

Corso di Laurea triennale in Fisica (L-30)



UNIVERSITÀ DEGLI STUDI DI MILANO

**Stima delle proprietà di sistemi ottici nelle
microonde tramite l'utilizzo di reti neurali**

Relatore: Maurizio TOMASI

Correlatore: Cristian FRANCESCHET

Correlatrice: Sabrina REALINI

Tesi di Laurea di:
Eleonora GATTI
Matricola: 885664

Anno Accademico 2019/2020

Indice

1 Sistemi ottici nelle microonde	5
1.1 Utilizzo delle microonde in astrofisica	5
1.2 Diagramma di radiazione	6
1.3 Simulazione di sistemi ottici	8
2 Regressione con reti neurali	11
2.1 Machine learning e tipi di rete	11
2.2 Struttura di una rete neurale	11
3 Previsione delle proprietà di un diagramma di radiazione	13
3.1 Interpolazione	13
3.1.1 Interp2d	13
3.1.2 Curve Fit	14
3.1.3 Risultati dell'interpolazione	15
3.2 Reti neurali	16
3.2.1 Architettura della rete	16
3.2.2 Pre Training	19
3.2.3 Training	20
3.3 Confronto di risultati	20
4 Conclusioni	21

Capitolo 1

Sistemi ottici nelle microonde

Tutta l'informazione che abbiamo a disposizione su oggetti astronomici molto distanti è contenuta nella radiazione elettromagnetica emessa. Per secoli l'unico tipo di analisi possibile è stata quella nello spettro del visibile. Oggigiorno esistono svariate branche dell'astrofisica che analizzano segnali provenienti dall'Universo a diverse frequenze elettromagnetiche; una di queste branche riguarda lo studio del cosmo attraverso le microonde.

1.1 Utilizzo delle microonde in astrofisica

Il range delle lunghezze d'onda delle microonde è $\lambda \sim 1\text{mm} \div 10\text{cm}$ ¹.

È estremamente importante fare osservazioni a grandi lunghezze d'onda, nel millimetrico e oltre, poichè esistono numerosissime sorgenti cosmologiche che emettono in questo range e possono essere analizzate tramite sistemi ottici nelle microonde. I segnali più comunemente studiati in radio astronomia e attraverso le microonde sono:

- Radiazione emessa da gas ionizzato;
- Radiazione di sincrotrone, dovuta al moto di particelle cariche libere di muoversi nell'Universo e deviate da campi magnetici;
- Effetto Sunyaev-Zeldovich, che permette di rivelare ammassi di galassie altrimenti non visibili;
- Emissioni del Sole;
- Radiazioni da regioni H_{II} , si tratta di regioni nello spazio in cui sono presenti stelle molto calde (di tipo O o di tipo B) che ionizzano il gas intorno ad esse il quale emette nelle microonde e nel radio;
- Supernovae e resti di Supernovae²;

¹Il confine tra onde radio e microonde non è netto, spesso si parla di radio estendendo il range di lunghezze d'onda anche a quello delle microonde.

²La Crab Nebula, per esempio, è estremamente visibile nelle microonde.

- Pulsar;
- Radio galassie;
- CMB.

La **CMB**, Cosmic Microwave Background, è la radiazione a microonde di fondo cosmico che permea l'intero Universo. Secondo il *Modello Cosmologico Standard (SCM)* l'Universo ha avuto origine circa 14 miliardi di anni fa da una singolarità iniziale: il *Big Bang*.

Nei primissimi istanti dopo il *Big Bang* vi fu una rapidissima fase di espansione (10^{-33} s), detta *inflazione*, seguita da un'espansione più lenta e regolare, che continua tutt'ora. Inizialmente materia e radiazione erano in equilibrio in un plasma estremamente caldo; la progressiva espansione ha causato un abbassamento della temperatura del plasma. L'equilibrio tra materia e radiazione venne a mancare quando la temperatura raggiunse $T \simeq 3000$ K. Questo causò il *disaccoppiamento* tra materia e radiazione: la maggior parte degli elettroni venne catturata dai nuclei atomici permettendo la formazione dei primi atomi neutri. Convenzionalmente si considera come tempo al quale si è verificato il *disaccoppiamento* tra materia e radiazione l'istante t_{dec} in cui il libero cammino medio dei fotoni divenne maggiore della scala dell'Universo osservabile. Questo accadde a circa 380 000 anni dal *Big Bang* e da allora i fotoni primordiali sono liberi di vagare nello spazio e sono i responsabili della radiazione a microonde di fondo cosmico.

Data la vastità di campi in cui può essere effettuato uno studio nelle microonde, è di fondamentale importanza avere a disposizione un sistema ottico che permetta questo tipo di analisi. In particolare è necessario riconoscere la direzione dalla quale un determinato segnale microonde proviene.

1.2 Diagramma di radiazione

Un sistema ottico è un dispositivo il cui scopo è quello di registrare la luce proveniente dal cielo e mandarla ad un rivelatore. La risposta di un sistema ottico ideale può essere rappresentata come una delta di Dirac: non nulla solo lungo la linea di vista. Tuttavia i fenomeni di interferenza e diffrazione rendono la situazione molto più complessa; in particolare nella radio astronomia e nell'astronomia a microonde il problema è particolarmente importante poichè le dimensioni degli elementi ottici degli strumenti sono comparabili alle lunghezze d'onda d'interesse.

