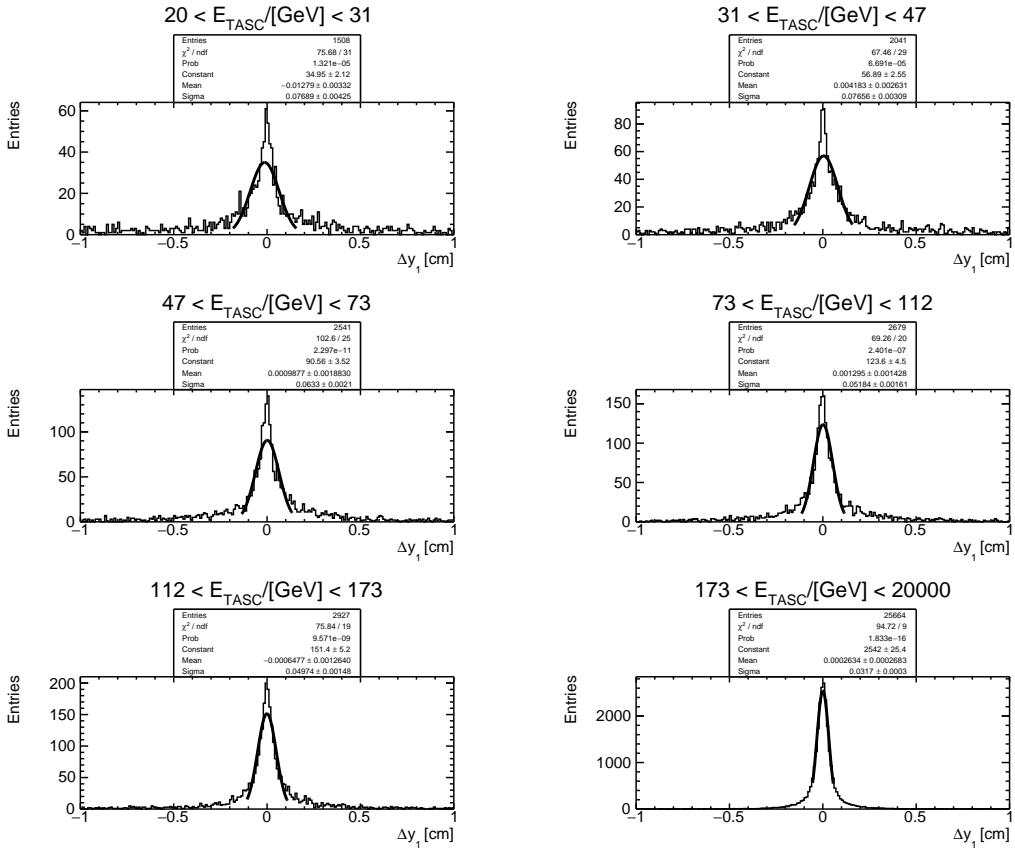


Figura B.4: distribuzione dei residui della variabile di impact point  $y_1$  predetta dal Kalman filter (istogramma in nero) sui dati simulati MC di elio. La curva sovrapposta agli istogrammi costituisce il fit gaussiano agli istogrammi.

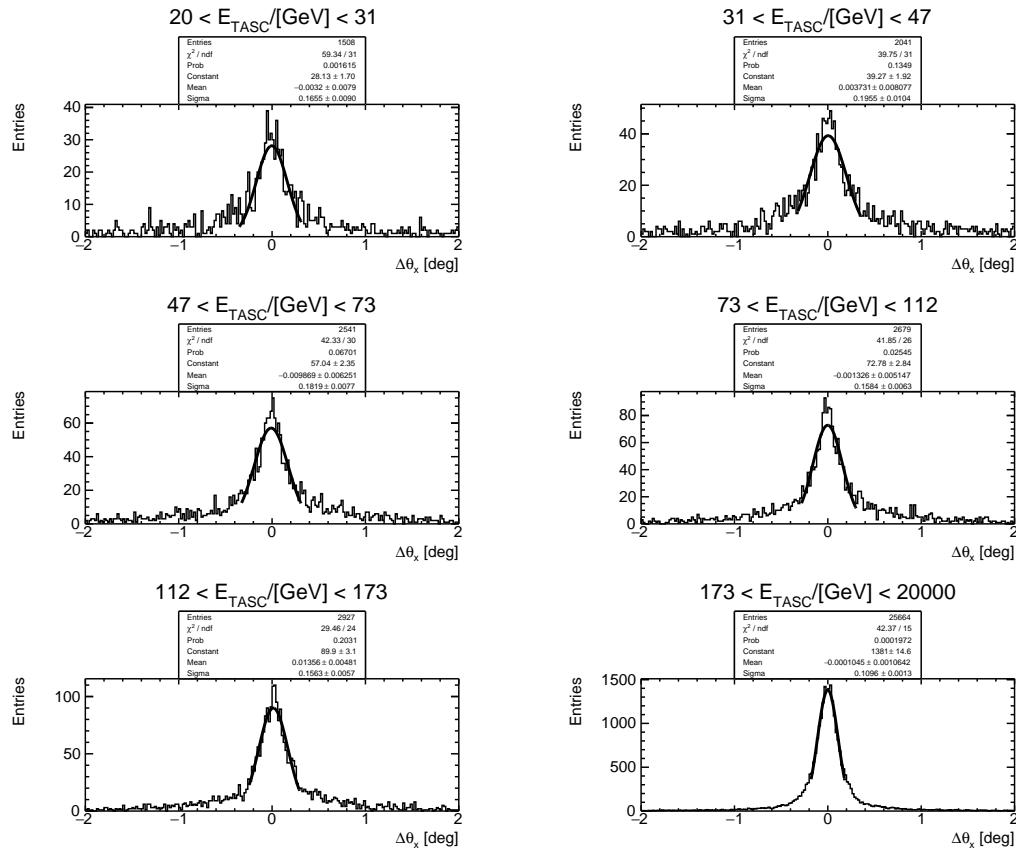


Figura B.5: distribuzione dei residui della variabile angolare  $\theta_x$  predetta dal Kalman filter (istogramma in nero) sui dati simulati MC di elio. La curva sovrapposta agli istogrammi costituisce il fit gaussiano agli istogrammi.

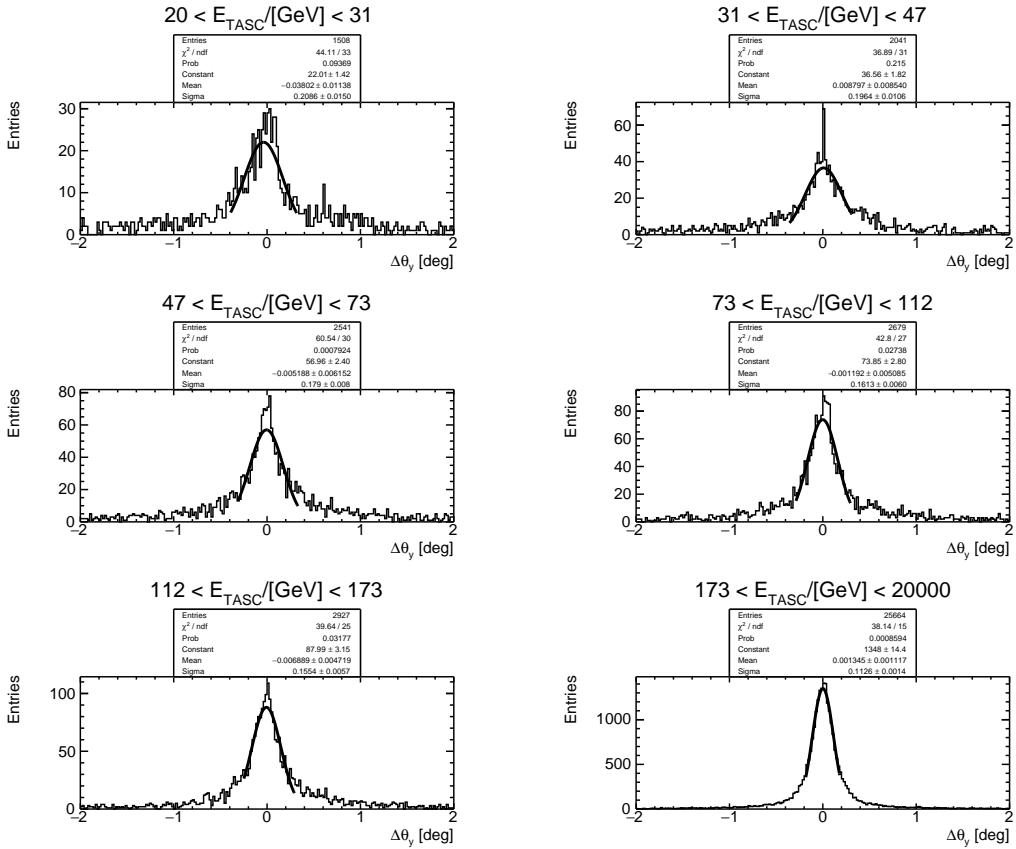


Figura B.6: distribuzione dei residui della variabile angolare  $\theta_y$  predetta dal Kalman filter (istogramma in nero) sui dati simulati MC di elio. La curva sovrapposta agli istogrammi costituisce il fit gaussiano agli istogrammi.

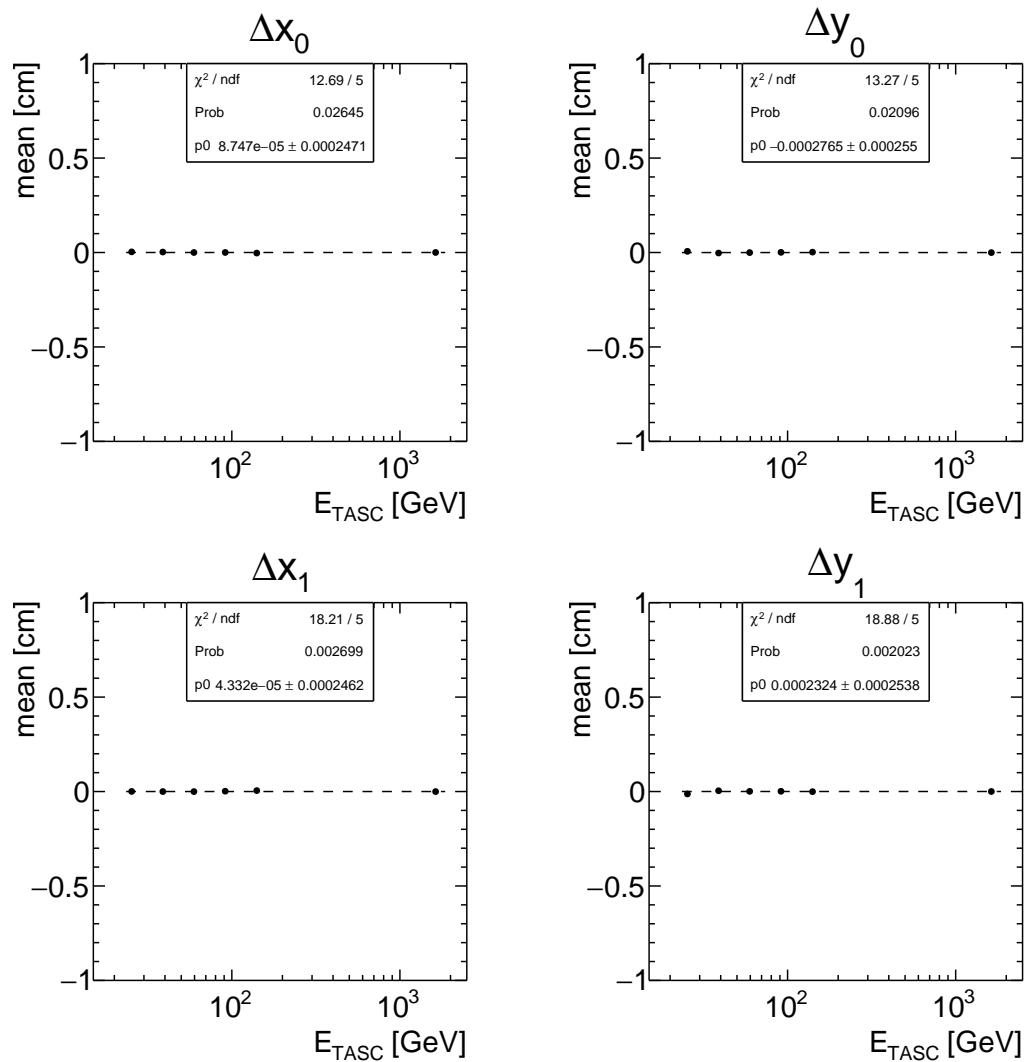


Figura B.7: valore medio della distribuzione dei residui dei punti di impatto predetti dal Kalman filter (in nero) sui dati simulati MC di elio, in funzione dell'energia media depositata in ciascun range.

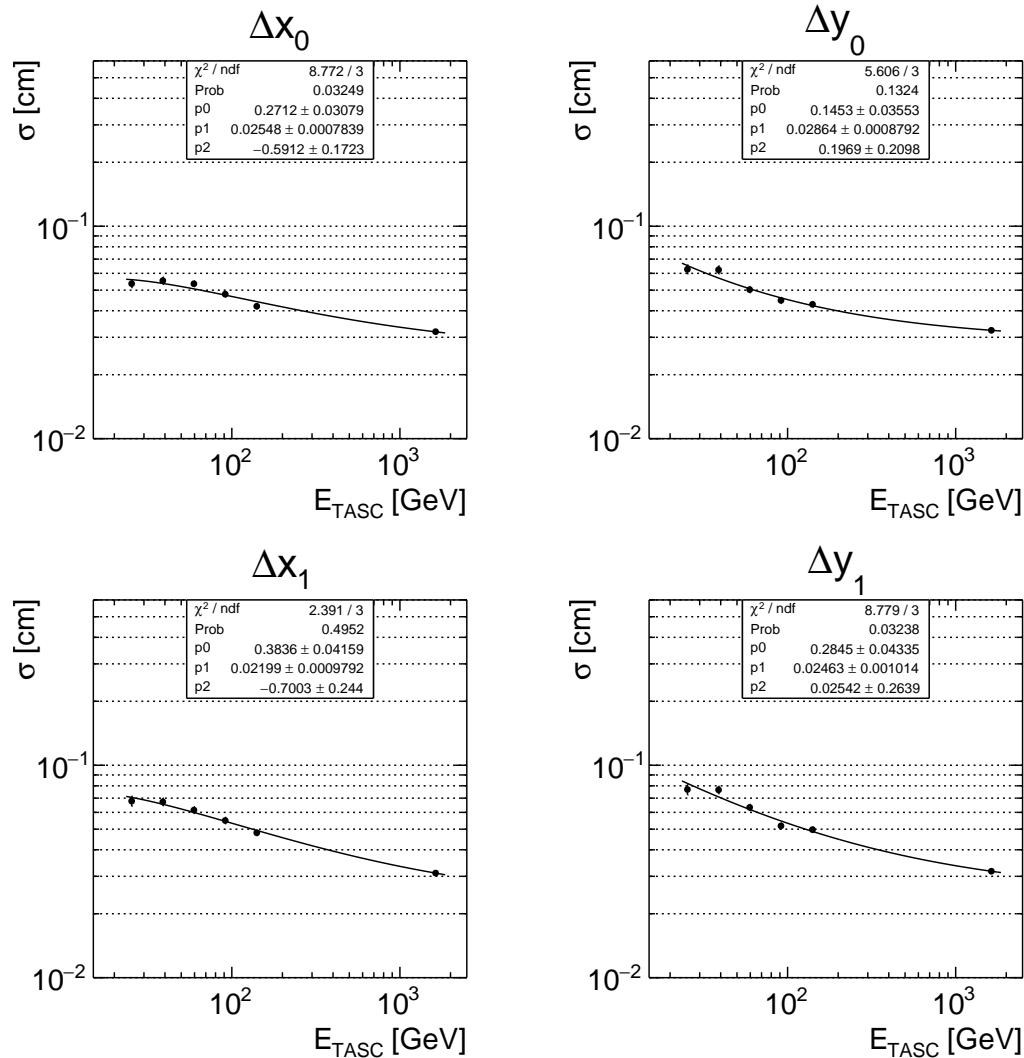


Figura B.8: risoluzione spaziale del Kalman filter (in nero) sui dati simulati MC di elio, in funzione dell'energia media depositata in ciascun range.

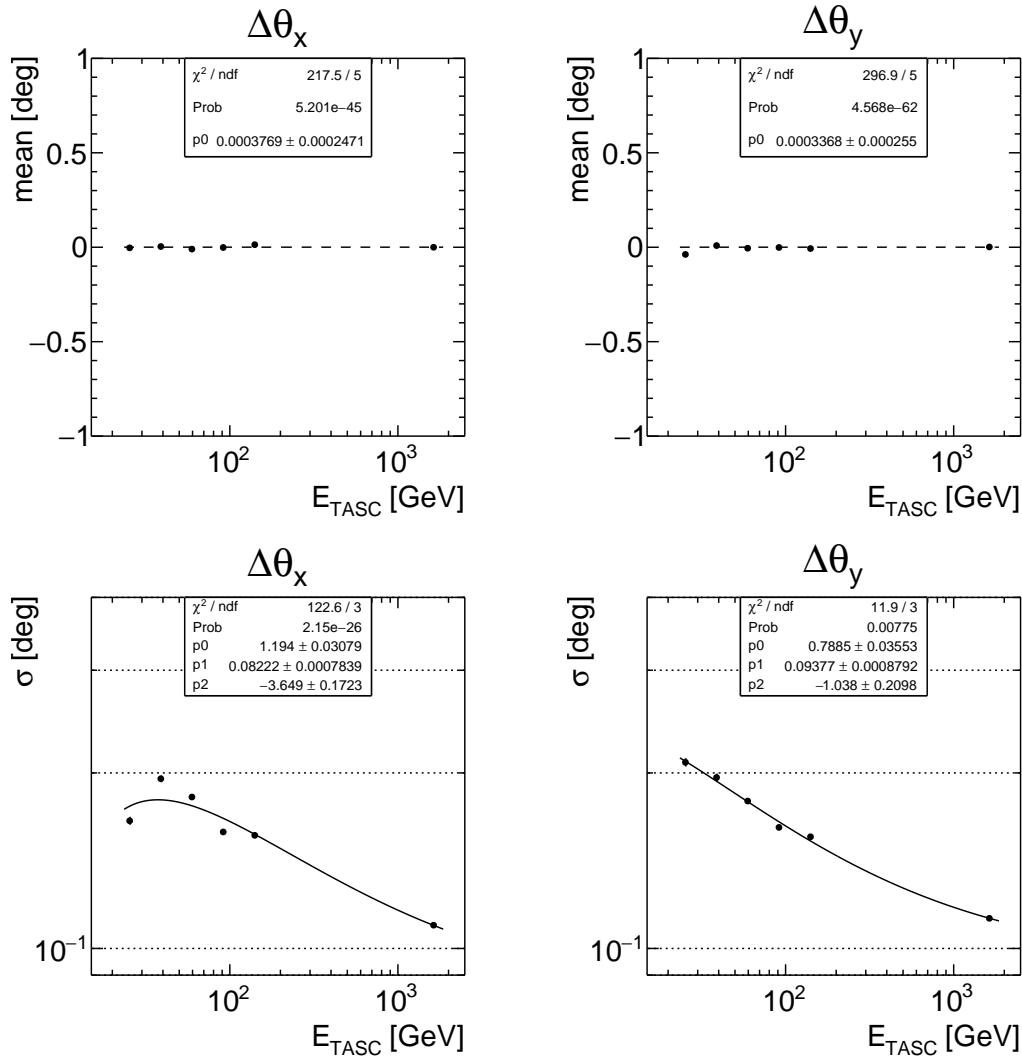


Figura B.9: nei grafici *sopra* il valore medio della distribuzione dei residui; *sotto* la risoluzione angolare della distribuzione dei residui delle variabili angolari del Kalman filter (in nero) sui dati simulati MC di elio, in funzione dell'energia media depositata in ciascun range.

## B.2 Campione di nuclei di He simulati MC

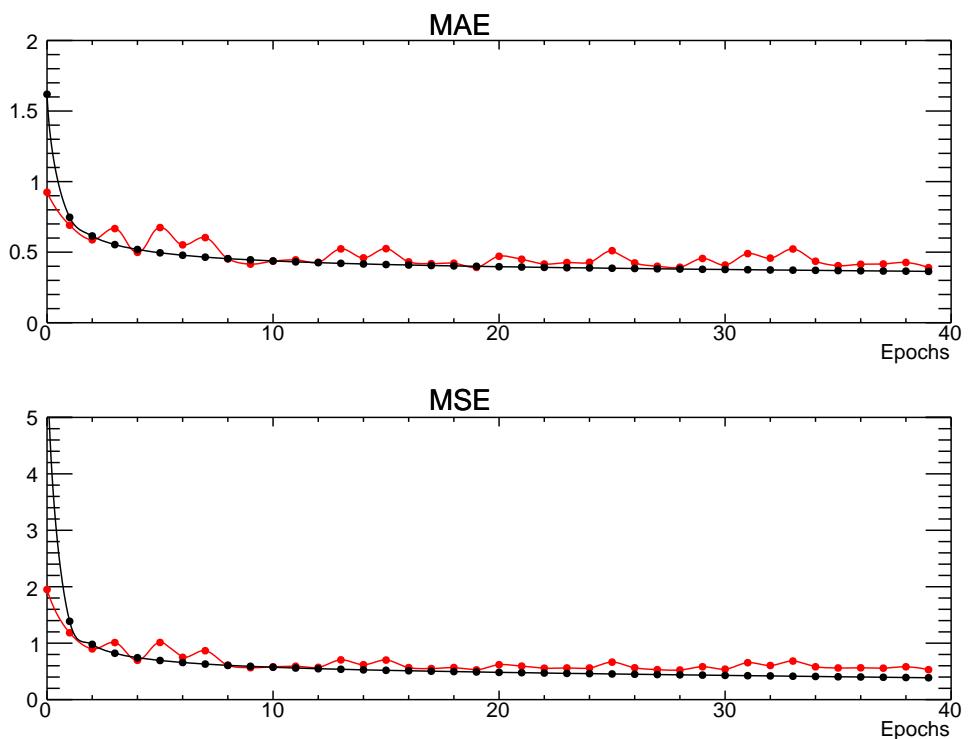


Figura B.10: grafici del *mean absolute error* (mae) e *mean square error* (mse) relative alla CNN allenata sui nuclei di elio. In nero la curva del training, in rosso quella della validazione.

