

Artificial Neural Networks for classification of EMG data in hand myoelectric control

Viviana Tarullo

Alma Mater Studiorum · Università di Bologna
Scuola di Scienze · Corso di Laurea Magistrale in Matematica

Relatore:

Chiar.ma Prof.ssa Giovanna Citti

Correlatori:

Prof. Davide Barbieri

Ing. Emanuele Gruppioni

Dott.ssa Noemi Montobbio

25 Ottobre 2019

- I. *Protesi mioelettriche*
- II. *Metodi di classificazione*
- III. *Risultati*

Background biomedico

- Tecnica dell'elettromiografia (EMG)
- Arto fantasma

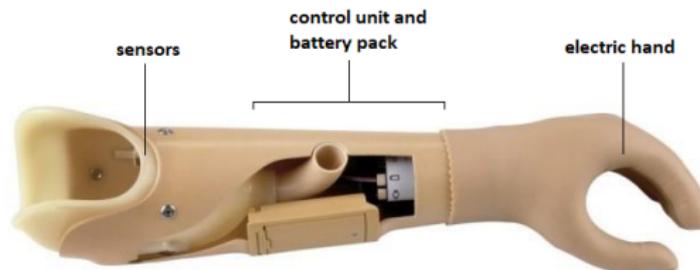


Figura: Protesi attiva mioelettrica

Metodi di classificazione

I possibili approcci alla classificazione multiclasse sono:

- combinare diversi classificatori binari per ottenere una classificazione multiclasse (approccio **one-versus-all**);
- costruire un unico classificatore che abbia come output un vettore di probabilità.

Logistic Regression

Dato un dataset $\{(x(n), t(n))\}_{n=1}^N$, dove $x(n) \in \mathbb{R}^m$ e $t(n) \in \{0, 1\}$, la probabilità che l' n -esimo campione appartenga alla classe 1 o alla classe 0 è data rispettivamente da

$$P(t(n) = 1|x(n), \theta) = \sigma(\theta^T x(n) + \theta_0) = y(n)$$

$$P(t(n) = 0|x(n), \theta) = 1 - \sigma(\theta^T x(n) + \theta_0) = 1 - y(n)$$

dove $\theta \in \mathbb{R}^m$, $\theta_0 \in \mathbb{R}$ e σ è la funzione **logistic sigmoid** definita come

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

La funzione d'errore $\xi(t, y)$, nota come **cross-entropy loss**, è

$$\xi(t, y) = - \sum_{n=1}^N [t(n) \log(y(n)) + (1 - t(n)) \log(1 - y(n))].$$

Logistic Regression

Dato un dataset $\{(x(n), t(n))\}_{n=1}^N$, dove $x(n) \in \mathbb{R}^m$ e $t(n) \in \{0, 1\}$, la probabilità che l' n -esimo campione appartenga alla classe 1 o alla classe 0 è data rispettivamente da

$$P(t(n) = 1|x(n), \theta) = \sigma(\theta^T x(n) + \theta_0) = y(n)$$

$$P(t(n) = 0|x(n), \theta) = 1 - \sigma(\theta^T x(n) + \theta_0) = 1 - y(n)$$

dove $\theta \in \mathbb{R}^m$, $\theta_0 \in \mathbb{R}$ e σ è la funzione **logistic sigmoid** definita come

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

La funzione d'errore $\xi(t, y)$, nota come **cross-entropy loss**, è

$$\xi(t, y) = - \sum_{n=1}^N [t(n) \log(y(n)) + (1 - t(n)) \log(1 - y(n))].$$

Softmax

Dato un dataset $\{(x(n), t(n))\}_{n=1}^N$, dove $x(n) \in \mathbb{R}^m$ e $t(n) \in \{1, \dots, K\}$, la probabilità che l' n -esimo campione appartenga alla classe k per ogni $k = 1, \dots, K$ è data da

$$P(t(n) = k | \mathbf{z}(n)) = \varsigma(\mathbf{z}(n))_k = y_k(n)$$

con $\mathbf{z}(n) = \Theta x(n) + \theta_0$, dove $\Theta \in \mathbb{R}^{K \times m}$, $\theta_0 \in \mathbb{R}^K$ e ς è la funzione **softmax**, definita per componenti come

$$\varsigma(\mathbf{z})_k = \frac{e^{z_k}}{\sum_{k=1}^K e^{z_k}} \quad \text{per } k = 1, \dots, K.$$

La funzione d'errore $\xi(\mathbf{t}, \mathbf{y})$, nota come **cross-entropy loss**, è

$$\xi(\mathbf{t}, \mathbf{y}) = - \sum_{n=1}^N \sum_{k=1}^K t_k(n) \log(y_k(n))$$

dove $t_k(n)$ è 1 se e solo se l' n -esimo dato appartiene alla classe k .