La risposta angolare di un sistema ottico è quantificata da una funzione $\gamma(\theta, \phi)$ detta *beam function* che definisce il **diagramma di radiazione**. Idealmente $\gamma(\theta, \phi)$ dovrebbe corrispondere a una delta di Dirac³, quello che viene in realtà osservato è una risposta simile a quella riportata in Fig. 1.1. È possibile osservare la presenza di un *main beam* e di lobi secondari. Tipicamente la maggior parte della radiazione

³Questa idealizzazione viene spesso chiamata *pencil beam idealization*

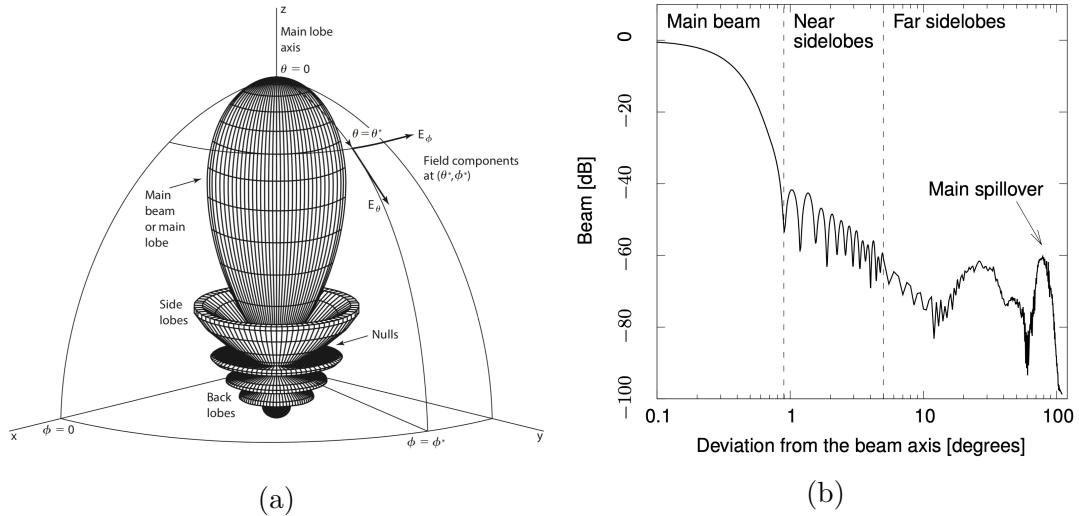


Figura 1.1: Tipico andamento della *beam function* $\gamma(\theta, \phi)$. La Fig. 1.1a rappresenta il diagramma di radiazione tridimensionale di un'antenna direzionale. [5]. La Fig. 1.1b rappresenta una sezione dello stesso grafico su un piano parallelo all'asse del main lobe. [2]

è contenuta nel main beam.

A partire dal diagramma di radiazione si definiscono alcuni parametri che permettono una sua descrizione; tali parametri sono i protagonisti di questo lavoro di tesi. La **Full Width Half Maximum** (FWHM) del main beam è la larghezza angolare a metà della sua altezza ed è uno dei parametri più importanti e più diffusi per la caratterizzazione della risoluzione di uno strumento. Nel corso dei prossimi capitoli verranno considerati due diversi valori per la FWHM, rispetto all'asse x e rispetto all'asse y . Attraverso queste due grandezze è possibile definire un ulteriore parametro: l'**ellitticità**. Questa è definita come il rapporto tra le due FWHM ponendo al numeratore la più grande tra FWHM_x e FWHM_y .

Hanno inoltre grande importanza i parametri che riguardano la polarizzazione. Supponiamo di studiare un'antenna in trasmissione⁴ e considerare una radiazione polarizzata. Fissiamo una direzione di polarizzazione; è allora possibile definire, rispetto a tale direzione, una componente *co-polare* ed una componente *cross-polare* della radiazione. La componente co-polare è data dalla frazione di radiazione con polarizzazione parallela alla direzione fissata mentre la componente cross-polare è data dalla frazione di radiazione con polarizzazione perpendicolare a quella fissata. In particolare i parametri utilizzati nei prossimi capitoli riguardano la componente **co-polare massima** e la componente **cross-polare massima**.

⁴Strumenti ottici per lo studio della CMB lavorano in ricezione ma è possibile studiare alternativamente un'antenna in trasmissione per il *principio di reciprocità*.

1.3 Simulazione di sistemi ottici

Simulare un sistema ottico significa studiare quanta potenza viene ricevuta in funzione dell'angolo. Tipicamente per simulare la risposta angolare di un sistema ottico si usa il software *GRASP* (General Reflector Antenna Software Package). Attraverso GRASP è stato possibile ottenere la tabella in Fig. 1.2 che riporta le caratteristiche del beam data una posizione (x, y, z) sulla superficie focale.

id	x	y	z	fw hm_x	fw hm_y	e	co_max	cx_max	
0	21	-0.316965	0.326756	50	0.398546	0.365183	1.091361	53.84	13,1
1	42	-0.318192	0.272044	50	0.364137	0.398634	1.094738	53.86	12,79
2	63	-0.319138	0.217441	50	0.364608	0.396464	1.087371	53.88	12,11
3	83	-0.320197	0.163140	45	0.361428	0.384678	1.064326	54.07	12,21
4	103	-0.321053	0.108822	40	0.358714	0.376443	1.049424	54.20	11,6
...	
164	3464	0.329983	-0.108774	45	0.351460	0.385294	1.096266	54.17	12,91
165	3486	0.330462	-0.163029	50	0.351985	0.395278	1.122996	54.04	13,01
166	3507	0.331717	-0.217468	50	0.351115	0.398046	1.133661	54.02	12,87
167	3528	0.333326	-0.271998	50	0.350770	0.401112	1.143520	53.99	13,38
168	3549	0.335293	-0.326679	50	0.350950	0.404582	1.152821	53.95	13,95

Figura 1.2: Dataset relativo allo strumento STRIP ottenuto tramite la simulazione in GRASP. Tale dataset è stato utilizzato per l'analisi descritta nei capitoli successivi.