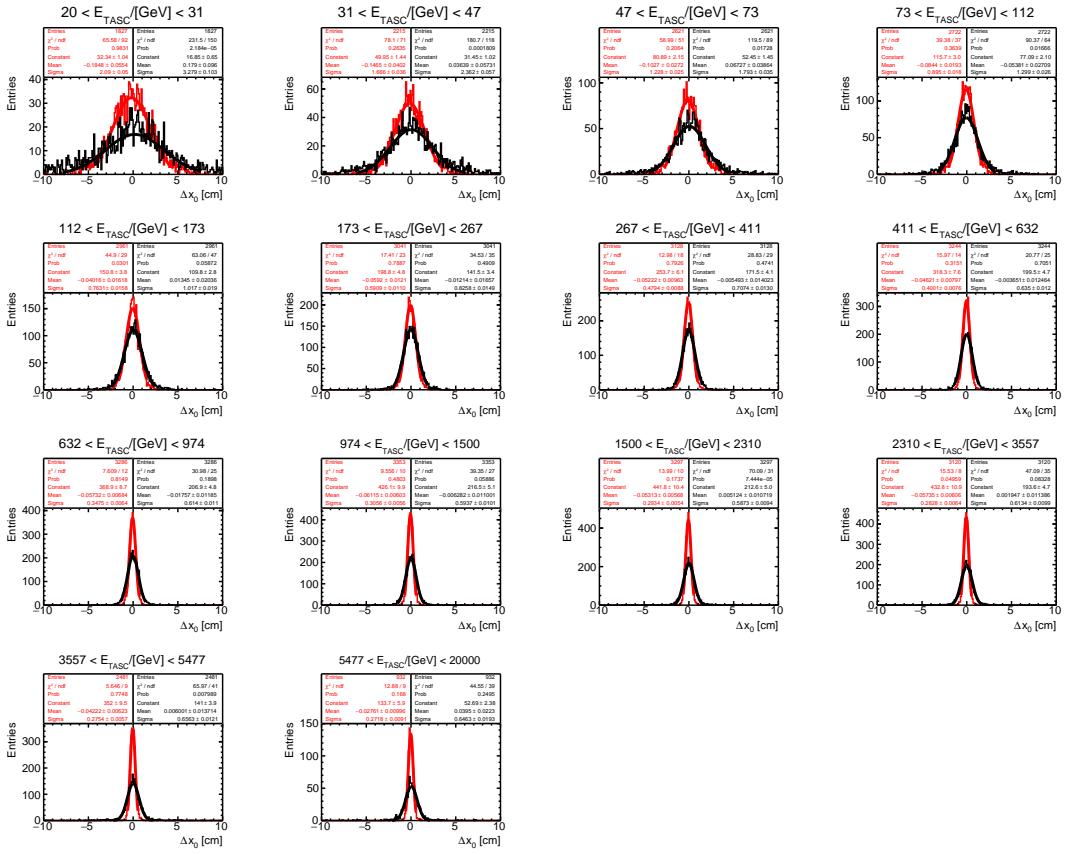


Figura B.11: distribuzione dei residui della variabile di impact point  $x_0$  predetta dalla CNN allenata sui nuclei di elio (istogramma in rosso) e dal metodo dell'asse dello sciame (istogramma in nero). Le curve sovrapposte agli istogrammi costituiscono i fit gaussiani agli istogrammi del medesimo colore.

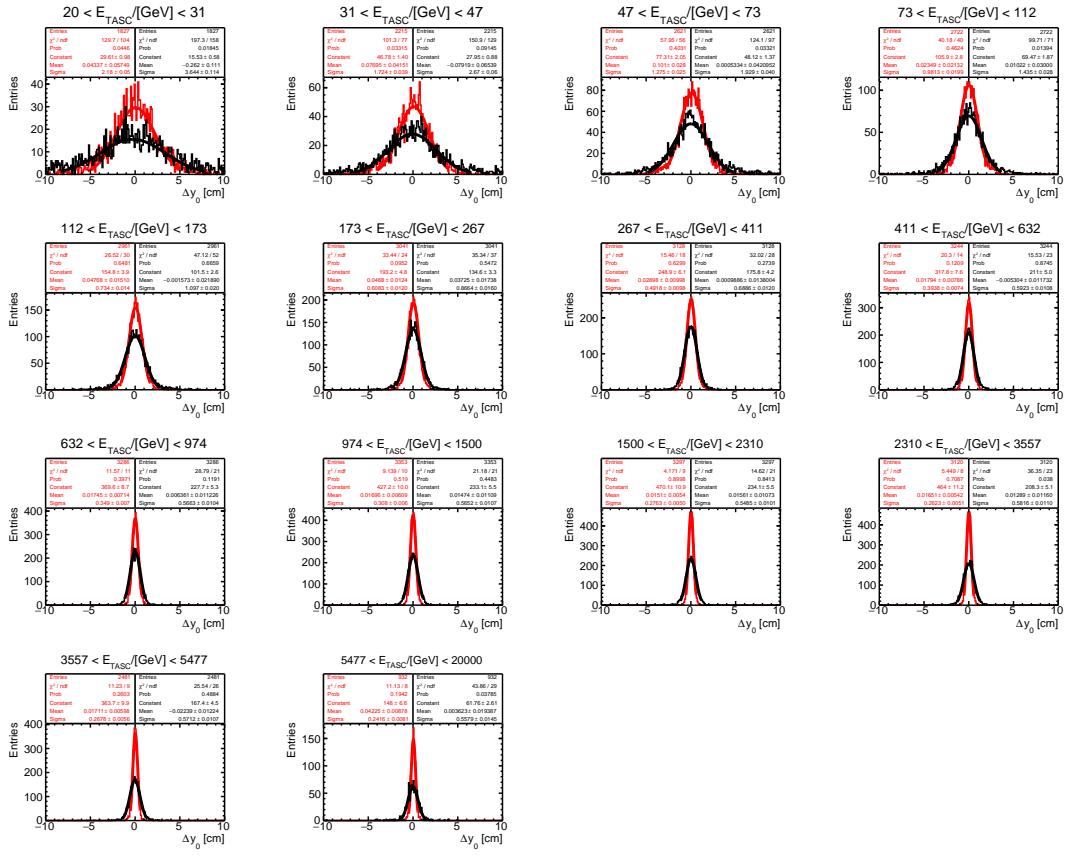


Figura B.12: distribuzione dei residui della variabile di impact point  $y_0$  predetta dalla CNN allenata sui nuclei di elio (istogramma in rosso) e dal metodo dell'asse dello sciame (istogramma in nero). Le curve sovrapposte agli istogrammi costituiscono i fit gaussiani agli istogrammi del medesimo colore.

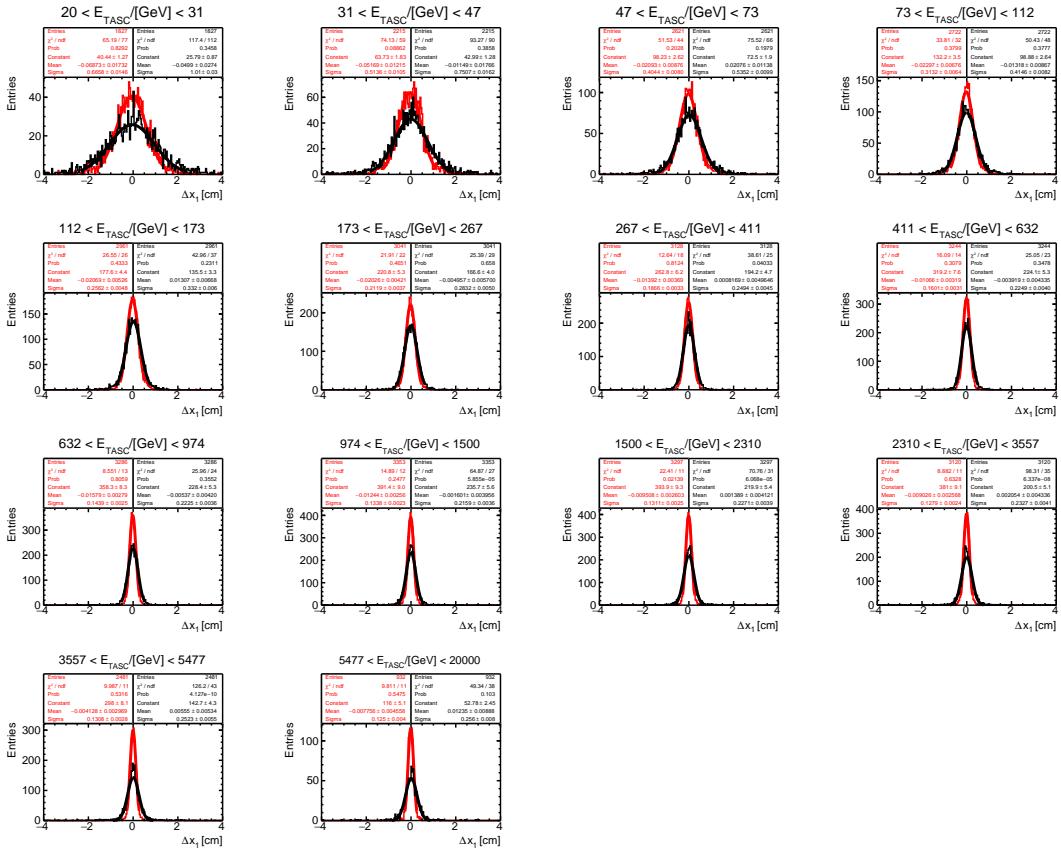


Figura B.13: distribuzione dei residui della variabile di impact point  $x_1$  predetta dalla CNN allenata sui nuclei di elio (istogramma in rosso) e dal metodo dell'asse dello sciame (istogramma in nero). Le curve sovrapposte agli istogrammi costituiscono i fit gaussiani agli istogrammi del medesimo colore.

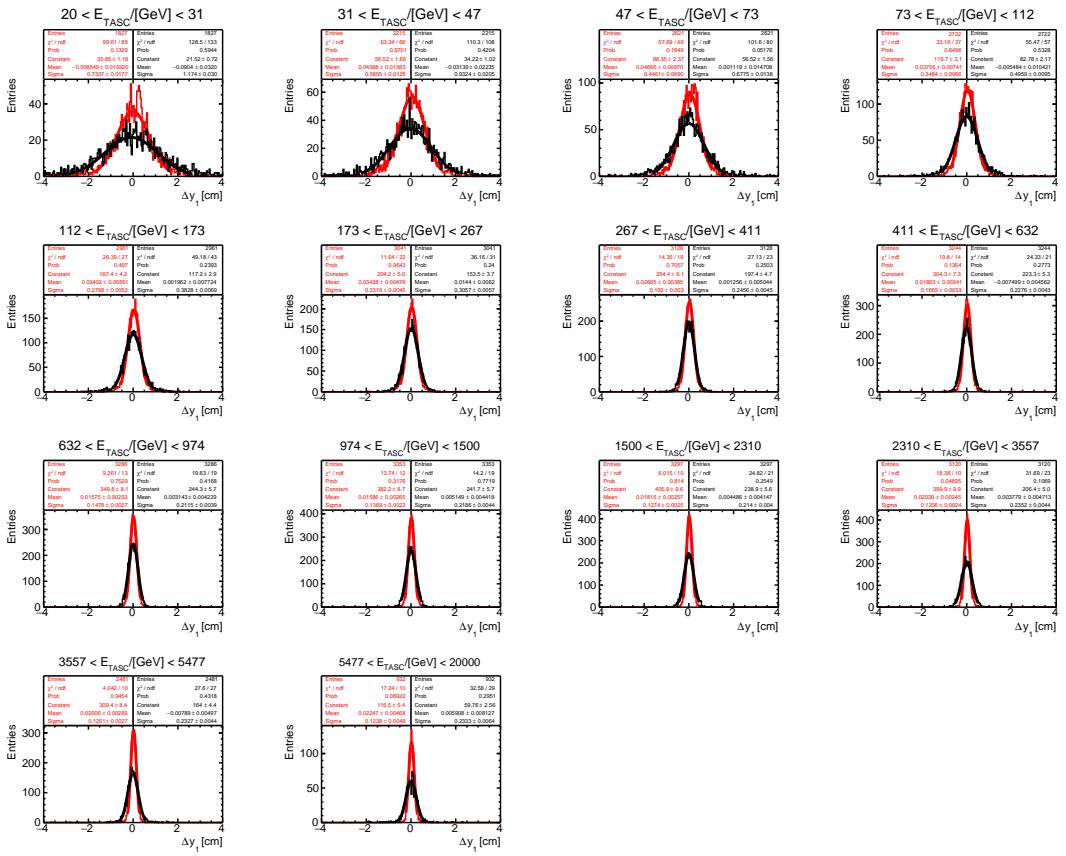


Figura B.14: distribuzione dei residui della variabile di impact point  $y_1$  pre detta dalla CNN allenata sui nuclei di elio (istogramma in rosso) e dal metodo dell'asse dello sciame (istogramma in nero). Le curve sovrapposte agli istogrammi costituiscono i fit gaussiani agli istogrammi del medesimo colore.

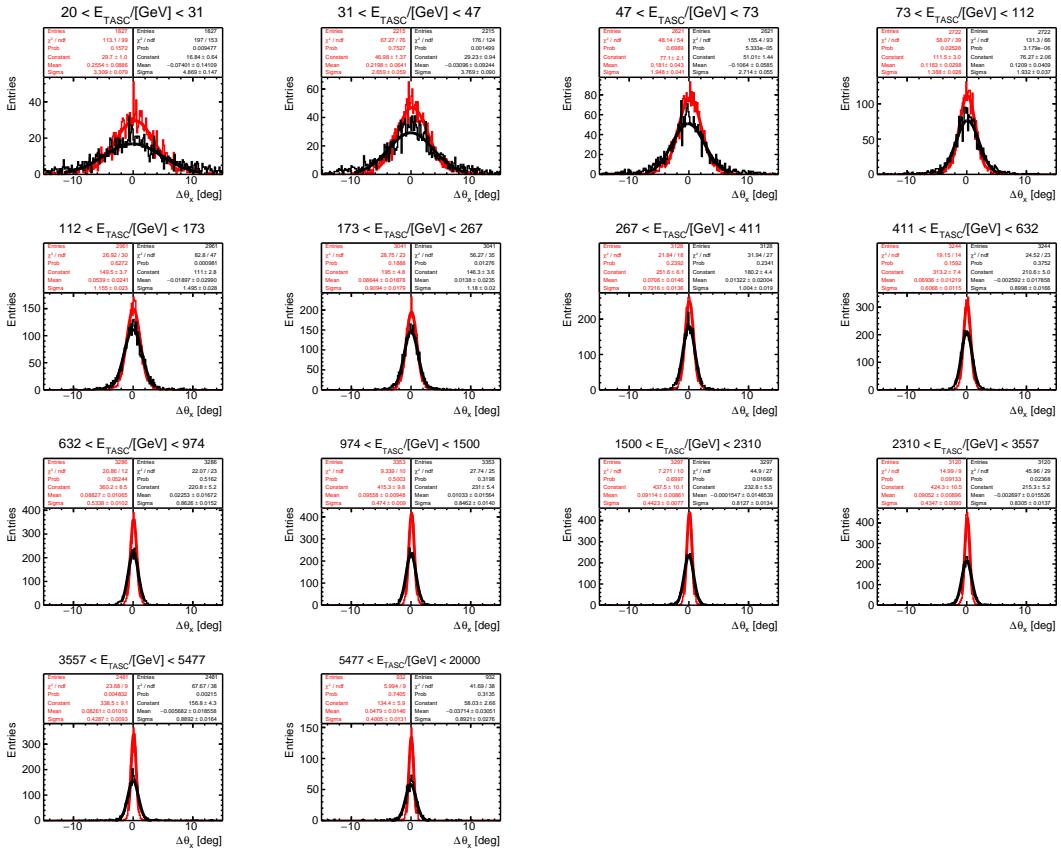


Figura B.15: distribuzione dei residui della variabile angolare  $\theta_x$  predetta dalla CNN allenata sui nuclei di elio (istogramma in rosso) e dal metodo dell'asse dello sciamo (istogramma in nero). Le curve sovrapposte agli istogrammi costituiscono i fit gaussiani agli istogrammi del medesimo colore.

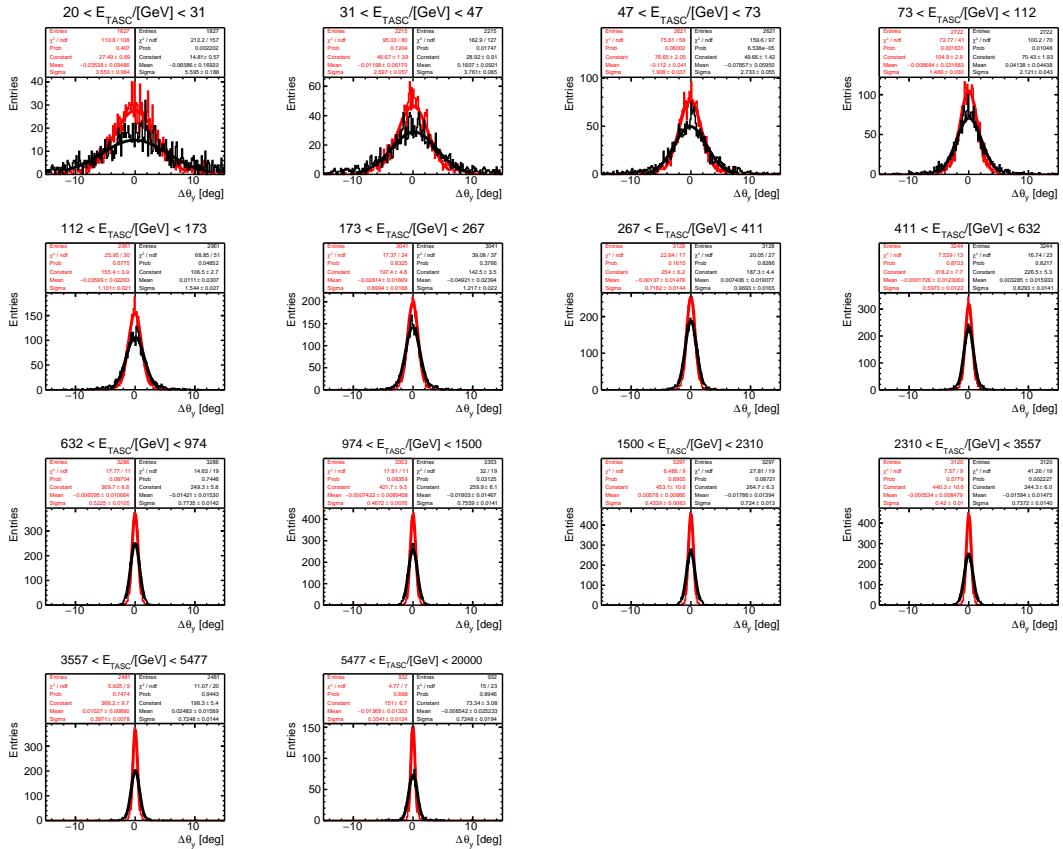


Figura B.16: distribuzione dei residui della variabile angolare  $\theta_y$  predetta dalla CNN allenata sui nuclei di elio (istogramma in rosso) e dal metodo dell’asse dello sciame (istogramma in nero). Le curve sovrapposte agli istogrammi costituiscono i fit gaussiani agli istogrammi del medesimo colore.

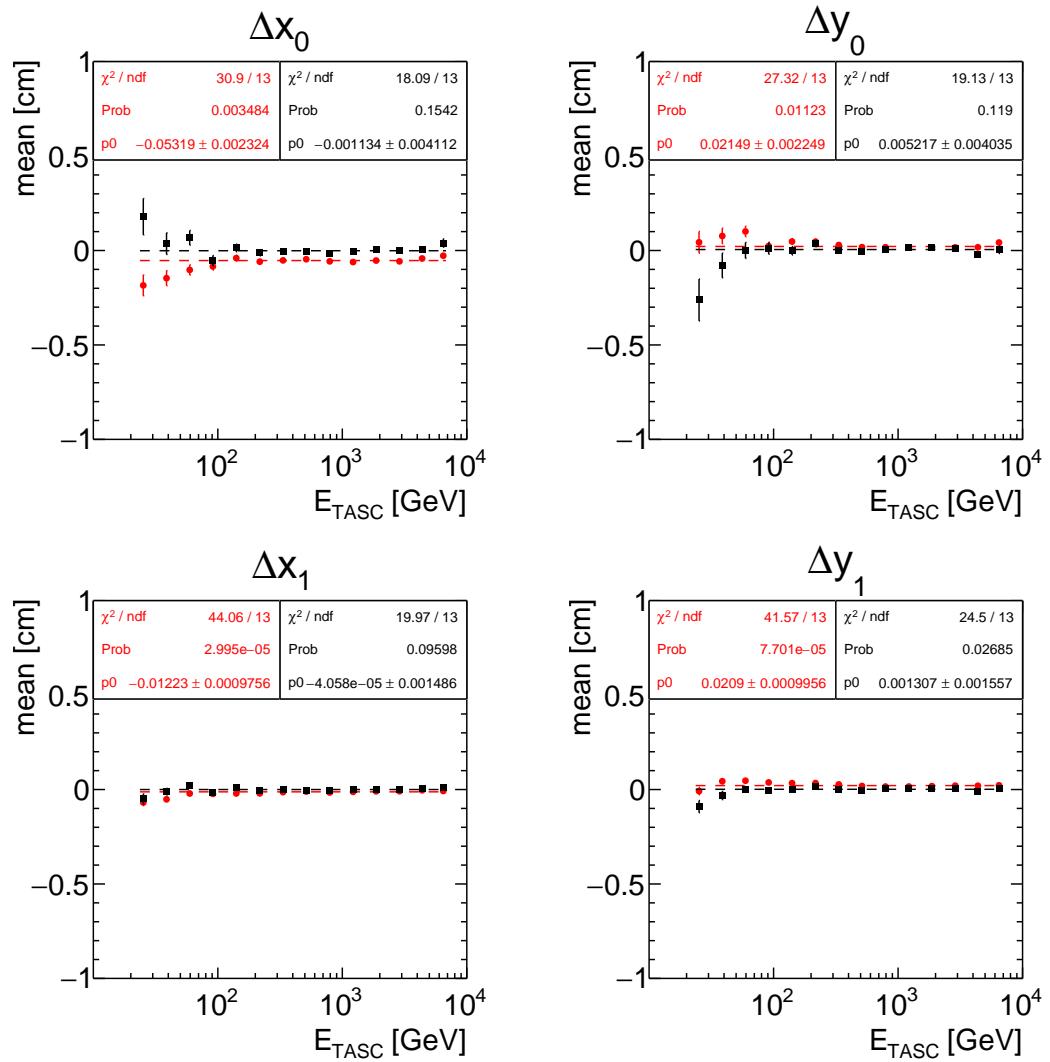


Figura B.17: valore medio della distribuzione dei residui dei punti di impatto della CNN allenata sui nuclei di elio (in rosso) e del metodo dell'asse dello sciame (in nero), in funzione dell'energia media depositata in ciascun range.

