



UNIVERSITÀ DEGLI STUDI DI TRIESTE

---

DIPARTIMENTO DI FISICA

Corso di Laurea in Fisica

Tesi di Laurea

STUDIO DI TECNICHE DI DECONVOLUZIONE  
DI EFFETTI SPERIMENTALI

Laureando:  
Alessandro Da Rold

Relatore:  
Prof. Giuseppe Della Ricca

---

ANNO ACCADEMICO 2014–2015

# Ringraziamenti

*Do not read so much, look about you and think of what you see there.*

*Richard P. Feynman*

I ringraziamenti sono solitamente molto formali e molto noiosi da leggere, cercherò quindi di dare un tono diverso a questi.

Per prima cosa (anche se lui non vuole) voglio ringraziare il mio relatore, Giuseppe Della Ricca, il quale si è reso disponibile in ogni momento della giornata, di qualsiasi giornata, con un'infinità di mezzi di comunicazione e ha avuto un'infinita pazienza nello spiegare e rispiegare, revisionare e correggere questo lavoro.

Voglio poi dire grazie alla mia famiglia, ad ognuno di loro. Ai miei genitori, ai miei nonni, a mio zio e tutti (ma davvero tutti) gli altri parenti il cui grado di parentela impiegherebbe mezza pagina per essere scritto e un'ora per essere compreso. Senza di loro non sarei la persona che sono, non avrei sviluppato le mie passioni e non sarei (probabilmente) qui oggi. Mi hanno sempre incondizionatamente sostenuto e spronato a fare meglio e, molto più prosaicamente, li voglio ringraziare per tutti i sacrifici che hanno fatto per sostenermi economicamente, altra condizione necessaria (ma non sufficiente) per essere qui oggi.

Voglio quindi ringraziare gli amici tutti. Quelli "vecchi" con cui ho passato mille avventure fin da bambino e con cui rido ancora ogni volta che ci rivediamo, con cui l'amicizia è salda e duratura. E quelli "nuovi", conosciuti a Trieste, con cui ho passato altre mille avventure (e sofferenze) e che hanno contribuito a questo momento con infinito aiuto e collaborazione.

Infine, senza che nessuno sopraccitato ne abbia a male, voglio ringraziare in particolare Marianna, che ha reso la mia vita fantastica negli ultimi anni, con tutte quelle piccole (e grandi) cose che ha fatto e continua a fare per me.

# Indice

<b>Introduzione</b>	<b>1</b>
<b>1 Il problema della deconvoluzione degli effetti strumentali</b>	<b>2</b>
1.1 Introduzione all'unfolding . . . . .	2
1.2 Il metodo SVD . . . . .	3
1.3 Il metodo Bayes . . . . .	5
<b>2 L'implementazione dell'unfolding in RooUnfold</b>	<b>8</b>
2.1 Esempio base . . . . .	8
2.2 Studio dei parametri di regolarizzazione . . . . .	11
2.2.1 Regolarizzazione per il metodo SVD . . . . .	11
2.2.2 Regolarizzazione per il metodo Bayes . . . . .	13
2.3 Confronto SVD-Bayes . . . . .	15
2.4 Metodo toy-1 per la stima degli errori e confronto con il metodo standard toy-0 . . . . .	18
<b>3 Una applicazione di deconvoluzione a dati reali</b>	<b>22</b>
3.1 Descrizione delle osservabili studiate: analisi $Z+b(b)$ . . . . .	22
3.2 Risultati di riferimento della deconvoluzione: regolarizzazione, SVD, Bayes . . . . .	24
3.3 Descrizione del metodo toy-2 e sua applicazione . . . . .	26
<b>Conclusioni</b>	<b>31</b>
<b>Bibliografia</b>	<b>32</b>

# Introduzione

Volendo misurare una quantità fisica in un esperimento di fisica delle alte energie, si deve tenere in considerazione la presenza di effetti propri del rivelatore utilizzato. In particolare, la distribuzione ottenuta non corrisponderà mai alla situazione reale che si è venuta a creare, ma sarà legata ad effetti di risoluzione e accettazione dello strumento impiegato per ottenerla. Per riottenere la distribuzione iniziale, quella che descrive propriamente l'evento fisico in esame, è necessario eseguire una *deconvoluzione* (*unfolding*) basata su un certo algoritmo matematico.

Scopo di questa tesi, è quello di testare due di questi algoritmi per mezzo di simulazioni numeriche nell'ambiente di Root e verificarne quindi proprietà e limiti.

In particolare, nel primo capitolo verrà presentata una breve introduzione al problema matematico legato alla deconvoluzione, prima di descrivere in maniera analitica il procedimento seguito dai due algoritmi che sarà poi implementato successivamente.

Il secondo capitolo tratterà la deconvoluzione nel caso di un esempio generato con il metodo di Monte Carlo. In questo modo saranno studiate le caratteristiche principali dei due algoritmi fondando l'analisi sulla conoscenza a priori del fenomeno. In particolare, si analizzerà il modo in cui varia la distribuzione ricostruita al variare del parametro di regolarizzazione prima di effettuare il confronto tra i risultati dei due metodi. Verrà quindi descritto e implementato un modo per calcolare gli errori sulle distribuzioni che verrà poi applicato al caso in esame e confrontato con il metodo utilizzato di base dal programma.

Nel terzo capitolo si seguiranno i processi del secondo capitolo, applicando tuttavia gli algoritmi a dati ottenuti dall'esperimento CMS. Verrà in particolare integrato il modo di calcolare l'errore e verranno presentati i risultati alla luce del fatto che non è in questo caso nota la funzione analitica che descrive il fenomeno.

# Capitolo 1

## Il problema della deconvoluzione degli effetti strumentali

### 1.1 Introduzione all'unfolding

Negli esperimenti di fisica delle alte energie spesso non è possibile ottenere direttamente la misura di una grandezza a causa degli effetti propri degli strumenti di misurazione. I principali motivi che fanno sì che non si sia in grado di ottenere una misurazione corretta di un evento fisico sono legati alla limitata accettazione, che impedisce di avere la certezza di registrare un evento, e alla limitata risoluzione, la quale introduce un certo errore nei dati raccolti [1].

Quello che si misura non è quindi la grandezza fisica desiderata, ma un'altra grandezza (prodotta dagli effetti di disturbo del rivelatore) e la sua distribuzione. Sia  $f(x)$  la distribuzione teorica dell'evento e  $g(y)$  quella misurata con il rivelatore, avremo quindi:

$$b(y) = \int A(y, z)x(z)dz \quad (1.1)$$

dove  $A(y, x)$  è una funzione che correla le due distribuzioni. Questa funzione di risposta può essere, in linea di principio, determinata tramite l'utilizzo del rivelatore per la misura di una quantità fisica ben determinata e nota a priori, come ad esempio un fascio di particelle ad energia fissata; in questo caso, supponendo il fascio ideale, avremmo una certa osservabile  $x = x_0$  e la funzione di risposta avrebbe la forma di una  $\delta$  da cui:

$$\int A(y, z)\delta(z - z_0)dz = A(y, z_0) \quad (1.2)$$

Questo approccio non è tuttavia applicabile nella gran parte delle misure sperimentali, per questo motivo si usa una simulazione Monte Carlo tramite la quale si è in grado di valutare qualitativamente quale sia la funzione di risposta del rivelatore. Ovviamente il caso in cui le misure siano affette solamente da una limitata accettazione è più facile da analizzare, in quanto è sufficiente ottenere una funzione che descriva l'efficienza dello strumento nell'intervallo relativo alla misura, effettuando un confronto bin per bin tra la distribuzione simulata e i dati raccolti.

Nel caso in cui gli effetti di risoluzione diventino importanti, si evidenzia un problema di migrazione dei dati coerentemente con le barre d'errore, per cui, nel caso ad esempio di un istogramma, un evento finirà nella colonna adiacente (o peggio ancora, in una colonna ancor più lontana da quella del dato effettivo) con una certa probabilità. In linea teorica, si può confrontare la distribuzione ottenuta dalle

misure con quella prevista teoricamente, invertendo l'equazione (1.1). Questo non è tuttavia sempre possibile, primariamente perché non è detto che esista l'inversa della funzione di risposta e secondariamente perché se anche esistesse potrebbe essere eccessivamente sensibile alla variazione del campione per dare una risposta soddisfacente.

Il problema può essere tuttavia affrontato da un differente punto di vista: si può cercare di ricostruire la distribuzione originaria dalla distribuzione misurata. Nel caso di una distribuzione reale, possiamo limitare l'analisi ad un sistema discreto di equazioni, che corrisponderanno in pratica ad un istogramma dei dati raccolti. In questo caso l'equazione (1.1) diventerà:

$$b_i = \sum_{j=1}^m A_{ij} x_j \quad (1.3)$$

dove  $A_{ij}$  è ora una matrice. Come affermato in precedenza, nel caso in cui ci siano solo effetti di accettazione da tenere in considerazione, il problema è piuttosto semplice da risolvere: la matrice è infatti diagonale e gli elementi della stessa sono i "pesi" che descrivono la sensibilità dello strumento nella raccolta dei dati del campione.

Nel caso in cui si tenga conto degli errori sulla risoluzione, come affermato in precedenza, si ha che la matrice non è più diagonale e l'inversione della stessa non porta ai risultati desiderati: la probabilità di misurare un dato evento in maniera corretta (ossia porlo nel corretto intervallo dell'istogramma) non è più 1 (come invece accadeva in precedenza). Applicando l'inversione della matrice con il procedimento analitico si ottiene un risultato che non è in genere soddisfacente: l'andamento della funzione è molto oscillante e si nota una notevole correlazione negativa tra bin adiacenti.

Per ovviare a questi problemi e ottenere quindi la corretta deconvoluzione, un metodo possibile prevede di generare l'istogramma che corrisponde alla previsione teorica con una simulazione Monte Carlo, applicare a questa distribuzione quella che si crede essere la matrice di risposta e quindi confrontare poi il risultato con la distribuzione ottenuta nell'esperimento. Un modo alternativo è quello di utilizzare dei parametri di regolarizzazione noti a priori (ad esempio che limitano l'effetto oscillante dell'istogramma ottenuto, imponendo che la soluzione abbia una certa regolarità). Questo procedimento, ovviamente, introduce dei bias che devono essere tenuti sotto controllo per non falsare il risultato.

Nel seguito verranno descritti due ulteriori metodi di deconvoluzione, che saranno poi ripresi nelle applicazioni pratiche.

## 1.2 Il metodo SVD

Il problema teorico da affrontare è quello descritto nel paragrafo precedente. Considerando una misurazione di un'osservabile otteniamo un istogramma che contiene i dati che vogliamo deconvolvere per ottenere la distribuzione degli stessi senza l'influenza del rivelatore. La formulazione è in questo caso analoga a quella proposta in (1.3) e può essere facilmente generalizzata al caso continuo riportato in (1.1).

L'algoritmo si basa sulla *singular value decomposition* (SVD) [2], ossia sulla fattorizzazione della matrice  $A$ . Supponendo che questa abbia dimensione  $m \times n$ , dove  $m$  è il numero di bin dell'istogramma misurato e  $n$  quello dell'istogramma deconvoluto

(nel seguito supporremo  $m \geq n$ ), possiamo infatti scrivere:

$$A = USV^T \quad (1.4)$$

in questo modo  $U$  è una matrice ortogonale  $m \times m$ ,  $V$  una matrice ortogonale  $n \times n$  e  $S$  è una matrice  $m \times n$  diagonale e non negativa. Gli elementi  $S_{ii} = s_i$  sono detti “valori singolari” (*singular values*) della matrice  $A$ . L’aver scritto la matrice in questo modo, permette di analizzare con maggiore facilità le sue caratteristiche.

Riprendendo l’equazione (1.3), si nota come si possa riformularla in termini di un problema di minimi quadrati, che in generale assumerà la forma:

$$(Ax - b)^T B^{-1} (Ax - b) = \min \quad (1.5)$$

dove  $B$  è la matrice di covarianza dal campione  $b$ .

Il problema principale, per il quale è stato introdotto il concetto di valore singolare, è quello di trovare una certa normalizzazione che permetta di mettere in evidenza i dati significativi per la deconvoluzione senza introdurre artifici matematici che aumenterebbero il peso dei dati non significativi. Nel caso riportato in equazione (1.5), infatti, l’introduzione di una certa normalizzazione cambia il valore degli elementi  $s_i$ , variando quindi la soluzione stessa del problema.

Un possibile approccio è quello di dividere i valori di  $x_i$  per un valore costante e al contempo moltiplicare i corrispondenti valori della matrice per lo stesso valore. Possiamo in questo modo riformulare il problema nel modo seguente:

$$\sum_{j=1}^{n_x} A_{ij} w_j = b_i \quad w_j = \frac{x_j}{x_j^{gen}} \quad (1.6)$$

dove  $x_j^{gen}$  è il valore del  $j$ -esimo bin della distribuzione generata con il Monte Carlo. Questa modifica ha avuto come risultato l’aver una matrice che non contiene più le probabilità di migrazione di un evento da un bin ad un altro, ma contiene i valori effettivi del numero di eventi generati in un bin che sono finiti in un bin differente. Eliminare la matrice con le probabilità permette inoltre di ovviare al problema delle colonne con pochi dati: nella matrice di probabilità, infatti, nel caso in cui ci sia un solo dato che termina nel bin “corretto”, questo verrà preso con un’efficienza del 100% nell’analisi, senza tenere conto che la significatività di questo evento è nulla. Nella matrice calcolata ora, invece, questo problema è risolto: le colonne che prima creavano problemi ora vengono pesate molto meno avendo queste una scarsa statistica. Come diretta conseguenza di questo fatto, si può dimostrare che per ottenere un buon risultato di deconvoluzione la statistica del Monte Carlo deve essere uno o due ordini di grandezza superiore rispetto a quella del campione.