Softmax

Dato un dataset $\{(x(n), t(n))\}_{n=1}^N$, dove $x(n) \in \mathbb{R}^m$ e $t(n) \in \{1, \dots, K\}$, la probabilità che l' n -esimo campione appartenga alla classe k per ogni $k = 1, \dots, K$ è data da

$$P(t(n) = k | \mathbf{z}(n)) = \varsigma(\mathbf{z}(n))_k = y_k(n)$$

con $\mathbf{z}(n) = \Theta x(n) + \theta_0$, dove $\Theta \in \mathbb{R}^{K \times m}$, $\theta_0 \in \mathbb{R}^K$ e ς è la funzione **softmax**, definita per componenti come

$$\varsigma(\mathbf{z})_k = \frac{e^{z_k}}{\sum_{k=1}^K e^{z_k}} \quad \text{per } k = 1, \dots, K.$$

La funzione d'errore $\xi(\mathbf{t}, \mathbf{y})$, nota come **cross-entropy loss**, è

$$\xi(\mathbf{t}, \mathbf{y}) = - \sum_{n=1}^N \sum_{k=1}^K t_k(n) \log(y_k(n))$$

dove $t_k(n)$ è 1 se e solo se l' n -esimo dato appartiene alla classe k .

Artificial Neural Networks (ANN)

Una **rete neurale artificiale** è una struttura modulare, come quella corticale, in cui ogni strato riceve in input dallo strato precedente e fornisce in output allo strato successivo.

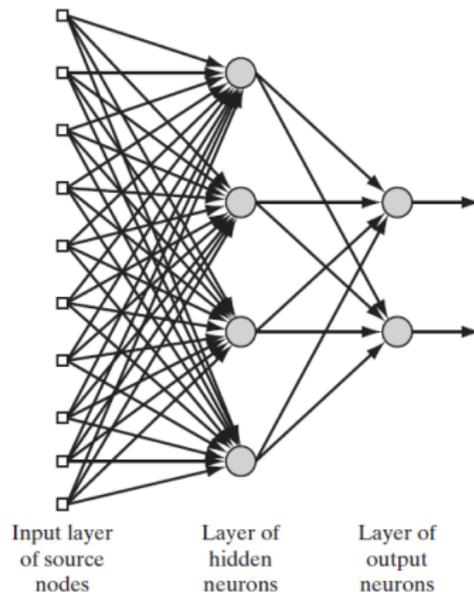


Figura: Rete neurale a due strati

Artificial Neural Networks (ANN)

Struttura dello strato i -esimo:

$$F^i(x) = \varphi(\Theta^i x + b^i)$$

Struttura della rete:

$$F = F^1 \circ F^2 \circ \dots \circ F^L$$

Artificial Neural Networks (ANN)

Incognite: Θ^i, b^i per ogni $i = 1, \dots, L$

Obiettivo: minimizzare la funzione d'errore per determinare i parametri ottimali

Setup dell'esperimento

- 20 pazienti con amputazione trans-radiale
- 6 sensori elettromiografici di superficie
- 5 gesti della mano da riprodurre
(2032 acquisizioni per sensore per ogni gesto, per 10 ripetizioni)



REST



SPHERICAL



PINCH



PLATFORM



POINT

Setup dell'esperimento

Il dataset ottenuto da un singolo paziente è

$$\{(x(n), t(n))\}_{n=1}^N$$

dove $x(n) = (x_1(n), \dots, x_6(n))$ contiene i valori del segnale registrati dai 6 sensori nell' n -esima acquisizione, $t(n) \in \{1, \dots, 5\}$ indica il gesto e $N = 2032 \cdot 5 \cdot 10 = 101600$ è il numero totale di acquisizioni.

INAIL Logistic Regression

Il modello a **Logistic Regression** implementato in INAIL è strutturato come segue:

1. Preprocessing dei dati;
2. Costruzione dei 5 classificatori NLR, dati da 5 reti neurali a uno strato con funzione di attivazione logistic sigmoid;
3. Training, test e procedura di voting.

Multilayer Feedforward ANN

La nostra proposta alternativa al classificatore NLR è un classificatore **Softmax** implementato su una **rete neurale a 2 strati**. I passi della classificazione sono i seguenti:

1. Preprocessing dei dati;
2. Costruzione del classificatore, dato da una rete neurale a 2 strati con funzioni di attivazione hyperbolic tangent nel primo strato e softmax nel secondo strato;
3. Training, test e procedura di voting.

I parametri di valutazione delle performances sono:

- *Occupazione di memoria* in termini di numero di pesi;
- *Accuratezza della classificazione*, data dal valore dell'**F1score** per ogni classe definito da

$$F1score = 2 \cdot \frac{Pr \cdot Re}{Pr + Re} \cdot 100;$$

- *Percentuale di astensione*;
- *Tempo di esecuzione*.

Risultati

Es. I)

	NLR, 1%	ANN, 1%	ANN, 80%
<i>#weights</i>	131	125 (10 hn)	125 (10 hn)
<i>F1Score (%)</i>	92.34	88.23	97.78
<i>Abs. (%)</i>	6.07	0.64	0
<i>Time</i>	33 s		283 s

Es. II)

	NLR, 1%	ANN, 1%	ANN, 80%
<i>#weights</i>	270	269 (22 hn)	269 (22 hn)
<i>F1Score (%)</i>	75.69	71.12	90.99
<i>Abs. (%)</i>	10.85	0.84	1.23
<i>Time</i>	42 s		612 s

Statistica sui 20 dataset:

	NLR, 1%	ANN, 1%	ANN, 80%
<i>#weights</i>	188	188 (15 hn)	188 (15 hn)
<i>F1Score (%)</i>	89.98	88.18	95.19
<i>Abs. (%)</i>	3.34	0.48	0.38
<i>Time</i>	36 s		320 s

Grazie per l'attenzione!