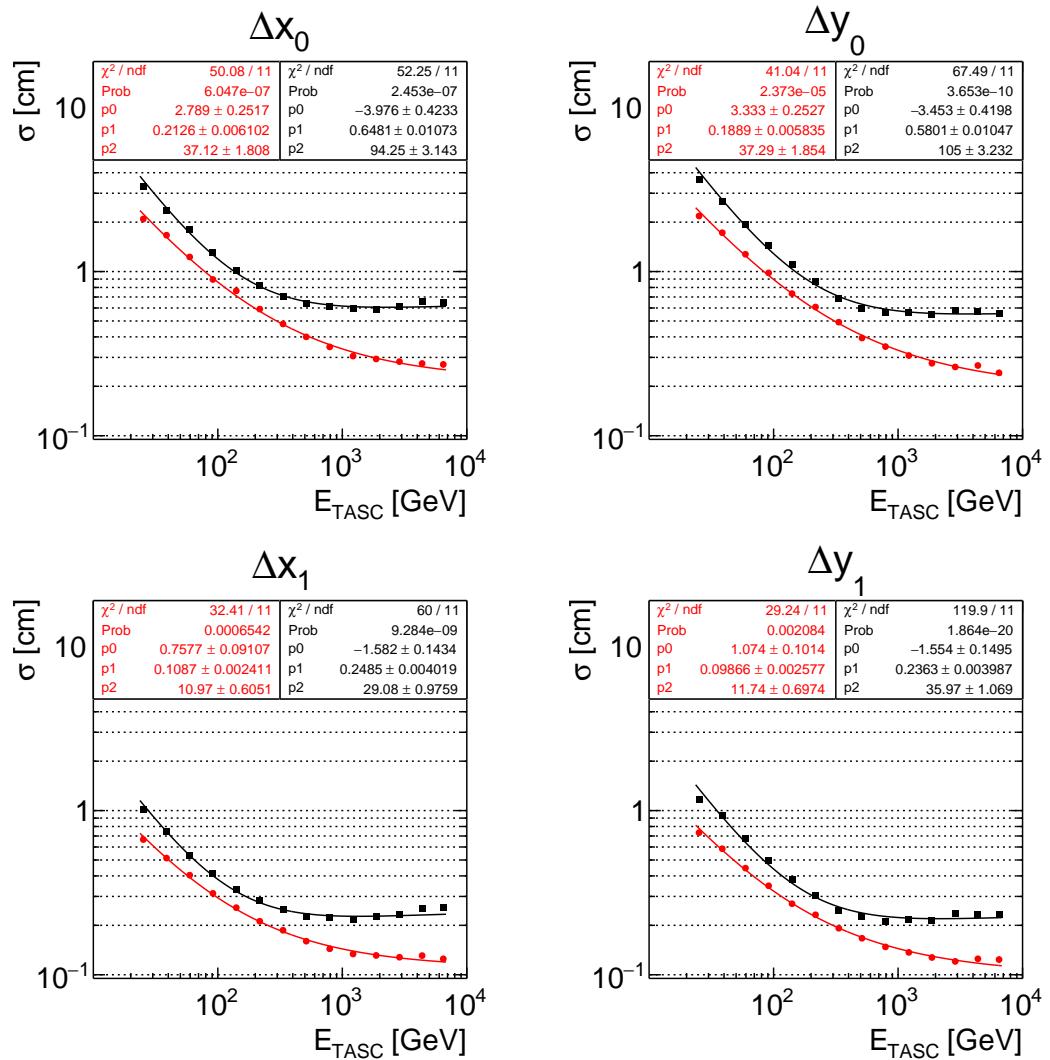


Figura B.18: risoluzione spaziale della CNN allenata sui nuclei di elio (in rosso) e del metodo dell’asse dello sciame (in nero), in funzione dell’energia media depositata in ciascun range.

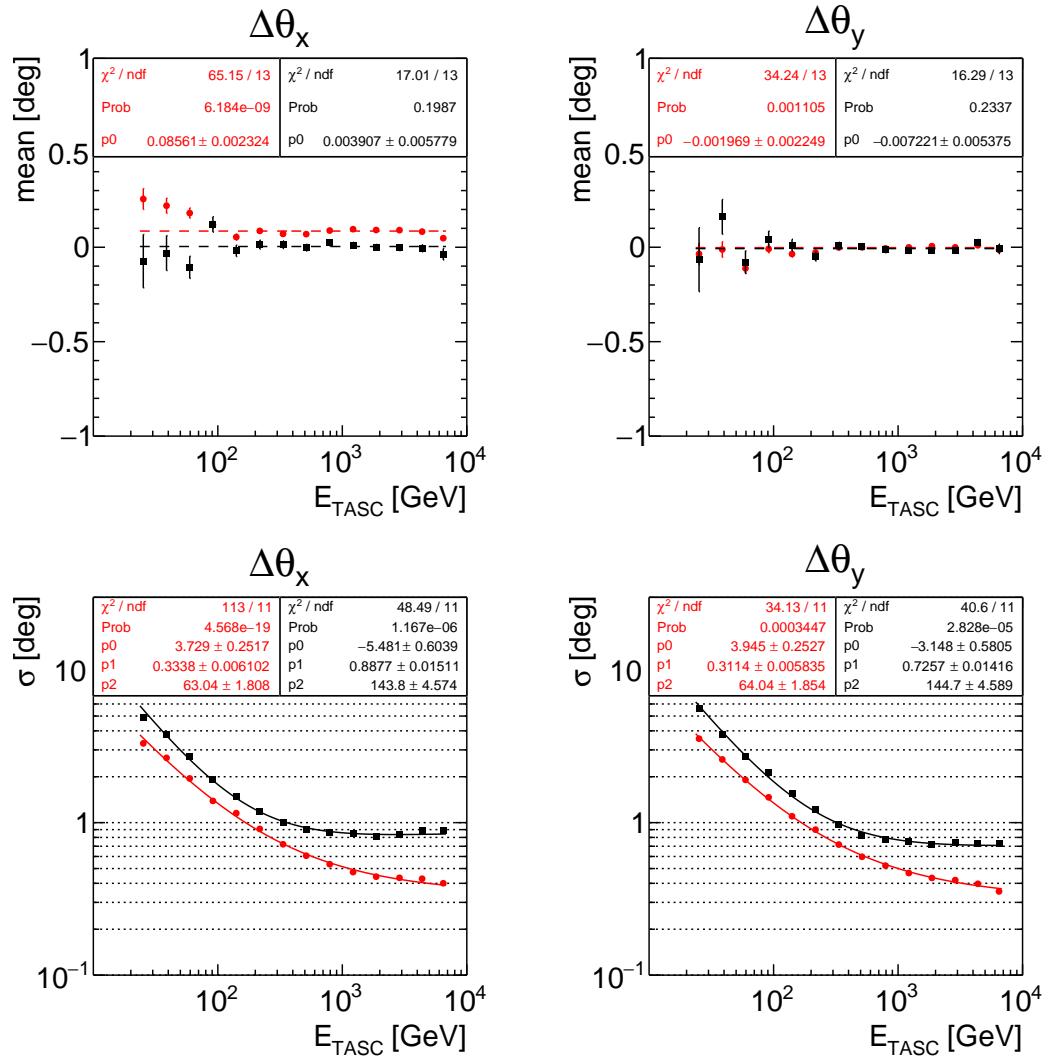


Figura B.19: nei grafici *sopra* il valore medio della distribuzione dei residui; *sotto* la risoluzione angolare della distribuzione dei residui delle variabili angolari della CNN allenata sui nuclei di elio (in *rosso*) e del metodo dell'asse dello sciame (in *nero*), in funzione dell'energia media depositata in ciascun range.

### B.3 Campione di p e nuclei di He simulati MC

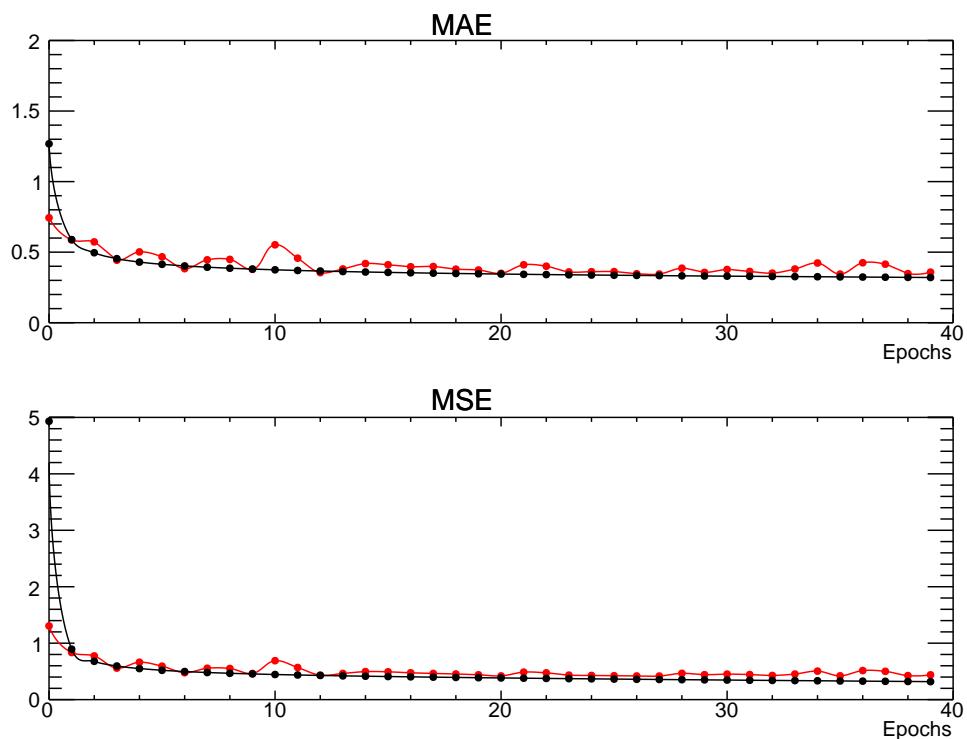


Figura B.20: grafici del *mean absolute error* (mae) e *mean square error* (mse) relative alla CNN allenata su dati combinati di p e He. In *nero* la curva del training, in *rosso* quella della validazione

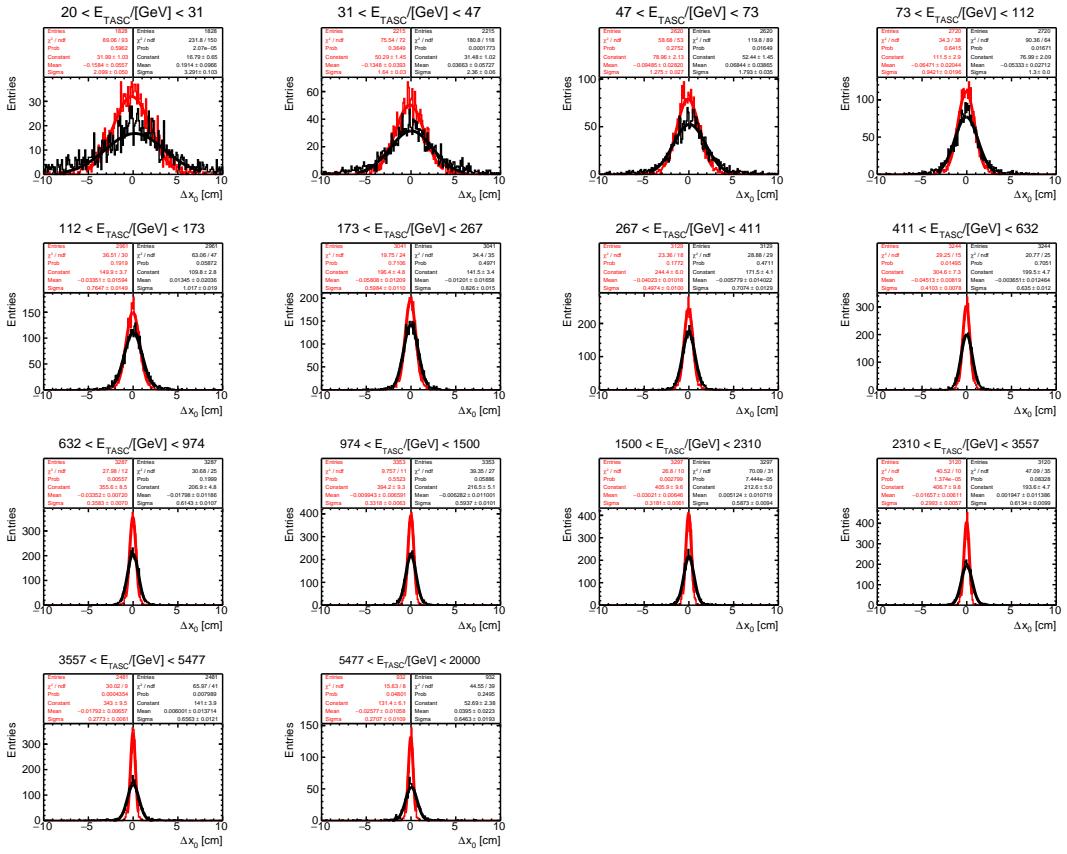


Figura B.21: distribuzione dei residui della variabile di impact point  $x_0$  predetta dalla CNN allenata su dati combinati di p e He (istogramma in rosso) e dal metodo dell'asse dello sciame (istogramma in nero). Le curve sovrapposte agli istogrammi costituiscono i fit gaussiani agli istogrammi del medesimo colore.

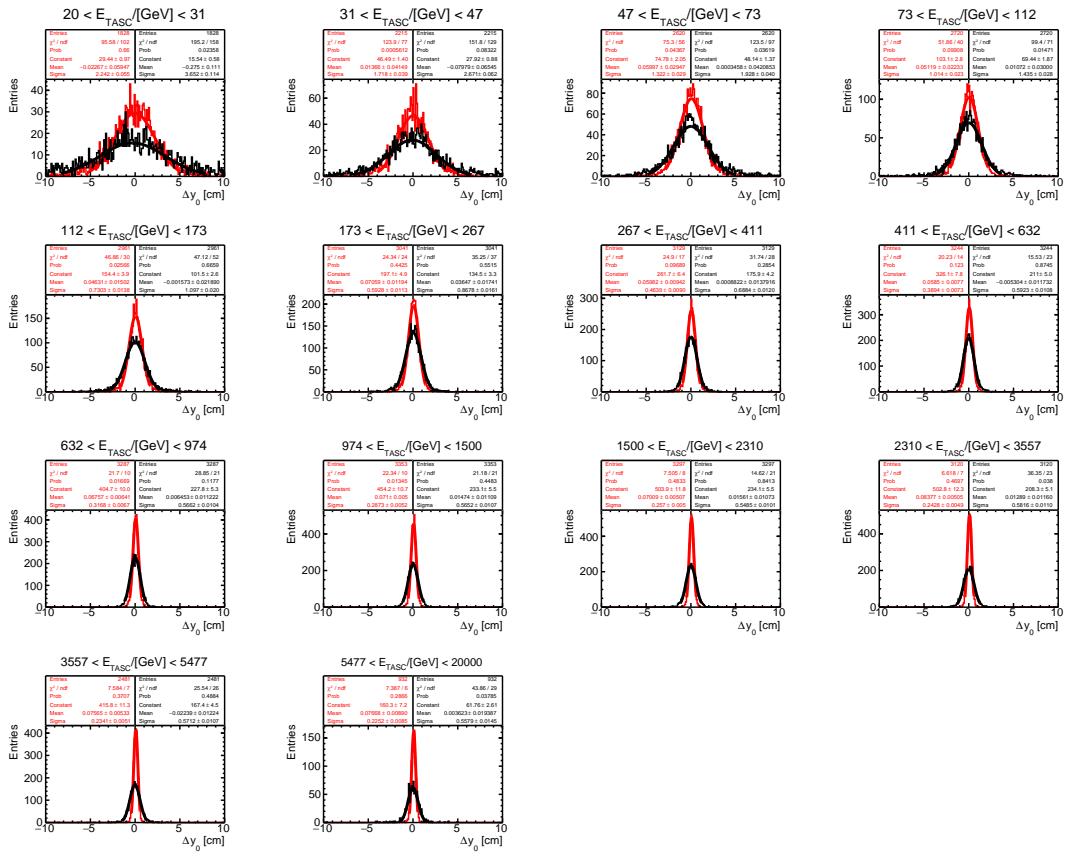


Figura B.22: distribuzione dei residui della variabile di impact point  $y_0$  pre detta dalla CNN allenata su dati combinati di p e He (istogramma in rosso) e dal metodo dell’asse dello sciame (istogramma in nero). Le curve sovrapposte agli istogrammi costituiscono i fit gaussiani agli istogrammi del medesimo colore.

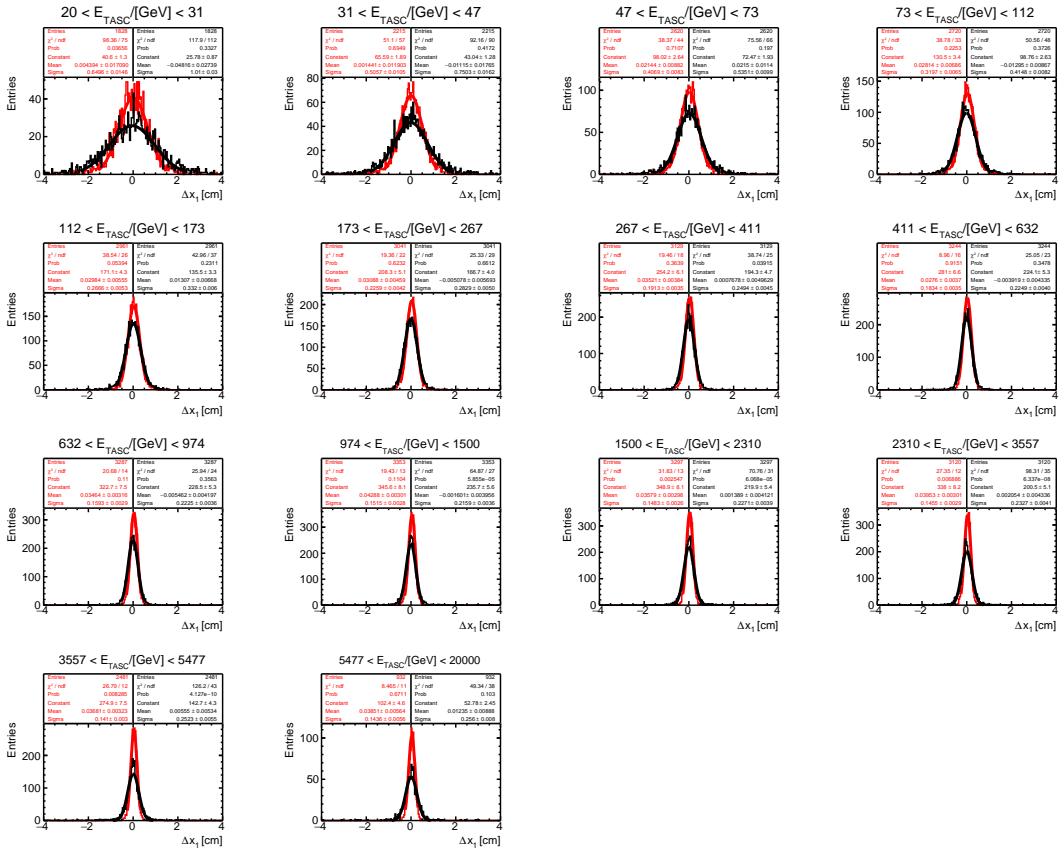


Figura B.23: distribuzione dei residui della variabile di impact point  $x_1$  pre detta dalla CNN allenata su dati combinati di p e He (istogramma in rosso) e dal metodo dell’asse dello sciame (istogramma in nero). Le curve sovrapposte agli istogrammi costituiscono i fit gaussiani agli istogrammi del medesimo colore.