Questo procedimento, tuttavia, non previene il fatto che le distribuzioni deconvolute abbiano un andamento fortemente oscillante da un bin al successivo (rendendo quindi inutilizzabile il risultato ottenuto), anche se questo effetto verrà in linea di massima smorzato rispetto al caso dei dati non normalizzati. Questa componente di oscillazione può essere eliminata introducendo una matrice che corrisponde ad una certa conoscenza a priori del problema. Per fare ciò viene aggiunto un *parametro di regolarizzazione* che modifica la forma da minimizzare in:

$$(\tilde{A}w - \tilde{b})^T (\tilde{A}w - \tilde{b}) + \tau \cdot (Cw)^T Cw = \min \quad (1.7)$$

dove le quantità indicate con il simbolo  $\tilde{\phantom{x}}$  (tilde) corrispondono a quelle trattate in precedenza facendo però riferimento al problema normalizzato,  $w$  è la soluzione del problema,  $C$  è la matrice che definisce la condizione a priori e  $\tau$  è il valore del parametro di regolarizzazione.

L'ipotesi che determina la forma di  $C$  è quella di avere una soluzione “smussata” ovvero che abbia una variazione di valore tra bin adiacenti contenuta. Per fare questo si usa una matrice del tipo:

$$C = \begin{pmatrix} -1 & 1 & 0 & 0 & \dots \\ 1 & -2 & 1 & 0 & \dots \\ 0 & 1 & -2 & 1 & \dots \\ \vdots & \vdots & \vdots & \ddots & \\ & & & 1 & -2 & 1 \\ & & & & 1 & -1 \end{pmatrix}$$

Imponendo inizialmente il parametro di regolarizzazione  $\tau = 0$  ci si riconduce ad un sistema di equazioni lineari risolvibile con la *SVD*:

$$\tilde{A}C^{-1} = USV^T \quad (1.8)$$

e definendo  $d \equiv U^T \tilde{b}$  e  $z \equiv V^T Cw$ , si è in grado di risolvere il problema:

$$s_i \cdot z_i = d_i, \quad i = 1, \dots, n \quad (1.9)$$

da cui:

$$z_i^{(0)} = \frac{d_i}{s_i} \quad \longrightarrow \quad w^{(0)} = C^{-1}Vz^{(0)} \quad (1.10)$$

dove si è usato l'apice (0) per indicare che è stato considerato il parametro di regolarizzazione nullo. Per ottenere la distribuzione deconvoluta bisogna ovviamente moltiplicare questo risultato per  $x^{gen}$ . La generalizzazione da questo problema al caso in cui  $\tau \neq 0$  segue lo stesso procedimento e non verrà trattata in dettaglio.

Il parametro  $d$  fornisce alcune importanti informazioni sul problema. Infatti per distribuzioni abbastanza smussate ci si aspetta che solo i primi termini della decomposizione siano significativi, mentre i successivi dovrebbero essere compatibili con lo zero. Riportando quindi in un grafico il valore di  $d_i$  contro  $i$  ci si aspetta un andamento decrescente (solitamente esponenziale); il valore di  $i$  dopo il quale i  $d_i$  non sono più significativi (sono cioè compatibili con lo 0 entro gli errori) determina il numero di equazioni linearmente indipendenti del sistema. Inoltre, trovato il valore di  $i$  “critico”, una buona scelta per il parametro di regolarizzazione  $\tau$  è quella di prendere il quadrato del valore singolare  $s_i$  corrispondente. In questo modo si riesce a determinare la matrice di risposta che risolve il problema della deconvoluzione.

### 1.3 Il metodo Bayes

Un ulteriore metodo per risolvere il problema della deconvoluzione è quello proposto da G. D'Agostini in [3]. Tra i vantaggi di questo metodo vi è quello di ottenere un ottimo risultato nel caso in cui si conosca a priori la forma della distribuzione che si deve deconvolvere; tuttavia nel caso in cui non si abbiano informazioni su di essa,



si ottiene un comunque un risultato soddisfacente partendo da una distribuzione uniforme.

È possibile formulare il teorema di Bayes in termini di un certo numero di cause  $C_i$  che producono un unico effetto  $E$ . Supponendo note la probabilità iniziale delle cause  $\mathcal{P}(C_i)$  e la probabilità che l' $i$ -esima causa produca l'effetto  $E$ , ossia  $\mathcal{P}(E|C_i)$ , si ha che:

$$\mathcal{P}(C_i|E) = \frac{\mathcal{P}(E|C_i)\mathcal{P}(C_i)}{\sum_{l=1}^{n_c} \mathcal{P}(E|C_l)\mathcal{P}(C_l)} \quad (1.11)$$

da cui si nota che, mentre la conoscenza di  $\mathcal{P}(C_i)$  migliora con l'aumentare della statistica, quella di  $\mathcal{P}(E|C_i)$  non è influenzata dal numero di eventi che si hanno a disposizione e deve quindi essere stimata con un Monte Carlo.

Nel caso in cui si abbia più di un possibile effetto, come nel caso in esame, ci si aspetta che osservando  $n(E_j)$  eventi collegati all'effetto  $E_j$  il numero di eventi che si possono attribuire per ogni causa sia:

$$\hat{n}(C_i) = n(E_j)\mathcal{P}(C_i|E_j) \quad (1.12)$$

La condizione  $\mathcal{P}(C_i|E_j)$  corrisponde quindi alla matrice di *smearing*  $A$  del caso analizzato precedentemente. È possibile quindi riformulare la (1.11) per più effetti come:

$$\mathcal{P}(C_i|E_j) = \frac{\mathcal{P}(E_j|C_i)\mathcal{P}_0(C_i)}{\sum_{l=1}^{n_c} \mathcal{P}(E_j|C_l)\mathcal{P}_0(C_l)} \quad (1.13)$$

dove si è usato il pedice 0 per indicare le probabilità iniziali. Bisogna notare che ogni effetto è, ovviamente, generato da una causa la quale potrebbe essere anche “estranea” al processo in questione e costituire quindi un fondo che dovrà successivamente essere analizzato. Non vale tuttavia la situazione opposta, cioè non è detto che ogni causa produca uno degli effetti in analisi: deve quindi essere introdotto un fattore che tiene in considerazione l'accettazione del rivelatore.

Applicando la (1.12) ad ognuno degli effetti osservati, tenendo conto dell'accettazione, si può ottenere la distribuzione deconvoluta come:

$$\hat{n}(C_i) = \frac{1}{\epsilon_i} \sum_{j=1}^{n_E} n(e_j)\mathcal{P}(C_i|E_j) \quad (1.14)$$

grazie alla quale è possibile calcolare il numero totale di eventi della distribuzione originale, la probabilità (e quindi l'istogramma stesso) delle cause (che è la distribuzione cercata) e l'efficienza complessiva del processo:

$$\hat{N} = \sum_{i=1}^{n_c} \hat{n}(C_i) \quad \hat{\mathcal{P}}(C_i) \equiv \mathcal{P}(C_i|\vec{n}(E)) = \frac{\hat{n}(C_i)}{\hat{N}} \quad \hat{\epsilon} = \frac{N_{oss}}{\hat{N}} \quad (1.15)$$

dove  $N_{oss}$  è il numero di eventi “osservati”, ossia presenti nella distribuzione da deconvolvere.

È da notare che l'efficienza  $\hat{\epsilon}$  che si ottiene in questo modo non è necessariamente in accordo con quella calcolata grazie alla simulazione iniziale di Monte Carlo. Come accennato in precedenza, infatti, nel caso in cui non si conosca la distribuzione teorica dei dati (o la si conosca in maniera approssimata), non vi sarà in generale accordo tra l'istogramma generato con il metodo di Monte Carlo e la distribuzione

ottenuta con la deconvoluzione. Si può dimostrare che la distribuzione  $\hat{\mathcal{P}}(C)$  ottenuta con la deconvoluzione è una approssimazione della distribuzione “vera” migliore rispetto a quella di partenza simulata con il Monte Carlo; questo fa sì che procedendo iterativamente si giunga ad ottenere una deconvoluzione sempre migliore. Il confronto tra le distribuzioni può essere fatto mediante un test di  $\chi^2$  sul quale è opportuno mettere una condizione di arresto una volta ottenuta una soluzione che non differisce apprezzabilmente dalla precedente: in questo caso si è trovata la miglior approssimazione dei dati iniziali ottenibile con questo metodo.

Per fare l’analisi degli errori che vengono a crearsi utilizzando questo metodo per la deconvoluzione, è opportuno riscrivere l’equazione precedente come:

$$\hat{n}(C_i) = \sum_{j=1}^{n_E} M_{ij} n e_j \quad \text{dove} \quad M_{ij} = \frac{\mathcal{P}(E_j|C_i)\mathcal{P}_0(C_i)}{\left[\sum_{l=1}^{n_E} \mathcal{P}(E_l|C_i)\right]\left[\sum_{l=1}^{n_E} \mathcal{P}(E_j|C_l)\mathcal{P}_0(C_l)\right]} \quad (1.16)$$

dove  $M_{ij}$  è la matrice di risposta, che come è stato sottolineato più volte non può semplicemente essere l’inversa della matrice di smearing  $A$ . Grazie a questa descrizione matematica si può valutare quali siano i contributi che entrano nella matrice di correlazione del campione ottenuto. Tra questi, gli errori sulla distribuzione usata inizialmente come “verità” nella simulazione con il Monte Carlo non vengono tenuti in considerazione in quanto influenzano allo stesso modo tutto il campione. Il campione di dati generati segue una distribuzione multinomiale, quindi l’errore viene calcolato di conseguenza. I termini  $\mathcal{P}(E_j|C_i)$  sono frutto di una considerazione a priori e sono soggetti ad errori sia intrinseci alle ipotesi fatte nella loro introduzione, sia dovuti al numero limitato di eventi generati; queste incertezze (come quelle sul campione generato) introducono inoltre effetti di migrazione dei dati da un bin ad un altro.

## Capitolo 2

# L'implementazione dell'unfolding in RooUnfold

Quanto esposto teoricamente nel capitolo precedente, è stato implementato in Root tramite il pacchetto RooUnfold [4]. Il vantaggio principale di questo pacchetto è legato al fatto che implementa più di un metodo di deconvoluzione, nello specifico sia il metodo SVD che quello basato sul teorema di Bayes. Oltre a questo, come si vedrà in dettaglio nel seguito, permette di calcolare gli errori in modi differenti, legati all'algoritmo in uso. Un ulteriore vantaggio è quello di avere due diversi modi di fornire i dati in ingresso per la fase di *train* (la fase iniziale del programma, in cui si istruisce lo stesso sul modo in cui trattare i dati che verranno successivamente immessi), sia inserendo la tripletta di istogrammi necessaria a questo procedimento, che generando i dati stessi nota la funzione analitica che regola il fenomeno fisico in esame. Fornendo quindi la distribuzione dei dati da deconvolvere, si otterrà in risposta un istogramma con la funzione deconvoluta ed i relativi errori.

### 2.1 Esempio base

Prima di affrontare la deconvoluzione di dati ottenuti da un esperimento reale, si è proceduto tramite una simulazione completa di una possibile situazione di analisi. Per fare questo, si è utilizzato un programma già implementato nel pacchetto di RooUnfold al quale sono state apportate di volta in volta le necessarie modifiche.

La prima parte del procedimento di deconvoluzione è detta *train* e consiste, come appunto suggerito dal nome, nell'istruire il programma al tipo di dati da deconvolvere tramite delle distribuzioni generate con una simulazione di Monte Carlo. In questo primo esempio si è scelto di costruire una distribuzione fittizia di dati prendendo una funzione analitica con forma simile a quella che poi descriverà i dati sperimentali. Questa accortezza è fondamentale in quanto la matrice di risposta può variare in maniera sensibile a seconda del campione che si sta analizzando e può portare a risultati piuttosto differenti nel momento in cui la si applichi alla distribuzione dei dati.

Come accennato in precedenza, nella prima fase del programma devono essere generati tre differenti oggetti: un istogramma che contenga i dati della distribuzione "vera" del fenomeno, uno che contenga i dati ottenuti dagli effetti del rivelatore e la matrice che collega le due serie. Nel caso in esame per la generazione della serie di dati che corrispondono al fenomeno teorico si è deciso di utilizzare la distribuzione

di Breit-Wigner la cui forma analitica generale è:

$$f_{(x_0, y_0)}(x) = \frac{1}{\pi} \frac{y_0}{(x - x_0)^2 + y_0^2} \quad (2.1)$$

Nel caso in esame è stata centrata sullo 0 e si è preso come valore della larghezza a mezza altezza 2.5 (questo in accordo con quanto verrà fatto in seguito nella generazione dei dati da deconvolvere). Questo procedimento è analogo a quello che viene fatto nel caso di dati sperimentali da analizzare, dove cioè si cerca di prevedere quale sia la forma della distribuzione avendo a disposizione la spiegazione teorica del fenomeno.

