

SCUOLA DI INGEGNERIA  
Sede di Forlì

Corso di Laurea in  
INGEGNERIA AEROSPAZIALE  
Classe L-9

Elaborato finale di Laurea in  
ING-IND/05:  
Meccanica Orbitale

**GENERAZIONE DI OSSERVABILI RADIOMETRICHE  
PER NAVIGAZIONE DEEP SPACE TRAMITE RETI  
NEURALI CONVOLUZIONALI**

CANDIDATO  
Andrea Ricci

RELATORE  
Paolo Tortora

CORRELATORE  
Andrea Togni

Anno Accademico 2021/2022

# Sommario

Grazie all'evoluzione degli strumenti di calcolo e delle strutture digitali, le intelligenze artificiali si sono evolute considerevolmente negli ultimi anni, permettendone sempre nuove e complesse applicazioni.

L'interesse del presente progetto di tesi è quello di creare un modello di studio preliminare di intelligenza artificiale definita come Rete Neurale Convolutionale, o *Convolutional Neural Network* (CNN), al fine di applicazioni nel campo della radioscienza e dell'esplorazione planetaria; in particolare, uno degli interessi principali di applicazione del modello è negli studi di geodesia compiuti tramite determinazione orbitale di satelliti artificiali nel loro moto attorno ai corpi celesti. Le accelerazioni causate dai campi gravitazionali planetari perturbano le orbite dei satelliti artificiali, queste variazioni vengono captate dai ricevitori radio a terra sottoforma di shift Doppler della frequenza del segnale, a partire dalla quale è quindi possibile determinare informazioni dettagliate sul campo di gravità e sulla struttura interna del corpo celeste in esame.

Per poter fare ciò occorre riuscire a determinare l'esatta frequenza del segnale in arrivo, il quale, per via di perdite e disturbi durante il suo tragitto, presenterà sempre una componente di rumore. Il metodo più comune per scindere la componente di informazione da quella di rumore e ricavarne la frequenza effettiva è l'applicazione di trasformate di Fourier a tempo breve, o *Short-time Fourier Transform* (STFT).

Con l'attività sperimentale proposta ci si è quindi posto l'obiettivo di istruire un CNN alla stima della frequenza di segnali reali sinusoidali rumorosi per avere un modello computazionalmente rapido e affidabile a supporto delle operazioni di pre-processing per missioni di radio-scienza.

# Indice

<b>Capitolo 1: Determinazione d'orbita e Radioscienza .....</b>	<b>1</b>
1.1 Cenni storici.....	1
1.2 Problema di determinazione dell'orbita .....	3
1.2.1 Introduzione al problema .....	3
1.2.2 Linearizzazione del problema .....	4
1.2.3 Soluzione ai minimi quadrati .....	5
1.3 Determinazione d'orbita tramite osservabili radiometriche .....	7
1.3.1 Principio di funzionamento di osservabili Range e Doppler.....	7
1.3.2 Applicazioni scientifiche ed esempi di missioni .....	9
<b>Capitolo 2: Reti Neurali .....</b>	<b>13</b>
2.1 Struttura e funzionamento delle reti neurali .....	13
2.2 Regressione tramite rete neurale.....	15
<b>Capitolo 3: Attività sperimentali .....</b>	<b>19</b>
3.1 Ipotesi e modelli .....	19
3.2 Risultati e discussione .....	23
<b>Conclusione .....</b>	<b>26</b>

# Capitolo 1

## Determinazione d'orbita e Radioscienza

### 1.1 Cenni storici

Lo studio della determinazione orbitale inizia attorno al 1610 con Johannes Kepler (1571-1630) con la pubblicazione del *Astronomia nova* nel quale enuncia le prime due delle tre leggi che portano il suo nome:

- I pianeti seguono orbite ellittiche di cui il sole occupa uno dei due fuochi
- Il raggio congiungente i centri del sole e del pianeta orbitante descrive aree uguali in tempi uguali
- Il rapporto tra il cubo del semiasse maggiore dell'orbita ed il quadrato del periodo orbitale è lo stesso per tutti i pianeti

In seguito fu Isaac Newton (1642-1727) a fornire le basi matematiche per lo studio orbitale grazie alle leggi del moto e di attrazione gravitazionale per masse puntiformi:

$$\mathbf{F} = m\mathbf{a} = m\ddot{\mathbf{R}} \quad (1.1)$$

Dove  $\mathbf{F}$  è la forza agente sul corpo,  $m$  è la sua massa inerziale ed  $\mathbf{a} = \ddot{\mathbf{R}}$  è l'accelerazione a cui esso è soggetto, ovvero la derivata della velocità  $\mathbf{v} = \dot{\mathbf{R}}$  che a sua volta è la derivata della posizione  $\mathbf{R}$  rispetto ad un sistema di riferimento inerziale.

$$\mathbf{F}_{g,12} = -G \frac{m_1 m_2}{r^3} \mathbf{r} \quad (1.2)$$

Dove  $\mathbf{F}_{g,12}$  è la forza di attrazione gravitazionale tra i due corpi di massa  $m_1$  ed  $m_2$ ,  $r$  è la distanza tra i due corpi relativa ai loro centri geometrici e  $G \approx 6.67 \times 10^{-11} \text{ m}^3 \text{ kg}^{-1} \text{ s}^{-2}$  è la Costante di Gravitazione Universale. Queste leggi sono le fondamenta della descrizione matematica della dinamica orbitale.

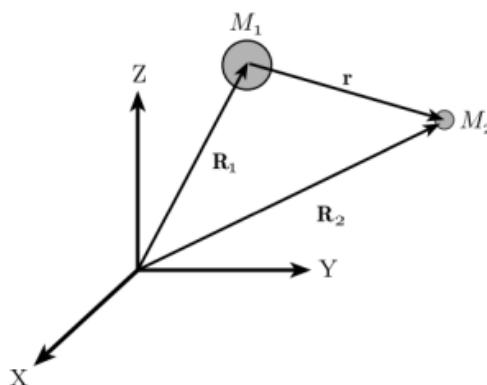


Figura 1.1: Sistema di riferimento inerziale X, Y, Z su cui si basa il problema dei due corpi [1].

Il primo caso di studio dell'orbita con una corretta predizione della posizione futura del corpo osservato viene attribuita a Carl Gauss (1777-1855) il quale, basandosi sulle osservazioni di Giuseppe Piazzi che lo portarono nell'1801 alla scoperta di Cerere, fu in grado di sviluppare una base analitica e computazionale per lo studio della posizione futura dell'asteroide. Partendo dal calcolo per trovare i sei elementi dell'orbita di un pianeta tramite due set di coordinate  $x, y, z$  eliocentriche, il quale richiedeva la risoluzione di un sistema di  $3 \times 2$  equazioni non-lineari in sei incognite, Gauss riuscì a ridurre il calcolo di un set di coordinate eliocentriche utilizzando due set di valori geocentrici osservabili  $\lambda_i, \beta_i$  tramite la soluzione di complesse equazioni trascendentali. Per una trattazione completa, troppo prolissa per essere descritta in questa tesi, si può fare riferimento al testo [2].

Infine fu la pubblicazione di A.-M. Legendre (1752-1833) *Nouvelles méthodes pour la détermination des orbites des comètes* del 1805 a presentare il metodo dei minimi quadrati in un'appendice del suo libro che, per la chiarezza dell'elaborato e gli esempi numerici proposti, lo resero rapidamente noto a tutta la comunità scientifica [3]. Questo metodo rimane ad oggi uno dei più utilizzati.

Il primo episodio documentato di tracciamento di un satellite artificiale avvenne quando il radar statunitense di Milestone Hill riuscì a rilevare il segnale proveniente dallo Sputnik II e, tramite metodo Doppler, trovarne e predirne l'orbita. Tali metodi furono poi sviluppati e migliorati durante il corso della guerra fredda, con lo scopo di rilevare potenziali missili balistici; gli avanzamenti compiuti in tale periodo resero i sistemi ed i sensori abbastanza avanzati da permetterne l'uso anche in campo spaziale, nonostante fossero stati progettati per scopi balistici.

Nello stesso periodo iniziarono i primi esperimenti di radiofisica mirati allo studio dei corpi del sistema solare; a meta anni '40, ad esempio, furono trasmessi segnali radio verso la Luna e gli echi di ritorno ricevuti mostrarono variazioni delle proprietà di diffusione della superficie dando origine alla tecnica del radar planetario, utilizzata negli anni seguenti anche per lo studio di Marte, Venere e della ionosfera terrestre. Tali esperimenti portarono alla nascita della radioscienza, in origine riferita all'utilizzo di onde radio per lo studio dell'ambiente terrestre; il termine fu poi adottato dalla comunità di esplorazione planetaria per descrivere le ricerche basate sull'analisi di segnali inviati dai satelliti artificiali ed analizzati nelle stazioni di terra [4].

## 1.2 Problema di determinazione dell'orbita

### 1.2.1 Introduzione al problema

Gli studi di determinazione dell'orbita per corpi celesti si basano sull'assunzione che i corpi si muovano solo sotto l'influenza di forze di attrazione centrali, mentre per satelliti artificiali, per via della loro massa ridotta, delle loro dimensioni e delle caratteristiche delle orbite da loro descritti, anche le forze di natura non-gravitazionale risultano di importanza significativa sul loro moto [1].

Lo studio orbitale ha come fine la determinazione dello Stato del corpo in analisi, ovvero un vettore  $n$ -dimensionale  $\bar{X}$  contenente posizione, velocità e parametri costanti che appaiono nel modello di forze. Per ricavare lo stato si parte dalla definizione dei vettori  $6 \times 1 \bar{\xi}$ , contenente le informazioni di posizione e velocità, e  $\bar{f}(\xi, t) = \dot{\bar{\xi}}$  che descrive velocità ed accelerazioni:

$$\bar{\xi} = \begin{bmatrix} x \\ y \\ z \\ u \\ v \\ w \end{bmatrix} \quad \bar{f}(\xi, t) = \begin{bmatrix} u \\ v \\ w \\ -\frac{\mu x}{r^3} + P_x \\ -\frac{\mu y}{r^3} + P_y \\ -\frac{\mu z}{r^3} + P_z \end{bmatrix} \quad (1.3)$$

Il termine  $-\frac{\mu}{r^3}[x, y, z]$  rappresenta la forza gravitazionale, mentre i termini  $P_x, P_y, P_z$  le forze non-centrali.

È possibile quindi ottenere il vettore di stato  $\bar{X}$  partendo da  $\bar{\xi}$  ed aggiungendogli un vettore  $q$ -dimensionale  $\alpha$ , contenente i parametri costanti ( $\dot{\alpha} = 0$ ) che descrivono il modello di forze:

$$\bar{X} = \begin{bmatrix} \bar{\xi} \\ \alpha \end{bmatrix} \quad \bar{F}(X, t) = \begin{bmatrix} f(\xi, \alpha, t) \\ \vdots \\ 0 \end{bmatrix} \quad (1.4)$$

In cui  $\bar{X}$  sarà quindi un vettore  $n$ -dimensionale con  $n = 6 + q$ , mentre  $\bar{F}(X, t) = \dot{\bar{X}}$ , combinato con una condizione iniziale nota  $\bar{X}_0 = \bar{X}(t_0)$ , crea un sistema di  $n$ -equazioni non-lineari differenziali del primo ordine:

$$\bar{F}(X, t) = \dot{\bar{X}} \quad \bar{X}_0 = \bar{X}(t_0) \quad (1.5)$$

Lo stato iniziale dell'orbita però non è mai noto con esattezza, inoltre alcune costanti o equazioni che descrivono il moto del satellite sono note solo in maniera approssimata; occorre dunque fare riferimento a dati ottenuti dalle osservazioni dirette dello spacecraft, chiamate *osservabili* e finzioni dello stato del satellite.

