



**UNIVERSITÀ POLITECNICA DELLE MARCHE**  
**FACOLTÀ DI INGEGNERIA**

---

Corso di Laurea magistrale in **Ingegneria Informatica e dell'Automazione**

**Metodi di classificazione dell'usura e di manutenzione  
predittiva dell'organo di taglio di macchine per la  
lavorazione del tondino d'acciaio da costruzione**

**Methods for wear classification and predictive  
maintenance of cutting device of machines for processing  
structural steel bars for construction**

**Relatore:**  
Prof. Andrea Bonci

**Tesi di laurea di:**  
Alessandro Zega

**Correlatore:**  
Dott. Luca Giulioni

A.A. 2020/2021

# Indice

Introduzione	3
Capitolo 1 Macchina staffatrice COIL16	5
1.1 Organo taglio	7
Capitolo 2 Problematica: usura della superficie della lama del coltello	9
Capitolo 3 Acquisizione dei segnali	14
3.1 Gestione delle acquisizioni	15
3.2 Acquisizioni effettuate	16
3.3 Andamento tipico dei segnali acquisiti	19
Capitolo 4 Analisi dei segnali: estrapolazione delle features	24
Capitolo 5 Classificazione dell'usura del coltello	27
5.1 Tondino da 12 mm	28
5.1.1 Differenza tra filo libero e filo fisso	28
5.1.2 Differenza tra velocità al 100% e al 30% della velocità massima	31
5.1.3 Confronto fra le usure	33
5.1.4 Estrapolazione delle features	35
5.1.5 Selezione delle features con il criterio One-Way ANOVA	37
5.1.6 Realizzazione classificatore con il <i>ClassificationLearner</i>	41
5.1.6.1 Classificatore SVM	42
5.1.6.2 Matrice di confusione	45
5.1.7 Test del classificatore	47
5.1.8 Coltello Hardox e Coltello C45	52
5.1.8.1 Clustering k-means	54
5.2 Tondino da 16 mm	59
5.2.1 Estrapolazione e selezione features	61

5.2.2	Realizzazione classificatore con <i>ClassificationLearner</i>	63
5.3	Osservazioni sui risultati ottenuti	63
Capitolo 6	Applicazione online dei classificatori	65
Capitolo 7	Predizione della RUL	68
7.1	Algoritmi implementati per la predizione della RUL del coltello	73
7.1.1	Algoritmo basato sul Degradation Model	73
7.1.2	Algoritmo basato sul Similarity Model con rete neurale LSTM	80
7.2	Osservazioni sui risultati ottenuti	87
Conclusioni		89
Bibliografia		93
Bibliografia non citata		95

## Introduzione

Obiettivo del presente lavoro di tesi è lo studio e lo sviluppo di algoritmi di apprendimento orientati alla classificazione dell'usura e alla manutenzione predittiva con applicazione sulla macchina staffatrice COIL16 prodotta dalla Schnell S.p.A., azienda presso cui è stato svolto il tirocinio.

La Schnell S.p.A è azienda leader mondiale nel settore delle macchine automatiche per la lavorazione del ferro per cemento armato e per la produzione di rete elettrosaldata e dei software di gestione per i centri di sagomatura. Capace di progettare e costruire prodotti innovativi, Schnell ha realizzato nel tempo una gamma completa di macchine in grado di soddisfare tutte le esigenze della lavorazione industrializzata del ferro per cemento armato.

Per far fronte alle problematiche riscontrate in seguito al degrado sui componenti delle macchine, l'azienda ha intrapreso progetti al fine di identificare e prevedere la formazione dell'usura attraverso l'analisi di segnali provenienti dalla macchina stessa.

L'attività di tirocinio è stata incentrata, in particolare, nello studio del coltello, componente destinato al taglio del tondino d'acciaio. Per la classificazione dell'usura sono stati, quindi, utilizzati i classificatori SVM, risultati quelli con le prestazioni migliori tra tutti i modelli a disposizione sul toolbox *ClassificationLearner* di Matlab<sup>1</sup>; per la predizione della vita utile rimanente, ai fini della manutenzione predittiva, sono stati implementati due algoritmi, basati sul Degradation Model e sul Similarity Model: nel primo metodo è stata utilizzata la funzione apposita di Matlab `predictAndUpdateState` e nel secondo metodo è stata utilizzata la rete neurale LSTM.

Nel presente elaborato sono prese in considerazione le caratteristiche tecniche della macchina COIL16 con la descrizione dell'organo del taglio e delle problematiche riscontrate sui componenti della macchina causate dal degrado del coltello (capitoli 1 e 2).

Successivamente, vengono analizzati i segnali acquisiti dalla macchina con le metodologie utilizzate per il loro rilevamento e vengono elencati gli indicatori di condizione (features) estrapolati dai segnali acquisiti, al fine di ridurre la dimensione dei dati per le successive analisi (capitoli 3 e 4).

---

<sup>1</sup> Tutti gli algoritmi e i grafici presenti nell'elaborato sono stati realizzati utilizzando Matlab, con la licenza resa a disposizione degli studenti dall'Università Politecnica delle Marche.

Quindi, sono proposti modelli per la classificazione dell'usura considerando due casi relativamente alla dimensione del tondino d'acciaio (capitolo 5). I modelli ottenuti sono stati, inoltre, applicati al caso pratico, classificando real-time l'usura del coltello (capitolo 6).

Nell'ultima parte, è stata analizzata la problematica della vita utile rimanente con la realizzazione di due modelli per la stima di essa (capitolo 7), con la finalità di comprendere il funzionamento e approfondire le tecniche, piuttosto che l'applicazione al caso pratico, come effettuato per i modelli di classificazione.

# Capitolo 1

## Macchina staffatrice COIL16

La macchina oggetto di studio è una staffatrice modello COIL16, opportunamente rivisitata e sovra-sensorizzata per la raccolta di dati utili a identificare e modellare malfunzionamenti e guasti sugli organi meccanici oppure elettrici. La macchina riceve in ingresso il tondino di acciaio avvolto in una bobina, che viene prima trainato e raddrizzato all'interno della macchina e infine tagliato e piegato, al fine di realizzare le staffe desiderate [1].

Lo schema di principio della macchina è descritto nella seguente figura.

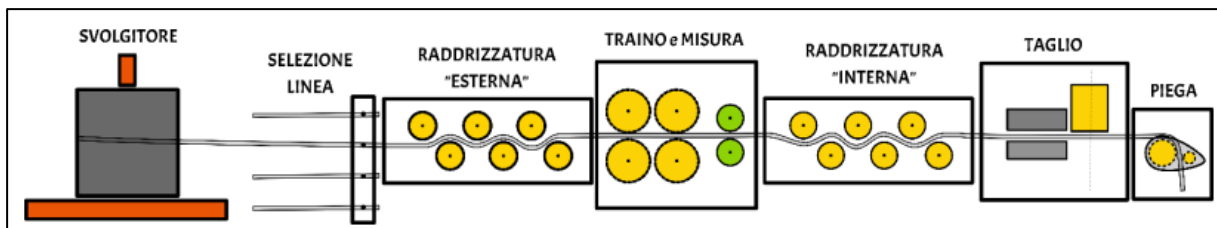


Figura 1.1 – Schema di principio della macchina COIL16 [1]

Come visibile dalla figura, si possono individuare diversi organi di lavoro:

- svolgitore: un aspo su cui è avvolta la bobina del filo (tondino) di acciaio;



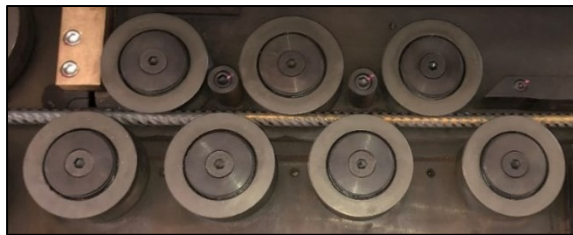
Figura 1.2 - Svolgitore

- selezione linea: una serie di canaline in ingresso alla macchina in cui si inserisce il filo di acciaio;



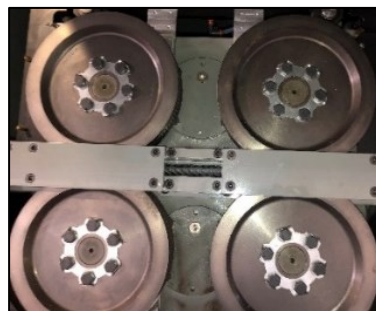
*Figura 1.3 – Selezione linea*

- raddrizzatura: una serie di ruote metalliche con lo scopo di raddrizzare il filo, in quanto esso è avvolto all'interno della bobina ed è necessario renderlo rettilineo prima del traino e delle lavorazioni finali di taglio e piega;



*Figura 1.4 – Ruote per la raddrizzatura*

- traino: un insieme di ruote metalliche di dimensioni maggiori rispetto alle precedenti con lo scopo di trainare il filo;



*Figura 1.5 – Ruote per il traino*

- taglio: un coltello collegato a una biella, che taglia il filo secondo le misure desiderate;



*Figura 1.6 – Organo del taglio*

- piega: un mandrino e un perno piegante, che piegano il tondino secondo le forme desiderate.



*Figura 1.7 – Organo della piega*

## **1.1 Organo taglio**

Il taglio è l'organo della macchina preso specificamente in considerazione durante l'attività di tirocinio e analizzato nella presente tesi.

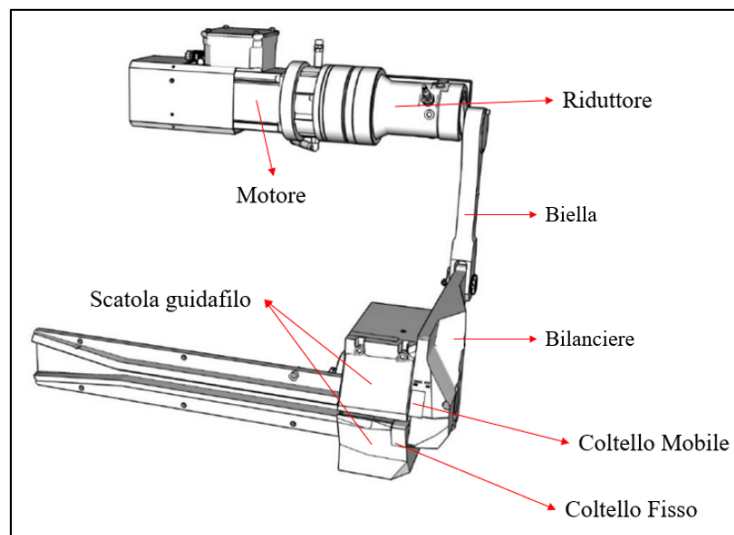
I componenti che lo costituiscono sono di seguito indicati [2].

- Il motore elettrico a corrente continua impartisce il movimento rotatorio all'intero gruppo di taglio.
- Il riduttore epicicloidale, collegato all'albero motore, riduce il numero di giri in uscita dal motore.
- Il sistema a biella trasforma il movimento rotatorio in uscita dal motore in un movimento rettilineo alternato, in particolare:
  - la “testa della biella” ha un movimento rotatorio;
  - il “piede della biella” ha un movimento rettilineo alternato.
- Il bilanciante, collegato al piede della biella attraverso un perno e una ghiera, subisce anch'esso il movimento rettilineo dettato dalla biella.



- Il coltello mobile, situato in un incavo della staffa collegata al piede della biella, è un prisma con spigoli che fungono da superfici taglienti a diretto contatto con il tondino di acciaio e ne causa il taglio.
- Il coltello fisso è un'ulteriore superficie tagliente su cui è appoggiato il tondino.
- La scatola guidafile è una scatola metallica fissata, senza possibilità di movimento, che guida il filo prima del taglio.

La scatola guidafile e il coltello fisso non sono componenti collegati alla movimentazione data dal motore, però sono fondamentali per la realizzazione del taglio: se il coltello mobile, abbassandosi e andando in contatto con il tondino, non incontra alcuna superficie che si oppone al suo movimento, non ha la possibilità di sviluppare alcun tipo di forza rendendo così impossibile il taglio.



*Figura 1.8 – Componente del taglio [1]*

La dinamica di taglio, quindi, avviene secondo i seguenti step.

- I. Viene fermato il traino del tondino d'acciaio da parte della macchina; il tondino, perciò, si trova fermo sotto il coltello mobile, oltre la scatola guidafile per la lunghezza desiderata.
- II. Il motoriduttore effettua un giro completo a velocità costante.
- III. La biella trasforma il movimento rotatorio in rettilineo alternato.
- IV. Il coltello mobile, muovendosi verso il basso, taglia il tondino d'acciaio, sfruttando le forze di reazione dovute alla presenza del coltello fisso.

## Capitolo 2

### Problematica: usura della superficie della lama del coltello

Nel corso degli anni l'azienda Schnell S.p.A. ha ricevuto segnalazioni da parte dei clienti, che notificavano rotture e malfunzionamenti dei vari componenti delle macchine [2].

Le segnalazioni dei clienti hanno fatto spesso riferimento alle seguenti problematiche.

- Rottura o formazione di cricche<sup>2</sup>, sia del bilanciante in prossimità del coltello mobile sia della scatola guidafile in prossimità del coltello fisso. Questi due componenti risultano essere molto esposti alla rottura in quanto, essendo vicini ai coltelli, subiscono le forze necessarie al taglio del tondino di acciaio.

Ad esempio, nella figura 2.1, foto inviata da un cliente, si può osservare un distacco completo di materiale di una parte del bilanciante, che necessita, quindi, di essere interamente sostituito.



Figura 2.1 – Rottura del bilanciante [2]

Un altro esempio è riportato in figura 2.2 in cui è visibile un vecchio bilanciante del magazzino componenti dell'azienda. Si può osservare la presenza di una cricca, originatasi nello spigolo nella parte inferiore della sede del coltello, che segue un percorso con punto di arrivo nel foro di fissaggio.

---

<sup>2</sup>“In metallurgia, difetto di pezzi (getti, laminati) consistente in una sottile crepatura più o meno profonda che prelude alla rottura; ha origine da difetti di formazione, lavorazione, disegno (per es., mancanza di raccordi), e si propaga a causa di sollecitazioni di fatica” Treccani.



*Figura 2.2 – Cricca sul bilanciante [2]*

In altri casi l'origine delle cricche ha interessato la scatola guidafile, come nella figura 2.3, foto inviata da un cliente, che ha osservato la formazione di due cricche.



*Figura 2.3 – Cricche sulla scatola guidafile e sul bilanciante [2]*

- Deformazione delle zone prossime alla sede del coltello. Durante il processo di taglio, per la forza di reazione tra tondino di acciaio e coltello, il coltello ruota all'interno della sua sede usando come fulcro di movimento la vite di fissaggio. A causa di tale rotazione, il coltello mobile va a spingere negli spigoli cambiando la loro geometria sempre di più all'aumentare del numero di cicli effettuati.

Negli anni passati tale situazione era considerata un fenomeno di particolare importanza, in quanto la rotazione del coltello mobile causava la rottura della parte filettata delle viti di fissaggio. Attualmente la problematica risulta essere risolta tramite l'utilizzo di viti più resistenti.

Le figure 2.4 e 2.5 mostrano due fotografie inviate da un cliente che segnala una deformazione e un cambiamento della geometria dello spigolo presente nella zona superiore e nella zona inferiore della sede del coltello.



Figura 2.4 – Deformazione nella zona superiore [2]



Figura 2.5 – Deformazione nella zona inferiore [2]

- Peggioramento della qualità del taglio del tondino. In condizioni di normale funzionamento, il profilo del tondino viene tagliato in modo netto, ma dopo diversi cicli di lavoro il profilo tende a diventare più obliquo. La differenza dei due tagli è mostrata nella seguente figura.



Figura 2.6 – Differenza tra tondino tagliato correttamente e in modo errato

Un profilo più obliquo e più appuntito, oltre a non rispettare standard di qualità del taglio, può risultare pericoloso per gli operatori stessi.

Inoltre, in alcuni casi ancora più estremi, il taglio non viene effettuato interamente e si ha il problema che più tondini successivi rimangono ancora uniti tra loro.

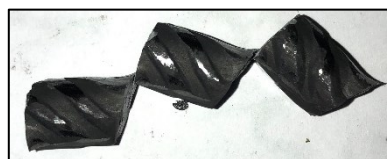


Figura 2.7 – Tagli errati

Tutte le precedenti situazioni mostrano coltelli con una superficie tagliente particolarmente deteriorata con presenza di fossette, come quelle mostrate in figura 2.8. In generale, infatti, per un coltello che ha lavorato per un elevato numero di cicli (decine di migliaia), si può osservare una consistente asportazione di materiale nelle superfici in contatto con il tondino di acciaio.



*Figura 2.8 – Coltello in presenza di degrado*

Questa correlazione ha richiamato l'attenzione da parte dei tecnici e dei progettisti dell'azienda, che, dopo diversi studi e analisi dei componenti, hanno confermato l'effettiva connessione fra i fenomeni. Infatti, l'aumento dell'usura della superficie del coltello determina sia un aumento della forza necessaria al taglio del tondino di acciaio, che si scarica quindi nei componenti prossimi ai due coltelli, sia un taglio non più netto del tondino.

Le problematiche descritte risultano dannose non solo per l'azienda a causa del danno economico, per la necessità di inviare ai clienti componenti e/o operatori che risolvano le problematiche, e del danno d'immagine, ma anche per il cliente a causa del fermo produzione inaspettato. Perciò l'obiettivo dell'azienda è quello di cercare di intervenire prima del guasto dei componenti, cercando di individuare i segnali di un deterioramento della superficie del coltello.

L'azienda, quindi, analizza in generale tutti gli organi di lavoro della macchina COIL16 al fine di individuare i possibili malfunzionamenti [1].

- Il traino, la raddrizzatura e la piega vengono studiati tramite i “banchi prova”, sistemi elettromeccanici in grado di simulare il funzionamento dell'organo in condizioni isolate e in un ambiente il più possibile controllato. Tali sistemi hanno lo scopo principale di permettere sperimentazioni evitando l'utilizzo del tondino e velocizzando il più possibile i cicli di esercizio, così da collezionare un maggior quantitativo di dati. Inoltre, permettono di creare artificialmente delle condizioni di guasto realistiche sulle quali sviluppare, testare e validare soluzioni di analisi dati. Il principale limite di tali soluzioni è l'inevitabile assenza di generalità, dovuta all'impossibilità di simulare tutte le effettive

condizioni di lavoro e, nello stesso tempo, l'assenza di disturbi provenienti dal resto dei gruppi funzionali del macchinario.

- Il taglio, che non prevede l'utilizzo di un dispositivo a sé stante, data la necessità di lavorare con il tondino per avere un comportamento realistico, viene analizzato con l'organo di taglio vero e proprio della macchina.

Tutte le analisi sono concentrate nello studio dei segnali acquisiti sia in condizioni di normale funzionamento sia in condizioni di guasto, con lo scopo di comprendere, tramite la loro differenza, in che modo si presenta l'usura.

## Capitolo 3

### Acquisizione dei segnali

Il presente lavoro si concentra nell'analisi del taglio per cercare di realizzare sia modelli di classificazione dell'usura del coltello, sia modelli per la predizione della vita utile rimanente.

Nell'organo del taglio i due elementi, chiamati precedentemente coltello mobile e coltello fisso, sono realizzati in materiale metallico Unimax, opportunamente trattato per resistere ad un elevato numero di cicli di lavoro. Tuttavia, essi sono soggetti ad usura progressiva all'aumentare del numero di tagli, richiedendo, quindi, di essere sostituiti per non rischiare di danneggiare la meccanica con le problematiche descritte in precedenza.

Per analizzare l'usura del coltello e delle sue conseguenze, sono state acquisite le misure derivanti dal driver motore:

- posizione del motore, in gradi sessagesimali [ $^{\circ}$ ];
- velocità del motore, in giri al minuto [Rpm];
- intensità della corrente del motore, in Ampere [A];
- temperatura del motore, in gradi Celsius [ $^{\circ}$ C].

Inoltre, con lo scopo di avere tutte le informazioni possibili, sono stati aggiunti appositamente i seguenti sensori:

- tre accelerometri a tre assi, posizionati sulla biella, sul riduttore e sul motore, per acquisire le vibrazioni su di essi in metro al secondo quadro [ $\text{m/s}^2$ ];
- un estensimetro elettrico, montato sulla biella, per la misura della forza, a cui essa è sottoposta, proporzionale al valore di tensione in milliVolt [mV].



Figura 3.1 – Accelerometro ed estensimetro posti sulla biella

### 3.1 Gestione delle acquisizioni

Il funzionamento dell'intera macchina è gestito dal PLC<sup>3</sup> (Programmable Logic Controller) di Beckhoff, programmato con linguaggio ST (Structured Text), utilizzando il software Twincat. Per l'acquisizione delle misure di interesse, è stata inserita nel codice del PLC la Function Block "FB\_dataLogger", che, al termine di ogni ciclo di taglio, crea un file .csv diviso in due parti:

- l'Header, contenente una serie di coppie chiave-valore, che danno informazioni riguardanti il ciclo di taglio appena terminato, come ad esempio il diametro del tondino di ferro, il numero di fili presenti e il numero di pezzi tagliati e da tagliare;
- l'Info, contenente tutti i valori acquisiti dai sensori presenti sulla macchina in ogni istante di tempo.

In particolare, la Function Block "FB\_dataLogger" esegue i seguenti step.

- I. Nel momento in cui inizia il ciclo di taglio, scrive in un buffer le coppie chiave-valore dell'header.
- II. Ad ogni istante di tempo, il cui valore di 1 ms è imposto dalla task del PLC, incrementa il buffer con i valori acquisiti dai sensori.
- III. Nel momento in cui termina il ciclo di taglio, smette di incrementare il buffer e apre un file il cui nome "FileLogger\_group\_inc" è dato dalle variabili:
  - *group*, che viene incrementata ogni 100 cicli di lavoro;
  - *inc*, che viene incrementata ad ogni ciclo di lavoro.
- IV. Copia nel file appena aperto il contenuto del buffer.
- V. Chiude il file e lo salva nella cartella di destinazione.

Ad esempio, il file "FileLogger\_0\_0.csv" così creato, aperto con Excel, ha il seguente aspetto.

---

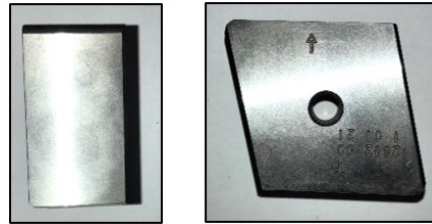
<sup>3</sup> Il PLC è un computer per l'industria, specializzato nella gestione e controllo dei processi industriali. Esso esegue un programma ed elabora i segnali digitali ed analogici provenienti da sensori e diretti agli attuatori presenti in un impianto industriale.





Come riportato nella tabella, sono state effettuate le seguenti variazioni.

- Il diametro del filo: 12 mm e 16 mm.
- L'usura della lama del coltello:
  - coltello nuovo (cioè in assenza di usura), di materiale Unimax, quello standard nelle operazioni di taglio del tondino;



*Figura 3.3 – Coltello Unimax nella condizione di assenza di usura*

- coltello di materiale Unimax con le seguenti usure realizzate manualmente con una mola, per simulare il degrado della lama e avere informazioni sul comportamento dell'organo di taglio in queste situazioni:
  - singola usura, per simulare una lama usurata nel tagliare il singolo filo fisso nella canalina,



*Figura 3.4 – Coltello Unimax con una singola usura realizzata manualmente*

- doppia usura, per simulare una lama usurata nel tagliare il doppio filo,



*Figura 3.5 – Coltello Unimax con una doppia usura realizzata manualmente*

- usura completa, per simulare una lama usurata nel tagliare il singolo filo libero di muoversi nella canalina e/o il doppio filo;

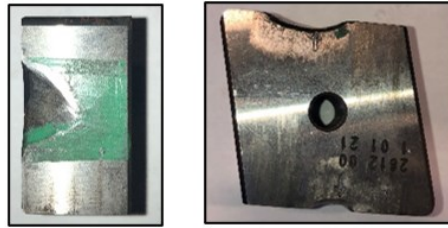


Figura 3.6 – Coltello Unimax con usura completa realizzata manualmente

- coltello di materiali non trattati, soggetti all'usura molto più velocemente rispetto all'Unimax, al fine di poter avere anche riscontri visivi dell'effetto vero e proprio dell'usura sulla lama, oltre che di studiarne i comportamenti:

- C45, materiale molto morbido,



Figura 3.7 – Coltello C45 con usura in seguito a 3000 tagli

- Hardox, materiale più duro del C45.



Figura 3.8 – Coltello Hardox con usura in seguito a 18000 tagli

- Il numero di fili tagliati contemporaneamente: nel caso di fili con diametro da 12 mm si possono tagliare anche due fili nello stesso istante, ma tale condizione non è stata approfondita.

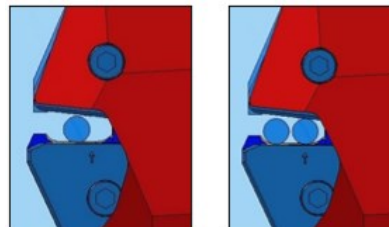


Figura 3.9 – Confronto tra singolo e doppio tondino

- La posizione del filo nella canalina: filo “fisso”, cioè tenuto fermo in una posizione utilizzando una guida in modo da incontrare la lama sempre nello stesso punto, e filo

“libero” di muoversi, che incontra il coltello in diversi punti della lama e in diversi istanti di tempo; infatti, quando il tondino si trova nella parte più interna viene tagliato qualche istante prima rispetto a quando si trova nella parte esterna.

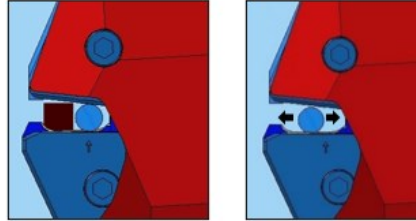


Figura 3.10 – Confronto tra tondino fisso e libero di muoversi nella canalina

Il filo libero è il caso più comune a livello pratico, in quanto la maggior parte degli operatori, che utilizzano la macchina, lasciano il tondino in tale condizione.

- La velocità del motore rispetto alla velocità massima: 30% e 100%. Il secondo è il caso più realistico a livello pratico, in quanto gli operatori tagliano il tondino alla velocità massima per effettuare più operazioni nel minor tempo possibile. Esiste, tuttavia, l'eventualità, soprattutto per la realizzazione di particolari semilavorati, che la velocità sia ridotta per evitare problemi nell'espulsione. In ogni caso, la velocità è scelta liberamente dall'utente.

### 3.3 Andamento tipico dei segnali acquisiti

Come descritto in precedenza, al termine di ogni ciclo di taglio i segnali vengono salvati in file .csv. Per essere analizzati, in Matlab vengono convertiti in file .mat utilizzando la funzione `readmatrix`, che prende in considerazione solamente la parte “Info” con i dati forniti dai sensori, tralasciando l’“Header”.

Gli andamenti generali dei segnali acquisiti<sup>4</sup> sono i seguenti.

---

<sup>4</sup> Tutti i grafici in questo capitolo fanno riferimento alle acquisizioni relative al caso di un singolo filo di diametro di 12 mm, libero di muoversi nella canalina, con lama del coltello nuova (quindi in assenza di usura) e con velocità del motore al 100%.

- Intensità di corrente del motore.

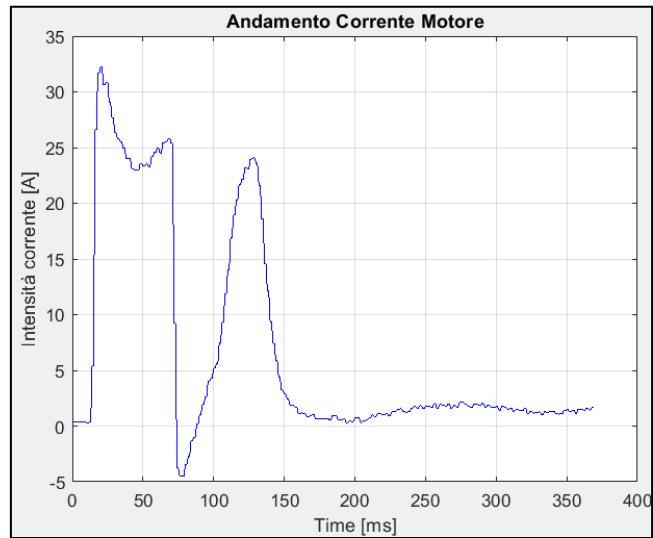


Figura 3.11 – Andamento della corrente del motore

L'andamento della corrente presenta un aumento iniziale dovuto alla messa in moto del motore per raggiungere la velocità di riferimento e un'ulteriore variazione intorno a 80 ms, istante in cui il coltello incontra il tondino per il taglio. Proprio questa seconda variazione verrà presa in considerazione per l'individuazione di un'eventuale usura della lama in quanto essa cambia da caso a caso.

- Posizione del motore.

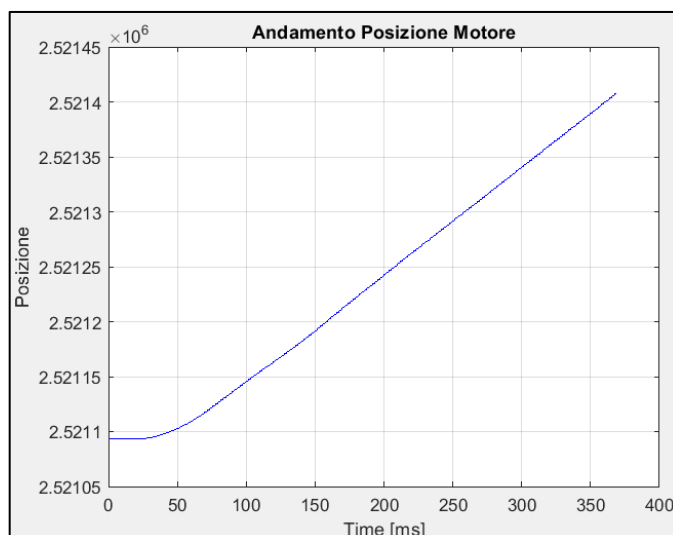


Figura 3.12 – Andamento della posizione del motore

Il segnale assume apparentemente dei valori elevati per la posizione del motore. Questo è dovuto al fatto che ad ogni ciclo di taglio la variabile continua ad aumentare senza mai

resettarsi. Per effettuare quindi le giuste considerazioni a riguardo, al valore misurato viene sottratto il valore iniziale del ciclo di lavoro, ottenendo il seguente segnale.