Ogni dato generato in questo modo viene quindi passato attraverso una funzione che simula l'effetto del rivelatore (funzione di *smearing*). Per fare questo, prima viene calcolata una variabile di efficienza con andamento iperbolico  $cost/x$  e in seguito una variabile con distribuzione uniforme. Nel caso in cui la variabile con distribuzione uniforme abbia un valore superiore rispetto a quella di efficienza, il dato in esame viene scartato. Dall'andamento iperbolico della funzione che genera la variabile di efficienza, si vede come sia più probabile che venga accettato un dato vicino all'origine (con valore assoluto di  $x$  piccolo) rispetto ad uno che si discosta sensibilmente dal valore nullo. Dato questo fatto, è quindi logico considerare una funzione centrata nell'origine (come appunto è stato fatto per la Breit-Wigner) la quale favorisce l'accettazione dei dati del picco (che nel caso di un esperimento corrispondono ai dati che descrivono più fedelmente il fenomeno). Nel caso in cui il dato venga accettato, su di esso viene simulato l'effetto di risoluzione, responsabile della variazione del valore dello stesso compatibilmente con l'errore ad esso associato. Per fare questo viene generata una variabile con distribuzione gaussiana la quale viene poi sommata al dato in entrata. La scelta dei parametri caratteristici della gaussiana, valor medio e varianza, deve essere fatta in maniera tale da alterare la distribuzione dei dati generati con il Monte Carlo mantenendone però il significato fisico. Nel caso in cui, ad esempio, si stia analizzando un serie di dati strettamente positiva non è fisicamente accettabile che alcuni di essi assumano un valore negativo dopo lo smussamento con la gaussiana.

Con questo procedimento abbiamo quindi due istogrammi: uno con i dati "veri" generati con il Monte Carlo e uno con i dati che sono stati "rovinati" simulando l'effetto del rivelatore. Ricordando la (1.3) (o una sua forma analoga tra le successive) si è quindi ora in grado di costruire la matrice che correla le due serie di dati, inserendo in ascissa i dati passati attraverso l'effetto del rivelatore e in ordinata quelli generati inizialmente. In questo modo è stata ottenuta la tripletta di grafici necessaria per eseguire la deconvoluzione.

Non avendo dati reali su cui applicare quanto appena ottenuto, si genera un'ulteriore serie allo stesso modo di quanto fatto in precedenza, utilizzando però una funzione analitica differente. Per quanto accennato in precedenza, la funzione analitica che segue l'andamento dei dati deve essere simile a quella utilizzata nella fase di *train*: si è quindi deciso di generare dei dati con distribuzione gaussiana, centrata sempre nell'origine e con una deviazione standard pari a 2.0, compatibile con la scelta effettuata per la Breit-Wigner. Non disponendo della serie di dati passati attraverso il rivelatore, si è in primo luogo generato l'istogramma con distribuzione gaussiana (che corrisponde a quello che sarà il risultato finale della deconvoluzione) e, analogamente a quanto fatto in precedenza, questa distribuzione è stata "rovinata" con la stessa funzione di smussamento. Questo ovviamente perché l'effetto del

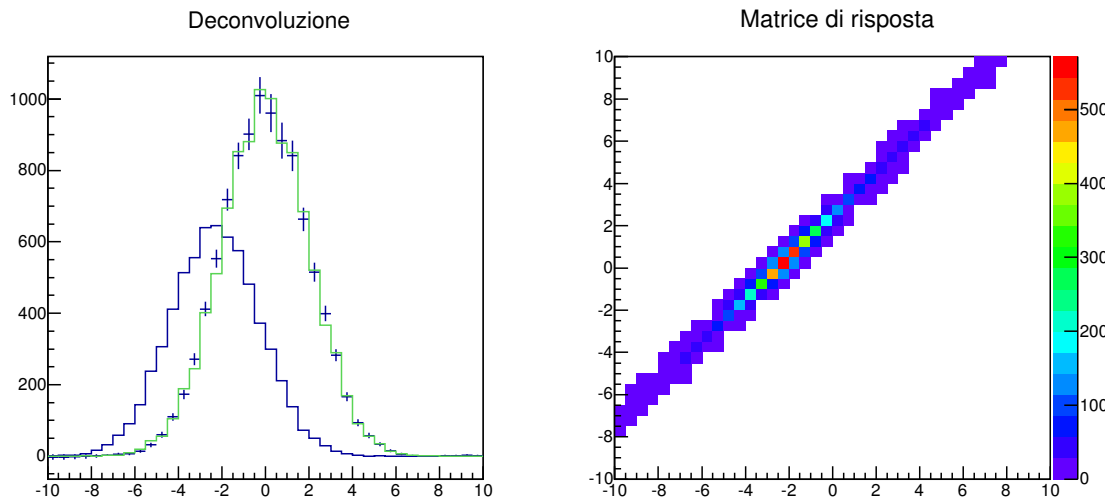


Figura 2.1: A sinistra un esempio di deconvoluzione (l’istogramma in verde contiene i dati “veri” che descrivono il fenomeno, l’istogramma in blu contiene i dati smussati dal rivelatore mentre i punti blu rappresentano i dati ricostruiti con la deconvoluzione), a destra la rispettiva matrice di risposta.

rivelatore (simulato appunto con la funzione di smussamento) deve essere lo stesso sia nel caso della simulazione che nel caso dei dati.

Nel caso della Breit-Wigner si sono generati dieci volte tanti eventi rispetto alla distribuzione “sperimentale” ( $10^5$  contro  $10^4$ ); questo per avere una maggiore statistica che ha permesso di determinare in maniera più accurata la matrice di correlazione. Questo modo di procedere rispecchia inoltre il caso realistico, in cui non si hanno a disposizione un numero grande a piacere di eventi. Per poter applicare la deconvoluzione, sempre in riferimento all’equazione (1.3), le dimensioni della matrice devono corrispondere al numero di intervalli dell’istogramma a cui andrà applicata. Nel caso in esame, dove sono state generate tutte le serie di dati, si è scelto di avere sempre istogrammi con 40 intervalli, in modo che ognuno di essi abbia una buona statistica.

Il risultato dell’operazione di deconvoluzione è riportato in Figura 2.1. Per questa simulazione è stato utilizzato l’algoritmo di deconvoluzione SVD ed è stato mantenuto il parametro di regolarizzazione predefinito, ossia la metà del numero di intervalli dell’istogramma (il variare del parametro verrà discusso in dettaglio nel paragrafo successivo). Nel grafico di sinistra sono riportati in verde la distribuzione iniziale generata per mezzo della gaussiana, in blu la stessa dopo aver passato i dati per la funzione di smussamento, mentre i punti blu (con le relative barre d’errore) corrispondono ai dati deconvoluti. Analizzando questi, si evince come l’algoritmo abbia dato un buon risultato: i dati ricostruiti risultano infatti compatibili con la distribuzione teorica iniziale e non presentano le oscillazioni anomale legate all’inversione della matrice. Nel grafico di destra è stata riportata la matrice di correlazione (ottenibile sia dalla costruzione iniziale che dalla routine che implementa l’algoritmo di SVD). Si vede come essa non sia diagonale: questo effetto è legato alla risoluzione dello strumento che ha fatto sì che alcuni dati migrassero da un intervallo a quelli adiacenti ad esso (aumentando la deviazione standard della gaussiana utilizzata nella funzione di smussamento si sarebbe ottenuto un risultato che si discostava an-

cor di più dalla diagonale). Il fatto inoltre che sia spostata rispetto alla diagonale principale è dovuto al valor medio della gaussiana che non è pari a zero.

## 2.2 Studio dei parametri di regolarizzazione

Come è stato detto nel primo capitolo, ognuno dei due algoritmi è legato ad un parametro di regolarizzazione che influenza il risultato della deconvoluzione. Differenti scelte portano infatti a risultati che non sempre sono compatibili.

### 2.2.1 Regolarizzazione per il metodo SVD

Per l'algoritmo di SVD si è dimostrato come una buona scelta del parametro di regolarizzazione sia legata alle componenti del vettore  $d$  il quale è a sua volta legato alla soluzione del problema di deconvoluzione dalla (1.10). Nel caso in cui si consideri il parametro di regolarizzazione  $\tau \neq 0$  si ottiene:

$$d_i^\tau = \frac{d_i s_i^2}{s_i^2 + \tau} \quad (2.2)$$

che ovviamente si riconduce a quanto ottenuto in precedenza nel caso  $\tau = 0$ . Data questa formula, si vede come sia possibile determinare il parametro di regolarizzazione dalla graficazione del parametro  $d$  in funzione di  $i$ . Dopo un certo valore, infatti, ci si aspetta che esso sia compatibile con lo zero e sia quindi ininfluenza alla fine dell'analisi: il sistema di equazioni linearmente indipendenti è infatti legato al numero di parametri diversi da 0.

Affrontando il problema in maniera più pratica, la scelta del parametro di regolarizzazione evidenzia quanto peso viene dato alla funzione teorica predetta con il Monte Carlo rispetto alla serie dei dati raccolta durante l'esperimento. Considerando infatti un parametro di regolarizzazione relativo al primo valore singolare della matrice  $S$  si ottiene come istogramma deconvoluto lo stesso istogramma che si è passato inizialmente con la simulazione (si da quindi affidabilità zero ai dati, ovviamente questo processo non è utile ai fini dell'analisi). Al contrario considerando il valore massimo del parametro di regolarizzazione, ossia, per come è stato definito  $\tau$ , il quadrato del valore singolare corrispondente all'ultimo elemento della matrice  $S$ , non si ha alcuna deconvoluzione. Riferirsi continuamente agli elementi diagonali della matrice  $S$  non è vantaggioso, pertanto si farà riferimento al parametro di regolarizzazione come l'indice  $k$  del valore  $s_k^2 = \tau$ ; in questo modo, ovviamente, il valore del parametro di regolarizzazione è compreso tra 1 e il numero di intervalli dell'istogramma.

Andando a considerare situazioni intermedie ai due casi sopracitati, si ha che dando troppo peso al campione generato con il Monte Carlo, nel caso in cui questo sia sensibilmente diverso dai dati, si ottiene una distribuzione con errori sistematici più grandi di quelli che sono in realtà. Al contrario, favorendo il campione dei dati (avendo quindi un parametro di regolarizzazione vicino al numero di intervalli), si ottengono degli errori statistici maggiori (il campione di dati avrà infatti meno eventi rispetto alla simulazione) che però possono essere valutati; il problema principale di questo metodo è che, dando troppo peso ai dati sperimentali, si rischia di considerare quelle che in realtà sono fluttuazioni statistiche come andamento effettivo della distribuzione, ottenendo quindi una descrizione errata del fenomeno.

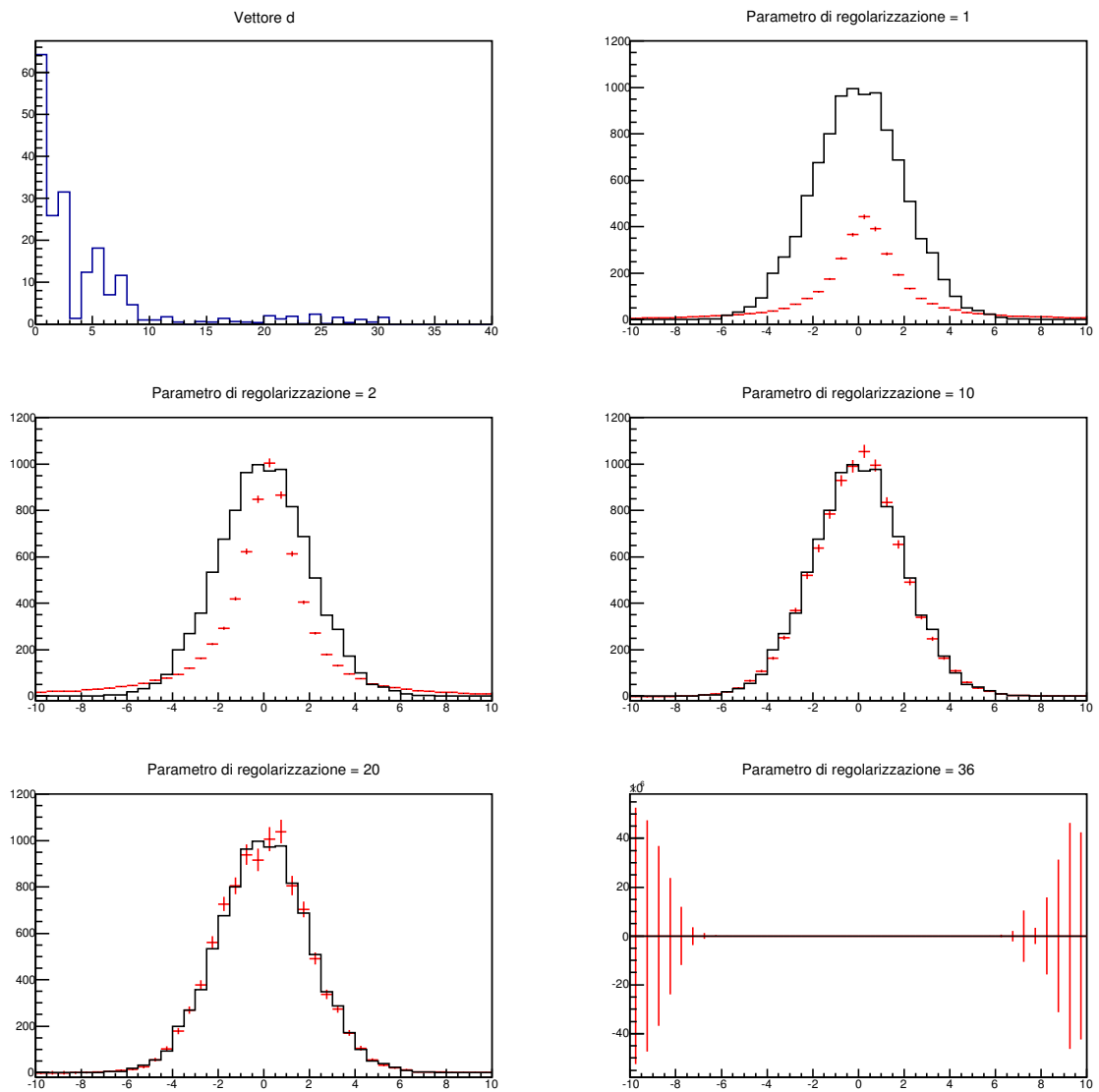


Figura 2.2: In alto a sinistra, grafico del vettore  $d$  in funzione dell'indice  $i$ . I grafici successivi rappresentano vari tentativi di deconvoluzione eseguiti con diversi valori del parametro di regolarizzazione (1, 2, 10, 20, 36); in nero è riportata la funzione "vera" che si vuole ottenere, mentre i dati in rosso rappresentano la ricostruzione ottenuta.

