UNIVERSITÀ DEGLI STUDI DI PADOVA

Dipartimento di Fisica e Astronomia "Galileo Galilei"

Corso di Laurea in Fisica

Tesi di Laurea

Optimal information extraction from cosmological summary statistics with neural networks

Relatore

Prof. Alvise Raccanelli

Correlatore

Dr. Federico Semenzato

Laureando

Nicola Zuccotti

Anno Accademico 2024/2025

Abstract

Ci troviamo in un'era in cui il numero di dati cosmologici è in costante aumento grazie all'uso di strumenti in grado di raccogliere sempre più informazione ed è quindi necessario sviluppare dei mezzi capaci di comprimere i dataset in summary statistics. Negli anni si è imposto l'uso di reti neurali per rispondere a questo problema, arrivando alle IMNN, in grado di analizzare attraverso l'uso di reti neurali spettri non lineari e altre statistiche più complesse.

Teoricamente conosciamo la massima quantità d'informazione estraibile dalle varie statistiche, ma vogliamo testare quanto le reti neurali si avvicinano a tale limite. Partendo da questa idea, in questo lavoro viene studiata la capacità di diversi Multilayer Perceptron (MLP) fully-conncted di estrarre in modo ottimale l'informazione cosmologica contenuta nello spettro di potenza lineare del modello ΛCDM. L'obiettivo è valutare in che misura diverse architetture possano avvicinarsi al limite teorico fissato dalla matrice di Fisher del power spectrum fiduciale (dato dai parametri delle osservazioni Planck2018), in modo tale da ottenere la struttura che in maniera più ottimale estragga informazione sui parametri al meno costo. Ciò è utile in diverse applicazioni, tra le quali Simulation-based inference, per le quali è necessario comprimere informazione in modo efficiente sia dal punto di vista dell'estrazione sia per quanto riguarda il costo computazionale.

Vengono generati 20 000 power spectrum tramite Latin Hypercube Sampling in due diversi insiemi di prior, uno ampio e uno ristretto attorno ai valori di Planck2018. I MLP vengono addestrati a ricostruire i parametri cosmologici, variando profondità e larghezza delle architetture, e valutati attraverso errori relativi e confronto diretto tra la Fisher teorica e la Fisher del MLP. È inoltre proposta una formulazione approssimata della Fisher del MLP, validata confrontando le predizioni con il limite lineare. L'analisi mostra che l'informazione del power spectrum lineare satura per reti con profondità maggiore di cinque strati, mentre l'aumento del numero di neuroni ha un impatto marginale. Le architetture piramidali risultano le più efficaci nel regime di prior ampi, dimostrando una migliore capacità di generalizzazione. Questo lavoro pone una base per estensioni future dell'analisi a scale non lineari e più complesse del power spectrum.

Indice

Abstract **ii**

1 Introduzione **1**
1.1 Obiettivi della tesi . . . 1
1.2 Stato dell'arte . . . 1
1.3 Struttura del lavoro . . . 2

2 Contesto Cosmologico **3**
2.1 Modello ΛCDM . . . 3
2.1.1 Contesto cosmologico . . . 3
2.1.2 Parametri . . . 5
2.2 Power Spectrum . . . 6
2.2.1 Crescita delle perturbazioni . . . 6
2.2.2 Spettro della materia . . . 7
2.3 Matrice dell'informazione di Fisher . . . 8

3 Deep Learning **9**
3.1 Cos'è il Deep Learning . . . 9
3.2 Latin Hypercube Sampling . . . 10
3.3 Multilayer Perceptron . . . 10
3.3.1 Matrice dell'informazione di Fisher del MLP . . . 12

4 Analisi e discussione **14**
4.1 Metodologia . . . 14
4.2 Approssimazione lineare . . . 17
4.3 Analisi sulla profondità . . . 18
4.3.1 Prior ampi . . . 19
4.3.2 Prior stretti . . . 21
4.4 Numero di simulazioni . . . 25

5 Conclusioni **27**

Capitolo 1

Introduzione

1.1 Obiettivi della tesi

Le reti neurali hanno dimostrato di poter catturare informazione non lineare inaccessibile alle sole statistiche di due punti, ma il loro successo dipende dalla scelta dell'architettura e dalla quantità di dati di addestramento disponibili.

Esistono diversi teoremi noti come *Universal approximation theorems* che affermano che una rete neurale con un singolo strato nascosto e un numero sufficiente di neuroni può approssimare qualsiasi funzione continua su un compatto, con arbitraria precisione [2] e che esistono funzioni che una rete deep può rappresentare con pochi neuroni, ma che una rete poco profonda può rappresentare solo con un numero esponenziale di neuroni [3].

Con questa idea in testa, in questo lavoro viene presentato uno studio di convergenza di reti neurali multi-layer perceptron (MLP) addestrate a ricostruire i parametri cosmologici a partire dallo spettro di potenza lineare su larga scala nel modello ΛCDM. Attraverso la matrice di Fisher si ha il riferimento teorico con cui valutare la capacità delle reti di saturare l'informazione. Si testano diverse architetture in termini di profondità e larghezza, confrontando l'informazione in uscita con quella fiduciale data dai parametri cosmologici delle osservazioni Planck2018 [7]. Il fine ultimo è quello di identificare una struttura che massimizzi l'efficienza computazionale e di estrazione dell'informazione.

1.2 Stato dell'arte

Negli ultimi anni l'interazione tra metodi di machine learning (ML), tecniche di compressione e cosmologia si è intensificata, con l'obiettivo di sfruttare al massimo l'informazione nascosta nei campi cosmologici e nelle statistiche rilevanti. Tradizionalmente l'inferenza cosmologica si basa su statistiche di secondo ordine, come il power spectrum, e sul calcolo della matrice di Fisher. Tuttavia, in regime non lineare e su grande volume dati, queste tecniche possono risultare subottimali o insufficienti.

Uno dei primi studi sul tema è un metodo lineare noto come MOPED (Multiple Optimised Parameter Estimation and Data compression), che costruisce vettori di compressione lineare ottimizzati per preservare l'informazione Fisher relativa ai parametri stimati [5]. Una generalizzazione non lineare e non gaussiana successiva è data dalle Information Maximising Neural Networks (IMNN) le quali addestrano reti neurali a produrre compressioni non lineari massimizzando direttamente una stima della Fisher information [9]. Metodi più recenti mirano a rendere l'ottimizzazione più stabile e scalabile su grandi dataset simulativi, migliorando la robustezza dell'apprendimento in presenza di rumore e di effetti non lineari severi. Tra questi TECMAR è una famiglia di approcci emergenti basati sul contrastive learning.

Accanto alla capacità predittiva, la robustezza e l'interpretabilità costituiscono sfide centrali: è essenziale che i modelli ML possano generalizzare al di fuori del set di simulazioni di training. Alcuni studi

mettono in guardia contro reti che imparano dettagli specifici delle simulazioni anziché relazioni fisiche generalizzabili; tali studi hanno mostrato come approssimazioni comuni possano portare a stime di errore subottimali o fuorvianti se non trattate con cautela [6].

Nel lavoro recente "How many simulations do we need for simulation-based inference in cosmology?" [4] si evidenzia che molti ensemble di simulazioni comunemente usati oggi non sono sufficienti per far sì che una rete neurale raggiunga un regime informativo ottimale, anche per il solo power spectrum. Gli autori identificano una legge empirica che lega il numero di simulazioni disponibili alla frazione dell'informazione estratta dalla rete rispetto al limite Cramér–Rao, e propongono come punto di riferimento il dataset Big Sobol Sequence composto da oltre 32 768 simulazioni per esplorare la convergenza delle reti neurali. Anche quando ci si limita al solo power spectrum, il numero di simulazioni richiesto per ottenere una compressione efficace e priva di bias risulta più elevato di quanto comunemente adottato. L'obiettivo di questa tesi si colloca in modo complementare rispetto a tale prospettiva. Qui l'attenzione non è sulla stima della numerosità di simulazioni necessaria, ma nella ricerca tra diverse architetture di Multilayer Perceptron fully-connected che siano in grado di avvicinarsi al limite teorico informativo nel caso in cui un insieme di simulazioni sia già disponibile. In particolare, si studia in che misura profondità e larghezza delle reti influenzino l'efficienza informativa della compressione.

1.3 Struttura del lavoro

Il lavoro di seguito riportato viene strutturato come segue: dopo il corrente capitolo introduttivo, viene fornito un contesto cosmologico del modello ΛCDM e del power spectrum con la relativa matrice dell'informazione di Fisher (FIM); nel terzo capitolo viene invece fornito un contesto in cui inquadrare i modelli di reti neurali analizzati; nel quarto capitolo vengono fornite le caratteristiche delle reti e della gestione dei dati di addestramento, una prova della formulazione della FIM del MLP e i risultati derivati dall'analisi delle varie strutture considerate.

Capitolo 2

Contesto Cosmologico

Nel seguente capitolo viene fornita un'introduzione sul modello cosmologico adottato nella trattazione successiva e sul power spectrum di tale modello, il quale sarà poi l'input di addestramento del MLP.

2.1 Modello ΛCDM

Il modello ΛCDM o modello standard della cosmologia è il modello cosmologico adottato attualmente per descrivere l'universo in cui viviamo e che meglio interpola i dati osservati disponibili [1]. Esso è emerso come paradigma principe alla fine degli anni '90 a seguito della pubblicazione di alcune prove che verranno trattate più avanti. Secondo il ΛCDM l'universo è spazialmente piatto, contiene materia (5%), materia oscura (25%) e una costante cosmologica Λ positiva che funge da energia oscura (70%). Con materia si intende tutte le particelle che è possibile descrivere tramite il modello standard. Ci si riferirisce ad essa anche come materia barionica e compone la porzione di universo che conosciamo meglio. La materia oscura è stata introdotta per spiegare effetti gravitazionali su larga scala. Essa è non barionica, quindi di natura diversa rispetto alla materia ordinaria, fredda, ovvero possiede una velocità non paragonabile alla velocità della luce all'epoca dell'uguaglianza radiazione-materia, e non può irradiare fotoni. Inoltre non interagisce con altra materia se non con la forza gravitazionale e forse la forza debole. Infine, l'energia oscura è stata introdotta per spiegare un fenomeno relativamente recente rispetto all'età dell'universo, che è l'accellerazione dell'espansione di esso.

2.1.1 Contesto cosmologico

La costante cosmologica venne introdotta per la prima volta da Albert Einstein a seguito della pubblicazione della teoria della relatività generale. In quel periodo storico la cosmologia era definita ancora come una scienza naturalistica e non era stata ancora accertata l'esistenza di galassie diverse dalla nostra, né erano noti i concetti di redshift e di espansione dell'universo. Quello che si osservava erano gli spettri delle stelle all'interno della Via Lattea, i quali non mostravano un allontanamento o un avvicinamento medio alla Terra. Einstein, pertanto, assunse che l'universo era statico e sviluppò un modello cosmologico compatibile con queste ipotesi.

Nel 1917 Einstein decise allora di modificare le equazioni di campo alla base della relatività generale nella seguente versione:

$$R_{\mu\nu} - \tfrac{1}{2} R g_{\mu\nu} + \Lambda g_{\mu\nu} = \frac{8\pi G}{c^4} T_{\mu\nu} \tag{2.1}$$

dove Λ è la costante cosmologica, avente le dimensioni $[L]^{-2}$. Tale modifica è ottenuta cambiando una delle ipotesi su cui si basa il procedimento per ricavare le equazioni di Einstein: *il tensore di Einstein $G_{\mu\nu}$ è composto solo da termini proporzionali a derivate di secondo ordine del tensore metrico $g_{\mu\nu}$(come il tensore di Ricci-Curbastro e lo scalare di Ricci)* viene rimpiazzata da *il tensore di Einstein $G_{\mu\nu}$ è composto solo da termini proporzionali a derivate di ordine ≤ 2 del tensore metrico $g_{\mu\nu}$.*

Friedmann, partendo dagli studi della relatività generale, scrisse le sue due equazioni, le quali legano il movimento di espansione e contrazione dell'universo e la sua geometria con l'energia, caratterizzata dalla densità totale di energia ρ e dalla pressione p, presente in esso:

$$H^2(t) \equiv \left[\frac{\dot{a}(t)}{a(t)}\right]^2 = \frac{8\pi G}{3}\rho(t) - \frac{Kc^2}{a^2(t)} \tag{2.2}$$

$$\frac{\ddot{a}(t)}{a(t)} = -\frac{4\pi G}{3}\left[\rho(t) + \frac{3\,p(t)}{c^2}\right] \tag{2.3}$$

In queste equazioni $a(t)$ è il fattore di scala, G e c sono rispettivamente le costanti universali di gravitazione e della velocità della luce nel vuoto. $\rho = \rho_m + \rho_\Lambda$ con $\rho_\Lambda = \frac{\Lambda}{8\pi G}$ e p sono la densità di massa volumetrica e la pressione. K definisce la geometria dello spazio: se $K > 0$ l'universo è sferico, se $K < 0$ è iperbolico e se $K = 0$ è piatto. Nel modello preso in considerazione, lo spazio è piatto, di conseguenza si pongono nulli tutti gli elementi relativi alla curvatura e verranno omesse ulteriori discussioni.