In un dato istante di tempo  $t_i$  (con  $i = 1, \dots, l$ ) le osservabili reali saranno un vettore di  $p$  misurazioni  $\bar{Y}_i$  per le quali vale la seguente relazione:

$$\bar{Y}_i = \bar{G}(\bar{X}_i, t_i) + \bar{\varepsilon}_i; \quad i = 1, \dots, l \quad (1.6)$$

Il termine  $\bar{G}(X_i, t_i)$  rappresenta le osservabili attese a seguito dell'applicazione della funzione non-lineare  $G$  allo stato  $\bar{X}_i$ , ed il termine  $\bar{\varepsilon}_i$  è il vettore degli errori delle misure. In generale  $p < n$ , occorre quindi raccogliere osservazioni per un numero di istanti successivi tali per cui i vettori  $\bar{Y}$ ,  $\bar{G}$  ed  $\bar{\varepsilon}$  abbiano dimensioni  $m = l \times p \gg n$  al fine di poter determinare il vettore stato ignoto  $\bar{X}_k \equiv \bar{X}(t_k)$  in un generico istante di tempo.

È possibile inoltre definire una matrice di cambio di stato  $\bar{\Theta}$  partendo dalla risoluzione dell'equazione (1.5) con la quale si può riscrivere lo stato in un istante generico  $t_i$  come:

$$\bar{X}_i = \bar{\Theta}(\bar{X}_k, t_k, t_i) \quad (1.7)$$

Sostituendo tale formulazione nella relazione dell'osservabile si ottiene che:

$$\bar{Y}_i = \bar{G}(\bar{\Theta}(\bar{X}_k, t_k, t_i)) + \bar{\varepsilon}_i = \tilde{G}(\bar{X}_k, t_k, t_i) + \bar{\varepsilon}_i \quad (1.8)$$

Formando così un sistema di  $m$  equazioni in  $n$  incognite dello stato ed  $m$  incognite di errori di misurazione. Nel caso ideale in cui  $\bar{\varepsilon} = 0$  basterebbe quindi prendere un set di  $n$  equazioni indipendenti per riuscire a ricavare lo stato iniziale e gli stati ad istanti successivi; mentre se  $\bar{\varepsilon} \neq 0$  occorre compiere una stima della soluzione migliore tra tutte le possibili risoluzioni.

## 1.2.2 Linearizzazione del problema

I sistemi così ottenuti, per via della loro natura non lineare, andrebbero risolti tramite metodi numerici iterativi, computazionalmente lenti e pesanti, come ad esempio Newton-Raphson. Se però è nota una traiettoria nominale di riferimento  $X^*$  e questa si mantiene sufficientemente simile alla traiettoria reale  $X$  durante tutto l'intervallo di tempo in analisi, allora si può ricavare la traiettoria del moto effettivo tramite espansione in serie di Taylor della traiettoria di riferimento ad ogni istante.

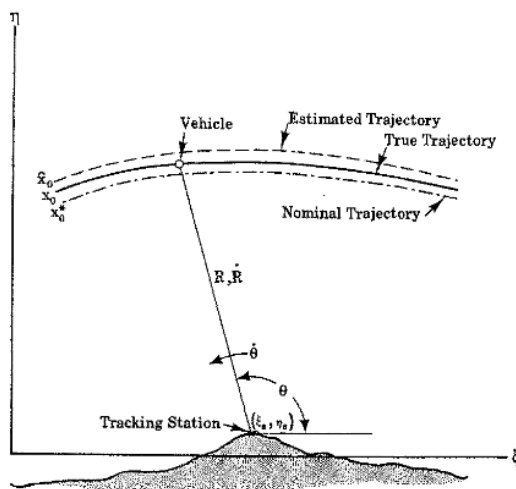


Figura 1.2: rappresentazione grafica della differenza tra traiettoria calcolata  $\hat{x}$ , traiettoria reale  $x$  e traiettoria nominale  $x^*$  [5].

Tale procedimento può quindi essere usato per trovare relazioni lineari tra la deviazione dello stato da quello nominale e la deviazione delle osservate (1.9); ricavando un sistema di equazioni lineari con coefficienti tempo-dipendenti.

$$\bar{x} = \bar{X}(t) - \bar{X}^*(t) \quad \bar{y} = \bar{Y}(t) - \bar{Y}^*(t) \quad (1.9)$$

Dove  $\bar{Y}_l^* = \bar{Y}^*(t_i) = \bar{G}(\bar{X}_l^*, t_i)$  e' il valore dell'osservata calcolata usando la traiettoria nominale  $\bar{X}_l^*$ . Espandendo in serie le equazioni (1.5) ed (1.6) arrestate al primo ordine, si ottengono due equazioni:

$$\bar{X}(t) = \bar{F}(\bar{X}, t) = \bar{F}(\bar{X}^*, t) + \left[ \frac{\partial \bar{F}}{\partial \bar{X}} \right]^* \bar{x} + (\dots) \quad (1.10)$$

$$Y_l = \bar{G}(\bar{X}_l, t_i) + \bar{\varepsilon}_l = \bar{G}(\bar{X}_l^*, t_i) + \left[ \frac{\partial \bar{G}}{\partial \bar{X}} \right]^* \bar{x}_l + (\dots)$$

Poiché  $\dot{\bar{X}}^* = \bar{F}(\bar{X}^*, t)$ , è possibile combinare le equazioni (1.9) ed (1.10) per ottenere:

$$\begin{aligned} \dot{\bar{x}} &= \dot{\bar{X}} - \dot{\bar{X}}^* = \frac{\partial \bar{F}(\bar{X}^*, t)}{\partial \bar{X}} \bar{x} = \bar{A}(t) \bar{x} \\ \bar{y}_l &= \frac{\partial \bar{G}(\bar{X}_l^*, t_i)}{\partial \bar{X}} \bar{x}_l + \bar{\varepsilon}_l = \bar{H}_l \bar{x}_l + \bar{\varepsilon}_l \end{aligned} \quad (1.11)$$

È possibile, inoltre, definire una nuova matrice del cambio di stato  $\bar{\phi}(t, t_k)$  tale da soddisfare l'equazione differenziale  $\dot{\bar{\phi}}(t, t_k) = \bar{A}(t) \bar{\phi}(t, t_k)$  e la si può utilizzare per riscrivere l'equazione (1.11) come segue:

$$\bar{x}(t) = \bar{\phi}(t, t_k) \bar{x}_k \quad (1.12)$$

Tale formulazione può essere usata per riscrivere il secondo sistema che diventa:

$$\bar{y}_l = \bar{H}_l \bar{\phi}(t_i, t_0) \bar{x}_0 + \bar{\varepsilon}_l \quad (1.13)$$

Togliendo la notazione indiciale, si ottiene quindi un sistema lineare  $\bar{y} = \bar{H} \bar{x} + \bar{\varepsilon}$  di  $m$  equazioni in  $n$  incognite dove  $\bar{y} = [m \times 1]$ ,  $\bar{H} = [m \times n]$ ,  $\bar{x} = [n \times 1]$  ed  $\bar{\varepsilon} = [m \times 1]$  in cui  $m$  è il numero delle osservazioni totali; basta quindi avere  $m > n$  per poterlo risolvere.

### 1.2.3 Soluzione ai minimi quadrati

Come accennato al termine del capitolo 1.2, qualora l'errore di misurazione  $\varepsilon_i$  sia diverso da 0, occorre stimare la soluzione migliore tra le possibili ovvero quella che minimizza il residuo tra le osservate reali e quelle calcolate. Il criterio più utilizzato per compiere tale stima è il metodo dei minimi quadrati, il quale consiste nel minimizzare la somma dei quadrati dell'errore di osservazione



$$J(\overline{X}_k^*) = \sum_{i=1}^l \varepsilon_i^{*T} \times \varepsilon_i^* = \sum_{i=1}^l [\overline{Y}_i - \tilde{G}(X_k^*, t_k, t_i)]^T [\overline{Y}_i - \tilde{G}(X_k^*, t_k, t_i)] \quad (1.14)$$

In cui  $\overline{X}_k^*$  è lo stato nominale all'istante  $k$ -esimo ed  $\varepsilon_i$  è la differenza tra l'osservata reale  $\overline{Y}_i$  e la predizione calcolata usando lo stato nominale  $\tilde{G}(X_k^*, t_k, t_i)$ . Occorre quindi trovare un  $\widehat{X}_k$  che minimizza  $J(\overline{X}_k^*)$ , in cui  $\varepsilon_i^2$  può essere pensato come una parabola e quindi presenta sempre uno e un solo minimo quando la sua concavità è rivolta verso l'alto. Tali condizioni possono essere soddisfatte imponendo rispettivamente che la derivata prima di  $J(\overline{X}_k^*)$  calcolata in  $\widehat{X}_k$  sia nulla e che la sua derivata seconda sia positiva:

$$\left. \frac{\partial J}{\partial X_k^*} \right|_{\widehat{X}_k} = 0, \quad \delta \overline{X}_k^T \left[ \frac{\partial^2 J}{\partial X_k^* \partial X_k^*} \right]_{\widehat{X}_k} \delta \overline{X}_k \geq 0 \quad (1.15)$$

Sostituendo i risultati nell'equazione precedente (1.14) si ottiene che:

$$\left. \frac{\partial J}{\partial X_k^*} \right|_{\widehat{X}_k} = - \sum_{i=1}^l [\overline{Y}_i - \tilde{G}(\widehat{X}_k, t_k, t_i)]^T \frac{\partial \tilde{G}(\widehat{X}_k, t_k, t_i)}{\partial \widehat{X}_k} = 0 \quad (1.16)$$

Il quale forma un sistema di  $n$  equazioni non-lineari algebriche con il vettore di  $n$ -incognite  $\widehat{X}_k$  da risolvere iterativamente tramite metodi numerici.

Risulta possibile alleggerire il carico computazionale applicando il metodo dei minimi quadrati al problema linearizzato, trovando la stima di  $x$  che minimizza la somma dei quadrati dei residui. Avendo a disposizione la deviazione delle osservate espanse in serie  $y$  e un valore specificato della deviazione dello stato  $x$ , che chiameremo  $x^\circ$ , si può riscrivere il termine di errore di osservazione  $\varepsilon^\circ$  all'interno di  $J(x^\circ)$  usando l'equazione (1.13) ottenendo così

$$J(x^\circ) = \varepsilon^{\circ T} \varepsilon^\circ = (\overline{y} - \overline{H}x^\circ)^T (\overline{y} - \overline{H}x^\circ) \quad (1.17)$$

Essendo questa una funzione quadratica di  $x$  presenterà quindi un minimo in  $\hat{x}$  tale per cui

$$\left. \frac{\partial J}{\partial x^\circ} \right|_{x^\circ = \hat{x}} = -2\overline{H}^T (\overline{y} - \overline{H}\hat{x}) = 0 \quad (1.18)$$

Da cui si ottiene che

$$\overline{H}^T (\overline{y} - \overline{H}\hat{x}) = 0 \quad \text{o anche} \quad (\overline{H}^T \overline{H})\hat{x} = \overline{H}^T \overline{y} \quad (1.19)$$

Se  $m \geq n$  allora la matrice  $\overline{H}^T \overline{H}$  è una matrice definita positiva di dimensioni  $n \times n$  e rango  $n$ , la quale permette di calcolare con facilità il valore di  $\hat{x}$  come

$$\hat{x} = (\overline{H}^T \overline{H})^{-1} \overline{H}^T \overline{y} \quad (1.20)$$

A partire dalla soluzione ai minimi quadrati ne evolvono diverse variazioni che puntano ad ottimizzarne le performance introducendo considerazioni di carattere qualitativo e statistico sulla natura degli errori.

Ad esempio la Soluzione ai Minimi Quadrati Pesata tiene in considerazione l'accuratezza delle misure, mentre la Stima della Varianza Minima include informazioni di carattere statistico sulla misurazione degli errori. Per la spiegazione dettagliata di tali metodi risolutivi, così come di altri non elencati, è possibile fare riferimento ai testi [1] e [5].

### 1.3 Determinazione d'orbita tramite osservabili radiometriche

#### 1.3.1 Principio di funzionamento di osservabili Range e Doppler

La determinazione dell'orbita è un processo iterativo che si basa sull'aggiustamento dei parametri dei modelli matematici che descrivono lo stato del satellite in esame e l'ambiente in cui si muove, usando la soluzione ai minimi quadrati per minimizzare lo scarto tra la traiettoria stimata dai modelli, detta orbita nominale, ed il suo valore effettivo ottenuto dal sistema di tracciamento. Tale processo deve essere iterativo perché, come visto nel capitolo 1.2.2, i risultati sono ottenuti linearizzando un problema non-lineare e quindi occorre ripetere le operazioni utilizzando i parametri calcolati al passaggio precedente fino ad arrivare a convergenza dei risultati.