In Figura 2.2 è presentata un'applicazione di quanto discusso. In particolare, si è utilizzato lo stesso procedimento descritto nel paragrafo precedente, utilizzando la stessa funzione di *train* e generando i dati con la stessa gaussiana. Il primo grafico rappresenta il valore del vettore  $d$  in funzione dell'indice del valore singolare considerato. Per quanto detto in precedenza, vedendo che solo i primi 9 valori si discostano sensibilmente dall'unità, è opportuno scegliere come parametro di regolarizzazione un valore prossimo alla decina. I grafici successivi mostrano diverse deconvoluzioni, eseguite con gli stessi dati in entrata, ma variando il valore del parametro. Dalla prima deconvoluzione si nota come la scelta del parametro uguale a 1 non produca un risultato ottimale; la funzione rispecchia infatti completamente la forma della Breit-Wigner utilizzata nella fase di *train* e non è utile ai fini dell'analisi dei dati. Aumentando il valore anche solo a 2 si ottiene un risultato che rispecchia in maniera più accurata la forma dei dati sperimentali, anche se il risultato non è ancora soddisfacente. Utilizzando come parametro di regolarizzazione 10 si ottiene un ottimo risultato di deconvoluzione; questo fatto è in completo accordo con il grafico del vettore  $d$ : si è ottenuto un risultato soddisfacente prendendo come parametro di regolarizzazione l'indice di uno dei primi valori singolari compatibili con lo 0. Osservando i grafici successivi, si nota l'effetto descritto in precedenza: aumentando il valore del parametro di regolarizzazione si ottiene una ricostruzione dei dati a cui è associato un errore sempre maggiore, in accordo con quanto predetto teoricamente. Aumentando ancora il valore del parametro di regolarizzazione, avvicinandosi al numero di intervalli dell'istogramma, si ottiene una deconvoluzione che non rispecchia assolutamente la distribuzione iniziale dei dati, ma ha anzi un comportamento oscillante che è ordini di grandezza lontano dalla funzione cercata ed è compatibile con essa solo in linea teorica (avendo barre d'errore sproporzionatamente grandi) ma non fornisce alcuna descrizione del fenomeno.

## 2.2.2 Regolarizzazione per il metodo Bayes

L'analisi della regolarizzazione per la deconvoluzione basata sul teorema di Bayes procede in maniera completamente differente. Essendo infatti l'algoritmo di Bayes basato su un processo iterativo, il parametro di regolarizzazione indica in questo caso il numero di cicli necessari per ottenere un buon risultato di deconvoluzione. Lo scopo dell'algoritmo è infatti quello di ottenere una distribuzione deconvoluta che rispecchi al meglio la distribuzione dei dati sperimentali partendo da una distribuzione uniforme. Nel caso in esame, essendo nota a priori una distribuzione che assomiglia a quella dei dati, ossia la Breit-Wigner nei confronti dei dati gaussiani, si sceglie questa come funzione di partenza; ogni iterazione porterà la funzione deconvoluta ad assomigliare sempre meno a quella iniziale a favore di quella sperimentale.

Essendo il processo iterativo, è fondamentale imporre una condizione che evidenzii quando il risultato sia soddisfacente. Nel caso in esame è comodo effettuare un test di minimizzazione di una variabile  $\chi^2$  per stabilire la condizione di arresto (essendo nota a priori la distribuzione "vera" dei dati):

$$\chi^2 = \sum_{i=1}^{n_{\text{intervalli}}} \frac{(x_i - \hat{x}_i)^2}{\sigma_{\hat{x}_i}^2 + \sigma_{x_i}^2} \quad (2.3)$$

dove  $x_i$  è il valore dell' $i$ -esimo intervallo della distribuzione deconvoluta,  $\hat{x}_i$  è il valore dell' $i$ -esimo intervallo della distribuzione originale dei dati e  $\sigma_{\hat{x}_i}^2$  e  $\sigma_{x_i}^2$  sono



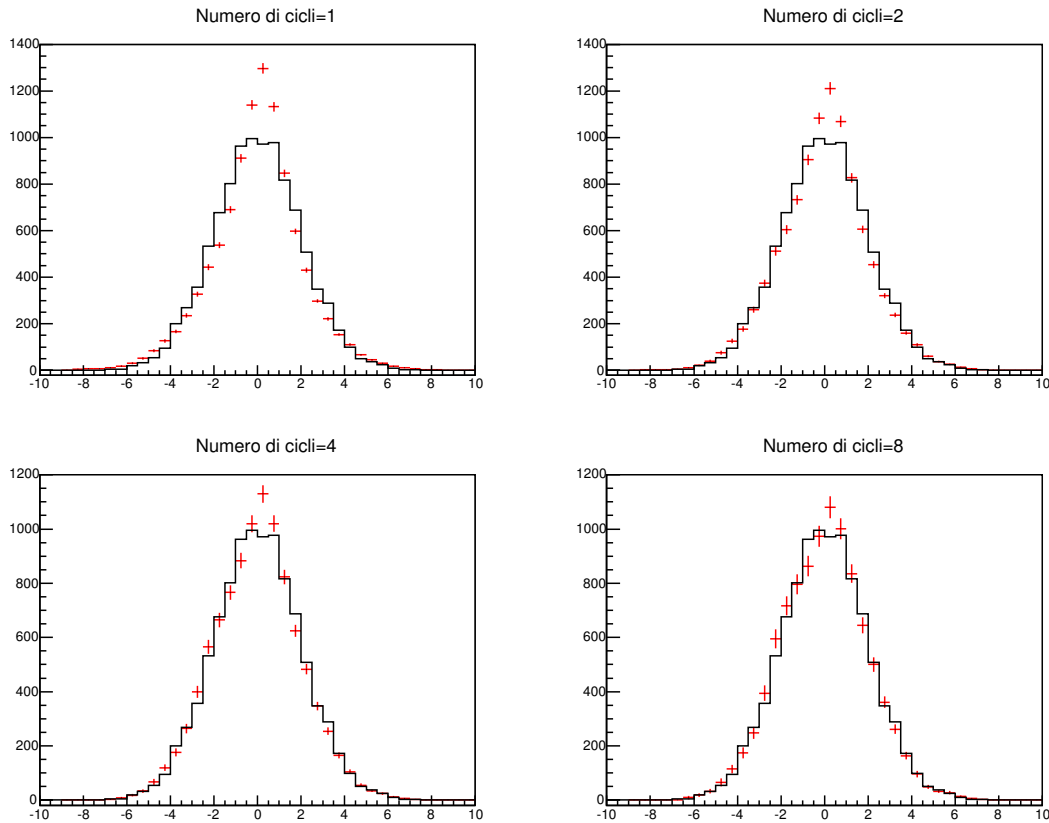


Figura 2.3: Esempi di deconvoluzione con il metodo di Bayes al variare del numero di cicli del programma (1, 2, 4, 8); in nero è riportata la funzione che si vuole ottenere (quella contenente i dati “veri”) mentre in rosso sono riportati i punti ricostruiti dalla deconvoluzione.

rispettivamente il quadrato dell’errore associato alla distribuzione originale e alla distribuzione deconvoluta. L’importanza di questa condizione di arresto dell’iterazione è duplice: da una parte permette di controllare il tempo di esecuzione del programma, dall’altra evita di ottenere una serie infinita di cicli nel caso in cui la soluzione continui ad oscillare attorno a quella desiderata senza mai raggiungerla.

In Figura 2.3 si riporta un esempio di applicazione di quanto appena spiegato. Nel caso specifico, sono riportati quattro differenti grafici, in ognuno dei quali è riportata la distribuzione “vera” dei dati (istogramma in nero) e la sua deconvoluzione con il metodo di Bayes (in rosso). Per ottenere un risultato in cui fosse apprezzabile il progresso fatto dall’algoritmo al variare del numero di cicli, è stata utilizzata una gaussiana nella funzione di smussamento con varianza superiore a quella utilizzata fino ad ora. Questo ovviamente ha portato ad avere una matrice di risposta meno diagonale rispetto ai casi precedenti (si veda la Figura 2.4). Come ci si aspettava, l’approssimazione migliora all’aumentare del numero di cicli: la funzione infatti passa dalla forma della Breit-Wigner data inizialmente a quella dei dati gaussiani. Dall’ultimo risultato, provando ad aumentare ulteriormente il numero di cicli, non si ottiene una soluzione migliore, anzi questa si stabilizza ad una configurazione definitiva che non varia più apprezzabilmente nemmeno aumentando di molto il numero di iterazioni. Si nota come inizialmente le barre d’errore aumentino all’aumentare del parametro di regolarizzazione, mentre per alti valori dello stesso non vi è più

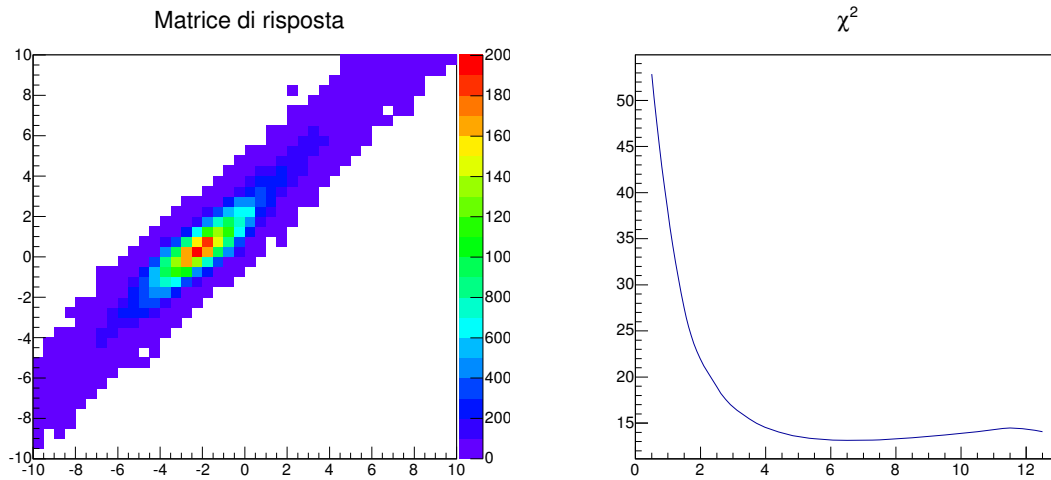


Figura 2.4: A sinistra la matrice di risposta ottenuta nel caso in esame, a destra l'andamento del valore del parametro  $\chi^2$  in funzione del numero di iterazioni.

una variazione apprezzabile (questo effetto sarà analizzato nel dettaglio in seguito).

Guardando ora la Figura 2.4, come accennato in precedenza, si nota come la matrice di risposta in questo caso risulti “meno diagonale”, ossia vi è più migrazione di dati da un intervallo ad un altro: questo appunto per la scelta fatta di avere una gaussiana di smussamento più larga rispetto al caso analizzato con l'algoritmo SVD. Nel secondo grafico riportato in Figura 2.4 è riportato il valore del parametro di  $\chi^2$  calcolato dalla (2.3) in funzione del numero di iterazioni dell'algoritmo. Come ci si aspettava (e come si può facilmente evincere dai grafici discussi in precedenza), l'approssimazione migliora di molto aumentando il numero di iterazioni. Tuttavia, si può vedere come l'andamento del parametro non sia sempre decrescente: si raggiunge un punto di minimo oltre il quale tende a stabilizzarsi. Ripetendo la deconvoluzione anche con un numero elevatissimo di iterazioni, si ottiene un valore di  $\chi^2$  che risulta comunque compatibile con quello del minimo, indicando che la soluzione si è stabilizzata.

## 2.3 Confronto SVD-Bayes

Per quanto visto fino a questo momento, i due metodi di analisi producono entrambi un risultato soddisfacente della deconvoluzione dei dati, anche se l'algoritmo che si trova alla base dei due presenta delle profonde differenze. Prima di queste è ovviamente l'iteratività dell'algoritmo di Bayes, contro l'analisi matriciale utilizzata per il metodo SVD.