Infine $H(t)$ è il paramentro di Hubble, il quale può essere ridefinito in funzione del redshift, sfruttando la relazione che presenta con il fattore di scala $a(z) = \frac{a_0}{1+z}$ ($a_0 = 1$ è il fattore di scala normalizzato del presente della Terra):

$$H(z) = H_0 E(z), \tag{2.4}$$

dove $H_0 = 100\frac{hkm}{sMpc}$ è il parametro di Hubble a $z = 0$ e $E(z)$ è una funzione che varia in base al modello cosmologico e successivamente sarà definita per il caso preso in considerazione. Proseguendo si ricava il valore critico della densità ρ, la quale rende l'universo piatto:

$$\rho_{crit}(z) = \frac{3H^2(z)}{8\pi G} \tag{2.5}$$

Si introduce il parametro di densità $\Omega_i(z) = \frac{\rho_i(z)}{\rho_{crit}(z)}$ in modo tale da riscrivere eq. 2.2 come segue:

$$\sum_{i=1}^{n} \Omega_i(z) = 1 \tag{2.6}$$

Le equazioni di Friedmann possono essere ricombinate nuovamente in un'equazione di conservazione dell'energia che specifica la relazione tra ρ e $a(t)$. Una soluzione di questa equazione richiede di specificare le proprietà di ciascuna componente energetica nella forma di un'equazione di stato, $p = p(\rho)$; si specifica quest'ultima in termini del parametro dell'equazione di stato $w \equiv p/\rho c^2$. Nel caso in cui w sia costante nel tempo, l'equazione di conservazione dell'energia implica

$$\rho_i(a) \propto a^{-3(1+w_i)} \tag{2.7}$$

Una volta note le relazioni $\rho_i(a)$, queste possono essere utilizzate nell'eq. 2.2 per trovare una soluzione per $a(t)$.

La densità di energia della materia a z piccoli è principalmente sotto forma di barioni e particelle di materia oscura fredda (CDM), che sono descritte da $w_b = w_c = 0$. La densità di radiazione fotonica è caratterizzata da $w_\gamma = 1/3$. Un contributo dai neutrini massivi può essere descritto da un parametro dell'equazione di stato variabile w_ν, che corrisponde a w_γ ai tempi primordiali e a w_c quando diventano non relativistici.

Per lo scopo di tale trattazione, si considera la densità di radiazione trascurabile, $\Omega_{\gamma,0} = 0$, e si trattano i neutrini massivi come parte del contributo totale della materia, con $\Omega_{m,0} = \Omega_{c,0} + \Omega_{b,0} + \Omega_{\nu,0}$, poiché sono non relativistici ai bassi redshift rilevanti per queste sonde.

L'accelerazione cosmica richiede ciò che viene comunemente denominata energia oscura. Il modello standard della cosmologia ΛCDM assume che questo fenomeno sia dovuto alla presenza di una costante cosmologica, indicata come Λ, descritta da un'equazione di stato costante $w_\Lambda = -1$, che, secondo eq.

2.7, corrisponde a una densità di energia indipendente dal tempo ρ_Λ. Il modello ΛCDM attualmente si adatta molto bene alle osservazioni [7], ma presenta diverse problematiche: il valore osservato della costante cosmologica è di molti ordini di grandezza più piccolo delle previsioni teoriche; il fatto che le densità di Λ e CDM siano simili oggi mentre si sono evolute in modo molto diverso nel tempo marca la nostra epoca come un tempo speciale nell'evoluzione dell'Universo; inoltre, il modello e i suoi parametri non possono essere predetti da principi fisici. Uno scenario più generale per la componente responsabile dell'accelerazione cosmica postula un'energia oscura dinamica, con un parametro dell'equazione di stato dipendente dal redshift $w_\lambda(z)$. Una parametrizzazione comunemente utilizzata e ben testata della dipendenza temporale è

$$w_\Lambda(z) = w_0 + w_a \frac{z}{1+z} \tag{2.8}$$

dove w_0 è il valore presente ($z = 0$) dell'equazione di stato e w_a è una misura della sua variazione temporale. Il modello ΛCDM pone $w_0 = -1$ e $w_a = 0$, essendo di fatto w_Λ indipendente dal tempo. Utilizzando eq. 2.7 nell'eq. 2.2, la funzione $E(z)$ definita in eq. 2.4 diventa

$$E(z) = \sqrt{\Omega_{m,0}(1+z)^3 + \Omega_{\Lambda,0}} \tag{2.9}$$

2.1.2 Parametri

Il modello ΛCDM standard è descritto da sei parametri, i quali sono sufficienti a spiegare la maggior parte degli eventi rilevati:

1. $\Omega_b h^2 = 0.02237 \pm 0.00015$ la densità di materia barionica quantifica la frazione di materia barionica (si considerano tutte le particelle del modello standard eccetto i neutrini, i quali vengono trattati a parte) pesata con il quadrato della costante di hubble ridotta $h = \frac{H_0}{100}$.
2. $\Omega_c h^2 = 0.1200 \pm 0.0012$ la densità di materia oscura fredda.
3. $100\theta_s = 1.0411 \pm 0.0003$ l'angolo acustico definisce la scala angolare caratteristica delle oscillazioni acustiche nel plasma primordiale, visibile nei picchi della CMB. Esso è utile per misurare l'espansione dell'universo e la sua geometria.
4. $\tau = 0.0544 \pm 0.0073$ la profondità ottica descrive la probabilità che un fotone della CMB sia stato diffuso dagli elettroni liberi durante la reionizzazione, circa un miliardo di anni dopo il Big Bang.
5. $n_s = 0.9649 \pm 0.0042$ l'indice spettrale quantifica come varia l'ampiezza delle fluttuazioni primordiali su diverse scale. Se fosse esattamente uguale a 1, lo spettro sarebbe invariante per scale, ma la deviazione misurata ci dice che le perturbazioni sono leggermente più intense su scale grandi rispetto a quelle piccole.
6. $A_s = (2.100 \pm 0.030)10^{-9}$ l'ampiezza dello spettro primordiale delle perturbazioni scalari quantifica l'intensità delle fluttuazioni di densità generate durante l'inflazione.

Da questi si derivano altri parametri che corrispondono a grandezze osservabili:

1. $H_0 = (67.36 \pm 0.54)\frac{km}{sMpc}$ il parametro di Hubble.
2. $\Omega_m h^2 = \Omega_c h^2 + \Omega_b h^2 = 0.1430 \pm 0.0011$ la densità di materia.
3. $\Omega_\Lambda = 0.6847 \pm 0.0073$ la densità di energia oscura.
4. $\Omega_r = (9.14 \pm 0.25)\, 10^{-5}$ la densità di radiazione derivabile dal CMB.
5. $\sigma_8 = 0.8111 \pm 0.0060$ misura l'ampiezza delle fluttuazioni di densità su scale di $8\frac{Mpc}{h}$, utile per confrontare teoria e osservazioni delle strutture su larga scala.
6. $t_0 = (13.797 \pm 0.023)Gyr$ l'età dell'universo.

Il modello si può poi estendere rilassando le ipotesi iniziali, inserendo altri parametri quali la massa dei neutrini, la densità di curvatura Ω_k e l'equazione di stato dell'energia oscura anzichè fissarla a $w = -1.0$.

2.2 Power Spectrum

Il principio Cosmologico afferma che la struttura dell'universo che osserviamo su larga scala è isotropa e omogenea. Essa è esprimibile attraverso una densità media globale con delle correzioni date da piccole fluttuazioni. Le fluttuazioni di densità per una data componente i sono caratterizzate dal contrasto di densità $\delta_i(\mathbf{x}, z) \equiv \rho_i(\mathbf{x}, z)/\bar{\rho}_i(z) - 1$ che quantifica le deviazioni del campo di densità $\rho_i(\mathbf{x}, z)$ attorno alla densità spaziale media $\bar{\rho}_i(z)$ nello spazio, dove $\mathbf{x}$ è una coordinata comovente tridimensionale a un redshift z. Una descrizione più conveniente è data nello spazio di Fourier decomponendo δ in onde piane, in quanto l'inflazione ci fornisce condizioni iniziali semplici espresse in termini di modi di Fourier e le osservazioni hanno dimostrato che in buona approssimazione le fluttuazioni primordiali sono descritte da un campo casuale gaussiano.

$$\delta_i(\mathbf{x}, z) = \int \frac{d^3k}{(2\pi)^3} \tilde{\delta}_i(\mathbf{k}, z) \exp(-i\mathbf{k} \cdot \mathbf{x}) \,. \tag{2.10}$$

Possiamo ora definire implicitamente lo spettro di potenza, $P_i(k, z)$, per la componente generica i come

$$\langle \tilde{\delta}_i(\mathbf{k}, z) \tilde{\delta}_i(\mathbf{k}', z) \rangle = (2\pi)^3 \delta_D(\mathbf{k} + \mathbf{k}') P_i(k, z) \tag{2.11}$$

dove δ_D è la delta di Dirac. Sotto le assunzioni di omogeneità e isotropia statistica, lo spettro di potenza può dipendere solo da $k = |\mathbf{k}|$ e z.

Lo spettro di potenza primordiale adimensionale della perturbazione di curvatura ζ generata dall'inflazione è parametrizzato come una legge di potenza

$$P_\zeta(k) = A_s \left(\frac{k}{k_0} \right)^{n_s - 1} \tag{2.12}$$

dove A_s è l'ampiezza della perturbazione scalare primordiale, l'indice spettrale scalare n_s misura la deviazione dall'invarianza di scala ($n_s = 1$), e k_0 è una scala pivot.

Lo spettro di potenza corrispondente definito nell'eq. 2.11 in termini di perturbazioni di densità è collegato a quello primordiale tramite la funzione di trasferimento T_i attraverso

$$P_i(k, z) = 2\pi^2 T_i^2(k, z) P_\zeta(k) k \tag{2.13}$$

per una componente generica i.

Nell'Universo primordiale durante la dominazione della radiazione, le perturbazioni di curvatura con scale comoventi più piccole dell'orizzonte sono soppresse, mentre le fluttuazioni super-orizzonte rimangono inalterate, finché non entrano nell'orizzonte. Nell'era dominata dalla materia, le perturbazioni di curvatura su tutte le scale rimangono costanti. Questo implica che una scala caratteristica corrispondente all'epoca di uguaglianza materia-radiazione è impressa sulla forma della funzione di trasferimento, e quindi sullo spettro di potenza della materia.

2.2.1 Crescita delle perturbazioni

La crescita delle fluttuazioni di densità obbedisce a un'equazione differenziale del secondo ordine. A tempi in cui le fluttuazioni sono ancora piccole, le equazioni dei fluidi possono essere linearizzate. Durante la dominazione della materia, considerando la materia come un fluido ideale senza pressione, l'equazione per l'evoluzione del contrasto di densità diventa

$$\ddot{\delta}_m(k, z) + 2H\dot{\delta}_m(k, z) - \frac{3H_0^2 \Omega_{m,0}}{2a^3} \delta_m(k, z) = 0 \tag{2.14}$$

Nello modello ΛCDM, questa equazione può essere scritta in termini del redshift come

$$\delta_m''(k,z) + \left[\frac{H'(z)}{H(z)} - \frac{1}{1+z}\right]\delta_m'(k,z) - \frac{3}{2}\frac{\Omega_m(z)}{(1+z)^2}\delta_m(k,z) = 0 \tag{2.15}$$

dove trascuriamo le perturbazioni dell'energia oscura, il primo si riferisce alla derivata rispetto a z, e $\Omega_m(z)$ è dato da

$$\Omega_m(z) = \frac{\Omega_{m,0}(1+z)^3}{E^2(z)} \tag{2.16}$$

Le soluzioni $\delta_m(k,z)$ dell'eq. 2.15, a z piccoli, sono indipendenti dalla scala e ciò motiva l'introduzione del fattore di crescita $D(z)$ attraverso

$$\delta_m(k,z) = \delta_m(k,z_i)\frac{D(z)}{D(z_i)} \tag{2.17}$$

dove z_i è un redshift di riferimento arbitrario nell'era dominata dalla materia.

Una quantità utile è il parametro del tasso di crescita, definito come

$$f(a) = \frac{d\ln D(a)}{d\ln a} = -\frac{d\ln D(z)}{d\ln(1+z)} \tag{2.18}$$

che approssimativamente in ΛCDM soddisfa $f(a) \simeq \Omega_m(a)^\gamma$ con $\gamma \simeq 0.56$. Usando queste definizioni, il tasso di crescita soddisfa un'equazione differenziale del primo ordine

$$f'(z) - \frac{f(z)^2}{1+z} - \left[\frac{2}{1+z} - \frac{H'(z)}{H(z)}\right]f(z) + \frac{3}{2}\frac{\Omega_m(z)}{1+z} = 0 \tag{2.19}$$

con condizione iniziale $f(z=z_i) = 1$.

Usando l'eq. 2.18 la soluzione per $D(z)$ può essere espressa in termini di $f(z)$ dall'integrale

$$D(z) = D(0)\exp\left[-\int_z^0 dz'\frac{f(z')}{1+z'}\right] \tag{2.20}$$

2.2.2 Spettro della materia

In ΛCDM la crescita tardiva della materia è indipendente dalla scala, così che la funzione di trasferimento $T_m(k,z)$ può essere divisa in una parte dipendente dalla scala $T_m(k) \sim \frac{1}{1-(\frac{k}{k_{eq}})^2}$ (normalizzata in modo che $T_m \to 1$ per $k \to 0$ e k_{eq} è il k al tempo di transizione tra era della dominazione della radiazione e era della materia) e il fattore di crescita indipendente dalla scala $D(z)$ introdotto sopra.

Un modo conveniente per esprimere lo spettro di potenza definito nell'eq. 2.13 per la materia è

$$P_{\delta\delta}(k,z) = \left(\frac{\sigma_8}{\sigma_N}\right)^2\left[\frac{D(z)}{D(0)}\right]^2 T_m^2(k)k^{n_s} \tag{2.21}$$

con la costante di normalizzazione

$$\sigma_N^2 = \frac{1}{2\pi^2}\int dk T_m^2(k)|W_{TH}(kR_8)|^2 k^{n_s+2} \tag{2.22}$$

dove $W_{TH}(x) = 3(\sin x - x\cos x)/x^3$ è la trasformata di Fourier del filtro top-hat, e $R_8 = 8h^{-1}$ Mpc. W_{TH} viene usata per mediare o filtrare la densità di materia su una scala specifica in questo caso data appunto da R_8.