GRASP è uno strumento molto potente che permette di simulare sistemi ottici complessi. Tuttavia i tempi di calcolo di GRASP sono elevati⁵; per ogni simulazione, e quindi per ogni antenna, vengono costruiti dei grafici come quelli riportati in Fig. 1.3 e poi da quelli vengono ricavati i valori riportati nel dataset 1.2.

Nel caso di STRIP è ancora possibile simulare l'intera ottica tramite GRASP poichè il numero di antenne è piuttosto limitato. Tuttavia per sistemi ottici più complessi in cui il numero di antenne diventa di circa 2 ordini di grandezza superiore, risulta del tutto impossibile effettuare una simulazione completa dell'intera ottica. È quindi nata la necessità di trovare una via alternativa che permetta di stimare i parametri che descrivono il beam in una qualsiasi posizione.

⁵Per produrre i dati in Fig. 1.2 sono stati impiegati due giorni.

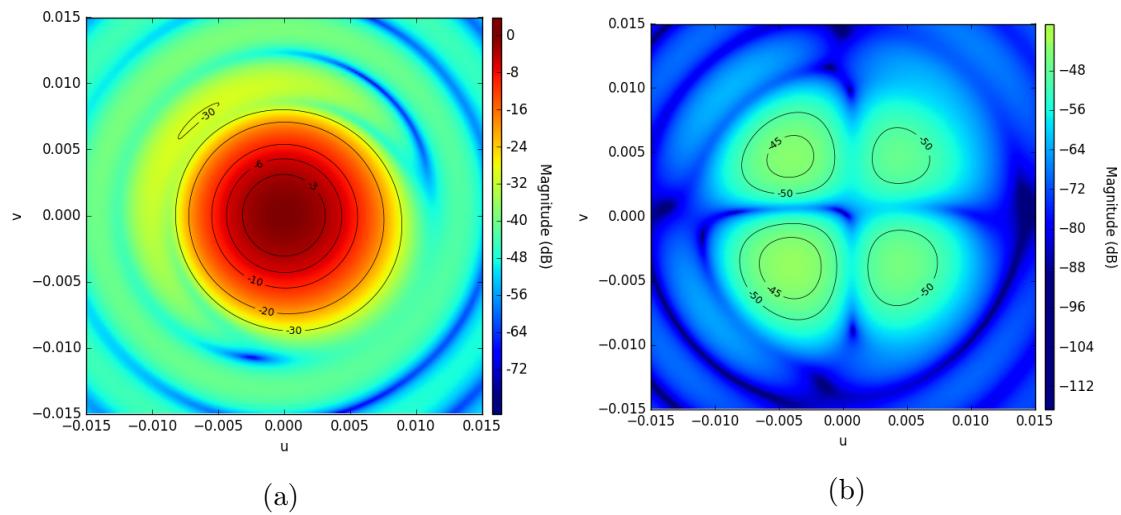


Figura 1.3: Diagramma della componente co-polare 1.3a e della componente cross-polare 1.3b di un'antenna off-axis. È possibile notare come il profilo del beam sia leggermente ellittico; per il sistema ottico di STRIP infatti l'ellitticità del fascio aumenta spostandosi dal centro del piano focale.

Capitolo 2

Regressione con reti neurali

2.1 Machine learning e tipi di rete

Negli ultimi decenni il campo del *machine learning* si è fortemente evoluto. È stata costruita un'ampia classe di algoritmi in grado di approssimare molto efficacemente processi non lineari. Tali architetture fanno parte di un particolare campo di ricerca, il *Deep Learning*, che vede protagoniste le reti neurali artificiali.

Queste tecniche si basano su l'apprendimento tramite esempi, i quali sono rappresentati da coppie input-output come un'immagine e la sua descrizione (input: foto di un gatto, output: "gatto") o una posizione a cui è associato un particolare valore di campo elettrico (input: (x, y, z) , output: $E(x, y, z)$). È quindi di fondamentale importanza avere a disposizione un ampio database attraverso il quale effettuare un *training*. Le reti neurali consentono quindi di approssimare una corrispondenza, vera o presunta, tra un input e un output.

Esistono due classi di problemi affrontati con la tecnica delle reti neurali: la *classificazione* e la *regressione*. La *classificazione* individua l'appartenenza ad una classe e può essere supervisionata o non supervisionata. Parliamo di classificazione supervisionata quando sono note a priori le diverse classi di appartenenza; se invece si vogliono determinare delle classi di similitudine senza conoscere a priori i pattern rappresentativi, si ha un problema di classificazione non supervisionata.

Le reti neurali per la *regressione* entrano in gioco quando, a partire da coppie input-output, si vuole determinare la funzione che approssimi al meglio la relazione.

2.2 Struttura di una rete neurale

L'idea utopica su cui si fondano le reti neurali artificiali è quella di simulare il comportamento del cervello umano. Questo è un sistema estremamente complesso basato sull'interconnessioni di unità fondamentali: i **neuroni**.

Una rete neurale può essere schematizzata come in figura 2.1. Un insieme di neuroni che agiscono allo stesso livello è detto **layer** e l'interconnessione di diversi layers forma una rete.