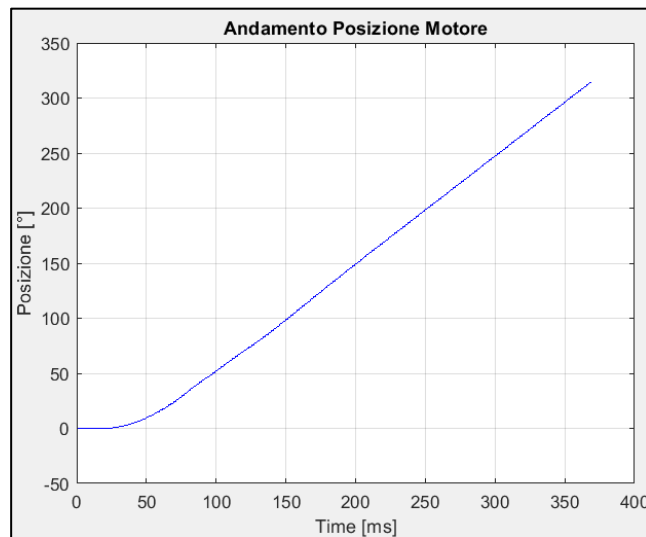


Figura 3.13 – Andamento della posizione del motore in gradi

Il grafico mostra che la posizione del motore varia da 0° fino oltre 300° (ovvero il motore effettua quasi un giro completo), non dando tuttavia informazioni, tramite variazioni, del momento in cui il coltello taglia il tondino e di un'eventuale usura.

- Temperatura del motore.

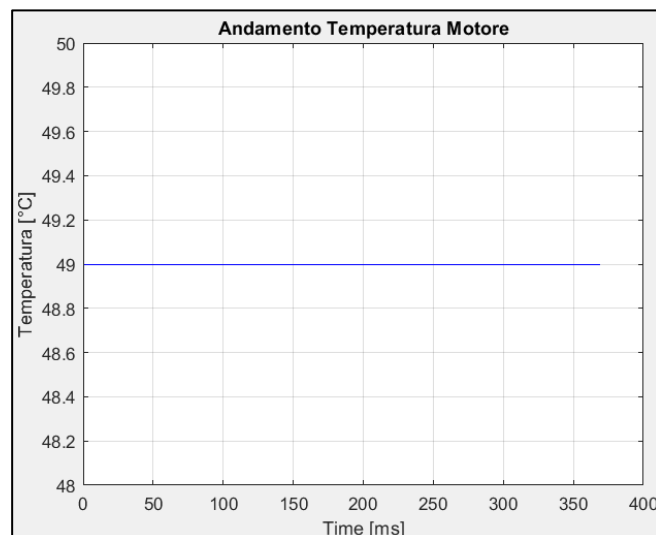


Figura 3.14 – Andamento della temperatura del motore

La temperatura, in questo caso di 49 °C, è un segnale che rimane costante nell'intero ciclo di taglio, non dando nessuna informazione utile né del momento in cui il coltello incontra il tondino né di un'eventuale usura della lama. Inoltre, è un segnale che aumenta lentamente nel corso di più acquisizioni e questa variazione dipende più dal numero di

cicli di lavoro a cui è sottoposto il motore che dall'usura. Tuttavia, come verrà spiegato successivamente, il valore della temperatura può influire sulle altre variabili, rendendo necessarie alcune considerazioni.

- Velocità del motore.

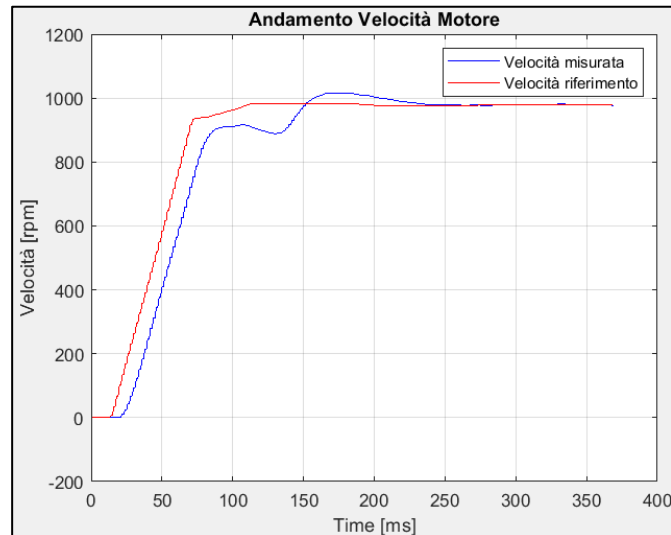


Figura 3.15 – Andamento della velocità del motore

La velocità del motore ha un andamento crescente al fine di raggiungere la velocità di riferimento. Intorno a 80 ms c'è una variazione negativa dovuta al taglio, tuttavia questo andamento risulta uguale per tutte le usure del coltello. Di conseguenza la velocità non verrà usata come segnale significativo per l'individuazione del degrado della lama.

- Forza rilevata rilevata dall'estensimetro.

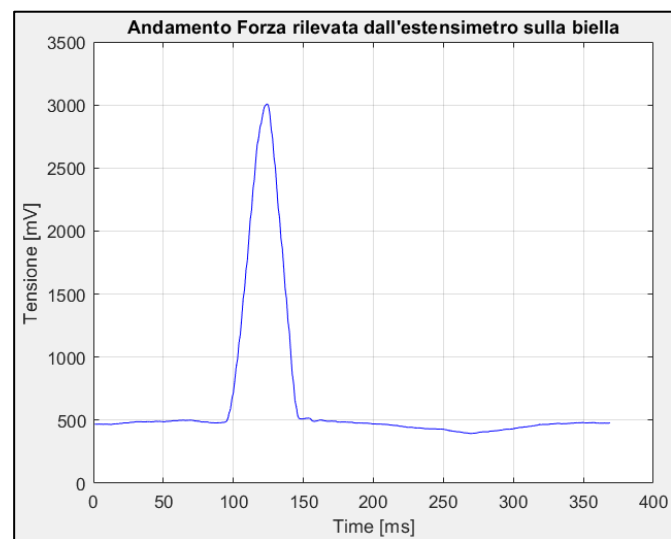


Figura 3.16 – Andamento della forza (proporzionale alla tensione) rilevato dall'estensimetro sulla biella

Il segnale proveniente dall'estensimetro posto sulla biella mostra un picco nell'istante in cui il coltello incontra il filo. Proprio per questo andamento, in cui c'è una sola una variazione significativa dovuta al taglio del tondino, i segnali forniti dall'estensimetro risultano essere i più importanti nell'individuazione dell'usura.

- Vibrazioni rilevate dagli accelerometri.

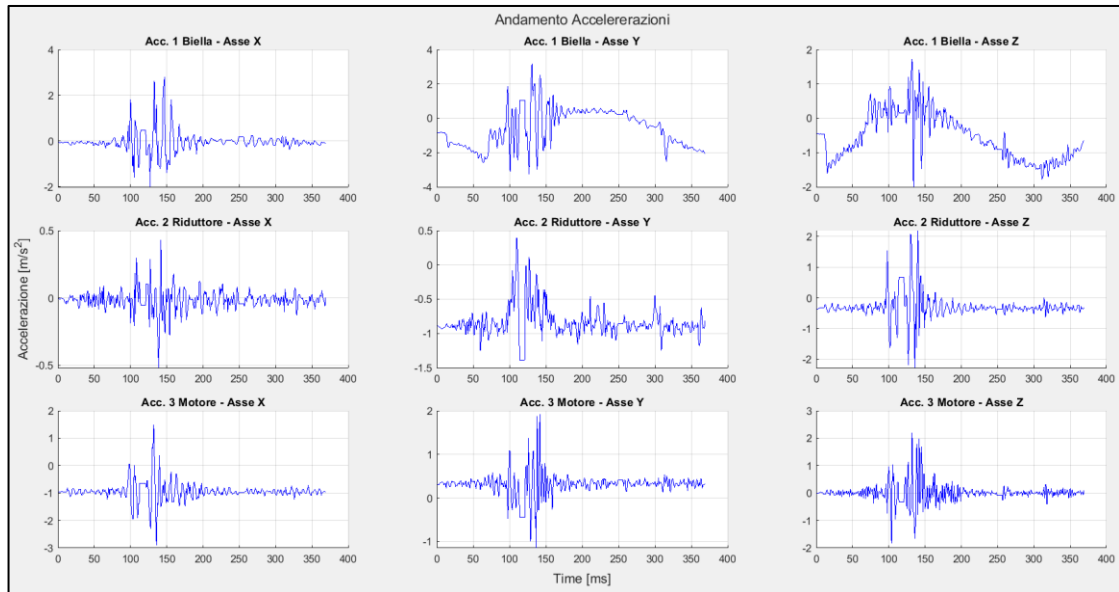


Figura 3.17 – Andamenti delle vibrazioni rilevata dai tre accelerometri

I segnali in figura fanno riferimento alle vibrazioni rilevate dai tre accelerometri posizionati rispettivamente sulla biella, sul riduttore e sul motore. Intorno a 80 ms, le vibrazioni aumentano in ampiezza dando, quindi, l'informazione sull'istante in cui la lama inizia a tagliare il tondino. Inoltre, si può notare come le accelerazioni lungo gli assi Y e Z dell'accelerometro sulla biella hanno andamenti diversi rispetto alle altre a causa del fatto che il sensore si muove fisicamente oltre che vibrare. Tuttavia, come verrà mostrato successivamente, questi segnali avranno lo stesso andamento per tutte le tipologie di usura del coltello, non risultando quindi significativi per l'individuazione e la classificazione della stessa.



## Capitolo 4

### Analisi dei segnali: estrapolazione delle features

Come visto nel paragrafo 3.3, i segnali acquisiti sono tempo-varianti all'interno di ogni acquisizione e ciclici, in quanto il funzionamento del sistema è discontinuo e ripetitivo: le acquisizioni di ogni ciclo hanno andamenti dei segnali con profilo simile. L'eventuale presenza dell'usura si manifesta come un'alterazione del profilo del singolo ciclo, visibile studiando l'evoluzione temporale dei cicli. Al fine di caratterizzare il degrado del coltello associato a questi segnali, la tecnica di analisi impiegata è stata l'extrapolazione delle features: esse sono caratteristiche individuali e indipendenti fra loro, che con un semplice valore sintetizzano il contenuto informativo dell'intera acquisizione.

Le features vengono, quindi, utilizzate come indicatori di condizione, che aiutano a distinguere il funzionamento "sano" da quello "difettoso", al fine di ottenere algoritmi per la classificazione di guasti/usure e per la predizione della vita utile residua. Infatti, con i dati grezzi acquisiti dalla macchina è difficile individuare la presenza di un guasto, ma usando questi indicatori si è in grado di discriminare operazioni sane da quelle difettose [3].



Figura 4.1 – Estrapolazione features [3]

Le features possono essere estratte dal dataset grezzo iniziale in tre modi:

- nel dominio del tempo, sono semplici caratteristiche statistiche dei segnali temporali come la media, la deviazione standard, il root-mean-square (errore quadratico medio);
- nel dominio della frequenza, sono utilizzate soprattutto per dati riguardanti le vibrazioni, al fine di trovare ad esempio l'ampiezza dei picchi e le frequenze in cui si manifestano i faults;
- nel dominio del tempo-frequenza, sono utilizzate quando si vuole eseguire un'analisi spettrale in tempo-frequenza, cioè caratterizzare i cambiamenti nello spettro di un segnale nel tempo.

Nel caso in esame, l'obiettivo è trovare le features con il contenuto informativo più significativo per l'individuazione dell'usura del coltello; quindi, sono state estrapolate queste caratteristiche discriminanti per ogni segnale di ogni acquisizione e poi sono state confrontate tra loro al fine di trovare quelle che variano maggiormente al variare dell'usura.

Le features sono state calcolate nel dominio del tempo e sono le seguenti [4].

- Statistiche di base:

- Valor medio: rapporto fra la somma tutti i valori  $x_i$  assunti dal segnale in un'intera acquisizione e il numero  $N$  complessivo dei dati acquisiti in un'intera acquisizione.

$$\bar{x} = \frac{1}{N} \sum_{i=1}^N x_i$$

- Root-Mean-Square (Rms): radice quadrata della media dei quadrati dei valori  $x_i$  assunti dal segnale in un'intera acquisizione.

$$x_{RMS} = \sqrt{\frac{1}{N} \sum_{i=1}^N |x_i|^2}$$

- Deviazione Standard: dispersione dei dati  $x_i$  intorno ad un indice di posizione, in questo caso la media  $\bar{x}$ .

$$\sigma = \sqrt{\frac{1}{N-1} \sum_{i=1}^N |x_i - \bar{x}|^2}$$

- Shape Factor: rapporto fra l'RMS e la media dei valori assoluti del segnale.

$$x_{SF} = \frac{x_{RMS}}{\frac{1}{N} \sum_{i=1}^N |x_i|}$$

- Statistiche di ordine superiore:

- Kurtosis: rapporto fra la media delle differenze alla quarta tra dati  $x_i$  e la media  $\bar{x}$  e la deviazione standard alla quarta.

$$x_{KURT} = \frac{\frac{1}{N} \sum_{i=1}^N (x_i - \bar{x})^4}{\sigma^4}$$

- Skewness: rapporto fra la media delle differenze alla terza tra dati  $x_i$  e la media  $\bar{x}$  e la deviazione standard alla terza.

$$x_{SKEW} = \frac{\frac{1}{N} \sum_{i=1}^N (x_i - \bar{x})^3}{\sigma^3}$$

- Metriche impulsive, relative ai picchi del segnale:

- Valore di Picco: valore massimo assunto dal segnale in un'intera acquisizione.

$$x_{PEAK} = \max(x_1, \dots, x_N)$$

- Crest Factor: rapporto fra il valore di picco e l'RMS.

$$x_{CREST} = \frac{|x_{PEAK}|}{x_{RMS}}$$

- Clearance Factor: rapporto fra il valore di picco e il quadrato della media delle radici dei valori assoluti assunti dal segnale in un'intera acquisizione.

$$x_{CLEARANCE} = \frac{|x_{PEAK}|}{\left(\frac{1}{N} \sum_{i=1}^N \sqrt{|x_i|}\right)^2}$$

- Impulse Factor: rapporto fra il valore di picco e la media dei valori assoluti del segnale.

$$x_{IF} = \frac{|x_{PEAK}|}{\frac{1}{N} \sum_{i=1}^N |x_i|}$$

Inoltre, è stata calcolata anche l'area sottesa dalla curva del segnale utilizzando il metodo dei trapezi, che scompone l'area in trapezi con aree più semplici da calcolare, con la funzione trapz. Per un'integrazione con N punti equidistanti, l'approssimazione è:

$$\int_a^b f(x) dx \approx \frac{b-a}{2N} \sum_{i=1}^N (f(x_i) + f(x_{i+1})) \quad \text{con } a = x_1 < \dots < x_N = b$$

## Capitolo 5

### Classificazione dell'usura del coltello

Da qualche decennio la fault detection e la fault classification sono attive aree di ricerca per le aziende al fine di assicurare la sicurezza e la qualità dei prodotti: la prima determina la presenza di faults nei processi, sensori e attuatori, mentre la seconda determina il tipo di fault individuato [5].

Negli anni sono stati proposti diversi metodi, che possono essere classificati in tre categorie:

- metodi basati su modelli quantitativi;
- metodi basati su modelli qualitativi;
- metodi basati su dati.

A causa della complessità dei processi moderni, realizzare modelli matematici risulta più impegnativo; quindi, la terza tecnica, che si basa su dati provenienti dai processi, sta avendo sempre più attenzione da parte delle aziende. Il passo fondamentale della tecnica è l'estrapolazione delle features, che permette di trasformare dataset di grandi dimensioni, in dati con più contenuto informativo e di dimensione minore. Come spiegato in precedenza, questo è stato il metodo utilizzato nel presente lavoro di tesi con lo scopo di ottenere classificatori che permettano di classificare il coltello come nuovo, semi-usurato e usurato.



*Figura 5.1 – Esempi di coltello nuovo, coltello semi-usurato, coltello usurato*

Quindi, prima sono state estrapolate le features dai segnali acquisiti, successivamente sono state selezionate solamente le features più significative della presenza dell'usura (ovvero quelle che cambiavano maggiormente al variare dell'usura sul coltello) e infine è stato realizzato il modello del classificatore, seguendo le tre fasi riportate di seguito [6].

1. Addestramento: il modello viene addestrato con il training-set, ovvero un set di dati costituito dai predittori (le features) con associato il target (le tre classi di usura che il

classificatore deve individuare) per un certo numero di osservazioni (le acquisizioni). Il classificatore imparerà, quindi, le relazioni tra predittori e target.

2. Validazione: il modello addestrato viene convalidato sul validation-set, ovvero un set di dati costituito dalle features scelte come predittori, di cui è noto il target. Il classificatore, quindi, predice il tipo di usura a partire dalle features in ingresso ed è sufficiente confrontare i target predetti con i target reali per analizzare le performance del modello. In questa fase, sulla base degli errori effettuati, il classificatore regolerà i propri parametri di predizione.
3. Test: il classificatore viene testato sul test-set, un set di dati che il modello non ha mai visto. Questo dataset, che a volte viene anche omesso, serve quasi esclusivamente per valutare e visualizzare le prestazioni e il funzionamento del modello.

Nel presente capitolo vengono analizzati gli step che hanno portato alla realizzazione dei modelli per la classificazione dell'usura del coltello nel caso del tondino di 12 mm e di quello di 16 mm di diametro.

## **5.1 Tondino da 12 mm**

Per la prima serie di acquisizioni sono state considerate tutte le possibili condizioni di lavoro in cui poteva venire tagliato il singolo tondino da 12 mm, ovvero:

- coltello con lama nuova e usurata manualmente (singola e doppia usura);
- un solo tondino tagliato alla volta;
- filo fisso e libero di muoversi nella canalina della scatola guidafile;
- velocità al 30% e al 100% rispetto la velocità massima del motore.

### **5.1.1 Differenza tra filo libero e filo fisso**

Inizialmente è stata studiata la differenza tra filo fisso nella canalina e filo libero di muoversi, confrontandoli per le diverse usure. Di seguito vengono riportati solamente i grafici relativi all'intensità di corrente del motore e alla forza rilevata dall'estensimetro: le considerazioni che vengono fatte si possono estendere anche ai segnali provenienti dai tre accelerometri, mentre per quanto riguarda posizione e velocità del motore non si notano differenze tra i due casi.

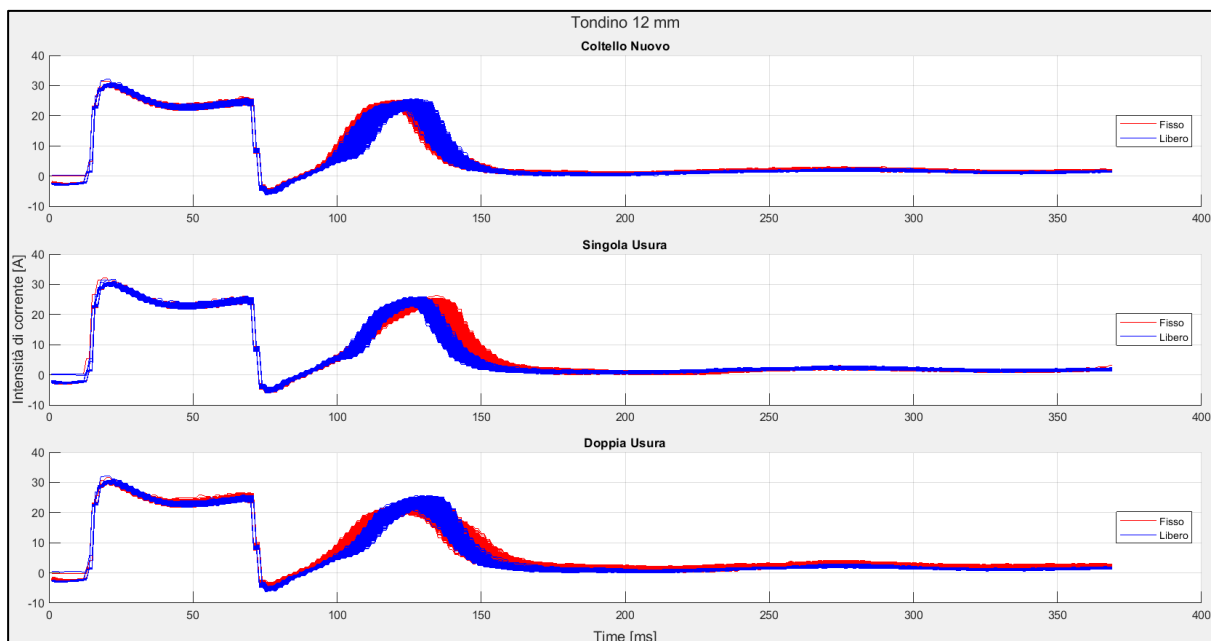


Figura 5.2 – Andamento dell'intensità di corrente del motore per le diverse usure, confrontando il filo fisso e il filo libero

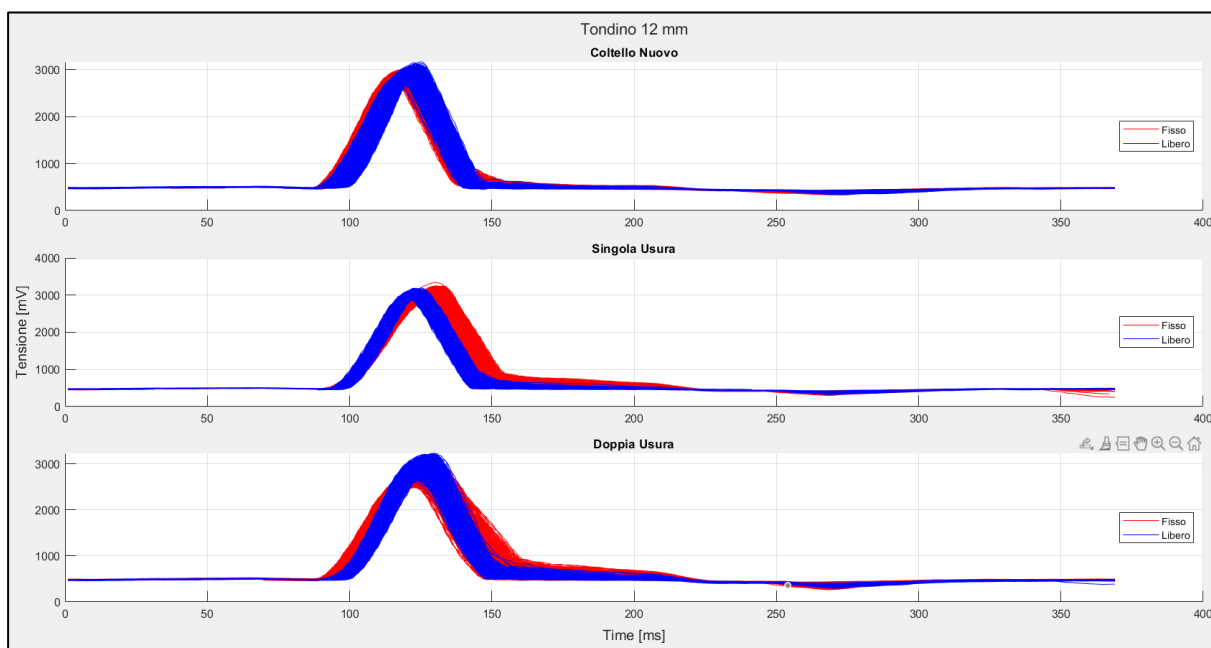
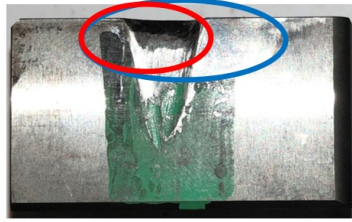


Figura 5.3 – Andamento della forza (proporzionale alla tensione) rilevata dall'estensimetro per le diverse usure, confrontando il filo fisso e il filo libero

Sia per la corrente del motore sia per la forza sulla biella possono essere fatte le seguenti osservazioni.

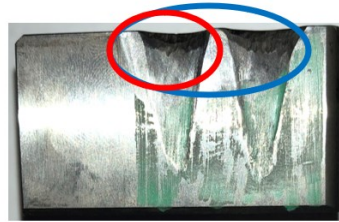
- In generale quando il filo viene fissato nella parte più interna incontra il coltello in anticipo rispetto a quando è libero di muoversi. Questo si può notare poco prima di 100 ms, in cui la curva in rosso (filo fisso) varia prima rispetto alla curva in blu (filo libero).

- Nel caso del coltello nuovo, i due andamenti sono sovrapposti in quanto tutta la lama del coltello taglia il filo allo stesso modo. Non si notano, cioè, delle variazioni dettate dall'usura.
- Nel caso della singola usura, il filo fisso mostra una variazione più ampia, in quanto subisce sempre il taglio dalla parte di lama usurata, mentre il filo libero, muovendosi nel corso dei tagli, subisce anche il taglio dalla lama nuova.



*Figura 5.4 - Zone della lama del coltello a singola usura: in rosso la parte di lama che taglia il filo fisso, mentre in blu la parte di lama che taglia il filo libero.*

- Nel caso della doppia usura, il filo fisso mostra una variazione più ampia in quanto subisce sempre il taglio della lama usurata, mentre il filo libero spostandosi nel corso dei tagli subisce il taglio della lama nuova.



*Figura 5.5 - Zone della lama del coltello a doppia usura: in rosso la parte di lama che taglia il filo fisso, mentre in blu la parte di lama che taglia filo libero.*

A conferma di quanto detto, di seguito vengono mostrati i grafici relativi alla forza rilevata dall'estensimetro tra 50 ms e 200 ms (intervallo di tempo in cui avviene il taglio), differenziando i casi tra filo libero e fisso.

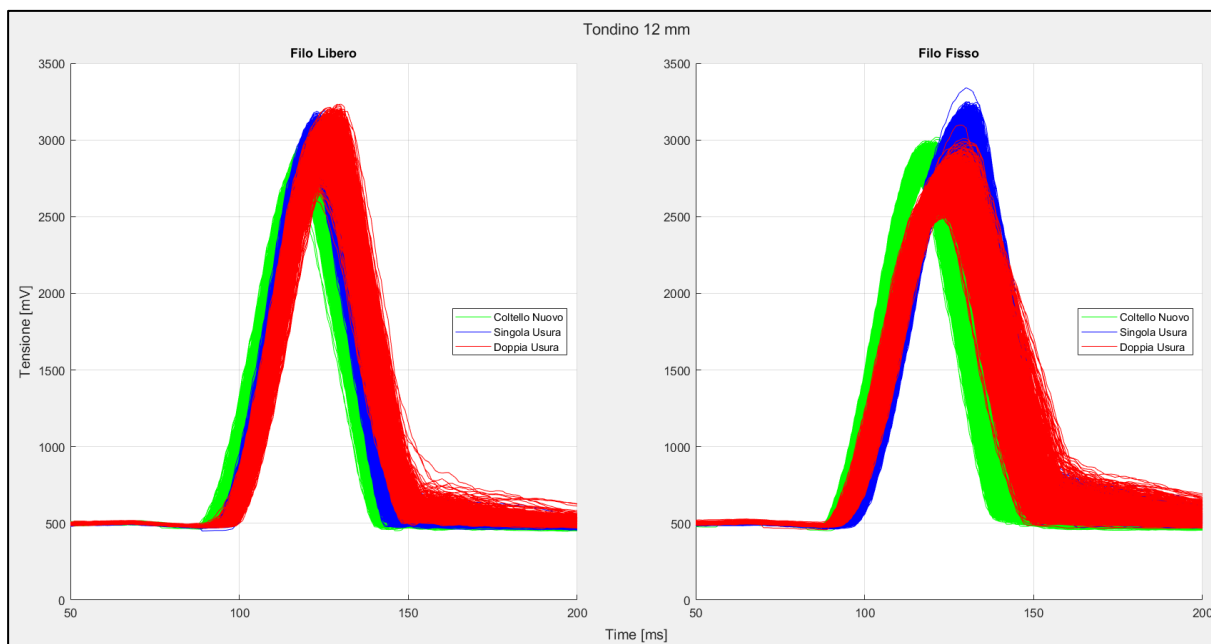


Figura 5.6 – Confronto della forza (proporzionale alla tensione) rilevata dall'estensimetro tra filo libero e filo fisso per le diverse usure

Nel caso di filo libero, i grafici delle diverse usure non si differenziano di molto, poiché il tondino viene tagliato dalla lama nuova anche nei coltelli usurati. Nel caso del filo fisso, invece, il tondino subisce sempre il taglio con lama usurata, poiché è fissato in quella posizione, e questo porta alla netta differenza tra le diverse usure. Inoltre, in questa situazione la differenza tra singola e doppia usura è dovuta al fatto che le due usure sono state realizzate manualmente con l'utilizzo di una mola e non tagliano quindi allo stesso modo.

La condizione di filo fisso distingue più facilmente la presenza di usura nel coltello, ma è meno comune nelle condizioni di lavoro; pertanto, questo caso non verrà più preso in considerazione nelle diverse prove effettuate e, da qui in poi, si farà sempre riferimento al caso con tondino libero di muoversi nella canalina.

### 5.1.2 Differenza tra velocità al 100% e al 30% della velocità massima

Successivamente è stata presa in considerazione la differenza nell'uso del motore al 100% della velocità massima o al 30%. La prima conseguenza è la diversa durata della singola acquisizione: al 100% un ciclo di taglio dura 369 ms, mentre al 30%, essendo il motore più lento, un ciclo dura 1225 ms.

