

Università della Calabria

Dipartimento di Economia Statistica e Finanza

Corso di Laurea in **Data Science per le Strategie Aziendali**

**Sviluppo di una Rete Neurale Artificiale
per la stima dell'età dell'Abalone**

Relatore

Prof. Paolo Carmelo Cozzucoli

Candidati

Pierfrancesco Lindia
Francesco De Nisi

256641
269762

Anno Accademico 2025 / 2026

Indice

Sommario	1
1 Introduzione e Contesto	1
2 Dataset Abalone: Definizione Target, Analisi Descrittiva e Preprocessing	1
2.1 Normalizzazione Variabile Target	2
2.2 Procedura operativa per normalizzazione e de-normalizzazione del target	3
2.3 Analisi Descrittiva Variabile Target	4
2.4 Analisi delle relazioni tra predittori e multicollinearità	5
2.5 Normalizzazione delle variabili esplicative	7
3 Progettazione della rete neurale artificiale per regressione	8
3.1 Inquadramento del modello: Multi-Layer Perceptron (MLP) feed-forward	9
3.2 Funzioni di attivazione nello strato nascosto: Logistic e Tanh	9
3.3 Output lineare: coerenza con un problema di regressione	10
3.4 Funzione di costo: SSE e relazione con MSE	10
3.5 Ottimizzazione iterativa e concetto di convergenza	11
3.6 Selezione degli iperparametri e strategia di tuning	11
4 Risultati del Tuning della Rete Neurale	12
4.1 Configurazione ottimale selezionata	14
4.2 Prestazioni sul test set (scala originale in anni)	14
4.3 Diagnostica del modello sul test set	15
5 Confronto con modelli benchmark: OLS e regressioni regolarizzate	17
6 Conclusioni	18

1 Introduzione e Contesto

L’abalone è un mollusco marino appartenente alla famiglia Haliotidae, caratterizzato da una conchiglia appiattita auriforme e da un corpo aderente al substrato bentonico grazie a un robusto piede muscolare. Tale specie riveste un elevato valore economico globale, rappresentando una risorsa cardine sia per la pesca professionale che per l’acquacoltura intensiva. Oltre alla rilevanza commerciale, l’abalone svolge un ruolo ecologico cruciale negli ecosistemi costieri, contribuendo al controllo delle biomasse algali. Tradizionalmente, la determinazione dell’età dell’abalone avviene tramite il conteggio degli anelli di accrescimento sulla conchiglia, una procedura analoga alla dendrocronologia arborea. Sebbene tale metodo rappresenti il gold standard biologico, esso presenta criticità insormontabili per l’applicazione su larga scala: richiede strumentazione di laboratorio, tempi lunghi e, soprattutto, il sacrificio dell’animale. Il presente studio propone un’alternativa basata sul Machine Learning per inferire l’età a partire da variabili fisiche (peso, dimensioni) rilevabili in situ, offrendo uno strumento di supporto decisionale rapido ed economico. Il presente elaborato descrive la progettazione e l’implementazione di una Rete Neurale Artificiale (ANN) per la stima dell’età di esemplari di abalone (*Haliotis spp.*) a partire da misurazioni morfometriche non invasive. L’obiettivo è valutare se un approccio non lineare possa ridurre le limitazioni delle procedure tradizionali di stima dell’età, spesso più onerose e potenzialmente invasive. Il modello proposto è stato addestrato e valutato confrontandone le prestazioni con tecniche di regressione lineare (OLS) e regolarizzata (Ridge, Lasso, Elastic Net).

2 Dataset Abalone: Definizione Target, Analisi Descrittiva e Preprocessing

Il dataset Abalone (UCI) raccoglie osservazioni relative all’esemplare descritte attraverso misure morfometriche (lunghezza, diametro, altezza) e componenti di massa (peso totale e pesi di alcune parti anatomiche). L’obiettivo modellistico è stimare l’età dell’animale mediante grandezze facilmente acquisibili, evitando procedure dirette più costose. Nel dataset originale l’età non è disponibile come misura diretta; viene invece utilizzato il numero di anelli del guscio (Rings) come proxy dell’età. In letteratura e nella documentazione UCI, l’età in anni viene approssimata come:

$$Age = Rings + 1.5$$

dove il termine costante rappresenta una correzione convenzionale legata alle fasi iniziali di crescita non contabilizzate dal conteggio degli anelli. Questa trasformazione consente di definire una variabile risposta continua e, di conseguenza, il problema si configura come un task di *regressione*. In tale contesto, l’obiettivo è stimare una funzione

$$f(\cdot)$$

in grado di mappare l’insieme delle variabili descrittive X in una stima dell’età

$$\widehat{Age} = f(X).$$

Tra i predittori è presente una variabile qualitativa relativa al sesso dell'animale (maschio, femmina, infant). Poiché sia le reti neurali sia i modelli lineari operano su input numerici, una variabile nominale deve essere trasformata in una rappresentazione adatta a non introdurre artificiosamente un ordinamento tra categorie. La scelta standard è la codifica one-hot (dummy encoding): ogni categoria viene rappresentata tramite una variabile binaria che indica la presenza/assenza di quel livello.

Questa strategia consente al modello di apprendere eventuali differenze sistematiche tra categorie senza assumere relazioni ordinali inesistenti (ad esempio, evitare codifiche tipo 1–2–3 che implicherebbero distanze numeriche tra classi). Dopo la codifica, la variabile originale non numerica non è più necessaria ed è sostituita dalle corrispondenti variabili binarie. Inoltre, c'è da specificare che la variabile Age è costruita da un'altra variabile presente nel dataset Ring che contiene informazioni quasi deterministiche su Age, l'inclusione di tale variabile tra i predittori sarebbe come "barare" ai fini predittivi, rendendo la stima artificialmente semplice e non rappresentativa di un contesto reale. Per questo motivo Rings viene esclusa dall'insieme delle variabili esplicative.

La variabile Sex è una variabile categoriale nominale. Per garantire una rappresentazione corretta, la variabile categoriale Sex viene trasformata tramite codifica *one-hot*, generando tre variabili binarie (Sex_F , Sex_M , Sex_I). Per ogni osservazione vale la condizione:

$$Sex_F + Sex_M + Sex_I = 1,$$

così che ciascun individuo appartenga esattamente a una delle tre categorie.

È stata inoltre verificata la qualità del dataset in termini di completezza e plausibilità dei valori. L'assenza (o l'eventuale limitata presenza) di valori mancanti e la natura fisica delle misure riducono la necessità di imputazione; tuttavia, possibili valori estremi o anomalie (ad esempio misure molto piccole rispetto alla distribuzione) vengono valutati mediante analisi descrittiva e, se necessario, considerati nell'interpretazione delle prestazioni.