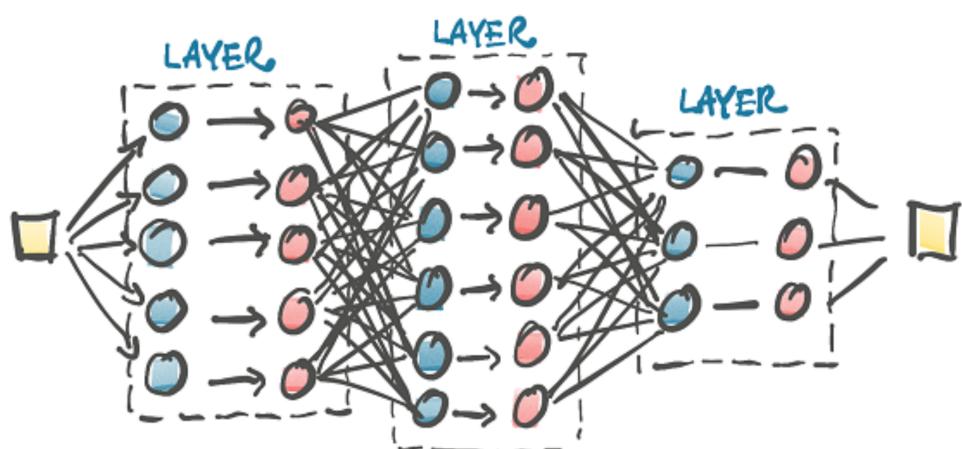


Figura 2.1: Schema di una rete neurale con tre hidden layers [4]

Capitolo 3

Previsione delle proprietà di un diagramma di radiazione

In questo capitolo varrà analizzato come sono stati previsti i parametri del diagramma di radiazione tramite i metodi di interpolazione e tramite l'utilizzo di reti neurali. Nota: nei seguenti paragrafi verranno mostrati alcuni grafici rappresentativi dei risultati ottenuti. L'analisi è stata effettuata su ogni parametro di interesse ma i risultati qui riportati riguardano esclusivamente l'ellitticità. I risultati sono infatti molto simili al variare del parametro e quelli riportati sono grafici che hanno carattere generale. I parametri analizzati sono: ellitticità, FWHM (rispetto a x), FWHM (rispetto a y), componente co-polare massima e componente cross-polare massima.

3.1 Interpolazione

Per stimare le proprietà di un diagramma di radiazione attraverso strumenti classici di interpolazione ho utilizzato due metodi: `interp2d` del modulo `scipy.interpolate`, basato su un'interpolazione lineare, e `curve fit` del modulo `scipy.optimize`, che ha permesso di fissare i dati attraverso un paraboloide. Per poter effettuare l'interpolazione dei parametri e in seguito verificare la sua bontà, il dataset in fig. 1.2 è stato suddiviso in due subsets, considerando righe alterne, che ho chiamato `data_int` e `data_check`. Così facendo è stato possibile utilizzare metà dei dati per l'interpolazione e l'altra metà per la valutazione dell'errore tra il parametro stimato e quello esatto.

3.1.1 Interp2d

Il metodo `interp2d` effettua l'interpolazione a partire da una griglia bidimensionale¹, nel mio caso questa è rappresentata dalle coppie (x, y) appartenenti al subset `data_int`. Una volta specificato il parametro di interesse, l'algoritmo effettua un'interpolazione lineare e ritorna una funzione in grado di prevedere il valore del parametro in nuovi punti.

¹La griglia non deve essere necessariamente regolare.

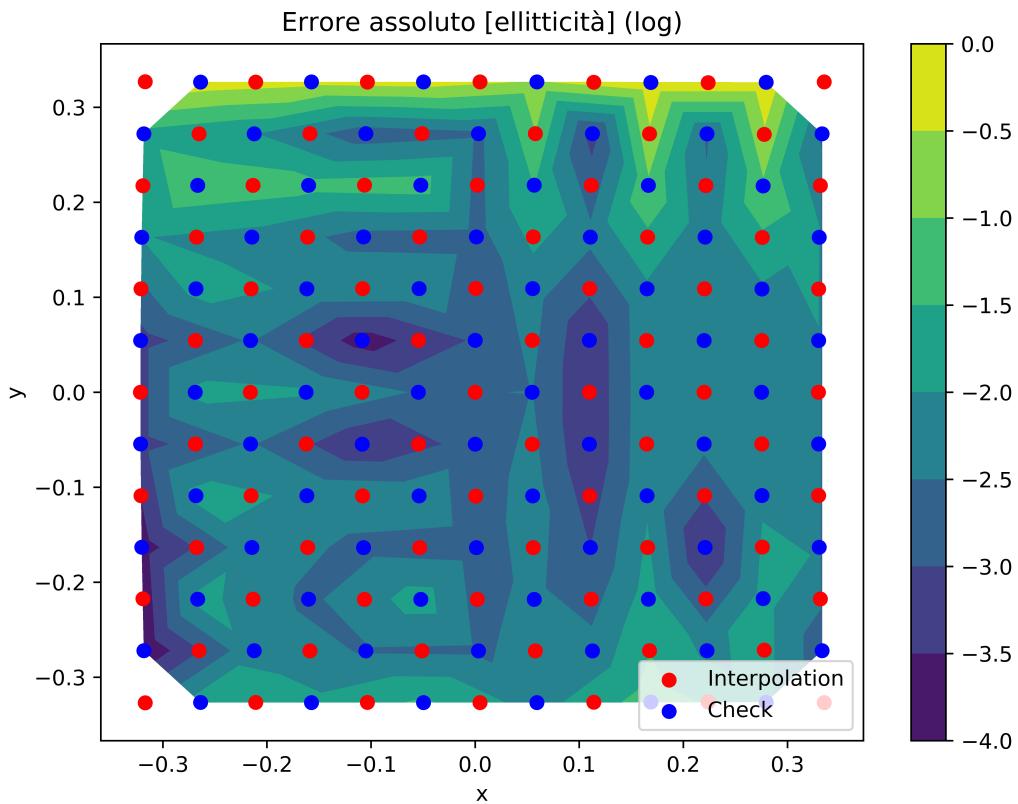


Figura 3.1: Contour plot in scala logaritmica dell’errore assoluto tra ellitticità interpolata tramite il metodo `interp2d` e ellitticità corretta. In rosso sono evidenziati i punti attraverso i quali è stata effettuata l’interpolazione mentre in blu sono rappresentati i punti nei quali è stato valutato l’errore.