In particolare, considerando sempre i dati gaussiani con *train* fatto per mezzo della Breit-Wigner, risulta utile verificare ancora una volta l'accordo tra la distribuzione deconvoluta e quella originale, utilizzando ad esempio il parametro definito in (2.3). Facendo questo test, ponendo i parametri di regolarizzazione ai valori ottimali (secondo quanto discusso nel paragrafo precedente) si ottiene sostanzialmente lo stesso accordo utilizzando i due metodi. Questo può essere spiegato per il metodo di Bayes dal fatto che la funzione iniziale sia “vicina” come forma a quella dei dati, mentre il buon risultato dell'algoritmo di SVD è legato alla presenza di un nu-

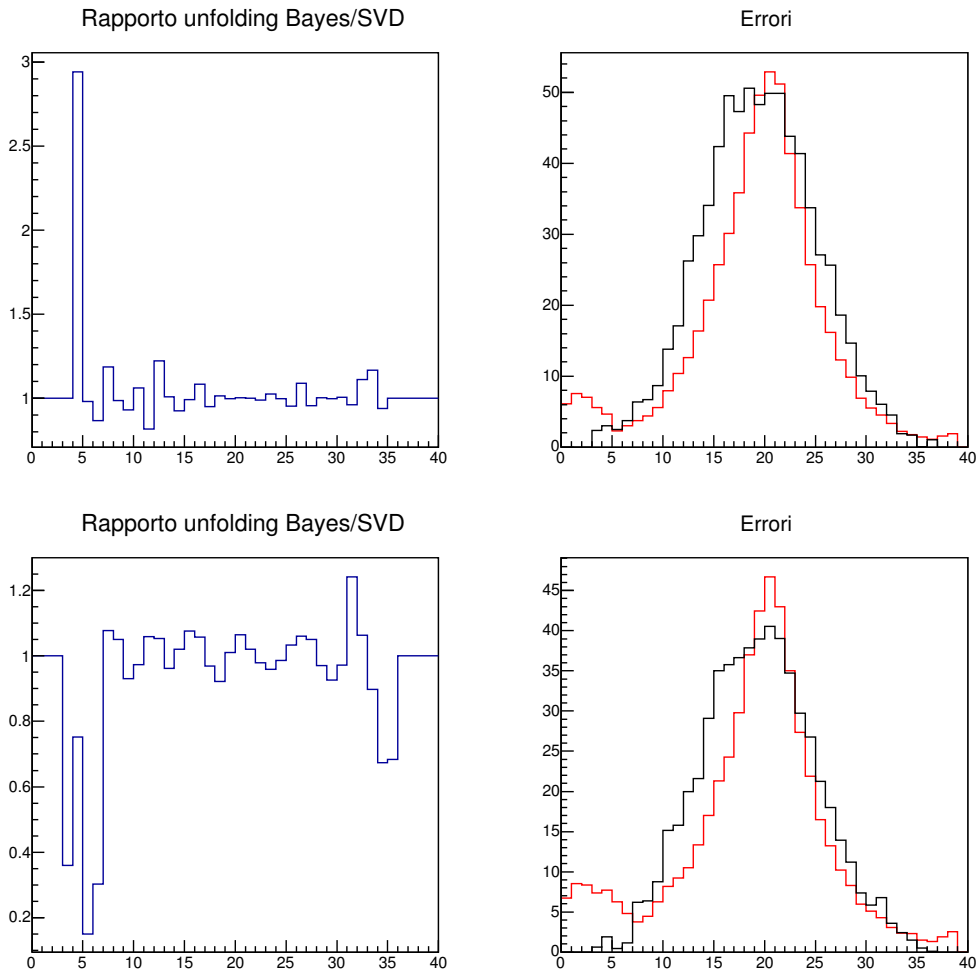


Figura 2.5: Rapporto dei risultati della deconvoluzione ottenuti con il metodo di Bayes e SVD (nei grafici a sinistra); nei grafici di destra vengono riportati in rosso gli errori associati al metodo SVD mentre in nero quelli relativi all’algoritmo di Bayes. I due grafici in alto fanno riferimento ad una simulazione con funzione di smussamento gaussiana con varianza 0.2, mentre in basso è stata fatta la stessa simulazione utilizzando però una gaussiana con varianza 0.8.

mero di equazioni linearmente indipendenti non troppo elevato e quindi facilmente risolvibile.

Per ottenere una comparazione migliore dell’efficacia dei due metodi è utile analizzare in dettaglio la deconvoluzione nei due casi per più funzioni. In Figura 2.5 viene riportato il rapporto tra la deconvoluzione fatta con Bayes e quella con SVD e i relativi errori associati. La funzione utilizzata per l’analisi nella figura in alto è quella descritta inizialmente, mentre quella in basso coincide con quella utilizzata nello studio della regolarizzazione con Bayes (la cui matrice di risposta è quella riportata in Figura 2.4 a sinistra). Come si può vedere, la deconvoluzione nei due casi ha prodotto praticamente lo stesso risultato: il rapporto tra le due distribuzioni si attesta infatti attorno al valore dell’unità. In particolare, fatta eccezione per il perfetto accordo ai bordi della distribuzione (dovuto al fatto che la deconvoluzione dava 0 con entrambi i metodi), vi sono solo pochi intervalli in cui l’accordo tra i due metodi non è rispettato. Questo avviene nel caso in cui si stiano ricostruendo inter-

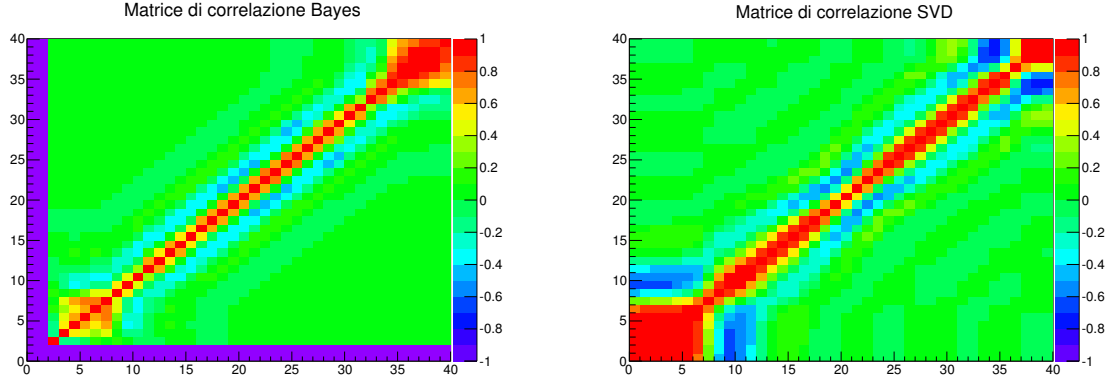


Figura 2.6: matrice di correlazione ottenuta tramite deconvoluzione con l’algoritmo Bayes (a sinistra) e SVD (a destra).

valli in cui il numero di dati è molto piccolo o addirittura 0 per cui è facile ottenere un rapporto tra le due ricostruzioni che devii sensibilmente dall’unità. Analizzando il parametro di equazione (2.3), si ottiene un risultato compatibile, anche se il valore ottenuto con il metodo SVD è leggermente inferiore, indicando un maggiore accordo tra la distribuzione ricostruita e quella originale.

Nei due grafici a destra di Figura 2.5 sono riportati gli errori associati alla ricostruzione con il metodo SVD (in rosso) e con il metodo di Bayes (in nero) per le due diverse funzioni. Sia nel primo caso che nel secondo, si nota come per i dati che corrispondono al picco gli errori associati al metodo SVD siano più piccoli rispetto a quelli dell’algoritmo di Bayes. Questo andamento risulta però invertito per i dati esterni al picco stesso, dove l’algoritmo di Bayes restituisce correttamente il valore di 0, mentre il metodo SVD associa agli intervalli con contenuto nullo un errore non nullo.

Uno dei principali effetti della deconvoluzione è quello di introdurre una correlazione non nulla tra intervalli vicini della distribuzione. Questo effetto è legato sia al parametro di regolarizzazione utilizzato che al metodo scelto per la deconvoluzione. In particolare, a valori ottimali del parametro di regolarizzazione (o a numero di cicli dell’algoritmo di Bayes che corrispondono al minimo del valore del parametro di  $\chi^2$ ) si ottiene la minor correlazione tra intervalli e quindi la miglior deconvoluzione in termini di errore associato ad ognuno di essi. Uno strumento utile per la verifica della dipendenza tra i vari intervalli è la matrice di correlazione:

$$V = \begin{pmatrix} 1 & \rho(x_1, x_2) & \rho(x_1, x_3) & \cdots & \rho(x_1, x_n) \\ \rho(x_2, x_1) & 1 & \rho(x_2, x_3) & \cdots & \vdots \\ \rho(x_3, x_1) & \rho(x_3, x_2) & 1 & \cdots & \vdots \\ \vdots & \vdots & \vdots & \ddots & \vdots \end{pmatrix}$$

che corrisponde alla matrice di covarianza normalizzata ai termini della diagonale.

Risulta a questo punto utile l’analisi della Figura 2.6 dove sono appunto riportate le matrici di correlazione per i due algoritmi. In entrambi i casi si nota una forte correlazione positiva per gli intervalli immediatamente adiacenti (che corrispondono nella matrice agli elementi adiacenti alla diagonale) tra i quali vi è quindi una sensibile migrazione nella ricostruzione dei dati. Per intervalli non adiacenti,

vi è correlazione nulla o leggera correlazione negativa. Questo andamento, che ben descrive gli intervalli relativi al picco, non può essere considerato valido per gli intervalli che contengono pochi o nessun dato (quelli ai margini della distribuzione). In questo caso, infatti, si evidenzia una forte correlazione anche con intervalli non adiacenti. In particolare nella ricostruzione effettuata con il metodo SVD; questo risulta in accordo con l'analisi dell'errore fatta in precedenza, dove si riscontrava (per SVD) un aumento delle barre d'errore negli intervalli "lontani" dal picco della funzione. Confrontando le due deconvoluzioni, si nota come in generale quella che utilizza l'algoritmo di Bayes produca meno correlazione nei dati deconvoluti (la zona viola al bordo è dovuta al fatto che in questo caso si sta dividendo per zero nella normalizzazione). Come accennato in precedenza, calcolando le matrici di correlazione nel caso in cui non si sia scelto il parametro di normalizzazione ottimale, si incorre in una maggiore correlazione tra i dati; nel caso estremo in cui la deconvoluzione non ottenga un risultato significativo, si ottiene una matrice completamente correlata sia positivamente che negativamente.

## 2.4 Metodo toy-1 per la stima degli errori e confronto con il metodo standard toy-0

L'errore associato ad ogni intervallo della distribuzione ricostruita con il metodo SVD è ottenuto dalla matrice di covarianza che, con la notazione utilizzata nel Capitolo 1, può essere calcolata come:

$$Z_{ik}^{(\tau)} = \frac{s_i^2}{(s_i^2 + \tau)^2} \delta_{ik} \quad \longrightarrow \quad W^{(\tau)} = C^{-1} V Z^{(\tau)} V^T C^{T-1} \quad (2.4)$$

la quale deve poi essere moltiplicata per i valori della distribuzione generata con il Monte Carlo (come fatto per la (1.10)).

Il calcolo dell'errore sui dati deconvoluti nell'algoritmo di Bayes viene effettuato seguendo il procedimento descritto nel primo capitolo, in particolare facendo uso dell'equazione (1.16). Questo procedimento, ricavato direttamente dall'analisi fatta in precedenza, risulta tuttavia fallace: il risultato da esso prodotto è valido solamente per la prima iterazione dell'algoritmo. L'errore di fondo, presente in [3] e corretto in [4], è quello di considerare, nella propagazione dell'errore, che la matrice  $M_{ij}$  sia indipendente dal campione di misure. Questo fatto è vero solamente per la prima iterazione (in cui si utilizza come informazione originale una distribuzione uniforme o come nel caso in esame la funzione del Monte Carlo), ma non per le successive: l'effetto è quello di una sottostima dell'errore. Tenendo conto di questo fatto, la propagazione dell'errore più corretta è la seguente:

$$\frac{\partial \hat{n}(C_i)}{\partial n(E_j)} = M_{ij} + \sum_{k=1}^{n_E} M_{ik} n(E_k) \left( \frac{1}{n_0(C_i)} \frac{\partial n_0(C_i)}{\partial n(E_j)} - \sum_{l=1}^{n_c} \frac{\epsilon_l}{n_0(C_l)} \frac{\partial n_0(C_l)}{\partial n(E_j)} M_{lk} \right) \quad (2.5)$$

da cui si vede che alla prima iterazione il termine tra parentesi è nullo, infatti la derivata del campione generato con il Monte Carlo rispetto alla distribuzione dei dati è nulla (le due distribuzioni sono indipendenti), mentre nei termini successivi si tiene conto del fatto che la distribuzione ottenuta nel passaggio precedente e utilizzata come base per la nuova iterazione ha una dipendenza dal campione in analisi. Dalla forma dell'equazione (2.5) si spiega inoltre il comportamento evidenziato

in precedenza per cui il valore dell'errore associato ai dati ricostruiti aumenta con l'aumentare del numero di iterazioni.

RooUnfold utilizza per il calcolo dell'errore un *toy Monte Carlo* implementato nella routine stessa. Il *toyMC* in questione altro non è che una simulazione eseguita utilizzando un Monte Carlo “semplificato” per descrivere la bontà di un algoritmo senza incorrere in tempi di calcolo eccessivamente lunghi. Nel caso in esame, esso è utilizzato per calcolare l'errore nei due casi utilizzando la (2.4) e la (2.5). Per come è implementato nel programma, è possibile inoltre selezionare il numero di simulazioni da utilizzare per questa valutazione: aumentando il valore delle stesse si ottiene un risultato sempre migliore, fino ad ottenere una stabilità sulla valutazione dell'errore che può essere considerata quella ottimale.