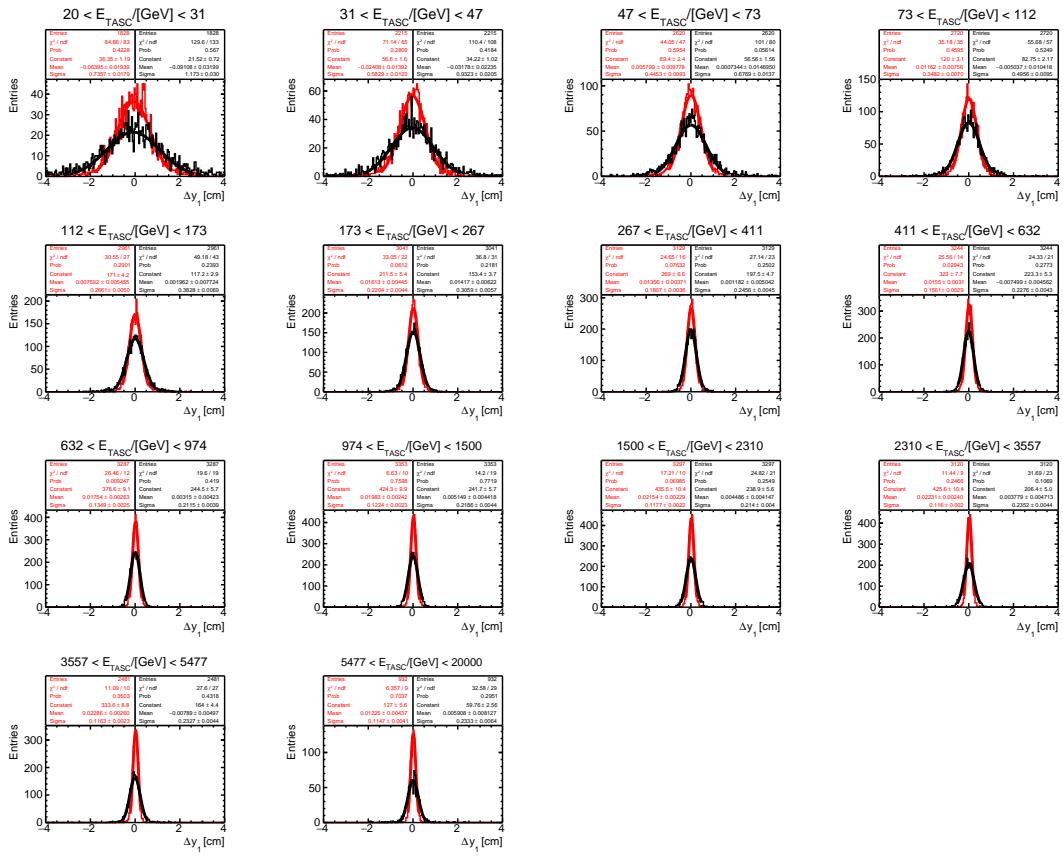


Figura B.24: distribuzione dei residui della variabile di impact point  $y_1$  pre detta dalla CNN allenata su dati combinati di p e He (istogramma in rosso) e dal metodo dell’asse dello sciame (istogramma in nero). Le curve sovrapposte agli istogrammi costituiscono i fit gaussiani agli istogrammi del medesimo colore.

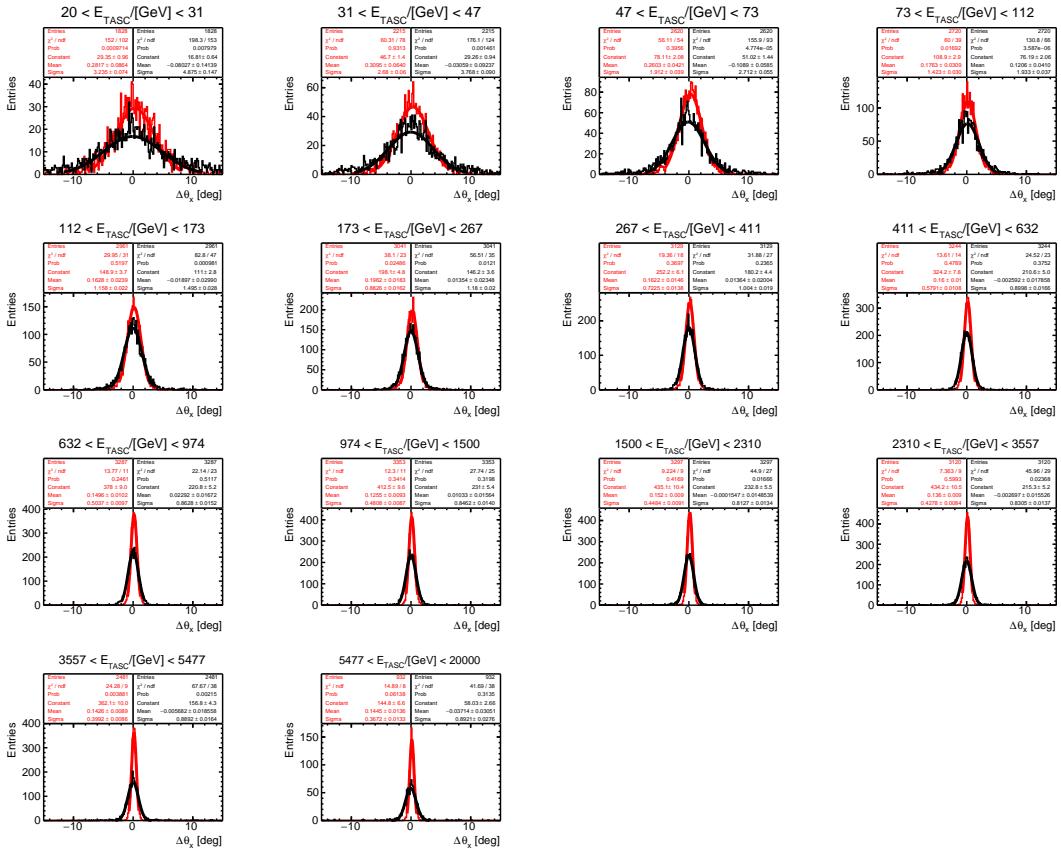


Figura B.25: distribuzione dei residui della variabile angolare  $\theta_x$  predetta dalla CNN allenata su dati combinati di p e He (istogramma in rosso) e dal metodo dell’asse dello sciame (istogramma in nero). Le curve sovrapposte agli istogrammi costituiscono i fit gaussiani agli istogrammi del medesimo colore.

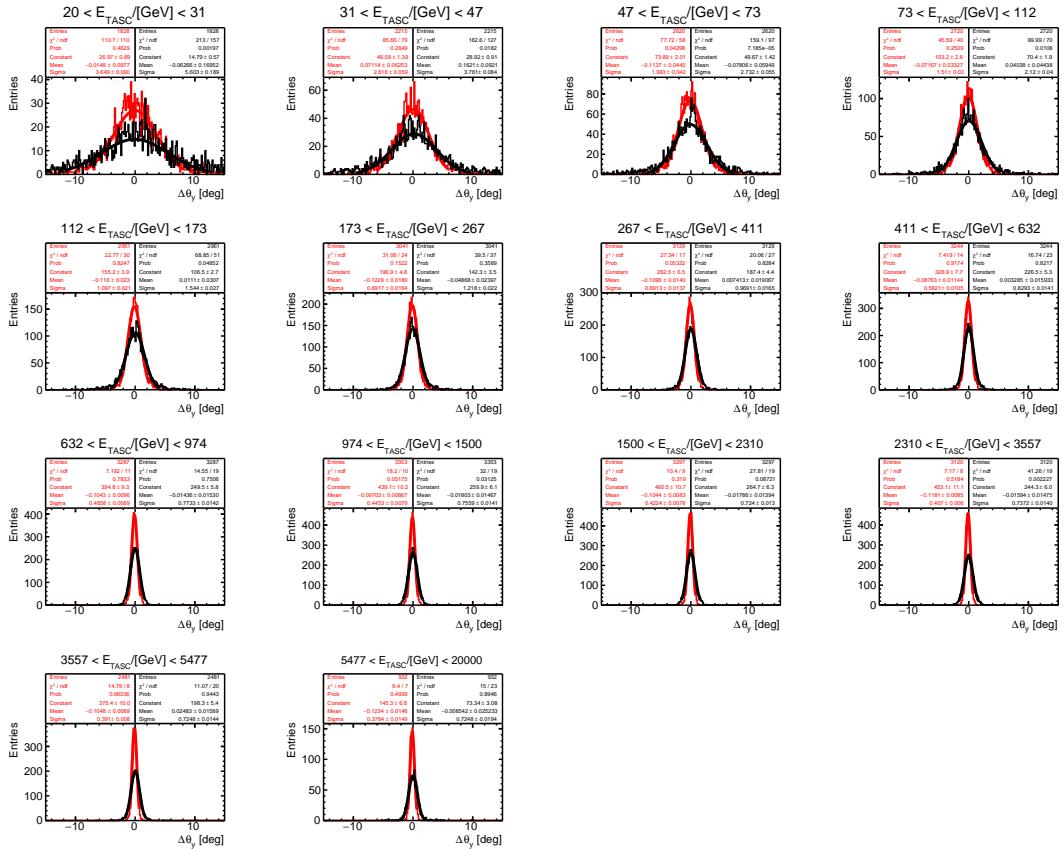


Figura B.26: distribuzione dei residui della variabile angolare  $\theta_y$  predetta dalla CNN allenata su dati combinati di p e He (istogramma in rosso) e dal metodo dell’asse dello sciame (istogramma in nero). Le curve sovrapposte agli istogrammi costituiscono i fit gaussiani agli istogrammi del medesimo colore.

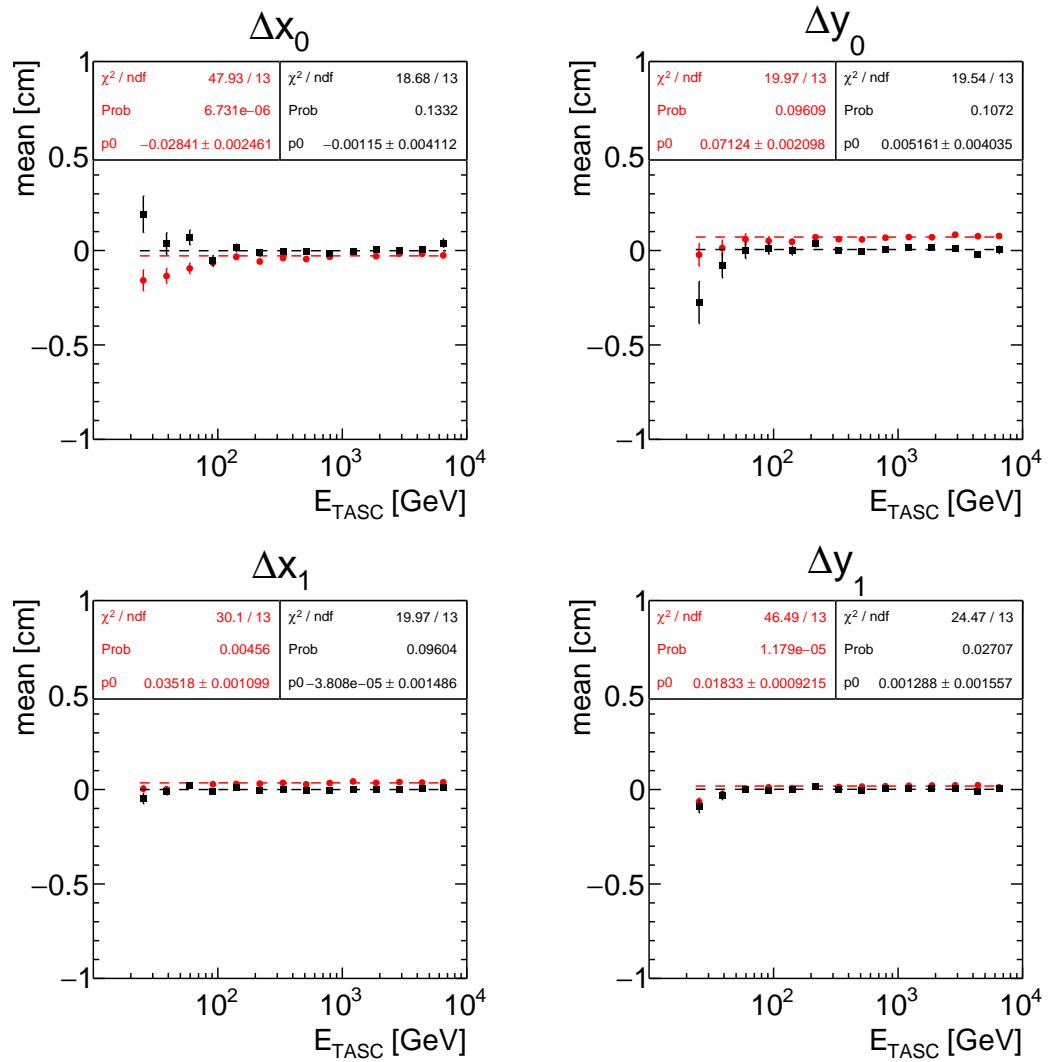


Figura B.27: risoluzione spaziale della distribuzione dei residui dei punti di impatto della CNN allenata su dati combinati di p e He (in rosso) e del metodo dell'asse dello sciame (in nero), in funzione dell'energia media depositata in ciascun range.

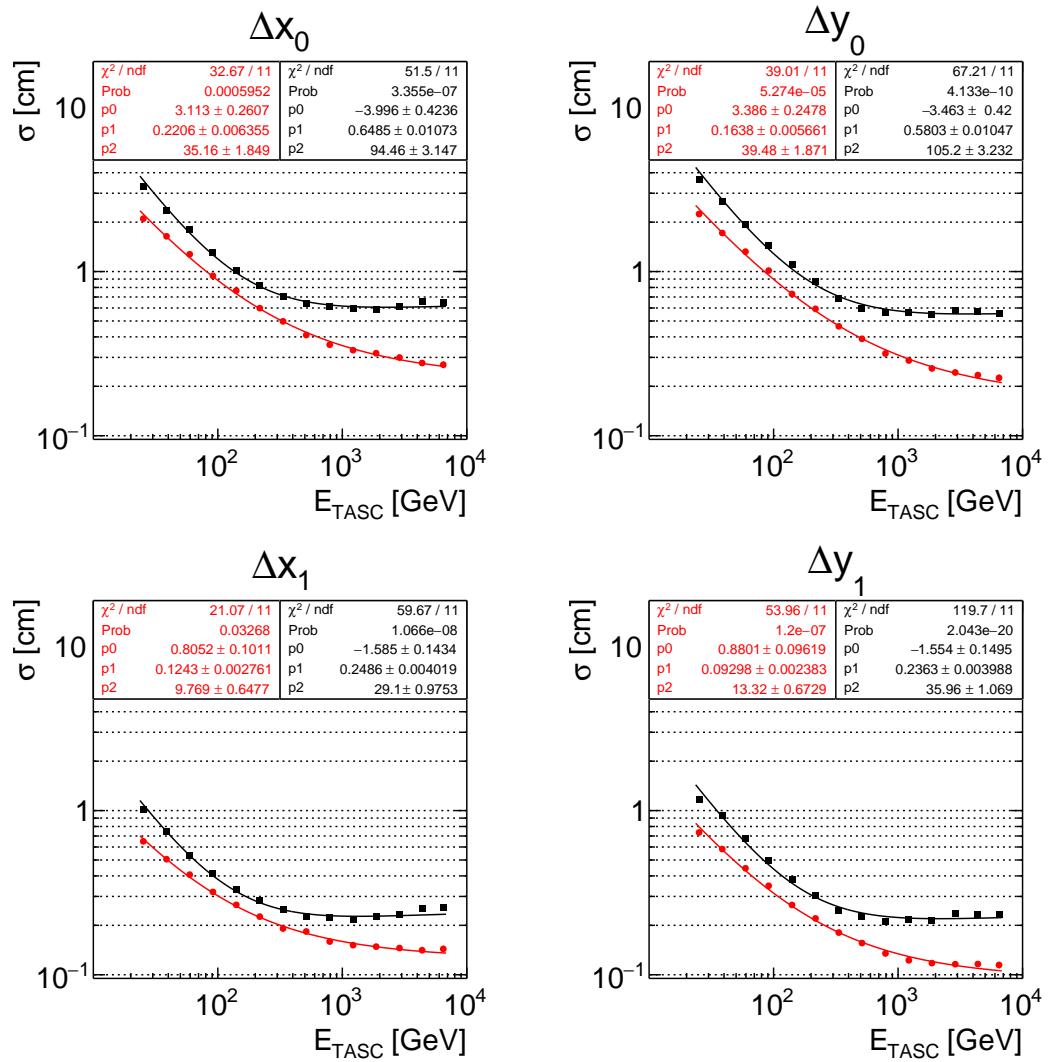


Figura B.28: risoluzione spaziale della CNN allenata su dati combinati di p e He (in rosso) e del metodo dell'asse dello sciame (in nero), in funzione dell'energia media depositata in ciascun range.

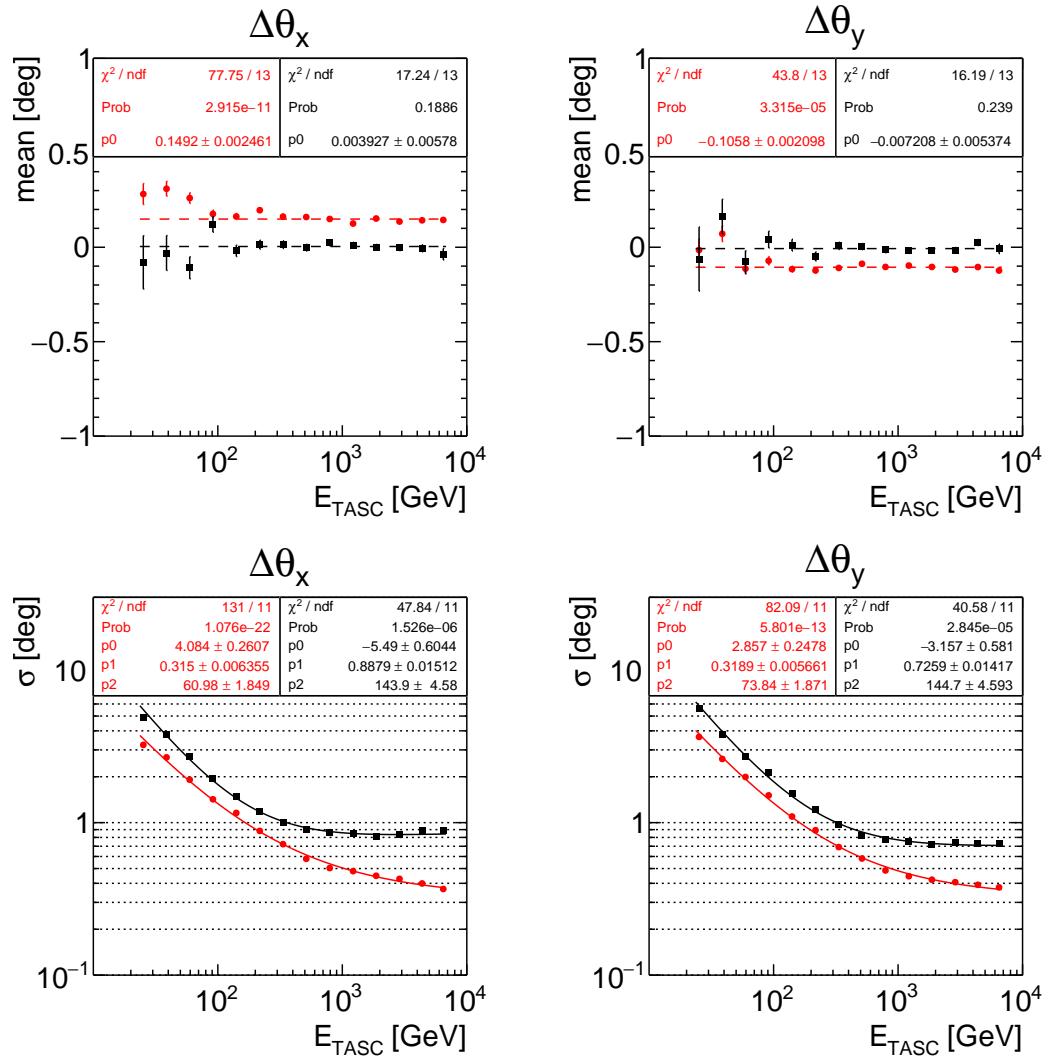


Figura B.29: nei grafici *sopra* il valore medio della distribuzione dei residui; *sotto* la risoluzione angolare della distribuzione dei residui delle variabili angolari della CNN allenata su dati nuclei combinati di p e He (in rosso) e del metodo dell'asse dello sciame (in nero), in funzione dell'energia media depositata in ciascun range.