La successiva analisi ha riguardato la valutazione dell’errore tra il parametro stimato nei punti appartenenti a `data_check` e il suo valore vero. In fig. 3.1 è mostrato l’errore assoluto relativo all’ellitticità. Si nota che i punti della prima riga a partire dall’alto sono quelli affetti da errore massimo e, come verrà specificato nella sez. 3.1.3, tali anomalie di bordo hanno determinato la scelta del metodo `curve fit` come termine di paragone con le reti neurali.

3.1.2 Curve Fit

Il metodo `curve_fit` permette di fissare i dati con una funzione non lineare tramite il metodo dei minimi quadrati. Il metodo restituisce i valori ottimali dei parametri `popt` che minimizzano la discrepanza $f(xdata, *popt) - ydata$ ². Anche in questo caso ho utilizzato i due subset, `data_int` e `data_check`, per effettuare rispettivamente la regressione e la valutazione dell’errore. Nel mio caso il fit è stato eseguito su

²In particolare `xdata` è una coppia (x, y) e `ydata` è il valore del parametro di interesse.

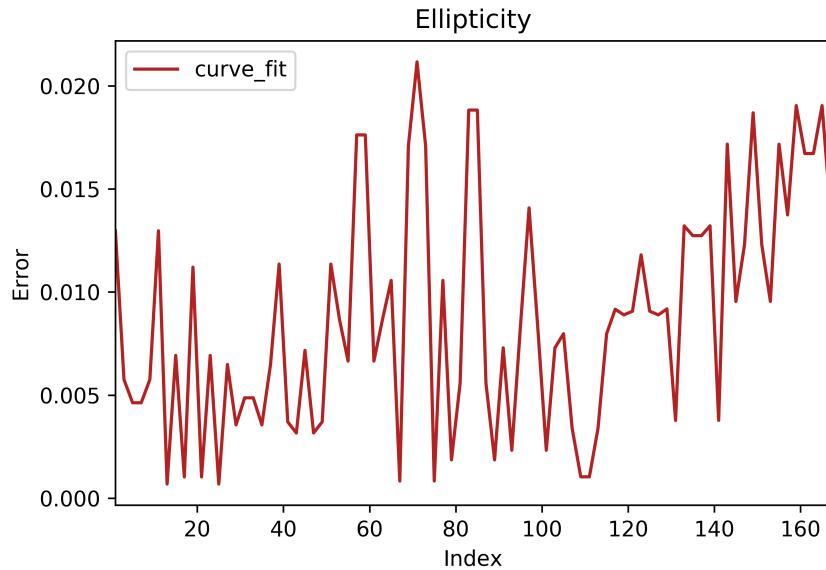


Figura 3.2: Plot dell'errore assoluto tra ellitticità interpolata tramite il metodo `curve_fit` e ellitticità corretta.

un paraboloide in quanto questo richiama la forma della superficie focale.

3.1.3 Risultati dell'interpolazione

In fig. 3.3 sono confrontate le curve che rappresentano l'errore tra dato interpolato e dato corretto per i due metodi utilizzati. Si nota immediatamente la presenza di alcuni punti problematici riguardanti `interp2d` che non rendono possibile un confronto tra le due curve.

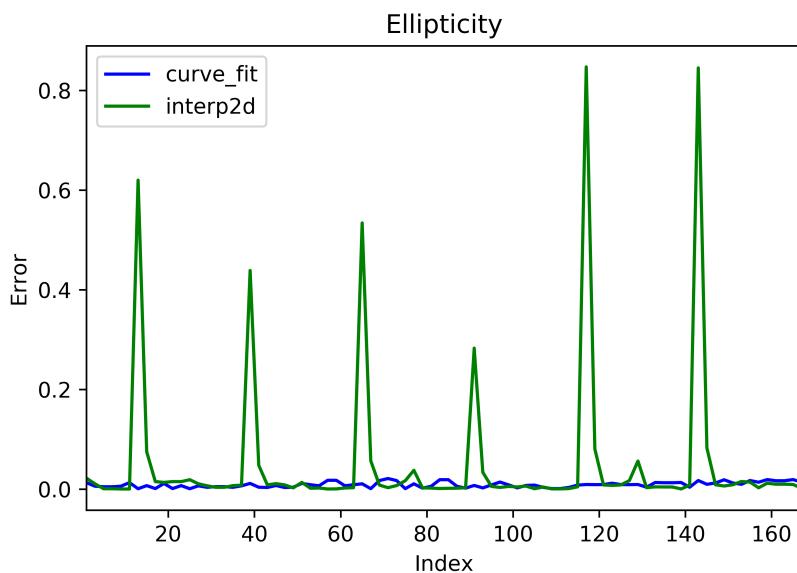


Figura 3.3: Confronto dell'errore relativo ai due metodi di interpolazione.

Analizzando più in dettaglio il comportamento del metodo `interp2d` è possibile mostrare che i punti affetti da errore maggiore sono punti di bordo. L'analisi è stata fatta piazzando il valore interpolato e valore esatto del parametro per ogni riga di punti della griglia 13x13. In fig. 3.4 è mostrato il diverso comportamento di `interp2d` per una riga di punti di bordo e una riga di punti interni alla griglia.

Per poter andare a confrontare gli errori dei due metodi di interpolazione ho creato un subset del dataset iniziale rimuovendo i punti più esterni: `data_mask`. Il passaggio successivo è stato piazzare nuovamente l'errore come in fig. 3.3 e il risultato ottenuto è mostrato in fig. 3.5.

Il plot in fig. 3.5 mostra ancora dei punti nei quali l'errore di `interp2d` è molto maggiore rispetto a quello di `curve_fit`, per tale motivo ho deciso di considerare esclusivamente il metodo `curve_fit` come termine di paragone tra i risultati relativi all'interpolazione e quelli relativi alle reti neurali, come verrà mostrato nella sezione 3.3.