Inoltre, andando ad analizzare i grafici delle velocità del motore, come ci si aspettava, la velocità di regime al 30% è circa un terzo inferiore rispetto al caso al 100%.



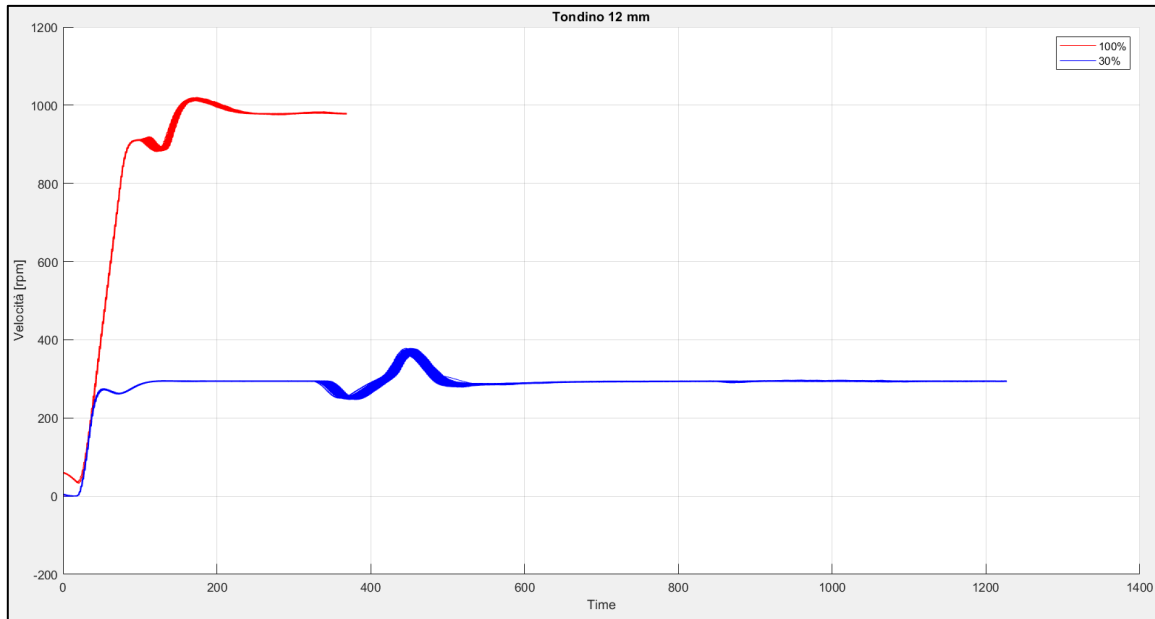


Figura 5.7 – Confronto tra la velocità al 100% e al 30%

Confrontando i grafici della forza in figura 5.8, al 30% l'effetto dell'usura è maggiore rispetto al 100%; infatti, andando più lentamente, si hanno più campioni che lo individuano.

In realtà gli operatori raramente utilizzano le macchine con velocità inferiori a quella massima al fine di ottimizzare i tempi di produzione; pertanto, sebbene con una velocità del 30% si distingue molto più facilmente la presenza dell'usura, anche questo caso non è stato più preso in considerazione nelle diverse prove effettuate. Da qui in poi si farà sempre riferimento al caso con una velocità del 100%.

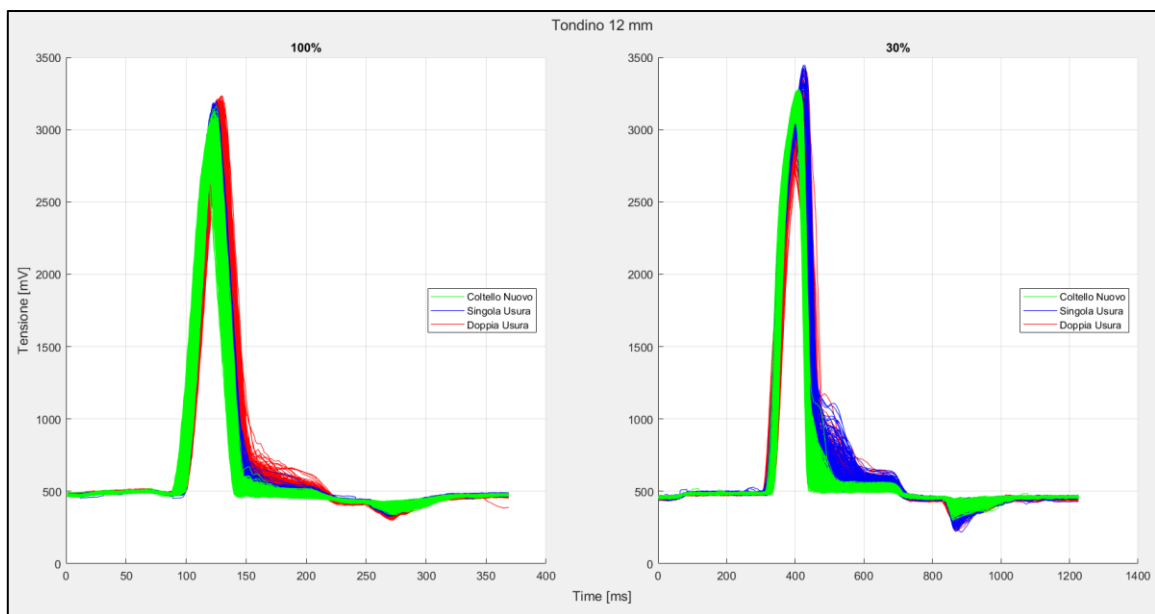


Figura 5.8 – Confronto della forza (proporzionale alla tensione) rilevata dall'estensimetro tra velocità al 100% e al 30% per le diverse usure

### 5.1.3 Confronto fra le usure

A questo punto, individuate le reali condizioni di lavoro (filo libero e velocità al 100%), si è posta l'attenzione esclusivamente sulla differenza dei comportamenti delle variabili al variare dell'usura della lama del coltello.

Per simulare l'usura di un filo che non rimane fisso in una posizione, è stata realizzata manualmente un'usura per tutta la superficie del tagliente come mostrato in figura 5.9, a differenza della singola e doppia usura relative a fili tagliati sempre nella stessa posizione.



Figura 5.9 – Usura completa realizzata manualmente

Di seguito vengono mostrati gli andamenti dei segnali acquisiti per le diverse usure.

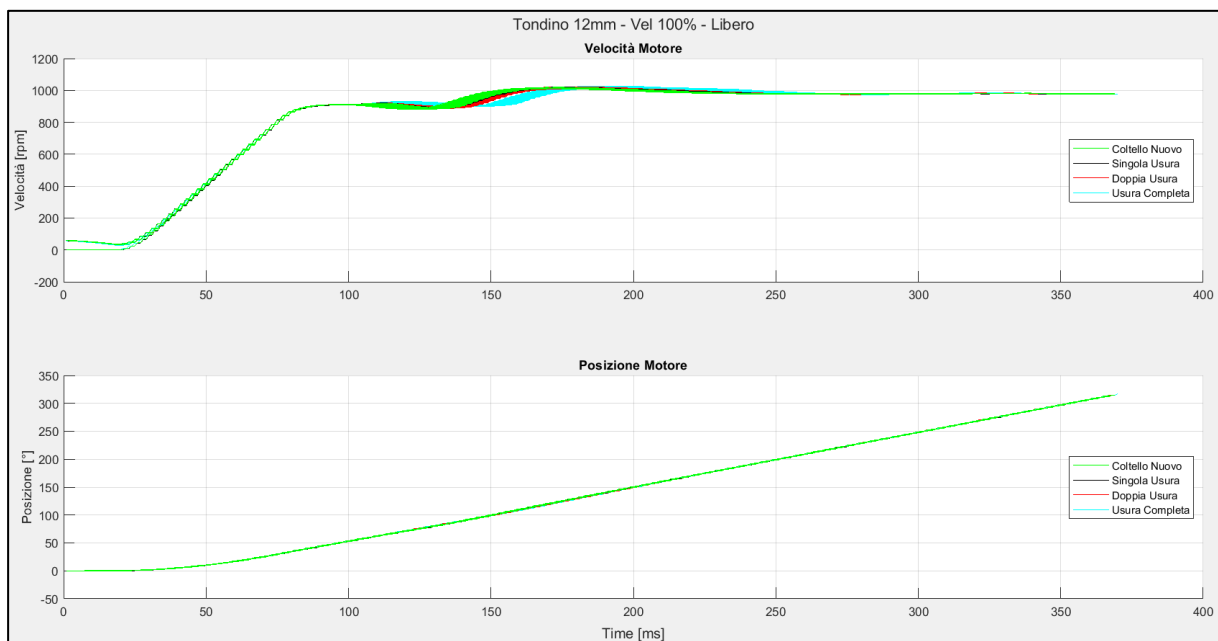


Figura 5.10 – Confronto tra le diverse usure per velocità e posizione motore

Per quanto riguarda la velocità del motore, gli andamenti per le diverse usure sono sovrapposti tranne che per l'intervallo 100 ms ÷ 200 ms, in cui avviene il taglio del tondino.

Per la posizione, invece, non si nota alcuna differenza tra le diverse usure, ovvero la posizione assunta dal motore in un ciclo di lavoro è indipendente dall'usura del coltello. Per tale motivo la variabile "Posizione Motore" non verrà più presa in considerazione.

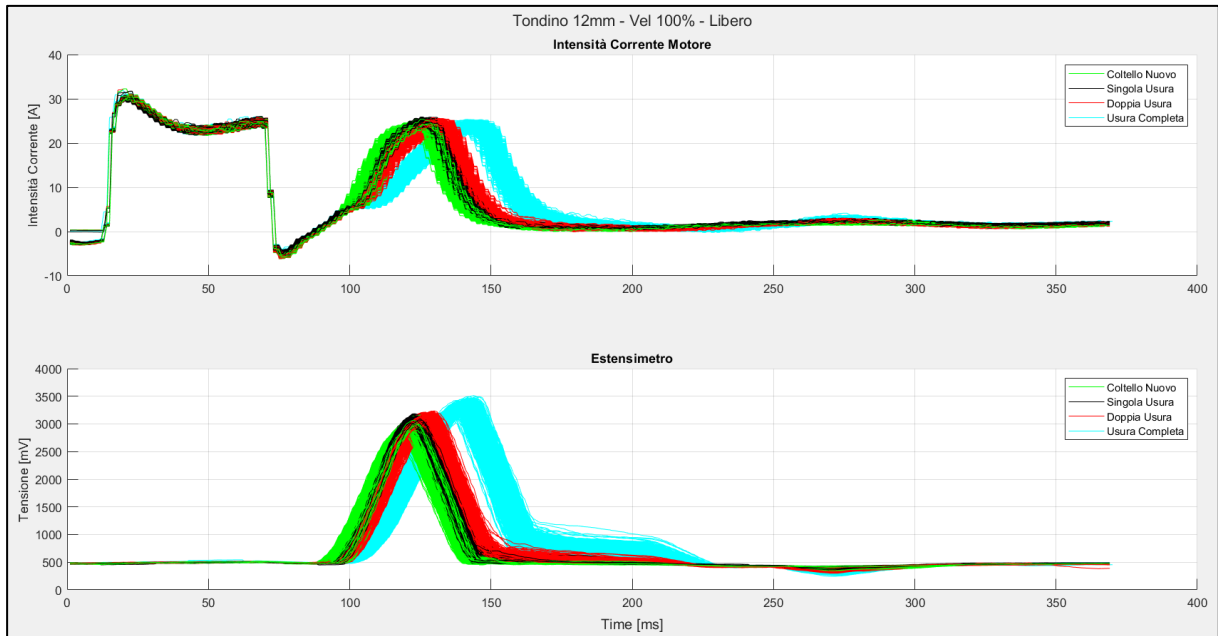


Figura 5.11 – Confronto tra le diverse usure per intensità di corrente del motore e forza (proporzionale alla tensione) rilevata dall'estensimetro sulla biella

Per entrambe le variabili in figura, all'aumentare dell'usura, il picco oltre i 100 ms, relativo al taglio del tondino, trasla in avanti nel tempo, rendendo più nette le differenze tra le diverse usure rispetto alla velocità del motore. Pertanto, sono stati presi in considerazione questi due segnali, non ponendo più l'attenzione sulla velocità del motore, per ridurre il più possibile il numero di variabili per l'individuazione dell'usura.

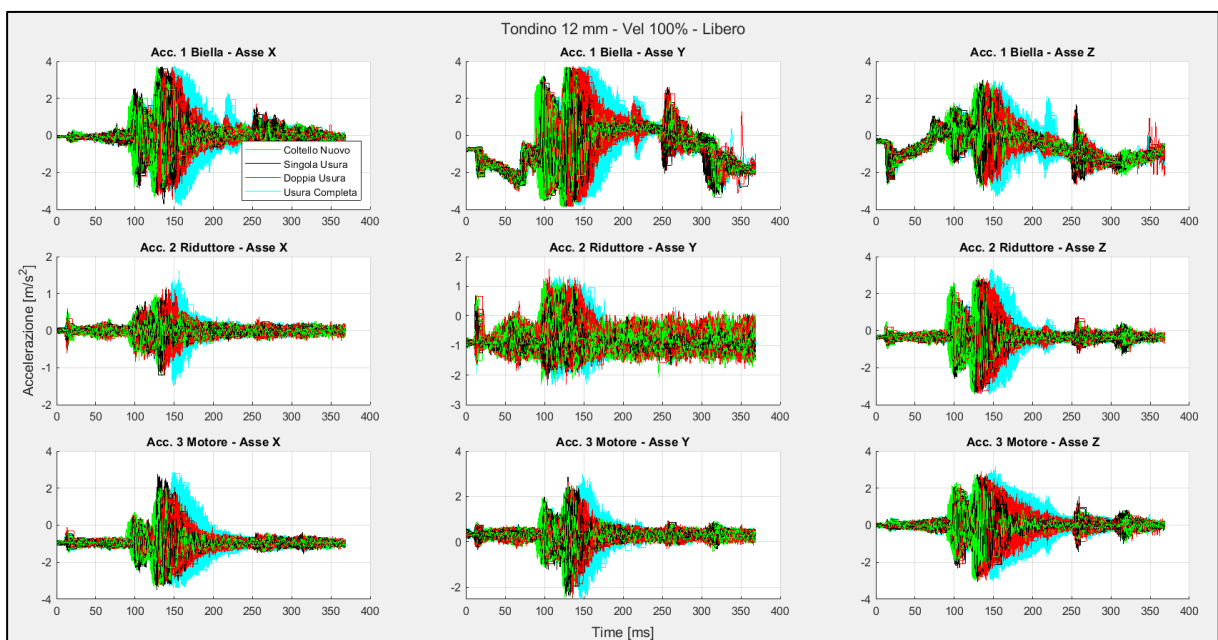


Figura 5.12 – Confronto tra le diverse usure per le accelerazioni rilevate dagli accelerometri posizionati sulla biella, riduttore e motore

Per tutte le vibrazioni rilevate dagli accelerometri posti sulla biella, sul riduttore e sul motore, la differenza tra le usure è visibile in particolare nell'intervallo di tempo tra 100 ms e 200 ms in cui avviene il taglio del tondino. Ciò significa che, nel momento del taglio, aumentano le vibrazioni nel gruppo di taglio ed esse sono tanto maggiori quanto maggiore è il degrado della lama del coltello.

#### **5.1.4 Estrapolazione delle features**

Una volta individuati i segnali più importanti per l'individuazione dell'usura, sono state estrapolate le relative features e successivamente ne sono state selezionate solo alcune, al fine di ridurre la dimensionalità dei dati per realizzare modelli di classificazione. Tuttavia, se da un lato la riduzione della dimensionalità dei dati porta vantaggi come un miglioramento delle prestazioni del classificatore, dall'altro una riduzione eccessiva potrebbe causare una mancanza di informazione nell'addestramento del classificatore.

Sono state estrapolate, quindi, le features descritte nel capitolo 4 (Valor medio, RMS, Deviazione Standard, Shape factor, Kurtosis, Skewness, Valore di picco, Crest factor, Clearance factor, Impulse factor) per le seguenti variabili:

- intensità corrente motore;
- forza rilevata dall'estensimetro posto sulla biella;
- accelerazione lungo X, Y, Z dell'accelerometro posto sul motore;
- accelerazione lungo X, Y, Z dell'accelerometro posto sul riduttore;
- accelerazione lungo X, Y, Z dell'accelerometro posto sulla biella.

Di seguito vengono riportati i valori delle features relative all'estensimetro e all'intensità di corrente del motore per le diverse usure.

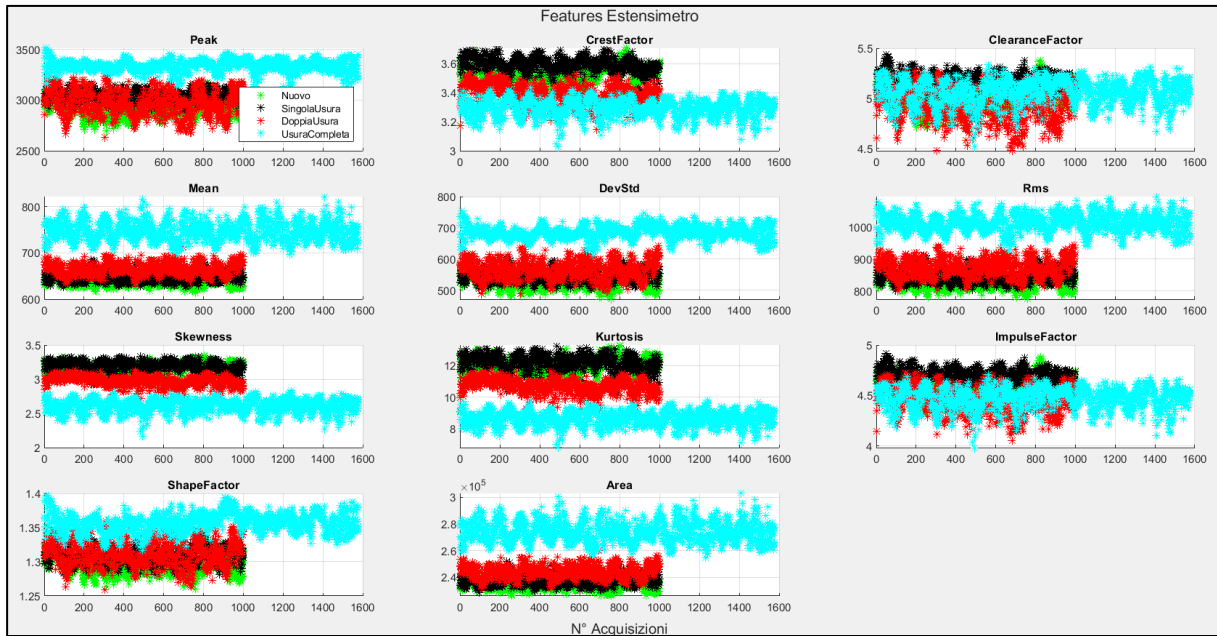


Figura 5.13 – Confronto tra features relative alla forza rilevata dall'estensimetro per le diverse usure

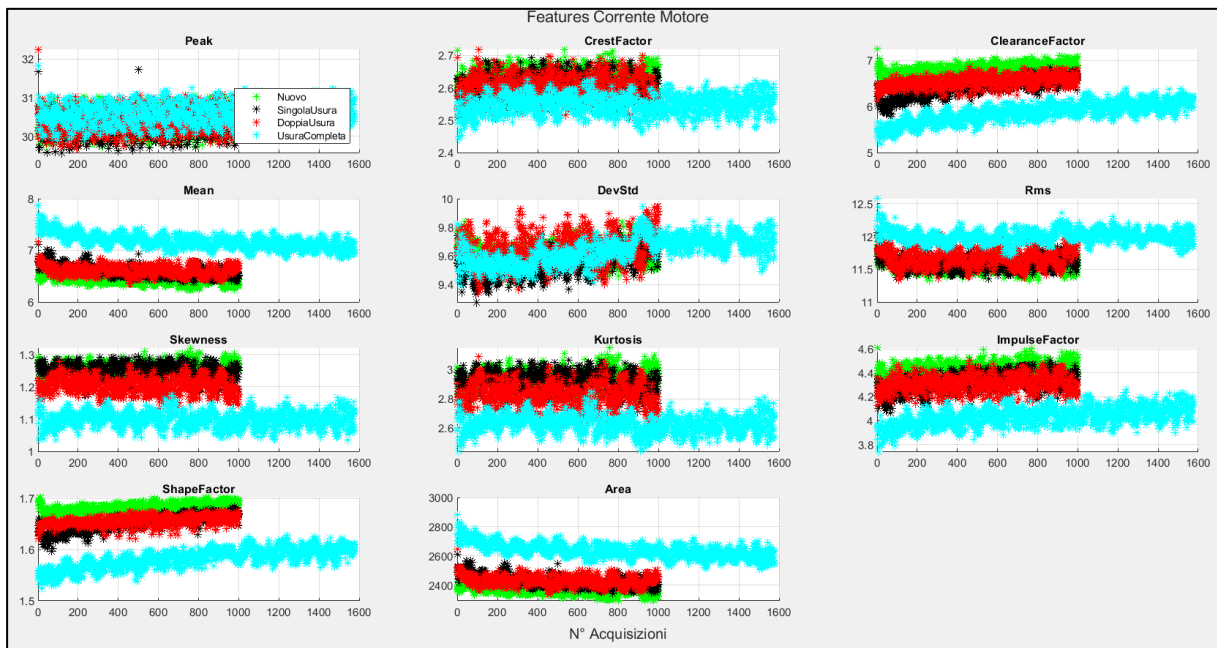


Figura 5.14 – Confronto features relative all'intensità di corrente del motore per le diverse usure

Osservando i grafici, si può ipotizzare quali features siano più determinanti nell'individuazione dell'usura: più le usure hanno valori distanti tra loro, più quelle features sono significative. Ad esempio, per la Deviazione Standard e il valore di Picco della corrente si potrebbe già pensare che esse non siano fondamentali, a differenza dell'Rms o della Deviazione Standard dell'estensimetro. Inoltre, come già era stato visto in precedenza nell'analisi dei grafici delle variabili acquisite durante i tagli, l'usura completa è ben distinguibile dalla lama nuova, a differenza della singola e della doppia usura che hanno valori di features sovrapposti.

Al fine di realizzare un modello che classifichi l'usura del coltello nelle tre classi "Nuovo", "Semi-usurato" e "Usurato", la singola e doppia usura vengono entrambe considerate come la seconda classe, mentre l'usura completa come la terza classe.

### 5.1.5 Selezione delle features con il criterio One-Way ANOVA

A questo punto, è stato utilizzato il criterio One-Way-ANOVA, che effettua un ranking delle features estrapolate, per capire quali siano le più importanti e, quindi, quelle da selezionare per addestrare il modello.

Il One-Way ANOVA è un particolare test d'ipotesi che utilizza l'analisi della varianza (ANalysis Of VARIances) per determinare se esiste una differenza statistica significativa tra le medie di tre o più gruppi di dati rispetto ad una categoria che li differenzia [7]. Il test accetta o rifiuta l'ipotesi nulla  $H_0$  "tutte le medie dei gruppi sono uguali" contro l'ipotesi alternativa  $H_1$  "almeno una media è differente". Ovvero, dati  $n$  gruppi di dati, il test determina se almeno una delle  $n$  medie è statisticamente differente dalle altre. Tuttavia, il test non dice quale delle medie è diversa statisticamente dalle altre, ma solamente se tutte le medie sono uguali statisticamente oppure no.

I risultati del criterio, in genere, vengono presentati servendosi di una tabella che mostra i seguenti sei parametri [8].

- Source: origine della varianza
  - Groups, variabilità tra gruppi;
  - Error, variabilità all'interno dei gruppi;
  - Totale, variabilità totale.
- SS: somma dei quadrati per ogni origine della varianza. SS permette di quantificare la variabilità di un dataset dalla media, calcolando la somma della varianza dovuta alle differenze tra i gruppi  $SS(\text{Groups})$  e la somma della varianza dovuta alle differenze all'interno di ogni gruppo  $SS(\text{Error})$ .
- df: gradi di libertà associati a ciascuna origine della varianza. Se  $N$  è il numero totale di osservazioni e  $k$  è il numero di gruppi,  $N-k$  sono i gradi di libertà all'interno dei gruppi (Error),  $k-1$  sono i gradi di libertà tra i gruppi (Groups) e  $N-1$  sono i gradi di libertà totali.

$$N-1 = (N-k) + (k-1)$$

- MS: media dei quadrati per ogni origine.

$$MS = \frac{SS}{df}$$

- F: statistica-F, che è la media quadratica del gruppo divisa per la media quadratica dell'errore.

$$F = \frac{MS(Groups)}{MS(Error)}$$

Se l'ipotesi nulla  $H_0$  è vera (cioè le medie sono uguali), F è vicino a 1, mentre se l'ipotesi  $H_0$  è falsa (cioè le medie non sono uguali), F è maggiore di 1.

- Prob > F: p-value, che è la probabilità che la statistica-F possa assumere un valore maggiore del valore calcolato della statistica del test. Più esso è piccolo più le differenze tra le medie dei gruppi sono significative.

Come esempio viene riportata la tabella ANOVA per l'Rms dell'estensimetro, in cui il numero totale di osservazioni è 4578 (cioè il numero totale di acquisizioni), mentre il numero dei gruppi è 3 (cioè i tre tipi di coltello in cui sono suddivise le acquisizioni).

Source	SS	df	MS	F	Prob>F
Groups	2.90963e+07	2	14548173.414	22444.21	0
Error	2.96548e+06	4575	648.193		
Total	3.20618e+07	4577			

Figura 5.15 – Tabella ANOVA relativa alla feature Rms dell'estensimetro

In questo caso  $p = 0$  indica che per l'Rms si riesce a distinguere molto bene le usure.

In Matlab, la funzione `anova1` calcola i parametri della tabella ANOVA per ogni feature. Per selezionare le features, esse vengono ordinate in base al parametro F.

Inizialmente, il criterio è stato applicato alle features relative all'intensità di corrente del motore e alla forza rilevata dall'estensimetro.

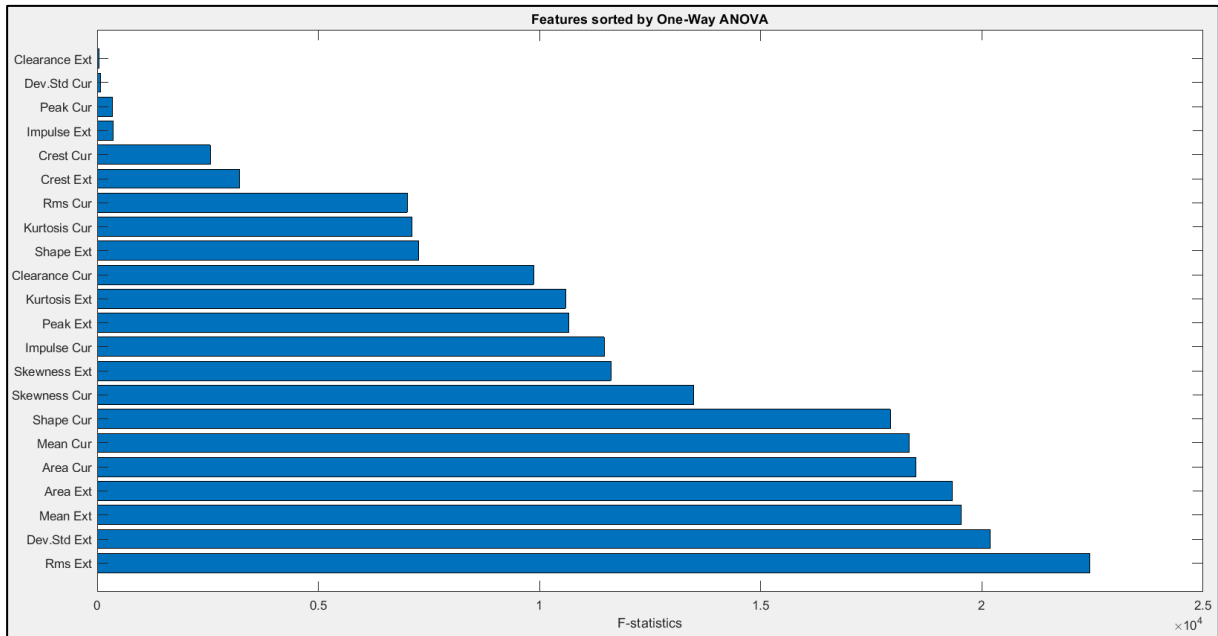


Figura 5.16 – Features ordinate secondo il parametro F dell'One-Way ANOVA

La figura 5.16 mostra il grafico a barre delle features ordinate secondo il parametro F. Come era già stato ipotizzato in precedenza per quanto riguarda l'estensimetro, le più importanti sono l'RMS e la Deviazione Standard, mentre le meno indicative sono il Clearance Factor dell'estensimetro e la Deviazione Standard della corrente.

Per facilitare la selezione delle features, il grafico è stato normalizzato rispetto il valore maggiore.