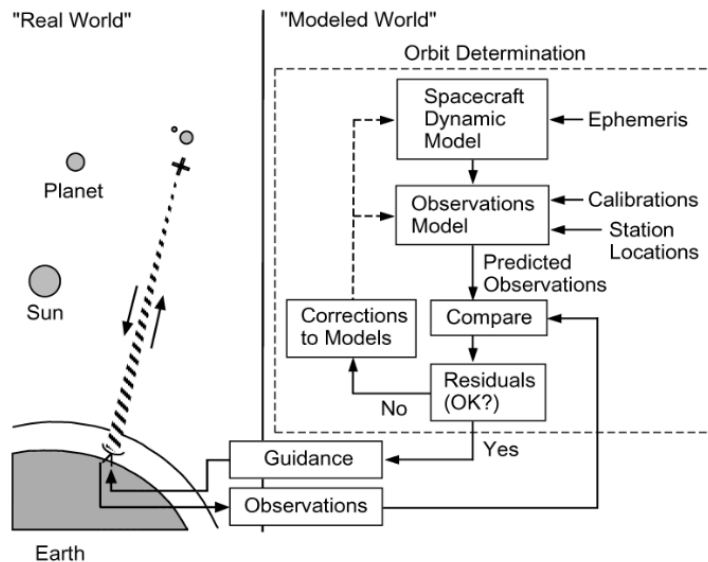


Figura 1.3: processo iterativo di determinazione dell'orbita e navigazione [6].

Una volta che i parametri sono stati ottimizzati e si è ragionevolmente sicuri che il sistema riesca a predire in modo accurato la traiettoria del satellite, si passa alle operazioni di navigazione, o *guidance*, che consistono nel calcolare eventuali correzioni di traiettoria per raggiungere l'obiettivo di missione in maniera ottimale.

Le osservabili utilizzate per la navigazione sono delle serie di toni sinusoidali, generate alla frequenza standard di una stazione appartenente solitamente al Deep Space Network, e modulate in portanti ad alta frequenza. Una volta che tale segnale raggiunge il satellite, questo lo aggancia tramite sistemi phase-locked-loop, i quali producono un segnale di riferimento coerente con la portante del segnale ricevuto.

Tale segnale viene inviato nuovamente verso la stazione di terra dove viene analizzato; il tempo di andata e ritorno del segnale permette di calcolare la distanza del satellite dalla stazione, detta *slant range* ( $\rho$ ); mentre dalla differenza tra la frequenza del segnale trasmesso e ricevuto, quindi il *Doppler*, si calcola la sua velocità lungo la congiungente stazione-spacecraft, anche detta *slant range rate* ( $\dot{\rho}$ ).

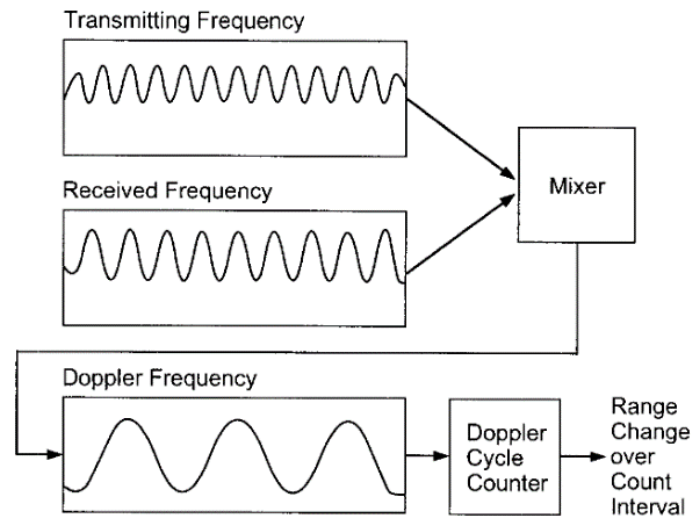


Figura 1.4: processo di estrazione della frequenza Doppler [6].

I metodi più accurati di determinazione sono ottenuti tramite configurazioni di tracciamento two-way, per le quali la stazione inviante e ricevente, e quindi la frequenza standard, coincidono.

Chiamando con  $f_T$  la frequenza trasmessa da una data stazione del DSN e con  $f_R$  la frequenza da essa ricevuta, per una trasmissione con configurazione two-way vale che:

$$f_R = \alpha \left( 1 - 2 \frac{\dot{\rho}}{c} \right) f_T \quad (1.21)$$

In cui il termine  $(2 \dot{\rho}/c)f_T$  è la frequenza Doppler,  $\alpha$  il *turn-around ratio* e  $c$  è la velocità della luce. Il termine 2 sta ad indicare il fatto che il segnale inviato dal satellite è una replica del segnale ricevuto, il quale ha subito un primo shift in frequenza durante il percorso di uplink e ne subisce un secondo durante il downlink, ovvero alla ricezione da parte della stazione di terra. Dallo shift Doppler si possono ricavare diverse informazioni su posizione e velocità del satellite a seguito dell'integrazione della traiettoria.

È inoltre possibile scrivere lo slant range rate in un generico istante di tempo come una somma tra il geocentric range rate ( $\dot{r}$ ) ed un termine che tiene conto della distanza della stazione DSN dall'asse di rotazione terrestre ( $r_s$ ), della velocità di rotazione media della terra ( $\omega_e$ ), dalla longitudine della stazione

a terra ( $\lambda_s$ ), della declinazione ( $\delta$ ) e dall'ascensione retta ( $\alpha$ ) dell'orbita del satellite. In forma approssimata, come descritto in [6]:

$$\dot{\rho}(t) = \dot{r}(t) + \omega_e r_s \cos(\delta) \sin(\omega_e t + \phi + \lambda_s - \alpha) \quad (1.22)$$

Il termine  $\phi$  rappresenta in genere l'angolo di fase mentre quando il tempo  $t$  viene espresso rispetto al tempo civile a Greenwich, ovvero come tempo universale,  $\phi$  rappresenta invece l'ascensione retta istantanea del sole medio.

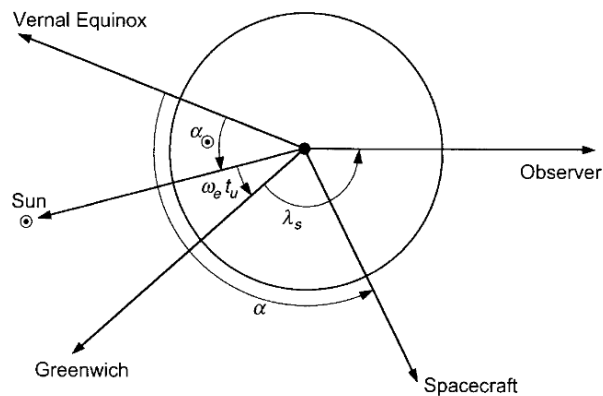


Figura 1.5: proiezione equatoriale delle coordinate [6].

È quindi possibile, ad ogni istanza, determinare non solo la velocità radiale del satellite ma anche la sua ascensione retta e la sua declinazione; inoltre correlando dati provenienti da diverse osservabili Doppler, raccolte su un periodo di diversi giorni, risulta possibile calcolare anche la velocità normale alla linea di vista.

### 1.3.2 Applicazioni scientifiche ed esempi di missioni

Come accennato nell'introduzione storica al passo 1.1, la radioscienza si occupa dello studio dei corpi celesti tramite l'utilizzo di osservabili radiometriche. Si divide in due macrogruppi che studiano aspetti differenti dei corpi analizzati.

La prima categoria si basa sull'analisi dei fenomeni che agiscono sulla propagazione dei segnali, studiando come essi variano a seguito dell'interazione con un corpo celeste. Dalle variazioni indotte sulle proprietà elettromagnetiche del segnale durante il transito è possibile ottenere dati sulla composizione delle atmosfere e ionosfere planetarie, studiare la corona solare e le code delle comete in transito all'interno del sistema solare. Fa parte di questa categoria anche l'applicazione di radar bistatici per studiare la riflettività e le asperità della superficie di un corpo celeste, analizzando la componente di un segnale da essa riflessa.

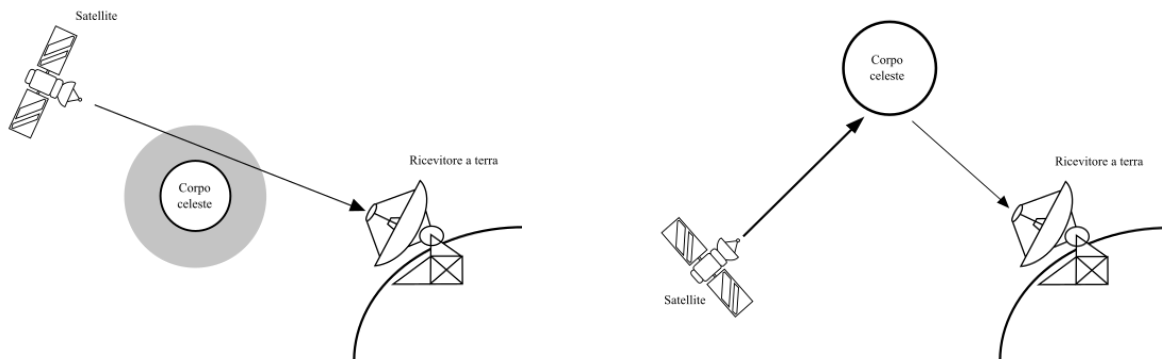


Figura 1.6: rappresentazioni grafiche dello studio di atmosfera planetaria a sinistra, e di radar bistatico a destra

La seconda categoria invece prende il nome di *Geodesia* e punta allo studio della struttura interna dei pianeti. Tali ricerche sono compiute analizzando la perturbazione del moto di un satellite artificiale messo in orbita attorno al corpo in analisi, la quale viene derivata a partire dalla frequenza Doppler del segnale di ritorno. Confrontando l'orbita nominale prevista con l'orbita effettiva, ottenuta dalle analisi del Doppler, è possibile calcolare con precisione le perturbazioni sull'orbita indotte dalla massa apparente e dalla distribuzione di densità del corpo celeste, permettendo così di studiarne la struttura interna ed il campo gravitazionale, assieme a compiere ricerche di natura relativistica [7].

Poiché lo scopo di questo documento è quello di utilizzare una rete neurale per la determinazione della frequenza di un segnale rumoroso, al fine di estrarre ed analizzare lo shift Doppler, ci si concentrerà sulla seconda categoria di applicazioni della radioscienza.

La natura di tali studi interseca tra loro la meccanica e l'elettromagnetismo: le perturbazioni gravitazionali indotte dai corpi celesti causano accelerazioni e variazioni di velocità del centro di massa del satellite che si ripercuotono sulla comunicazione radio a terra sotto forma di variazioni di frequenza indotte dall'effetto Doppler. Per massimizzare l'accuratezza delle misure degli effetti del campo gravitazionale, è importante avere modelli precisi per la descrizione di tutte le forze di natura non-gravitazionale agenti sul satellite, e tenere in considerazione le principali fonti di rumore che possono disturbare o intaccare il segnale durante il transito. Poiché le misure del campo gravitazionale da sole non riescono a fornire modelli accurati sulla composizione o sulla distribuzione di densità interna del corpo in analisi, vengono integrate, quando possibile, con dati di natura chimica o geologica. Oltre a dati informazioni prettamente gravitazionali, tramite osservabili Doppler è possibile anche rinvenire informazioni sulla rotazione dei pianeti, così come sulla nutazione e precessione dei poli [7].

Le informazioni raccolte dallo studio gravitazionale vengono infine raccolte in modelli matematici tramite una serie di armoniche sferiche bidimensionali.