Per verificare l'accuratezza e il modo esatto in cui viene calcolato l'errore mediante questo metodo, si è deciso di implementare nel programma una simulazione “manuale” che riproducesse quanto fatto automaticamente dal *toyMC* stesso. L'idea alla base di questo procedimento è quella di generare una serie di distribuzioni di Monte Carlo ognuna delle quali produce una distribuzione deconvoluta: tramite la conoscenza di un elevato numero di queste, è possibile valutare l'errore da associare ad ogni intervallo ricostruito.

La parte chiave di questo metodo, è la generazione della serie di distribuzioni di Monte Carlo. Per costruire ognuna di esse, si è presa in considerazione la distribuzione iniziale ottenuta tramite il Monte Carlo con la Breit-Wigner (quella contenente la “verità”, ossia non ancora filtrata dalla funzione di smussamento) a cui si è applicato il metodo del rigetto (o di Von Neumann) [5]. Nel caso in esame, innanzitutto si è generata una variabile aleatoria con distribuzione uniforme compresa tra il minimo e il massimo in ascissa dell'istogramma; ottenuta questa, la si è associata all'intervallo corrispondente della distribuzione. Si è quindi generata un'ulteriore variabile casuale con distribuzione uniforme compresa tra 0 e il massimo dell'istogramma iniziale: se il valore ottenuto è inferiore rispetto al numero di eventi contenuti nel corrispondente intervallo della distribuzione iniziale, il dato viene accettato, altrimenti viene scartato. Questo procedimento viene ripetuto un numero di volte tale da ottenere una quantità di eventi nell'istogramma che coincida con quello dell'istogramma iniziale generato con il Monte Carlo. Per avere un'ulteriore casualità nella generazione della serie di istogrammi, si è deciso di generare una variabile con distribuzione gaussiana centrata sul numero di eventi dell'istogramma iniziale con varianza pari alla sua radice quadrata e utilizzare questo valore come condizione di arresto nella generazione dei dati. Le distribuzioni generate in questo modo seguono quindi la forma di quella iniziale, anche se non la rispecchiano perfettamente.

Ottenuto quindi il dato da associare all'intervallo, lo si è filtrato mediante la funzione di smussamento: in questo modo, ad ogni valore accettato per la distribuzione che sarà poi la “verità”, viene associato un valore che presenta gli effetti del rivelatore con cui si costruisce la distribuzione degli eventi misurati. Questa è la differenza principale tra il metodo implementato nel programma iniziale e quello appena descritto; il *toyMC*, infatti, genera prima la distribuzione con i dati veri, per poi utilizzare la conoscenza della distribuzione smussata ottenuta dal Monte Carlo iniziale per costruire la distribuzione dei dati misurati. In questo modo, tuttavia, i due istogrammi generati non sono completamente correlati (come invece dovrebbero essere, essendo legati dalla funzione di smussamento), ma generati indipendentemente a partire da due distribuzioni che sono correlate. È stato scelto di procedere

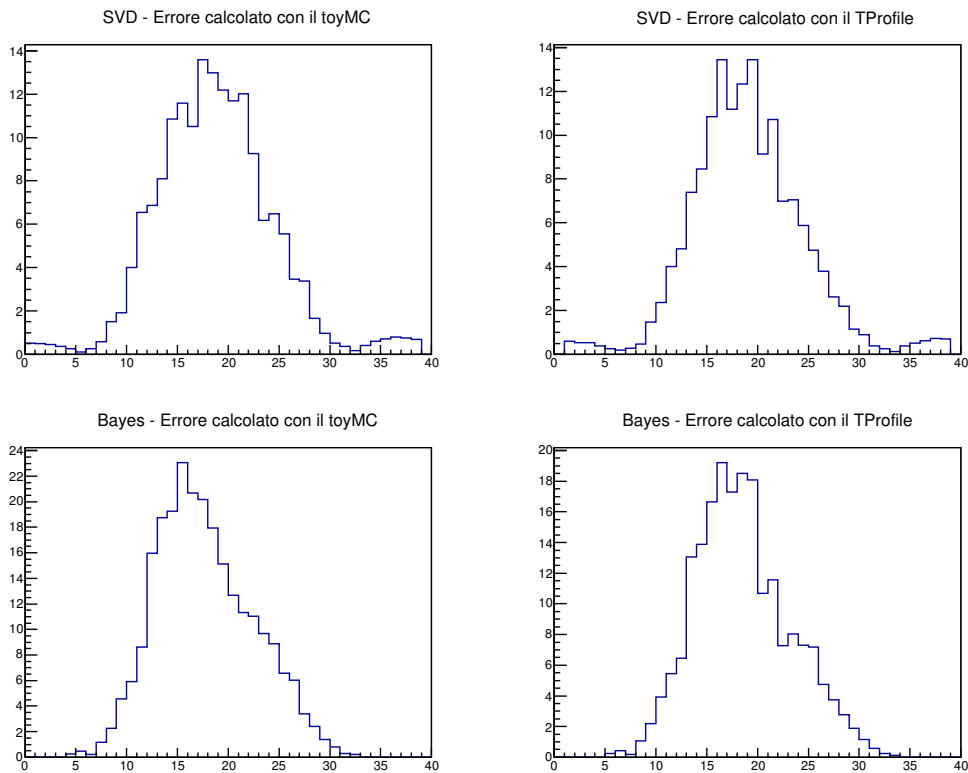


Figura 2.7: Nel grafico in alto sono riportati gli errori associati alla distribuzione deconvoluta con il metodo SVD: a destra quelli ottenuti con il toyMC predefinito del programma, a sinistra quelli ottenuti con il metodo che utilizza i cicli del programma descritto in precedenza. I grafici in basso riportano le stesse quantità ottenute però con la deconvoluzione per mezzo dell’algoritmo di Bayes.

in maniera differente per rispettare maggiormente questo fatto che rispecchia in maniera più fedele il reale processo avvenuto.

Avendo quindi la distribuzione originale e quella misurata, si è quindi in grado di ottenere la matrice di risposta che lega le due così da poter applicare la deconvoluzione ad ogni ciclo. Il numero di questi è stato scelto in modo da coincidere con quello del toyMC così da avere un raffronto diretto tra i due metodi di valutazione dell’errore.

Per tenere traccia della deconvoluzione di ogni ciclo, si è utilizzata una struttura di Root (il *TProfile*) la quale, dopo aver terminato il numero di cicli richiesto, contiene il valor medio del valore di ogni bin (nel caso in esame contiene quindi il valor medio di tutte le ricostruzioni) e l’errore ad esso associato, calcolato come la deviazione standard su ogni evento divisa per la radice quadrata del numero di eventi.

Anche in questo caso si è applicato quanto detto all’esempio consueto del Monte Carlo che segue la distribuzione di Breit-Wigner con dati gaussiani; il risultato è riportato in Figura 2.7. Per la simulazione si è scelto di utilizzare 50 cicli di simulazione poiché, per quanto detto prima, dopo un dato numero di essi il valore dell’errore si assesta (come osservato già in precedenza nello studio del parametro di regolarizzazione); nel caso in esame 50 cicli rispecchiano già questo comportamento. Si noti inoltre che sono riportati solamente gli errori legati alla

funzione deconvoluta e non quelli relativi alla distribuzione iniziale del Monte Carlo (in particolare l'errore relativo al metodo toyMC è la radice quadrata del valore corrispondente della diagonale della matrice di covarianza). Confrontando i risultati tra i due metodi, si evidenzia un sostanziale accordo. In particolare si ottiene una leggera sovrastima dell'errore con il toyMC rispetto al metodo implementato manualmente. Questo è legato a quanto evidenziato in precedenza, ossia che il metodo predefinito (il toyMC) applica la funzione di smussamento separatamente ai due istogrammi (dati "veri" e dati misurati) generati dal Monte Carlo iniziale. Ancora una volta gli errori calcolati con il metodo SVD sono inferiori a quelli ottenuti con l'algoritmo di Bayes (i quali sono destinati ad aumentare ad ogni iterazione per la (2.5)). La somiglianza dei risultati con i due metodi di rispecchia anche nelle "code" della distribuzione degli errori di SVD già evidenziate in precedenza e nella leggera asimmetria della distribuzione ottenuta con l'algoritmo di Bayes.

Si può quindi concludere che i due metodi, anche se leggermente differenti per il modo in cui vengono generate le distribuzioni, danno la stessa risposta in termini di valutazione dell'errore.



# Capitolo 3

## Una applicazione di deconvoluzione a dati reali

Quanto discusso fino ad ora è stato trattato in maniera puramente computazionale, applicando cioè l'algoritmo di deconvoluzione ad una simulazione numerica. Lo scopo della deconvoluzione, ovviamente, è l'applicazione a dati sperimentali. In quest'ultimo capitolo si utilizzeranno gli strumenti discussi fino a questo momento su alcune distribuzioni di misure di grandezze fisiche ottenute dall'esperimento CMS del Large Hadron Collider presso il CERN.

### 3.1 Descrizione delle osservabili studiate: analisi $Z+b(b)$

Prima di passare alla descrizione teorica del fenomeno, verranno brevemente descritti l'apparato sperimentale e il sistema di coordinate utilizzato.

L'esperimento CMS (*Compact Muon Solenoid*) è uno dei quattro esperimenti di LHC (*Large Hadron Collider*), il più grande acceleratore di particelle del mondo, situato al confine tra Svizzera e Francia presso il CERN (*Conseil Européen pour la Recherche Nucléaire*). LHC è un acceleratore a fasci incrociati di protoni, che può raggiungere un'energia nel centro di massa pari a 14 TeV (i dati che analizzeremo nel seguito si riferiscono alla prima fase dell'esperimento, in cui la massima energia raggiunta era pari a 7TeV). I due fasci, che procedono in verso opposto curvati da magneti superconduttori, vengono fatti collidere in quattro diversi punti dell'anello, in coincidenza con i quattro esperimenti principali. Lo scopo delle collisioni è quello di ottenere come prodotti particelle massive (e instabili) le quali, decadendo, provocano uno sciame di particelle secondarie: dall'analisi di questo sciame è possibile ricostruire le principali caratteristiche della particella originale.

La rivelazione delle particelle secondarie prodotte dal decadimento di una particella instabile, nel caso in esame sarà un bosone vettore  $Z$ , è uno degli obiettivi per cui è stato costruito CMS. L'esperimento è costituito da una serie di rivelatori, ognuno preposto alla misura di una data quantità fisica o ad un determinato tipo di particelle. In particolare, il primo rivelatore (quello più vicino al punto di collisione) è il *Tracker*, costituito da rivelatori al silicio, permette di ottenere la traiettoria delle particelle interagendo il meno possibile con esse. Esternamente a questo sono posizionati in sequenza il calorimetro elettronico e il calorimetro adronico: entrambi

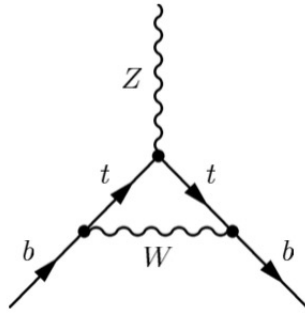


Figura 3.1: diagramma di Feynman di un possibile decadimento del bosone  $Z$  nella coppia  $b\bar{b}$ .

sfruttano l'interazione delle particelle con la materia fino a fermarle per misurare in questo modo la loro energia (il primo per fotoni, elettroni e positroni mentre il secondo per gli adroni). Nella parte più esterna è posizionato il magnete dell'esperimento e il rivelatore di muoni.

La struttura del rivelatore è progettata in modo da ottenere dati da particelle che provengono da ogni direzione dopo la collisione. Per questo motivo, oltre alla struttura precedentemente citata (denominata *barrel*), c'è anche una serie di rivelatori che "chiude" il cilindro principale da ambo le parti (*endcaps*); in questo modo si ottiene una copertura completa in tutte le direzioni di emissione dei fasci di particelle.

Per l'analisi dei dati successiva, è utile introdurre il sistema di coordinate utilizzato nell'esperimento. La terna di assi cartesiani a cui si fa riferimento è costruita in modo che l'asse  $z$  sia nella direzione tangente all'anello e sul piano dello stesso, l'asse  $x$  punti al centro del cerchio descritto dall'acceleratore e l'asse  $y$  sia perpendicolare a questi due. Data questa terna, è possibile definire alcune grandezze utili per l'analisi dei dati, in particolare:

$$\theta = \arctan\left(\frac{y}{x}\right) \quad \eta = -\ln\left(\tan\left(\frac{\theta}{2}\right)\right) \quad (3.1)$$

Il parametro  $\eta$ , detto *pseudorapidità*, è correlato quindi all'angolo di emissione delle particelle rispetto alla linea del fascio. Per come è costruito il rivelatore, il barrel copre un angolo  $\eta < 1.47$ , mentre gli endcaps coprono la regione  $1.47 < \eta < 3$ .

La collisione di protoni ad alte energie può essere vista come un'interazione tra quark e gluoni (insieme detti *partoni*) la quale produce una serie di altri gluoni e di coppie quark-antiquark. Essendo i quark dotati di carica di colore, non esistono in forma libera, ma si ricombinano per formare adroni oppure decadono in altre particelle. Un quark "libero" può emettere un bosone  $Z$  tramite un processo di corrente neutra.