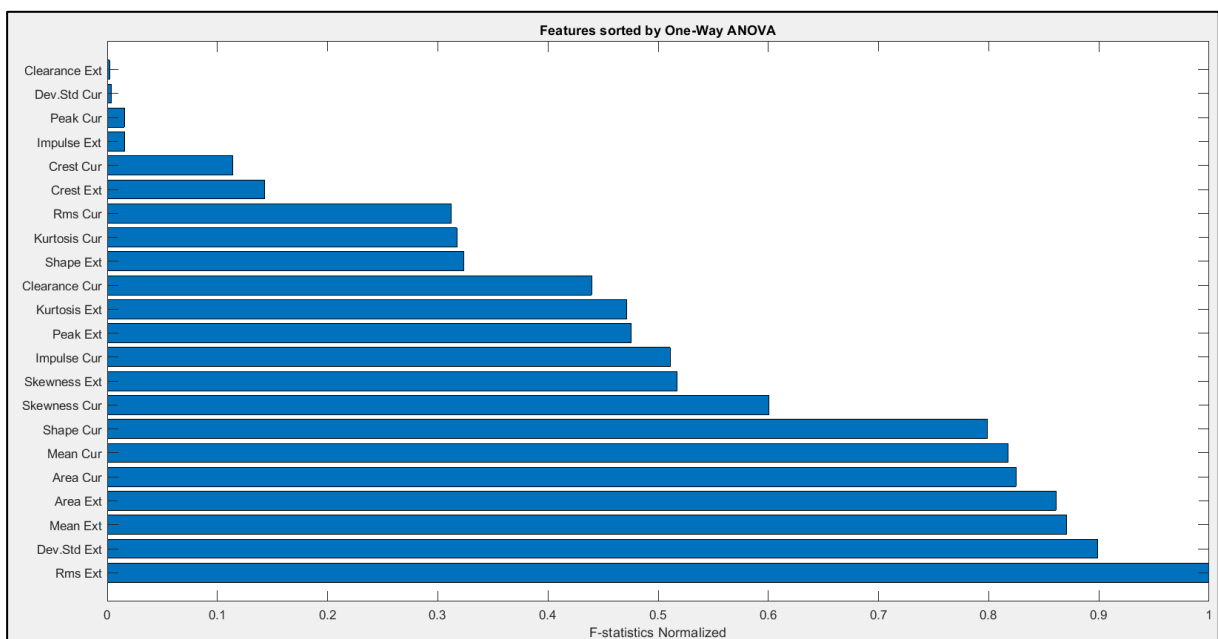


Figura 5.17 – Features normalizzate ordinate secondo il parametro F dell'One-Way ANOVA



Al fine di addestrare il classificatore, sono state selezionate le features con valore di normalizzazione maggiore di 0.6, ovvero quelle più significative della presenza dell'usura:

1. Estensimetro - Rms;
2. Estensimetro - Deviazione Standard;
3. Estensimetro - Mean;
4. Estensimetro - Area;
5. Corrente - Area;
6. Corrente - Mean;
7. Corrente - Shape Factor.

Successivamente, il criterio è stato applicato alle features delle accelerazioni. Di seguito vengono mostrate solamente le features relative all'accelerometro posizionato sulla biella rispetto l'asse X, in quanto le stesse considerazioni si possono estendere anche per le features di tutte le altre accelerazioni.

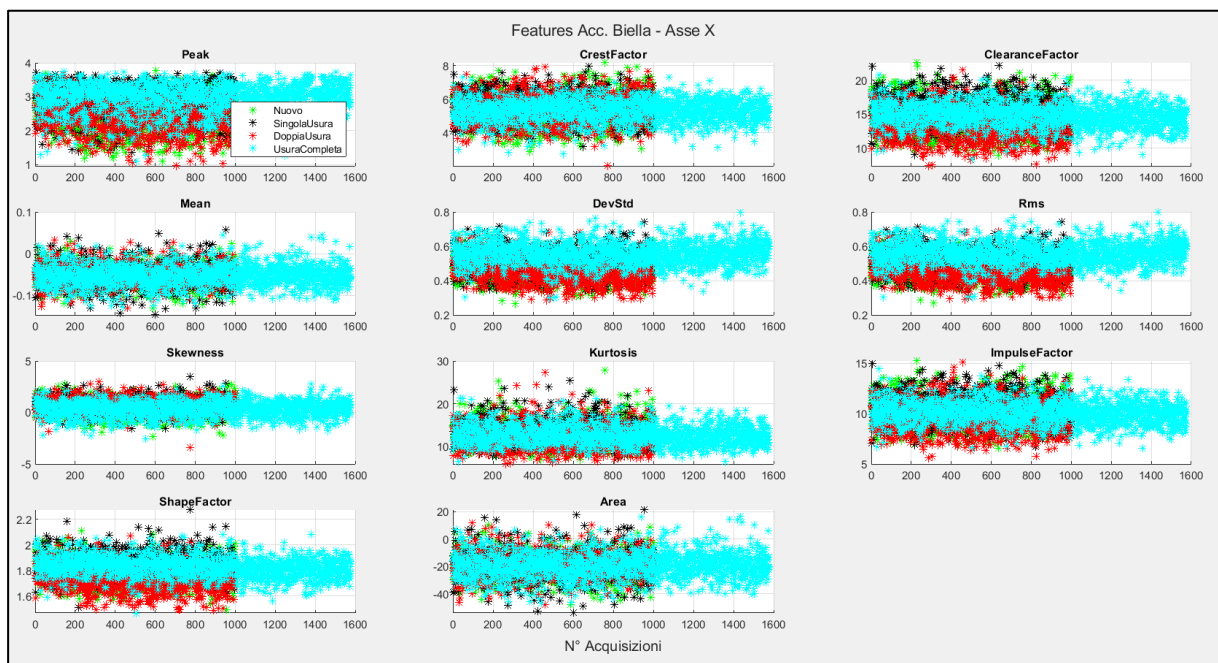


Figura 5.18 – Confronto fra features relative all'accelerometro lungo l'asse X della biella per le diverse usure

Dall'osservazione delle features delle accelerazioni si può ipotizzare che esse abbiano valori meno significativi rispetto a quelli relativi alla forza rilevata dall'estensimetro e all'intensità di corrente del motore.

A conferma di quanto ipotizzato, il ranking delle features secondo il One-Way ANOVA è il seguente.

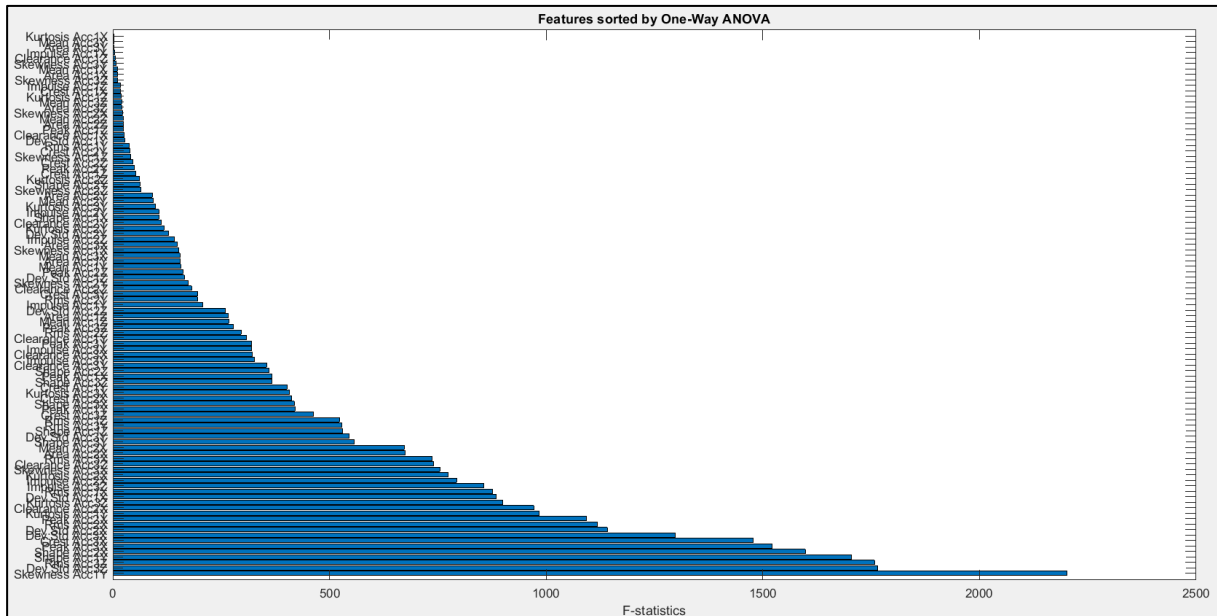


Figura 5.19 – Features delle accelerazioni ordinate secondo il parametro F dell'One-Way ANOVA

Il valore maggiore della statistica-F per le accelerazioni è inferiore al valore dello Shape Factor della corrente, ultima feature considerata in precedenza; di conseguenza, non è stata selezionata alcuna feature delle accelerazioni nell'addestramento del classificatore. Perciò, non essendo ritenuti utili i valori degli accelerometri, essi non sono stati più considerati e da qui in poi si farà sempre riferimento all'intensità di corrente del motore e alla forza rilevata dall'estensimetro.

### 5.1.6 Realizzazione classificatore con il *ClassificationLearner*

Una volta definiti i segnali e, di essi, quali features considerare, è stato realizzato il classificatore per l'individuazione dell'usura utilizzando il toolbox *ClassificationLearner* di Matlab. Il toolbox riceve in ingresso un dataset costituito dai predittori e dai target e addestra i vari modelli di classificatori a disposizione; alla fine verrà scelto quello con il maggior valore di accuracy, metrica utilizzata per valutare le prestazioni del modello sul validation-set; infatti, il dataset iniziale in ingresso al classificatore può essere suddiviso in due parti: una con cui addestrare direttamente il modello (training-set) e una da utilizzare per convalidare il modello appena addestrato (validation-set). Le tecniche di suddivisione del dataset messe a disposizione dal toolbox sono le seguenti [9].

- Cross Validation: il dataset viene diviso in k parti di ugual numero, si esclude iterativamente una parte alla volta per k volte e si cerca di predire la parte esclusa con le restanti k-1 parti non escluse, al fine di verificare la bontà del modello di predizione utilizzato.

- **Holdout Validation:** il dataset viene separato direttamente nel training-set e nel validation-set, con una numerosità scelta in percentuale dall'utente. Rispetto alla tecnica precedente, richiede meno tempo per il calcolo, ma la sua valutazione dipende molto da quali punti finiscono nel training-set e quali finiscono nel validation-set; quindi, la valutazione può essere significativamente diversa a seconda di come viene fatta la divisione.
- **No Validation:** il modello viene addestrato utilizzando tutto il dataset a disposizione.

Nei classificatori realizzati e descritti nella presente tesi è stata sempre scelta la prima tecnica, settando a 5 il numero di parti in cui suddividere il dataset: ad ogni iterazione, quattro parti vengono utilizzate per l'addestramento, mentre la parte restante per la validazione. Il vantaggio di questa tecnica è la protezione del modello dall'overfitting, ovvero la situazione in cui il modello si adatta alle caratteristiche che sono specifiche solo del training set, ma che non hanno riscontro nel resto dei casi. Tuttavia, lo svantaggio di questo metodo è che l'algoritmo di addestramento deve essere eseguito nuovamente da zero per k volte; ciò significa che occorrono k volte calcoli per fare una valutazione.

Quindi, per il dataset iniziale in ingresso, i predittori sono le features selezionate precedentemente, mentre il target è il valore dell'usura associato. I valori di target scelti sono:

- “0” = Coltello nuovo, associato alle features del coltello in assenza di usura;
- “1” = Coltello semi-usurato, associato alle features dell'usura singola e usura doppia;
- “2” = Coltello usurato, associato alle features dell'usura completa.

A questo punto sono stati addestrati tutti i modelli disponibili nel ClassificationLearner. Il classificatore migliore, ovvero quello con il valore di Accuracy maggiore (97,6%) è risultato essere il “Quadratic SVM”.

#### **5.1.6.1 Classificatore SVM**

Il Support Vector Machine (SVM) è un algoritmo di apprendimento automatico supervisionato, utilizzato per la classificazione [6]. L'obiettivo dell'algoritmo è quello di trovare un iperpiano affine (ovvero che non necessita di passare per l'origine) che divide un set di dati in più classi, massimizzando il margine tra le classi stesse, ovvero la distanza minima dall'iperpiano ai punti delle classi.

L'iperpiano, in uno spazio p-dimensionale, è il sottospazio di dimensione p-1: in due dimensioni è una linea, in tre dimensioni è un piano, ecc. L'equazione dell'iperpiano è:

$$\beta_0 + \beta_1 X_1 + \beta_2 X_2 + \dots + \beta_p X_p = 0$$

in cui  $X_i$  è il vettore delle  $p$  osservazioni per la caratteristica  $i$ , mentre  $\beta_0, \beta_1, \dots, \beta_p$  sono i parametri.

Esso divide lo spazio nelle due regioni:

$$\beta_0 + \beta_1 X_1 + \beta_2 X_2 + \dots + \beta_p X_p > 0$$

$$\beta_0 + \beta_1 X_1 + \beta_2 X_2 + \dots + \beta_p X_p < 0$$

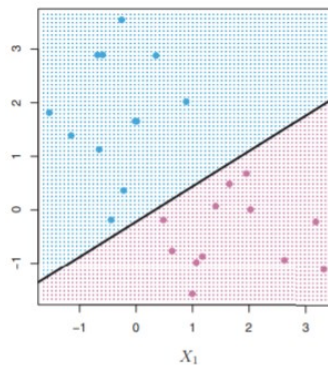


Figura 5.20 – Iperpiano che divide i dati nelle due classi [6]

In generale, possono essere individuati più iperpiani, come mostrato nella seguente figura.

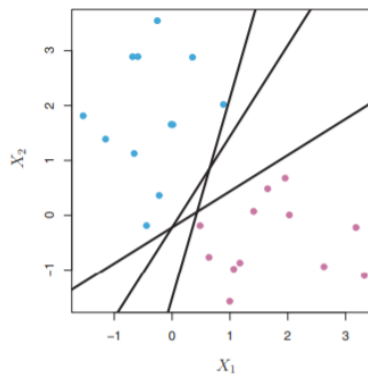


Figura 5.21 – Più iperpiani che dividono i dati nelle due classi [6]

Per decidere quale degli iperpiani scegliere, la tecnica è selezionare quello che massimizza il margine, ovvero la minima distanza di ogni dato tra ciascuno degli iperpiani trovati. Infatti, molto probabilmente, più i dati sono lontani dall'iperpiano, più essi sono classificati correttamente. Di seguito, viene indicato quale iperpiano della figura 5.21 massimizza il margine.

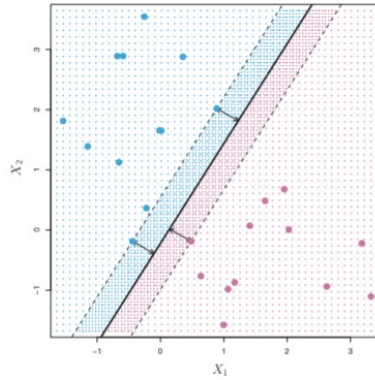


Figura 5.22 – Iperpiano che massimizza il margine [6]

La figura 5.22, inoltre, mostra tre osservazioni equidistanti dall'iperpiano: esse sono note come vettori di supporto (*support vectors*), in quanto supportano l'iperpiano; infatti, se esse si spostano di poco, l'iperpiano potrebbe cambiare.

Tuttavia, non sempre si riesce a individuare un iperpiano che divida lo spazio linearmente.

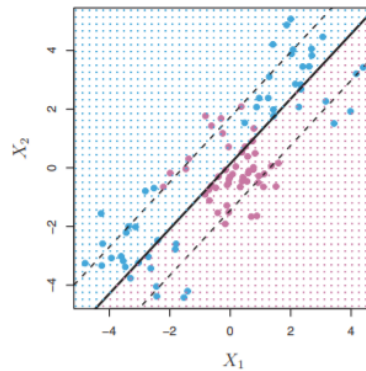


Figura 5.23 – Dati non separabili linearmente [6]

L'SVM, allora, separa le classi con un limite non lineare, mappando il dataset con il “metodo del kernel”. Il kernel viene definito come:

$$K(\mathbf{X}_i, \mathbf{X}_j) = \langle \mathbf{X}_i, \mathbf{X}_j \rangle$$

dove  $K$  è la funzione del kernel,  $\mathbf{X}_i$  e  $\mathbf{X}_j$  sono vettori di input a dimensione  $p$  e  $\langle, \rangle$  è il prodotto scalare.

La scelta della funzione del kernel può influire sulle prestazioni del modello SVM. Le funzioni più diffuse sono le seguenti.

- Kernel lineare:

$$K(\mathbf{X}_i, \mathbf{X}_j) = \mathbf{X}_i \cdot \mathbf{X}_j$$

- Kernel polinomiale di grado  $d$ :

$$K(\mathbf{X}_i, \mathbf{X}_j) = (\mathbf{X}_i \cdot \mathbf{X}_j + c)^d$$

in cui  $c$  è una costante, mentre  $d$  è un grado di libertà: se  $d = 1$  il kernel è lineare, mentre se  $d > 1$  si ha un modello più complesso.

Nel caso del classificatore ottenuto con il ClassificationLearner, si ha un iperpiano non lineare con funzione kernel quadratica, ovvero con  $d = 2$ .

- Kernel gaussiano:

$$K(\mathbf{X}_i, \mathbf{X}_j) = e\left(\frac{-|\mathbf{X}_i - \mathbf{X}_j|^2}{2\sigma^2}\right)$$

Il kernel gaussiano è un limite decisionale più complesso rispetto ai precedenti.

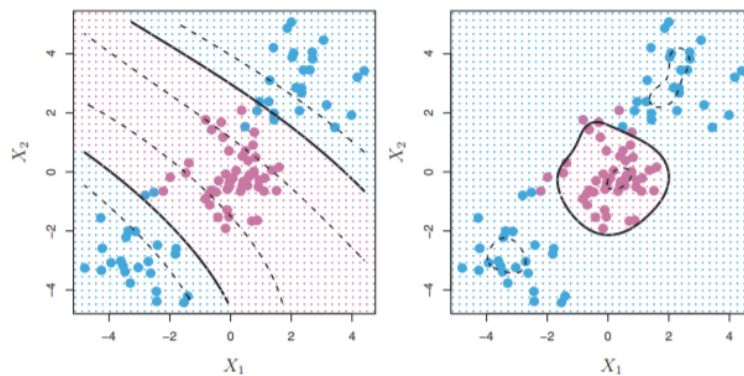


Figura 5.24 – A sinistra SVM con kernel polinomiale di grado 3 e a destra SVM con kernel gaussiano [6]

In Matlab, comunque sia, è il toolbox ClassificationLearner che, in modo automatico, applica la funzione kernel.

### 5.1.6.2 Matrice di confusione

Al fine di valutare le prestazioni del modello ottenuto, un utile strumento è la matrice di confusione, una tabella in cui le colonne rappresentano le previsioni del modello, mentre le righe rappresentano lo stato reale [10]. La matrice di confusione riporta le prestazioni del classificatore, addestrato sul training-set, sul validation-set. Ad esempio, nel caso in esame, la matrice di confusione, relativa al modello Quadratic SVM, è la seguente.

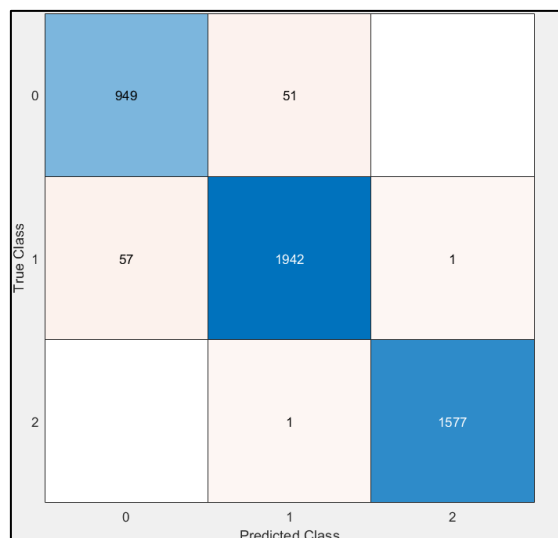


Figura 5.25 – Matrice di confusione relativa al classificatore *Quadratic SVM*

Osservando la tabella, si può notare che il modello riconosce correttamente il tipo di usura. In particolare, il coltello usurato viene sempre individuato correttamente (tranne che per un solo caso), mentre il coltello nuovo e il coltello semi-usurato a volte vengono scambiati. In ogni caso le prestazioni sono soddisfacenti, perché il problema sarebbe stato nella classificazione della classe 0 come la 3 o viceversa, ovvero la confusione di situazioni estreme.

Per comprendere meglio la relazione tra valori reali e previsti, alla matrice di confusione vengono associati i seguenti termini:

- true positive (TP, o veri positivi): casi in cui la previsione corrisponde alla classe reale;
- true negative (TN, o veri negativi): casi in cui il modello prevede che il caso in esame non corrisponde ad una certa classe e non lo è veramente;
- false positive (FP, o falsi positivi): casi in cui il modello classifica il caso in esame come un'altra classe, ma in realtà non lo è (conosciuti anche come errori di tipo 1);
- false negative (FN, o falsi negativi): casi in cui il modello prevede che il caso in esame non è una certa classe, anche se in realtà lo è (conosciuti anche come errori di tipo 2).

Ad esempio, per quanto riguarda la matrice di confusione di figura 5.26, i termini appena descritti sono riportati nella tabella 2.

Tabella 2 – Termini associati alla matrice di confusione in esame

Classi	TP	TN	FP	FN
<b>0</b>	949	$1942 + 1577 = 3519$	57	51
<b>1</b>	1942	$949 + 1577 = 2526$	$51 + 1 = 52$	$57 + 1 = 58$
<b>2</b>	1577	$949 + 1942 = 2891$	1	1

A partire da questi valori, Matlab mette a disposizione le seguenti metriche per valutare le prestazioni del classificatore per ognuna delle classi [11].

- True Positive Rates (TPR): percentuale di osservazioni correttamente classificate per classe vera.

$$TPR = \frac{TP}{TP + FN} \%$$

- False Positive Rates (FPR): percentuale di osservazioni erroneamente classificate per classe vera.

$$FPR = \frac{FN}{TP + FN} \%$$

- Positive Predictive Values (PPV): percentuale di osservazioni correttamente classificate per classe prevista.

$$PPV = \frac{TP}{TP + FP} \%$$

- False Discovery Rates (FDR): percentuale di osservazioni erroneamente classificate per classe prevista.

$$FDR = \frac{FP}{TP + FP} \%$$

L'accuracy (precisione), che è la metrica con cui viene scelto il classificatore, è la media del TPR per ogni classe. Ad esempio, nel caso in esame, le metriche sono riportate nella seguente tabella.

*Tabella 3 – Metriche del classificatore in esame*

<b>Classi</b>	<b>TPR</b>	<b>FPR</b>	<b>PPV</b>	<b>FDV</b>
<b>0</b>	94,9%	5,1%	94,3%	5,7%
<b>1</b>	97,1%	2,9%	97,4%	2,6%
<b>2</b>	99,9%	0,1%	99,9%	0,1%
<b>Accuracy</b>	97,3%			

### 5.1.7 Test del classificatore

Per verificare le prestazioni del classificatore realizzato, esso è stato testato su delle nuove acquisizioni, di cui l'usura era nota a priori (ovvero era noto il tipo di coltello utilizzato nei tagli).



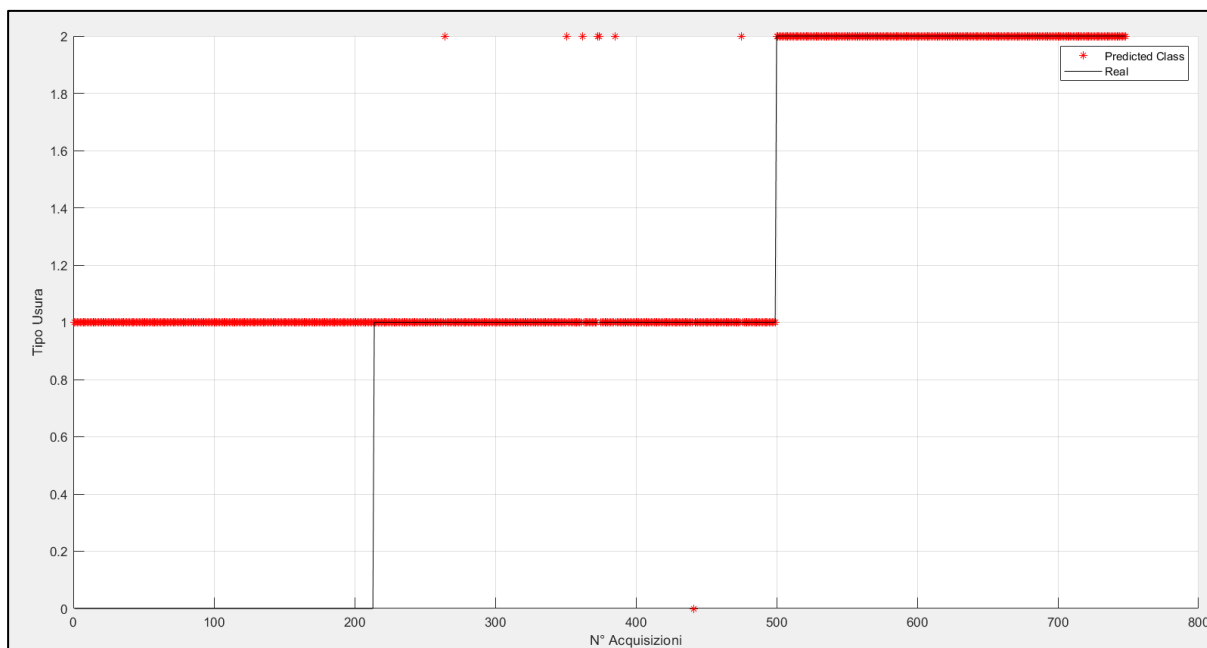


Figura 5.26 – Test del classificatore su nuove acquisizioni, di usura nota a priori

Osservando il grafico, in cui in nero c'è il valore reale dell'usura, mentre in rosso il valore predetto dal classificatore, si può affermare che:

- il coltello usurato viene classificato correttamente;
- il coltello semi-usurato viene classificato come coltello usurato solamente per pochissime acquisizioni;
- il coltello nuovo viene classificato sempre come coltello semi-usurato, sbagliando per tutte le acquisizioni.

Focalizzando i possibili motivi dietro agli errori del classificatore, l'unica differenza tra le acquisizioni utilizzate per addestrare il classificatore e quelle per il test è la diversa temperatura del motore alla quale iniziano le acquisizioni; infatti, le nuove prove sono state acquisite a temperature nettamente inferiori rispetto a quelle iniziali, come mostrato nella seguente tabella.

Tabella 4 – Temperature iniziali del motore per le diverse acquisizioni effettuate

Usura	Temperatura di inizio acquisizioni di train del classificatore	Temperatura di inizio acquisizioni di test del classificatore
Coltello Nuovo	49 °C	27 °C
Usura Singola	37 °C	19 °C
Usura Completa	38 °C	23 °C

Quindi, dalla temperatura, che fino a questo momento non è mai stata presa in considerazione in quanto essa aumenta per il numero di tagli effettuati e non per l'usura, dipendono le altre

variabili considerate, in particolare l'intensità di corrente del motore, come visibile nella figura 5.27.

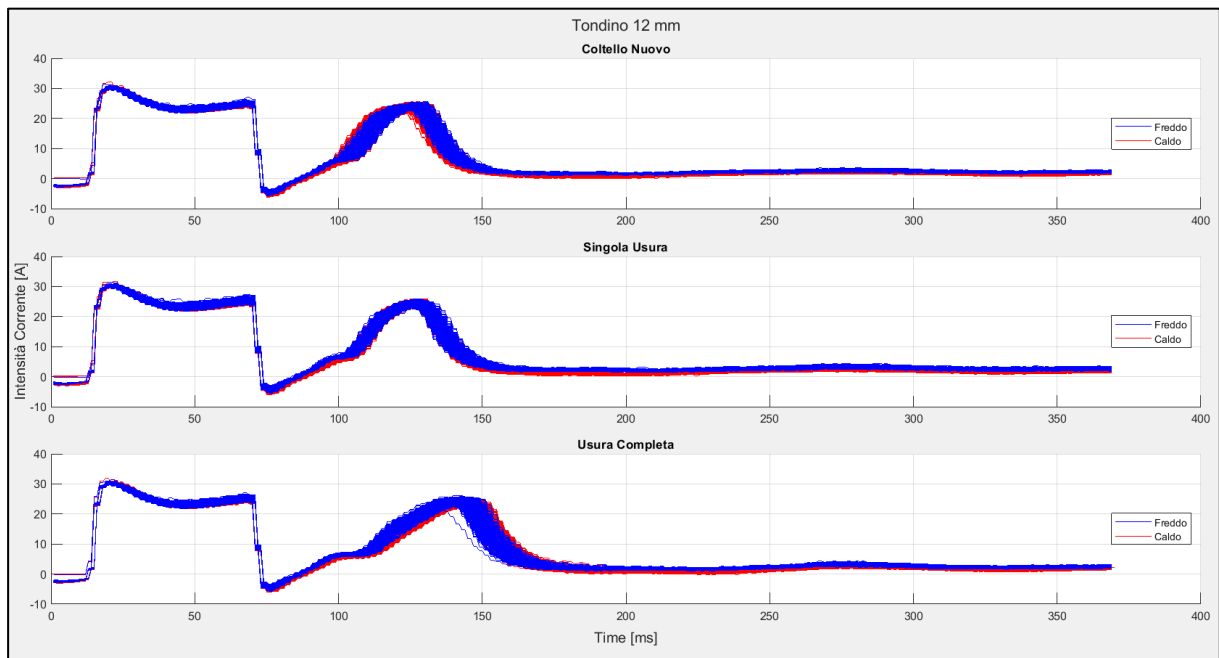


Figura 5.27 – Andamento dell'intensità di corrente del motore per le diverse usure, confrontando acquisizioni a caldo e a freddo

Tuttavia, la netta differenza causata dalle diverse temperature si nota nelle features relative alla corrente. Di seguito vengono mostrati i valori delle features del coltello nuovo utilizzate nel training (a caldo), di quelle del coltello nuovo utilizzate nel test (a freddo) e di quelle del coltello semi-usurato utilizzate nel training (a caldo).