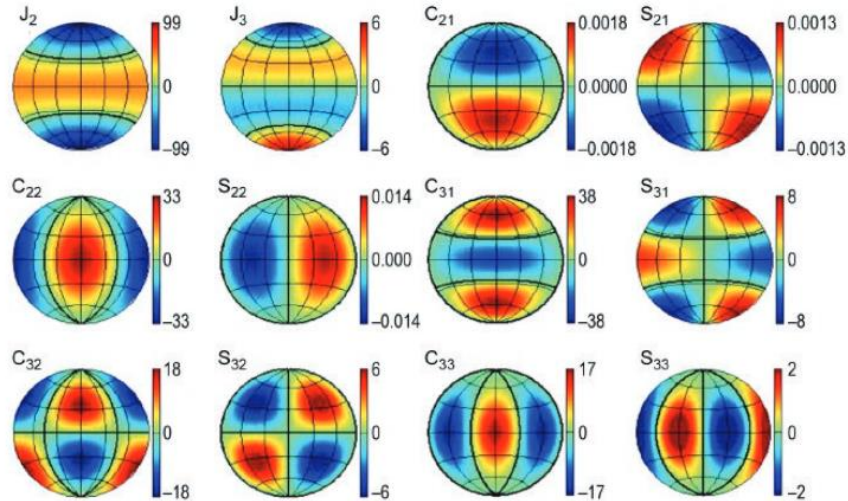


Figura 1.7: Serie di armoniche sferiche che descrivono il campo gravitazionale lunare [7].

Esempi di tale pratica li si possono vedere in azione sia in missioni passate che attuali, come ad esempio è il caso per la determinazione della massa e della gravità di Giove. Combinando i dati dei fly-by delle missioni Pioneer 10 e 11, assieme a quelli delle missioni Voyager 1 e 2, i ricercatori sono riusciti a calcolare la massa del pianeta, senza però ottenere sostanziali informazioni sulle armoniche sferiche che ne descrivono il campo gravitazionale; i primi dati su quel fronte sono stati ottenuti durante la missione Galileo che, oltre a studiare le caratteristiche rotazionali e mareali dei quattro grandi satelliti di Giove (Io, Europa, Ganimede e Callisto), permise anche di calcolare alcuni bassi coefficienti armonici pari, senza però riuscire a compiere uno studio dettagliato del campo gravitazionale planetario, a causa dell'orbita alta imposta dalle forti cinture di radiazione gioviane. È stata infine la missione Juno a mappare il campo magnetico e quello gravitazionale del pianeta grazie ad una strumentazione radio avanzata basata su configurazioni two-way e three-way estremamente stabili, con le quali si sono ottenute misurazioni sul range-rate del satellite, ottenuto partendo dalla frequenza Doppler, con una precisione dei  $3 \mu\text{m}/\text{s}$  su un tempo di integrazione di 1000 s. Grazie alla sua orbita polare fortemente eccentrica ( $e = 0.9466$ ), con un periodo di 53.5 giorni ed il pericentro ad un'altitudine di circa 3500-5000 km, risulta altamente sensibile al campo gravitazionale zonale; viene inoltre esposto a significativi effetti relativistici, come ad esempio l'accelerazione data dalla precessione di Lense-Thirring che contribuisce alla comprensione del momento angolare planetario [7].

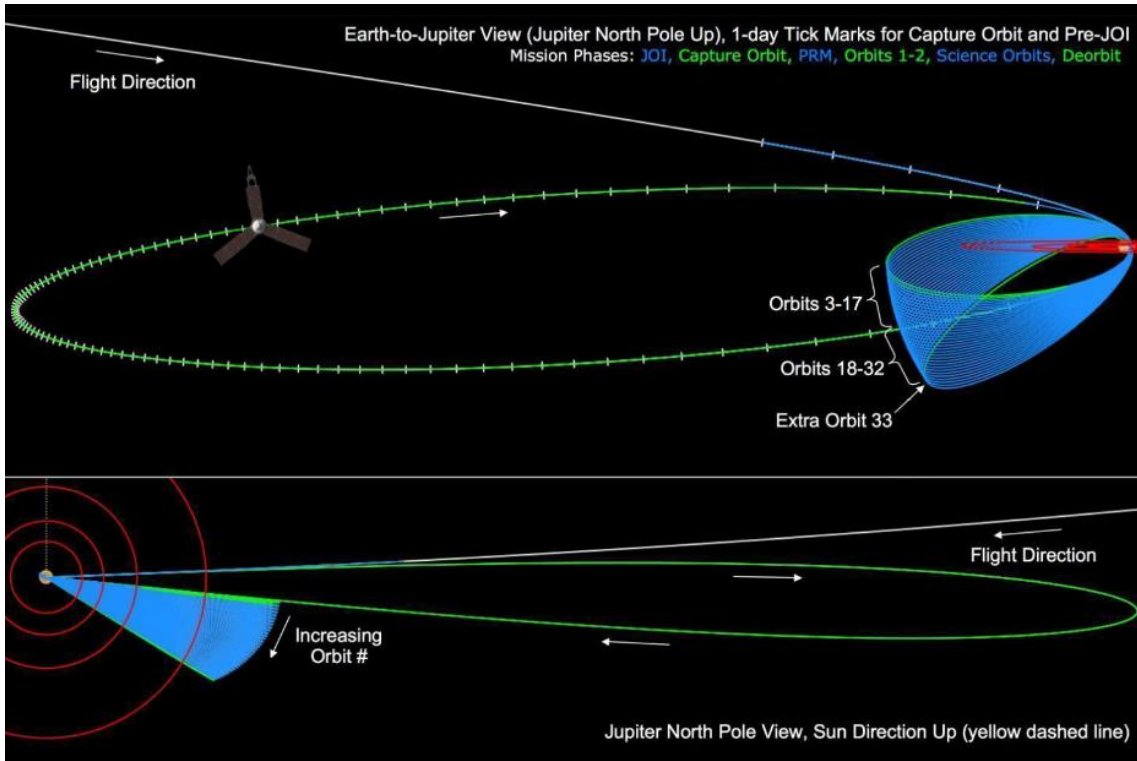


Figura 1.8: Orbita di Juno attorno a Giove [8].

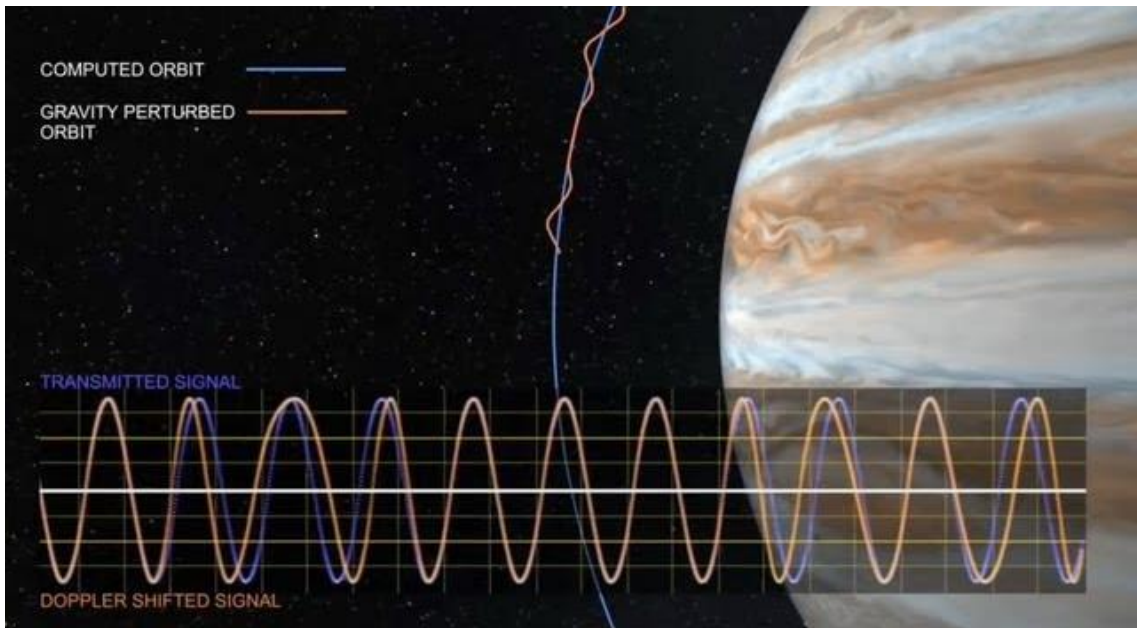


Figura 1.9: Differenza tra orbita ipotizzata e orbita reale, ottenuta dal segnale shiftato dal Doppler [8].

## Capitolo 2

### Reti Neurali

#### 2.1 Struttura e funzionamento delle reti neurali

Grazie all'evoluzione dell'elettronica e dei sistemi di elaborazione dati è diventata sempre più comune l'applicazione di sistemi di deep learning, ovvero algoritmi in grado di apprendere tramite calcoli che gli permettono di ottimizzare se stessi per lo scopo a loro designato. Alla base del concetto di deep learning ci sono le Reti Neurali, composte da strati di nodi (o neuroni) interlacciati tra loro a creare una struttura simile a quella dei neuroni umani. Il primo strato della Rete prende il nome di input e contiene i neuroni che vengono attivati con i dati del problema da analizzare; seguono poi una serie di *hidden layers*, ognuno con un numero variabile di neuroni, ed infine vi è un layer di output che indica la risposta della rete al problema.

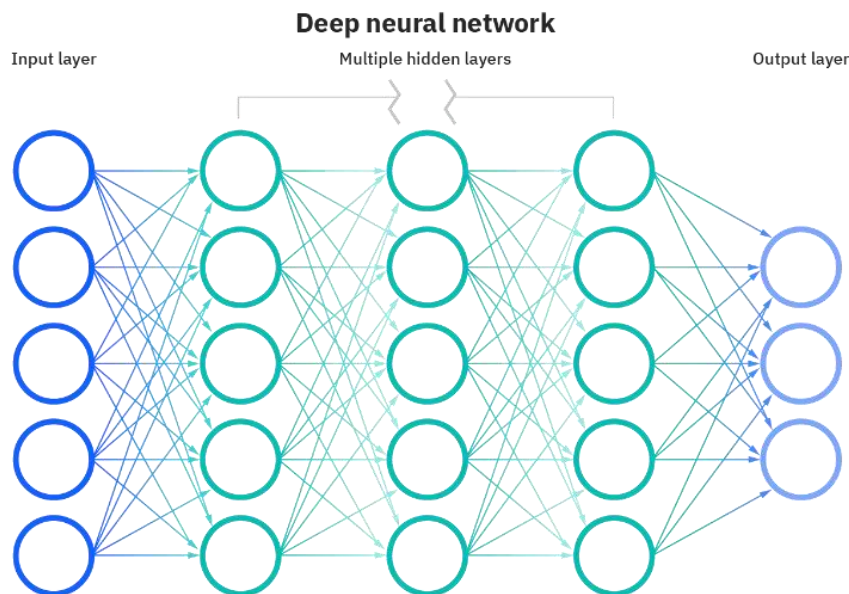


Figura 2.1 struttura di una rete neurale o neural network [9].

I neuroni sono collegati tra loro in modo che ogni nodo sia legato a tutti quelli del layer precedente tramite rami, ai quali vengono associati dei pesi; ad ogni nodo al di fuori del layer di input viene associato un bias che svolge il compito di soglia per determinare quanto il neurone debba accendersi. Quando un neurone si attiva compie una combinazione lineare dei valori dei neuroni precedenti moltiplicati per il peso del relativo ramo e a questa sommatoria si aggiunge il bias del nodo che sta compiendo l'operazione.



$$z_j^l = \sum_k x_k^{l-1} w_{l,k}^l + b_j^l \quad (2.1)$$

L'apice  $l$  indica il layer di appartenenza del nodo  $j$  a cui è associato il bias  $b_j^l$ , mentre nell'indicizzazione di  $w$  il pedice  $k$  indica il neurone di partenza ( $x_k^{l-1}$ ) nello strato  $l - 1$  mentre il pedice  $j$  indica il neurone di arrivo nel layer  $l$ . L'insieme dei pesi e dei bias prende il nome di iperparametri della rete.