## B.4 Campione di nuclei di He dei dati di volo

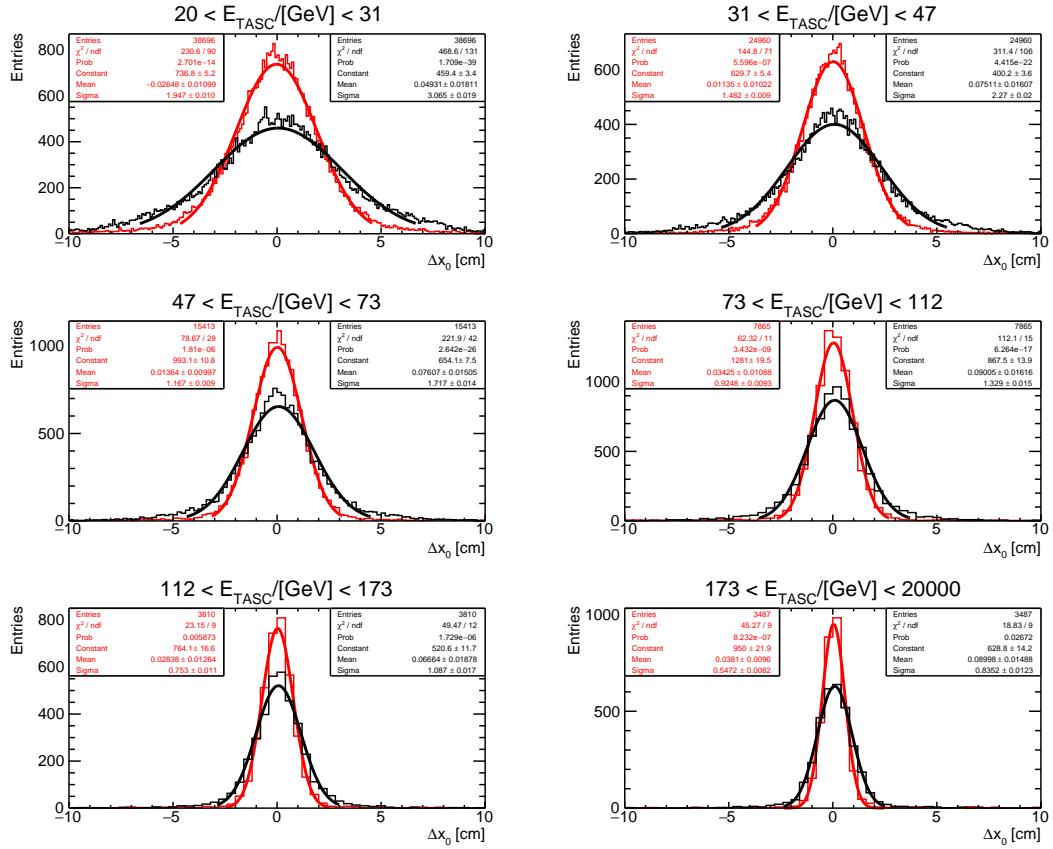


Figura B.30: distribuzione dei residui della variabile di impact point  $x_0$  predetta dalla CNN allenata sui nuclei di elio (istogramma in rosso) e dal metodo dell'asse dello sciame (istogramma in nero). Le curve sovrapposte agli istogrammi costituiscono i fit gaussiani agli istogrammi del medesimo colore.

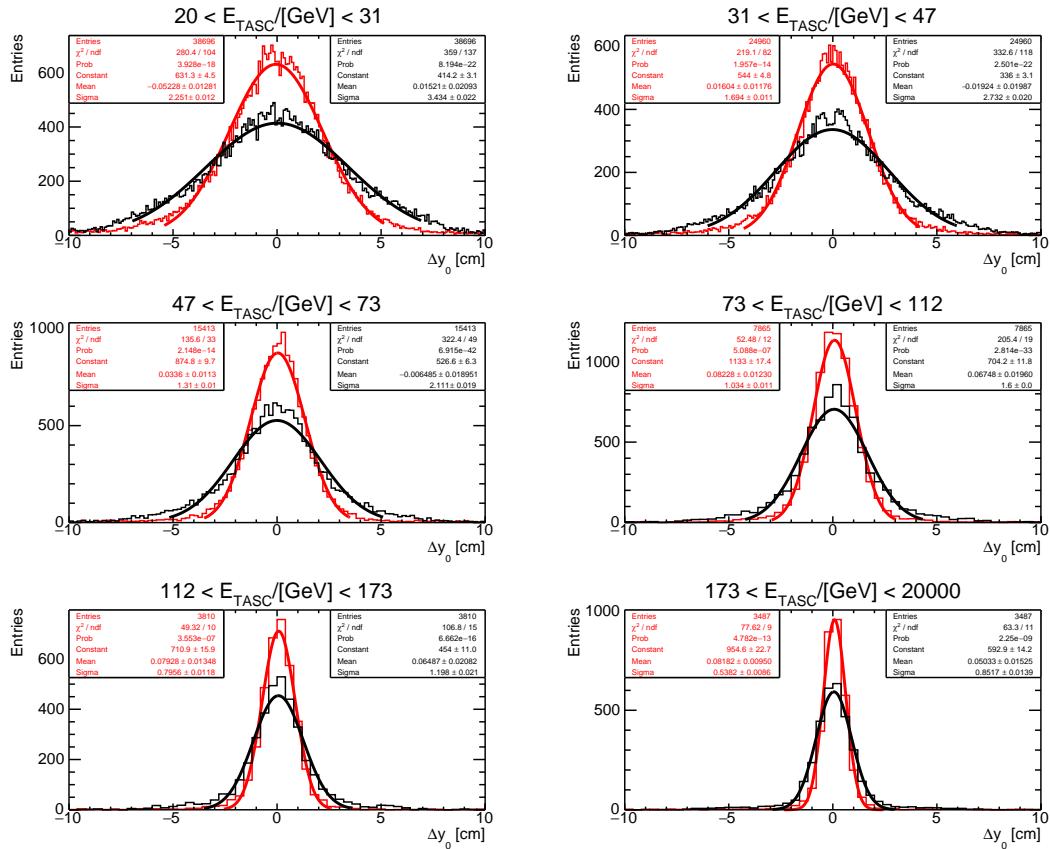


Figura B.31: distribuzione dei residui della variabile di impact point  $y_0$  predetta dalla CNN allenata sui nuclei di elio (istogramma in rosso) e dal metodo dell'asse dello sciame (istogramma in nero). Le curve sovrapposte agli istogrammi costituiscono i fit gaussiani agli istogrammi del medesimo colore.

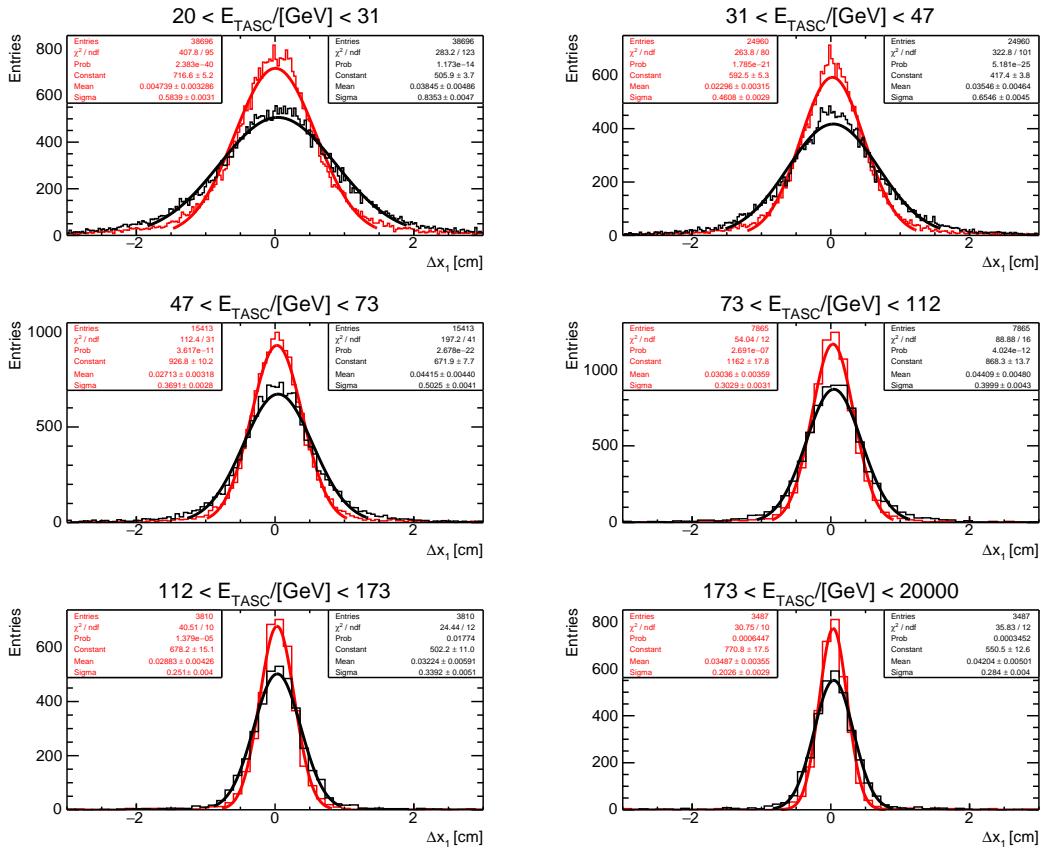


Figura B.32: distribuzione dei residui della variabile di impact point  $x_1$  predetta dalla CNN allenata sui nuclei di elio (istogramma in rosso) e dal metodo dell'asse dello sciame (istogramma in nero). Le curve sovrapposte agli istogrammi costituiscono i fit gaussiani agli istogrammi del medesimo colore.

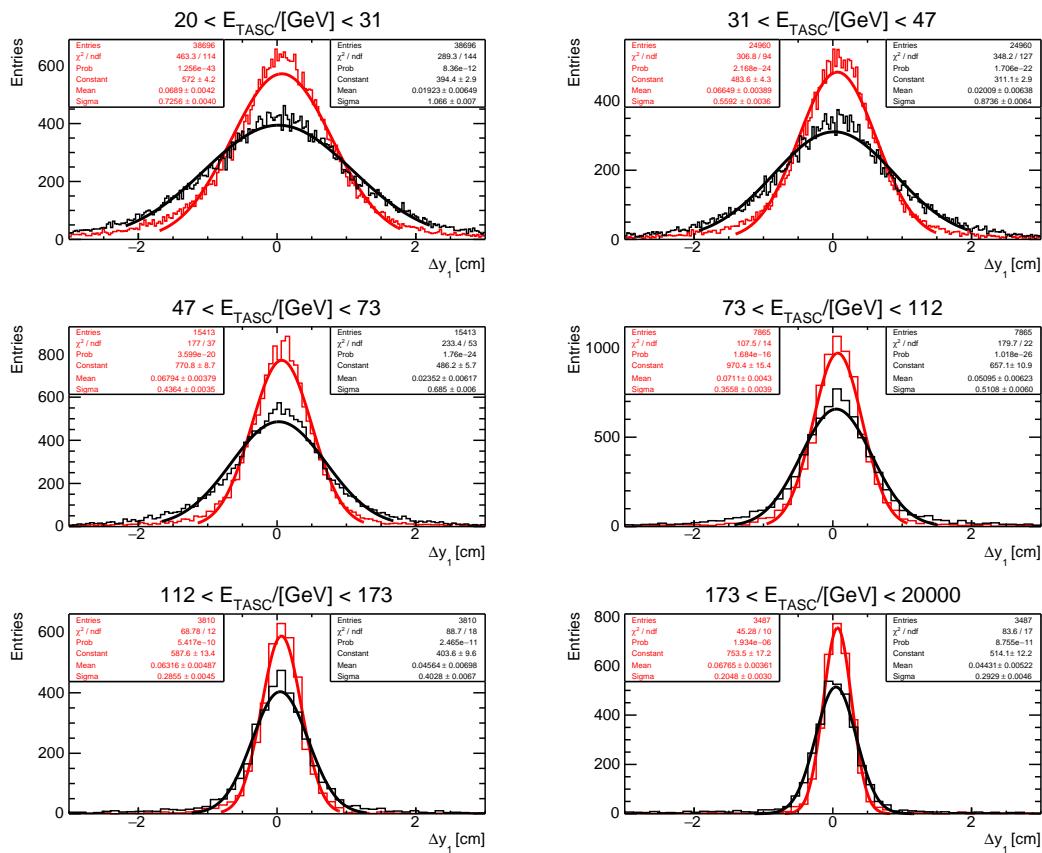


Figura B.33: distribuzione dei residui della variabile di impact point  $y_1$  predetta dalla CNN allenata sui nuclei di elio (istogramma in rosso) e dal metodo dell’asse dello sciame (istogramma in nero). Le curve sovrapposte agli istogrammi costituiscono i fit gaussiani agli istogrammi del medesimo colore.

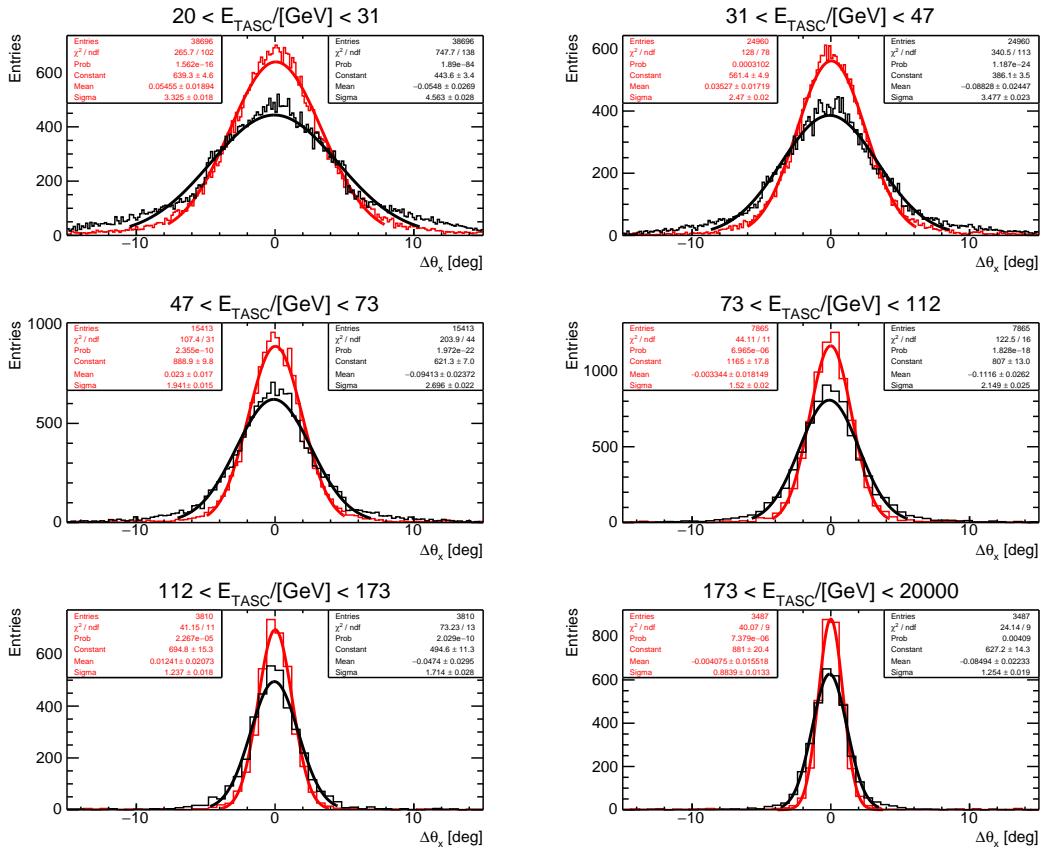


Figura B.34: distribuzione dei residui della variabile angolare  $\theta_x$  predetta dalla CNN allenata sui nuclei di elio (istogramma in rosso) e dal metodo dell’asse dello sciame (istogramma in nero). Le curve sovrapposte agli istogrammi costituiscono i fit gaussiani agli istogrammi del medesimo colore.

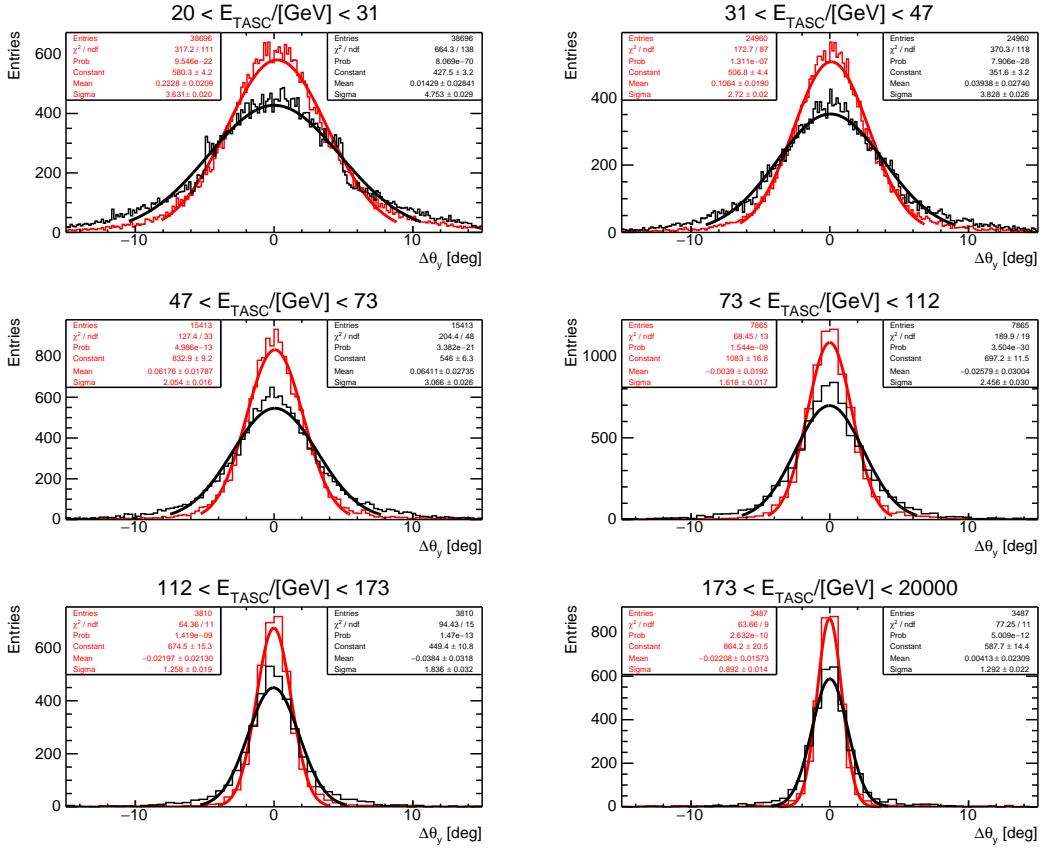


Figura B.35: distribuzione dei residui della variabile angolare  $\theta_y$  predetta dalla CNN allenata sui nuclei di elio (istogramma in rosso) e dal metodo dell'asse dello sciame (istogramma in nero). Le curve sovrapposte agli istogrammi costituiscono i fit gaussiani agli istogrammi del medesimo colore.

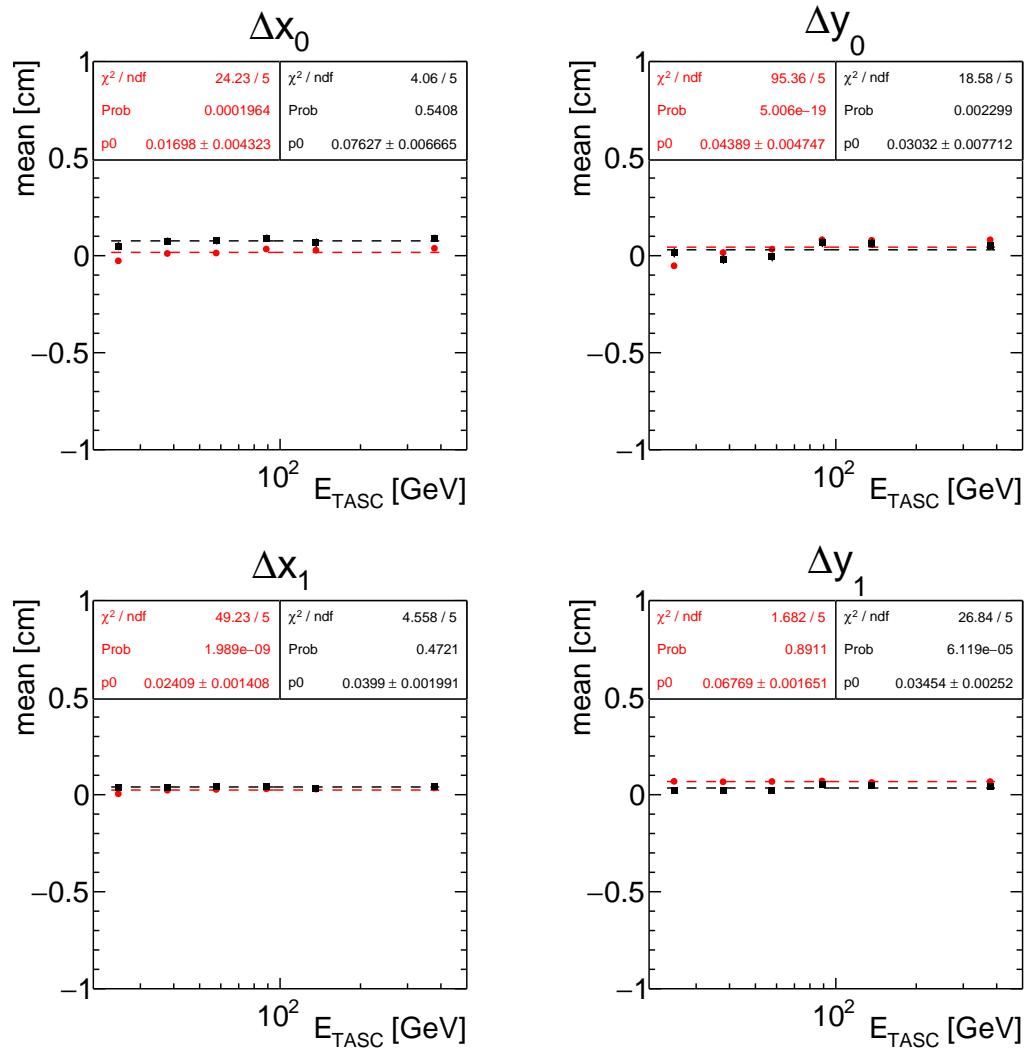


Figura B.36: valore medio della distribuzione dei residui dei punti di impatto della CNN allenata sui nuclei di elio (in rosso) e del metodo dell'asse dello sciame (in nero), in funzione dell'energia media depositata in ciascun range.