Il bosone vettore  $Z$  è uno dei tre bosoni, insieme a  $W^\pm$ , responsabili dell'interazione debole. Avendo esso un tempo di decadimento dell'ordine di  $10^{-25}$  s è impossibile averne traccia direttamente; per effettuare una misura delle sue caratteristiche è quindi necessario analizzare i prodotti del suo decadimento. Essendo esso l'antiparticella di sé stesso, tutti i suoi numeri quantici sono pari a 0; questo fatto

limita i suoi decadimenti alla formazione di una particella assieme alla corrispettiva antiparticella. Uno dei modi in cui esso può decadere, è quello descritto in Figura 3.1 in cui viene prodotta una coppia quark anti-quark; nel caso specifico si analizza il caso di un quark  $b$  e un antiquark  $\bar{b}$ , ossia  $Z \rightarrow b\bar{b}$ .

Per quanto detto in precedenza, non è possibile avere un quark libero in natura: entrambi i prodotti del decadimento della  $Z$  decadono a loro volta, provocando due sciami adronici (*jet*). Fondamentale per l'analisi dati è la ricostruzione della posizione del vertice primario (punto in cui i due protoni hanno interagito) e quella del vertice secondario dove è avvenuto il decadimento in esame. Per fare questo, i rivelatori precedentemente descritti rivelano momento e direzione delle particelle del jet. Tramite un algoritmo di *b-tagging*, è possibile determinare la probabilità che il jet in esame sia stato prodotto da un quark  $b$  (e non da un quark leggero) e quindi ricostruire la posizione del vertice secondario.

Nel seguito saranno presi in considerazione i dati relativi al momento e al parametro  $\eta$  dei jet prodotti dal decadimento dei quark  $b$ .

## 3.2 Risultati di riferimento della deconvoluzione: regolarizzazione, SVD, Bayes

Quanto studiato in precedenza per una funzione analitica, verrà ora applicato al caso di dati sperimentali. In particolare, come descritto in precedenza, verranno analizzate le distribuzioni del momento e del parametro  $\eta$  relative ad un  $b - jet$  prodotto dal decadimento di un bosone  $Z$ . Lo scopo sarà quello di applicare le diverse opzioni descritte in precedenza (algoritmo SVD e Bayes, variazione dei parametri di regolarizzazione) per vedere il modo in cui varia la ricostruzione dei dati. In questo caso non si avrà più una funzione con un picco (come lo era la gaussiana utilizzata fino ad ora), ma una distribuzione decrescente (quella del momento) ed una distribuzione globalmente convessa (per la pseudorapidità), così da avere un'ulteriore validazione del metodo per distribuzioni sostanzialmente diverse.

A differenza di quanto fatto in precedenza, in questo caso gli istogrammi relativi alla fase di *train* con il Monte Carlo non sono stati generati all'interno del codice, ma sono stati letti dall'esterno. Il procedimento seguito è nelle sue altre parti uguale a quello riportato nel primo paragrafo del secondo capitolo. In Figura 3.2 sono riportati in alto i risultati della deconvoluzione per i dati sul momento ricostruito dei  $b-jet$  (con la relativa matrice di risposta), mentre in basso sono mostrati gli stessi grafici nel caso della ricostruzione della pseudorapidità. La deconvoluzione eseguita per mezzo del metodo SVD è riportata in rosso, mentre quella ottenuta mediante l'algoritmo di Bayes è rappresentata in verde.

In entrambi i casi si nota come la funzione deconvoluta contenga molti più eventi rispetto a quella originale in confronto a quanto accadeva durante la simulazione del capitolo precedente: questo porta a pensare che l'accettanza del rivelatore sia più limitata rispetto al caso simulato in precedenza. Per quanto riguarda la deconvoluzione in sé, si può vedere come i due metodi riproducano sostanzialmente lo stesso risultato per entrambe le distribuzioni. In particolare, nel caso della distribuzione del momento si ha che ogni punto ricostruito risulta compatibile, mentre per il caso della pseudorapidità questo non è sempre vero. In questo secondo caso, infatti, c'è discrepanza tra i due metodi in particolare nella regione del picco. Il metodo

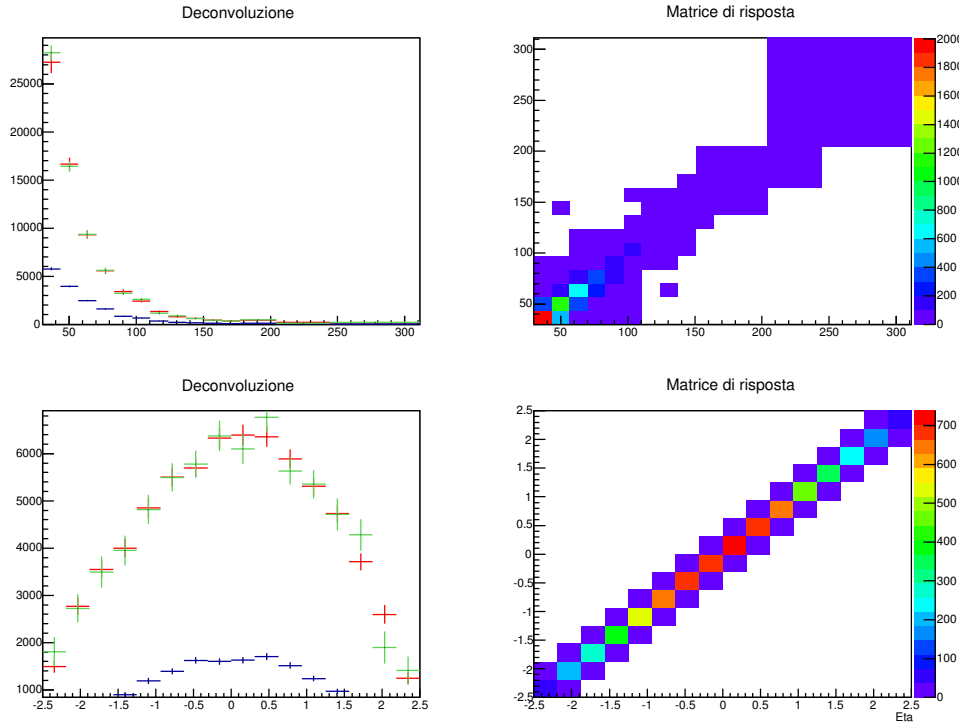


Figura 3.2: Deconvoluzione delle distribuzioni di momento (sopra) e pseudorapidità (sotto) con il metodo SVD (in rosso) e con l’algoritmo di Bayes (in verde); a destra le matrici di risposta per i due casi.

di Bayes, infatti, non riesce a descrivere in maniera appropriata l’andamento dei dati in questa regione, tenendo troppo in considerazione l’andamento leggermente decrescente presente nella serie raccolta sperimentalmente. Aumentando il numero di iterazioni, il risultato non migliora anche se, per l’effetto descritto in precedenza dell’aumento delle barre d’errore, i risultati risultano maggiormente compatibili.

Per lo studio del parametro di regolarizzazione si è deciso di utilizzare la distribuzione del momento. I risultati di questa operazione sono riportati in Figura 3.3. Come in precedenza, per il caso dell’algoritmo di SVD si è riportata prima la graficazione del vettore  $d$  in funzione dell’intervallo. Da questo grafico (in alto a sinistra) si nota come vi sia un solo valore del vettore sensibilmente diverso da 0: da questo deduciamo come sia sufficiente un parametro di regolarizzazione pari a 2 per ottenere un buon risultato della deconvoluzione. Come fatto in precedenza, tuttavia, si è preferito non considerare il primo valore ottimale, ma si è deciso di variare il parametro di regolarizzazione tra 3 (in rosso) e 8 (in verde); ci si è fermati a 8 in quanto con valori superiori (fino al massimo di 14, pari al numero di intervalli) si ottengono dei risultati peggiori, fino al fallimento della ricostruzione (come visto nel secondo capitolo). Dai casi analizzati, si vede come all’aumentare del valore del parametro non vi sia una sostanziale variazione della distribuzione ricostruita anche se la tendenza è quella di avere la funzione sempre con meno dati per intervallo. Questo è in accordo con quanto descritto in precedenza: aumentando il valore del parametro di regolarizzazione si tende a dare più valore alla funzione misurata (riportata in blu) rispetto a quella del Monte Carlo.

Nel grafico in basso della Figura 3.3 è riportata la deconvoluzione ottenuta con l’algoritmo di Bayes in funzione del variare del numero di cicli da 1 (rosso) a 4

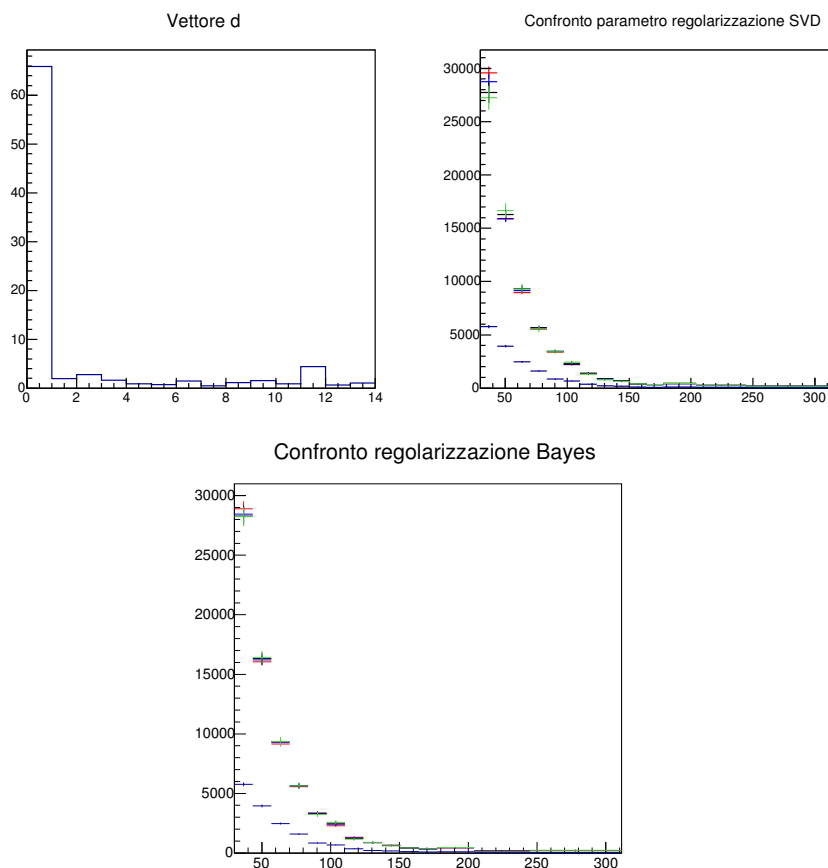


Figura 3.3: Il grafico in alto a sinistra riporta il vettore  $d$  in funzione dell'indice dell'intervallo  $i$ . In alto a destra è riportata la deconvoluzione dei dati sul momento ottenuta con il metodo SVD al variare del parametro di regolarizzazione. In basso è riportata la deconvoluzione della stessa distribuzione ottenuta con l'algoritmo di Bayes al variare del numero di cicli.

(verde). Da questo si può vedere come non vi sia una particolare variazione del risultato della deconvoluzione: in particolare, si nota la stessa tendenza descritta per il metodo SVD, ossia all'aumentare del numero di iterazioni la soluzione tende a decrescere.

### 3.3 Descrizione del metodo toy-2 e sua applicazione

L'analisi degli errori nel caso in esame presenta delle difficoltà aggiuntive non banali rispetto al caso analizzato in precedenza. Le difficoltà derivano principalmente da due fatti: non è nota la funzione analitica che regola il fenomeno e non è nota analiticamente la funzione di risposta del rivelatore. Volendo quindi procedere in maniera analoga a quanto descritto nel capitolo precedente, è quindi necessario ovviare a questi problemi per poter utilizzare il metodo dei cicli per simulare quanto fatto dal toyMC implementato nel programma stesso.

Il fatto di non conoscere la funzione analitica che descrive il fenomeno, non influisce nella costruzione del primo istogramma della terna, quello che contiene la

“verità” simulata con il Monte Carlo. Come detto in precedenza, infatti, per questa analisi si sono usati degli istogrammi precedentemente generati per la fase di *train*. Nota quindi la distribuzione dell’istogramma, si è proceduto in maniera analoga a quanto descritto nel paragrafo 2.4 per la generazione ad ogni ciclo di un istogramma di verità.

Il problema a questo punto, seguendo il procedimento utilizzato in precedenza, è quello di non poter generare un istogramma contenente i dati simulati, in quanto non è nota a priori la funzione di smussamento. Una possibile soluzione è quella di riprodurre una buona funzione di smussamento e quindi cercare di riottenere una distribuzione dei dati misurati che sia compatibile con quella fornita dalla tripletta esterna. Questo problema, tuttavia, è di difficile soluzione in quanto la funzione di risposta del rivelatore contiene al suo interno sia effetti di limitata risoluzione che effetti di limitata accettazione e procedendo nel modo appena descritto si dovrebbero controllare contemporaneamente entrambe le variabili.

Un modo più interessante (e più maneggevole) di affrontare il problema, è quello di tenere in considerazione che la matrice di risposta lega la distribuzione contenente la verità con quella dei dati: cercando quindi di riprodurla si riesce a individuare quale possa essere la funzione che regola la limitata risoluzione. A questo punto è sufficiente cercare una funzione che controlli l’accettazione in modo da riprodurre la distribuzione desiderata.