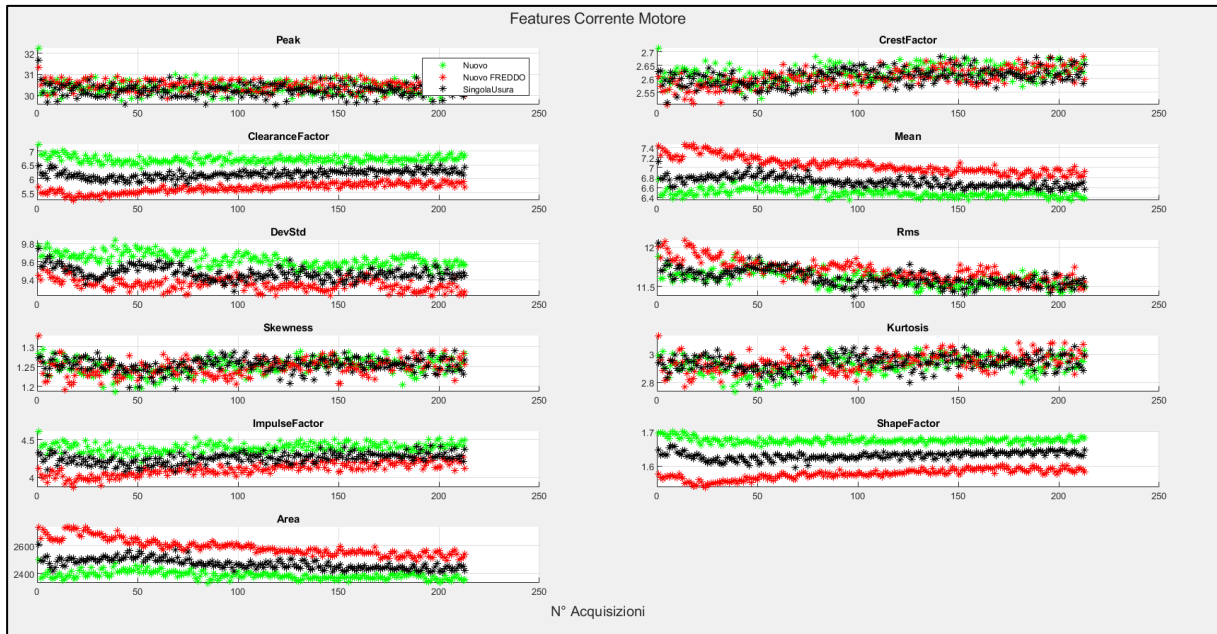


Figura 5.28 – Confronto fra le features delle acquisizioni a caldo e a freddo relative al coltello nuovo e di quelle a caldo relative al coltello semi-usurato

Per alcune features della corrente, come l'Area, lo Shape Factor e la Media (quelle utilizzate per il training del modello), c'è una netta differenza tra quelle a caldo e quelle a freddo del coltello nuovo. Inoltre, queste ultime risultano molto più simili a quelle a caldo della singola usura: questo spiega il motivo per cui il modello classifica il coltello nuovo come coltello semi-usurato.

Perciò, il modello è stato addestrato nuovamente, seguendo gli stessi step descritti in precedenza.

- I. Applicazione del Criterio One-Way ANOVA, che determina come features più significative:
  1. Estensimetro - Rms;
  2. Estensimetro - Deviazione Standard;
  3. Estensimetro - Mean;
  4. Estensimetro - Area;
  5. Corrente - Skewness;
  6. Estensimetro - Skewness;
  7. Estensimetro - Kurtosis.

Come era prevedibile, le features più significative sono tutte relative all'estensimetro, meno dipendente dalla temperatura. L'unica eccezione è la Skewness della corrente, che

come mostrato in figura 5.29, è la feature della corrente in cui i valori tra acquisizioni a caldo e a freddo sono più sovrapposti e quindi meno dipendenti dalla differenza di temperatura.

- II. Realizzazione del classificatore con il ClassificationLearner; il classificatore con le migliori prestazioni (Accuracy = 93,9%) è il Quadratic SVM, con la seguente matrice di confusione.

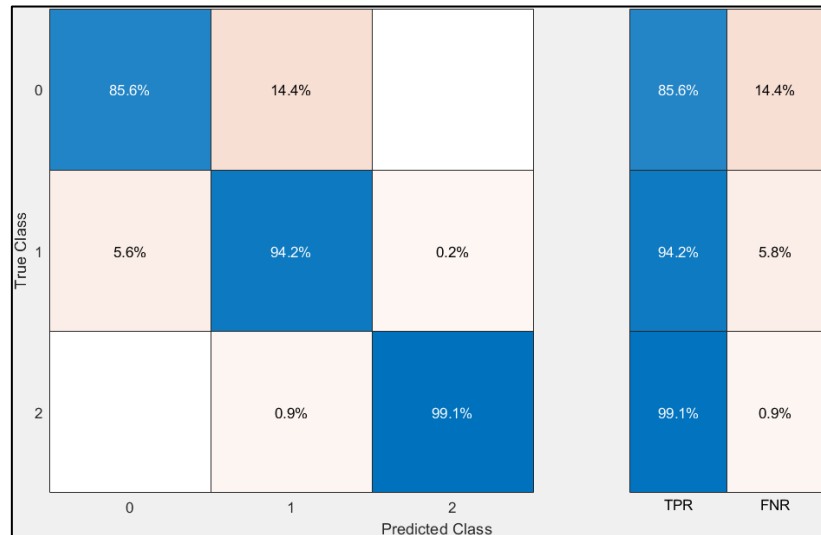


Figura 5.29 – Matrice di confusione del classificatore per il tondino 12 mm, addestrato anche sulle acquisizioni a freddo

Rispetto alla matrice di confusione del precedente classificatore, le prestazioni in generale sono peggiorate. Il motivo è che in questo caso nel training-set e nel validation-set ci sono anche i dati acquisiti a freddo.

Di seguito viene mostrato il confronto tra i due classificatori ottenuti finora su acquisizioni realizzate con diverse usure e a diversa temperatura iniziale del motore.

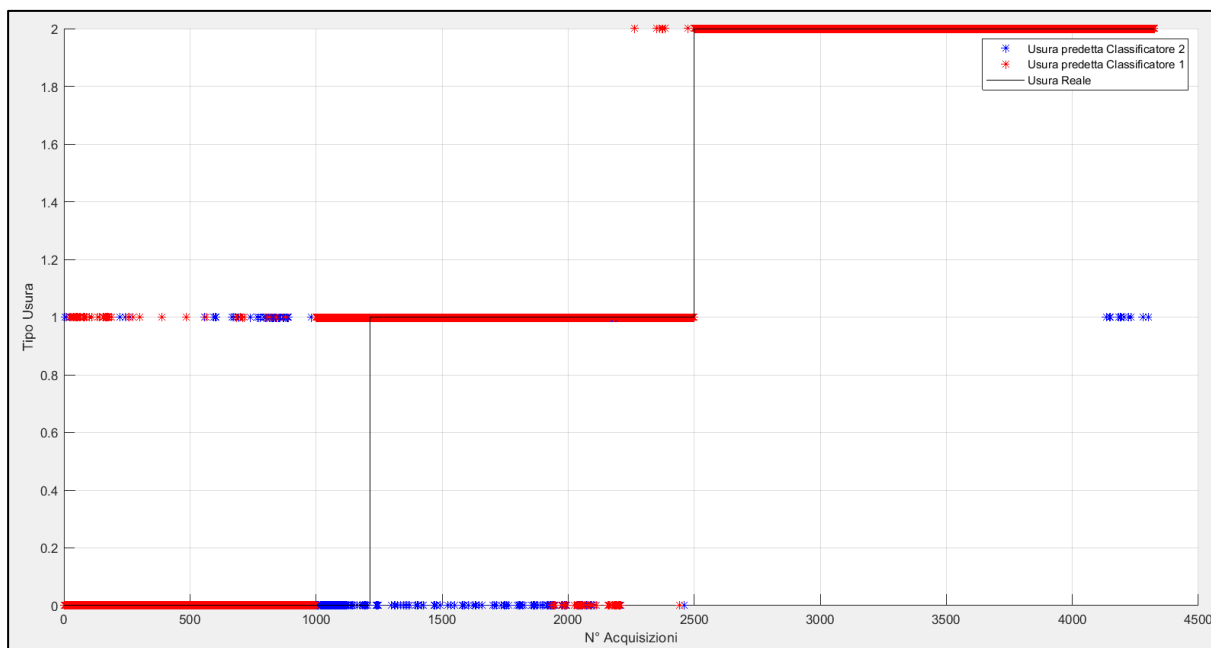


Figura 5.30 – Confronto tra predizioni dei classificatori

Anche se con il secondo modello peggiorano le classificazioni del coltello semi-usurato (“1”), migliorano però quelle del coltello nuovo rispetto al primo modello.

### 5.1.8 Coltello Hardox e Coltello C45

A questo punto sono state effettuate delle acquisizioni con coltelli Hardox e C45, materiali non trattati, più “morbidi” rispetto all’Unimax, quello standard utilizzato nelle operazioni di taglio. Il motivo di tale scelta è che, se con l’Unimax sono necessari diversi migliaia di tagli prima di vedere gli effetti dell’usura sulla lama del coltello, con i materiali più morbidi ne servono molti di meno. Di seguito, vengono mostrati gli effetti sulla lama dell’Hardox dopo 18000 tagli (a sinistra) e sul C45 dopo 3000 tagli (a destra).



Figura 5.31 – Confronto fra usura Hardox (a sinistra) e usura C45 (a destra), viste dal basso



Figura 5.32 – Confronto fra usura Hardox a (sinistra) e usura C45 a (destra), viste di lato

Il C45, nettamente più morbido dell'Hardox, subisce maggiormente l'effetto dell'usura nel corso dei tagli; l'Hardox, invece, mostra un degrado sulla lama con un profilo simile all'usura completa realizzata manualmente, anche se meno evidente.



Figura 5.33 – Confronto tra usura Hardox (a sinistra) e usura completa realizzata manualmente (a destra)

Confrontando i grafici dell'estensimetro per le diverse usure e per i diversi coltelli, si ottiene la seguente figura.

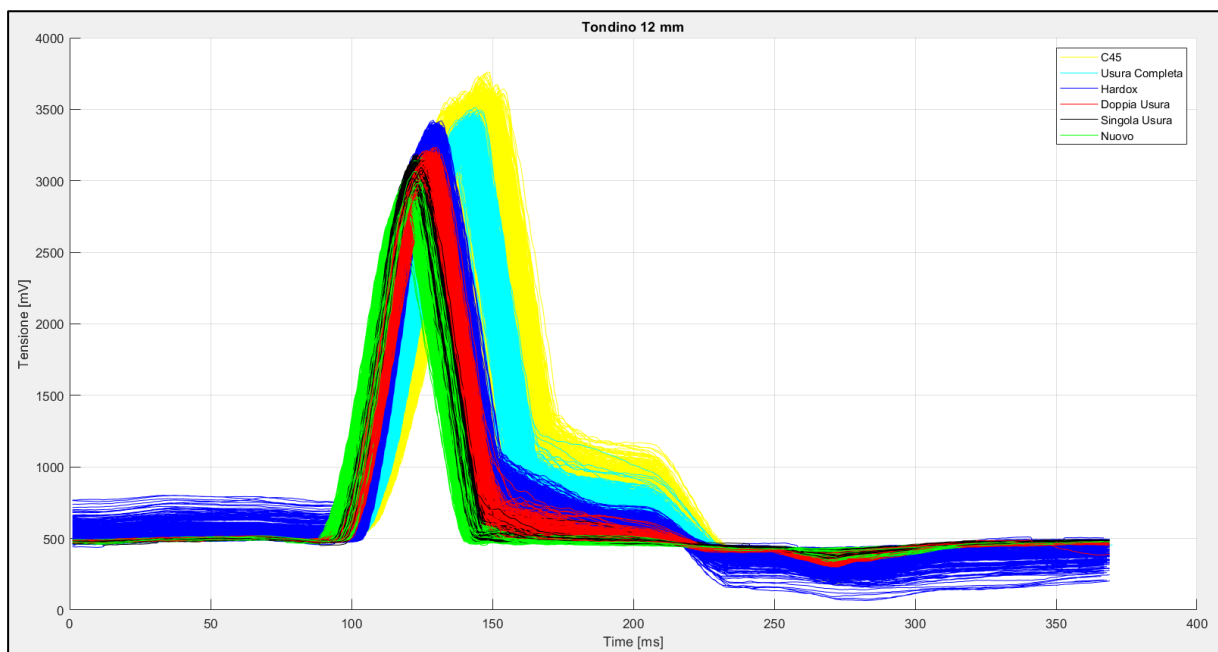


Figura 5.34 – Andamento della forza (proporzionale alla tensione) rilevata sulla biella dall'estensimetro per le diverse usure e per i diversi coltelli

All'aumentare dell'usura il picco trasla in avanti nel tempo e aumenta in ampiezza; in presenza di degrado, l'area sottesa dalla curva relativa alla forza aumenta ad indicare che l'energia immessa nel sistema è maggiore. Questo si spiega in quanto un taglio poco netto implica una deformazione ulteriore del materiale (come visibile nella problematica di figura 2.6) che richiede energia.

Inoltre, come già ipotizzato osservando il degrado sulla lama, la presenza dell'usura nell'Hardox è meno netta rispetto a quella completa realizzata manualmente.

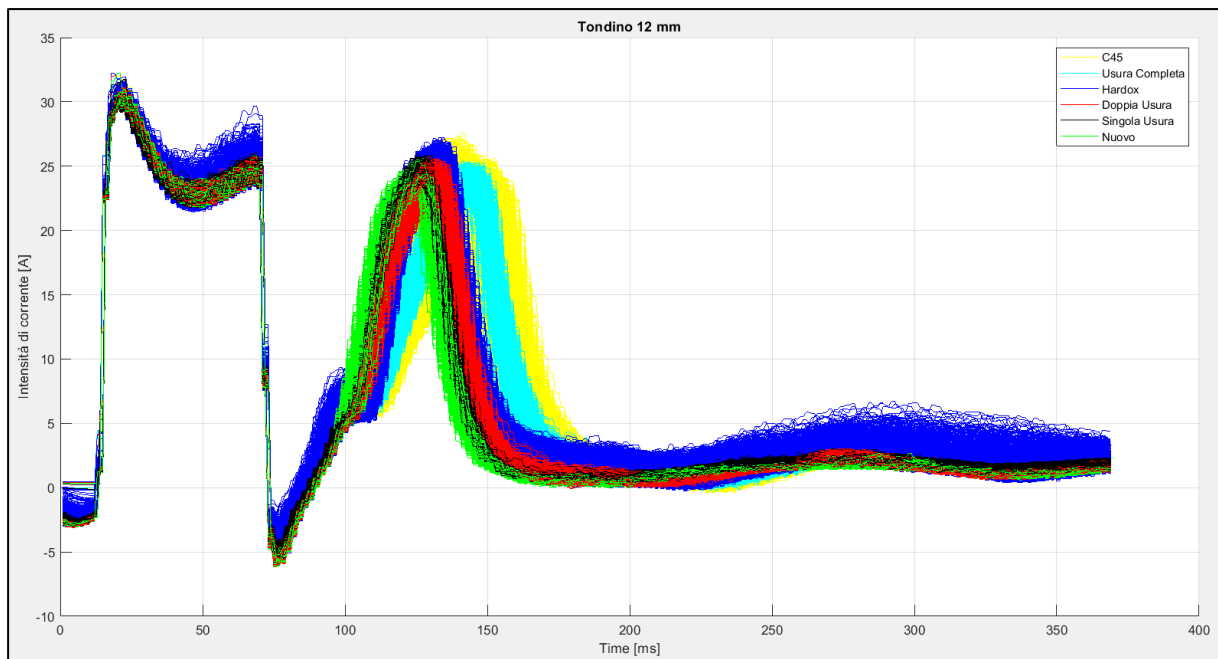


Figura 5.35 – Andamento dell'intensità di corrente del motore per le diverse usure e per i diversi coltelli

Per quanto riguarda la corrente del motore, nell'intervallo in cui avviene il taglio, il picco rimane all'incirca costante, ma la presenza dell'usura lo trasla in avanti nel tempo.

### 5.1.8.1 Clustering k-means

Per migliorare il classificatore realizzato, sono stati aggiunti nel training-set del modello anche le features dell'Hardox e del C45. Tuttavia, il problema di questi dati è che non si conosce le condizioni del coltello se nuovo, semi-usurato o usurato, in quanto per entrambi i materiali si parte dal caso di assenza di degrado per arrivare alla fine delle acquisizioni alla presenza di usura. Inoltre, come visibile nello scatter plot relativo all'Rms e alla Deviazione Standard dell'estensimetro di seguito riportato, le features dell'Hardox e del C45 condividono punti con quelle relative al caso di assenza di usura e con quelle dell'usura completa.

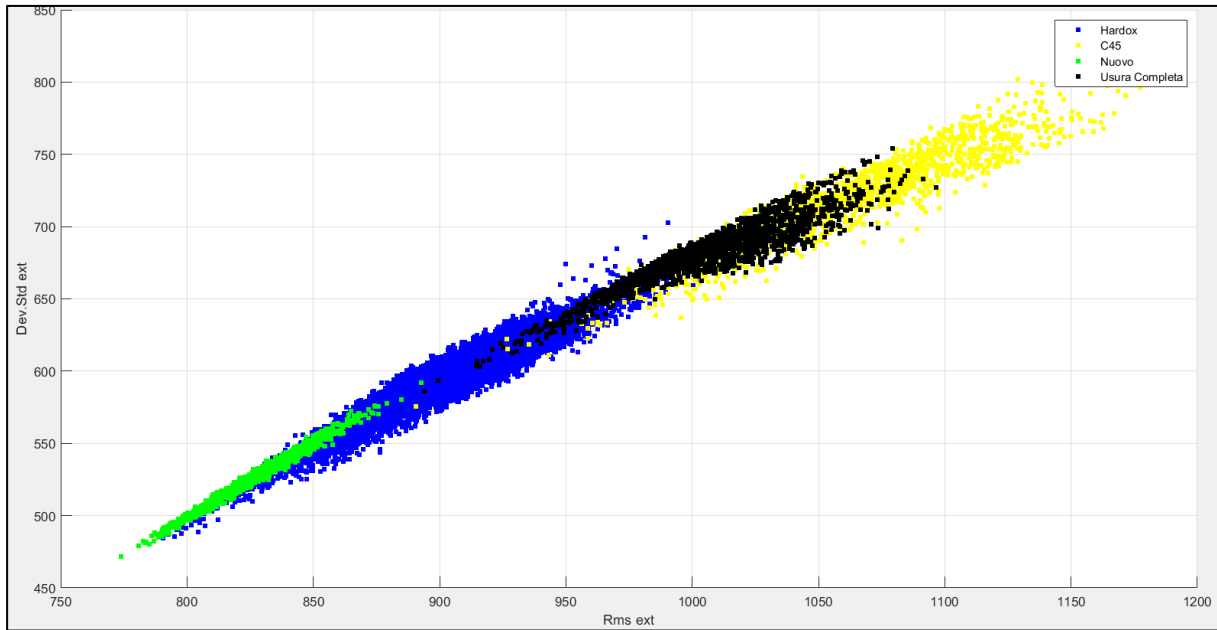


Figura 5.36 – Scatter plot rispetto al'Rms e alla Dev.Std relative all'estensimetro per Hardox, C45, coltello nuovo e usura completa

Quindi, per associare i dati di Hardox e C45 alle tre classi, è stata utilizzata la tecnica del Clustering k-means [12].

Il clustering è il processo di divisione dei dati in gruppi (cluster) in base alle loro caratteristiche. A differenza del Classificatore SVM, questa è una tecnica di apprendimento non supervisionato, in quanto ci sono solo le variabili indipendenti e nessun target. I cluster hanno le seguenti proprietà:

- tutti i punti in un cluster dovrebbero essere simili tra loro;
- i punti di diversi cluster dovrebbero essere più diversi possibili.

Il K-means è un algoritmo che riduce al minimo la somma delle distanze tra i punti e il rispettivo centroide (posizione media di tutti i punti) del cluster. L'algoritmo segue gli step riportati di seguito.

I. Dati N punti, scegliere il numero K di cluster.

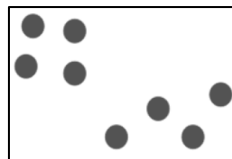


Figura 5.37 – N punti a cui applicare il clustering K-means [12]

II. Selezionare K punti casuali dai dati come centroidi.



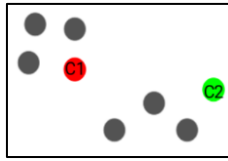


Figura 5.38 –  $K=2$ ,  $C1$  e  $C2$  rappresentano i centroidi [12]

III. Assegnare tutti i punti al centroide del cluster più vicino.

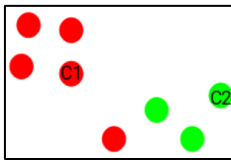


Figura 5.39 – I punti più vicini a  $C1$  sono assegnati al cluster rosso, mentre i punti più vicini a  $C2$  sono assegnati al cluster verde [12]

IV. Ricalcolare i centroidi dei cluster appena formati.

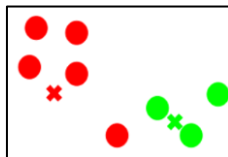


Figura 5.40 – Le croci rossa e verde sono i nuovi centroidi [12]

V. Ripetere gli step III e IV.

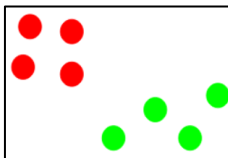


Figura 5.41 – Cluster finali trovati [12]

L'algoritmo termina quando uno dei seguenti tre criteri viene soddisfatto:

- i centroidi non cambiano;
- i punti rimangono nello stesso cluster;
- è stato raggiunto il numero massimo di iterazioni.

La tecnica, quindi, è stata applicata ai dati dell'Hardox e del C45, scegliendo  $K=3$ , ovvero le 3 classi di usura del coltello. Di seguito viene riportato lo scatter plot relativo al Rms e alla Deviazione Standard dell'estensimetro.

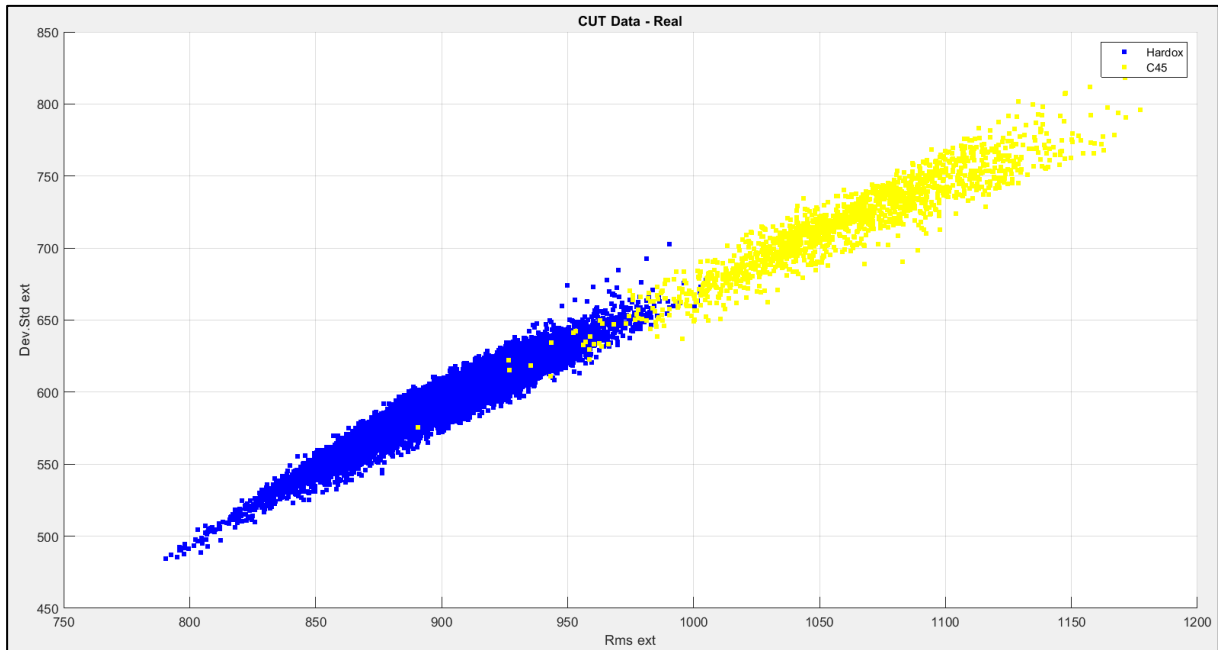


Figura 5.42 – Scatter plot rispetto all'Rms e alla Deviazione standard dell'estensimetro dei dati di Hardox e C45

L'obiettivo è dividere il dataset in tre cluster in modo da capire quali punti facciano riferimento al coltello nuovo, quali al semi-usurato e quali all'usurato. Perciò, applicando il Clustering K-means con K pari a 3, si ottiene il seguente grafico.

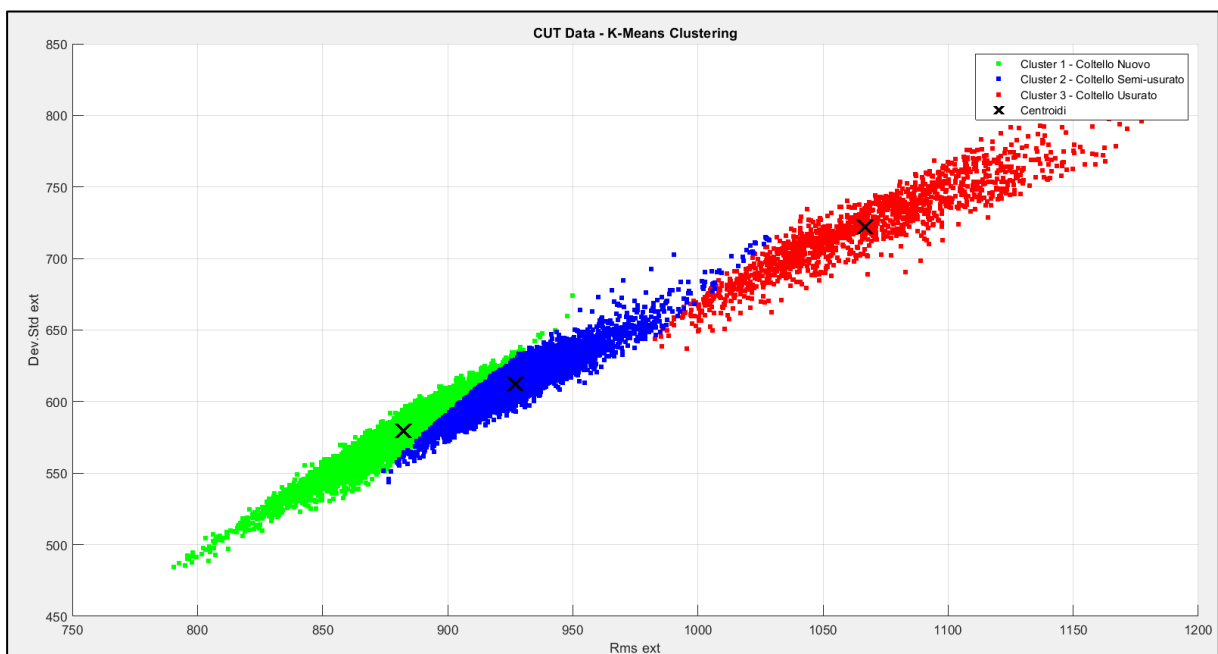


Figura 5.43 - Scatter plot rispetto all'Rms e alla Deviazione standard dell'estensimetro dei dati di Hardox e C45 clusterizzati nelle tre classi

A questo punto, è stato addestrato il classificatore con il seguente training-set:

- “0” = Coltello nuovo, associato alle features del coltello in assenza di usura e del cluster 1;
- “1” = Coltello semi-usurato, associato alle features dell’usura singola, dell’usura doppia e del cluster 2;
- “2” = Coltello usurato, associato alle features dell’usura completa e del cluster 3.

Gli step seguiti sono i sempre gli stessi.

I. Applicazione del Criterio One-Way ANOVA, che determina come features più significative:

1. Estensimetro - Area;
2. Estensimetro - Mean;
3. Corrente - Skewness;
4. Estensimetro - Rms;
5. Estensimetro - Kurtosis;
6. Estensimetro - Skewness;
7. Estensimetro - Deviazione Standard.

II. Realizzazione del classificatore con il ClassificationLearner; il classificatore con le migliori prestazioni (Accuracy = 94,3%) è il Fine Gaussian SVM, con la seguente matrice di confusione.

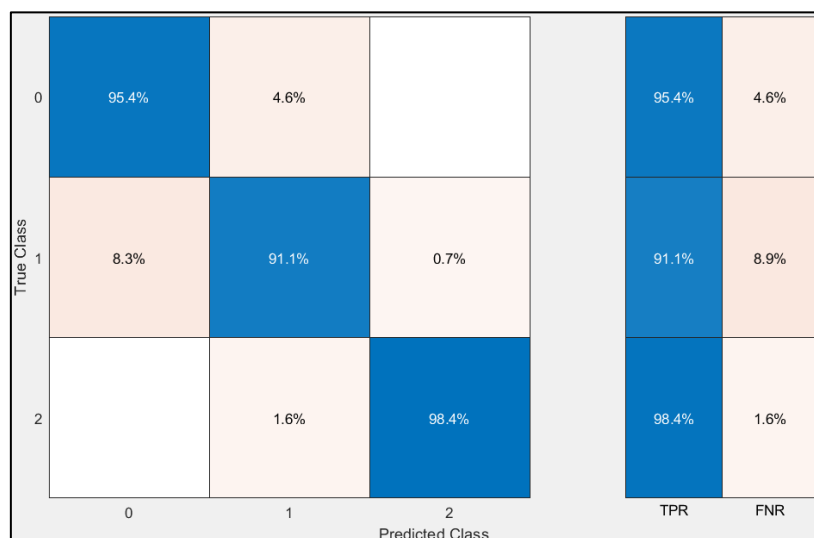


Figura 5.44 – Matrice di confusione del classificatore per il tondino da 12 mm, addestrato anche con i dati dell’Hardox e del C45

Il modello, quindi, riesce a classificare bene le varie condizioni di usura del coltello. Gli unici errori sono nella classificazione del coltello nuovo come coltello semi-usurato e viceversa: il

motivo è dovuto al fatto che i dati relativi alla singola e doppia usura (considerati come coltello semi-usurato) condividono caratteristiche con il coltello in assenza di usura.