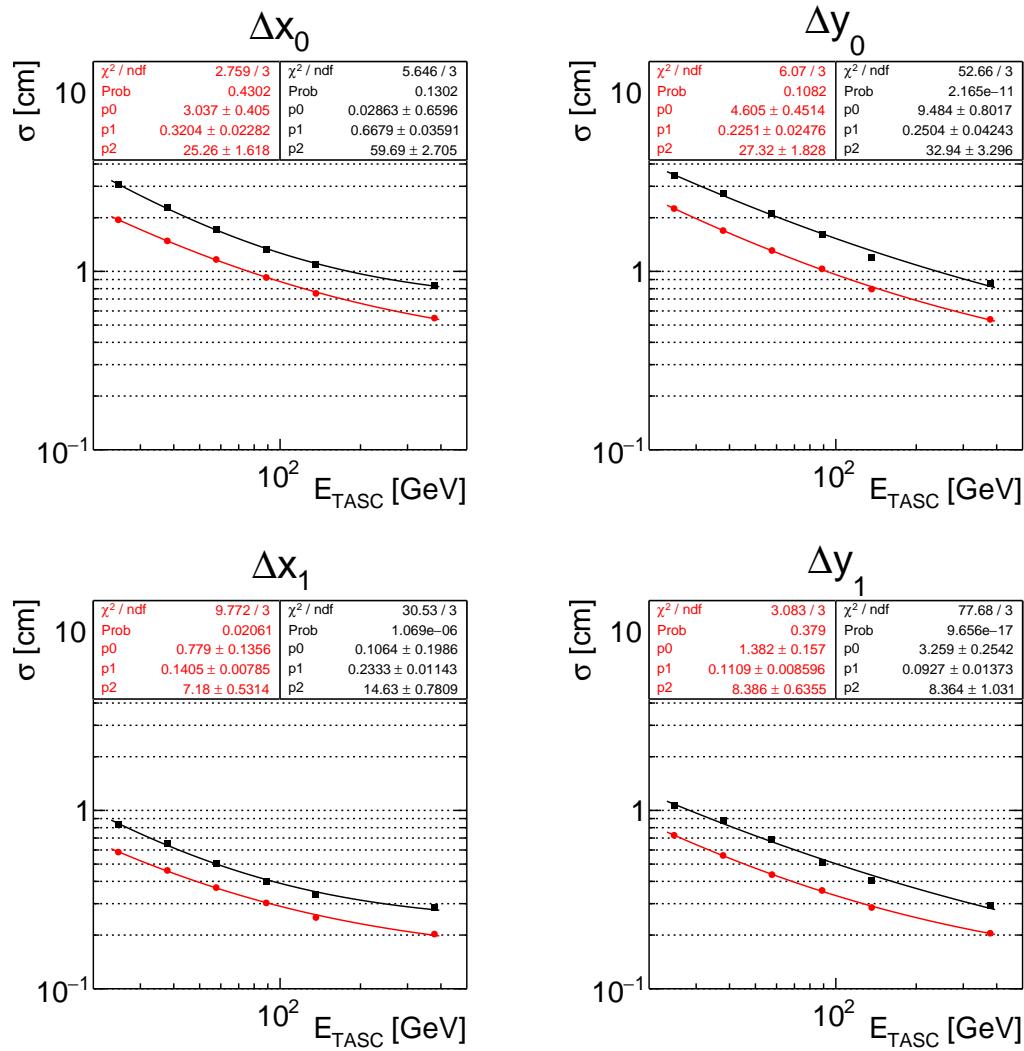


Figura B.37: risoluzione spaziale della CNN allenata sui nuclei di elio (in rosso) e del metodo dell’asse dello sciame (in nero), in funzione dell’energia media depositata in ciascun range.

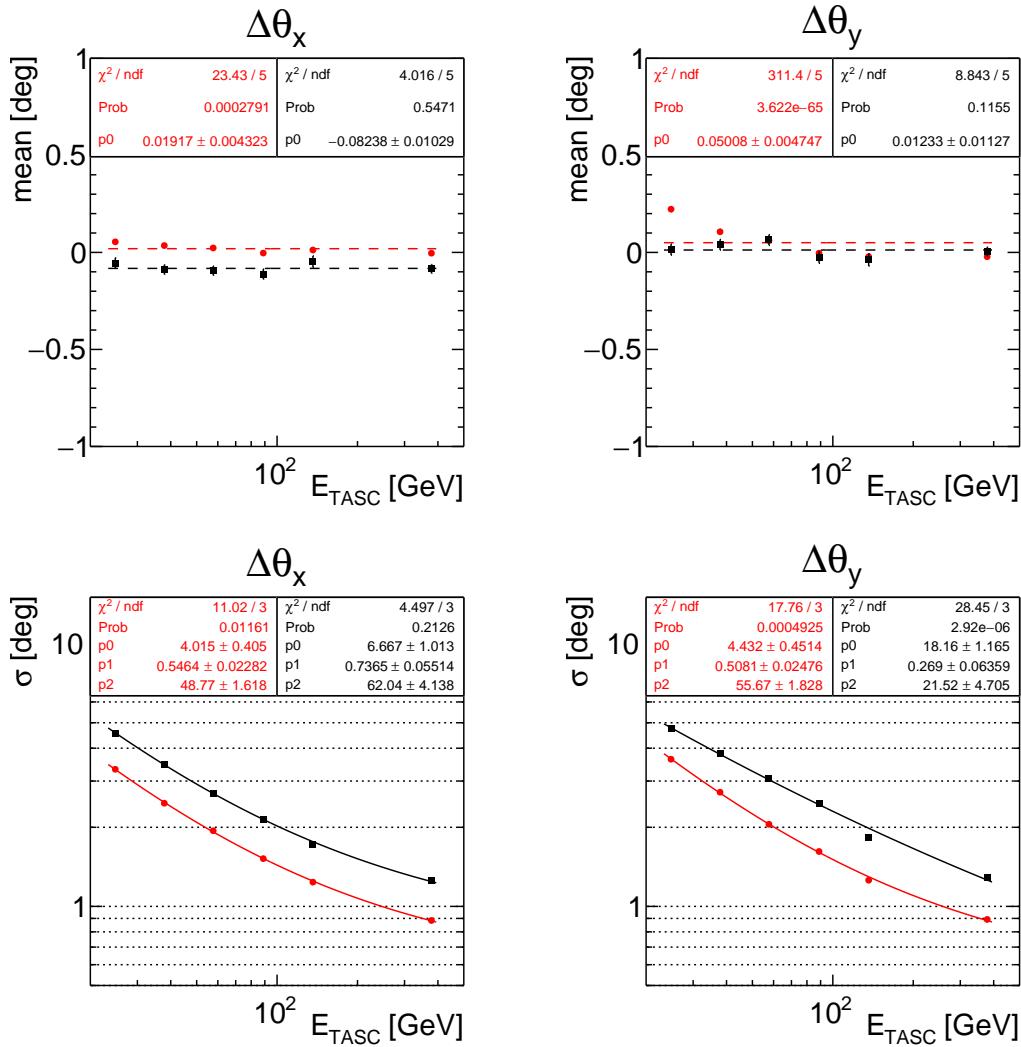


Figura B.38: nei grafici *sopra* il valore medio della distribuzione dei residui; *sotto* la risoluzione angolare della distribuzione dei residui delle variabili angolari della CNN allenata sui nuclei di elio (in *rosso*) e del metodo dell'asse dello sciame (in *nero*), in funzione dell'energia media depositata in ciascun range.

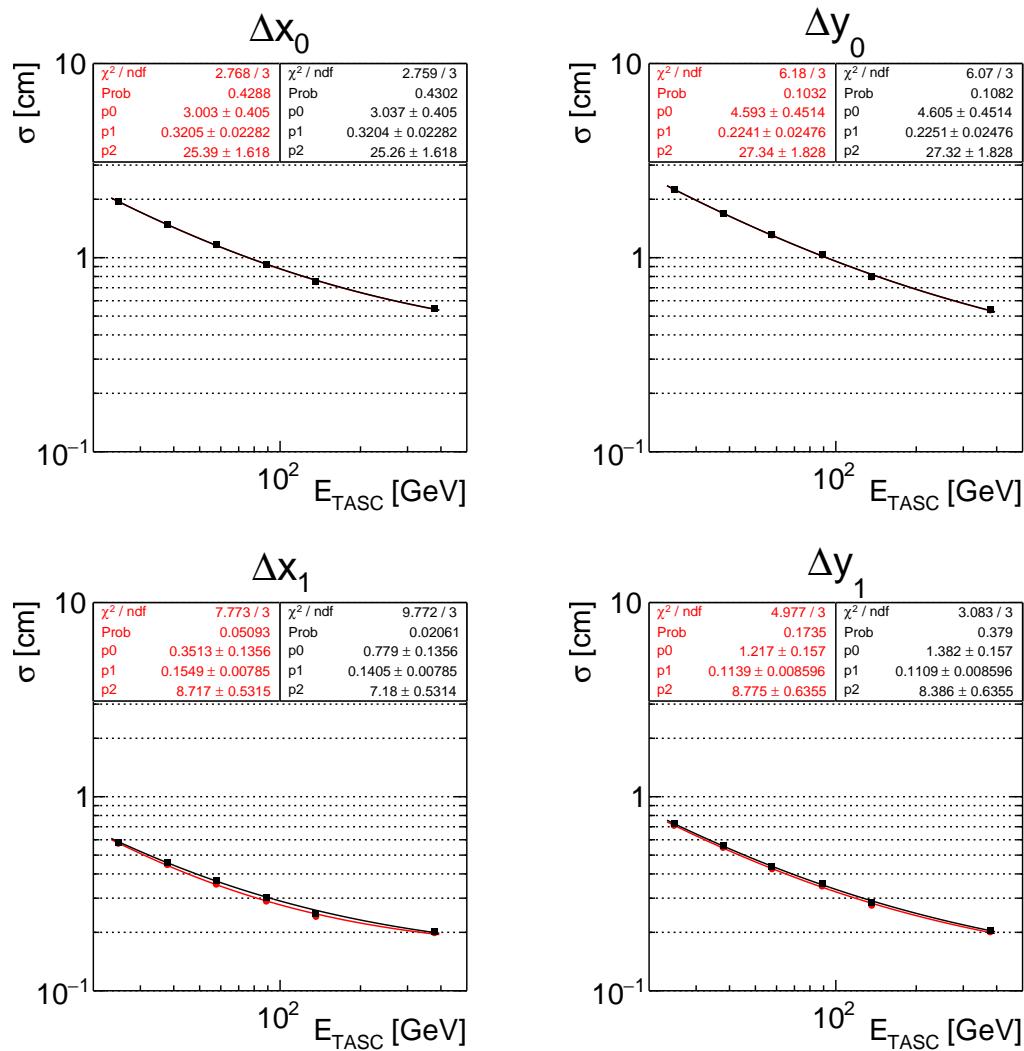


Figura B.39: risoluzione spaziale *effettiva* della CNN allenata sui nuclei di elio (in *rosso*) e del metodo Kalman filter (in *nero*), in funzione dell'energia media depositata in ciascun range.

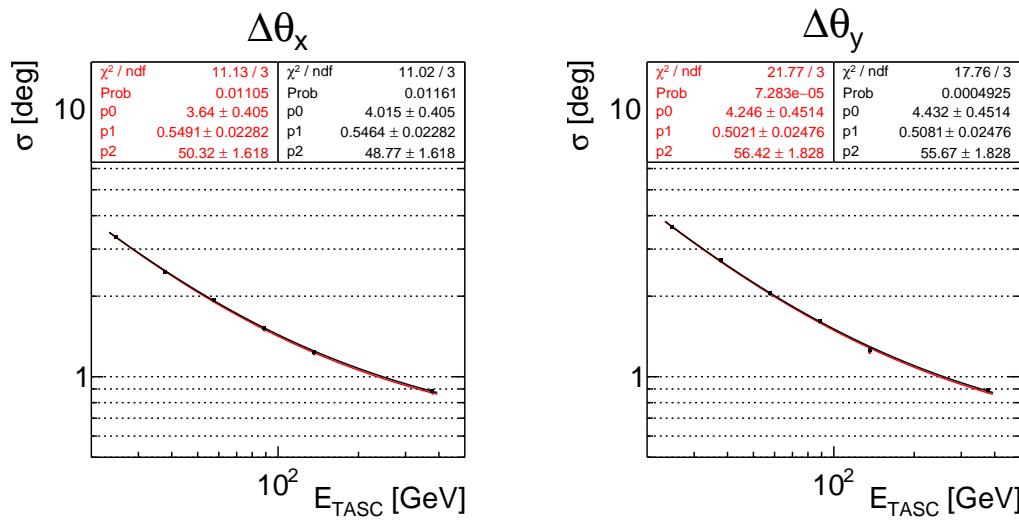


Figura B.40: risoluzione angolare *effettiva* della CNN allenata sui nuclei di elio (in *rosso*) e del metodo Kalman filter (in *nero*), in funzione dell’energia media depositata in ciascun range.

## B.5 Campione di p e nuclei di He dei dati di volo

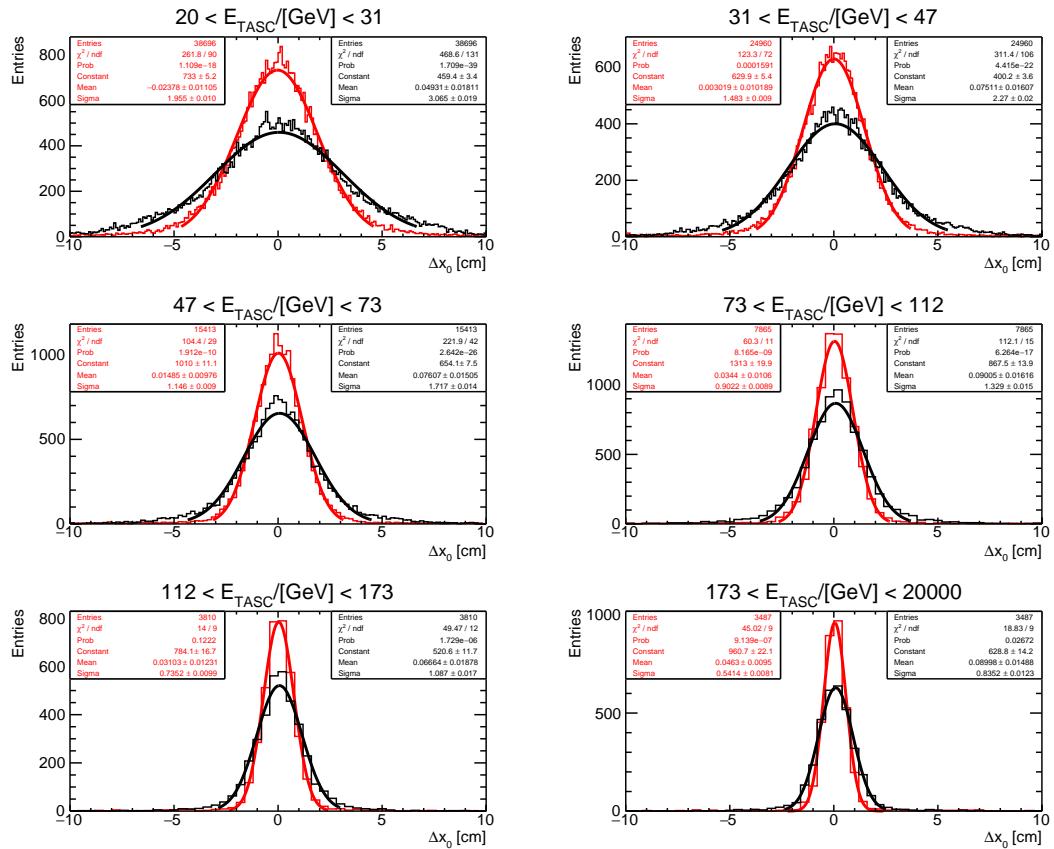


Figura B.41: distribuzione dei residui della variabile di impact point  $x_0$  predetta dalla CNN allenata su dati combinati di p e He (istogramma in rosso) e dal metodo dell’asse dello sciame (istogramma in nero). Le curve sovrapposte agli istogrammi costituiscono i fit gaussiani agli istogrammi del medesimo colore.

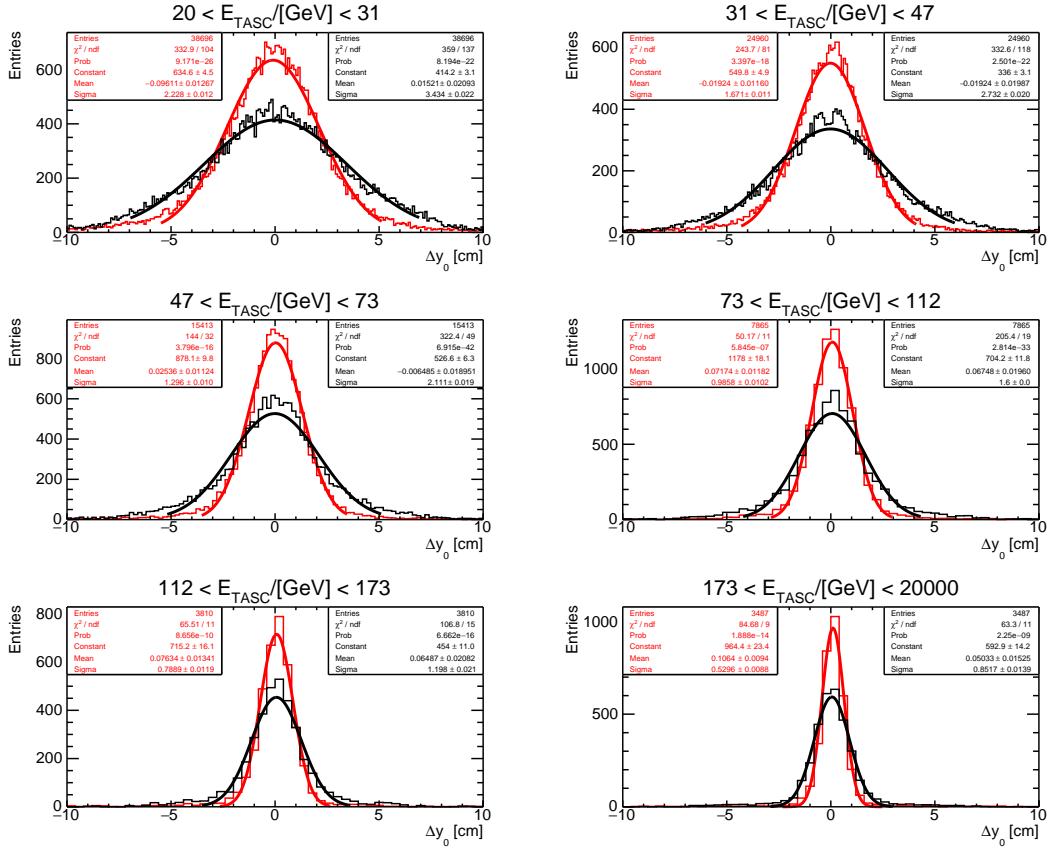


Figura B.42: distribuzione dei residui della variabile di impact point  $y_0$  predetta dalla CNN allenata su dati combinati di p e He (istogramma in rosso) e dal metodo dell'asse dello sciame (istogramma in nero). Le curve sovrapposte agli istogrammi costituiscono i fit gaussiani agli istogrammi del medesimo colore.

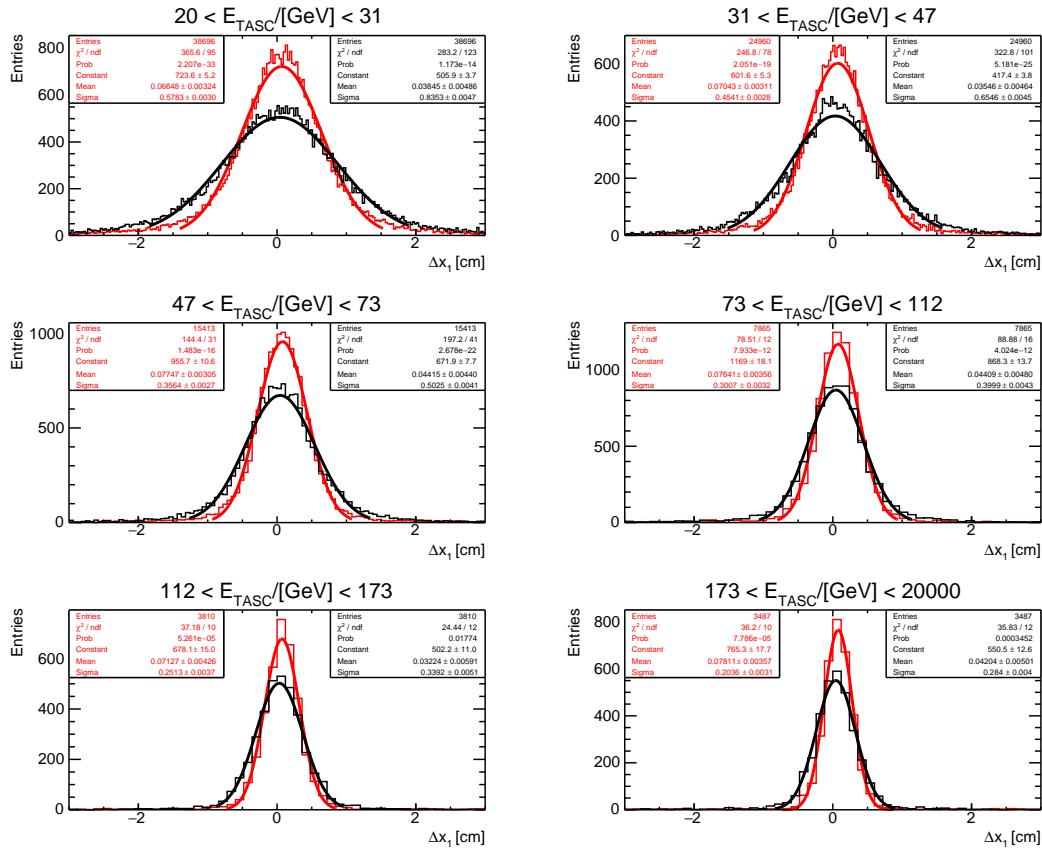


Figura B.43: distribuzione dei residui della variabile di impact point  $x_1$  predetta dalla CNN allenata su dati combinati di p e He (istogramma in rosso) e dal metodo dell'asse dello sciame (istogramma in nero). Le curve sovrapposte agli istogrammi costituiscono i fit gaussiani agli istogrammi del medesimo colore.