Al calcolo illustrato precedentemente viene infine applicata una funzione detta attivazione, la quale serve per avere un andamento continuo dell'output del nodo in funzione delle variazioni dei pesi e dei bias. Esempi di funzioni di attivazione utilizzate sono la sigmoide e la ReLU:

$$\text{Sigmoide: } \sigma(z) = \frac{1}{1+e^{-z}} \quad \text{ReLU: } r(z) = \max(0, z) \quad (2.2)$$

Quindi l'output di un generico neurone  $j$  appartenente ad un generico layer  $l$  diventa  $x_j^l = \sigma(z_j^l)$  se si utilizza la sigmoide come attivazione oppure  $x_j^l = r(z_j^l)$  se si usa la ReLU.

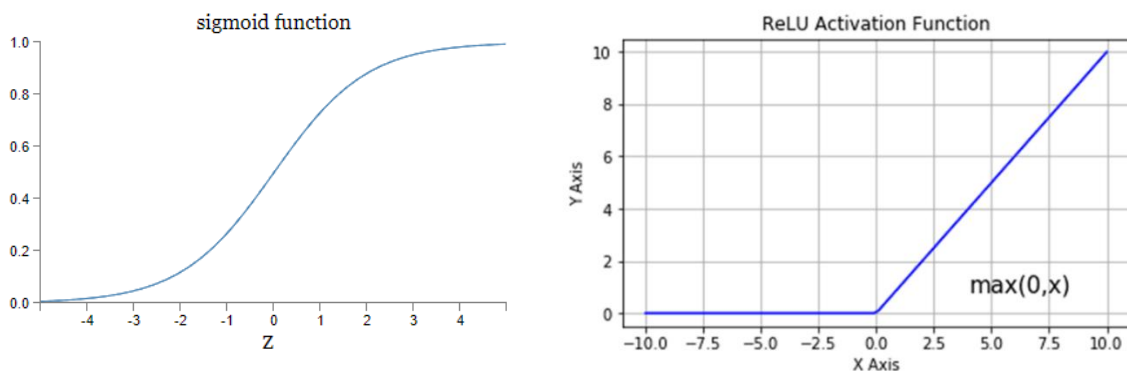


Figura 2.2 grafici delle funzioni di attivazione [10][11].

Per valutare l'efficacia della rete si utilizzano delle funzioni dette *cost function* che comparano gli output del network con i risultati corretti del problema; maggiore è il valore della funzione, minori sono le performance. Una delle cost function più usate, soprattutto per problemi di regressione, esplorati in dettaglio nella sezione seguente, è la *Mean Square Error* (MSE), ovvero l'errore quadratico medio che, essendo una funzione liscia e facilmente derivabile, permette di verificare con precisione anche l'effetto che di piccole variazioni dei pesi e dei bias hanno sui risultati della rete. Indicando con  $y$  il valore atteso e con  $a$  l'output ottenuto dalla rete l'errore quadratico medio può essere espresso come

$$MSE = \frac{1}{2n} \sum_{j=1}^n (y_j - a_j)^2 \quad (2.3)$$

Per riuscire a minimizzare la funzione di costo si studia e utilizza il suo gradiente, che indica il percorso più breve per raggiungere un minimo della funzione. Da considerazioni di carattere matematico è possibile indicare la variazione del costo  $C$  a seguito di una variazione dei parametri della rete  $p$  come:

$$\Delta C = \frac{\partial C}{\partial p} \times \Delta p = \nabla C \times \Delta p \quad (2.4)$$

Scegliendo quindi una variazione dei parametri pari a  $\Delta p = -\eta * \nabla C$ , in cui il termine  $\eta$  è un termine costante, sempre positivo e piccolo a piacere, la variazione della *cost function* risulterà essere:

$$\Delta C = -\eta \times (\nabla C)^2 \quad (2.5)$$

Questa avrà quindi sempre valore negativo, permettendo così alla rete di migliorarsi attraverso le epoche di training. Questo metodo prende il nome di Discesa del Gradiente.

Solitamente per addestrare la rete, soprattutto al fine di riconoscere e catalogare input differenti, si ricorre ad un approccio stocastico, che consiste nel dividere i dati di addestramento in lotti, detti *batch*, ed utilizzare come parametro di ottimizzazione della rete la media dei gradienti di ciascun campione presente nel lotto. Per implementare l'utilizzo delle batch nel metodo di discesa del gradiente, che prende quindi il nome di *Stochastic Gradient Descent*, occorrono particolari e complessi algoritmi di ottimizzazione come ad esempio l'algoritmo di Adam [12] e sue correzioni date dall'implementazione dell'accelerazione del gradiente di Nesterov [13]. Data la spiegazione prolissa di tali metodi, si rimanda ai testi in bibliografia.

## 2.2 Regressione tramite rete neurale

La regressione è un modello di stima del valore atteso di una variabile dipendente partendo da un set di dati di variabili indipendenti. La prima pubblicazione contenente un'applicazione di un modello di regressione è da attribuire a A.-M. Legendre con il metodo dei minimi quadrati, introdotto nel capitolo 1.1 ed esplorato nel dettaglio nel capitolo 1.2.3. In particolare, il modello di regressione lineare si basa sull'assunzione che il valore di output atteso possa essere espresso come una somma pesata degli input ed il fine di talo modello è quello di trovare, tramite considerazioni di carattere algebrico e statistico, i valori dei pesi per cui il risultato del modello sia il più vicino possibile al risultato atteso. Tale processo può essere facilmente ricreato usando una rete neurale addestrata tramite apprendimento con supervisione, ovvero usando un set di dati associati a relative etichette che indicano la corretta soluzione del problema di regressione, che chiameremo  $y$ .

L'algoritmo di implementazione del metodo di regressione tramite rete neurale prende il nome di Retropropagazione, o *Backpropagation*. Esso dapprima computa l'uscita della rete  $a$  a seguito di un input (*feedforward*), calcola il costo e con esso, partendo dal layer di output e procedendo all'indietro nella rete, valuta l'errore indotto da ogni singolo neurone. Infine, con considerazioni sulle derivate calcola di quanto variare gli iperparametri della rete per minimizzare il costo implementando il metodo della discesa del gradiente. Si basa inoltre su due assunzioni principali: la prima è che la funzione di costo sia scrivibile come media dei costi dei singoli output della rete (nel caso in cui questa presenti più nodi nel layer di output, come ad esempio nelle reti di riconoscimento immagini), e la seconda è che la funzione di costo sia scrivibile come funzione dell'output della rete. La funzione di costo quadratico

illustrata in precedenza (2.3) soddisfa entrambi i requisiti, ed inoltre permette di implementare nella rete il metodo dei minimi quadrati.

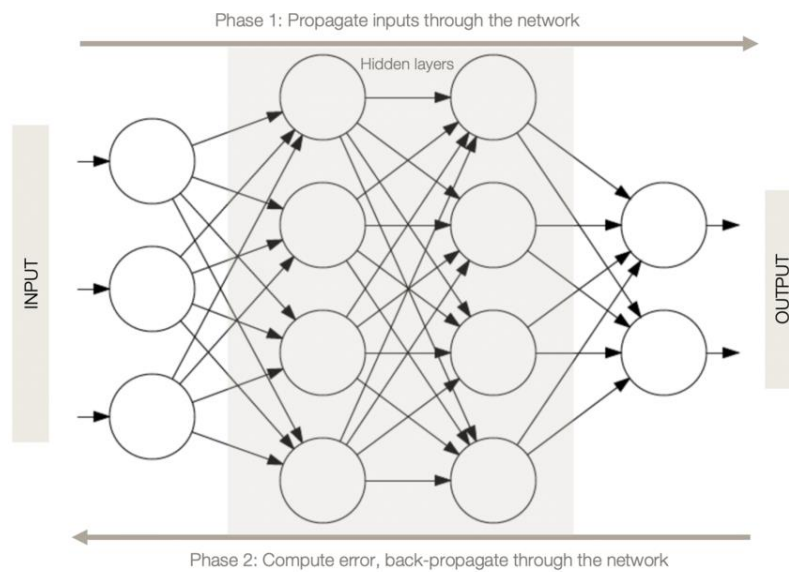


Figura 2.3 Rappresentazione grafica della Retropropagazione [14].

Poiché il problema di regressione tramite reti neurali necessita di un layer di output a neurone singolo  $x^L$  e usa l'errore quadratico medio come funzione di costo, la spiegazione dell'algoritmo di implementazione del metodo di retropropagazione verrà qui illustrato facendo riferimento ad una rete similare.

Facendo riferimento alla indicizzazione definita nell'equazione 2.1, l'errore indotto da un generico neurone  $j$  sul costo finale può essere calcolato come:

$$\delta_j^l = \frac{\partial C}{\partial z_j^l} \quad (2.6)$$

Che indica quanto una modifica di  $z_j^l$  impatta sul costo finale della rete.

Come suggerisce il nome dell'algoritmo, occorre procedere all'indietro nella rete partendo quindi dal layer di output. L'errore indotto dal neurone di output può essere scritto come:

$$\delta^L = \frac{\partial C}{\partial z^L} = \frac{\partial C}{\partial x^L} \frac{\partial x^L}{\partial z^L} = \frac{\partial C}{\partial x^L} \times f'(z^L) \quad (2.7)$$

Dove  $f(z)$  è una generica funzione di attivazione (esempio: sigmoide o ReLU),  $x^L = f(z^L)$  è il valore del neurone di output a seguito dell'applicazione della funzione di attivazione e  $f'(z^L)$  è la derivata di  $x^L$  calcolata rispetto a  $z^L$ , ovvero alla combinazione lineare degli input al neurone. Il primo termine del prodotto indica quanto velocemente varia il costo in funzione del valore del nodo attivato, mentre il secondo termine rappresenta quanto velocemente cambia il valore del neurone stesso a seguito di variazioni di  $z$ .

Il primo termine inoltre può essere riscritto utilizzando come costo il MSE (2.3) e come output atteso l'output dell'ultimo nodo della rete  $x^L$ :

$$\frac{\partial C}{\partial x^L} = -2 \times \left[ \frac{1}{2} (y - x^L) \right] = (x^L - y) \quad (2.8)$$

Che rappresenta il gradiente del costo calcolato rispetto all'output della rete; sostituendolo all'interno dell'equazione dell'errore dell'ultimo nodo, esso diventa  $\delta^L = (x^L - y) * f'(z^L)$ .

Una volta calcolato  $\delta^L$  è possibile procedere all'indietro nella rete calcolando l'errore dei neuroni del layer  $L - 1$  e proseguire con i layer precedenti. In generale l'errore dei neuroni di ogni  $l$ -esimo layer può essere scritto come un vettore di dimensioni pari al numero dei nodi contenuti nel suddetto layer:

$$\bar{\delta}^l = \frac{\partial C}{\partial z^l} = \frac{\partial C}{\partial x^l} \times \bar{f}'(x^l) \quad (2.9)$$

Tramite studio delle derivate risulta possibile riscrivere il primo termine del prodotto come:

$$\frac{\partial C}{\partial x^l} = \frac{\partial C}{\partial x^{l+1}} \frac{\partial \bar{x}^{l+1}}{\partial z^{l+1}} \frac{\partial z^{l+1}}{\partial x^l} \quad (2.10)$$

I primi due termini del prodotto combinati assieme formano il vettore dell'errore dei nodi del layer successivo  $\bar{\delta}^{l+1}$ , mentre il terzo membro indica la derivata dell'input pesato dei neuroni del layer successivo calcolata rispetto ai nodi dell' $l$ -esimo layer, ovvero i loro pesi. Questi possono essere raccolti in una matrice dei pesi  $\bar{w}^{l+1}$  che avrà dimensioni  $J * K$  con  $J$  numero di nodi nel layer  $l + 1$  e  $K$  numero di nodi nel layer  $l$ :

$$\bar{\delta}^l = \left[ (\bar{w}^{l+1})^T * \bar{\delta}^{l+1} \right] \odot \bar{f}'(x^l) \quad (2.11)$$

In cui  $\odot$ , che indica il prodotto Hadamar, prende due vettori  $n$ -dimensionali  $\bar{a}$  e  $\bar{b}$  e ne genera un terzo delle stesse dimensioni, indicato con  $\bar{h}$ , tale per cui  $h_i = a_i * b_i$  per  $i = 1, \dots, n$ .