3.2 Reti neurali

In questa sezione mostro come sono state create le architetture delle reti neurali utilizzate per effettuare la regressione, come sono stati gestiti i dati a disposizione per definire `training set` e `validazion set` e com'è avvenuto il processo di training. Infine verrà presentato il confronto dei risultati ottenuti tramite l'utilizzo dell'interpolazione e delle reti (3.3).

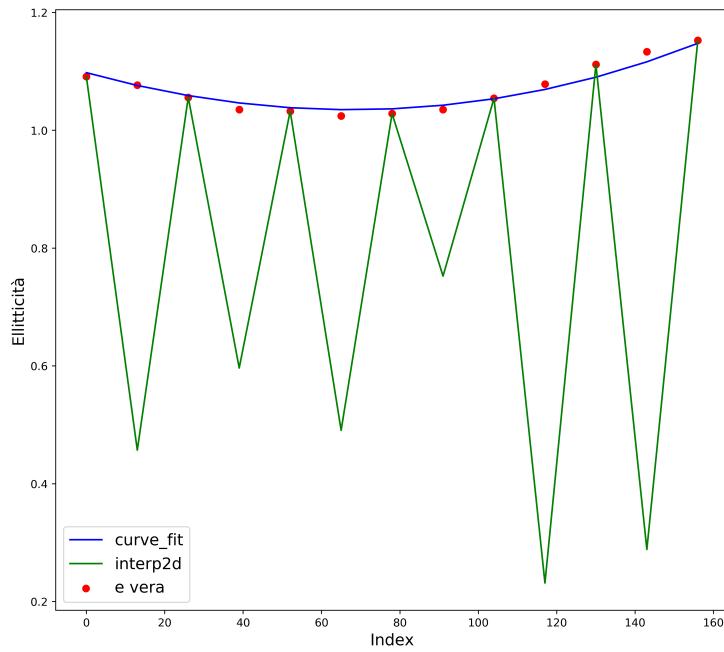
3.2.1 Architettura della rete

Quando si costruisce una rete neurale è necessario definire alcuni elementi che andranno a caratterizzare una particolare architettura di rete. Tali elementi sono:

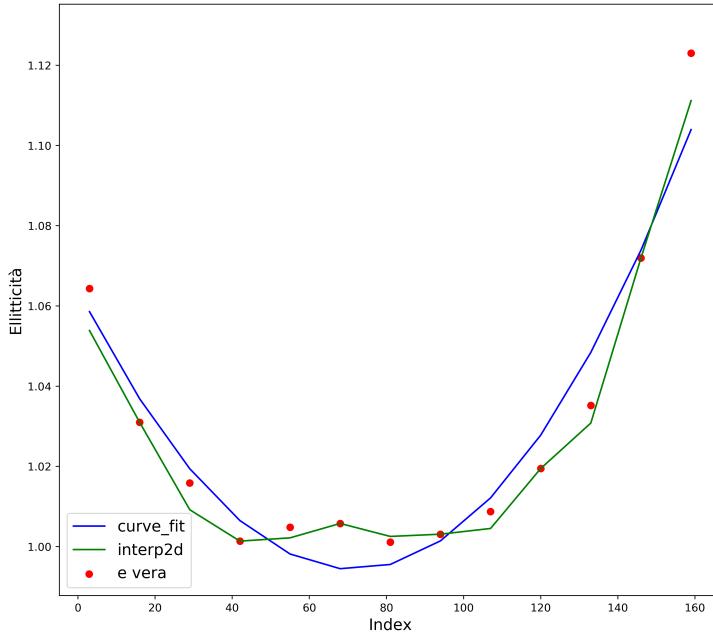
- La dimensione dell'*input layer*.
- La dimensione dell'*output layer*.
- Il numero di *hidden layers*.
- Il numero di *neuroni* per ogni hidden layer.
- La *funzione di attivazione*.
- L'*optimizer*.
- La *loss function*.

La scelta dei loro valori non è mai univoca, non esiste una regola fissa per avere la ricetta perfetta che porta ai risultati migliori. Nel mio lavoro di tesi ho utilizzato 6 diverse architetture; alcuni degli elementi appena citati sono stati mantenuti invariati per ogni architettura³, mentre altri sono stati modificati.

³L'input, per esempio, è sempre una coppia (x, y) che definisce una posizione mentre l'output è un singolo valore di un particolare parametro in quel punto.



(a) Punti di bordo



(b) Punti interni

Figura 3.4: Plot del valore dell'ellitticità per due diverse righe di punti. Nel grafico 3.4a è mostrato l'andamento dell'ellitticità per una riga di punti di bordo mentre nel grafico 3.4b viene analizzata l'ellitticità di una riga interna della griglia di punti.

Per poter descrivere al meglio la scelta delle architetture è necessario fare alcune osservazioni.

Uno dei problemi più diffusi nell'ambito delle reti neurali è quello dell'**overfitting**. Si va incontro ad overfitting quando, per esempio, si utilizza un numero estrema-

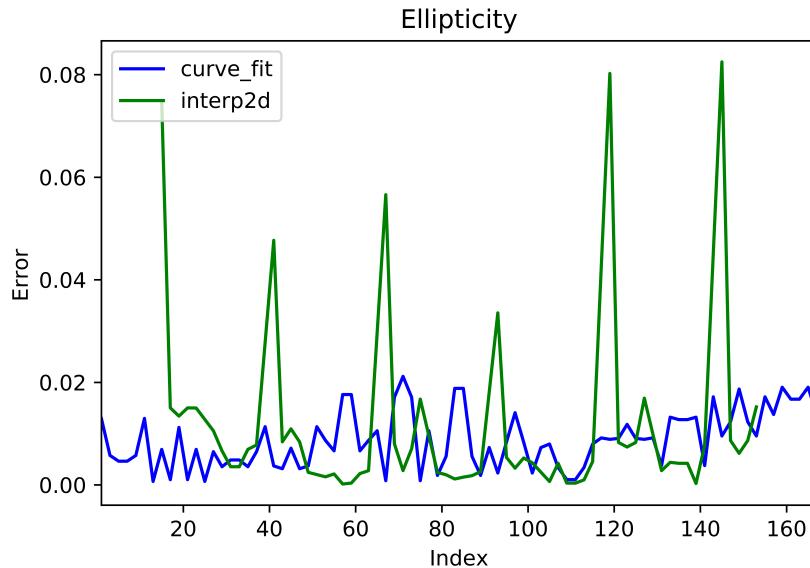


Figura 3.5: Confronto dell’errore relativo ai due metodi di interpolazione dopo aver rimosso i punti di bordo.