La scala pivot k_0 e l'ampiezza del modo scalare A_s sono assorbite nella normalizzazione, che è progettata per dare un valore desiderato di σ_8, la radice quadrata media (rms) delle fluttuazioni di densità evolute linearmente al presente in sfere di $8h^{-1}$ Mpc, che è dato da

$$\sigma_8^2 = \frac{1}{2\pi^2}\int dk P_{\delta\delta}(k,0)|W_{TH}(kR_8)|^2 k^2 \tag{2.23}$$

Quindi, lo spettro di potenza generico descritto nell'eq. 2.13 si collega allo spettro di potenza della materia dell'eq. 2.21 quando la funzione di trasferimento della specie i è

$$T_m(k,z) = \frac{\sigma_8 k_0^{(n_s-1)/2}}{\sigma_N \pi \sqrt{2A_s}} \frac{D(z)}{D(0)} T_m(k) \tag{2.24}$$

2.3 Matrice dell'informazione di Fisher

La matrice dell'informazione di Fisher (FIM) rappresenta uno degli strumenti statistici fondamentali applicati alla cosmologia, in quanto permette di quantificare la quantità di informazione contenuta in un dato osservabile riguardo ai parametri cosmologici. Formalmente, essa è definita come l'aspettativa del valore negativo della derivata seconda del logaritmo della funzione di verosimiglianza rispetto ai parametri di interesse:

$$F = -\langle \nabla_\theta \nabla_\theta^T L \rangle = \langle \nabla_\theta L \nabla_\theta^T L \rangle \tag{2.25}$$

In altre parole, la matrice di Fisher misura la curvatura locale della likelihood nello spazio dei parametri, fornendo una stima della precisione massima (limite di Cramér–Rao) con cui tali parametri possono essere vincolati.

$$Var(\theta) \geq F_{\theta\theta}^{-1} \tag{2.26}$$

Nel limite asintotico di un grande numero di modi, e per statistiche informative, la verosimiglianza dello spettro di potenza è approssimativamente gaussiana, quindi la log-verosimiglianza assume la forma:

$$L = -\frac{1}{2}(d - P(\theta))^T C^{-1} (d - P(\theta)) - \frac{1}{2} \ln |2\pi C| \tag{2.27}$$

Ciò porta a una formulazione della Fisher come segue:

$$F = \nabla_\theta P^\top C^{-1} \nabla_\theta P \tag{2.28}$$

dove $\nabla_\theta P$ è il laplaciano rispetto al parametro θ e C è la covarianza data dalla seguente formula, nell'approssimazione di scale grandi e gaussianità [15]:

$$C(k_1, k_2) = \frac{2}{N_k} (P(k))^2 \delta_{k_1,k_2} \tag{2.29}$$

nella quale $N_k = \frac{Vk^2 \Delta k}{2\pi^2}$ è il numero di modi attorno a k. Lo shot noise viene estromesso dalla covarianza, in quanto è il contributo alla stima del power spectrum dato dal fatto che la distribuzione della materia non viene misurata come un campo continuo, ma come un insieme discreto di galassie e oggetti cosmici: essendo a scale grandi non si vede un insieme discreto di oggetti. Nel caso del power spectrum teorico della distribuzione di materia, la matrice di Fisher si costruisce a partire dalla dipendenza dello spettro di potenza $P(k)$ dai parametri cosmologici θ_i. In pratica, per ciascun parametro si calcola la derivata parziale del power spectrum teorico, valutando quanto una piccola variazione di quel parametro modifichi l'osservabile. Questa formulazione evidenzia come la sensibilità del power spectrum a ciascun parametro entri direttamente nella stima della Fisher, e come le correlazioni tra parametri diversi siano codificate nei termini incrociati. In questo senso, la matrice di Fisher costruita a partire dal power spectrum teorico fornisce una stima ideale, legata unicamente alle proprietà statistiche del segnale e alla forma analitica delle derivate.

Capitolo 3

Deep Learning

3.1 Cos'è il Deep Learning

Il normale approccio risolutivo con cui si affrontano problemi con l'ausilio una macchina computazionale è attraverso la stesura di un codice più o meno complesso in base alla complessità del problema, ma soprattutto codificato come un rigido insieme di regole che specificano con precisione come il computer debba comportarsi. Esistono però problemi che richiedono un grande sforzo cognitivo al fine di elaborare tutte le condizioni necessarie affinchè il codice lavori come desiderato. Alcuni esempi di tali problemi possono essere il riconoscimento delle persone e degli oggetti in un immagine o la previsione del meteo per il giorno successivo. La quantità di informazioni e la difficoltà nel categorizzarle rende tale approccio di computazione inadatto alla situazione.

La risposta a tali problemi la dà il Machine Learning, ovvero lo studio di algoritmi in grado di apprendere dall'esperienza. Man mano che un algoritmo di apprendimento automatico accumula più esperienza, tipicamente sotto forma di dati osservativi o interazioni con un ambiente, le sue prestazioni migliorano. Grazie a questo approccio bastano poche righe di codice per risolvere problemi complessi. Il processo di allenamento spesso si può ricondurre a questi semplici passaggi: si inizia prendendo un modello inizializzato in modo casuale che non è in grado di svolgere alcuna funzione utile, poi si prendono alcuni dei dati a disposizione e si modificano le impostazioni per migliorare le prestazioni del modello sulla base di tali esempi. Dopodichè si ripetono gli ultimi due passaggi fino a quando il modello non sarà perfetto. Ciò di cui si ha la necessità sono un grande e diversificato set di dati, un modello adatto alla struttura del problema, una funzione che quantifichi quanto il modello stia ragionando bene e un algoritmo che permetta di aggiustare il modello in base al risultato della funzione precedente.

Negli ultimi decenni lo sviluppo dei computer in termini di RAM non ha tenuto il passo con la crescita esponenziale della larghezza dei set di dati. Allo stesso tempo, l'aumento della potenza di calcolo ha superato la crescita dei dataset. Ciò significa che i modelli statistici devono diventare più efficienti in termini di memoria e quindi sono liberi di dedicare più cicli di calcolo all'ottimizzazione dei parametri. Di conseguenza, il punto di forza dell'apprendimento automatico e della statistica si è spostato dai modelli lineari e dai metodi kernel alle reti neurali profonde. Questo è anche uno dei motivi per cui molti dei pilastri del deep learning, come i perceptron multistrato [12] e le reti neurali convoluzionali [14] sono stati riscoperti.

Il deep learning è il sottoinsieme dell'apprendimento automatico che si occupa di modelli basati su reti neurali a più livelli. È profondo proprio nel senso che i suoi modelli apprendono molti livelli di trasformazioni. Sebbene questo possa sembrare limitato, il deep learning ha dato origine a una vasta gamma di modelli, tecniche, formulazioni di problemi e applicazioni. Sono state sviluppate molte intuizioni per spiegare i vantaggi della profondità. Ciò che differenzia il deep learning è che le operazioni apprese a ciascuno dei numerosi livelli di rappresentazione vengono apprese congiuntamente dai dati.

Quando i dati sono scarsi, è necessario fare affidamento su ipotesi semplificative sulla realtà per ottenere modelli utili, mentre quando i dati sono abbondanti, questi possono essere sostituiti da modelli non parametrici che si adattano meglio ai dati: in una certa misura, questo rispecchia i progressi che la fisica ha sperimentato a metà del secolo scorso con la disponibilità dei computer; invece di risolvere manualmente approssimazioni parametriche del comportamento degli elettroni, ora si può ricorrere a simulazioni numeriche delle equazioni differenziali parziali associate. Ciò ha portato a modelli molto più accurati, anche se spesso a scapito dell'interpretazione.

3.2 Latin Hypercube Sampling

Il Latin Hypercube Sampling (LHS) è una tecnica di campionamento statistico che mira a garantire una copertura più uniforme dello spazio dei parametri rispetto al campionamento puramente casuale. Introdotto da McKay, Beckman e Conover nel 1979 [10], il metodo è ampiamente utilizzato quando si devono esplorare funzioni complesse o simulazioni costose in spazi multidimensionali.

La costruzione di un Latin Hypercube prevede che ogni variabile sia divisa in N intervalli equiprobabili e che, per ciascuna di esse, si estragga esattamente un valore da ogni intervallo. Successivamente, i valori campionati vengono combinati in modo che in ogni proiezione unidimensionale compaia una sola realizzazione per intervallo. Questa proprietà garantisce che le distribuzioni marginali delle variabili siano ben rappresentate e che lo spazio venga coperto in maniera più regolare rispetto al campionamento Monte Carlo tradizionale.

Formalmente, fissato il numero di dimensioni d e di campioni N, il Latin Hypercube produce una matrice $N \times d$ i cui elementi appartengono all'intervallo unitario $[0, 1]$. In ciascuna colonna della matrice compaiono esattamente una volta tutti gli intervalli $\left[\frac{i}{N}, \frac{i+1}{N}\right]$ con $i = 0, \ldots, N-1$. Ogni punto del campione occupa così una cella distinta in ciascuna dimensione, realizzando una stratificazione controllata delle proiezioni marginali.

Il vantaggio principale del LHS risiede nell'efficienza: a parità di numero di campioni, esso riduce la varianza stimata delle grandezze d'interesse rispetto al campionamento casuale indipendente. Inoltre, garantendo che ogni intervallo di ogni variabile sia rappresentato, il metodo riduce la possibilità che alcune regioni dello spazio vengano trascurate.

In applicazioni di cosmologia numerica, come nelle simulazioni Quijote [13], il Latin Hypercube è stato utilizzato per generare cataloghi di parametri cosmologici ben distribuiti entro intervalli fisicamente plausibili, permettendo di costruire emulatori accurati per lo spettro di potenza e altre osservabili.

3.3 Multilayer Perceptron

Le reti *deep* più semplici sono chiamate Multilayer Perceptron e sono costituite da più strati di neuroni, ciascuno completamente connesso a quelli dello strato sottostante (da cui ricevono input) e a quelli superiori (che a loro volta influenzano). Con questo tipo di rete si possono superare le limitazioni date da reti lineari. La linearità implica l'ipotesi più debole di monotonicità, ovvero che qualsiasi aumento in una *feature* debba sempre causare un aumento nell'output del modello (se il peso corrispondente è positivo), ma questa non si applica a problemi come il riconoscimento di immagini. Il modo più semplice per superare questa limitazione è sovrapporre molti livelli completamente connessi uno sopra l'altro. Ogni livello fornisce informazione al livello superiore, finché non si genera un output. Possiamo pensare ai primi $N-1$ livelli come alla rappresentazione e al livello finale come al predittore lineare. Ci si accorge però che una struttura tale non ha alcuna vantaggio effettivo rispetto alla semplice struttura formata dai soli input e output layers.

Si identifica con la matrice $X \in \mathbb{R}^{n \times d}$ l'input di n esempi, ognuno con d features e con $H \in \mathbb{R}^{n \times h}$ gli output dello strato nascosto. Attraverso i pesi e i bias dello strato nascosto e di output è possibile calcolare gli output $O \in \mathbb{R}^{n \times q}$ dell'MLP nel seguente modo:

$$H = XW^{(1)} + b^{(1)} \tag{3.1}$$

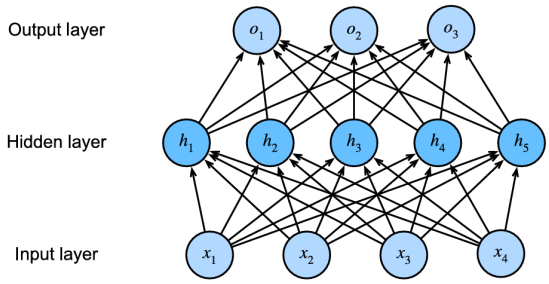


Figura 3.1: Struttura di un MLP con uno strato nascosto. Immagine presa da [18]

$$O = HW^{(2)} + b^{(2)}. \tag{3.2}$$

Ma ciò può essere riscritto come:

$$O = (XW^{(1)} + b^{(1)})W^{(2)} + b^{(2)} = X(W^{(1)}W^{(2)}) + (b^{(1)}W^{(2)} + b^{(2)}) = XW + b. \tag{3.3}$$

Il motivo è che i valori nascosti sono dati da una trasformazione affine degli input, e gli output sono semplicemente una trasormazione lineare degli output degli strati nascosti. Al fine di ottenere un vantaggio, è necessaria l'introduzione di una funzione di attivazione σ non lineare da applicare ad ogni unità nascosta dopo la trasformazione affine. In questo modo non è più possibile collassare il MLP in un modello lineare:

$$H = \sigma(XW^{(1)} + b^{(1)}), \tag{3.4}$$

$$O = HW^{(2)} + b^{(2)}. \tag{3.5}$$

Esempi di funzioni di attivazione sono $ReLu(x) = max(x, 0)$ e le sue varianti, $sigmoid(x) = \frac{1}{1+e^{-x}}$ e $tanh(x) = \frac{1-e^{-2x}}{1+e-2x}$. Nella struttura del MLP utilizzato si è sfruttata la $LeakyReLu$, la quale risolve il problema della $ReLu$, ovvero tutti gli input negativi vengono azzerati, azzerandone anche i gradienti. Ciò viene risolto introducendo una pendenza positiva minore dell'unità per gli input negativi, consentendo ai neuroni di rimanere attivi anche per tali input.

Il funzionamento di un generico MLP si può riassumere nel seguente modo:

1. Forward propagation: i dati fluiscono dallo strato di ingresso all'uscita tramite moltiplicazioni di matrici e applicazioni delle funzioni di attivazione.

2. Activation function: vengono introdotte per rendere il modello non lineare ed avere un vantaggio di apprendimento e risoluzione dei problemi rispetto a modelli più semplici.

3. Training: i parametri $\{W, b\}$ vengono aggiornati minimizzando una funzione di costo tramite stochasti gradient descent o algoritmi adattivi e retropropagazione degli errori. Questo processo viene effettuato attraverso l'ausilio di funzioni di stop, che bloccano l'apprendimento quando l'errore di apprendimento satura.