2.1 Normalizzazione Variabile Target

Nel contesto delle reti neurali per regressione, la scelta della scala delle variabili (in particolare della variabile risposta) ha un impatto diretto sulla stabilità numerica dell'addestramento e sulla rapidità di convergenza. L'apprendimento di una ANN avviene minimizzando una funzione di perdita (ad esempio SSE/MSE) mediante metodi iterativi di ottimizzazione basati sul gradiente. In tali procedure, l'ampiezza dei gradienti dipende dalla scala dei dati: se il target assume valori relativamente grandi o se presenta una scala molto diversa rispetto agli input, l'ottimizzazione può diventare più instabile e sensibile agli iperparametri (in particolare al learning rate), con possibili fenomeni di oscillazione, rallentamento della convergenza o difficoltà nel raggiungimento di un minimo soddisfacente.

Per mitigare questi effetti, la variabile risposta Age viene normalizzata mediante trasformazione *min–max*, che mappa i valori nell'intervallo $[0, 1]$. Indicando con age_{\min} e age_{\max} rispettivamente il minimo e il massimo della risposta sulla scala originale, la

trasformazione è definita da:

$$Age_{\text{scaled}} = \frac{Age - age_{\min}}{age_{\max} - age_{\min}}.$$

Questa scelta presenta due vantaggi principali:

- **Stabilità dell'ottimizzazione:** la funzione di perdita e i gradienti tendono a rimanere su ordini di grandezza più controllati, facilitando l'addestramento e riducendo la sensibilità a valori estremi della risposta.
- **Coerenza con la normalizzazione degli input:** riportare *target* e predittori su scale comparabili evita che alcune variabili “dominino” numericamente il processo di apprendimento, migliorando la convergenza del modello.

Poiché il modello viene addestrato nello spazio normalizzato, anche le predizioni risultano espresse nella scala [0, 1]. Per interpretarle correttamente in anni e per calcolare metriche di errore significative sulla scala originale, è necessario applicare la trasformazione inversa (de-normalizzazione):

$$\widehat{Age} = \widehat{Age}_{\text{scaled}} \cdot (age_{\max} - age_{\min}) + age_{\min}.$$

La conservazione dei parametri age_{\min} e age_{\max} è quindi essenziale: senza di essi non sarebbe possibile ricostruire in modo coerente l'età stimata nella scala originale, mantenendo la corrispondenza con la trasformazione impiegata in fase di training.

2.2 Procedura operativa per normalizzazione e de-normalizzazione del target

La procedura adottata si articola nei passaggi seguenti.

1. **Determinazione dei parametri sulla scala originale.** Dopo aver definito la variabile risposta *Age* in anni, si calcolano i parametri necessari alla trasformazione min–max:

$$age_{\min} = \min(Age), \quad age_{\max} = \max(Age).$$

Questo passaggio deve essere eseguito prima della normalizzazione, poiché una volta trasformata la variabile in [0, 1] i suoi estremi diverrebbero trivialmente 0 e 1, rendendo impossibile ricostruire correttamente la scala in anni.

2. **Normalizzazione del target (fase di “andata”).** Utilizzando tali parametri, l'età viene mappata nell'intervallo [0, 1] tramite:

$$Age_{\text{scaled}} = \frac{Age - age_{\min}}{age_{\max} - age_{\min}}.$$

Da questo momento in poi, la rete neurale viene addestrata a stimare Age_{scaled} , ossia l'età nella scala normalizzata.

3. **De-normalizzazione delle predizioni (fase di “ritorno”).** Le predizioni prodotte dal modello risultano espresse come $\widehat{Age}_{\text{scaled}}$. Per ottenere una stima interpretabile in anni, si applica l’ inversa della trasformazione:

$$\widehat{Age} = \widehat{Age}_{\text{scaled}} \cdot (\text{age}_{\max} - \text{age}_{\min}) + \text{age}_{\min}.$$

La trasformazione inversa viene applicata prima del calcolo delle metriche finali, così che l’errore sia espresso in unità reali (anni). Per coerenza, anche i valori osservati del set di test vengono riportati alla scala originale prima del confronto.

4. **Coerenza tra training e test (nota metodologica).** Affinché la procedura sia corretta, i parametri age_{\min} e age_{\max} devono essere gli stessi in tutte le fasi (normalizzazione dei dati e de-normalizzazione delle predizioni). La prassi più rigorosa prevede di stimare tali parametri sul solo training set e applicarli invariati al test, evitando qualunque influenza (anche indiretta) delle informazioni del set di valutazione.

2.3 Analisi Descrittiva Variabile Target

Prima di addestrare un modello è necessario comprendere la struttura della variabile risposta, in questo caso l’età Age . L’analisi descrittiva ha tre scopi principali:

- **Quantificare la scala del problema**, analizzando l’intervallo di variazione e la variabilità dell’età;
- **Valutare la forma della distribuzione**, con riferimento a simmetria/asimmetria, presenza di code e grado di concentrazione;
- **Individuare eventuali valori estremi**, potenzialmente in grado di influenzare l’addestramento e le metriche di valutazione.

Questa fase non mira a “spiegare” causalmente l’età, ma permette di anticipare possibili criticità e di interpretare correttamente le prestazioni dei modelli stimati.

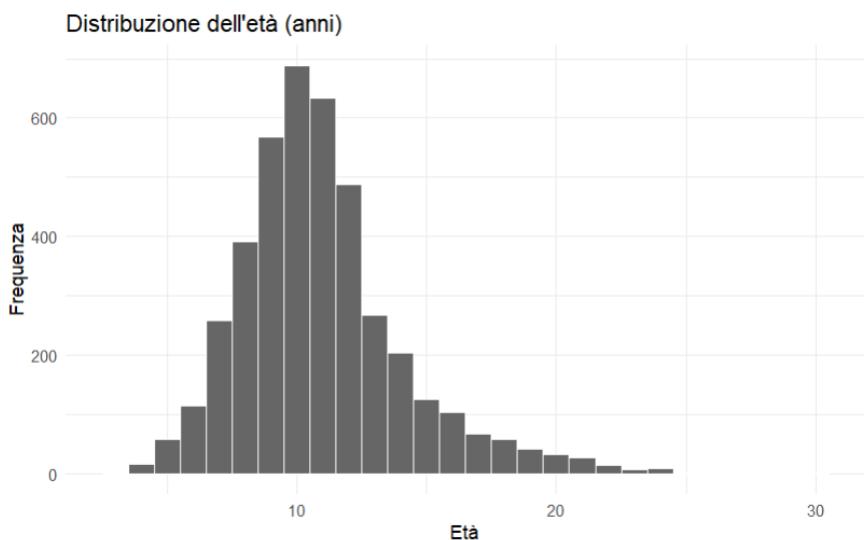


Figura 1: Distribuzione Variabile Target

La Figura 5 mostra la distribuzione empirica della variabile risposta *Age* (in anni). La distribuzione appare unimodale, con un picco principale concentrato attorno a valori prossimi a 10–11 anni, e presenta una marcata asimmetria positiva: la maggior parte delle osservazioni è concentrata nelle età intermedie, mentre la frequenza decresce progressivamente al crescere dell'età, formando una coda destra che si estende verso valori più elevati (circa oltre 18–20 anni). Sulla base di quanto emerso ci sono delle implicazioni che devono essere necessariamente.