mente elevato di neuroni se confrontato con gli elementi che si hanno a disposizione per il training. Ogni neurone è infatti caratterizzato da due parametri liberi (*weight* e *bias*), la presenza di un elevato numero di parametri liberi fa sì che la rete impari perfettamente la corrispondenza tra input e output degli elementi nel training set, ma in tal modo si perde il carattere predittivo necessario per ottenere risultati validi. La dimensione del *trainig set* andrà dunque a fissare un limite superiore del numero di neuroni totali della rete.

Un’ulteriore osservazione di fondamentale importanza riguarda la normalizzazione dei dati. I dati relativi al parametro d’interesse, ovvero i dati di output, sono stati infatti normalizzati per agevolare la convergenza della rete. Tale normalizzazione è descritta in dettaglio nella sezione 3.2.2. Non è stato invece necessario normalizzare l’input poichè appartenente, già in partenza, a un range ottimale⁴.

Con tali premesse è ora possibile giustificare le scelte fatte nella definizione dell’architettura delle reti. I parametri rimasti sempre invariati sono:

- Dimensione dell’**input layer**: 2
- Dimensione dell’**output layer**: 1
- **Optimizer**: optim.Adam⁵
- **Loss function**: MSELoss⁶

Le 6 architetture costruite sono state ottenute tramite una diversa combinazione di funzioni di attivazione e numero di hidden layers. Le funzioni di attivazione che

⁴Sia i valori di x che di y oscillano tra -0.34 e +0.34

⁵È uno dei più comuni *optimizer* forniti da *PyTorch*.

⁶Errore quadratico medio.

ho utilizzato sono la **Tanh** e **Sigmoid**, rappresentate in fig. 3.6. Per ogni funzione di attivazione ho considerato 3 reti con un numero crescente di hidden layers. Al variare del numero di hidden layers ho modificato il numero di neuroni relativo ad ogni layer. Le architetture finali sono riportate in tabella 3.2.1.

Funzione d'attivazione	# hidden layers	# neuroni/hidden layer
Tanh	1	7
Tanh	2	4
Tanh	3	3
Sigmoid	1	7
Sigmoid	2	4
Sigmoid	3	3

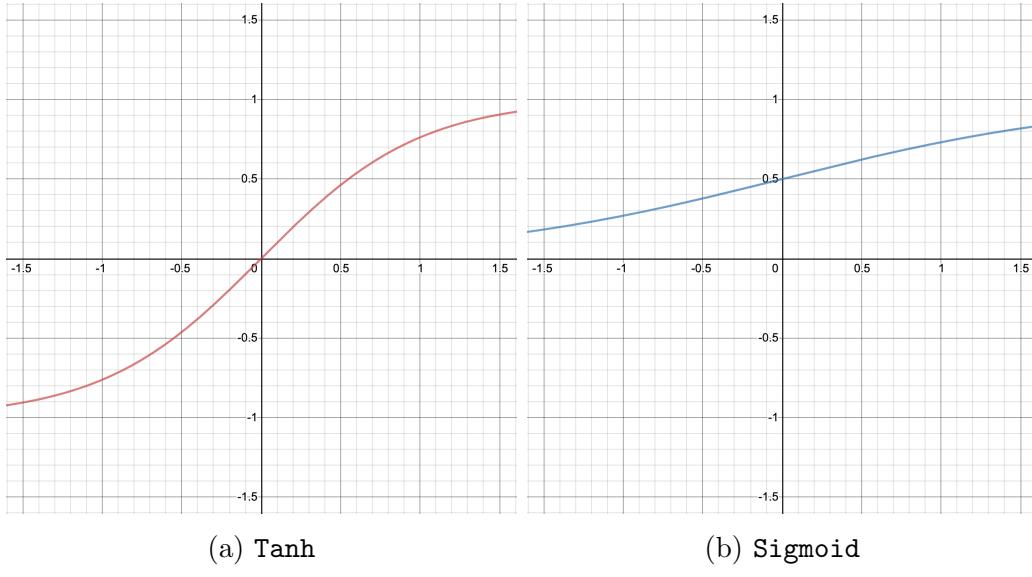


Figura 3.6: Funzioni di attivazione utilizzate. La funzione sigmoidea rappresentata in figura 3.6b è definita come: $P(t) = \frac{1}{1+e^{-t}}$.

3.2.2 Pre Training

In fase di pre-training sono state effettuate due operazioni principali: la normalizzazione dei dati e lo split del dataset in un `training_set` ed un `validation_set`. Gli output sono stati normalizzati in modo da appartenere all'intervallo $[0, 1]$, nel caso di **Sigmoid** come funzione di attivazione, oppure a $[-1, 1]$ nel caso di **Tanh**. Le formule utilizzate per la normalizzazione nei due casi sono:

$$y' = \frac{y + y_m}{y_m - y_M} \quad (3.1)$$

$$y' = \frac{2y - y_m - y_M}{y_M - y_m} \quad (3.2)$$

dove y_m corrisponde al minimo valore del parametro mentre y_M corrisponde al massimo.