Nella struttura del MLP vengono effettuati degli accorgimenti per migliorare l'apprendimento e la velocità. La scelta dell'inizializzazione dei pesi è importante, in quanto è necessaria per evitare gradienti troppo piccoli o troppo grandi. L'input viene normalizzato per una maggiore stabilità numerica e per aumentare la velocità di convergenza, oltre a evitare che una feature sia più dominante di altre nell'apprendimento. Viene dunque scelto un ottimizzatore: l'algoritmo che aggiorna i parametri di una rete neurale (pesi e bias) minimizzando la funzione di costo; esso utilizza i gradienti calcolati con la retropropagazione per determinare la direzione e l'ampiezza della variazione dei parametri. Uno dei principali problemi è l'overfitting, ovvero la condizione per cui un modello apprende non solo i pattern generali nei dati di addestramento, ma anche il rumore e le particolarità non generalizzabili. Ne consegue che si hanno prestazioni elevate sul training set e scarse sul test set. Gli accorgimenti per evitare un tale scenario sono dati da:

1. Aumento dei dati di input

2. Regularizzazione: ovvero aggiunta di termini alla funzione di costo per penalizzare pesi troppo grandi
3. Dropout: una disattivazione casuale di neuroni durante l'addestramento per ridurre la co-dipendenza interna
4. Early stopping: interruzione anticipata dell'addestramento quando l'errore sul validation set smette di diminuire

3.3.1 Matrice dell'informazione di Fisher del MLP

Quando si utilizza un modello di machine learning, come un Multilayer Perceptron (MLP) addestrato a mappare i power spectrum in parametri cosmologici, il calcolo della matrice di Fisher prevede come oggetto primario non più il power spectrum teorico, ma la funzione appresa dalla rete neurale che realizza la trasformazione inversa: dai dati osservabili allo spazio dei parametri. La sensibilità del modello rispetto ai parametri si stima attraverso le derivate della funzione appresa dal MLP rispetto agli input. La matrice di Fisher derivata dal MLP quantifica quindi quanto l'architettura addestrata riesca a preservare e sfruttare l'informazione contenuta nei power spectrum, evidenziando eventuali degenerazioni o perdite informative introdotte dal processo di apprendimento.

In modo analogo a eq. 2.28 la Fisher del MLP si può scrivere:

$$F_{MLP} = J_\mu^\top C_{\hat{\theta}}^{-1} J_\mu \tag{3.6}$$

dove $J_\mu = \partial\mu_f/\partial\theta$ è la derivata della media delle uscite della rete rispetto ai parametri, e $C_{\hat{\theta}}$ è la covarianza delle stime compresse.

Per collegare esplicitamente le espressioni delle due FIM è sufficiente applicare la regola della catena [16]:

$$J_\mu; =; \frac{\partial\hat{\theta}}{\partial\theta} = \frac{\partial\hat{\theta}}{\partial P}, \frac{\partial P}{\partial\theta} = J_{\text{comp}}, J_P, \tag{3.7}$$

dove $J_{\text{comp}} = \partial\hat{\theta}/\partial P$ è la jacobiana della rete neurale rispetto all'input $P(k)$. In modo analogo, la covarianza delle stime compresse si ottiene trasformando la covarianza del power spectrum:

$$C_{MLP} = J_{\text{comp}}\, C_P\, J_{\text{comp}}^\top. \tag{3.8}$$

con C_P la covarianza di eq. 2.29 La Fisher per le stime compresse è dunque:

$$F_{MLP} = (J_{\text{comp}} J_P)^\top \left(J_{\text{comp}} C_P J_{\text{comp}}^\top\right)^{-1} (J_{\text{comp}} J_P) \tag{3.9}$$

Mentre la Fisher del power spectrum teorico rappresenta un limite legato alle proprietà intrinseche dell'osservabile, la Fisher associata al MLP è condizionata anche dalla capacità del modello, dalla qualità dell'addestramento e dalla copertura del training set nello spazio dei parametri. Confrontare le due matrici permette dunque di valutare fino a che punto l'algoritmo di machine learning sia in grado di saturare il contenuto informativo teorico, e in quali direzioni parametriche invece si osservino perdite di precisione o amplificazione di degenerazioni.

Si vuole di seguito dare una breve dimostrazione di come si è arrivati all formulazione della FIM del MLP. Siano:

- $\theta = (\theta_1, \ldots, \theta_p)^\top \in \mathbb{R}^p$ il vettore dei parametri (denotato θ_0 quando valutato al punto fiduciale);
- $P(\theta) = (P_1(\theta), \ldots, P_{N_k}(\theta))^\top \in \mathbb{R}^{N_k}$ il vettore del power spectrum valutato sui N_k bin; in particolare $P_0 \equiv P(\theta_0)$;
- $C_P(\theta) \in \mathbb{R}^{N_k \times N_k}$ la matrice di covarianza del power spectrum fiduciale;
- $J_P(\theta) \in \mathbb{R}^{N_k \times p}$ la Jacobiana di P rispetto a θ e in particolare $J_P \equiv J_P(\theta_0)$;

- $f : \mathbb{R}^{N_k} \to \mathbb{R}^m$ il compressore (il MLP che mappa P nelle stime $\hat{\theta}$); si denota $J_{\text{comp}}(P) \in \mathbb{R}^{m \times N_k}$ la Jacobiana di f rispetto a P

Si procede ora a vedere come dalla forma della FIM per il caso gaussiano con $\mu(\theta) = P(\theta)$ si ottenga la FIM per il compressore $f(P)$, attraverso la propagazione tramite la regola della catena e la linearizzazione:

$$C_{MLP} \approx J_{\text{comp}}\, C_P\, J_{\text{comp}}^{\top}, \tag{3.10}$$

$$F_{MLP} \approx (J_{\text{comp}} J_P)^{\top} (J_{\text{comp}} C_P J_{\text{comp}}^{\top})^{-1} (J_{\text{comp}} J_P). \tag{3.11}$$

Si definiscono la media e la covarianza:

$$\mu_{MLP}(\theta) = \mathbb{E}_{P|\theta}[f(P)], \tag{3.12}$$

$$C_{MLP}(\theta) = \text{Cov}_{P|\theta}[f(P)]. \tag{3.13}$$

Si assume che le fluttuazioni del dato P attorno a $\mu_P = P_0$ siano piccole o che f sia sufficientemente regolare nella regione considerata; allora si applica una approssimazione di primo ordine:

$$f(P) \approx f(\mu_P) + J_{\text{comp}}(\mu_P)\,(P - \mu_P), \tag{3.14}$$

dove $J_{\text{comp}}(\mu_P) = \left.\dfrac{\partial f}{\partial P}\right|_{P=\mu_P}$. Sotto questa approssimazione si ha:

$$\mu_{MLP}(\theta) \approx f(\mu_P). \tag{3.15}$$

La matrice di covarianza dei summary compressi è definita come

$$C_{MLP} = \mathbb{E}[(f(P) - \mu_{MLP})(f(P) - \mu_{MLP})^T]. \tag{3.16}$$

Sostituendo l'approssimazione lineare:

$$C_{MLP} \simeq \mathbb{E}\big[J_{\text{comp}}(P - \mu_P)(P - \mu_P)^T J_{\text{comp}}^T\big]. \tag{3.17}$$

Poiché J_{comp} è costante (valutata in μ_P), si può portare fuori dal valore atteso:

$$C_{MLP} \simeq J_{\text{comp}}\, \mathbb{E}\big[(P - \mu_P)(P - \mu_P)^T\big]\, J_{\text{comp}}^T. \tag{3.18}$$

Infine, riconoscendo che $\mathbb{E}[(P - \mu_P)(P - \mu_P)^T] = C_P$, si ottiene:

$$C_{MLP} \simeq J_{\text{comp}}\, C_P\, J_{\text{comp}}^T. \tag{3.19}$$

Per la derivata delle medie rispetto a θ vale la regola della catena (ancora in approssimazione lineare):

$$\frac{\partial \mu_{MLP}}{\partial \theta} \approx J_{\text{comp}}(\mu_P)\, \frac{\partial \mu_P}{\partial \theta} = J_{\text{comp}}(\mu_P)\; J_P(\theta). \tag{3.20}$$

Valutando al punto fiduciale θ_0 e scrivendo $J_{\text{comp}} \equiv J_{\text{comp}}(P_0)$, $J_P \equiv J_P(\theta_0)$, $C_P \equiv C_P(\theta_0)$ si ottiene:

$$\left.\frac{\partial \mu_{MLP}}{\partial \theta}\right|_{\theta_0} \approx J_{\text{comp}} J_P, \tag{3.21}$$

$$C_{MLP} \approx J_{\text{comp}} C_P J_{\text{comp}}^{\top}. \tag{3.22}$$

Applicando la formula della Fisher :

$$F_{MLP} \approx \left(\frac{\partial \mu_t}{\partial \theta}\right)^{\top} C_t^{-1} \left(\frac{\partial \mu_t}{\partial \theta}\right), \tag{3.23}$$

si ottiene la formula desiderata:

$$F^{MLP} \approx (J_{\text{comp}} J_P)^{\top} \big(J_{\text{comp}} C_P J_{\text{comp}}^{\top}\big)^{-1} (J_{\text{comp}} J_P). \tag{3.24}$$

Capitolo 4

Analisi e discussione

4.1 Metodologia

Il focus del lavoro viene rivolto al $P(k)$ a redshift $z = 0$, generato attraverso i parametri cosmologici $[\Omega_c, \Omega_b, h, n_s, \sigma_8]$ per $k \in [0.008; 0.1]$ $h\ Mpc^{-1}$, pertanto si vogliono studiare le strutture cosmiche su grandi scale e la scala presa in considerazione è la scala lineare. I valori dei parametri cosmologici vengono presi dalle osservazioni Planck2018 [7]:

$$\Omega_b = 0.0493 \pm 0.0010 \quad \Omega_c = 0.264 \pm 0.007 \quad h = 0.6736 \pm 0.0054$$

$$n_s = 0.9649 \pm 0.0042 \quad \sigma_8 = 0.8111 \pm 0.0060$$

Si procede generando degli intervalli intorno ai parametri sui quali poi verrano estratti dei valori a campionamento per generare i $P(k)$ di addestramento e test del MLP. Vengono generati due tipi di intervalli al fine di osservare come variano gli output del MLP al variare della grandezza degli intervalli. L'intervallo più largo è dato dagli intervalli usati nelle Quijote simulations [13]:

$$\Omega_c \in [0.1, 0.5] \quad \Omega_b \in [0.03, 0.07] \quad h \in [0.5, 0.9] \quad n_s \in [0.8, 1.2] \quad \sigma_8 \in [0.6, 1.0]$$

Un secondo intervallo invece prevede di prendere un bound del 10% attorno ai valori fiduciali. Successivamente, si riprendono questi due intervalli rispettivamente come *prior ampi* e *prior stretti*.

Successivamente, tali intervalli vengono campionati con la tecnica del Latin Hypercube Sampling sez. 3.2, generando una matrice di dimensione $N \times 5$, in cui ciascuna riga rappresenta un set distinto di parametri per le simulazioni.

Quindi vengono generati N diversi $P(k)$ attraverso eq. 2.21 come liste di lunghezza $N_k = 128$, andando a coprire l'intervallo dei k preso in considerazione, quindi con $k_{min} = 0.008$ e $K_{max} = 0.1$. I $P(k)$ vengono generati attraverso l'ausilio di una funzione built-in nella libreria *jax-cosmo* [19], una libreria cosmologica differenziabile e accelerata tramite GPU in JAX. N deve essere un numero abbastanza grande al fine di evitare situazioni di underfitting; per lo sviluppo dell'attuale lavoro sono state campionate 20000 diverse simulazioni dei power spectrum. Pertanto, si ottengono due matrici $N \times 128$ e $N \times 5$, le quali identificano rispettivamente i power spectra e i parametri cosmologici ad essi associati.

Una volta generati i power spectra, affinchè possano essere utilizzati come input per l'MLP devono essere standardizzati come segue:

$$x' = \frac{x - \mu}{\sigma}, \tag{4.1}$$

con μ la media sulle colonne e σ la deviazione standard, in maniera tale che ogni feature abbia poi media nulla e varianza pari a uno. Per l'addestramento del MLP è necessario dare in input anche i parametri cosmologici associati ai singoli $P(k)$, pertanto anche questi vanno normalizzati.

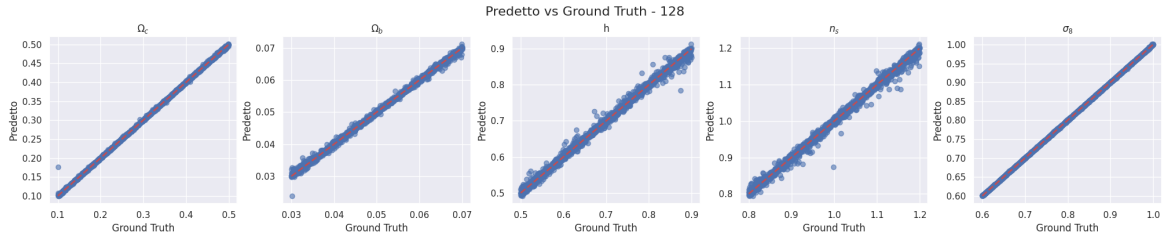
Una volta diviso il dataset in tre parti, una per l'addestramento (80%), una per la validazione (10%) e una per il test (10%), si è pronti ad addestrare il MLP.

Il modello è una rete neurale multistrato fully connected. La rete è composta da una sequenza di strati, ciascuno formato da un certo numero di neuroni che ricevono in ingresso i valori dallo strato precedente e producono un vettore di uscita che diventa l'ingresso dello strato successivo.