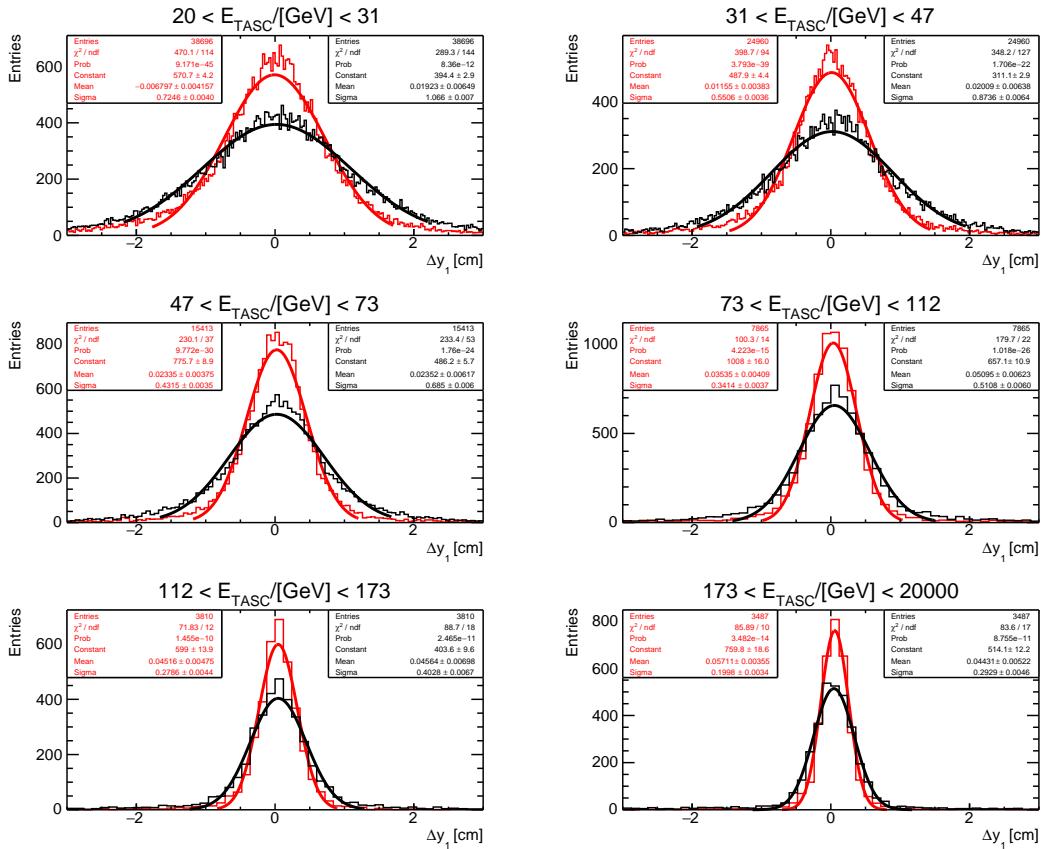


Figura B.44: distribuzione dei residui della variabile di impact point  $y_1$  predetta dalla CNN allenata su dati combinati di p e He (istogramma in rosso) e dal metodo dell'asse dello sciame (istogramma in nero). Le curve sovrapposte agli istogrammi costituiscono i fit gaussiani agli istogrammi del medesimo colore.

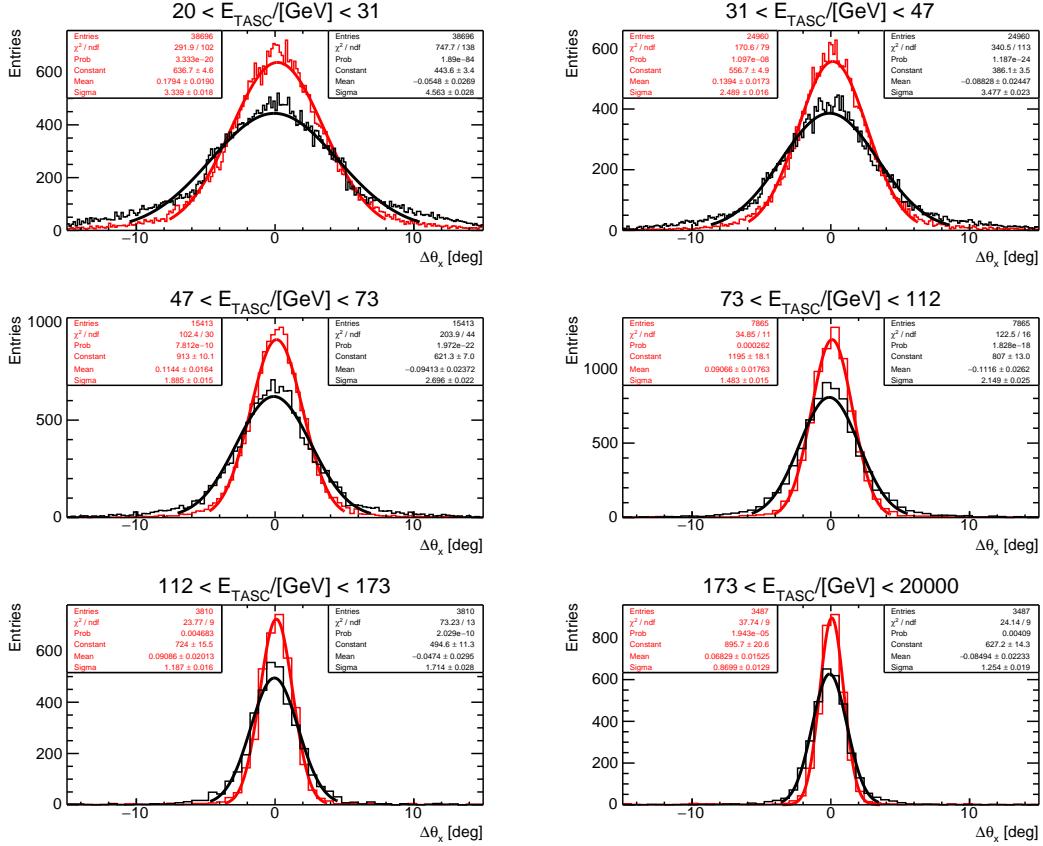


Figura B.45: distribuzione dei residui della variabile angolare  $\theta_x$  predetta dalla CNN allenata su dati combinati di p e He (istogramma in rosso) e dal metodo dell'asse dello sciame (istogramma in nero). Le curve sovrapposte agli istogrammi costituiscono i fit gaussiani agli istogrammi del medesimo colore.

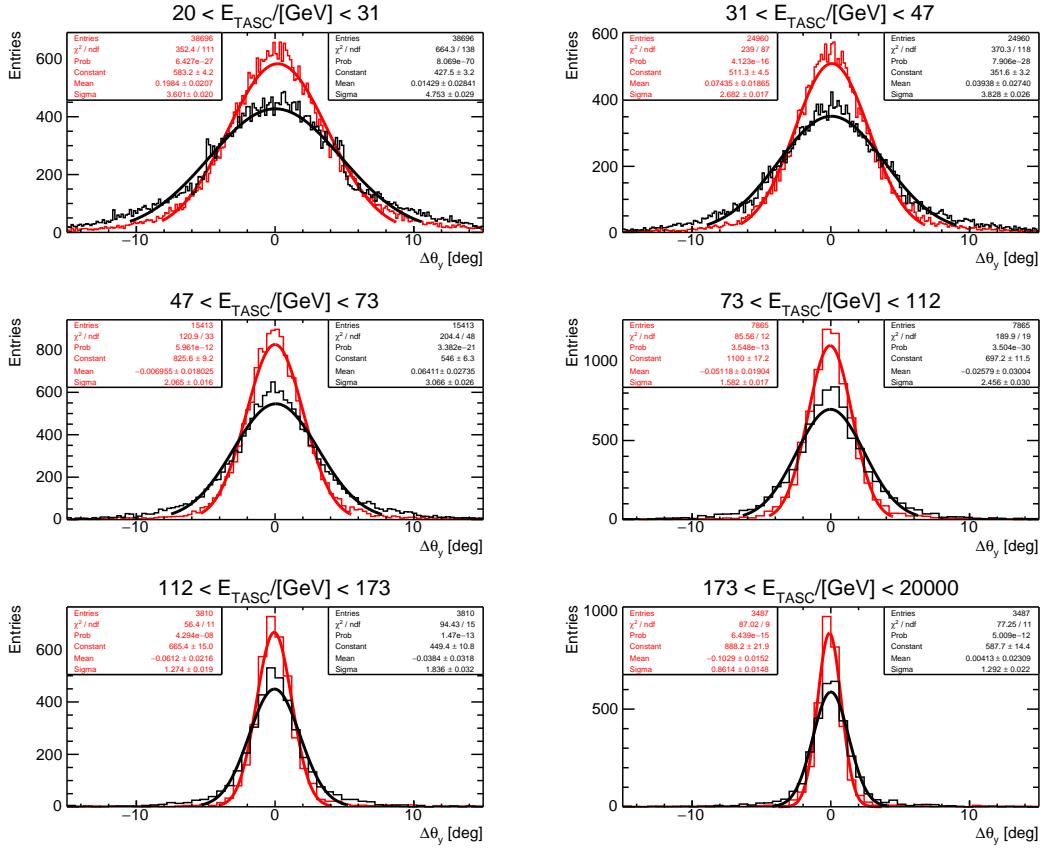


Figura B.46: distribuzione dei residui della variabile angolare  $\theta_y$  predetta dalla CNN allenata su dati combinati di p e He (istogramma in rosso) e dal metodo dell'asse dello sciame (istogramma in nero). Le curve sovrapposte agli istogrammi costituiscono i fit gaussiani agli istogrammi del medesimo colore.

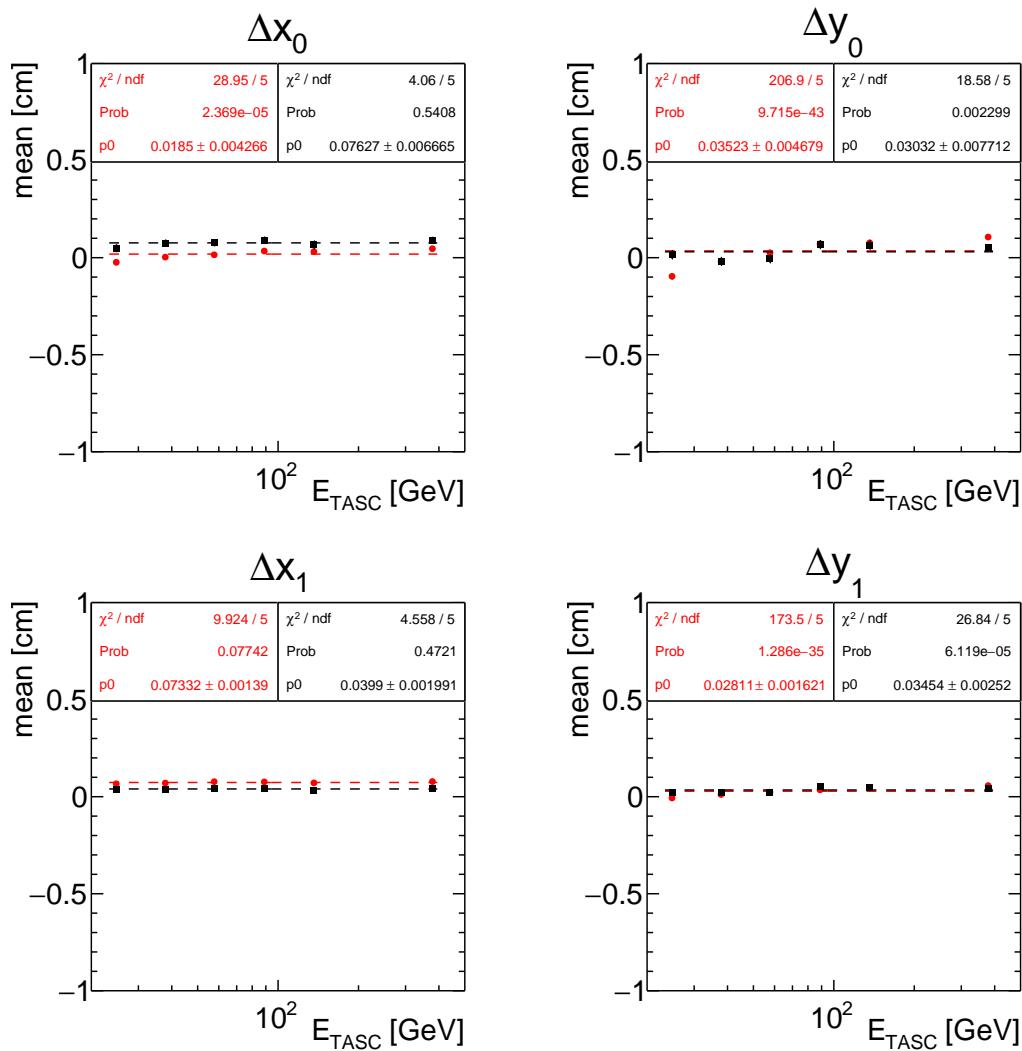


Figura B.47: valore medio della distribuzione dei residui dei punti di impatto della CNN allenata su dati combinati di p e He (in rosso) e del metodo dell’asse dello sciame (in nero), in funzione dell’energia media depositata in ciascun range.

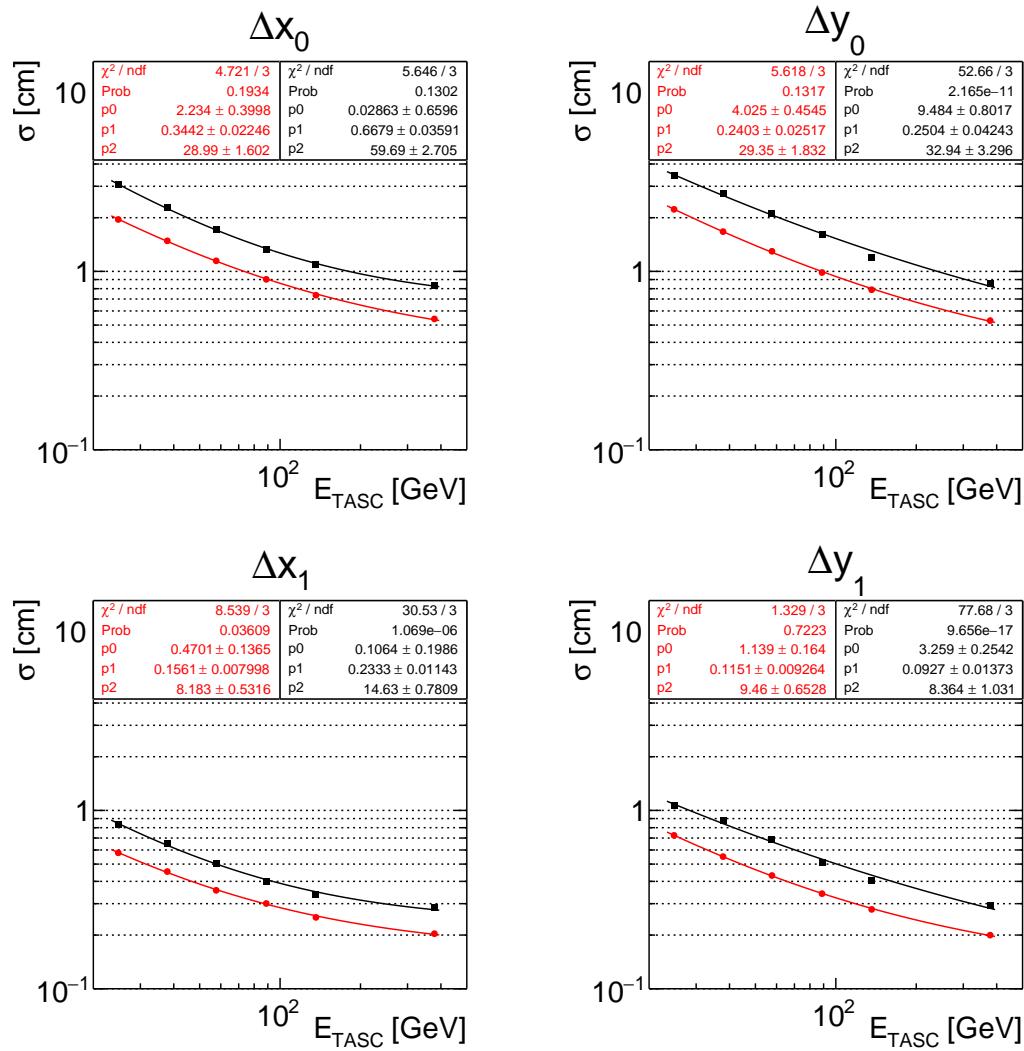


Figura B.48: risoluzione spaziale della CNN allenata su dati combinati di p e He (in rosso) e del metodo dell'asse dello sciame (in nero), in funzione dell'energia media depositata in ciascun range.

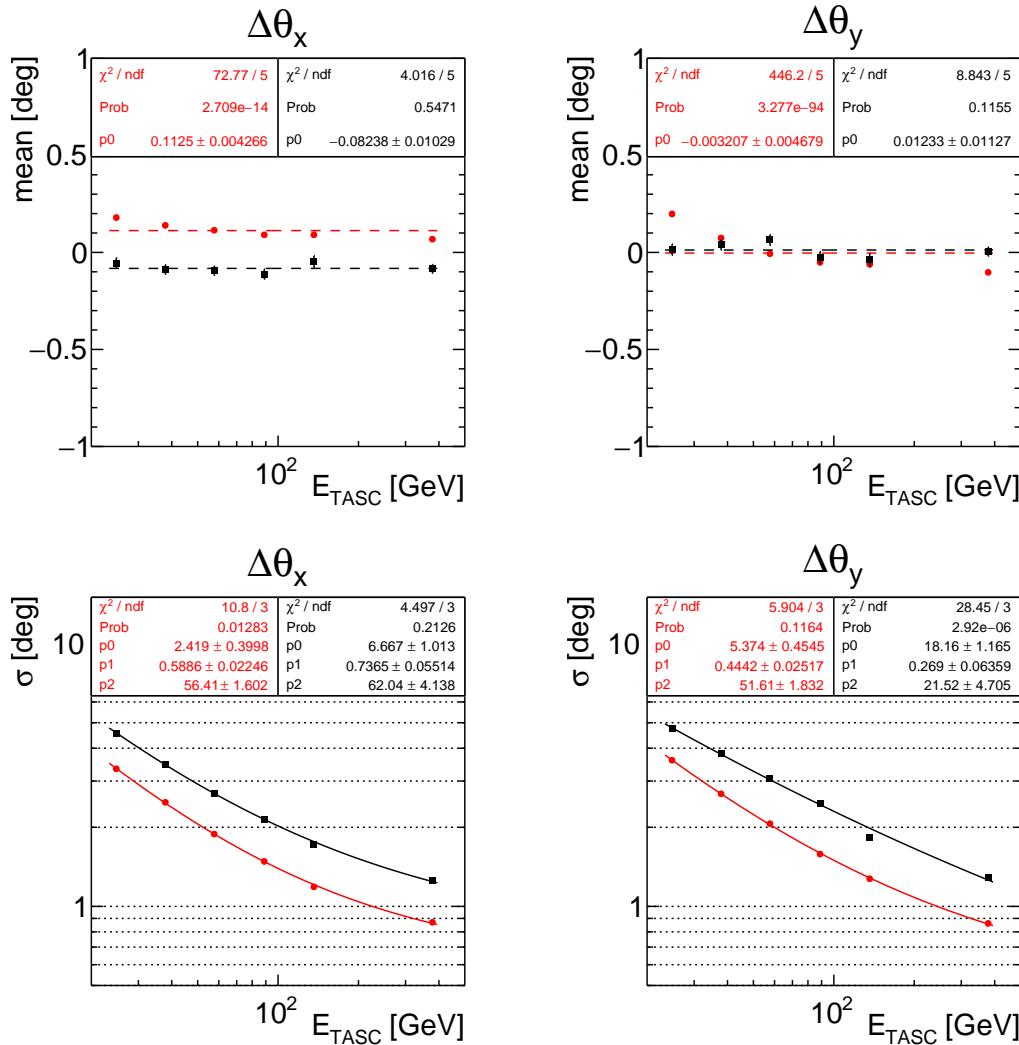


Figura B.49: nei grafici *sopra* il valore medio della distribuzione dei residui; *sotto* la risoluzione angolare della distribuzione dei residui delle variabili angolari della CNN allenata su dati combinati di p e He (in rosso) e del metodo dell'asse dello sciame (in nero), in funzione dell'energia media depositata in ciascun range.