- **Sbilanciamento del target e generalizzazione sugli estremi** Poiché le osservazioni nelle fasce di età elevate sono poche, il modello dispone di meno esempi da cui apprendere il comportamento in coda. Di conseguenza, è plausibile attendersi prestazioni migliori nella regione centrale (età più frequenti) e peggiori sugli estremi, soprattutto per età alte.
- **Scelta e interpretazione delle metriche (MSE vs MAE)** In presenza di una coda destra e di valori rari ma distanti, l'uso della MSE (o RMSE) è particolarmente sensibile agli errori sulle osservazioni estreme, poiché penalizza quadraticamente gli scostamenti. Per questo motivo è utile affiancare anche la MAE, più robusta rispetto a pochi errori grandi e più facilmente interpretabile come “errore medio in anni”.
- **Coerenza con ipotesi di non linearità** La crescita biologica e la relazione tra dimensioni/pesi e età possono non essere lineari e possono variare tra fasce di età (ad esempio, crescita più rapida nelle fasi iniziali e più lenta in età avanzata). La presenza di una distribuzione asimmetrica e di dinamiche diverse lungo la scala dell'età rende plausibile che un modello non lineare (ANN) possa catturare meglio tali pattern rispetto a modelli puramente lineari.

2.4 Analisi delle relazioni tra predittori e multicollinearità

Dopo aver caratterizzato la variabile risposta, è opportuno analizzare le relazioni tra i predittori quantitativi e, in generale, tra predittori e target. Questa fase ha due scopi principali:

- **Valutare l'associazione** tra ciascuna variabile quantitativa e l'età, identificando segnali potenzialmente utili per la predizione;
- **Diagnosticare la ridondanza informativa:** misure morfometriche e pesi descrivono spesso aspetti sovrapposti della dimensione dell'animale e possono risultare fortemente correlati tra loro.

Per una prima misura dell'associazione lineare tra due variabili quantitative X e Y si utilizza la correlazione di Pearson:

$$\rho_{X,Y} = \frac{\text{Cov}(X, Y)}{\sigma_X \sigma_Y},$$

dove ρ assume valori in $[-1, 1]$. In generale:

- $\rho \approx 1$: forte associazione lineare positiva;

- $\rho \approx -1$: forte associazione lineare negativa;
- $\rho \approx 0$: assenza di legame lineare (ma non necessariamente assenza di legame non lineare).

Osservando le correlazioni con *Age*, emerge che l'associazione lineare è moderata per tutte le variabili considerate (indicativamente tra ~ 0.42 e ~ 0.63). In particolare:

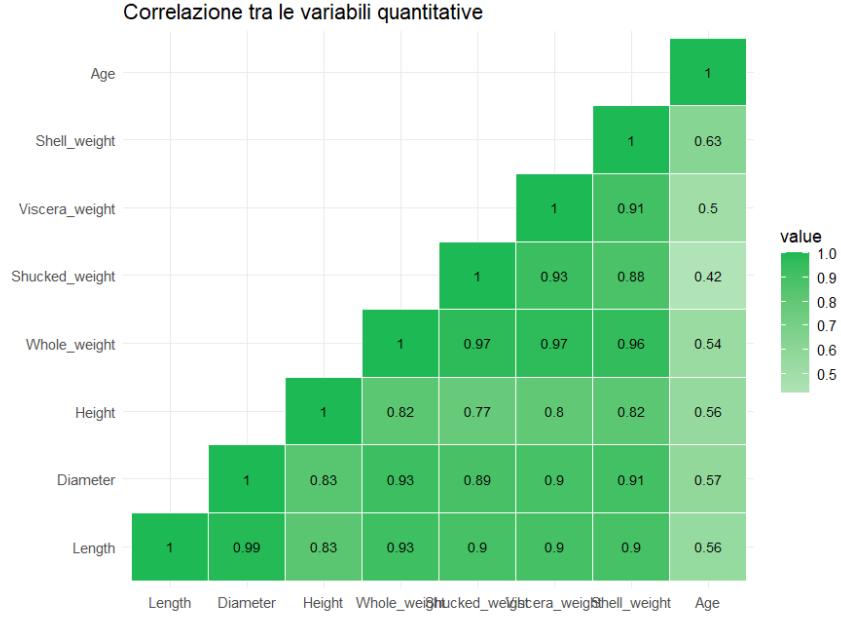


Figura 2: Enter Caption

- la variabile più associata linearmente all'età è *Shell_weight* ($\rho \approx 0.63$), suggerendo che il peso del guscio contenga un segnale informativo relativamente più forte per la stima dell'età;
- le misure geometriche *Length*, *Diameter* e *Height* mostrano correlazioni simili con *Age* ($\rho \approx 0.56$ – 0.57);
- *Whole_weight* presenta una correlazione moderata ($\rho \approx 0.54$);
- *Shucked_weight* risulta la meno correlata con *Age* ($\rho \approx 0.42$), indicando un legame lineare più debole.

Nel complesso, questi valori indicano che la predizione dell'età non può essere ricondotta a una singola variabile dominante: il segnale informativo è distribuito tra più misure e, con buona probabilità, include componenti non lineari o interazioni tra predittori, coerenti con la complessità dei processi biologici di crescita.

Un altro risultato molto evidente è la presenza di correlazioni molto elevate tra i predittori. Ad esempio:

- *Length* e *Diameter* risultano quasi perfettamente correlate ($\rho \approx 0.99$);

- i pesi sono fortemente inter-correlati: *Whole_weight* con *Shucked_weight* e *Viscera_weight* raggiunge valori molto alti ($\rho \approx 0.97$), mentre con *Shell_weight* è anch'essa elevata ($\rho \approx 0.96$);
- anche tra le componenti di peso si osservano correlazioni elevate (ad esempio *Shucked_weight–Viscera_weight* con $\rho \approx 0.93$ e *Viscera_weight–Shell_weight* con $\rho \approx 0.91$).