Dall’analisi della matrice stessa (riportata in Figura 3.2), si vede come sia nel caso del momento che nel caso della pseudorapidità ci sia forte correlazione sulla diagonale, la quale cala poi rapidamente negli intervalli successivi. In particolare, quella del parametro  $\eta$  ha elementi diversi da zero solo sulla diagonale e sugli elementi immediatamente adiacenti ad essa: una funzione gaussiana abbastanza piccata descrive bene questo andamento.

L’analisi della matrice di risposta relativamente al caso del momento, è più complicata. Per prima cosa si nota come vi sia un maggiore effetto del rivelatore, che si ripercuote in una maggiore correlazione di dati anche non appartenenti ad intervalli immediatamente adiacenti. Si nota, inoltre, che si ha una significativa correlazione solamente per i primi intervalli (in basso a sinistra nella matrice rappresentata). Un primo tentativo di stimare la funzione di risposta è stato fatto nuovamente con una gaussiana. Il risultato non è accettabile in quanto vi sono due tendenze opposte che dominano l’andamento della funzione: la necessità di avere forte correlazione per intervalli immediatamente adiacenti che però crolla bruscamente nei successivi e quella di avere una coda di correlazione che coinvolge anche intervalli relativamente distanti. Per ovviare a questo problema, si è tentata la somma di una funzione gaussiana e di una Breit-Wigner; l’aggiunta di questa (anche opportunamente pesata) introduce però degli effetti che non rispecchiano l’andamento cercato. In particolare si ottiene una correlazione eccessiva per intervalli non adiacenti. La miglior soluzione la si è ottenuta utilizzando la somma di due gaussiane. Una di queste descrive il picco, mentre la seconda, con varianza molto superiore alla prima, riproduce l’andamento degli intervalli lontani dalla diagonale della matrice.

In Figura 3.4 si riportano i risultati di quanto descritto: a sinistra la tripletta di istogrammi originale, a destra quella generata con il metodo appena descritto (si fa riferimento ad una di quelle utilizzate nel ciclo). Analizzando in prima istanza il grafico sulla verità, si vede come vi sia ottimo accordo con la distribuzione originale, anche se nell’ultimo intervallo non vi è nessun dato. Questo evento è compatibile

con il modo in cui è stato generato l'istogramma: la distribuzione iniziale contiene di per sé un numero molto piccolo di dati in questo intervallo ed è quindi comprensibile che non ne venga generato alcuno. Confrontando le due matrici di risposta, si nota come vi sia sostanziale accordo tra quella considerata inizialmente e quella generata, soprattutto per quanto riguarda i punti con maggiore correlazione. Si nota, tuttavia, come quella generata presenti minore correlazione tra gli intervalli lontani; questo non deve tuttavia screditare la bontà di quest'ultima in quanto la discrepanza si limita a pochi eventi. La regione vuota in alto a destra è legata all'intervallo che non contiene dati nell'istogramma di verità citato in precedenza. Analizzando le due distribuzioni misurate (al centro della figura) si nota come vi sia accordo sia nell'andamento della distribuzione che nel numero di eventi nei vari intervalli. Questo è stato ottenuto mediante una funzione che riproduceva l'accettanza del rivelatore inserita dopo aver calcolato la matrice.

Avendo quindi ottenuto un modo per generare la tripletta di grafici, è stato possibile riprodurre lo studio dell'errore come descritto nel precedente capitolo. I risultati di quanto effettuato per la distribuzione dei momenti sono riportati in Figura 3.5. In questo caso, a differenza di quanto fatto nel Capitolo 2, si è voluta analizzare l'affinità dei due metodi di valutazione dell'errore utilizzando 100 cicli sia del programma che del toy implementato. Come si evince dalla figura, c'è un ottimo accordo tra i due metodi. Il fatto di aver aumentato il numero di cicli è utile per fare un ulteriore "confronto a distanza" con quanto ottenuto in precedenza utilizzandone solamente 50. Facendo riferimento alla Figura 2.7 si vede come in questo caso l'accordo sia molto migliore. Facendo ulteriori test nel caso dei dati sperimentali, si è verificato che l'algoritmo interno al programma tende a stabilizzarsi ad una soluzione che l'algoritmo implementato "manualmente" raggiunge con un numero maggiore di iterazioni.

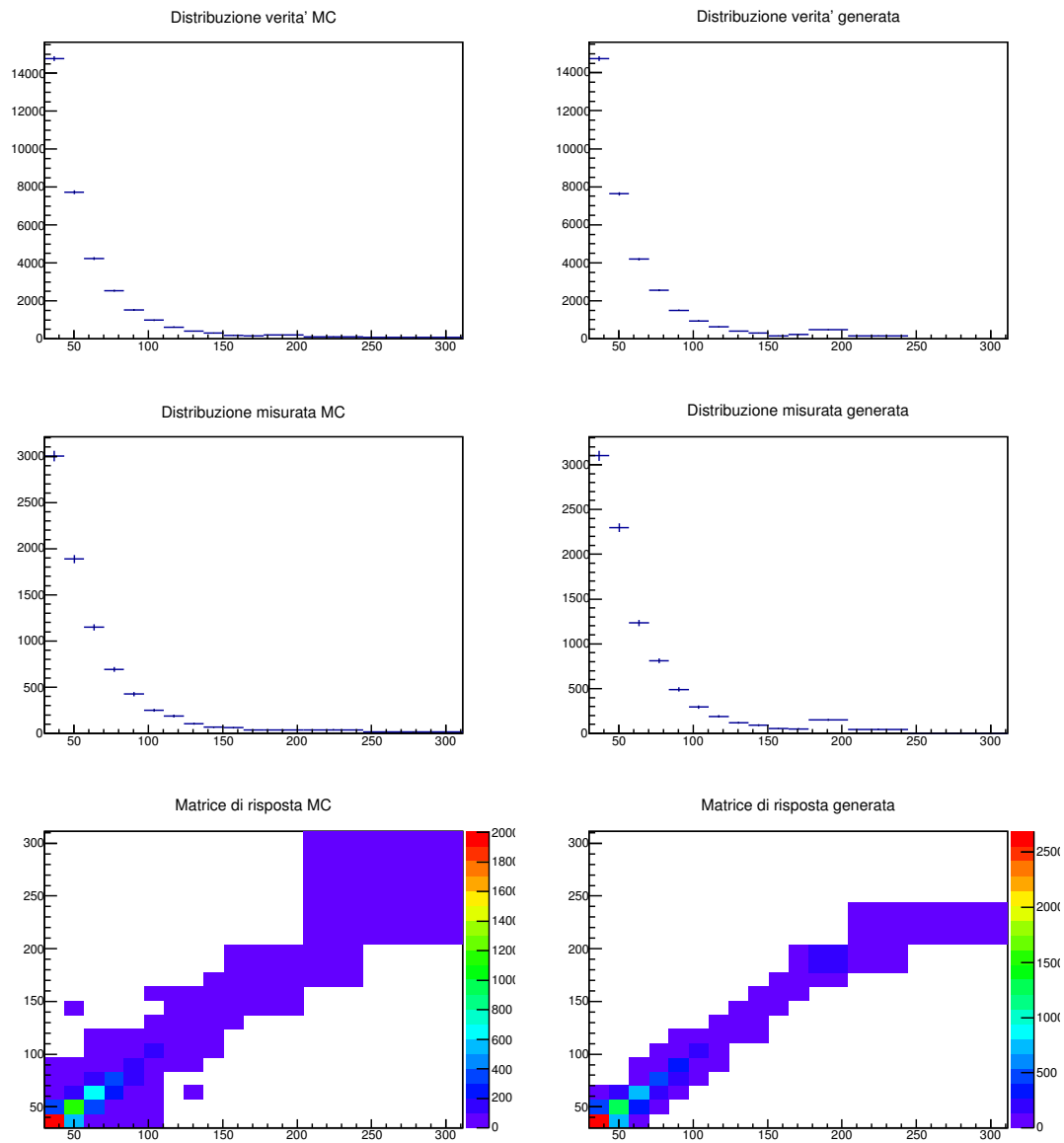


Figura 3.4: Nella colonna di sinistra sono riportati gli istogrammi letti esternamente dal programma, mentre in quella di destra quelli ottenuti con il metodo precedentemente descritto. Dall'alto in basso essi corrispondono all'istogramma della verità, a quello dei dati misurati e alla relativa matrice di risposta.



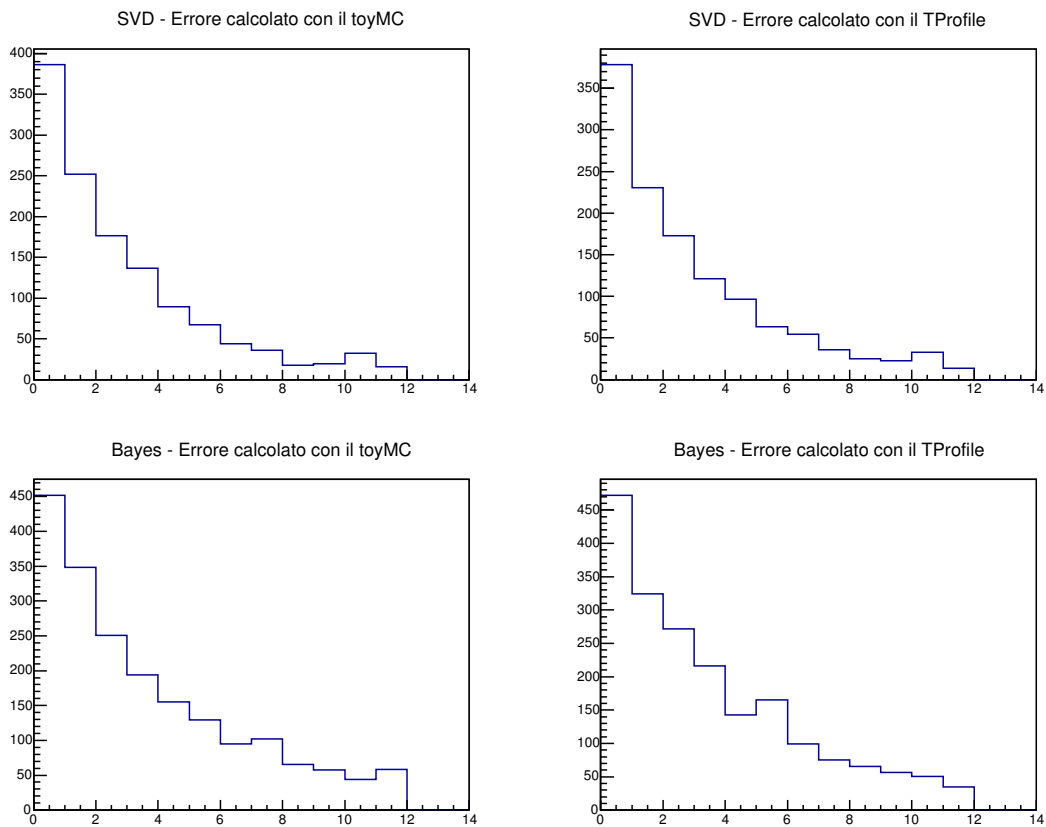


Figura 3.5: Valutazione dell'errore con il toyMC (a sinistra) e il procedimento a più cicli (a destra); in alto si riportano i grafici relativi alla deconvoluzione con il metodo SVD mentre in basso gli stessi ottenuti con l'algoritmo di Bayes.

# Conclusioni

Questo lavoro di tesi ha trattato il problema della deconvoluzione, con particolare attenzione ai problemi legati alla soluzione analitica, e ha studiato come risolverlo utilizzando i due metodi algoritmici SVD e Bayes.

In particolare, è stato implementato un programma di base, grazie al quale sono stati studiati i vari aspetti fondamentali della deconvoluzione. Tra questi, lo studio del parametro di regolarizzazione, che ha dimostrato come si possa ottenere una buona deconvoluzione regolando correttamente il valore di quest'ultimo, sia per l'algoritmo di SVD che per quello di Bayes. Si è inoltre fatto uno studio approfondito del metodo di calcolo dell'errore, implementandone una nuova versione che presenta una leggera differenza nell'approccio teorico, ma che ha riprodotto risultati compatibili con il metodo utilizzato di base dal programma. Sono stati inoltre comparati i risultati ottenuti per la stessa distribuzione dei dati con i due metodi (SVD e Bayes), evidenziandone il sostanziale accordo.

Si è quindi applicato quanto descritto nella fase precedente a delle distribuzioni di dati ottenute dall'esperimento CMS; in particolare alle distribuzioni legate al momento e alla pseudorapidità di un  $b$ -jet. È stata riproposta l'analisi eseguita per la funzione analitica, studiando quindi il parametro di regolarizzazione e confrontando i risultati ottenuti con i due algoritmi, evidenziandone ancora una volta l'accordo. Si è riproposta quindi l'analisi dell'errore, con le necessarie modifiche rispetto al caso precedente, ottenendo ancora una volta una stima in accordo con i risultati ottenuti dal metodo implementato inizialmente nel programma.

# Bibliografia

- [1] V. Blobel. Unfolding methods in high-energy physics experiments. *DESY 84-118*, 1984.
- [2] V. Kartvelishvili A. Hocker. Svd approach to data unfolding. *NIM A 372 (1996) 469*, 1996.
- [3] G. D'Agostini. A multidimensional unfolding method based on bayes' theorem. *Nuclear Instruments and Methods in Physics Research, A 362 (1995) 487-498*.
- [4] T. Adye. Unfolding algorithms and tests using roounfold. *arXiv:1105.1160v1*, 2011.
- [5] K. A. Olive et al. Review of Particle Physics. *Chin. Phys.*, C38:090001, 2014.