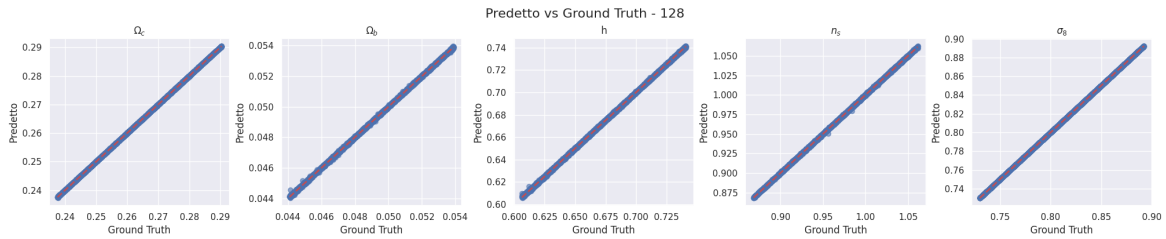
La struttura della rete è definita in modo flessibile attraverso una lista di dimensioni, che stabilisce il numero di neuroni per strato. Tutti gli strati, tranne l'ultimo, sono seguiti da una funzione di attivazione non lineare del tipo leakyReLU, che introduce una piccola pendenza (slope = 0.5) anche per i valori negativi, evitando così la completa inattivazione dei neuroni e migliorando la propagazione del gradiente durante l'addestramento. L'ultimo strato è invece lineare, cioè fornisce l'uscita del modello senza alcuna trasformazione non lineare, così che i valori prodotti rappresentino direttamente le quantità da prevedere. Il modello della funzione di attivazione è stato estrapolato dal lavoro di A. Bairagi, B. Wandelt e F. Villaescusa-Navarro [4], i quali vogliono comprendere quante simulazioni siano necessarie per allenare un modello di machine learning al fine di estrapolare dell'informazione dalle summary statistic del campo di densità cosmologico.

Il modello viene costruito variando la sua profondità, cioè il numero di strati nascosti, mantenendo costante la larghezza di ciascun livello, ovvero il numero di neuroni per strato. In questo modo si può studiare come la capacità della rete di apprendere e generalizzare dipenda dalla complessità della sua architettura. Ogni configurazione di profondità viene addestrata separatamente, partendo da pesi iniziali casuali.

Durante l'addestramento, i dati vengono suddivisi in piccoli sottoinsiemi, o batch, di dimensione pari a 16 ciascuno che vengono utilizzati per aggiornare progressivamente i parametri del modello.



(a) 128 - prior ampio



(b) 128 - prior stretto

Figura 4.1: Plot Ground truth vs valore predetto per un modello esemplificativo a 128 neuroni per strato e 6 strati nascosti.

La funzione di perdita utilizzata misura la differenza tra le previsioni della rete e i valori reali, calcolando per ciascun output l'errore quadratico medio e poi combinandoli attraverso una somma logaritmica [11]:

$$L = \sum_i \ln \left(\frac{1}{N_{\text{batch}}} \sum_{n=0}^{N_{\text{batch}}-1} \left(\theta_i^{(n)} - \hat{\theta}_i^{(n)} \right)^2 \right) \tag{4.2}$$

Questa formulazione permette di trattare in modo bilanciato vari output con scale di grandezza differenti. In cosmologia l'obiettivo è stimare contemporaneamente sia parametri cosmologici sia parametri

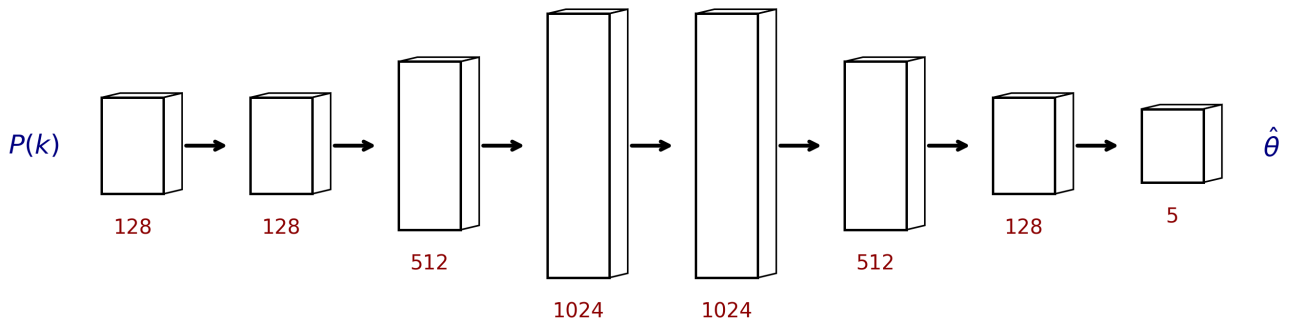


Figura 4.2: Struttura della rete piramidale

astrofisici legati a processi come il feedback stellare o la formazione delle galassie. Questi parametri presentano sensibilità molto diverse rispetto ai dati osservati.

Utilizzare una funzione di loss basata sulla somma aritmetica degli errori comporta che la rete neurale concentri gran parte dello sforzo di apprendimento su alcuni parametri, trascurando quelli che presentano una sensibilità inferiore, con stime dei parametri cosmologici meno accurate. L'operazione di logaritmo consente di dare un peso più equilibrato a tutti i parametri durante l'addestramento della rete. In particolare, viene dato un peso maggiore agli errori più piccoli e minore a quelli più grandi. Nel caso di intersse i parametri astrofisici non rientrano, ma esiste una differenza consistente tra i vari parametri cosmologici in termini di errore.

Passando al processo di ottimizzazione, questo si basa su una discesa del gradiente adattiva, in cui il tasso di apprendimento diminuisce gradualmente nel tempo per favorire la stabilità della convergenza. Dopo ogni ciclo di addestramento, la perdita viene calcolata sia sui dati di addestramento sia su un insieme di validazione indipendente, così da monitorare le prestazioni del modello e prevenire l'overfitting. L'ottimizzatore adottato è Adam con un learning rate iniziale pari a 0.001 che decade con un tasso $\gamma = 0.9$ ogni 10 epoche con l'ausilio di uno scheduler.

Per evitare un addestramento eccessivo o inefficiente, il modello implementa un meccanismo di arresto anticipato. Questo interrompe la procedura quando la perdita di validazione smette di migliorare per un certo numero di iterazioni, quando la curva di errore mostra segni di saturazione oppure quando la sua variazione diventa trascurabile nel tempo. In questo modo il modello conserva la versione dei parametri che ha ottenuto le migliori prestazioni in validazione. In particolare il modello implementa una forma di early stopping basata su tre criteri: la mancata riduzione della perdita di validazione per un certo numero di epoche (patience), la saturazione della curva di validazione (valutata con la varianza su una finestra scorrevole), oppure la stabilità della pendenza della perdita (verificata tramite una regressione lineare dei valori recenti).

Allenate le singole architetture, esse vengono validate sia attraverso l'errore nella predizione per il dataset di test rispetto alle epoche di allenamento, sia attraverso un confronto tra valori reali e valori predetti dal MLP (fig. 4.1).

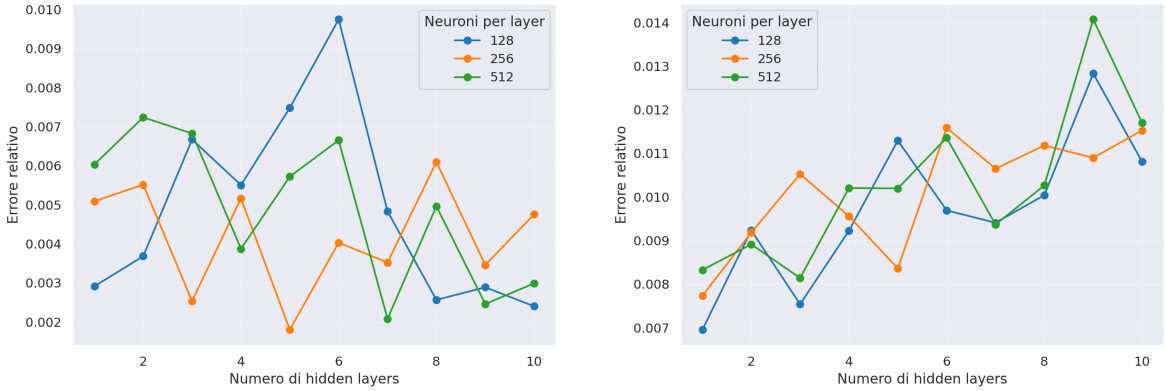
Successivamente, viene fornita una misura più corretta della validità delle predizioni dei parametri da parte delle diverse architetture del MLP, attraverso il calcolo della FIM come descritto in sez. 3.3.1. In questo è possibile vedere non solo come interagiscono i parametri nel power spectrum teorico, ma anche come variano in base all'architettura. Oltre alle architetture a numero fisso di neuroni per strato nascosto viene considerata un'architettura piramidale, al fine di effettuare un confronto con strutture con uguale o maggior numero di strati nascosti, ma minor numero di parametri, ovvero la somma dei bias e dei pesi imparati dal modello. Partendo da 128 neuroni lo strato successivo aumenta il numero di neuroni fino ad arrivare a 1024, dopodichè si riparte da 1024 per riarrivare a 128. Gli strati nascosti hanno quindi una struttura simmetrica esplicitata come segue:

$$[128, 512, 1024, 1024, 512, 128]$$

Il numero totale di parametri è 2 248 581 e, tra i modelli analizzati, è inferiore alla sola struttura con 10 strati nascosti da 512 neuroni per strato. Questa struttura è stata identificata in quanto in pochi strati nascosti riesce ad avere una grande complessità data dal numero di parametri, quindi è in grado di garantire un miglior equilibrio tra capacità di rappresentazione e stabilità dell'addestramento quando il numero di simulazioni è limitato. Una rete piramidale riduce progressivamente il numero di neuroni negli strati successivi, comprimendo l'informazione e favorendo l'estrazione di caratteristiche globali e robuste dai dati cosmologici. Questo approccio contrasta l'overfitting tipico delle architetture con numero crescente o costante di neuroni, che tendono ad apprendere rumore quando le simulazioni sono poche [17].

4.2 Approssimazione lineare

Il calcolo della FIM indicata in sez. 3.3.1 si basa su un'assunzione di linearità del funzionamento del MLP intorno al power spectrum fiduciale, che implica che si considera una covarianza indipendente dai parametri, in quanto nell'equazione della linearizzazione non si tiene conto della correlazione tra i parametri. Tale assunzione deve essere prima di tutto provata prendendo in considerazione le simulazioni in un piccolo intorno del power spectrum fiduciale (variazione dell'1%) e verificando l'errore relativo dei valori ottenuti da eq. 3.14. Tale verifica è stata effettuata per tutte le architetture prese



(a) Errore relativo con prior ampi

(b) Errore relativo con prior stretti

Figura 4.3: Errore relativo medi dei parametri predetti dal MLP rispetto ai corrispettivi valori predetti attraverso la linearizzazione all'aumentare del numero di strati nascosti. Le incertezze sui valori non superano 0.01

in considerazione e per ciascuna l'errore relativo viene calcolato come segue:

$$e_r = \frac{\hat{\theta} - f(P)}{\hat{\theta}}, \tag{4.3}$$

con $\hat{\theta} \in \mathbb{R}^5$ è il vettore dei cinque paramteri ricavati attraverso il MLP, mentre $f(P) \in \mathbb{R}^5$ è il vettore dei cinque parametri ricavato dalla linearizzazione eq. 3.14.

Si ha un set di $N_s = 2000$ simulazioni dei power spectrum intorno al punto fiduciale, le quali vengono sottoposte alle varie architetture già addestrate. Per ognuna si calcola l'errore relativo eq. 4.3 e si effettua la media sui 5 parametri e tale valore è quello riassunto in fig. 4.3.

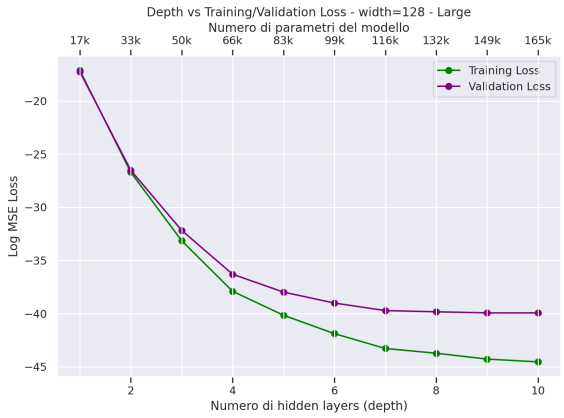
Ciò che si ottiene è che si ha che per ogni struttura l'errore relativo risulta minore dell'effettiva variazione rispetto al power spectrum fiduciale o entro l'incertezza calcolata sull'errore relativo come riportato in fig. 4.3. La distribuzione dell'errore relativo risulta diversa tra le architetture allenate con vincoli diversi sui parametri: un andamento più costante è rilevabile nei prior ampi, mentre è riconoscibile un aumento lungo la profondità nelle architetture allenate con vincoli più stretti. Generalemente, l'errore relativo risulta più piccolo per i prior ampi; ciò è spiegabile col fatto che nelle architetture allenate con i prior stretti l'intorno del power spectrum fiduciale è noto in maniera più accurata, permettendo

di sviluppare funzioni non lineari che interpolino in maniera più accurata i dati in tale regione, quindi l'approssimazione lineare perde leggermente di significato rispetto alle architetture allenate sui prior ampi.

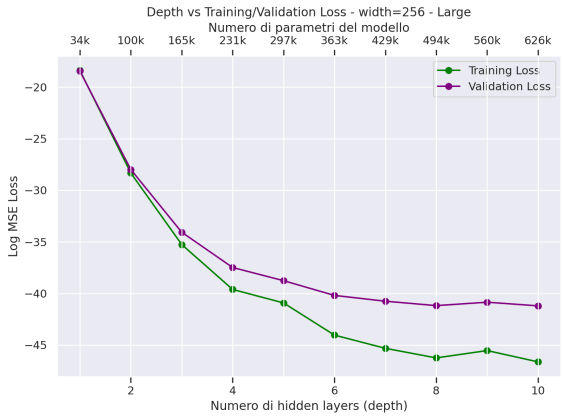
Esposte alcune considerazioni sui grafici, è bene sottolineare che gli errori relativi rientrano nell'aspettativa rispetto alla variazione sul power spectrum fiduciale. Si può quindi affermare che il calcolo della FIM sviluppato in sez. 3.3.1 è in accordo, nell'intorno dello spettro fiduciale preso in considerazione, con la reale FIM del MLP, la quale richiederebbe un elevato costo computazionale rispetto alla formulazione esposta e non utile ai fini della presente tesi.