Questa struttura è tipica dei dataset morfometrici: molte variabili descrivono la taglia complessiva dell'animale e, di conseguenza, tendono a muoversi congiuntamente. Dal punto di vista della modellazione ciò implica **multicollinearità**, ossia la presenza di predittori parzialmente ridondanti. La multicollinearità ha impatti diversi a seconda del modello:

- nei modelli lineari OLS può aumentare la varianza delle stime dei coefficienti e rendere l'interpretazione instabile, poiché i coefficienti diventano sensibili anche a piccole variazioni del campione;
- giustifica l'inclusione di modelli **regolarizzati** (Ridge, Lasso, Elastic Net), che introducono penalizzazioni sui coefficienti migliorando la stabilità delle stime e la capacità di generalizzazione;
- nelle reti neurali non impedisce la predizione, ma può aumentare ridondanza e complessità del modello, rendendo ancora più importante una corretta normalizzazione degli input e un'accurata selezione/tuning degli iperparametri.

In sintesi, questa fase ha permesso di definire formalmente la variabile target, rendere tutte le feature compatibili con la modellazione numerica, predisporre la successiva de-normalizzazione per un'interpretazione in anni e ottenere una prima evidenza descrittiva sulla distribuzione dell'età e sulle dipendenze lineari tra variabili, che motivano le scelte di preprocessing e i modelli di confronto adottati nelle sezioni successive.

2.5 Normalizzazione delle variabili esplicative

Le variabili di input del dataset *Abalone* hanno scale e unità molto diverse: misure geometriche (lunghezze) e masse (pesi) possono assumere ordini di grandezza differenti. In un modello basato su ottimizzazione iterativa (come una rete neurale), questa eterogeneità può compromettere l'efficienza dell'addestramento perché:

- i parametri associati alle variabili con scala maggiore tendono a ricevere gradienti più grandi, influenzando sproporzionalmente l'aggiornamento dei pesi;
- l'ottimizzazione può diventare più lenta o instabile, richiedendo *learning rate* più piccoli e aumentando il rischio di oscillazioni;
- il problema risulta meno “ben condizionato”, rendendo più difficile raggiungere una convergenza affidabile.

Normalizzare le *feature* consente di portare tutte le variabili su una scala comparabile, migliorando la stabilità numerica e la convergenza dell'algoritmo di apprendimento. In particolare, quando si utilizzano funzioni di attivazione come sigmoide o

tangente iperbolica, mantenere gli input in un intervallo contenuto riduce il rischio di saturazione (regioni in cui la derivata è molto piccola), condizione che può rallentare l'apprendimento. Si adotta una normalizzazione *min–max* per le variabili quantitative, coerente con la normalizzazione applicata alla variabile target. Per una generica variabile X , la trasformazione è:

$$X_{\text{scaled}} = \frac{X - X_{\min}}{X_{\max} - X_{\min}}.$$

Il risultato è una variabile scalata nell'intervallo $[0, 1]$, con due vantaggi pratici:

- controlla l'ampiezza degli input evitando valori numericamente troppo grandi;
- rende confrontabile il contributo numerico dei predittori durante l'ottimizzazione.

Analogamente a quanto discusso per la variabile risposta, i parametri della trasformazione (minimo e massimo di ciascun predittore) dovrebbero essere stimati sul training set e applicati invariati al test, per evitare che informazioni del set di valutazione influenzino indirettamente la trasformazione dei dati. Questo principio rafforza la correttezza della stima di generalizzazione.

3 Progettazione della rete neurale artificiale per regressione

L'obiettivo non è soltanto ottenere un errore basso sui dati osservati, ma costruire un modello capace di generalizzare, cioè di produrre predizioni accurate su nuovi esempi provenienti dallo stesso processo generativo. Un modello valutato sugli stessi dati utilizzati per l'addestramento tende infatti a fornire una stima ottimisticamente distorta dell'errore, soprattutto quando il modello è flessibile (come una rete neurale) e può adattarsi anche a componenti di rumore presenti nel campione. Per questo motivo il dataset viene suddiviso in due insiemi disgiunti:

- **Training set:** utilizzato per apprendere i parametri del modello (pesi della rete neurale, coefficienti nei modelli lineari e regolarizzati);
- **Test set:** utilizzato esclusivamente per la valutazione finale delle prestazioni, fornendo una stima più realistica dell'errore di generalizzazione.

La suddivisione train/test viene effettuata tramite un campionamento casuale delle osservazioni; la casualità viene controllata fissando un seme pseudo-casuale (*seed*), così da rendere l'intera procedura riproducibile.

La scelta di allocare circa il 70% delle osservazioni al training set e il 30% al test set rappresenta un compromesso comune tra:

- disponibilità di dati sufficienti per apprendere un modello stabile (*training* ampio);
- disponibilità di un campione abbastanza grande per stimare con affidabilità l'errore di generalizzazione (*test* non troppo piccolo).

In generale, percentuali come 70/30 (o 80/20) sono adeguate quando il dataset contiene un numero moderato di osservazioni e si desidera mantenere un test set sufficientemente rappresentativo. Nel caso *Abalone*, tale scelta consente di addestrare la rete neurale con un numero di esempi sufficiente e, al contempo, di valutare le prestazioni su un insieme indipendente non trascurabile. Per garantire una valutazione corretta, il test set deve rimanere isolato dal processo di sviluppo: viene utilizzato soltanto nella fase finale per il calcolo delle metriche (ad esempio MSE e MAE) e per le analisi diagnostiche (grafico osservato–predetto, residui). In questo modo, le prestazioni riportate riflettono in modo più attendibile la capacità del modello di generalizzare su dati non visti.

3.1 Inquadramento del modello: Multi-Layer Perceptron (MLP) feed-forward

Per stimare l'età dell'abalone viene adottata una rete neurale *feed-forward* di tipo *Multi-Layer Perceptron* (MLP). In una MLP l'informazione fluisce in un'unica direzione, dallo strato di input verso lo strato di output, senza cicli o retroazioni. La rete implementa una funzione parametrica $f(\cdot)$ che approssima la relazione tra predittori X ed età Age :

$$\widehat{Age} = f(X; W),$$

dove W rappresenta l'insieme dei parametri del modello (pesi e bias) appresi durante l'addestramento, mentre si definisce *feed-forward* perché l'informazione scorre solo in avanti:

$$\text{Input} \rightarrow \text{Strato nascosto} \rightarrow \text{Output}.$$

La scelta di partire da una rete con un solo strato nascosto è coerente con il fatto che, teoricamente, una MLP con uno strato nascosto è sufficientemente ampio può approssimare un'ampia classe di funzioni non lineari (universal approximation), mantenendo però una complessità controllabile e una maggiore interpretabilità del processo di tuning.