Una volta ottenuti gli errori di tutti nodi della rete risulta possibile calcolare il gradiente del costo in funzione dei pesi e dei bias della rete tramite studio delle derivate. Per un generico layer  $l$  vale che:

$$\frac{\partial C}{\partial w^l} = \frac{\partial C}{\partial x^l} \frac{\partial x^l}{\partial z^l} \frac{\partial z^l}{\partial w^l}; \quad \frac{\partial C}{\partial b^l} = \frac{\partial C}{\partial x^l} \frac{\partial x^l}{\partial z^l} \frac{\partial z^l}{\partial b^l} \quad (2.12)$$

In cui i primi due termini del prodotto combinati assieme danno l'errore dei nodi dell' $l$ -esimo layer  $\bar{\delta}^l$  (2.9). Ricordando la definizione di  $z^l$  (2.1) e scrivendola in forma vettoriale come  $z^l = w^l * x^{l-1} + b^l$ , è possibile scrivere la derivata di  $z$  rispetto ai pesi e ai bias come:

$$\frac{\partial z^l}{\partial w^l} = x^{l-1}; \quad \frac{\partial z^l}{\partial b^l} = 1 \quad (2.13)$$

Sostituendo nella formula precedente otteniamo infine che le variazioni del costo indotte dai pesi e dai bias della rete sono:

$$\frac{\partial C}{\partial \overline{w}^l} = \overline{x^{l-1}} \times \overline{\delta^l}; \quad \frac{\partial C}{\partial \overline{b}^l} = \overline{\delta^l} \quad (2.14)$$

Che possono essere riscritti passando dalla notazione vettoriale a quella indiciale, usando la notazione descritta a seguito dell'equazione 2.1, come:

$$\frac{\partial C}{\partial w_{jk}^l} = x_k^{l-1} \times \delta_j^l; \quad \frac{\partial C}{\partial b_j^l} = \delta_j^l \quad (2.15)$$

Le formule così calcolate possono infine essere usate per implementare il metodo della discesa del gradiente (2.4, 2.5) per l'addestramento della rete. Per ogni esempio di addestramento  $a$  nel dataset di training si calcola l'uscita della rete e con essa il costo; propagando all'indietro si computano i gradienti del costo in funzione degli iperparametri della rete e si calcola la media dei gradienti su tutti gli esempi di training ( $a = 1, \dots, m$ ); si usa infine la media dei gradienti per calcolare di quanto modificare i pesi ed i bias:

$$\overline{w}^l = \overline{w}^l - \frac{\eta}{m} \sum_{a=1}^m \overline{\delta^{l,a}} \times (\overline{x^{l-1,a}})^T; \quad \overline{b}^l = \overline{b}^l - \frac{\eta}{m} \sum_{a=1}^m \overline{\delta^{l,a}} \quad (2.16)$$

Tale processo viene iterato più volte per riuscire a trovare un minimo del costo e quindi risolvere con maggior accuratezza possibile il problema di regressione; queste iterazioni prendono il nome di Epoche. Il termine  $\eta$  è un coefficiente positivo che prende il nome di *Learning Rate* (lr); prendendone un valore alto ( $\eta \geq 0.01$ ) la rete impara molto velocemente nelle prime epoche di training ma raggiunge uno stallo nell'apprendimento molto presto, scegliendone un valore basso ( $\eta \leq 0.001$ ) la rete migliora più lentamente ma fornisce risultati più affidabili a seguito di numerose epoche di addestramento.

Se si vuole implementare l'approccio stocastico occorre dividere il dataset di  $m$  esempi di training in un numero arbitrario di batch, tali per cui ognuna abbia lo stesso numero di campioni al suo interno. In ogni epoca di addestramento occorre calcolare la media dei gradienti dei campioni contenuti in un a batch, modificare gli iperparametri e ripetere l'operazione per le restanti batch. Questo permette di migliorare l'apprendimento soprattutto per database di grandi dimensioni, in cui l'utilizzo della media sulla totalità dei campioni implica la necessità dell'uso di numeri molto elevati di epoche di training.

## Capitolo 3

### Attività sperimentali

#### 3.1 Ipotesi e modelli

Il fine dell'attività sperimentale è stato quello di programmare una rete neurale in grado di determinare la frequenza di un segnale sinusoidale, discreto, reale e rumoroso, partendo da un insieme di campioni distribuiti omogeneamente in un intervallo di tempo, ovvero compiere una regressione tramite rete neurale. Per fare ciò ci si è basati, inizialmente, sulle considerazioni di Iman Sajedian [15] riproponendo il loro esperimento e generando un database di 100000 segnali reali con frequenze  $f$  casuali comprese tra 1 e 10  $kHz$ , una frequenza di campionamento  $f_s$  di 1  $MHz$  ed un tempo di campionamento di 2 ms; così facendo ad ogni segnale corrispondono 2000 campioni rappresentanti l'ampiezza ad ogni  $\mu s$ . La componente di rumore dei segnali, nell'esperimento di partenza, viene valutata utilizzando come parametro di riferimento il SNR, *Signal-to-Noise Ratio*, ovvero il rapporto tra la potenza del segnale e la potenza del rumore, in particolare generando tutti i segnali con un valore fisso pari a  $SNR = 25dB$  "as a typical setting for a good signal" [15].

Per applicazioni relative alla comunicazione satellitare si utilizza invece il rapporto tra la potenza della portante e la densità di potenza del rumore  $C/N_0$ , *carrier-to-noise-density ratio*, che rappresenta la capacità del ricevitore di agganciarsi al segnale della portante, distinguendola dal rumore; viene espressa in  $dBHz$  e può essere calcolata come il prodotto tra la SNR e la  $f_s$ .

Facendo riferimento ad una distribuzione di rumore gaussiana a media nulla e varianza  $\sigma^2$  le equazioni per il passaggio da SNR a  $C/N_0$  risultano le seguenti:

$$SNR = \frac{C}{2\sigma^2} \quad (3.1)$$

$$C/N_0 = SNR \times f_s \rightarrow C/N_{0dB} = SNR_{dB} + 10 \times \log_{10}(f_s)$$

Poiché la frequenza di campionamento di partenza è  $1MHz = 10^6 Hz$  il valore del rapporto  $C/N_0$  risulta pari a  $25 + 60 = 85 dBHz$ . Una volta calcolato è possibile, invertendo le formule del  $C/N_0$  e del SNR, ricavare il valore della potenza del rumore e con essa la varianza della distribuzione gaussiana, da utilizzare per la generazione della componente rumorosa dei segnali.

$$\sigma^2 = \frac{N_0}{2} f_s \quad (3.2)$$

Poiché il metodo di regressione è un algoritmo di apprendimento con supervisione, sono stati utilizzati dei tensori per la creazione del database, in modo da poter associare ad ogni segnale la sua frequenza corretta. Prima di essere salvate, le frequenze sono state normalizzate per ottimizzare l'apprendimento della rete, poiché le reti di regressione funzionano meglio se il loro output risulta compreso tra 0 ed 1 [15]; tale normalizzazione è stata compiuta dividendo la frequenza reale del segnale per la frequenza

massima possibile, ovvero 10 kHz. Così facendo i valori di output rimangono compresi tra un valore minimo di  $0,1 \left( \frac{1kHz}{10kHz} \right)$  ed uno massimo pari a  $1 \left( \frac{10kHz}{10kHz} \right)$ .

Una volta terminata la generazione dei segnali, questi sono stati divisi in tre gruppi: un primo più massiccio, 72% dei campioni totali, compone l'insieme degli esempi di addestramento, un secondo gruppo, 18% dei campioni totali, è stato usato per la validazione mentre il restante 10% per testare la rete a fine apprendimento. Lo scopo del set di validazione è quello di verificare, alla fine di ogni epoca di training, l'efficacia dell'addestramento presentando alla rete un nuovo insieme di dati, diversi da quelli di training; per fare ciò si calcola la media della loss function di tutti i campioni nel set di validazione dell'epoca corrente e la si compara con quella delle epoche precedenti, se risulta più bassa allora il modello attuale è migliore e lo si salva per usi successivi.

La struttura della rete invece presenta un layer di input di 2000 nodi, pari al numero di campioni di ogni segnale, tre layer nascosti con 2, 2 e 3 neuroni ed un layer di output a singolo neurone.

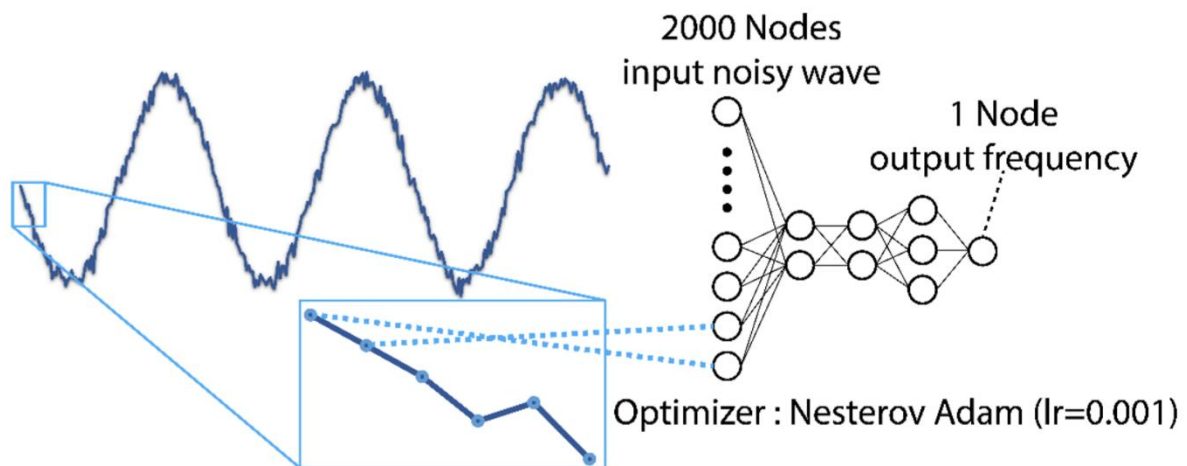


Figura 3.1: struttura della rete di partenza sviluppata da Iman Sajedian [15]

Come funzioni di attivazione per i neuroni è stata scelta la sigmoide (2.2) mentre come loss function è stato utilizzato lo scarto quadratico medio MSE (2.3) ed infine come algoritmo di ottimizzazione il Nesterov-Adam, con un learning rate di 0,001. I risultati ottenuti sono stati in linea con quelli dell'esperimento in bibliografia, al quale si rimanda per una consultazione dettagliata [15].

A seguito di sperimentazioni successive si è riusciti a migliorarne le prestazioni modificando la struttura della rete: la configurazione che si è dimostrata maggiormente efficace presenta 4 layer nascosti da 10, 6, 8 e 4 nodi aventi come funzione di attivazione la ReLU (2.2) per permettere di avere maggiore precisione per segnali con frequenze prossime a 1 KHz o a 10 KHz. La funzione di costo e l'algoritmo di ottimizzazione sono rimaste invariate nel modello modificato mentre il learning rate è stato dimezzato ( $lr = 0,0005$ ) per raggiungere prestazioni migliori a fine addestramento.

Con questi parametri è stato possibile ridurre di quasi un ordine di grandezza il costo calcolato sul dataset di validazione ed è stato ottenuto un errore medio tra i campioni di test pari a 0.5842%.