Il classificatore appena realizzato rappresenta il classificatore finale della tipologia di usura sulla lama del coltello per il tondino da 12 mm.

## **5.2 Tondino da 16 mm**

Al fine di ottenere il classificatore per il caso di tondino da 16 mm sono stati seguiti gli stessi step della realizzazione dei classificatori per il tondino da 12 mm. Sono state effettuate, quindi, le acquisizioni con le seguenti condizioni:

- filo libero di muoversi nella canalina (il caso con filo fisso non è stato considerato per il motivo descritto in precedenza);
- velocità del motore al 100% della velocità massima (il caso con velocità al 30% non è stato considerato per il motivo descritto in precedenza);
- coltelli di materiale Unimax in assenza di usura e con usura manuale (singola, doppia e completa) e coltelli di materiale Hardox e C45.

Nei grafici di seguito indicati vengono mostrati gli andamenti della forza rilevata dall'estensimetro sulla biella e dell'intensità della corrente del motore. La velocità del motore, la posizione del motore e i segnali acquisiti dagli accelerometri non sono stati riportati, in quanto per essi valgono le stesse considerazioni fatte per il tondino da 12 mm, ovvero non sono significativi nell'individuazione della presenza dell'usura tanto quanto la forza sulla biella e l'intensità di corrente del motore.

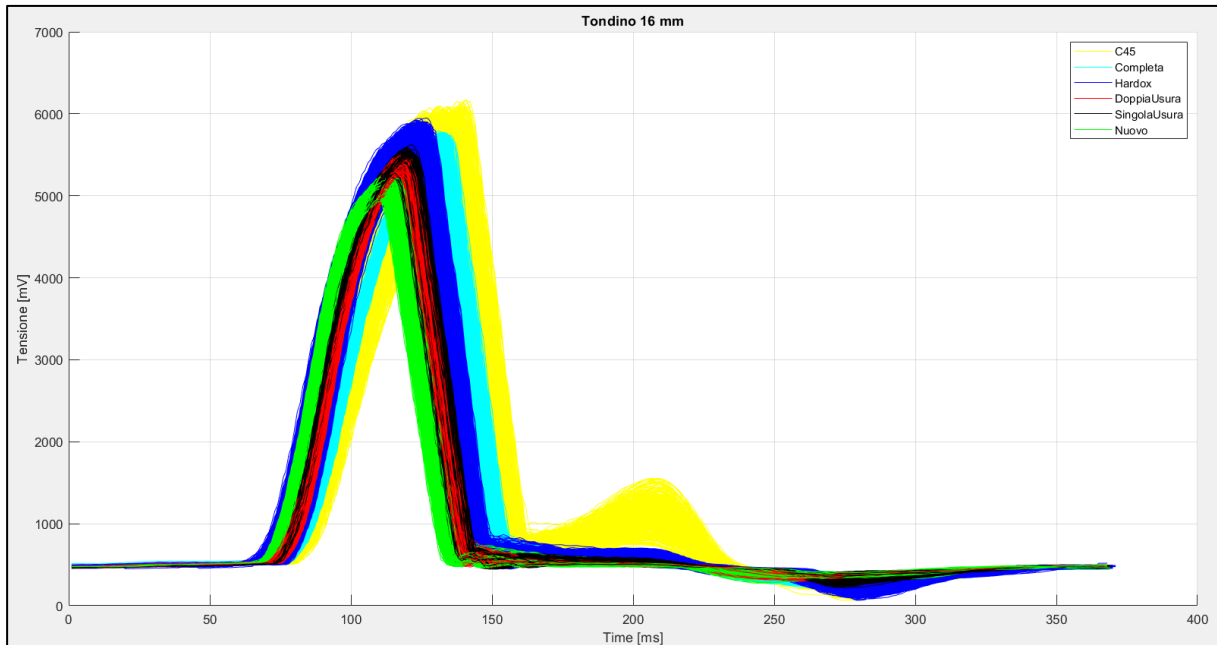


Figura 5.45 – Andamento della forza (proporzionale alla tensione) rilevata sulla biella dall'estensimetro per le diverse usure e per i diversi coltelli

Come già avviene per il tondino da 12 mm, all'aumentare dell'usura il picco trasla in avanti nel tempo e aumenta in ampiezza, ovvero la forza rilevata sulla biella è maggiore, in quanto il coltello necessita di più tempo per completare il taglio e richiede maggiore energia. Inoltre, per quanto riguarda il C45, si può osservare un secondo picco intorno ai 200 ms, dovuto al fatto che, per l'elevata presenza di usura, il tondino viene tagliato in due istanti diversi.

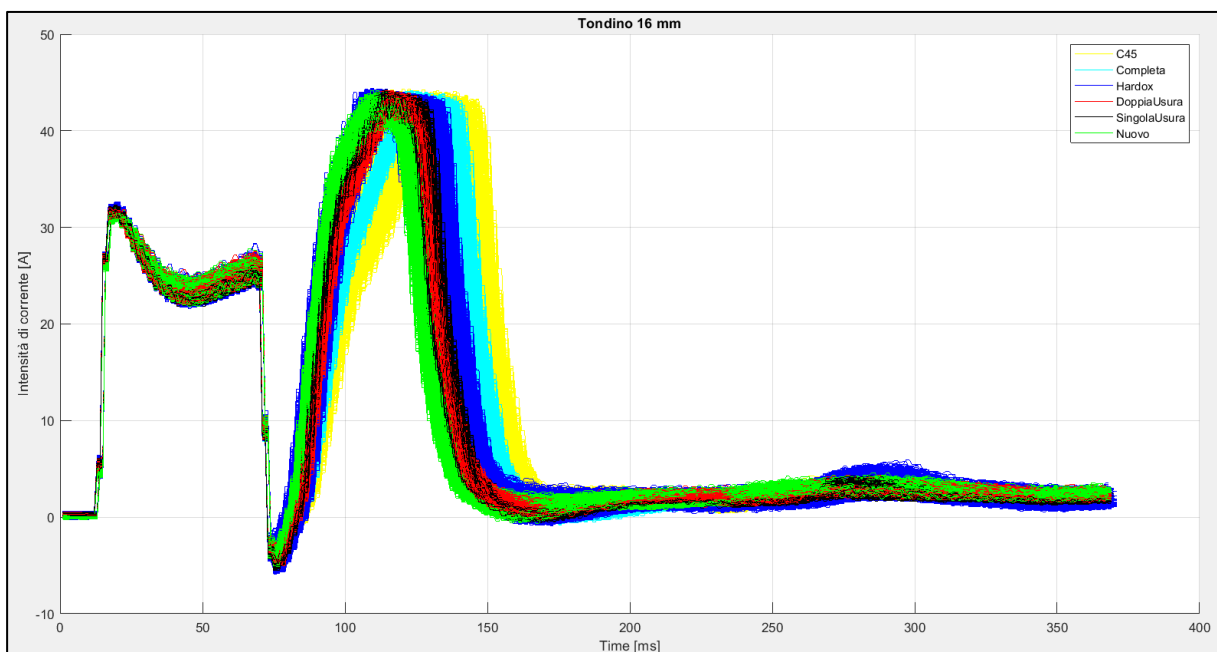


Figura 5.46 – Andamento dell'intensità di corrente del motore per le diverse usure e per i diversi coltelli

Come accade per l'estensimetro, il picco nell'intervallo in cui avviene il taglio del tondino (80 ms ÷ 170 ms) trasla in avanti nel tempo all'aumentare dell'usura del coltello; tuttavia, l'ampiezza del picco rimane costante per tutti i tipi di degrado della lama.

### 5.2.1 Estrapolazione e selezione features

Sono state, quindi, estrapolate le features per la forza e per la corrente.

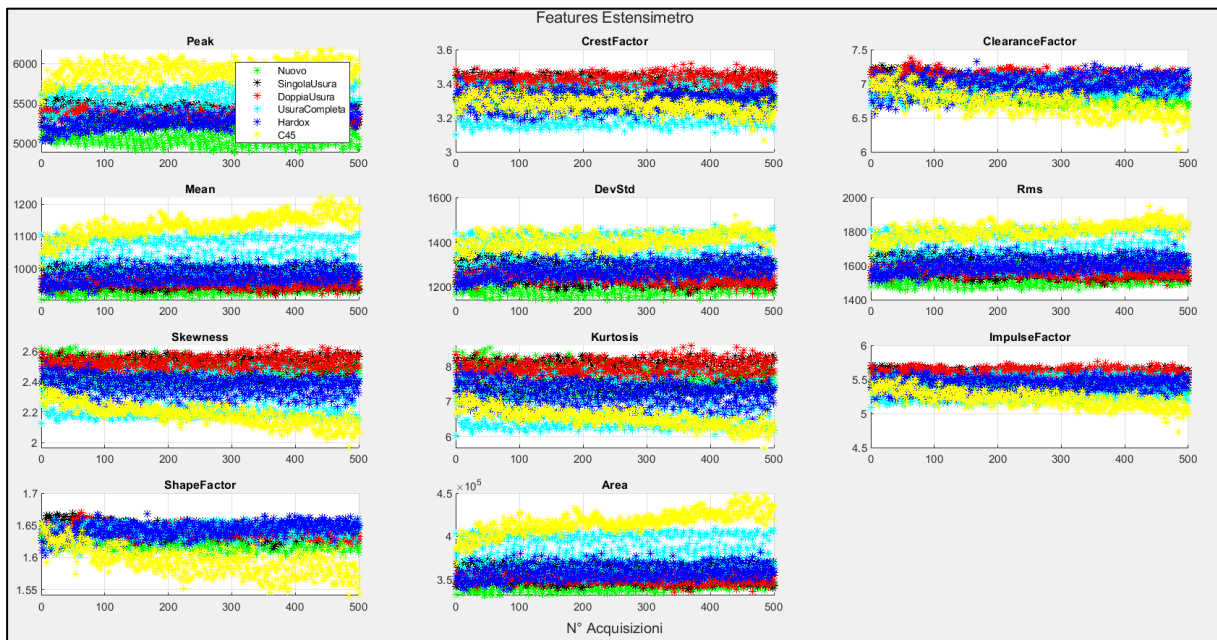


Figura 5.47 – Features dell'estensimetro per il tondino da 16 mm

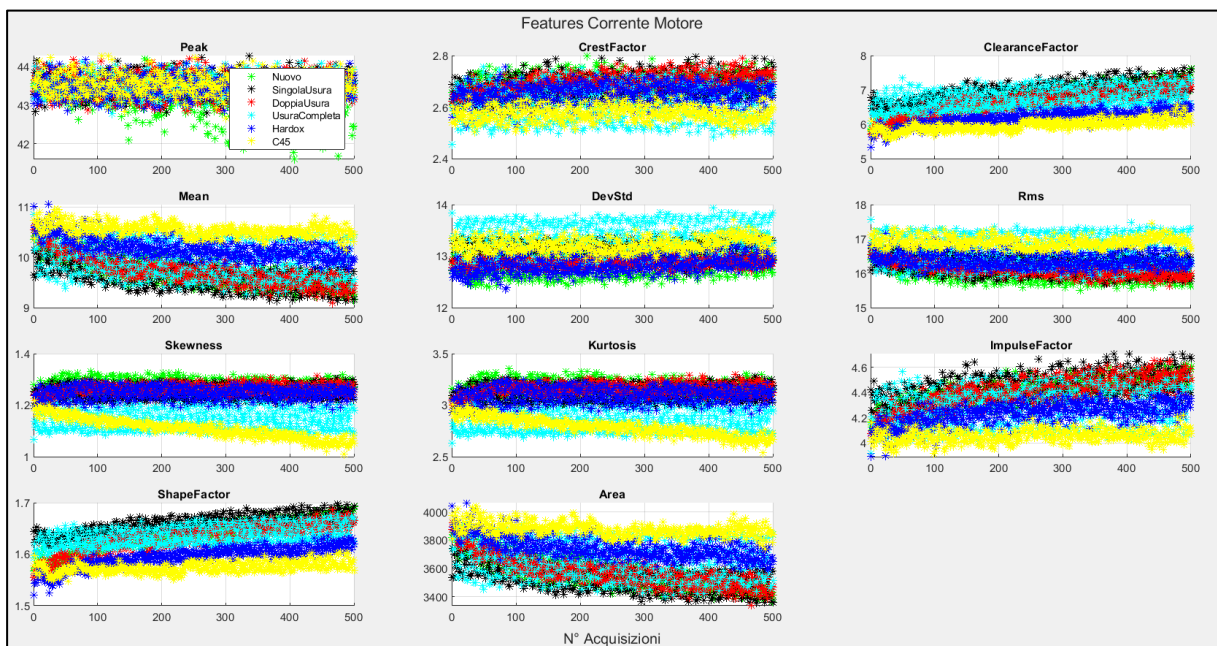


Figura 5.48 – Features dell'intensità di corrente del motore per il tondino da 16 mm

Osservando le figure, si possono notare delle differenze per le diverse usure e al fine di selezionare le più significative, anche in questo caso, è stato applicato il criterio One-Way ANOVA. In questo caso, sono state associate alla classe “Coltello semi-usurato” le features delle usure manuali e dell’Hardox: infatti, osservando lo scatter plot relativo al Rms e alla Deviazione Standard dell’estensimetro, si può notare come esse siano sovrapposte a differenza di quelle del C45.

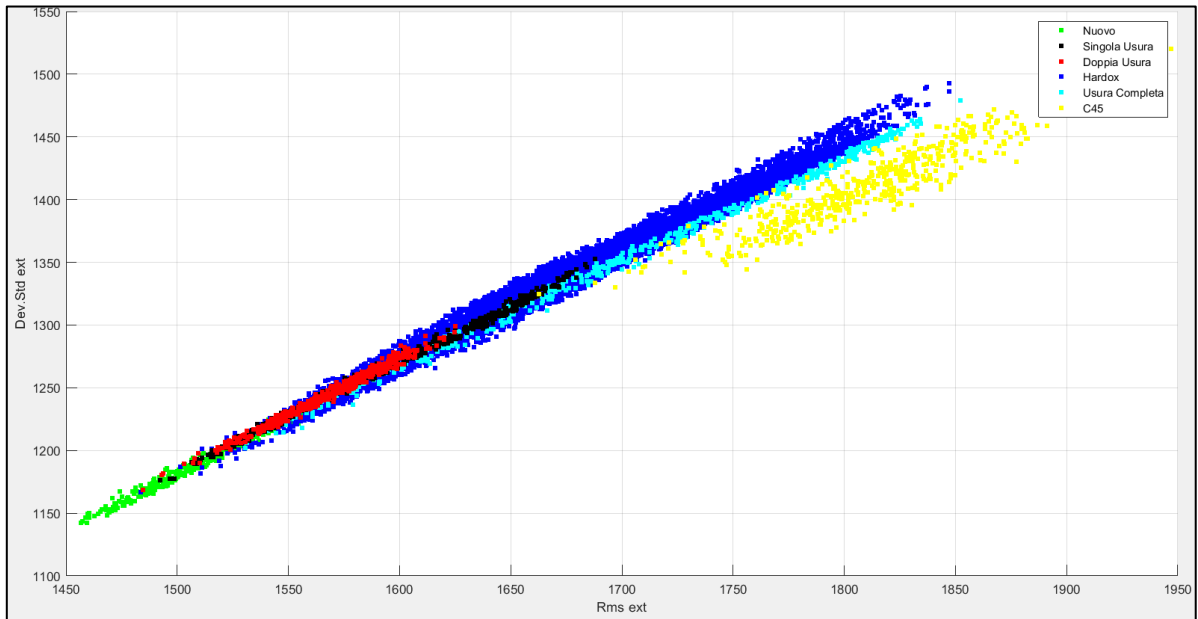


Figura 5.49 – Scatter plot rispetto all’Rms e alla Dev.Std dell’estensimetro per le diverse usure nel caso di tondino da 16 mm

Quindi, i risultati del criterio sono i seguenti.

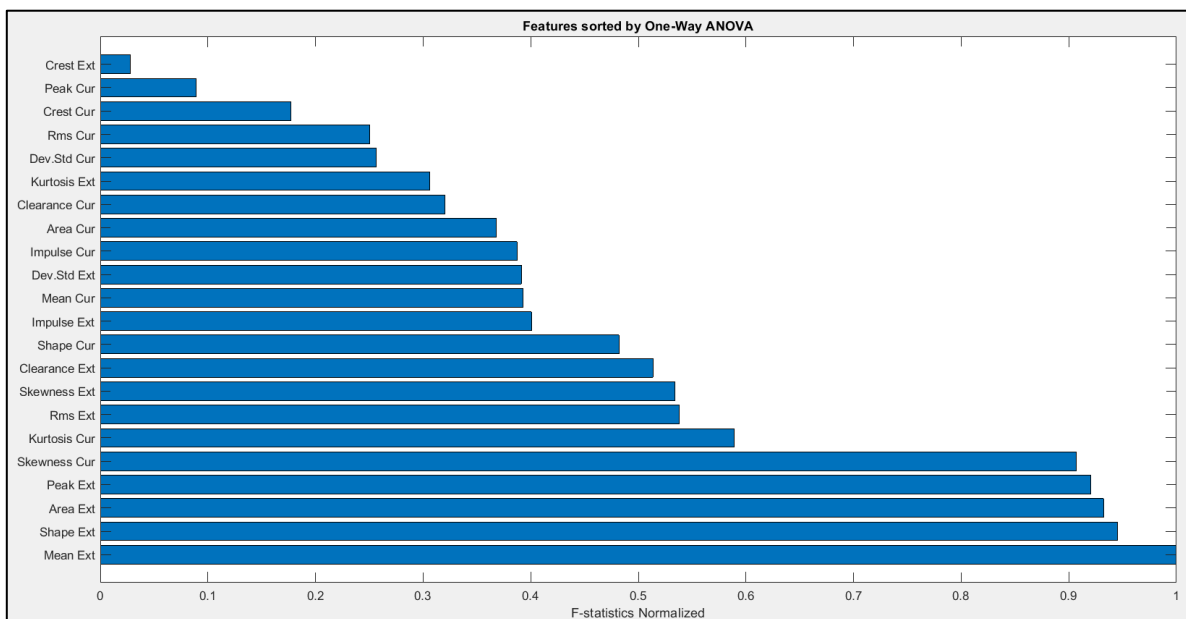


Figura 5.50 – Features normalizzate e ordinate secondo il parametro F dell’One-Way ANOVA

Perciò le features selezionate per addestrare il classificatore sono:

1. Estensimetro - Mean;
2. Estensimetro - Shape Factor;
3. Estensimetro - Area;
4. Estensimetro - Peak;
5. Corrente - Skewness;
6. Corrente - Kurtosis.

### 5.2.2 Realizzazione classificatore con *ClassificationLearner*

Il modello di classificazione con le prestazioni migliori (Accuracy = 99,2 %) è risultato essere il Cubic SVM, con la seguente matrice di confusione.

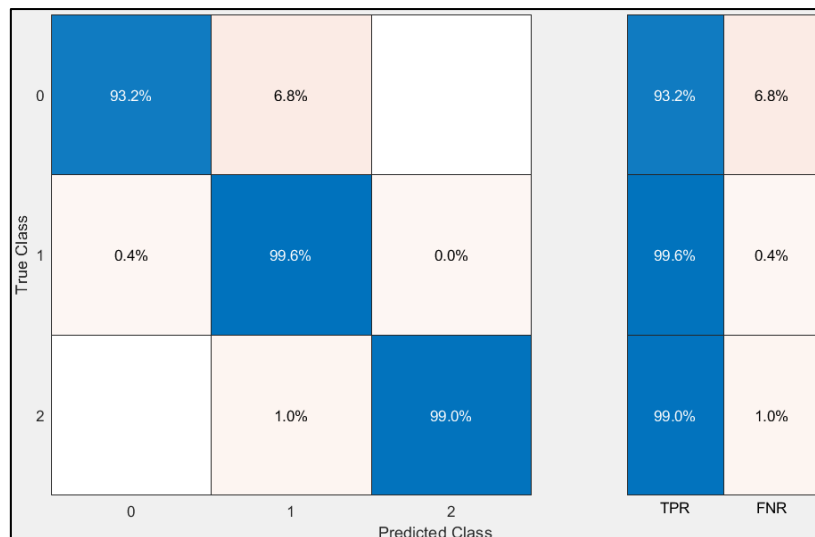


Figura 5.51 – Matrice di confusione del classificatore per il tondino da 16 mm

Il modello, quindi, riesce a classificare bene le varie condizioni di usura del coltello. Gli unici errori sono nella classificazione del coltello nuovo come coltello semi-usurato: il motivo è dovuto al fatto che i dati relativi alla singola e doppia usura (considerati come coltello semi-usurato) condividono caratteristiche con il coltello in assenza di usura.

### 5.3 Osservazioni sui risultati ottenuti

Sia per il tondino da 12 mm sia per quello da 16 mm, i modelli hanno ottenuto buoni risultati di classificazione: l'obiettivo che si cercava di raggiungere, infatti, non era avere prestazioni perfette (che, comunque sia, sono state quasi conseguite), ma avere dei modelli che non confondessero tra le situazioni estreme di coltello nuovo e coltello usurato.



Inoltre, nel caso specifico, all'azienda non interessa la classificazione della singola acquisizione, ma individuare in una serie di acquisizioni il trend delle classificazioni. Ad esempio, se in una serie acquisizioni, alcune di esse (in basso numero) vengono classificate come coltello semi-usurato, chi utilizza la macchina non deve preoccuparsi; se esse, invece, fossero in numero molto elevato, allora si potrebbe immaginare la presenza dell'usura sulla lama.

Tuttavia, per migliorare ulteriormente le prestazioni dei classificatori, sarebbe necessario avere a disposizione le acquisizioni del coltello Unimax con l'usura reale; infatti, questa condizione sarebbe quella vera per osservare il comportamento dei vari segnali acquisiti in presenza di usura.

## Capitolo 6

### Applicazione online dei classificatori

Una volta addestrati e testati, i classificatori sono stati utilizzati per l'individuazione dell'usura real-time. Mettendo in comunicazione Matlab e il PLC che gestisce la macchina, tramite cavo Ethernet, al termine di ogni ciclo di taglio, viene mostrato nel Command Window di Matlab il valore indicativo del tipo di usura presente nel coltello.

La comunicazione è stata resa possibile sfruttando l'ADS (Automation Device Specification), il protocollo di comunicazione di Twincat, che permette lo scambio di dati e il controllo dei sistemi Twincat tramite connessioni seriali o di rete [13]. Inoltre, esso consente:

- l'accesso ai task di I/O;
- il rilevamento dei cambiamenti di stato;
- accesso alle variabili per nome;
- comunicazione sincrona e asincrona;
- messaggi ciclici e basati su eventi.

L'algoritmo implementato in Matlab per la comunicazione con il PLC, basato sulla notifica di eventi dall'ADS, è il seguente [14].

- I. Importare in Matlab la libreria *TwincAT.Ads.dll*, scaricabile dal sito di Twincat, per la connessione con il protocollo ADS.

```
AdsAssembly =  
NET.addAssembly('C:\TwincAT\AdsApi\.NET\v4.0.30319\TwincAT.Ads.dll');  
import TwincAT.Ads.*;
```

- II. Realizzare la connessione con il PLC, tramite cavo ethernet, indicando l'indirizzo IP del PLC e numero della porta.

```
tcClient.Connect('192.168.6.20.1.1',851);
```

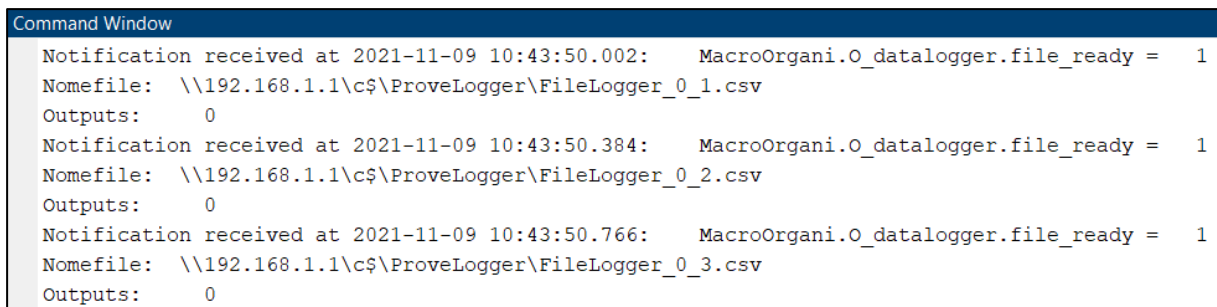
- III. Definire la variabile `MacroOrgani.O_datalogger.file_ready` da leggere dal PLC tramite notifica. Questa è una variabile booleana che viene messa a TRUE dal PLC quando è stato creato il file .csv, mentre viene messa a FALSE all'inizio del ciclo di taglio.

- IV. Se arriva la notifica dal PLC che `MacroOrgani.O_datalogger.file_ready` ha cambiato valore ed è pari a 1, leggere dal PLC i valori delle variabili `GVL.inc` e

GVL.group per aprire il file appena creato (il nome del file da aprire è FileLogger\_group\_inc.csv).

- V. Aprire il file con la funzione `readmatrix` e di esso considerare solo la parte Info contenente i valori forniti dai sensori, tralasciando l'Header.
- VI. Calcolare le features relative al file.
- VII. Mettere le features in input al classificatore.
- VIII. Attendere il valore indicativo del tipo di usura presente nel coltello, restituito dal classificatore nel Command Window.
- IX. Tornare al punto IV.

Di seguito viene riportato il Command Window durante l'applicazione dell'algoritmo.



```
Command Window
Notification received at 2021-11-09 10:43:50.002:   MacroOrgani.O_datalogger.file_ready = 1
Nomefile:   \\192.168.1.1\c$\ProveLogger\FileLogger_0_1.csv
Outputs:    0
Notification received at 2021-11-09 10:43:50.384:   MacroOrgani.O_datalogger.file_ready = 1
Nomefile:   \\192.168.1.1\c$\ProveLogger\FileLogger_0_2.csv
Outputs:    0
Notification received at 2021-11-09 10:43:50.766:   MacroOrgani.O_datalogger.file_ready = 1
Nomefile:   \\192.168.1.1\c$\ProveLogger\FileLogger_0_3.csv
Outputs:    0
```

Figura 6.1 – Command Window di Matlab, che al termine di ogni ciclo di taglio, restituisce il numero indicativo dell'usura del coltello

Quindi, ogni volta che arriva la notifica che `MacroOrgani.O_datalogger.file_ready` ha cambiato stato ed è pari a 1, viene mostrato in output: l'ora della notifica, il nome del file a cui si fa riferimento e il valore di usura restituito dal classificatore.

Tuttavia, dopo diverse applicazioni, l'algoritmo ha dato alcuni problemi, come file saltati e letture errate dei file, dovuti al fatto che le operazioni di apertura del file, salvataggio della parte "Info" con i valori dei segnali, calcolo delle features e uso del classificatore, che avvengono in Matlab, sono più lente rispetto al ciclo di taglio.

Per evitare tali problemi, l'algoritmo precedente è stato migliorato: le features sono calcolate direttamente dal PLC nella Function Block "FB\_dataLogger" e, successivamente, vengono lette da Matlab. Il nuovo algoritmo implementato in Matlab è il seguente.

- I. Importare in Matlab la libreria `TwinCAT.Ads.dll`, scaricabile dal sito di Twincat, per la connessione con il protocollo ADS.

- II. Realizzare la connessione con il PLC, tramite cavo ethernet, indicando l'indirizzo IP del PLC e numero della porta.
- III. Definire la variabile `MacroOrgani.O_datalogger.features_ready` da leggere dal PLC tramite notifica. Questa è una variabile booleana che viene messa a TRUE dal PLC quando sono state calcolate le features, mentre viene messa a FALSE all'inizio del ciclo di taglio. Il PLC al termine del ciclo di taglio calcola le features riempiendo un buffer `MacroOrgani.O_datalogger.buff_features`.
- IV. Se arriva la notifica dal PLC che `MacroOrgani.O_datalogger.features_ready` ha cambiato valore ed è pari a 1, leggere i valori delle features calcolate dal PLC dal buffer `MacroOrgani.O_datalogger.buff_features`.
- V. Mettere le features in input al classificatore.
- VI. Attendere il valore indicativo del tipo di usura presente nel coltello, restituito dal classificatore nel Command Window.
- VII. Tornare al punto IV.

Ciò che appare nel Command Window è simile a quello del primo algoritmo implementato, con la differenza che in questo caso viene mostrata solamente l'ora della notifica e l'output restituito dal classificatore.

Command Window		
Notification received at 2021-11-24 16:45:18.584:		Output = 0
Notification received at 2021-11-24 16:45:18.977:		Output = 0
Notification received at 2021-11-24 16:45:19.355:		Output = 0
Notification received at 2021-11-24 16:45:19.736:		Output = 0
Notification received at 2021-11-24 16:45:20.111:		Output = 0
Notification received at 2021-11-24 16:45:20.494:		Output = 1
Notification received at 2021-11-24 16:45:20.870:		Output = 0
Notification received at 2021-11-24 16:45:21.250:		Output = 0
Notification received at 2021-11-24 16:45:21.629:		Output = 0
Notification received at 2021-11-24 16:45:22.010:		Output = 0
Notification received at 2021-11-24 16:45:22.384:		Output = 0
Notification received at 2021-11-24 16:45:22.771:		Output = 0
Notification received at 2021-11-24 16:45:23.145:		Output = 0
Notification received at 2021-11-24 16:45:23.524:		Output = 1
Notification received at 2021-11-24 16:45:23.909:		Output = 0

Figura 6.2 – Command Window di Matlab relativo al secondo algoritmo implementato

## Capitolo 7

### Predizione della RUL

Nel seguente capitolo viene presa in considerazione la manutenzione predittiva, in particolare la previsione della vita utile rimanente (Remaining useful life, RUL), e vengono descritti due algoritmi, implementati in Matlab, per la predizione della vita utile rimanente del coltello che taglia il tondino sulla macchina COIL16. L'obiettivo è quello di comprendere il funzionamento di due tecniche di predizione, piuttosto che realizzare dei modelli da applicare subito al caso pratico, come effettuato per i modelli di classificazione.