Ciascun neurone dello strato nascosto calcola una combinazione lineare degli input, seguita da una trasformazione non lineare (funzione di attivazione). Indicando con x il vettore di input, il neurone j dello strato nascosto produce:

$$h_j = \varphi\left(\sum_i w_{ij}x_i + b_j\right),$$

dove w_{ij} sono i pesi sinaptici, b_j è il bias e $\varphi(\cdot)$ è la funzione di attivazione.

Lo strato di output combina poi i segnali provenienti dallo strato nascosto per produrre la stima finale \widehat{Age} .

Il ruolo cruciale della non linearità è che, in assenza di attivazioni non lineari, la composizione di più strati lineari collasserebbe in un'unica trasformazione lineare; di conseguenza, la rete non sarebbe più espressiva di una regressione lineare.

3.2 Funzioni di attivazione nello strato nascosto: Logistic e Tanh

Nel progetto vengono considerate due funzioni di attivazione classiche.

- **Sigmoide logistica:**

$$\varphi(z) = \frac{1}{1 + e^{-z}}.$$

Produce output nell'intervallo $(0, 1)$ ed è storicamente molto utilizzata, ma può soffrire di *saturazione* per valori di z molto grandi in valore assoluto (derivate piccole e apprendimento più lento).

- **Tangente iperbolica (tanh):**

$$\varphi(z) = \tanh(z).$$

Produce output nell'intervallo $(-1, 1)$ ed è spesso preferita alla logistica perché è centrata intorno a 0, condizione che può facilitare l'ottimizzazione e rendere più bilanciati gli aggiornamenti dei pesi.

La scelta tra *logistic* e *tanh* viene quindi trattata come un iperparametro: non esiste una funzione universalmente migliore e la preferenza dipende dalla scala dei dati, dalla distribuzione degli input e dalla dinamica dell'ottimizzazione.

3.3 Output lineare: coerenza con un problema di regressione

Poiché il target è una variabile quantitativa (età), lo strato di output utilizza un'attivazione lineare. In questo modo la rete può produrre valori reali senza essere vincolata a un intervallo limitato (come accadrebbe con un'uscita sigmoide). Formalmente:

$$\widehat{\text{Age}} = \sum_j v_j h_j + b_0,$$

dove h_j sono le attivazioni dello strato nascosto, v_j i pesi verso l'output e b_0 il bias di uscita. Questa scelta è standard nelle ANN per regressione e rende coerente la minimizzazione di funzioni di errore basate su scarti numerici.

3.4 Funzione di costo: SSE e relazione con MSE

L'addestramento della rete avviene minimizzando una funzione di costo basata sugli errori tra valori osservati y e predetti \widehat{y} . Nel progetto viene adottata la Somma dei Quadrati degli Errori (SSE):

$$SSE = \sum_{k=1}^n (y_k - \widehat{y}_k)^2.$$

La SSE è strettamente collegata al Mean Squared Error (MSE):

$$MSE = \frac{1}{n} \sum_{k=1}^n (y_k - \widehat{y}_k)^2.$$

Minimizzare SSE o MSE conduce allo stesso optimum, poiché le due quantità differiscono soltanto per la costante moltiplicativa $1/n$. Nel report, l'uso del MSE risulta particolarmente utile perché consente un confronto immediato tra modelli e fornisce una misura "media" dell'errore.

3.5 Ottimizzazione iterativa e concetto di convergenza

I pesi della rete vengono stimati tramite un processo iterativo di ottimizzazione che aggiorna i parametri nella direzione opposta al gradiente della funzione di costo. In forma generica:

$$W^{(t+1)} = W^{(t)} - \eta \nabla L(W^{(t)}) ,$$

dove η è il *learning rate*, che controlla l'ampiezza degli aggiornamenti, e ∇L è il gradiente della loss rispetto ai pesi.

Un *learning rate* troppo elevato può causare instabilità e oscillazioni, mentre uno troppo piccolo può rallentare eccessivamente l'apprendimento. Per questo motivo η viene trattato come iperparametro nella fase di tuning.

La convergenza si intende come il raggiungimento di una soluzione in cui ulteriori aggiornamenti producono una riduzione dell'errore trascurabile. Operativamente, l'addestramento viene interrotto quando:

- la riduzione della funzione di costo scende sotto una soglia prefissata (criterio di arresto basato sulla convergenza, *threshold*);
- oppure viene raggiunto un numero massimo di iterazioni (*stepmax*), per evitare addestramenti indefiniti in casi non convergenti.

La combinazione tra scaling delle variabili, scelta dell'attivazione, *learning rate* e criteri di arresto determina la probabilità di convergenza e la qualità finale del modello.

3.6 Selezione degli iperparametri e strategia di tuning

A differenza dei pesi e dei bias, che vengono appresi automaticamente durante l'addestramento, gli *iperparametri* definiscono l'architettura e la dinamica di ottimizzazione della rete neurale e devono essere selezionati dall'analista. Poiché tali scelte influenzano in modo significativo convergenza e capacità di generalizzazione, è stata adottata una procedura di *grid search*, valutando combinazioni predefinite di iperparametri e selezionando la configurazione con prestazioni migliori.

Nel progetto sono stati considerati i seguenti iperparametri:

- **Numero di neuroni nello strato nascosto (H)**: controlla l'ampiezza del modello e quindi la sua capacità di rappresentare relazioni non lineari. Valori troppo piccoli possono generare *underfitting*, mentre valori troppo grandi aumentano il rischio di *overfitting* e la complessità computazionale.
- **Funzione di attivazione (φ)** nello strato nascosto: sono state confrontate le attivazioni *logistic* e *tanh*, poiché influenzano la non linearità introdotta e la stabilità dell'ottimizzazione.
- **Learning rate (η)**: determina l'ampiezza degli aggiornamenti dei pesi durante l'ottimizzazione. Un valore eccessivo può causare instabilità, mentre un valore troppo basso può rallentare la convergenza.
- **Soglia di convergenza (*threshold*)**: definisce il criterio di arresto basato sulla convergenza, interrompendo l'addestramento quando la riduzione della funzione di costo diventa trascurabile.

Per ciascuna combinazione di iperparametri la rete è stata addestrata sul training set e valutata sul test set tramite metriche di errore, calcolate riportando le predizioni alla scala originale (anni) mediante trasformazione inversa. La selezione del modello migliore è stata effettuata minimizzando il **Mean Squared Error** (MSE) sul test set; in aggiunta, è stato monitorato anche il **Mean Absolute Error** (MAE) per fornire una misura più robusta rispetto a possibili errori estremi.

Infine, per ridurre la probabilità di non convergenza e rendere più stabile la procedura di tuning, è stato imposto un numero massimo elevato di iterazioni (*stepmax*), così da permettere all'algoritmo di ottimizzazione di completare l'addestramento anche in presenza di configurazioni più difficili da ottimizzare.