L'equazione 3.1 permette una normalizzazione dell'output in $[0, 1]$ mentre la 3.2 in $[-1, 1]$.

inoltre ho definito il numero di epoche totali trasformazione lineare per ottenere l'output nell'intervallo desiderato.. facilito il lavoro della rete

3.2.3 Training

Bla bla bla

3.3 Confronto di risultati

Bla bla bla

Capitolo 4

Conclusioni

Bibliografia

- [1] F. Chollet. *Deep Learning with Python*. Manning, 2018.
- [2] Planck Collaboration, Aghanim, N., Armitage-Caplan, C., Arnaud, M., Ashdown, M., Atrio-Barandela, F., Aumont, J., Baccigalupi, C., Banday, A. J., Barreiro, R. B., Battaner, E., Benabed, K., Benoît, A., Benoit-Lévy, A., Bernard, J.-P., Bersanelli, M., Bielewicz, P., Bobin, J., Bock, J. J., Bonaldi, A., Bonavera, L., Bond, J. R., Borrill, J., Bouchet, F. R., Bridges, M., Bucher, M., Burigana, C., Butler, R. C., Cappellini, B., Cardoso, J.-F., Catalano, A., Chamberlain, A., Chen, X., Chiang, L.-Y., Christensen, P. R., Church, S., Colombi, S., Colombo, L. P. L., Crill, B. P., Curto, A., Cuttaia, F., Danese, L., Davies, R. D., Davis, R. J., de Bernardis, P., de Rosa, A., de Zotti, G., Delabrouille, J., Dickinson, C., Diego, J. M., Dole, H., Donzelli, S., Doré, O., Douspis, M., Dupac, X., Efstathiou, G., Enßlin, T. A., Eriksen, H. K., Finelli, F., Forni, O., Frailis, M., Franceschi, E., Gaier, T. C., Galeotta, S., Ganga, K., Giard, M., Giardino, G., Giraud-Héraud, Y., Gjerløw, E., González-Nuevo, J., Górski, K. M., Gratton, S., Gregorio, A., Gruppuso, A., Hansen, F. K., Hanson, D., Harrison, D., Henrot-Versillé, S., Hernández-Monteagudo, C., Herranz, D., Hildebrandt, S. R., Hivon, E., Hobson, M., Holmes, W. A., Hornstrup, A., Hovest, W., Huffenberger, K. M., Jaffe, A. H., Jaffe, T. R., Jewell, J., Jones, W. C., Juvela, M., Kangaslahti, P., Keihänen, E., Keskitalo, R., Kisner, T. S., Knoche, J., Knox, L., Kunz, M., Kurki-Suonio, H., Lagache, G., Lähteenmäki, A., Lamarré, J.-M., Lasenby, A., Laureijs, R. J., Lawrence, C. R., Leach, S., Leahy, J. P., Leonardi, R., Lesgourgues, J., Liguori, M., Lilje, P. B., Linden-Vørnle, M., López-Caniego, M., Lubin, P. M., Macías-Pérez, J. F., Maino, D., Mandlesi, N., Maris, M., Marshall, D. J., Martin, P. G., Martínez-González, E., Masi, S., Massardi, M., Matarrese, S., Matthai, F., Mazzotta, P., Meinhold, P. R., Melchiorri, A., Mendes, L., Mennella, A., Migliaccio, M., Mitra, S., Moneti, A., Montier, L., Morgante, G., Mortlock, D., Moss, A., Munshi, D., Naselsky, P., Natoli, P., Netterfield, C. B., Nørgaard-Nielsen, H. U., Novikov, D., Novikov, I., O'Dwyer, I. J., Osborne, S., Paci, F., Pagano, L., Paladini, R., Paoletti, D., Partridge, B., Pasian, F., Patanchon, G., Pearson, D., Peel, M., Perdereau, O., Perotto, L., Perrotta, F., Pierpaoli, E., Pietrobon, D., Plaszczynski, S., Pointecouteau, E., Polenta, G., Ponthieu, N., Popa, L., Poutanen, T., Pratt, G. W., Prézeau, G., Prunet, S., Puget, J.-L., Rachen, J. P., Rebolo, R., Reinecke, M., Remazeilles, M., Ricciardi, S., Riller, T., Rocha, G., Rosset, C., Rossetti, M., Roudier, G., Rubiño-Martín, J. A., Rusholme, B., Sandri, M., Santos, D., Scott, D., Seiffert,

- M. D., Shellard, E. P. S., Spencer, L. D., Starck, J.-L., Stolyarov, V., Stompor, R., Sureau, F., Sutton, D., Suur-Uski, A.-S., Sygnet, J.-F., Tauber, J. A., Tavagnacco, D., Terenzi, L., Toffolatti, L., Tomasi, M., Tristram, M., Tucci, M., Tuovinen, J., Türler, M., Umana, G., Valenziano, L., Valiviita, J., Van Tent, B., Varis, J., Vielva, P., Villa, F., Vittorio, N., Wade, L. A., Wandelt, B. D., Watson, R., Wilkinson, A., Yvon, D., Zacchei, A., and Zonca, A. Planck 2013 results. v. lfi calibration. *A&A*, 571:A5, 2014.
- [3] S. Silver and I. of Electrical Engineers. *Microwave Antenna Theory and Design*. IEE electromagnetic waves series. P. Peregrinus, 1984.
 - [4] E. Stevens and L. Antiga. *Deep Learning with PyTorch*. Manning, 2019.
 - [5] M. Tomasi, C. Franceschet, and S. Realini. The quest for cmb b-modes.
 - [6] T. Wilson and S. Hüttemeister. *Tools of Radio Astronomy: Problems and Solutions*. Astronomy and Astrophysics Library. Springer Berlin Heidelberg, 2012.