4.3 Analisi sulla profondità

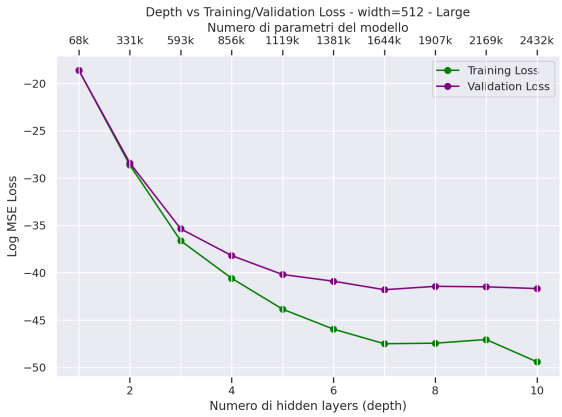
Costruito l'MLP è bene capire quale sia la struttura ottimale per ottenere il maggiore numero di informazioni possibili con l'errore minimo. Vengono effettuati dei test sul numero di layer ottimale, fissando il numero di neuroni per layer. I test effettuati sono stati svolti per larghezze degli strati pari a 128, 256 e 512. Per ciascuna larghezza si sono testati modelli con uno strato nascosto fino a dieci strati nascosti.



(a) 128 neuroni per layer



(b) 256 neuroni per layer



(c) 512 neuroni per layer

Figura 4.4: Andamento della validation e train loss all'aumentare del numero di strati nascosti e conseguentemente del numero dei parametri della rete. *Prior ampi*

Dai grafici si ricava come la curva *depth vs loss* satura dal quinto strato nascosto in poi. Una struttura ottimale dal punto di vista computazionale per l'estrazione di informazione dei parametri ha quindi bisogno di cinque o sei strati nascosti, dopodichè il costo computazionale aumenta in modo esponenziale rispetto all'informazione in più in grado di essere estratta dal MLP. Inoltre, dal confronto delle curve

depth vs loss per varie larghezze si nota come il numero di neuroni influisce in maniera minima all'errore, con un miglioramento di quest'ultimo all'aumentare del numero di neuroni.

I test sono stati ripetuti per dataset creati a partire da intervalli sui parametri cosmologici più o meno ampi. Un primo test è stato effettuato con gli intervalli usati nel progetto delle Quijote simulations [13], già definiti nella metodologia. Tali intervalli sono larghi rispetto a quelle che sono le ultime stime dei parametri derivanti da osservazioni come Planck 2018, ma tale ampiezza è utile per addestrare il MLP in modo tale che possa rispondere ad una più ampia gamma di spettri e non incorra nel problema dell'overfitting. Successivamente si è ristretto l'intervallo al 10% del valore dei parametri dati dalle osservazioni di Planck2018 [7], con il fine di vedere come varia la predizione del modello sottoponendolo a un dataset meno ampio.

4.3.1 Prior ampi

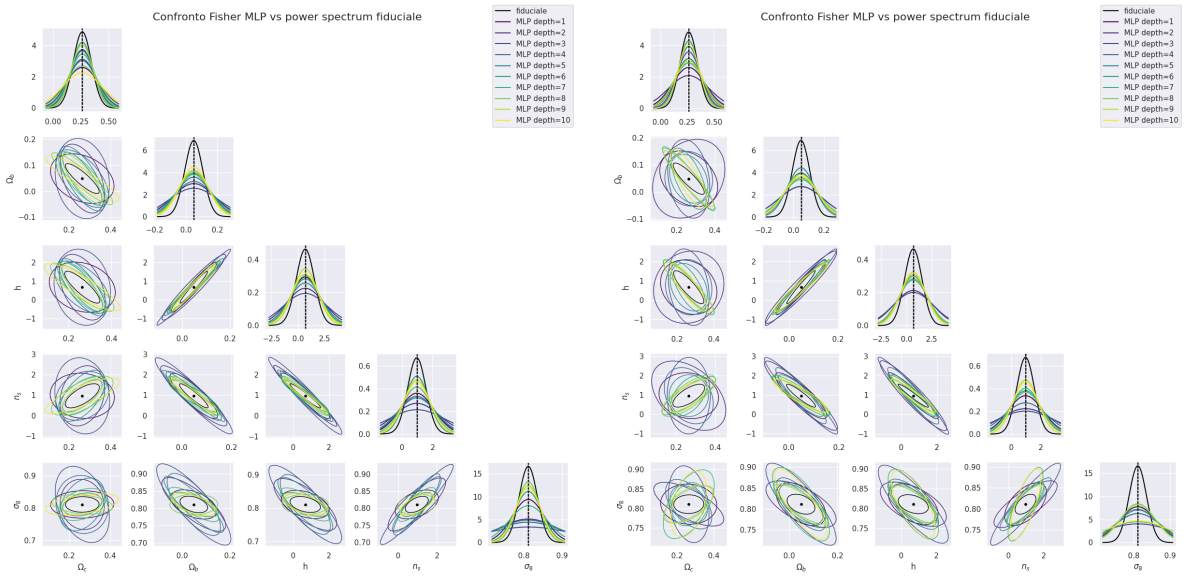
Il paragone tra le *loss* dei vari modelli, in realtà dà ben poche informazioni riguardo all'output del MLP e alla sua connessione con il power spectrum fiduciale e i parametri ad esso associati. Per avere una comprensione di tale aspetto è necessario confrontare la FIM del power spectrum fiduciale con quella del MLP. In questo modo è possibile capire quanto bene il MLP saturi l'informazione a disposizione e come si comporta per i diversi parametri.

Nonostante i vari modelli da cinque strati nascosti e più saturino la loss essi rispondono in maniera differente alla stima dei parametri e ciò è ricavabile dall'analisi delle FIM dei diversi modelli. In alcuni casi, profondità intermedie (da 4 a 7 strati) forniscono in apparenza stime più precise di profondità maggiori. Una possibile spiegazione è data dal rumore statistico presente in addestramento con un numero finito di campioni che si rispecchia poi nella FIM delle varie architetture, quindi piccole fluttuazioni possono far sembrare un modello migliore. In altre parole esiste una stocasticità intrinseca all'addestramento dei diversi modelli che permetterebbe di ottenere risultati diversi riallenando più volte la stessa architettura con lo stesso dataset di addestramento, ma variando le inizializzazioni. Un altro aspetto è dato dalle diverse profondità che possono enfatizzare aspetti differenti dello spettro in quella che risulta una specializzazione per parametro. La motivazione più convincente rimane la prima e verrà discussa più approfonditamente in una sezione successiva.

Tali considerazioni sono visibili in fig. 4.5, dove viene presentata una rappresentazione visiva della covarianza dei parametri stimati, ottenuta come l'inverso della FIM. Le ellissi rappresentano le regioni di confidenza 1σ. Più sono piccole e strette, maggiore è la precisione nella stima del parametro. Inoltre, l'orientamento delle ellissi mostra come i parametri sono correlati tra loro: se le ellissi sono inclinate, allora i parametri sono degeneri, ovvero possiedono una dipendenza reciproca, nel senso che i parametri hanno effetti simili sulle osservazioni; se le ellissi hanno una forma quasi circolare, i parametri risultano essere poco correlati.

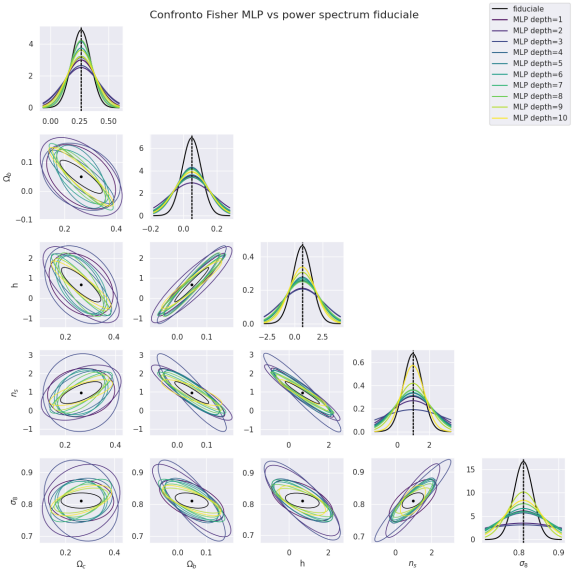
Alcuni parametri, come σ_8 e h, hanno effetti forti e globali e risultano più facili da stimare, mentre altri, Ω_b e n_s ad esempio, hanno effetti sottili e degeneri e pertanto richiedono reti più profonde e un allenamento del modello con simulazioni generate su una più ampia scala di grandezza di k, rispetto a quella usata. Emerge inoltre che i parametri sono correlati tra di loro in maggiore accordo alla correlazione fiduciale quanto più è profonda la rete, senza eccessivi mutamenti per architetture che hanno già saturato la loss. Si vede pertanto come Ω_c e Ω_b siano correlati negativamente, così come lo è con h, il quale ha una grande correlazione con Ω_b. n_s, invece, è fortemente anticorrelato con i due parametri appena menzionati e presenta una robusta correlazione positiva con σ_8, debolemtne anticorrelato con gli altri parametri. Non si vede invece la forte correlazione tra Ω_c e σ_8 in quanto l'effetto di questo parametro è ben visibile per scale più piccole rispetto a quelle prese in considerazione. Dall'analisi della covarianza dei parametri in funzione della profondità delle diverse strutture del MLP si osservano tre regimi:

1. Profondità bassa (1–3 strati): la rete cattura solo informazione globale. Le ellissi sono larghe e le degenerazioni tra parametri, ad esempio Ω_c-σ_8 e n_s-σ_8 risultano sovrastimate.



(a) 128 neuroni per layer

(b) 256 neuroni per layer



(c) 512 neuroni per layer

Figura 4.5: Matrice di covarianza dei parametri per i *Prior ampi* del MLP confrontata con quella dello spettro fiduciale. Le strutture del MLP variano da 1 a 10 strati nascosti.

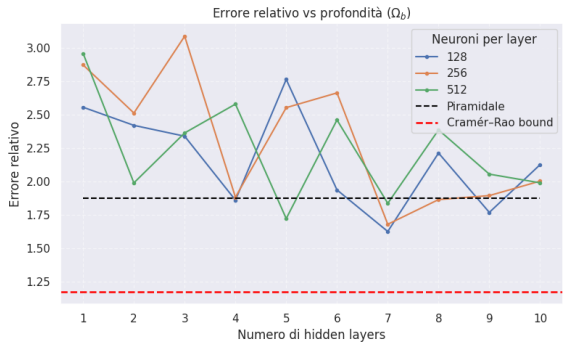
2. Profondità intermedia (4–6 strati): La rete inizia a distinguere dettagli della forma dello spettro, riducendo le degenerazioni e stimando bene parametri più semplici come h e σ_8.

3. Profondità alta (7–10 strati): la rete si avvicina alla precisione del metodo fiduciale. Le correlazioni residue coincidono con quelle fisiche intrinseche dello spettro di potenza, non introdotte dal modello.

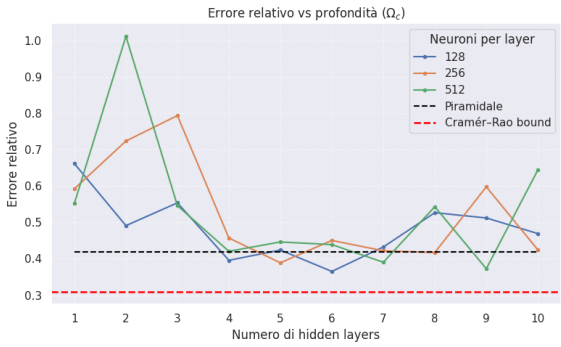
Strettamente collegato al discorso delle FIM è interessante vedere l'andamento dell'errore associato ai parametri predetti (fig. 4.5).

$$\sigma_i = \sqrt{F_{ii}^{-1}} \tag{4.4}$$

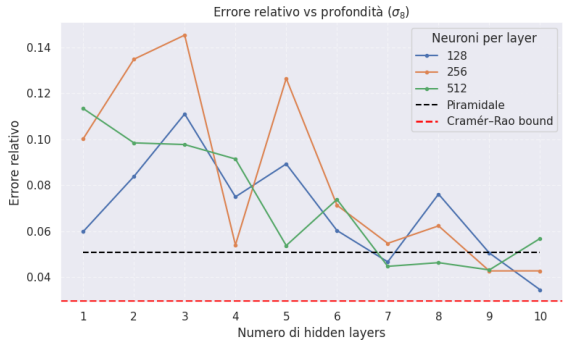
Dalle figure si nota un comportamento caotico, ma si riesce ad individuare un andamento al ribasso dell'errore all'aumentare della profondità della rete. Ad eccezione di Ω_c la rete piramidale risulta sempre dalle strutture con errore relativo minore; presenta pertanto una migliore capacità di individuare i parametri cosmologici, superata solamente dalla struttura a nove strati nascosti da 512 neuroni.