4 Risultati del Tuning della Rete Neurale

Una volta definita l'architettura di base (MLP feed-forward con un singolo strato nascosto) e fissata la procedura di addestramento, è stata effettuata una fase di tuning finalizzata a individuare la configurazione di iperparametri in grado di massimizzare l'accuratezza predittiva sul set di test.

La ricerca della configurazione ottimale è stata condotta tramite una procedura di *grid search* su un insieme finito di combinazioni di iperparametri. In particolare, sono stati esplorati tre valori per il numero di neuroni nello strato nascosto:

$$H \in \{3, 5, 8\},$$

due valori per la soglia di convergenza:

$$\text{threshold} \in \{0.01, 0.05\},$$

Tre valori per il *learning rate*:

$$\eta \in \{0.001, 0.005, 0.01\},$$

e due funzioni di attivazione:

$$\{\text{logistic}, \tanh\}.$$

Complessivamente, la griglia comprende il seguente numero di combinazioni:

$$3 \times 2 \times 3 \times 2 = 36$$

Tutte le combinazioni considerate hanno prodotto una rete valida, consentendo di effettuare la selezione unicamente sulla base delle prestazioni predittive (MSE e MAE) sul test set. Per aumentare la trasparenza del processo di selezione, vengono riportate le configurazioni valutate e le relative prestazioni sul test set, ordinate rispetto al valore dell'MSE. La tabella permette di osservare non solo il modello migliore, ma anche la sensibilità delle prestazioni rispetto alle diverse scelte di iperparametri.

Tabella 1: Risultati della grid search (36 configurazioni) per il tuning della MLP.

#	Hidden	Threshold	LR	Act.	MSE	MAE
1	3	0.01	0.001	logistic	4.137544	1.439680
2	3	0.01	0.005	logistic	4.137544	1.439680
3	3	0.01	0.010	logistic	4.137544	1.439680
4	8	0.05	0.005	logistic	4.191294	1.454138
5	8	0.05	0.010	logistic	4.191294	1.454138
6	8	0.05	0.001	logistic	4.191294	1.454138
7	3	0.05	0.010	logistic	4.228399	1.450243
8	3	0.05	0.001	logistic	4.228399	1.450243
9	3	0.05	0.005	logistic	4.228399	1.450243
10	5	0.05	0.010	logistic	4.258772	1.457873
11	5	0.05	0.005	logistic	4.258772	1.457873
12	5	0.05	0.001	logistic	4.258772	1.457873
13	8	0.01	0.001	logistic	4.261958	1.455752
14	8	0.01	0.010	logistic	4.261958	1.455752
15	8	0.01	0.005	logistic	4.261958	1.455752
16	5	0.01	0.005	logistic	4.330567	1.453030
17	5	0.01	0.010	logistic	4.330567	1.453030
18	5	0.01	0.001	logistic	4.330567	1.453030
19	3	0.01	0.001	tanh	4.368758	1.485911
20	3	0.01	0.010	tanh	4.368758	1.485911
21	3	0.01	0.005	tanh	4.368758	1.485911
22	8	0.05	0.001	tanh	4.634230	1.493787
23	8	0.05	0.005	tanh	4.634230	1.493787
24	8	0.05	0.010	tanh	4.634230	1.493787
25	3	0.05	0.001	tanh	5.104743	1.520981
26	3	0.05	0.005	tanh	5.104743	1.520981
27	3	0.05	0.010	tanh	5.104743	1.520981
28	8	0.01	0.005	tanh	5.638601	1.503327
29	8	0.01	0.001	tanh	5.638601	1.503327
30	8	0.01	0.010	tanh	5.638601	1.503327
31	5	0.05	0.010	tanh	6.258844	1.549578
32	5	0.05	0.001	tanh	6.258844	1.549578
33	5	0.05	0.005	tanh	6.258844	1.549578
34	5	0.01	0.005	tanh	10.841702	1.634389
35	5	0.01	0.001	tanh	10.841702	1.634389
36	5	0.01	0.010	tanh	10.841702	1.634389

Dai risultati emerge che le configurazioni con funzione di attivazione logistica occupano sistematicamente le prime posizioni: le prime 18 configurazioni con MSE più basso utilizzano *logistic*, mentre le reti con *tanh* compaiono solo a partire da errori più elevati. In alcuni casi, la combinazione *tanh* e $H = 5$ porta a un incremento marcato dell'errore, evidenziando una minore stabilità/adequatezza di tale attivazione nel setting adottato.

Un ulteriore aspetto rilevante è che, fissati H , *threshold* e funzione di attivazione,

le metriche risultano spesso invarianti al variare del *learning rate* nel range considerato. Questo comportamento suggerisce che l'ottimizzazione converge verso soluzioni predittivamente equivalenti e che, entro gli intervalli testati, la performance finale è più influenzata dalla struttura del modello e dalla funzione di attivazione che non da piccole variazioni di η . Infine, il fatto che la migliore configurazione utilizzi $H = 3$ indica che un incremento della complessità (ad esempio passando a 5 o 8 neuroni) non produce necessariamente un miglioramento sul test set: oltre una certa soglia, maggiore flessibilità può tradursi in un beneficio marginale nullo o in una lieve perdita di generalizzazione.

4.1 Configurazione ottimale selezionata

All'interno della griglia esplorata, la rete che ha ottenuto le prestazioni migliori è risultata una MLP con:

- numero di neuroni nello strato nascosto: $H = 3$;
- funzione di attivazione: logistica (*logistic*);
- *learning rate*: $\eta = 0.001$;
- soglia di convergenza: *threshold*= 0.01;
- uscita lineare, coerente con il task di regressione.

Questa scelta rappresenta un buon compromesso tra capacità di modellare non linearità (garantita dalla presenza dello strato nascosto e dell'attivazione) e controllo della complessità (architettura poco profonda e con ampiezza contenuta), riducendo il rischio di adattamento eccessivo al campione di training.

4.2 Prestazioni sul test set (scala originale in anni)

Le prestazioni del modello selezionato sono state valutate sul test set tramite MSE e MAE calcolati dopo il riporto delle predizioni alla scala originale (anni). Questo passaggio è essenziale perché il modello è addestrato sul target normalizzato, mentre l'interpretazione e il confronto tra modelli richiedono la scala naturale del problema. La rete ottimale ottiene:

$$MSE = 4.1375, \quad MAE = 1.4397.$$

Interpretando queste metriche, il MAE indica che l'errore assoluto medio della stima è pari a circa 1.44 anni; l'MSE, essendo più sensibile a errori grandi, segnala inoltre la presenza di alcune osservazioni più difficili da predire (tipicamente individui agli estremi della distribuzione dell'età o casi con combinazioni morfometriche atipiche).