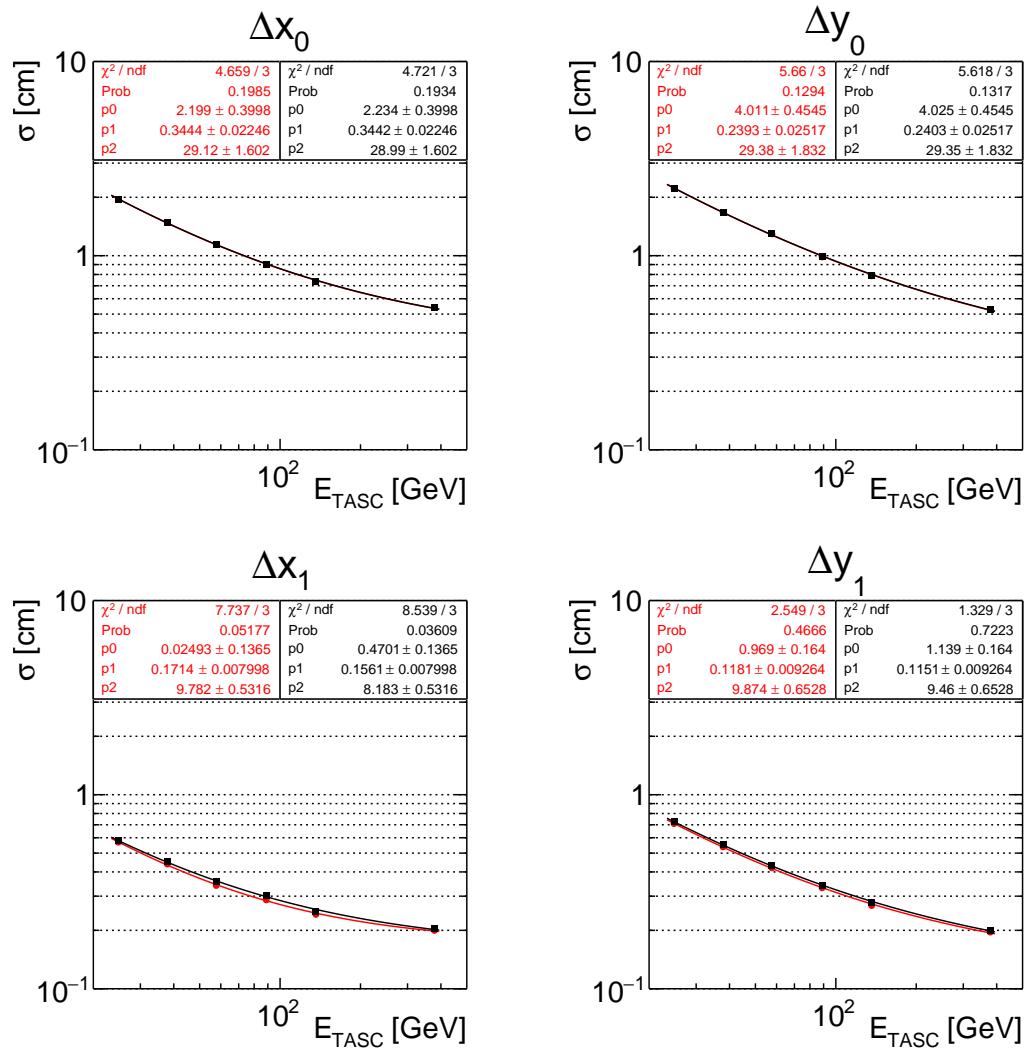


Figura B.50: risoluzione spaziale *effettiva* della CNN allenata su dati combinati di p e He (in rosso) e del metodo Kalman filter (in nero), in funzione dell'energia media depositata in ciascun range.

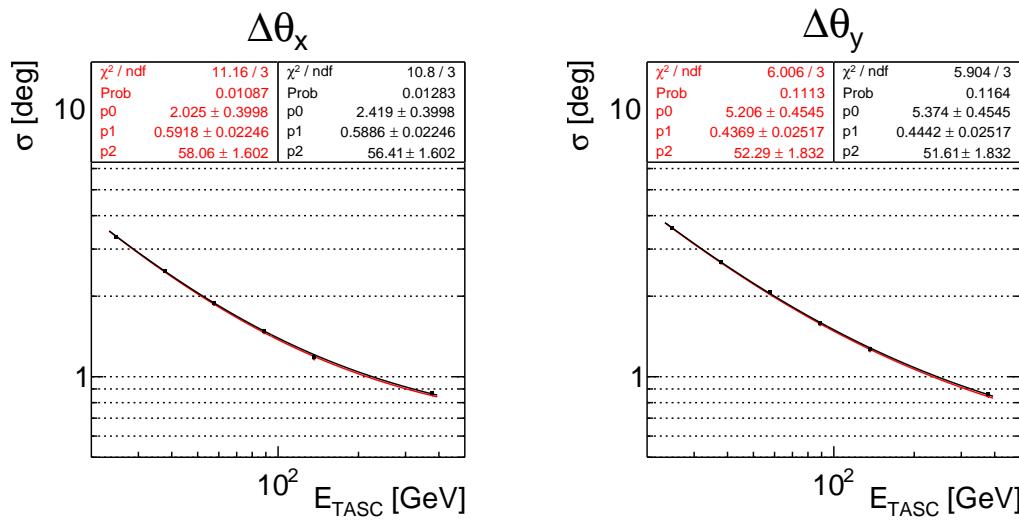


Figura B.51: risoluzione angolare *effettiva* della CNN allenata su dati combinati di p e He (in rosso) e del metodo Kalman filter (in nero), in funzione dell'energia media depositata in ciascun range.

## B.6 Selezione della carica dei nuclei

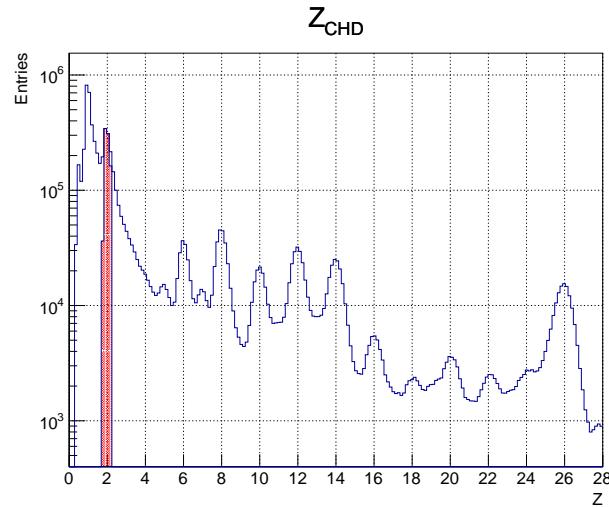


Figura B.52: in *blu* la carica ricostruita nei due piani del CHD nel range (0, 28); in *rosso* i nuclei di elio selezionati nel range (1.8, 2.2).

Si osservano i picchi in corrispondenza dei nuclei (p, He, C, O, Ne, Mg, Si, S, Ar, Ca, Ti, Cr, Fe).

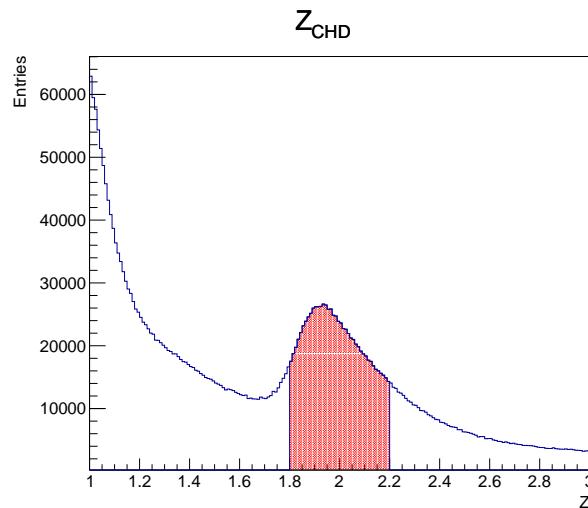


Figura B.53: in *blu* lo zoom della carica ricostruita nei due piani del CHD nel range (1, 3); in *rosso* i nuclei di elio selezionati nel range (1.8, 2.2).

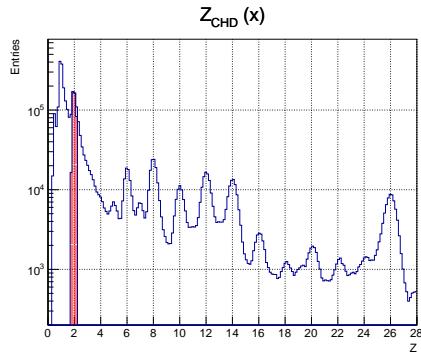


Figura B.54: in *blu* la carica ricostruita nel piano *xz* del CHD nel range (0, 28); in *rosso* i nuclei di elio selezionati nel range (1.8, 2.2).

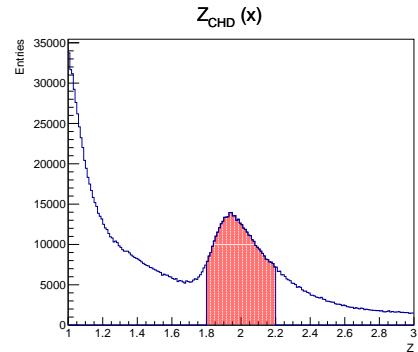


Figura B.55: in *blu* lo zoom della carica ricostruita nel piano *xz* del CHD nel range (1, 3); in *rosso* i nuclei di elio selezionati nel range (1.8, 2.2).

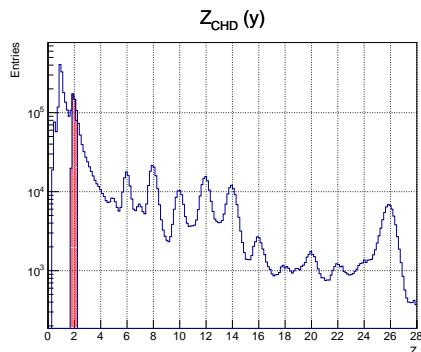


Figura B.56: in *blu* la carica ricostruita nel piano *yz* del CHD nel range (0, 28); in *rosso* i nuclei di elio selezionati nel range (1.8, 2.2).

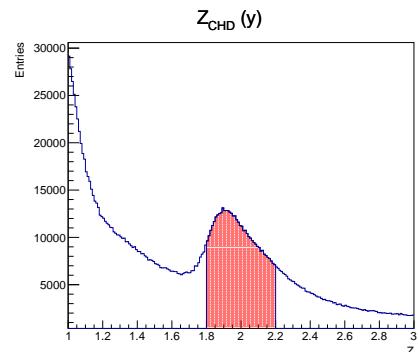


Figura B.57: in *blu* lo zoom della carica ricostruita nel piano *yz* del CHD nel range (1, 3); in *rosso* i nuclei di elio selezionati nel range (1.8, 2.2).

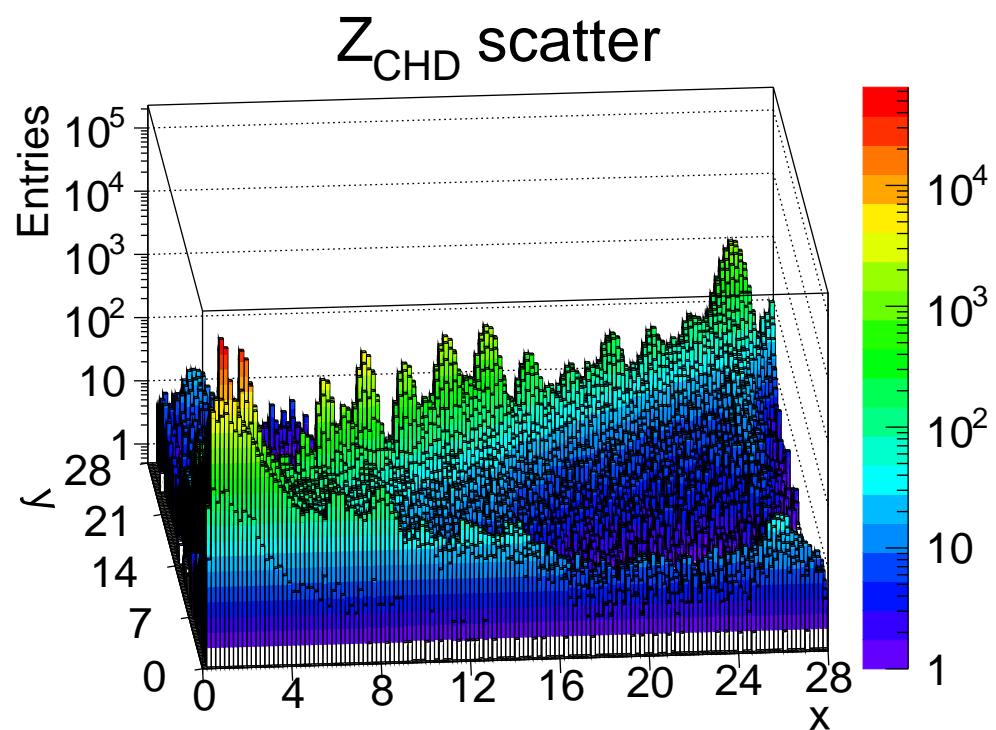


Figura B.58: scatter plot tridimensionale della carica ricostruita nei piani  $xz$  e  $yz$  del CHD nel range (0, 28); la scala sull'asse  $z$  è logaritmica; è possibile apprezzare i picchi relativi ai nuclei misurati nel calorimetro, come riportato in B.52.

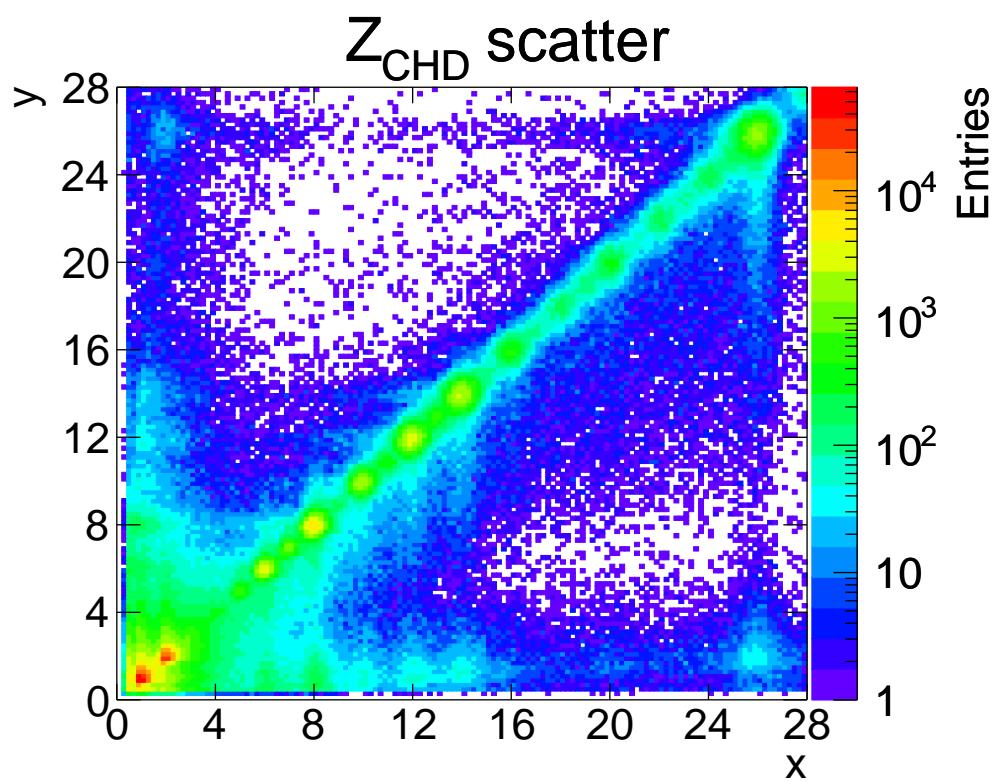


Figura B.59: scatter plot bidimensionale (con visuale *dall'alto*) della carica ricostruita nei piani  $xz$  e  $yz$  del CHD nel range (0, 28); la scala sull'asse  $z$  è logaritmica; è possibile apprezzare i picchi relativi ai nuclei misurati nel calorimetro, come riportato in B.52.

# Bibliografia

- [1] V. Hess. Penetrating Radiation in seven free balloon flight. *Z. Phys.*, 13:1084, 1912.
- [2] A. Burkov. *The hundred-page machine learning book*, volume 1. Andriy Burkov Quebec City, QC, Canada, 2019.
- [3] F. Chollet. *Deep learning with Python*. Simon and Schuster, 2021.
- [4] I. Goodfellow, Y. Bengio, and A. Courville. *Deep learning*. MIT press, 2016.
- [5] Y. LeCun et al. Generalization and network design strategies. *Connectionism in perspective*, 19(143-155):18, 1989.
- [6] W. R. Leo. *Techniques for nuclear and particle physics experiments: a how-to approach*. Springer Science & Business Media, 1994.
- [7] E. Segrè. *Nuclei e particelle: introduzione alla fisica nucleare e subnucleare*. Zanichelli, 1982.
- [8] R. Gambelli. *Sviluppo di un algoritmo basato su Hough transform per la ricostruzione di tracce di particelle cariche nell'esperimento CALET*. Bachelor's thesis, Università degli Studi di Siena, 2019.
- [9] B. Rossi and K. Greisen. Cosmic-Ray Theory. *Rev. Mod. Phys.*, 13:240–309, Oct 1941.
- [10] S. Torii and P. S. Marrocchesi. The CALorimetric Electron Telescope (CALET) on the International Space Station. *Advances in Space Research*, 64(12):2531–2537, 2019. Advances in Cosmic-Ray Astrophysics and Related Areas.
- [11] S. Torii. The CALorimetric Electron Telescope (CALET) on the ISS: Preliminary Results from On-orbit Observations since October, 2015. *PoS, ICRC2017:1092*, 2017.
- [12] N. Cannady et al. Characteristics and performance of the CALorimetric Electron Telescope (CALET) calorimeter for gamma-ray observations. *The Astrophysical Journal Supplement Series*, 238(1):5, 2018.

- [13] P. S. Marrocchesi, O. Adriani, et al. Beam test performance of a scintillator-based detector for the charge identification of relativistic ions. *Nuclear Instruments and Methods in Physics Res. Aearch Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 659(1):477–483, 2011.
- [14] Y. Asaoka, S. Ozawa, et al. On-orbit operations and offline data processing of CALET onboard the ISS. *Astroparticle Physics*, 100:29–37, 2018.
- [15] Y. Asaoka, Y. Akaike, et al. Energy calibration of CALET onboard the International Space Station. *Astroparticle Physics*, 91:1–10, 2017.
- [16] G. Mazzamuto. *Simulazione del calorimetro per raggi cosmici CALET e stima della separazione elettrone-positrone*. Master thesis, Università degli Studi di Firenze, 2011.
- [17] O. Adriani, Y. Akaike, et al. Direct Measurement of the Cosmic-Ray Helium Spectrum from 40 GeV to 250 TeV with the Calorimetric Electron Telescope on the International Space Station. *Phys. Rev. Lett.*, 130:171002, Apr 2023.
- [18] O. Adriani, Y. Akaike, et al. Observation of Spectral Structures in the Flux of Cosmic-Ray Protons from 50 GeV to 60 TeV with the Calorimetric Electron Telescope on the International Space Station. *Phys. Rev. Lett.*, 129:101102, Sep 2022.
- [19] O. Adriani, Y. Akaike, et al. Direct Measurement of the Cosmic-Ray Proton Spectrum from 50 GeV to 10 TeV with the Calorimetric Electron Telescope on the International Space Station. *Phys. Rev. Lett.*, 122:181102, May 2019.
- [20] O. Adriani, Y. Akaike, et al. Direct Measurement of the Cosmic-Ray Carbon and Oxygen Spectra from 10 GeV/ $n$  to 2.2 TeV/ $n$  with the Calorimetric Electron Telescope on the International Space Station. *Phys. Rev. Lett.*, 125:251102, Dec 2020.
- [21] O. Adriani, Y. Akaike, et al. Measurement of the Iron Spectrum in Cosmic Rays from 10 GeV/ $n$  to 2.0 TeV/ $n$  with the Calorimetric Electron Telescope on the International Space Station. *Phys. Rev. Lett.*, 126:241101, Jun 2021.
- [22] O. Adriani, Y. Akaike, et al. Direct Measurement of the Nickel Spectrum in Cosmic Rays in the Energy Range from 8.8 GeV/ $n$  to 240 GeV/ $n$  with CALET on the International Space Station. *Phys. Rev. Lett.*, 128:131103, Apr 2022.
- [23] K. Kasahara. Introduction to Cosmos and some Relevance to Ultra High Energy Cosmic Ray Air Showers. In *International Cosmic Ray Conference*, volume 1 of *International Cosmic Ray Conference*, page 399, January 1995.

- [24] Y. Akaike. CALET observational performance expected by CERN beam test. In *33rd International Cosmic Ray Conference*, page 0726, 2013.
- [25] Y. Akaike, N. Cannady, CALET Collaboration, et al. Simulations for CALET energy calibration confirmed using CERN-SPS beam tests. *UMBC Physics Department*, 2015.
- [26] T. T. Böhlen, F. Cerutti, et al. The FLUKA Code: Developments and Challenges for High Energy and Medical Applications. *Nuclear Data Sheets*, 120:211–214, 2014.
- [27] F. Alemanno, Q. An, et al. Measurement of the Cosmic Ray Helium Energy Spectrum from 70 GeV to 80 TeV with the DAMPE Space Mission. *Phys. Rev. Lett.*, 126:201102, May 2021.
- [28] M. Aguilar, D. Aisa, et al. Precision Measurement of the Helium Flux in Primary Cosmic Rays of Rigidities 1.9 GV to 3 TV with the Alpha Magnetic Spectrometer on the International Space Station. *Phys. Rev. Lett.*, 115:211101, Nov 2015.
- [29] Y. S. Yoon, H. S. Ahn, et al. Comsmic-ray proton and helium spectra from the first CREAM flight. *The Astrophysical Journal*, 728(2):122, jan 2011.
- [30] S. Roesler, R. Engel, and J. Ranft. *The Monte Carlo Event Generator DPMJET-III*, page 1033–1038. Springer Berlin Heidelberg, 2001.
- [31] R. Brun and F. Rademakers. ROOT – An object oriented data analysis framework. *Nucl. Instrum. Methods Phys. Res. ANuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 389(1):81–86, 1997. New Computing Techniques in Physics Research V.
- [32] F. Chollet et al. Keras. <https://keras.io>, 2015.
- [33] A. Simonelli. *Sviluppo di un metodo di Deep Learning per la ricostruzione della traiettoria dei raggi cosmici nell'esperimento CALET*. Bachelor's thesis, Università degli Studi di Siena, 2023.
- [34] R. H. Charles et al. Array programming with NumPy. *Nature*, 585(7825):357–362, September 2020.
- [35] S. M. Ross. *Introduzione alla statistica*. Maggioli Editore, 2014.
- [36] R. Frühwirth. "Application of Kalman filtering to track and vertex fitting". *Nucl. Instrum. Methods Phys. Res. A. and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 262(2):444–450, 1987.

- [37] P. Maestro and N. Mori. Particle tracking in the CALET experiment. *PoS*, ICRC2017:208, 2017.
- [38] J. J. Gomez, J. Velasco, and E. Maestro. Study of an algorithm for electron-hadron separation in modular homogeneous calorimeters. *Nucl.ear Instrum.ents and Methods in Phys.ics Res. Aearch Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 262(2):284–290, 1987.