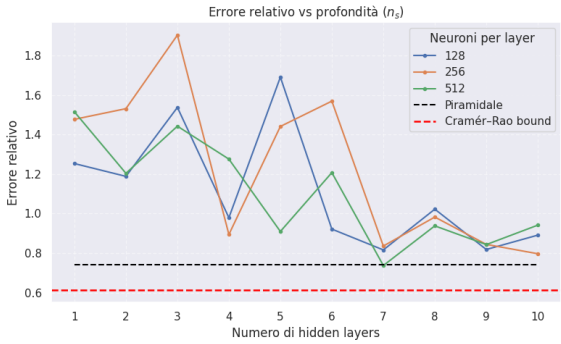
(a) Ω_b



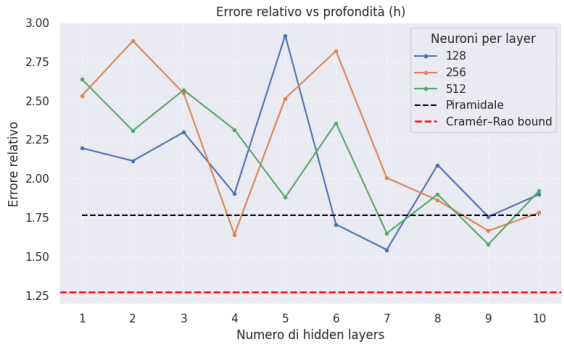
(b) Ω_c



(c) σ_8



(d) n_s



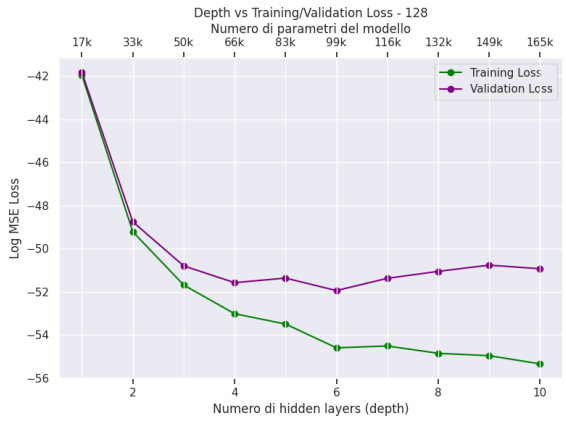
(e) h

Figura 4.6: Andamento dell'errore relativo ai parametri ricostruiti dalla rete neurale all'aumentare del numero di strati nascosti. *Prior ampi*

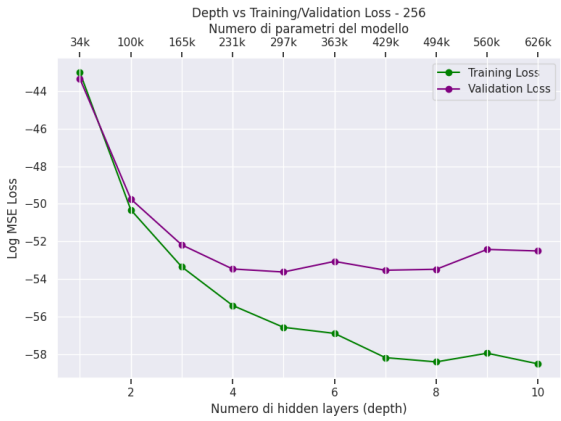
Dal Confronto con il Cramér-Rao bound si evince come le architetture riescano ad avvicinarsi ad esso con l'aumentare della profondità. La stessa architettura piramidale è tra le architetture che si avvicinano maggiormente al limite teorico per tutti e cinque i parametri, comportando che essa riesce ad avere dei vincoli robusti a variazioni significative del segnale di input, catturando pertanto la struttura generale del problema fisico e non caratteristiche locali.

4.3.2 Prior stretti

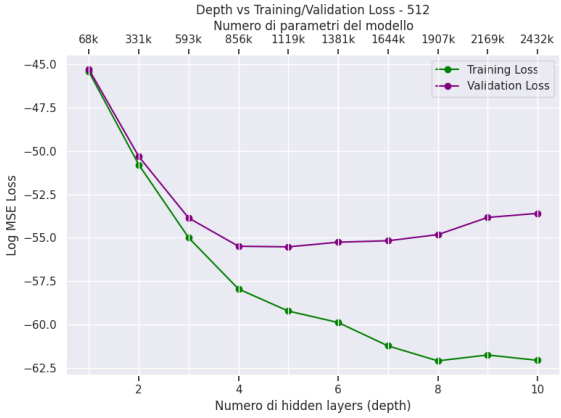
Successivamente, si sono ristretti gli intervalli di campionamento dei parametri cosmologici ad un errore del 10% attorno al valore dei parametri dato dalle osservazioni Planck2018 [7]. In questo modo, si ha un riscontro di come anche il dataset di training influisca nella saturazione dell'informazione di Fisher. Non si è ristretta ulteriormente la larghezza degli intervalli in quanto il modello perderebbe di generalità e sarebbe fortemente influenzato dalla forma del set di dati di allenamento, rivelandosi pertanto un *cattivo* predittore per variazioni più pronunciate rispetto ai valori fiduciali.



(a) 128 neuroni per layer



(b) 256 neuroni per layer



(c) 512 neuroni per layer

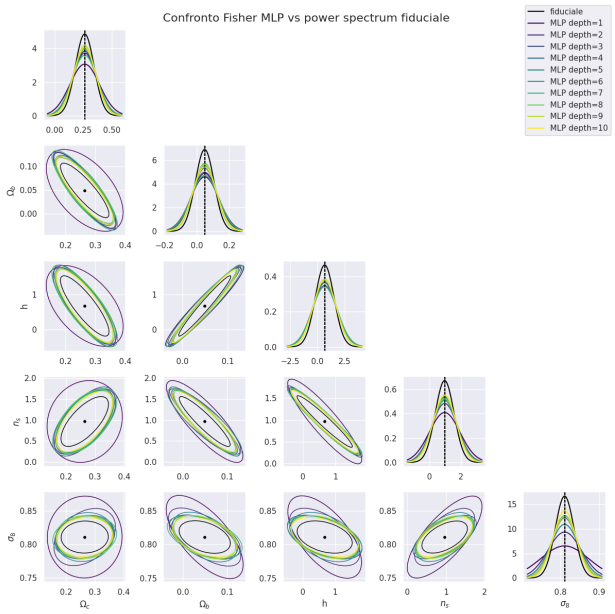
Figura 4.7: Andamento della validation e train loss all'aumentare del numero di strati nascosti e conseguentemente del numero dei parametri della rete. *Prior stretti*

Rispetto alle strutture allenate sui prior ampi si notano due principali differenze: un sostanziale abbassamento della validation loss in tutte le strutture e un lieve rialzo della curva per profondità alte. La prima differenza è spiegabile in quanto per intervalli più piccoli la mappa $P(k) \to \theta$ diventa più lineare e meno complessa. Inoltre per come è calcolata la loss eq. 4.2, essa riflette la varianza dell'output; ne consegue che se gli output variano su un range più piccolo, anche gli errori saranno più piccoli. Le loss in fig. 4.7 sono normalizzate alle varianze.

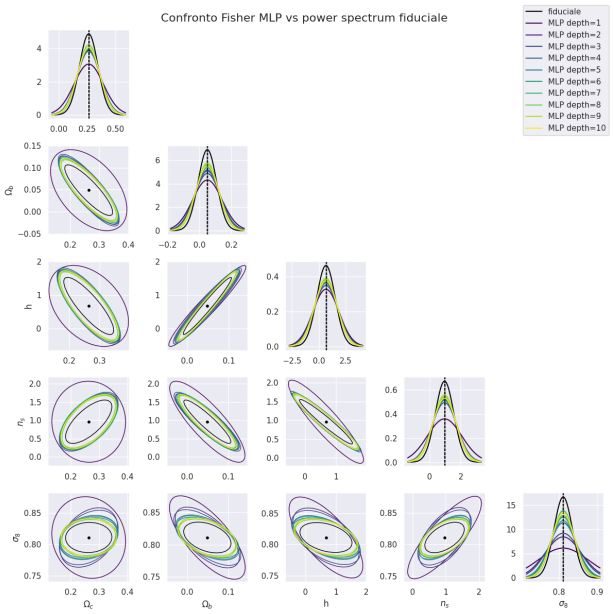
Con intervalli più stretti sui dati si ha una ricchezza maggiore d'informazione attorno ai valori veri dei parametri e come si osserva in fig. 4.9, ciò viene riflesso anche negli errori relativi associati ai parametri: in tutti e cinque i casi gli errori degli intervalli al 10% risultano globalmente minori e con un andamento più stabile lungo la profondità della rete rispetto agli errori relativi agli intervalli *ampi*. Le stesse ellissi nei grafici delle FIM, mostrano una maggiore saturazione dell'informazione.

Riprendendo il fatto che la funzione da apprendere sia quasi lineare vicino al fiduciale, ciò permette di dare una spiegazione alla seconda differenza: modelli troppo complessi hanno molta più capacità computazionale di quella necessaria e pertanto si incorre nell'overfitting per architetture molto profonde. Reti oltre 5 strati entrano pertanto in un regime dove memorizzano rumore numerico o piccole fluttuazioni, i gradienti si indeboliscono e l'ottimizzazione peggiora e l'early stopping scatta prima, trovando minimi più alti di quelli ottenibili senza early stopping.

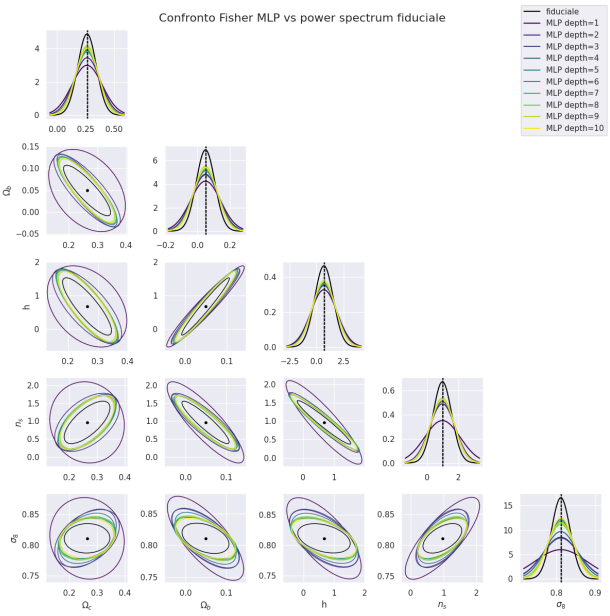
Mettendo assieme le osservazioni effettuate finora, si evince come una concentrazione maggiore nella distribuzione dei dati permetta di ricavare un modello predittivo il più vicino al teorico, ma, allo stesso tempo, incrementa la possibilità di overfitting per strutture profonde.



(a) 128 neuroni per layer



(b) 256 neuroni per layer

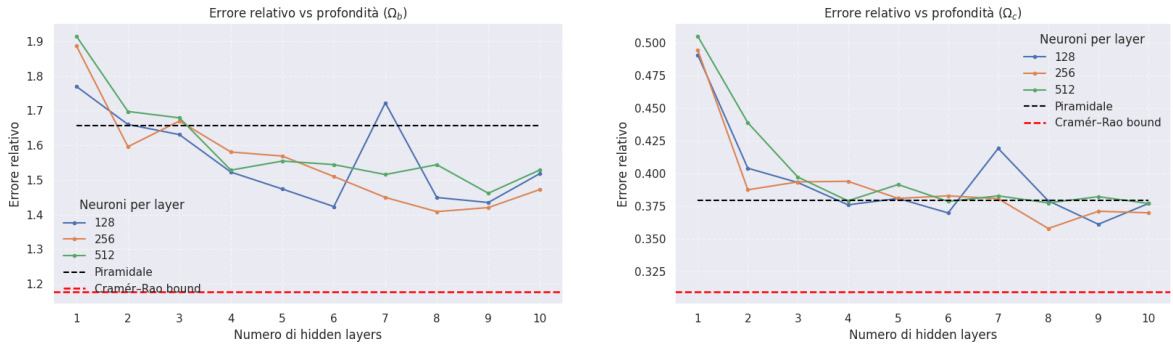


(c) 512 neuroni per layer

Figura 4.8: Matrice di covarianza dei parametri per *Prior stretti* del MLP confrontata con quella dello spettro fiduciale. Le strutture del MLP variano da 1 a 10 strati nascosti.

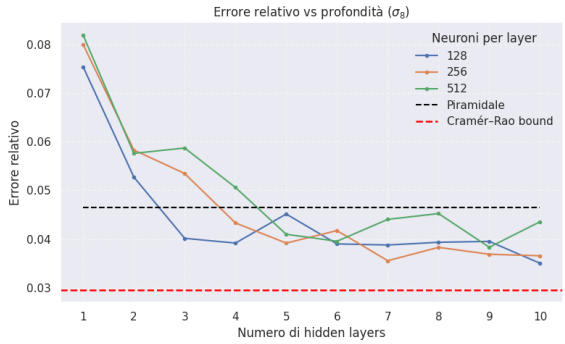
Da queste considerazioni si ricava che la struttura ottimale è data da una configurazione intermedia con cinque o sei strati nascosti, la quale permette di ricavare la maggior quantità d'informazione possibile rimanendo in una situazione dove la soluzione al problema fisico rimane generale, ovvero per la quale il bias costituito dai dati di allenamento è attenuato.

Se si passa all'analisi delle ellissi delle FIM delle varie architetture fig. 4.8, si osserva come ad un primo impatto vengono rispettate le correlazioni tra i parametri discusse nella sezione precedente, in quanto la fisica non varia, ma le ellissi risultano meglio allineate e strette rispetto alle precedenti. A diverse profondità le ellissi risultano più coerenti fra loro e con il fiduciale: questo è un segno che il Fisher ricostruito cattura meglio le correlzioni tra i parametri. Inoltre, le ellissi si presentano più strette e la varianza per profondità alte quasi satura quella fiduciale. Se per i *Prior ampi*, per strutture alla saturazione non vi era un proprio ordine, in questo caso si nota una disposizione più regolare delle

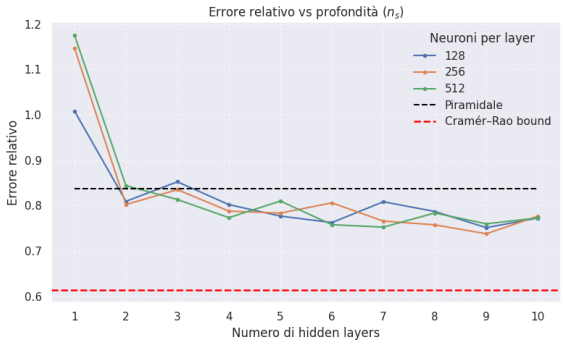


(a) Ω_b

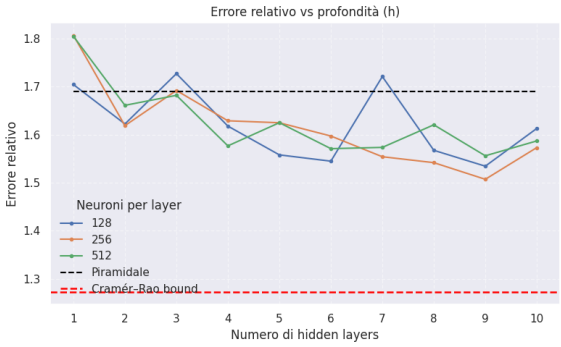
(b) Ω_c



(c) σ_8



(d) n_s



(e) h

Figura 4.9: Andamento dell'errore relativo ai parametri ricostruiti dalla rete neurale all'aumentare del numero di strati nascosti. *Prior stretti*

curve che mostra il profilo di convergenza con la profondità.