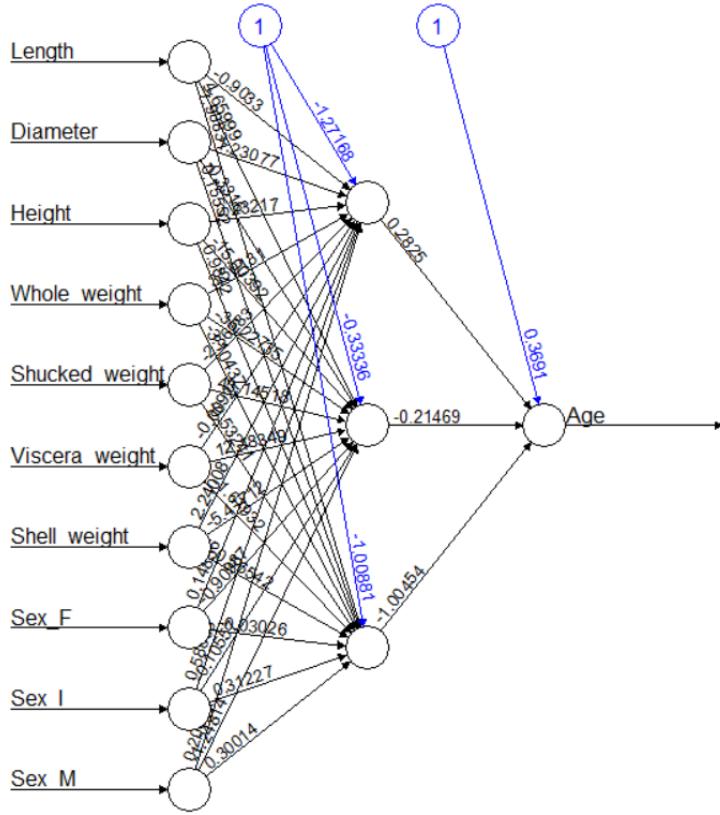


Figura 3: Rete Neurale

La Figura mostra la struttura della rete neurale selezionata dal tuning e i pesi appresi durante l’addestramento. A sinistra sono riportate le variabili di input (misure morfometriche e variabili dummy del sesso), al centro lo strato nascosto con tre neuroni, e a destra il nodo di output che produce la stima finale di *Age*. La rete, come ampiamente citato in precedenza, è di tipo *feed-forward*: l’informazione scorre solo in avanti dagli input verso l’output.

I numeri lungo le connessioni rappresentano i pesi stimati: ciascun neurone nascosto combina linearmente gli input tramite i pesi in ingresso e applica la funzione di attivazione (logistica, nel modello ottimale). Le attivazioni dei tre neuroni vengono poi combinate nello strato di output (lineare) per ottenere la previsione finale. I nodi indicati con “1” corrispondono ai bias (intercette).

4.3 Diagnostica del modello sul test set

Dopo aver quantificato la bontà predittiva tramite MSE e MAE, è utile analizzare il comportamento del modello in modo qualitativo tramite grafici diagnostici. Le metriche sintetiche, infatti, descrivono l’errore medio ma non permettono di capire se gli errori siano distribuiti in modo uniforme o se esistano pattern sistematici (ad esempio sottostima per età elevate). La diagnostica grafica consente quindi di valutare la coerenza tra osservazioni e predizioni e di identificare eventuali criticità strutturali del modello.

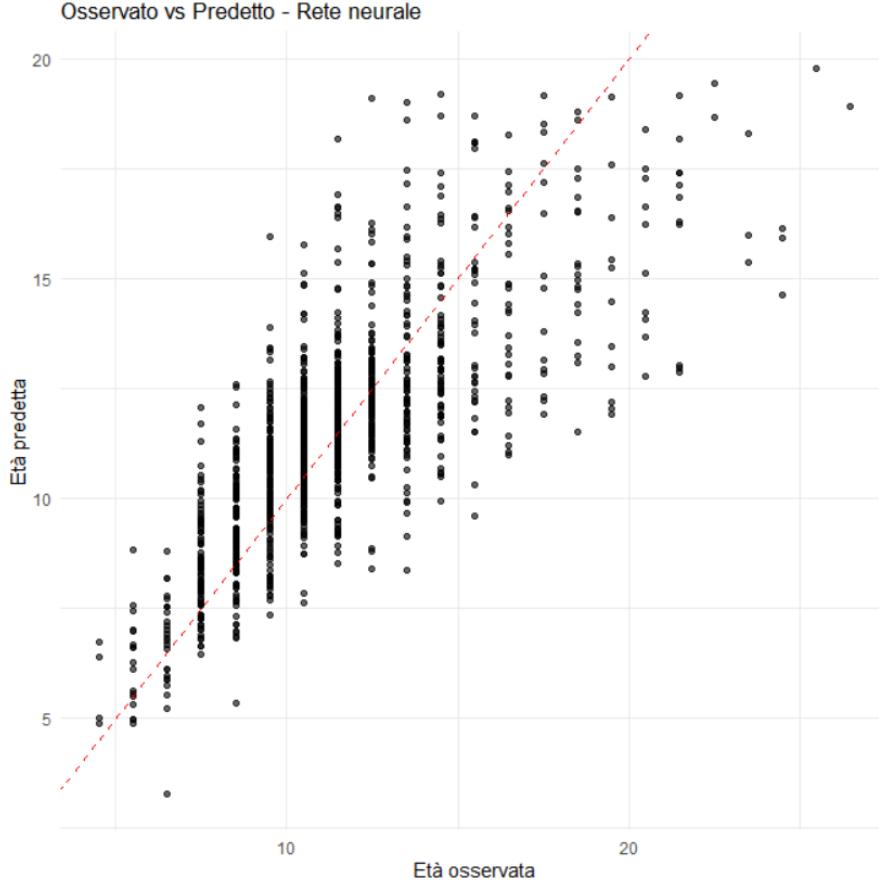


Figura 4: Osservato vs Predetto: Calibrazione e Accuratezza

Il grafico “Osservato vs Predetto” confronta per ciascuna osservazione del test set l’età reale con quella stimata. La retta tratteggiata rappresenta la bisettrice (predizione perfetta). Una buona qualità predittiva si manifesta con punti vicini alla bisettrice e senza deviazioni sistematiche (bias). Nel grafico si osserva che la rete riproduce in modo soddisfacente l’andamento complessivo: nella fascia centrale dell’età (dove la densità di osservazioni è maggiore) i punti risultano più concentrati vicino alla bisettrice. All’aumentare dell’età osservata, la dispersione tende ad aumentare: questo comportamento è coerente con lo sbilanciamento della distribuzione della target (pochi osservazioni per età elevate), che rende più complesso apprendere regole predittive robuste in coda. In pratica, il modello generalizza bene nel “cuore” della distribuzione e diventa meno preciso sulle fasce rare. Successivamente è stata implementata un’analisi dei residui (osservato – predetto), utile per verificare se gli errori siano compatibili con rumore casuale oppure se presentino strutture sistematiche. In un modello ben calibrato, i residui dovrebbero oscillare attorno a zero senza pattern evidenti.