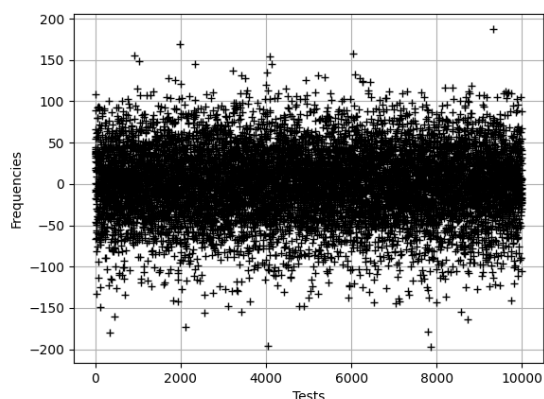


Figura 3.2: differenza in Hz tra frequenza vera e frequenza predetta

Poiché il fine del lavoro è quello di addestrare la rete al riconoscimento di osservabili radiometriche per applicazioni di navigazione deep space, l'esperimento è stato adattato utilizzando segnali generati con una frequenza di campionamento di 1000 Hz; questa scelta è stata fatta perché tali valori sono i più utilizzati per applicazioni di determinazione e studio orbitale. Nello specifico, mentre i ricevitori Radio Science Reciver (RSR) del DSN permettono frequenze di campionamento fino a 16 MHz, l'accuratezza della modellazione preliminare del modello dinamico agente sul satellite permette di sottocampionare e filtrare i dati ricevuti, ottenendo generalmente valori di campionamento dell'ordine delle migliaia [16]. Inoltre per limitazioni indotte dalle operazioni di combinazione lineare computate tra i nodi della rete, le quali accettano come dati di ingresso solo numeri interi o reali, i segnali utilizzati sono stati privati della parte immaginaria, che può essere analizzata separatamente per stimare con più confidenza il risultato creato. Le osservabili sono quindi state generate come segnali monocomponente ergodici sotto l'ipotesi di rumore bianco gaussiano additivo, la cui varianza è stata calcolata tramite formula 3.2 con valori del rapporto  $C/N_0$  di 0, 30, 40, 50, 60 e 90 dBHz per testare l'efficacia della rete addestrata con diversi valori della componente di rumore.

È stato inoltre introdotto un layer di convoluzione bidimensionale all'ingresso della rete avente un kernel di dimensioni 3, uno stride di 1 e padding singolo.

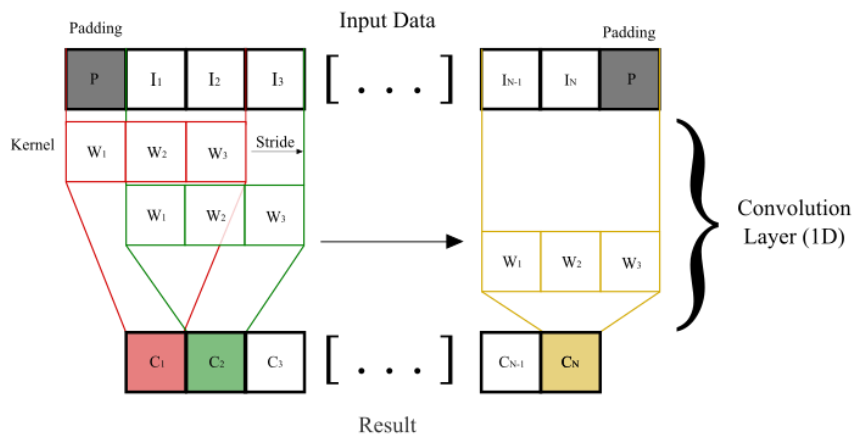


Figura 3.3: Layer convoluzionale bidimensionale con kernel=3, stride=1 e padding=1.



Tale passaggio agisce come un filtro sui dati in ingresso alla rete, permettendo, teoricamente, di attenuare la componente di rumore ottenendo risultati più precisi. L'introduzione del layer di convoluzione ha però causato un notevolmente rallentato delle operazioni di addestramento dal punto di vista computazionale; per ovviare a tale problema l'addestramento della rete è stato suddiviso in 4 fasi ognuna da 50 epoche ciascuna. Per fare ciò è stato caricato il modello migliore ottenuto del ciclo di training precedente, ovvero quello che presentava il costo minore sul dataset di validazione, prima dell'inizio del ciclo di addestramento successivo, riuscendo inoltre così a scartare eventuali modelli mal performanti e a abbassare il livello del learning rate ad ogni ciclo per permettere maggiore precisione delle modifiche degli iperparametri della rete. Il primo ciclo è stato compiuto con un learning rate di 0.001, il secondo ed il terzo con 0.0005 e l'ultimo con 0.0001. La rete che è risultata maggiormente performante, a seguito di numerose prove e modelli, presenta 3 strati nascosti completamente connessi con rispettivamente 16, 8, e 4 neuroni. Essi sono posti a valle di un layer di convoluzione a singolo canale di input, il quale accetta quindi dati tensoriali di dimensioni  $[1, N]$  con  $N$  numero di campioni (nel caso del modello 1000), e quattro canali di output ottenendo quindi un tensore  $[4, 1000]$ ; questo viene filtrato a sua volta da un layer di *Max-Pooling* con filtro di lunghezza 4 e stride di 4 che riduce la dimensione del tensore dei dati a  $[4, 250]$ . Così facendo è possibile concatenare i dati per ottenere un tensore delle stesse dimensioni di quello di input originale, ovvero  $[1, 4 \times 250] = [1, 1000]$ , al quale è però stato applicato un filtro che ne assorbe il rumore.

Example of MaxPooling operation:  
filter size = 1x3, stride = 3

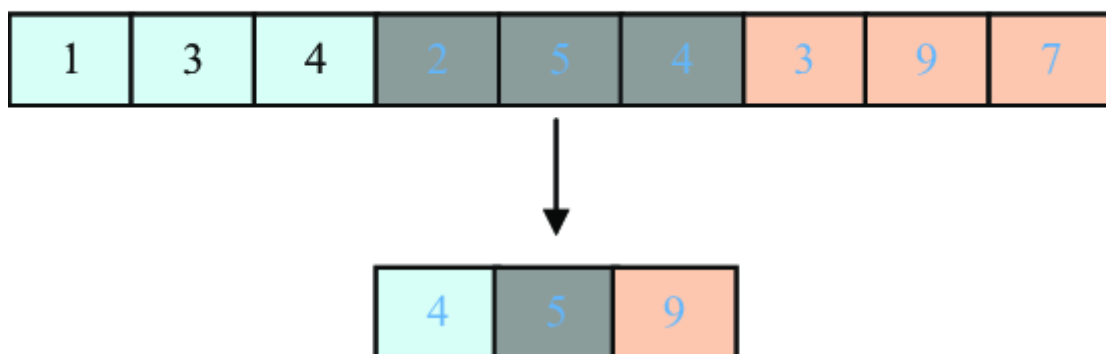


Figura 3.4: esempio di Max-Pooling con filtro di dimensione 3 e stride di 3 [17].

## 3.2 Risultati e discussione

I segnali utilizzati per lo studio sono stati generati con una frequenza di campionamento  $f_s$  di 1 kHz e i valori delle frequenze dei singoli segnali  $f_r$  soddisfano i limiti del teorema di campionamento di Shannon:

$$f_s \geq 2 * \text{Max}(f_r) \quad (3.3)$$

Imponendo quindi il range di frequenze dell'esperimento tra 1 Hz e 500 Hz. Per quanto invece riguarda la componente rumorosa è stato invece optato per un valore del rapporto  $C/N_0$  di 60 dBHz, al quale quindi corrisponde un SNR di 30 dB. Con questi dati è stato possibile ottenere un valore della loss function sul dataset di training poco inferiore a  $2.8 \times 10^{-6} \text{ Hz}^2$  e di circa  $2.6 \times 10^{-6} \text{ Hz}^2$  su quello di validazione. Poiché i costi sono calcolati usando il MSE, è possibile confrontarli direttamente con la disuguaglianza di Cramèr-Rao (CRLB), che fornisce una stima sul minimo errore quadratico medio raggiungibile con un determinato set di dati per la stima di frequenza.

Per il dataset utilizzato, il CRLB risulta essere circa  $1.52 \times 10^{-7} \text{ Hz}^2$ .

$$\text{CRLB} = \frac{6f_s^3}{(2\pi)^2 N(N^2 - 1)C/N_0} \quad (3.4)$$

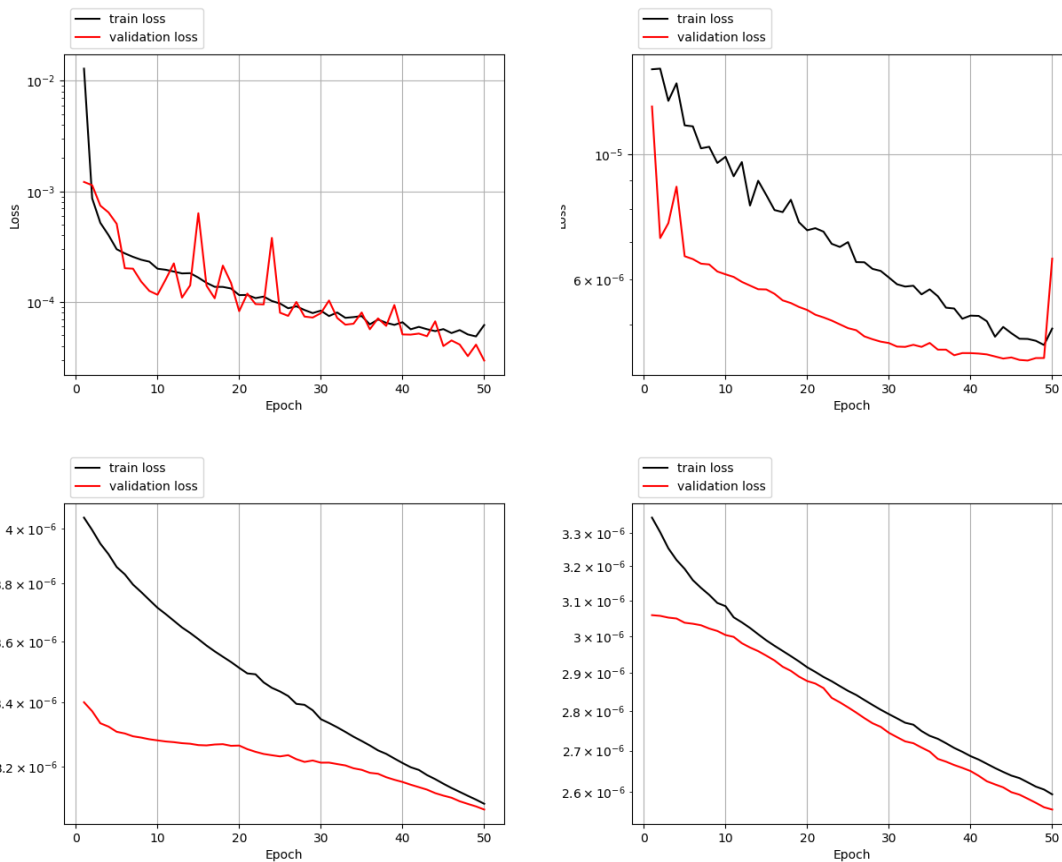


Figura 3.5: Andamento della loss function sul dataset di addestramento, in nero, e di validazione, in rosso, durante i 4 cicli di addestramento: Primo ciclo in alto a sinistra con  $lr=0.001$ ; secondo ciclo in alto a destra con  $lr=0.0005$ ; terzo ciclo in basso a sinistra con  $lr=0.0001$  ed ultimo ciclo in basso a destra con  $lr=0.0001$ .

La rete addestrata ha presentato un errore medio sul dataset di test pari a 0.5431%, calcolato come la media dell'errore percentuale dei singoli test, a sua volta valutato come segue:

$$error_{\%} = 100 \times \frac{|f_p - f_r|}{f_r}$$

Dove  $f_p$  è la frequenza predetta dalla rete mentre  $f_r$  è la frequenza reale con cui il segnale è stato generato. Utilizzando la stessa definizione di errore è stata anche definita l'accuratezza della rete, considerando una predizione corretta quando l'errore fosse minore dell'1% della frequenza corretta (esempio: se la frequenza vera vale 1 Hz, allora la predizione è corretta se  $|f_p - 1| \leq 0.01\text{Hz}$ ). Con la soluzione sopra illustrata, la rete ha presentato un'accuratezza del 95.76% su un totale di 10000 campioni adibiti alla fase di test.