La manutenzione viene definita dalla norma UNI EN 13306 come la *combinazione di tutte le azioni tecniche, amministrative e gestionali, durante il ciclo di vita di un'entità, volte a mantenerla o riportarla in uno stato in cui possa eseguire la funzione richiesta*.

La manutenzione riveste una funzione importante all'interno degli impianti produttivi in quanto garantisce la continuità degli stessi. Esistono diversi approcci alla manutenzione che nel tempo si sono evoluti, passando dalla manutenzione correttiva, alla manutenzione preventiva, fino alla manutenzione predittiva [15].

Tradizionalmente, infatti, il tipo di manutenzione impiegata è stata quella correttiva o a guasto, definita dalla norma UNI EN 13306 come la *manutenzione eseguita a seguito della rilevazione di un'avaria e volta a riportare un'entità nello stato in cui essa possa eseguire una funzione richiesta*. I manutentori, quindi, intervengono solo dopo che si è verificato un malfunzionamento.

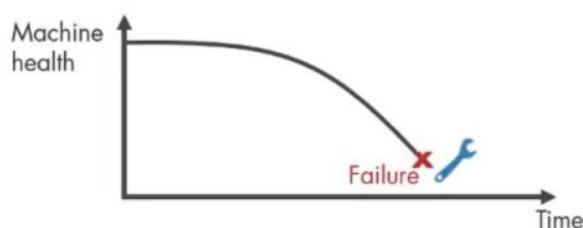


Figura 7.1 – Manutenzione correttiva [16]

Il vantaggio di questa tecnica è il dover sostituire solamente i pezzi guasti senza cambiare quelli non ancora arrivati a fine vita. Tuttavia, gli svantaggi sono in numero superiore e sono i seguenti:

- perdita di ricavi dovuti al fermo macchina per il guasto;

- imprevedibilità di intervento;
- elevati costi di riparazione, ovvero un guasto ad un componente può danneggiare a cascata altri componenti della macchina.

L'applicazione di questo approccio di manutenzione, perciò, risulta valido nel caso in cui le tipologie di guasto siano facilmente riparabili e il fermo macchina non comporta gravi danni alla produzione: è il caso in cui le macchine sono duplicate e possono essere facilmente sostituite da altre unità.

Un altro approccio è la manutenzione preventiva, definita dalla norma UNI EN 13306 come la *manutenzione eseguita ad intervalli pre-determinati o in base a criteri prescritti e volta a ridurre la probabilità del guasto o il degrado del funzionamento di un'entità*. Rispetto alla precedente, tende a intervenire prima del guasto, quando ancora il sistema ha prestazioni soddisfacenti. Essa è distinta nelle due seguenti categorie.

- La manutenzione programmata (preventiva ciclica) prevede interventi definiti a calendario, con lo scopo di sostituire un determinato componente della macchina, ancora perfettamente funzionante, con uno nuovo, in modo da prevenirne il guasto.

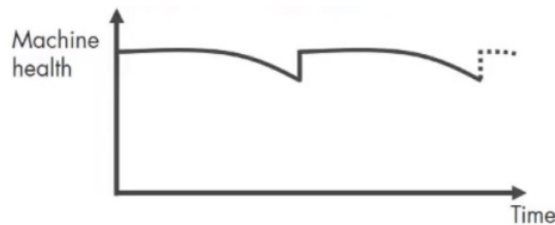


Figura 7.2 – Manutenzione programmata [16]

- La manutenzione su condizione (preventiva dinamica), che individua la necessità di sostituire un componente, ancora con prestazioni accettabili, sulla base dello stato di salute attuale, ovvero in seguito ad una campagna di rilievo periodico di parametri tecnici.

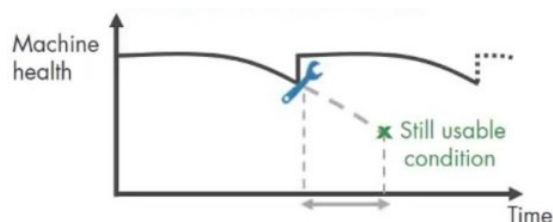


Figura 7.3 – Manutenzione su condizione [16]

Lo svantaggio della manutenzione preventiva il rischio è di intervenire su componenti che non hanno bisogno di essere cambiati o, al contrario, di non arrivare in tempo, perché il componente si è guastato prima del momento fissato per la manutenzione.

Infine, l'approccio più moderno è la manutenzione predittiva, definita dalla norma UNI ENI 13306 come la *manutenzione su condizione eseguita in seguito a una previsione derivata dall'analisi e dalla successiva valutazione dei parametri significativi afferenti il degrado dell'entità*. Essa è un'evoluzione della manutenzione su condizione, in cui il controllo dei parametri è continuo e non più periodico. Ciò permette di pianificare le manutenzioni future e prevedere la vita residua dei componenti, basandosi sullo stato attuale.

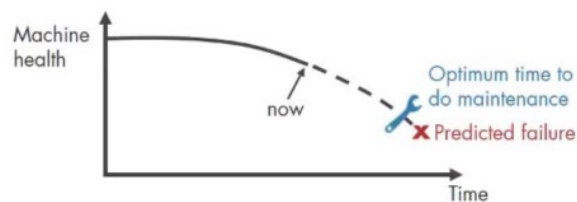


Figura 7.4 – Manutenzione predittiva [16]

La manutenzione predittiva, quindi, presenta i seguenti vantaggi:

- continuità di funzionamento dei processi produttivi;
- riduzione significativa delle ore di fermo, evitando interventi anticipati inutili;
- aumento della vita residua dei componenti;
- riduzione dei costi di manutenzione rispetto alle altre tecniche di manutenzione.

Nell'applicare la manutenzione predittiva è necessario seguire le seguenti fasi:

- I. acquisizione continua ed elaborazione dei segnali relativi al componente in esame, tramite l'utilizzo di sensori opportunamente scelti;
- II. estrazione delle features;
- III. addestramento del predittore della vita utile rimanente del componente, tramite tecniche di machine learning.

Nel presente lavoro di tesi sono stati implementati due algoritmi al fine di stimare la vita utile rimanente del componente coltello affetto da usura. La RUL è il tempo (in termini di ore, giorni, cicli di lavoro, ecc..) tra la condizione attuale del componente ed il guasto. Un modello per la stima della RUL riceve in input le features, estratte dai dati storici dei componenti, il comportamento delle quali cambia in modo prevedibile quando il sistema si degrada o opera in modalità diverse. Il metodo utilizzato dipende dal tipo di dati disponibili:

- i dati che indicano quanto tempo simili componenti hanno impiegato per raggiungere la condizione di guasto;
- il valore della soglia dell'indicatore di condizione che rileva il guasto;
- lo storico di componenti simili a quello che di cui si vuole stimare la RUL, dallo stato sano alla rottura.

Matlab, tramite il toolbox *Predictive Maintenance Toolbox*, propone i tre modelli di seguito riportati per la stima della RUL, ognuno relativo al tipo di dati [17, 18].

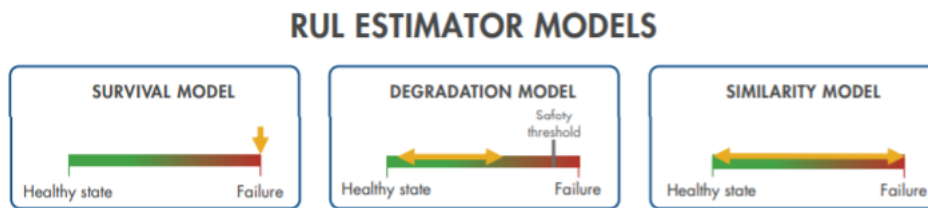


Figura 7.5 – Modelli per la stima della RUL [17]

- Survival model, quando i dati sono relativi solamente al momento del guasto dei componenti. Il metodo può determinare, così, quanti componenti si sono guastati dopo un certo numero di cicli e/o per quanti cicli sono stati in funzione. Per stimare la RUL del componente in esame, esso utilizza una distribuzione di probabilità di questi dati. Un semplice esempio è la stima del tempo di scarica di una batteria in base ai tempi di scarica passati e alle *covariate*, variabili come l'ambiente in cui la batteria ha operato (temperatura) e il carico su di essa. Il grafico in Figura 7.6 mostra la probabilità che una batteria si guasti in base a quanto tempo è stata in funzione: se è in funzione per 75 cicli, ha una probabilità del 90% di essere alla fine della sua vita utile.

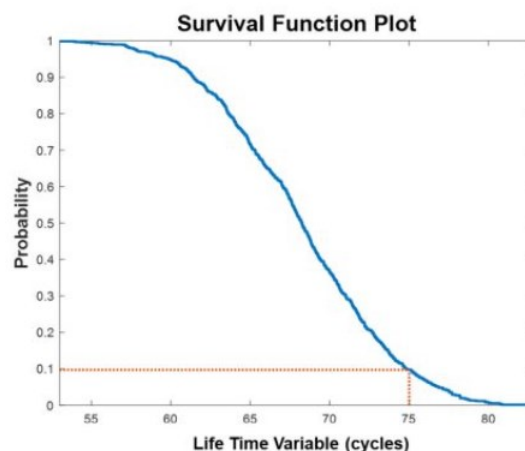


Figura 7.6 – Survival model: dopo 75 cicli, la probabilità che la batteria continui a funzionare è del 10% [18]



- Degradation model, quando si ha a disposizione dati tra lo stato sano e la rottura e, inoltre, è nota la soglia che non dovrebbe essere superata per evitare il guasto. Il metodo addestra il predittore con l'informazione sulla soglia e con lo storico dei dati a disposizione del componente in esame al fine di prevedere l'andamento dell'indicatore di condizione. In questo modo viene stimato la RUL, prevedendo quando l'indicatore di condizione supererà la soglia.

Ad esempio, la figura seguente mostra un degradation model che tiene traccia del guasto in un cuscinetto di una turbina eolica. Il modello prevede che il cuscinetto supererà il valore di soglia in circa 9,5 giorni.

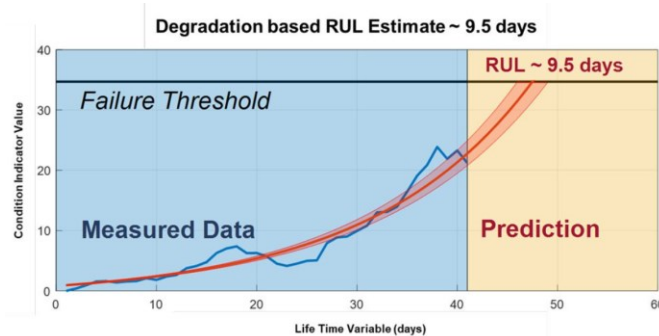


Figura 7.7 – Degradation Model: il cuscinetto ha una RUL stimata di 9,5 giorni in base ai dati sulle condizioni attuali (in blu) e al modello di degradazione (in rosso) [18]

- Similarity model, quando si dispone dello storico di componenti simili, dallo stato sano alla rottura. Il metodo, quindi, stima la RUL sulla base dei profili di degrado che maggiormente corrispondono con i dati del componente in esame.

Ad esempio, nella figura 7.8 i profili di degrado sono mostrati in blu, mentre in rosso è mostrato il dato corrente: sulla base del profilo che si avvicina più a quello del motore, si stima che la RUL sia di circa 65 cicli.

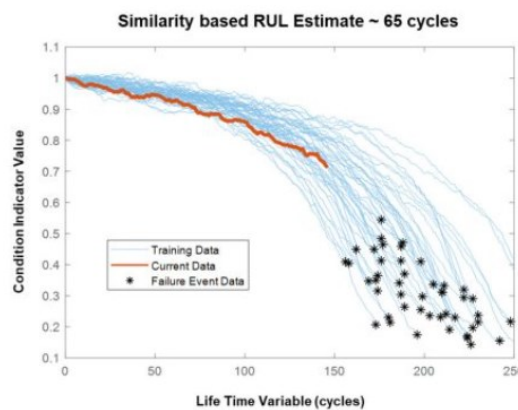


Figura 7.8 – Similarity Model: il dato attuale ha una RUL di 65 cicli sulla base dei profili di degradazione più vicini [18]

## 7.1 Algoritmi implementati per la predizione della RUL del coltello

Di seguito vengono descritti i due algoritmi implementati con due diverse tecniche, a seconda della tipologia dei dati a disposizione: il primo è basato sul Degradation Model utilizzando la funzione `predictRUL` proposta da Matlab, mentre il secondo è basato sul Similarity Model utilizzando la rete neurale LSTM. In entrambi i casi, sono stati considerati i dati relativi al tondino da 12 mm.

### 7.1.1 Algoritmo basato sul Degradation Model

Come spiegato in precedenza, il Degradation Model predice la vita utile rimanente di un componente sulla base della conoscenza della soglia dell'indicatore di condizione e sull'andamento dell'indicatore stesso [19].

Non avendo a disposizione l'andamento della vita del coltello Unimax, se non nella condizione di nuovo e di usure manuali, è stato ipotizzato che la vita del coltello fosse quella costituita dalla sequenza delle acquisizioni effettuate in condizioni di coltello nuovo, coltello a singola usura, coltello a doppia usura, coltello Hardox, coltello a usura completa e coltello C45. Considerando l'andamento filtrato delle features relative all'estensimetro e all'intensità della corrente del motore, si ottengono i seguenti grafici.

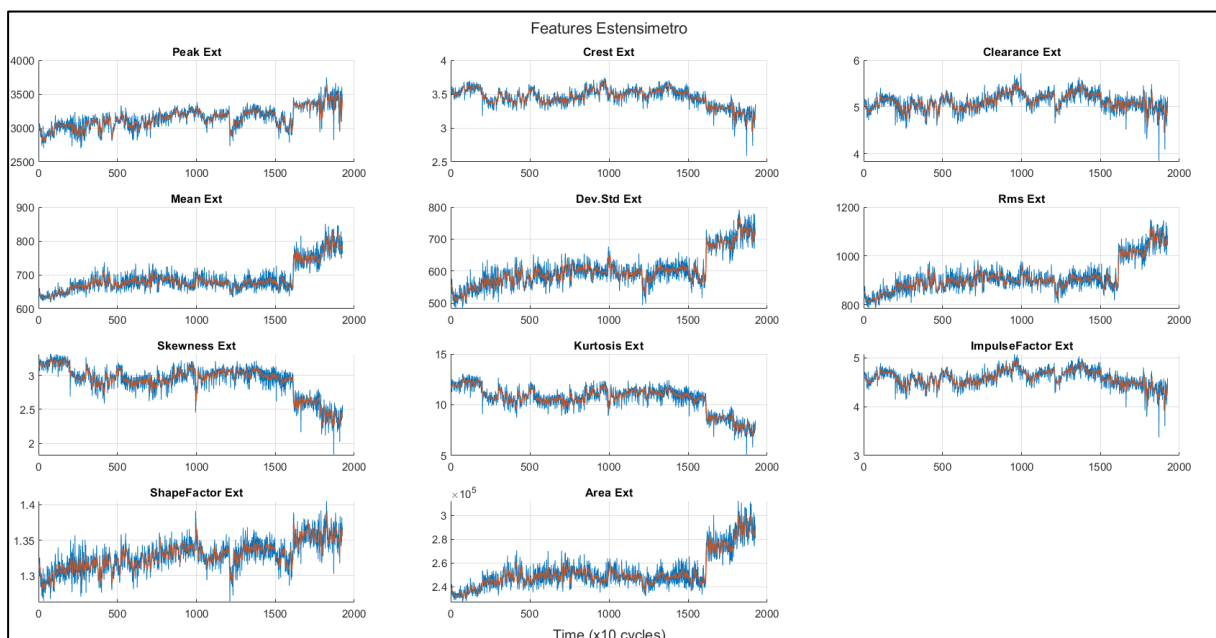


Figura 7.9 – Features filtrate della vita intera del coltello, relative all'estensimetro

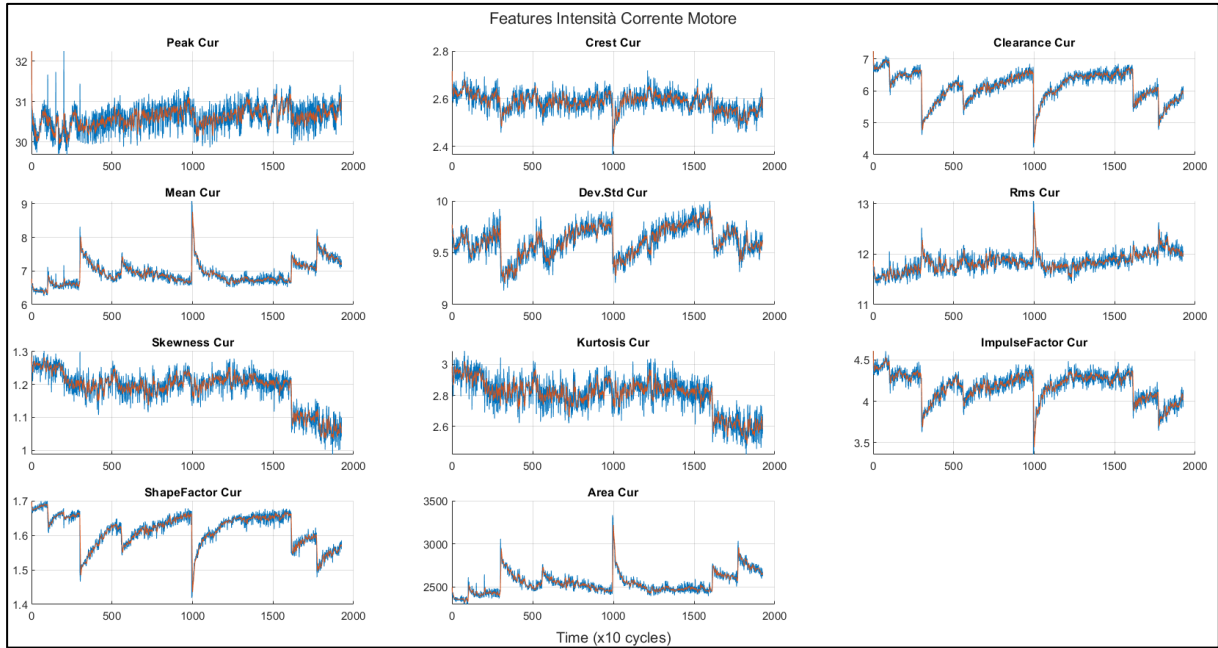


Figura 7.10 – Features filtrate della vita intera del coltello, relative all'intensità di corrente

Gli andamenti delle features riportati nelle figure 1.9 e 1.10 rappresentano la vita del coltello dalla condizione di sano fino alla condizione di usurato. Nelle ascisse sono riportati come unità di misura del tempo i cicli di lavoro, in cui il “x10” indica che viene considerata un'acquisizione ogni dieci cicli di taglio.

A questo punto, si supponga di trovarsi circa al 40% della vita del coltello e di voler stimare la RUL. A tale scopo, sono state ordinate le features secondo il livello di monotonicità sui dati acquisiti finora (ovvero quelli fino al 40%), tramite la funzione *monotonicity*, che quantifica la tendenza monotona delle features, mentre il sistema evolve verso il guasto: i valori restituiti variano da 0 (non monotono) a 1 (perfettamente monotono). Il metodo utilizzato è il “Metodo del coefficiente di correlazione per ranghi di Spearman” che calcola la monotonicità tramite la seguente formula [20]:

$$monotonicity = \frac{1}{M} \sum_{j=1}^M |corr(rank(x_j), rank(t_j))| = \frac{1}{M} \sum_{j=1}^M \left| 1 - \frac{6 \sum_i d_i^2}{N(N^2 - 1)} \right|$$

in cui  $M$  è il numero di sistemi monitorati,  $x_j$  è il vettore delle misure di una feature sul  $j$ -esimo sistema,  $t_j$  è il vettore temporale corrispondente al vettore  $x_j$ ,  $N$  è il numero di osservazioni e  $d_i = rank(x_j(i)) - rank(t_j(i))$  è la differenza dei ranghi della prima variabile e della seconda variabile della  $i$ -esima osservazione. Il metodo calcola la correlazione monotona tra le due variabili, anche se la loro relazione non è lineare.

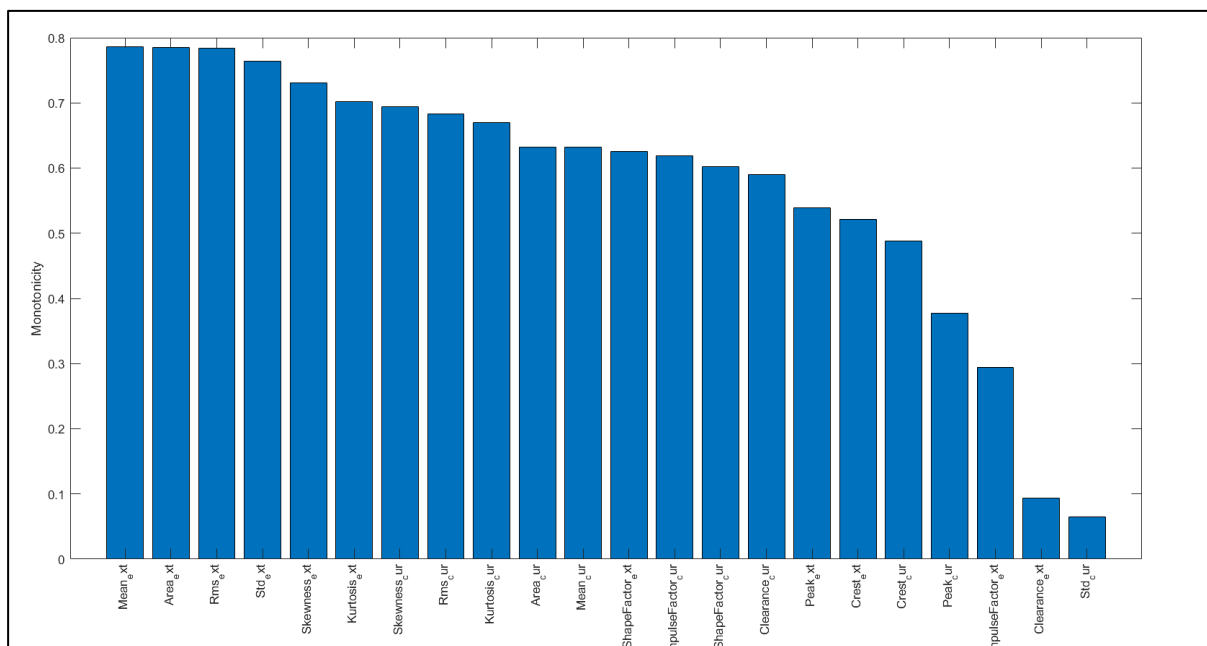


Figura 7.11 – Features ordinate in base al valore di monotonicità

I modelli di degradazione funzionano solo con un singolo indicatore di condizione. Tuttavia, è possibile utilizzare tecniche di fusione, come l'analisi dei componenti principali (PCA), per generare un indicatore di condizione che incorpori informazioni da più possibili indicatori. Perciò, sono state selezionate le features con un valore di monotonicità maggiore di 0.6, considerate le più significative, ovvero quelle che mostrano maggiormente un trend crescente o decrescente. Al fine di trovare l'indicatore di condizione fuso, alle features selezionate è stata prima applicata la normalizzazione, riportandole tutte nella stessa scala, e poi è stata applicata la PCA (Principle Component Analysis). L'idea base della tecnica è ridurre la dimensionalità delle variabili mediante la proiezione delle stesse su un numero limitato di componenti (dette componenti principali), preservando la variabilità (varianza) presente nel set di dati originali. In Matlab, la funzione `pca` restituisce i coefficienti delle componenti principali della matrice  $n$  (osservazioni)  $\times$   $p$  (variabili). La matrice dei coefficienti è  $p \times p$ , in cui le colonne sono in ordine decrescente di varianza. Nel caso in esame sono stati calcolati i coefficienti della PCA sui dati di training e, poi, sono stati applicati sull'intero dataset. Considerando le prime due componenti principali (quelle con maggior varianza), si ottiene il seguente grafico.

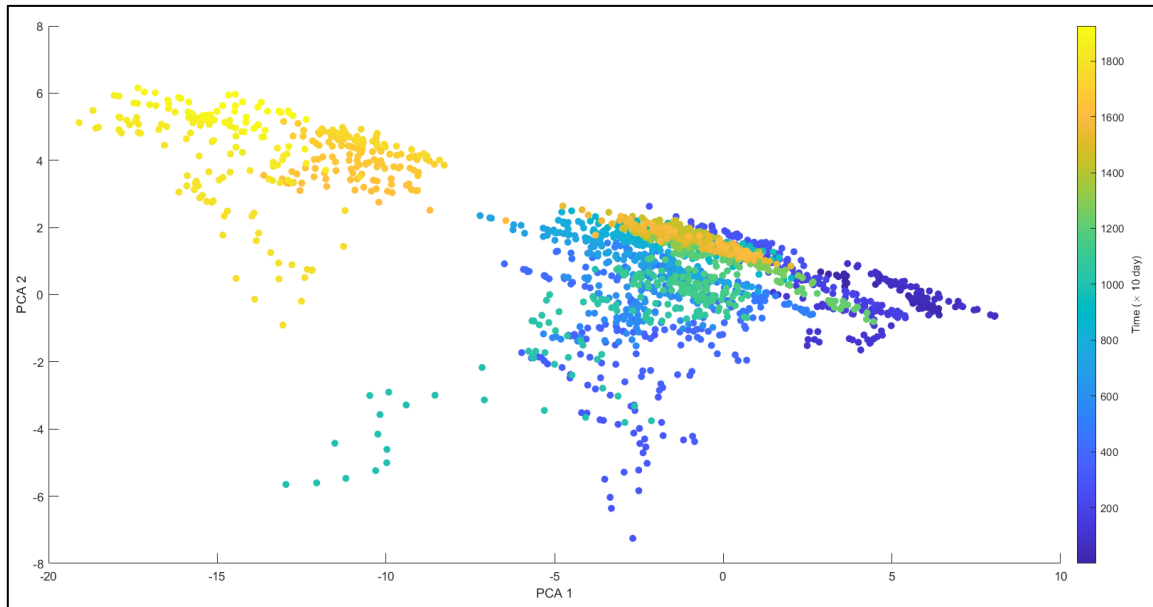


Figura 7.12 – Scatter plot relativo alle prime due componenti principali

Il grafico mostra che il primo componente (PCA1) diminuisce man mano che ci si avvicina al guasto, quindi esso può essere considerato come un promettente indicatore di condizione.

Nel grafico seguente viene mostrato l'andamento dell'indicatore di condizione scelto per tutta la vita del coltello. Come spiegato in precedenza, si supponga di essere nell'istante indicato dalla linea nera e di stimare il tempo di vita rimanente prima che l'indicatore raggiunga la soglia, calcolata come media degli ultimi 25 valori dell'HealthIndicator, in quanto relativi alla condizione di coltello usurato.

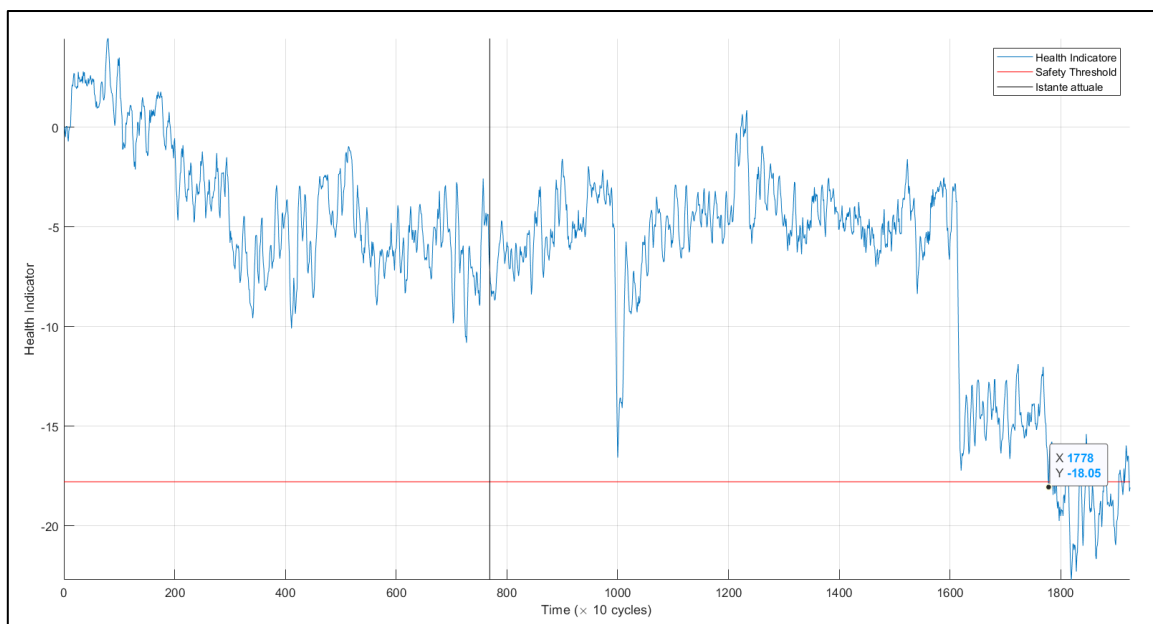


Figura 7.13 – Andamento dell'indicatore di condizione relativo alla vita intera del coltello; in rosso è riportata la soglia e in nero è indicato l'istante da cui predire la RUL

Dalla figura 7.13 si può osservare che il valore che dobbiamo predire è 1778, ovvero il ciclo di taglio in cui l'indicatore di condizione raggiunge la soglia.