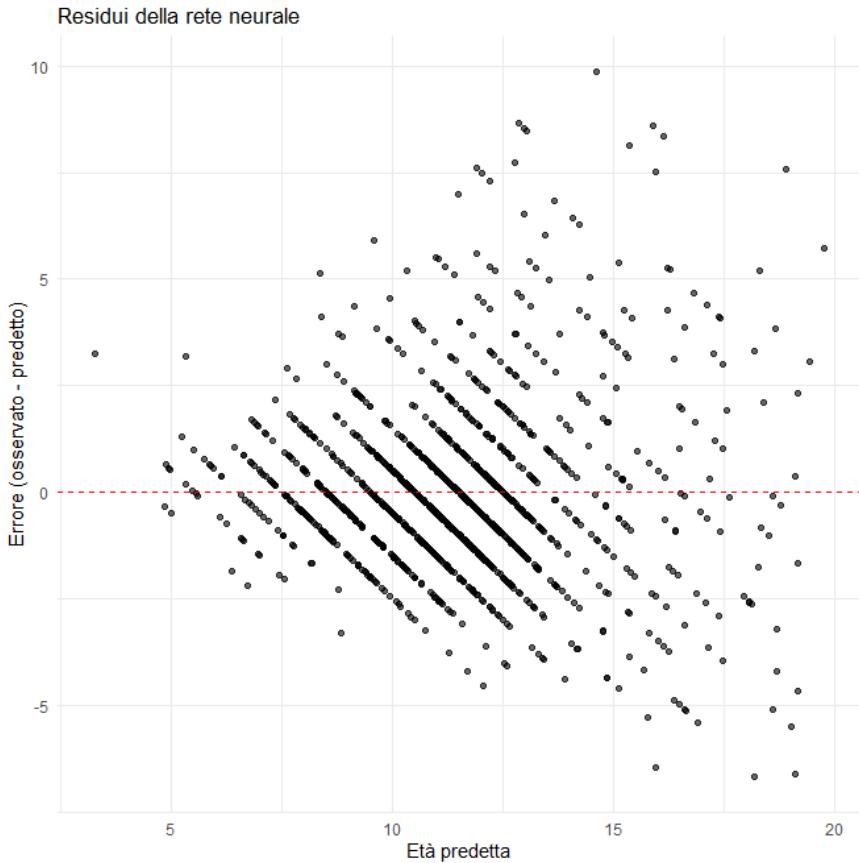


Figura 5: Enter Caption

Come possiamo vedere dal grafico, gli errori appaiono globalmente centrati intorno allo zero (assenza di un bias medio marcato). Tuttavia, la dispersione degli errori tende ad aumentare per alcune fasce di età predetta, suggerendo che l'incertezza del modello non è costante lungo tutta la scala (possibile eteroschedasticità). Questa evidenza è compatibile con la natura biologica del problema: le misure morfometriche possono essere molto informative in alcune fasi di crescita e meno in altre; inoltre, gli individui più anziani sono rari e potenzialmente più eterogenei, aumentando la difficoltà di predizione.

5 Confronto con modelli benchmark: OLS e regressioni regolarizzate

Per valutare in modo corretto il contributo di un modello non lineare come una rete neurale, è fondamentale confrontarne le prestazioni con modelli *baseline* più semplici e consolidati. Questo passaggio consente di distinguere un reale miglioramento predittivo da un guadagno solo apparente dovuto alla maggiore flessibilità del modello. In particolare:

- La regressione lineare OLS rappresenta un benchmark naturale: è interpretabile, stabile in condizioni ideali e consente di verificare quanto della relazione tra predittori e target sia spiegabile tramite una combinazione lineare;

- I modelli regolarizzati (Ridge, Lasso, Elastic Net) sono rilevanti in questo caso perché nel dataset *Abalone* esiste una forte correlazione tra molte variabili morfometriche e di peso. In presenza di multicollinearità, OLS può produrre stime dei coefficienti instabili e più sensibili al campione; la regolarizzazione introduce una penalizzazione che riduce la varianza del modello e tende a migliorare la generalizzazione.

I parametri di regolarizzazione sono selezionati tramite validazione incrociata, e le prestazioni finali vengono valutate sul test set riportando le predizioni in anni, così da essere confrontabili direttamente con rete neurale e OLS.

Tabella 2: Confronto delle prestazioni sul test set (scala originale in anni).

Modello	MSE (anni ²)	MAE (anni)
Rete neurale (MLP)	4.1375	1.4397
OLS	5.2346	1.5896
Ridge	5.5588	1.6195
Lasso	5.2286	1.5887
Elastic Net ($\alpha = 0.75$)	5.2323	1.5892

L’ordine dei modelli per accuratezza (MSE crescente) evidenzia che la rete neurale fornisce la miglior capacità predittiva sul test set, mentre i modelli lineari—anche regolarizzati—rimangono su livelli di errore più alti e molto simili tra loro.

6 Conclusioni

Nel presente elaborato è stata sviluppata una rete neurale artificiale per la stima dell’età dell’abalone a partire da misure morfometriche non invasive. L’obiettivo era verificare se un modello non lineare fosse in grado di catturare meglio la relazione tra caratteristiche fisiche ed età rispetto a modelli lineari tradizionali, fornendo uno strumento predittivo utilizzabile senza ricorrere al conteggio degli anelli di accrescimento.

I risultati mostrano che la rete neurale selezionata tramite tuning (MLP *feed-forward* con un solo strato nascosto e $H = 3$ neuroni, attivazione logistica e output lineare) ottiene le prestazioni migliori sul test set, con MSE pari a 4.1375 e MAE pari a 1.4397 anni. Il confronto con i modelli benchmark evidenzia che OLS e regressioni regolarizzate (Ridge, Lasso ed Elastic Net) presentano errori più elevati e molto simili tra loro (MSE circa 5.23–5.56). L’analisi diagnostica conferma una buona qualità predittiva soprattutto nella fascia centrale dell’età, dove la densità di osservazioni è maggiore. Per età più elevate la dispersione tra osservato e predetto tende ad aumentare, coerentemente con lo sbilanciamento della distribuzione del target (pochi esempi in coda) e con una maggiore eterogeneità biologica degli individui più anziani. I residui risultano complessivamente centrati intorno a zero (assenza di un bias medio marcato), ma mostrano una variabilità non costante lungo la scala delle predizioni, indicando che l’incertezza del modello cresce in alcune regioni del dominio.