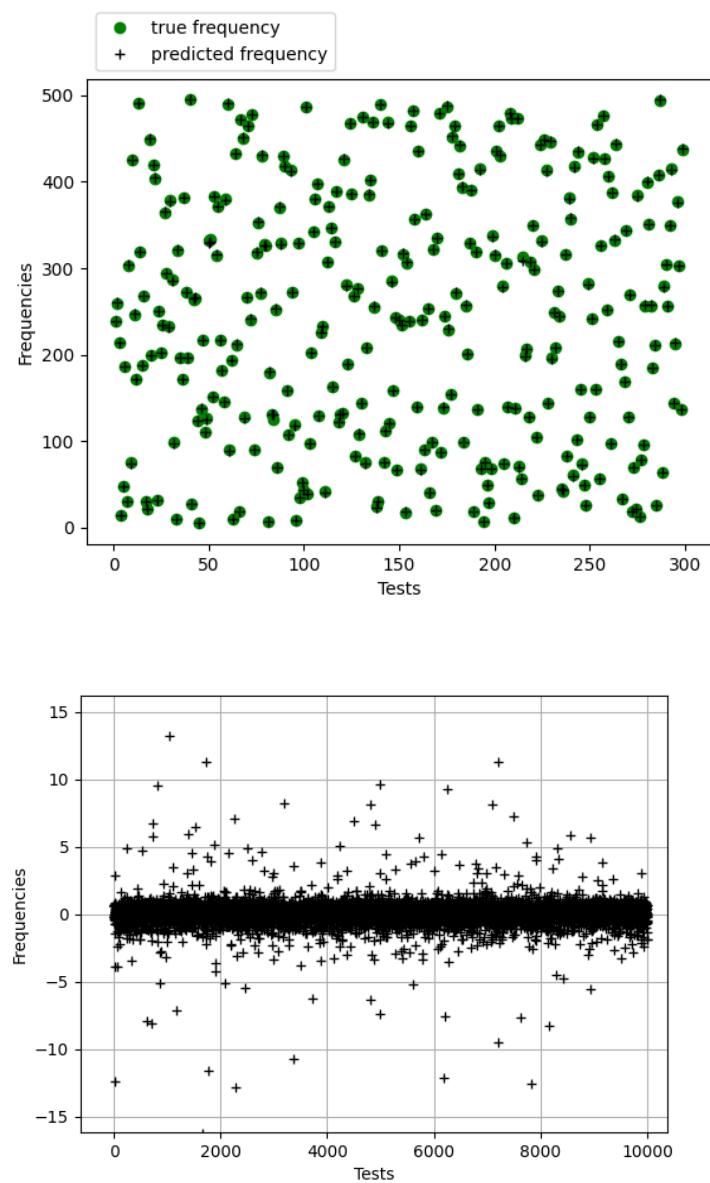


Figura 3.6: Rappresentazioni della precisione della rete, nell'immagine in alto sono rappresentate in verde le frequenze reali e a destra quelle predette; in quella in basso sono graficate le differenze in Hz tra il segnale predetto e quello reale.

Ripetendo l'addestramento per diversi valori del rapporto  $C/N_0$  è stato infine possibile verificare le prestazioni della rete al variare della componente di rumore dei segnali. In particolare è possibile notare come la rete presenti performance estremamente povere per segnali in cui la componente di rumore risulta elevata ( $0 \leq C/N_{0_{dBHz}} < 30$ ); per osservabili generate con un  $C/N_0$  compreso tra 30 e 60 dBHz l'errore quadratico medio tra i campioni di test risulta essere in prossimità del minimo teorico stimato tramite CRLB (3.4), per poi stabilizzarsi attorno a  $10^{-6}$  per segnali poco rumorosi.

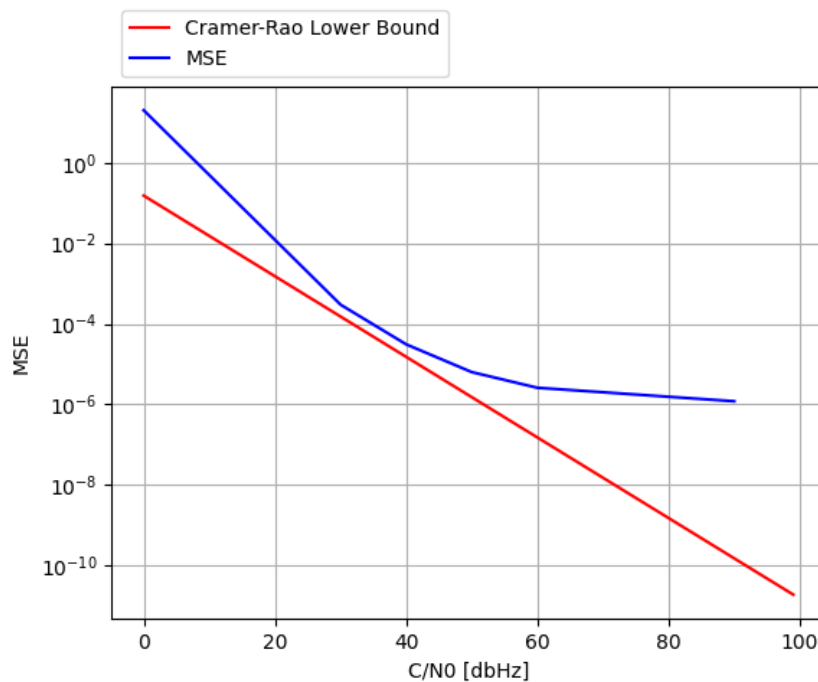


Figura 3.7: Confronto tra CRLB e MSE al variare del  $C/N_0$

Per valori alti di rumore ci si discosta dal minimo teorico, ciò risulta essere un avvenimento comune per la risoluzione di problemi non lineari come la stima di frequenza e quindi l'errore è attribuibile alla natura del problema più che al modello utilizzato per la soluzione. Per bassi valori della componente di rumore invece le performance della rete tendono ad un asintoto, assestando l'accuratezza attorno al 95%, impedendo quindi di raggiungere i livelli di precisioni richiesti per applicazioni Deep Space, per le quali è richiesta un'accuratezza almeno pari a 99.9%. Per il range di rumore nominale, ovvero per valori di  $C/N_0$  compresi in un intervallo tra 30 e 50 dBHz, i risultati ottenuti dalla rete risultano soddisfacenti mentre per segnali con componenti di rumore al di fuori di tale intervallo i risultati ottenuti non soddisfano le richieste del campo di applicazione, richiedendo quindi ulteriori studi degli algoritmi di stima.

# Conclusione

In conclusione, le reti neurali hanno dimostrato essere potenzialmente un'utile strumento per utilizzi in ambito di analisi di segnali per applicazioni deep space. Una volta addestrate risultano essere generalmente accurate e rapide, impiegando soli pochi secondi per analizzare grandi quantità di dati (per testare il modello di rete addestrata sull'intero database di 100000 campioni occorrono solo 11s); ciò non è però senza limitazioni:

La prima è imposta dai dati di input, se si vuole una rete altamente performante occorre limitare la variabilità dei dati in ingresso, andando così a creare un programma specializzato nel riconoscimento di un set finito di dati; esempio di ciò lo si può vedere nel presente lavoro, nel quale per poter ottenere una buona accuratezza è stato necessario limitare la generazione dei segnali ad un singolo valore di frequenza di campionamento  $f_s$  e del  $C/N_0$ . Per ovviare a tale limitazione si potrebbero creare dei dataset molto più massicci ed eterogenei, per i quali sarebbe richiesta la costruzione di reti di grandi dimensioni, portando quindi a lunghi tempi di addestramento e alti pesi dei file dei modelli salvati durante tali procedure.

La seconda limitazione è data dalla potenza di calcolo a disposizione, poiché dovendo valutare numerose combinazioni lineari e relative funzioni di attivazione ad ogni passaggio, assieme ai loro gradienti al fine dell'ottimizzazione dell'apprendimento, gli aumenti delle dimensioni della rete portano alla necessita di avere dispositivi dedicati oppure disporre di risorse computazionali elevate.

Complessivamente gli algoritmi sviluppati si sono dimostrati non solo funzionali, ma anche semi-ottimali per il range ben definito della componente di rumore delimitato da valori del  $C/N_0$  compresi tra 30 e 50 dBHz; per i quali il MSE della rete addestrata risulta prossimo al CRLB, ottenendo così risultati affidabili in tempi ridotti e permettendo quindi l'implementazione di tali metodi in casi di stima real-time.

Mentre il presente lavoro dimostra la fattibilità dell'implementazione di reti neurali convoluzionali per la stima di frequenza di osservabili reali, per lavori futuri si potrebbe esplorare l'applicazione dei layer di convoluzione a doppio canale di ingresso per riuscire a studiare non solo segnali reali ma anche segnali analitici generati scindendo la parte reale da quella immaginaria e rimoltiplicando quest'ultima per un fattore  $-j$  in modo da ottenere dei tensori di dimensione  $[2, N]$ , ovvero con due canali contenenti uno la componente reale del segnale ed il secondo la componente immaginaria, tradotta in numeri reali per permetterne la computazione. Tale passaggio potrebbe garantire maggiore accuratezza dei risultati poiché permette la reintegrazione nel problema di una parte del segnale altrimenti non utilizzabile per via delle limitazioni imposte dagli algoritmi di combinazione lineare e delle funzioni di attivazione.

# Bibliografia

- [1] Bob Schutz, Byron Tapley, and George H Born. *Statistical orbit determination*. Elsevier, 2004.
- [2] Jonathan Tennenbaum, Bruce Director. *How Gauss Determined The Orbit of Ceres*. Schiller Institute, 1998.
- [3] A. Abdulle, G.Wanner. *200 years of least squares method*. Elem. Math. 2002
- [4] David A Vallado. *Fundamentals of astrodynamics and applications*, volume 12. Springer Science & Business Media, 2001.
- [5] Christoph Reigber, F Sanso, and R Rummel. *Theory of satellite geodesy and gravity field determination. Lecture notes in Earth sciences*, 25, 1989.
- [6] Catherine L Thornton and James S Border. *Radiometric tracking techniques for deep-space navigation*. John Wiley & Sons, 2003.
- [7] Sami W. Asmar, *Radio Science Techniques for Deep Space Exploration*, Wiley 2022.
- [8] Andrea Togni, *Astrodynamic, Orbit Determination, and Applications*.
- [9] IBM Cloud Education, *Reti Neurali*, [Online]. Available: <https://www.ibm.com/it-it/cloud/learn/neural-networks>
- [10] Michael Nielsen, *Neural Networks and Deep Learning*, [Online]. Available: <http://neuralnetworksanddeeplearning.com/index.html>
- [11] Applied Intelligence, *Rectified Linear Unit (ReLU)* [Online]. Available: [https://www.researchgate.net/figure/Rectified-Linear-Unit-ReLU-72\\_fig4\\_348715472](https://www.researchgate.net/figure/Rectified-Linear-Unit-ReLU-72_fig4_348715472)
- [12] Sebastian Bock, Martin Weiß, *A Proof of Local Convergence for the Adam Optimizer*, International Joint Conference on Neural Networks (IJCNN) 2019
- [13] Timothy Dozat, *INCORPORATING NESTEROV MOMENTUM INTO ADAM*, Workshop track - ICLR 2016
- [14] The Economist, *Deep-learning backpropagation in a nutshell*, [Online] Available: [https://www.researchgate.net/figure/Deep-learning-backpropagation-in-a-nutshell-The-Economist-2016-Tim-Jones-2017\\_fig11\\_330382022](https://www.researchgate.net/figure/Deep-learning-backpropagation-in-a-nutshell-The-Economist-2016-Tim-Jones-2017_fig11_330382022)

- [15] Iman Sajedian, Junsuko Rho, *Accurate and instant frequency estimation from noisy sinusoidal waves by deep learning*, Nano Convergence 2019.
- [16] Deep Space Network (DSN) External Interface Specification, *Radio Science Receiver Standard Formatted Data Unit (SFDU)*, 2009
- [17] *The 1D MaxPooling with kernel size 1x3 and stride of 3*, [Online] Available: [https://www.researchgate.net/figure/The-1D-MaxPooling-with-kernel-size-1x3-and-stride-of-3\\_fig1\\_348645502](https://www.researchgate.net/figure/The-1D-MaxPooling-with-kernel-size-1x3-and-stride-of-3_fig1_348645502)