Analizzando invece gli errori associati ai parametri ricavati tramite eq. 4.4 si possono notare dei profili più ordinati in fig. 4.9 rispetto ai *Prior ampi*. Si nota un chiaro decremento degli errori all'aumentare della profondità della rete. La struttura piramidale predice meglio Ω_c, ma perde capacità predittiva rispetto alle altre strutture sui parametri più piccoli, ovvero Ω_b e h, sebbene rimanga in accordo con i valori ricavati tramite i *prior ampi*. Riguardo al Cramér-Rao bound si nota che non viene raggiunta la saturazione, ma esiste un limite superiore ad esso dato dal numero finito di parametri della rete. In rapporto ai *Prior ampi* l'errore relativo ai parametri risulta inferiore mediamente del 30% per ciascun parametro con una massimo per Ω_b del 45%; in particolare le architetture che hanno ottenuto un miglioramento maggiore sono quelle con un numero di neuroni per strato pari a 128. Da questo si comprende come le architetture poco complesse, ovvero con un numero minore di parametri lavorino meglio per problemi più specializzati, mentre architettture, come quella piramidale, con un numero consistente di parametri riescono a mantenere la loro capacità di estrazione informativa anche con

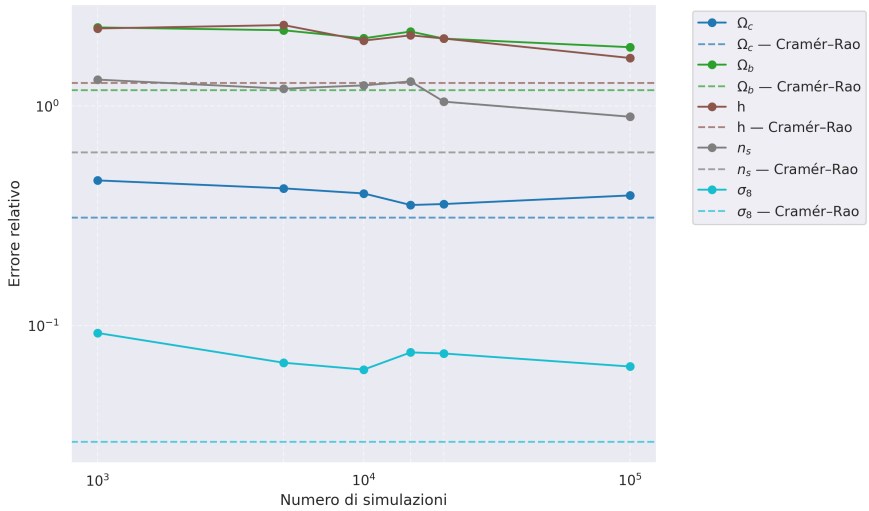


Figura 4.10: Andamento, rispetto al numero simulazioni di addestramento, degli errori della FIM per il modello piramidale con prior larghi.

problemi più generalizzati.

4.4 Numero di simulazioni

L'analisi svolta finora si basa sul confronto di diverse architetture per profondità e larghezza degli strati nascosti, ma senza valutare quale sia il numero di simulazioni ottimale per l'addestramento. Fissata un'architettura si vuole pertanto osservare come variano gli errori sui parametri al variare del numero di simulazioni di training.

Considerando l'architettura piramidale e le simulazioni generate nei prior ampi, si è generato un LH di 10^5 simulazioni dal quale vengono campionati in modo casuale 5 sottoinsiemi da 1 000, 5000, 10000, 15000 e 20000 simulazioni. Ciascun sottoinsieme addestra la stessa architettura e si calcola la FIM come nelle sezioni precedenti, ottenendo il grafico in fig. 4.10. Dalla figura non si nota una sostanziosa decrescita degli errori, sebbene sia percepibile un andamento globale al ribasso. Ciò che più è evidente è l'andamento altalenante che sembra supporre una stocasticità intrinseca nell'addestramento della rete. Per dimostrare ciò si è ripetuto l'allenamento per 1 000 simulazioni 50 volte ottenendo una distribuzione degli errori esposta in fig. 4.11. Tale distribuzione dei valori conferma la stocasticità dell'addestramento. Quindi, per avere una stima più corretta degli errori della FIM del MLP è necessario svolgere più simulazioni per la stessa architettura con lo stesso set di dati di addestramento.

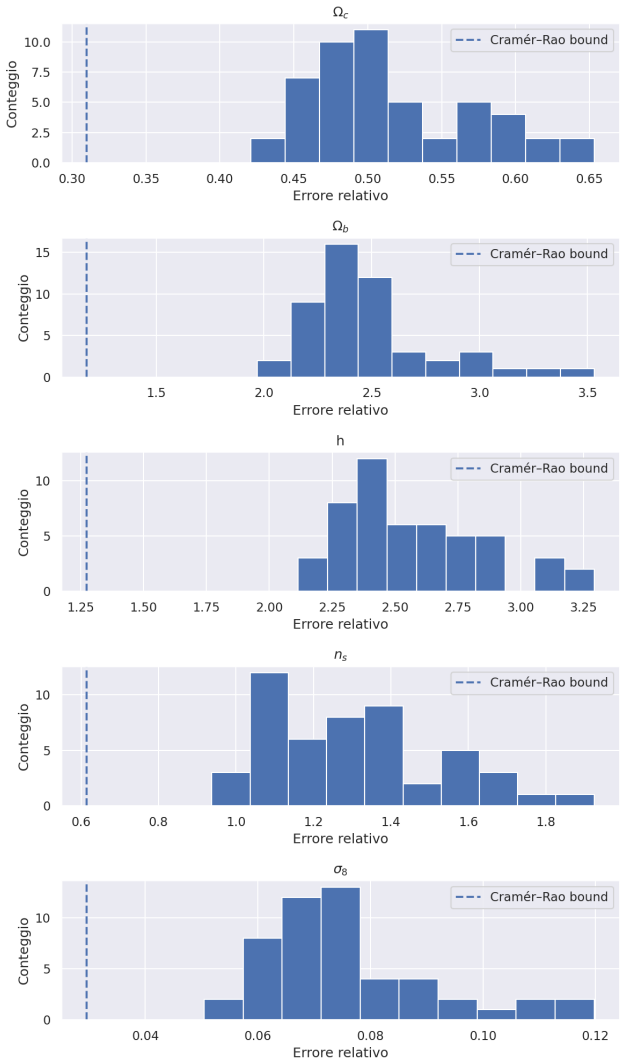


Figura 4.11: Istogrammi degli errori relativi della FIM di 50 riproduzioni del MLP con struttura piramidale allenato con le stesse 1000 simulazioni.

Capitolo 5

Conclusioni

Grazie all'avanzamento tecnologico degli strumenti di rivelazione, si è in grado di raccogliere sempre più grandi quantità di dati, le quali hanno bisogno di statistiche che li riassumano al fine di avere una comprensione della cosmologia. L'uso di reti neurali è stato ampliamente adottato per semplificare e rendere più efficiente l'analisi di larghi dataset, ma è necessario testare quali siano le architetture che ottimizzano sia la quantità di informazione estratta sia il costo computazionale. A tale scopo, si vuole affrontare un problema il cui limite teorico di estrazione dell'informazione è già noto e quindi verificare come viene saturato da diverse strutture neurali. Si è partiti dunque dall'analisi del power spectrum a scale lineari, per cercare dei modelli neurali in grado di espandere la propria analisi a scale non lineari e più complesse.

L'obiettivo posto è quello di identificare un'architettura di un MLP in grado di massimizzare l'informazione dei parametri estraibile dal power spectrum nella scala lineare ottimizzando il costo computazionale. Ciò è stato ottenuto analizzando, attraverso la matrice dell'informazione di Fisher, diverse strutture per profindità e larghezza e variando l'ampiezza dei prior dai quali vengono generate le simulazioni di addestramento.

Un primo risultato è stato ottenuto analizzando come varia l'informazione estratta all'aumentare del numero di strati nascosti, individuando con 5-6 strati nascosti la lunghezza ottimale dal punto di vista computazionale. Successivamente confrontando le strutture addestrate su prior più o meno ampi si è compreso che strutture con più neuroni generalizzano meglio il problema in quanto possiedono più parametri. La struttura piramidale offre un compromesso tra il numero di strati di nascosti oltre i quali l'informazione estratta satura e il numero di parametri da addestrare nel modello, ovvero una struttura con 6 strati nascosti e un numero di parametri pari a 2 248 581 e confrontabile con la struttura a 10 strati nascosti da 512 neuroni.

L'ultima sezione dell'analisi invece ha mostrato come la stocasticità dell'allenamento del MLP è da essere tenuta in considerazione in quanto inizializzazioni diverse danno origine a errori sui parametri molto distribuiti. Sono quindi necessarie ulteriori verifiche variando le inizializzazioni per addestramenti con più simulazioni in input, osservando come modifica la varianza sulla distribuzione degli errori sui parametri, in modo tale da verificare quando e se l'addestramento diventa stabile rispetto all'inizializzazione.

Alla luce di quanto visto, tra le strutture analizzate, quella che riesce a mantenere dei vincoli stretti intorno al valore teorico, attenuando bias locali e apprendendo le relazioni generali del problema in maniera più efficiente è la struttura piramidale, la quale, per tale motivo, è un ottima base di partenza per una struttura semplice atta allo studio di problemi più complessi e non lineari sullo spettro di potenza o altre tipologie di statistiche.

Bibliografia

[1] A. Blanchard et al., *Euclid preparation: VII. Forecast validation for Euclid cosmological probes*, A&A 642, A191 (2020), arXiv:2006.08616.

[2] G. Cybenko, *Approximation by superpositions of a sigmoidal function*, Math. Control Signals Systems **2**, 303–314 (1989), doi:10.1007/BF02551274.

[3] R. Eldan & O. Shamir, *The power of depth for feedforward neural networks*, in Proceedings of the 29th Conference on Learning Theory (COLT 2016), (2016), arXiv:1512.03965.

[4] A. Bairagi, B. Wandelt & F. Villaescusa-Navarro, *How many simulations do we need for simulation-based inference in cosmology?*, (2025), arXiv:2503.13755 [astro-ph.CO].

[5] A. Heavens, R. Jimenez & O. Lahav,*Massive lossless data compression and multiple parameter estimation from galaxy spectra*,MNRAS **317**, 965–972 (2000),arXiv:astro-ph/9911102.

[6] N. Bellomo, J. L. Bernal, G. Scelfo, A. Raccanelli & L. Verde,*Beware of commonly used approximations I: errors in forecasts*,(2020),arXiv:2005.10384.

[7] N. Aghanim et al. (Planck Collaboration), *Planck 2018 results. VI. Cosmological parameters*, A&A 641, A6 (2020), arXiv:1807.06209.

[8] M. Greiner & T. A. Enßlin, *Log-transforming the matter power spectrum*, A&A 574, A86 (2015), arXiv:1304.2897.

[9] T. Charnock, G. Lavaux & B. D. Wandelt, *Information Maximising Neural Networks*, Phys. Rev. D 97, 083004 (2018), arXiv:1802.03537.

[10] M. D. McKay, R. J. Beckman & W. J. Conover, *A Comparison of Three Methods for Selecting Values of Input Variables in the Analysis of Output from a Computer Code*, Technometrics 21(2), 239–245 (1979), JSTOR:1268522.

[11] F. Villaescusa-Navarro, S. Genel, D. Anglés-Alcázar et al., *(2022c)*, ApJS **259**, 61, doi:10.3847/1538-4365/ac5ab0.

[12] M. Rosenblatt,*The Perceptron: A probabilistic model for information storage and organization in the brain*,Psychol. Rev. **65**(6), 386–408 (1958),doi:10.1037/h0042519.

[13] F. Villaescusa-Navarro et al., *The Quijote Simulations*, Astrophys. J. Suppl. Ser. 250(1), 2 (2020), arXiv:1909.05273.

[14] Y. LeCun, L. Bottou, Y. Bengio & P. Haffner,*Gradient-Based Learning Applied to Document Recognition*,Proc. IEEE **86**(11), 2278–2324 (1998),doi:10.1109/5.726791.

[15] R. Takahashi, *Presentation at Texas Symposium on Relativistic Astrophysics (2009)*, http://cosmo.phys.hirosaki-u.ac.jp/takahasi/presen/texas_2009_4.pdf.

[16] J. Alsing & B. Wandelt, *Generalized massive optimal data compression*, *Mon. Not. R. Astron. Soc. Lett.* **476**, L60–L64 (2018), doi:10.1093/mnrasl/sly029.

[17] I. Ullah & A. Petrosino, *About Pyramid Structure in Convolutional Neural Networks*, (2016), arXiv:1608.04064 [cs.CV].

[18] A. Zhang, Z. C. Lipton, M. Li, and A. J. Smola, "Dive into Deep Learning," *arXiv preprint* arXiv:2106.11342, 2023. Available: `https://arxiv.org/abs/2106.11342`

[19] J.-E. Campagne, F. Lanusse, J. Zuntz, A. Boucaud, S. Casas, M. Karamanis, D. Kirkby, D. Lanzieri, A. Peel & Y. Li,*JAX-COSMO: An End-to-End Differentiable and GPU Accelerated Cosmology Library*,Open J. Astrophys. **6** (2023),doi:10.21105/astro.2302.05163.