A tale scopo, Matlab propone i due seguenti modelli di degradazione [21, 22].

- Il modello lineare descrive il comportamento di degradazione come un processo stocastico lineare e viene utilizzato quando il sistema non presenta un degrado cumulativo. Il modello implementa:

$$S(t) = \varphi + \theta(t) t + \varepsilon(t)$$

in cui:

- $\varphi$  è l'intercetta<sup>5</sup> del modello ed è costante;
  - $\theta(t)$  è la pendenza, modellata come una variabile causale con una distribuzione normale con media *Theta* e varianza *ThetaVariance*;
  - $\varepsilon(t)$  è il rumore additivo, modellato come una distribuzione normale con media zero e varianza *NoiseVariance*.
- Il modello esponenziale descrive il comportamento di degrado come un processo stocastico esponenziale e viene utilizzato quando il componente di test subisce un degrado cumulativo. Il modello implementa:

$$S(t) = \varphi + \theta(t)e^{(\beta(t)t + \varepsilon(t) - \frac{\sigma^2}{2})}$$

in cui:

- $\varphi$  è l'intercetta del modello, è costante ed è considerata come il limite inferiore o superiore della regione ammissibile di degradazione: se il segno di  $\theta$  è positivo allora  $\varphi$  è il limite inferiore, mentre se il segno di  $\theta$  è negativo allora  $\varphi$  è il superiore;
- $\theta(t)$  è una variabile random, modellata come una distribuzione lognormale con media *Theta* e varianza *ThetaVariance*;
- $\beta(t)$  è una variabile random, modellata come una distribuzione normale con media *Beta* e varianza *BetaVariance*;
- $\varepsilon(t)$  è il rumore additivo ed è modellato come una distribuzione normale con media zero e varianza *noisevariance*;
- $\sigma^2$  è uguale alla *noisevariance*.

---

<sup>5</sup> Valore della coordinata nel punto di intersezione tra la retta e l'asse relativo alla coordinata stessa.

Non essendo noto il comportamento effettivo dell'usura (se lineare o esponenziale), in quanto l'indicatore di condizione è stato considerato come la sequenza di usure diverse, sono stati applicati entrambi i modelli di degradazione. Perciò, sono stati creati i due modelli rispettivamente con le funzioni *linearDegradationModel* e *exponentialDegradationModel*.

I parametri utilizzati dai modelli non sono noti a priori, perciò vengono lasciati quelli di default. Essi poi vengono aggiornati automaticamente tramite la funzione `update` per tutta la durata dei dati acquisiti, quindi, in questo caso fino al 40% della vita del coltello. L'unico parametro noto è *Theta*: poiché l'indicatore di condizione ha un andamento decrescente, esso viene settato pari a -1.

La predizione della RUL secondo i due modelli è la seguente.

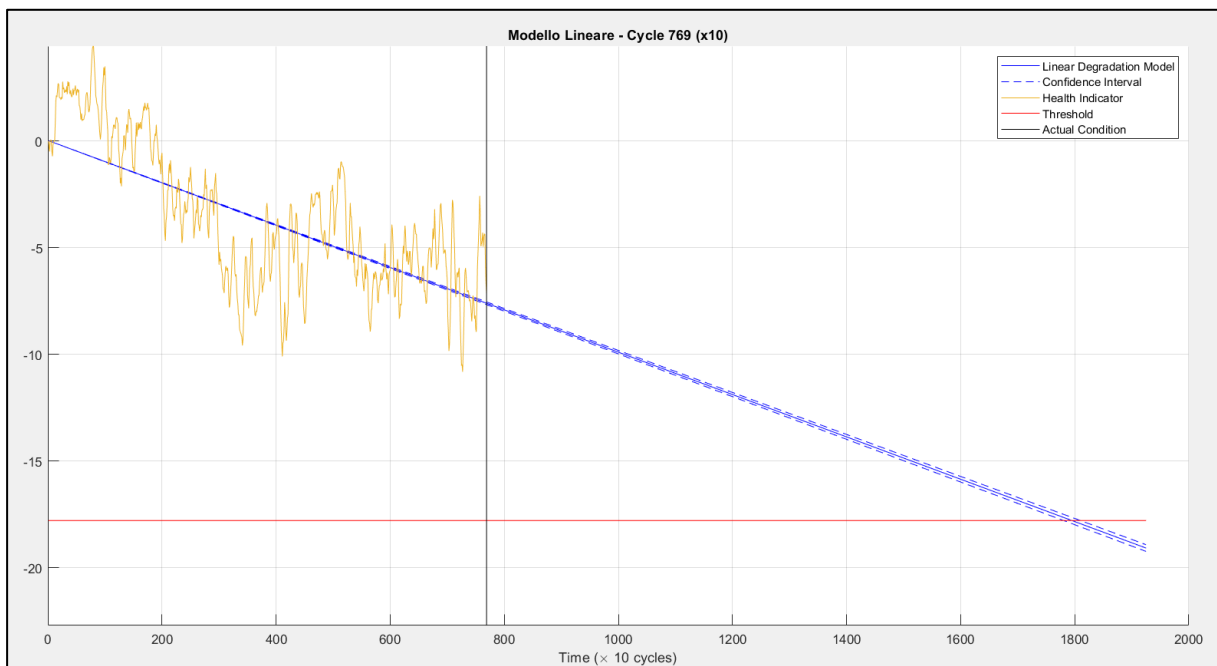


Figura 7.14 – Predizione della RUL con il linear degradation model

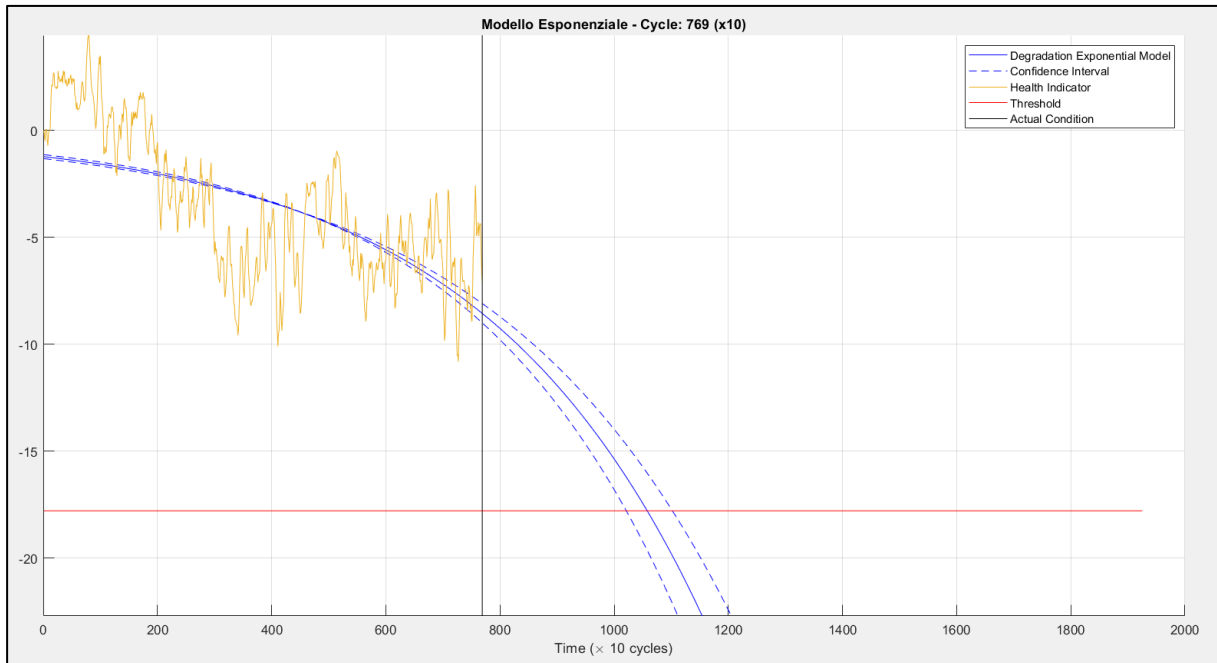


Figura 7.15 – Predizione RUL con l'exponential degradation model

I due modelli stimano due RUL diverse:

- il modello lineare predice 1053 (x10) cicli rimanenti, ovvero l'indicatore supererà la soglia al ciclo di taglio 1822 (x10);
- il modello esponenziale predice circa 356 (x10) cicli rimanenti, ovvero l'indicatore supererà la soglia al ciclo di taglio 1125 (x10).

Considerando che, secondo la vita reale totale considerata, il ciclo vero in cui il coltello raggiunge la situazione di allarme è il 1778, il modello lineare risulta molto più preciso rispetto a quello esponenziale.

Tuttavia, anche il modello lineare non è del tutto corretto, come mostra la seguente figura, in cui vengono confrontati le stime della RUL dall'inizio dei cicli fino all'istante attuale (quello considerato per ipotesi al 40%).



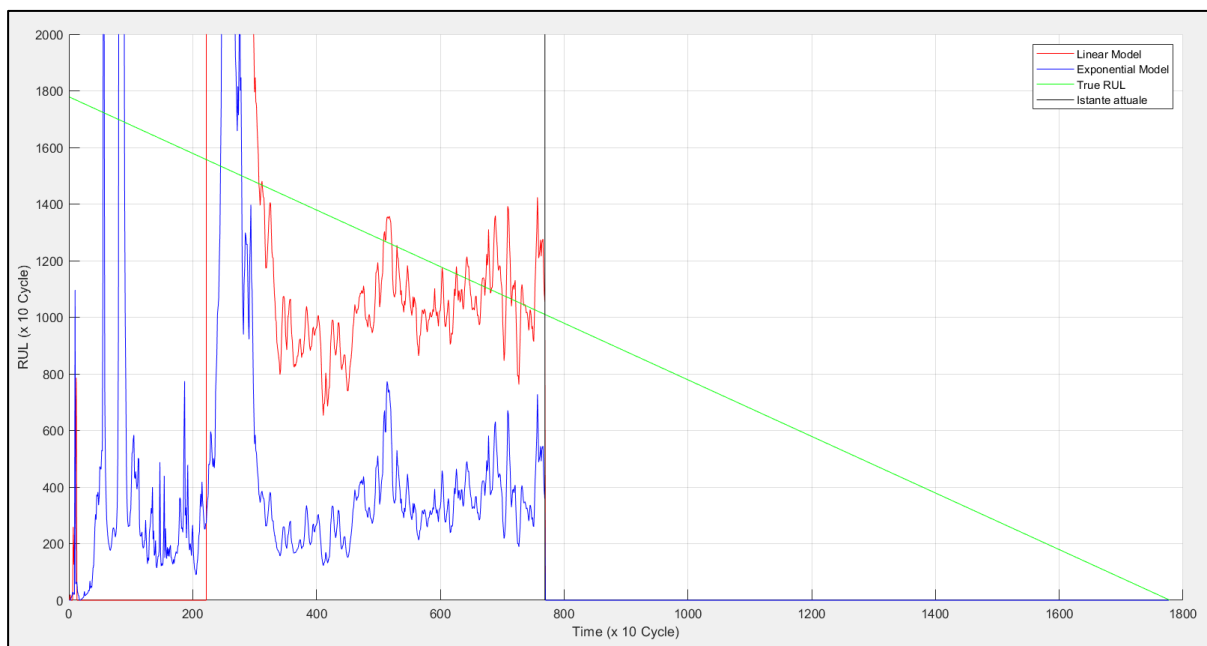


Figura 7.16 – Confronto tra le predizioni della RUL effettuate con il modello lineare (in rosso) e il modello esponenziale (in blu)

Il motivo di tali errori risiede nel fatto che l'indicatore di condizione in questione fa riferimento non al vero degrado, ma all'unione di diverse usure, come spiegato in precedenza. Per migliorare l'algoritmo si dovrebbe osservare l'andamento reale del coltello e avere a disposizione lo storico del degrado di ulteriori coltelli, al fine di addestrare il modello su parametri reali. Infatti, come visibile dalla figura 7.16, la stima della RUL risulta completamente errata all'inizio dei cicli, in quanto il modello non ha conoscenza del sistema in esame e i parametri impiegano diverso tempo prima di tendere ai valori più corretti.

### 7.1.2 Algoritmo basato sul Similarity Model con rete neurale LSTM

Per il secondo algoritmo è stato implementato il metodo del Similarity Model, non utilizzando la funzione apposita proposta da Matlab, ma il deep learning tramite la rete neurale LSTM (Long Short-Term Memory). Nel presente paragrafo viene introdotta la rete, tralasciando il dettaglio matematico, e viene descritto l'algoritmo.

In generale, le reti neurali artificiali (ANN, dall'inglese Artificial Neural Net, o semplicemente NN) sono modelli di calcolo, ispirati al funzionamento delle reti neurali biologiche, ovvero modelli costituiti da unità, chiamate neuroni, distribuite in più livelli e collegate tra loro da connessioni pesate, come indica di seguito la struttura tipica di una rete neurale semplice [23, 24].

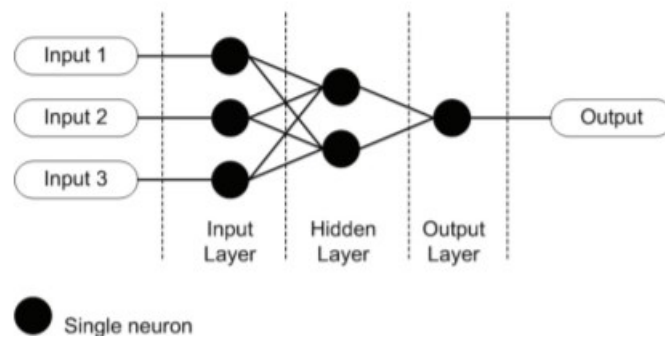


Figura 7.17 – Rete neurale artificiale semplice [23]

Come visibile dalla figura, le reti neurali sono costituite da almeno tre strati (layer):

- un input layer, con i dati di ingresso;
- uno o più hidden layer, dove avviene l’elaborazione;
- un output layer, con il risultato finale.

Ad ogni nodo viene applicata una funzione di attivazione che elabora il valore in ingresso attivandosi o rimanendo inattivo a seconda che venga superata (o meno) la soglia di attivazione prima di passarlo al nodo successivo.

Alla base della rete c’è la regolarizzazione dei pesi delle connessioni al fine di ottenere il giusto output a partire dagli input, effettuata tramite machine learning con la tecnica della “discesa del gradiente”. Infatti, inizialmente, i pesi hanno valori casuali e le prime risposte della rete sono errate; viene, perciò, calcolata una funzione di costo, che rappresenta l’errore (differenza tra output e valore atteso) quadratico medio di tutti gli output, da minimizzare con la discesa del gradiente. La tecnica, tramite iterazioni successive, parte da un punto casuale per avvicinarsi il più possibile al minimo della funzione.

Le reti neurali si dividono in feedforward, in cui i neuroni sono collegati solo agli strati successivi, e reti RNN (Recurrent Neural Network), in cui i neuroni sono collegati anche a strati precedenti o allo stesso livello [25]. Quest’ultima caratteristica introduce il concetto di memoria di una rete, in quanto l’output di un neurone può influenzare se stesso in uno step temporale successivo o i neuroni di un layer precedente.

La rete LSTM è una tipologia di rete RNN, capace di apprendere da lunghe sequenze temporali e conservarne la memoria, consentendo di monitorare lo storico del processo di degradazione di un sistema e predire correttamente la RUL. Alla base della LSTM c’è il concetto di cella, che trasporta le informazioni rilevanti, anche quelle provenienti da strati temporali precedenti. Man

mano che lo stato della cella procede, le informazioni vengono aggiunte o rimosse dalle porte (gate), che decidono, tramite apprendimento, quali informazioni è importante conservare o dimenticare. La funzione di applicazione utilizzata dai gate è la sigmoidea, che schiaccia i valori tra 0 (informazione da dimenticare) e 1 (informazione da conservare) [26].

Di seguito viene mostrato l'elaborazione delle informazioni all'interno della cella.

- I. Il forget gate decide quali informazioni devono essere dimenticate o conservate: le informazioni dallo stato nascosto precedente  $h_{i-1}$  e le informazioni dall'input corrente  $x_i$  vengono passate attraverso la funzione sigmoide.

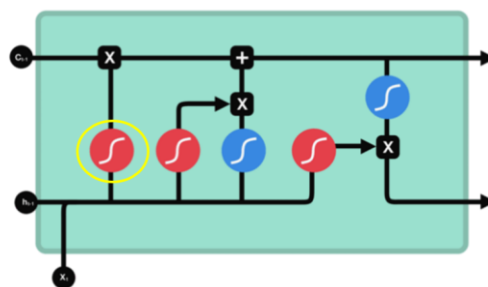


Figura 7.18 – Il forget gate (cerchiato in giallo) [26]

- II. L'input gate decide quali valori aggiornare, passando  $h_{i-1}$  e  $x_i$  in una funzione sigmoide. Essi vengono passati anche alla funzione tanh (come la sigmoide, con la differenza che i valori variano tra -1 e 1) per regolare la rete. Infine, vengono moltiplicate le uscite delle due funzioni di applicazione: l'output della sigmoide decide quali informazioni è importante mantenere dall'output tanh.

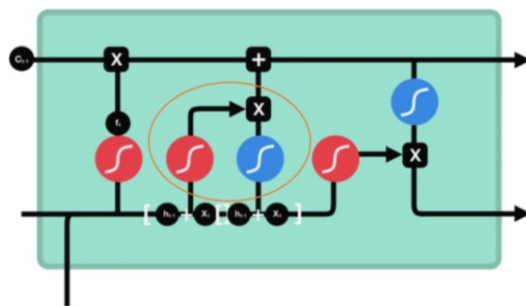


Figura 7.19 – L'input gate (cerchiato in arancione) [26]

- III. Lo stato precedente della cella  $c_{i-1}$  viene moltiplicato per l'output del forget gate  $f_i$  al fine di diminuire l'importanza dei valori vicini a 0. Il risultato viene sommato con l'output dell'input gate, che aggiorna lo stato della cella al nuovo valore  $c_i$ .

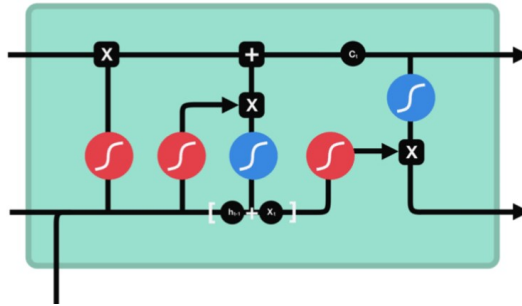


Figura 7.20 – Aggiornamento dello stato della cella [26]

- IV. L'output gate decide quale dovrebbe essere il successivo stato nascosto  $h_i$ . Per prima cosa, vengono passati  $h_{i-1}$  e  $x_i$  in una funzione sigmoide e  $c_i$  alla funzione tanh. Successivamente, viene ottenuto  $h_i$ , moltiplicando i valori restituiti dalle due funzioni di applicazione. Il nuovo stato della cella  $c_i$  e il nuovo stato nascosto  $h_i$  vengono quindi riportati al passaggio temporale successivo.

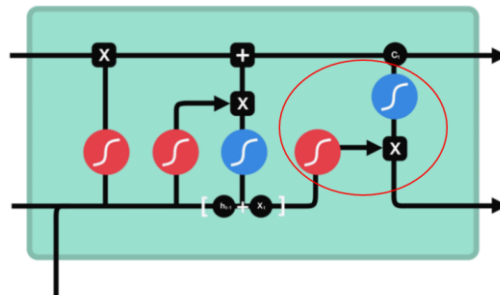


Figura 7.21 – L'output gate (cerchiato in rosso) [26]

Per lo sviluppo dell'algoritmo, tutte le considerazioni appena effettuate sulla rete vengono svolte in automatico da Matlab. La rete viene ottenuta settando i parametri relativi e addestrandola sul training set.

Come per il primo algoritmo, anche in questo caso è stata considerata come vita intera del coltello quella costituita dalla sequenza delle acquisizioni effettuate in condizioni di coltello nuovo, coltello a singola usura, coltello a doppia usura, coltello Hardox, coltello a usura completa e coltello C45. Successivamente sono state considerate solamente le features con il trend monotono maggiore, tramite la funzione `monotonicity` con il "Metodo del coefficiente di correlazione di Spearman", selezionando quelle con valori maggiori di 0.5, considerate le più significative, ovvero quelle che mostrano maggiormente un trend crescente o decrescente.

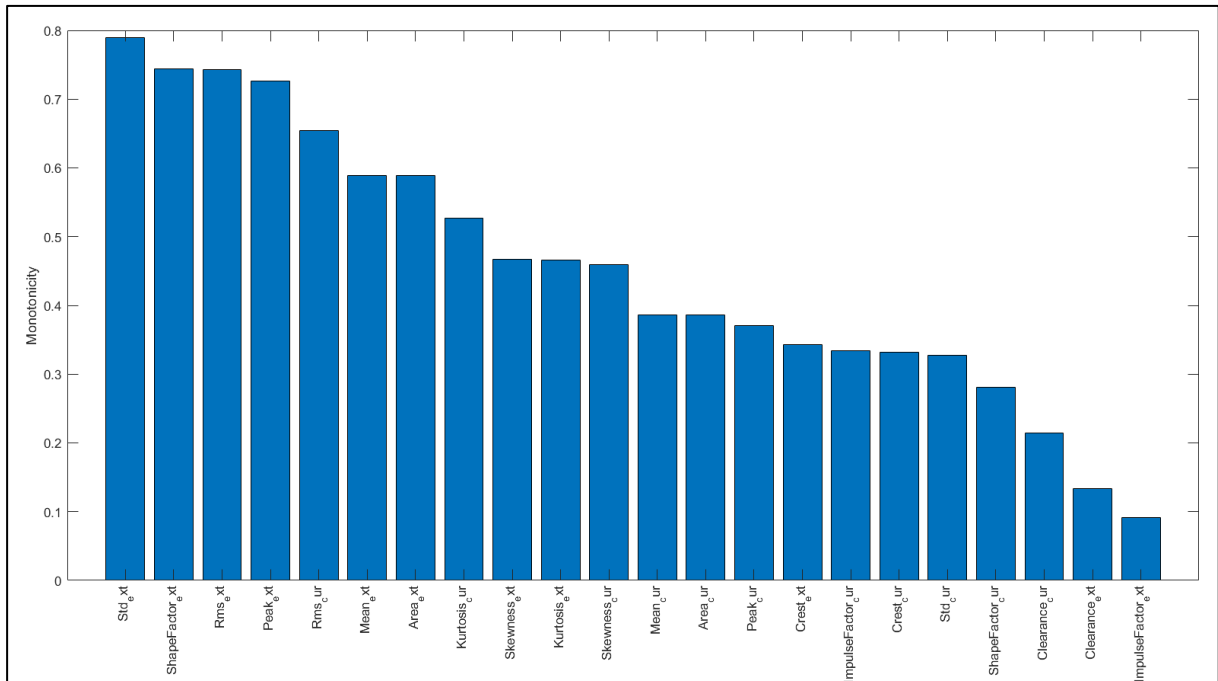


Figura 7.22 – Features ordinate in base al valore di monotonicità

A questo punto sono stati preparati i dati per il training della rete. Poiché non era disponibile lo storico di componenti simili a quello in esame, è stato diviso il dataset iniziale in 25 gruppi, ognuno costituito considerando un elemento ogni 25, cioè:

Dataset iniziale = [coltello nuovo, singola usura, doppia usura, Hardox, usura completa, C45] = 19245 acquisizioni

Gruppo 1 = Dataset iniziale (1, 26, 51,...)

Gruppo 2 = Dataset iniziale (2, 27, 52,...)

...

Gruppo 25 = Dataset iniziale (25, 50, 75,...)

Il gruppo 1 è stato considerato come test-set, mentre gli altri 24 gruppi come training-set per addestrare la rete; ovvero è stato ipotizzato che ogni gruppo costituisse lo storico, considerando un'acquisizione ogni 25 cicli di taglio, di un coltello dallo stato sano alla condizione di guasto/usurato.

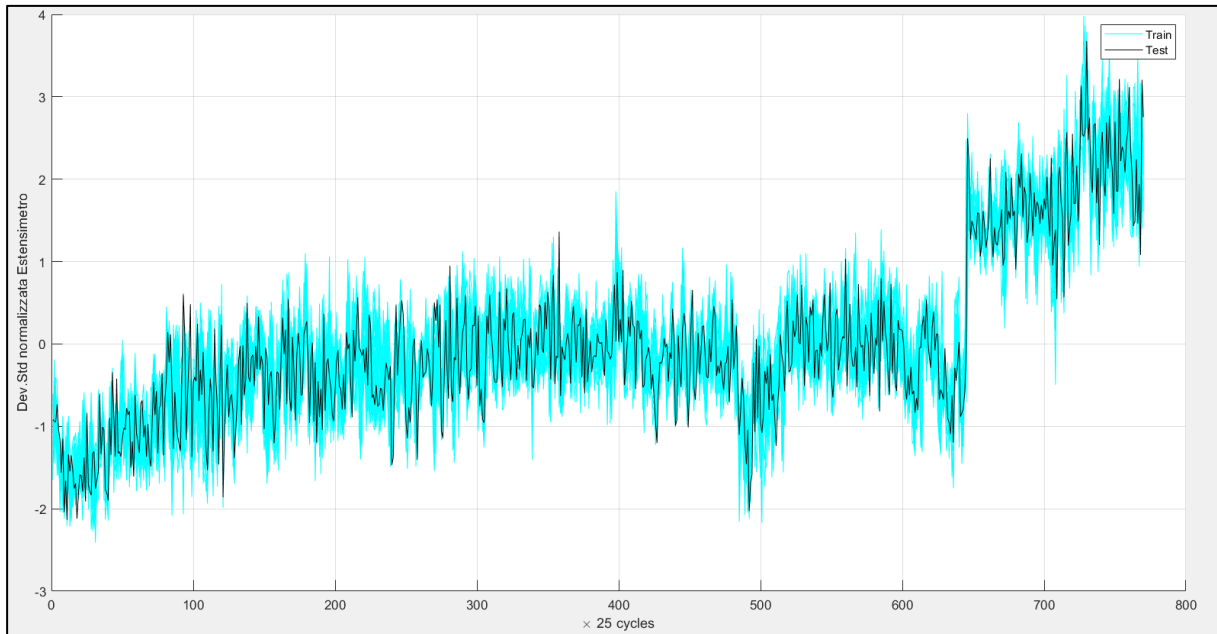


Figura 7.23 – Andamento dei dati di train (in azzurro) e dei dati di test (in nero)

Ottenuto il training-set, è stata definita la rete LSTM avente 24 come dimensione di dati nel layer di input (ovvero i 24 gruppi di training), 1 come dimensione del layer di uscita e i seguenti parametri [27].

- Numero di epoche = 150, massimo numero di epoche da utilizzare per l'addestramento. Un'epoca è il passaggio completo dell'intero dataset attraverso la rete. I numerosi passaggi sono finalizzati ad aggiornare i pesi e migliorare le prestazioni dell'algoritmo. Tuttavia, nella scelta del numero di epoche bisogna considerare che un numero troppo basso conduce all'*underfitting*<sup>6</sup>, mentre un numero troppo alto conduce all'*overfitting*. Inoltre, l'intero dataset ha una dimensione eccessiva, quindi esso viene diviso in parti (*batch*) di dimensione minore.
- Dimensione dei batch = 4.
- Initial Learn Rate = 0.1, tasso di apprendimento iniziale utilizzato per l'addestramento. Se esso è troppo basso, l'addestramento richiede molto tempo, se troppo alto potrebbe raggiungere un risultato non ottimale.

Ad esempio, nel caso in esame, ci sono 24 gruppi di training da dividere in batch di dimensione pari a 4; quindi, ci vorranno 6 iterazioni per completare un'epoca. Considerando, che il numero di epoche massimo è stato settato a 150, le iterazioni totali saranno  $150 * 6 = 900$ .

<sup>6</sup> Scenario in cui il modello non riesce ad apprendere la relazione tra input e output.

I parametri sono stati scelti in seguito a diverse prove effettuate al fine di avere la rete con le prestazioni migliori. La rete, quindi, è stata addestrata sul training-set con la funzione `trainNetwork`, monitorandone i progressi come riportato nella figura seguente.

Training on single CPU.					
Epoch	Iteration	Time Elapsed (hh:mm:ss)	Mini-batch RMSE	Mini-batch Loss	Base Learning Rate
1	1	00:00:01	444.27	98688.1	0.0100
9	50	00:00:22	367.03	67354.0	0.0100
17	100	00:00:46	297.18	44158.6	0.0100
25	150	00:01:14	233.35	27226.0	0.0100
34	200	00:01:47	186.44	17379.8	0.0100
42	250	00:02:17	169.58	14378.1	0.0100
50	300	00:02:49	140.90	9927.0	0.0100
59	350	00:03:19	97.40	4743.6	0.0100
67	400	00:03:51	57.68	1663.4	0.0100
75	450	00:04:20	47.55	1130.6	0.0100
84	500	00:04:51	24.79	307.2	0.0100
92	550	00:05:20	17.82	158.8	0.0100
100	600	00:05:52	13.13	86.2	0.0100
109	650	00:06:22	10.57	55.9	0.0100
117	700	00:06:53	9.89	48.9	0.0100
125	750	00:07:23	9.23	42.6	0.0100
134	800	00:07:52	8.45	35.7	0.0100
142	850	00:08:23	8.18	33.5	0.0100
150	900	00:08:57	7.37	27.1	0.0100

Figura 7.24 – Informazioni per monitorare il training della rete neurale

La figura 7.24 riporta il numero di epoche, il numero di iterazioni, il tempo trascorso, l'errore quadratico medio sul mini-batch, la perdita sul mini-batch e il tasso di apprendimento iniziale.

Infine, è stata predetta la RUL del gruppo 1, ovvero quello considerato per il test, tramite la funzione `predictAndUpdateState`, la quale, man mano che riceve in ingresso i dati, aggiorna i propri parametri ed esegue la predizione. Di seguito viene riportato il grafico della RUL stimata in ogni istante, confrontandola con la RUL reale.

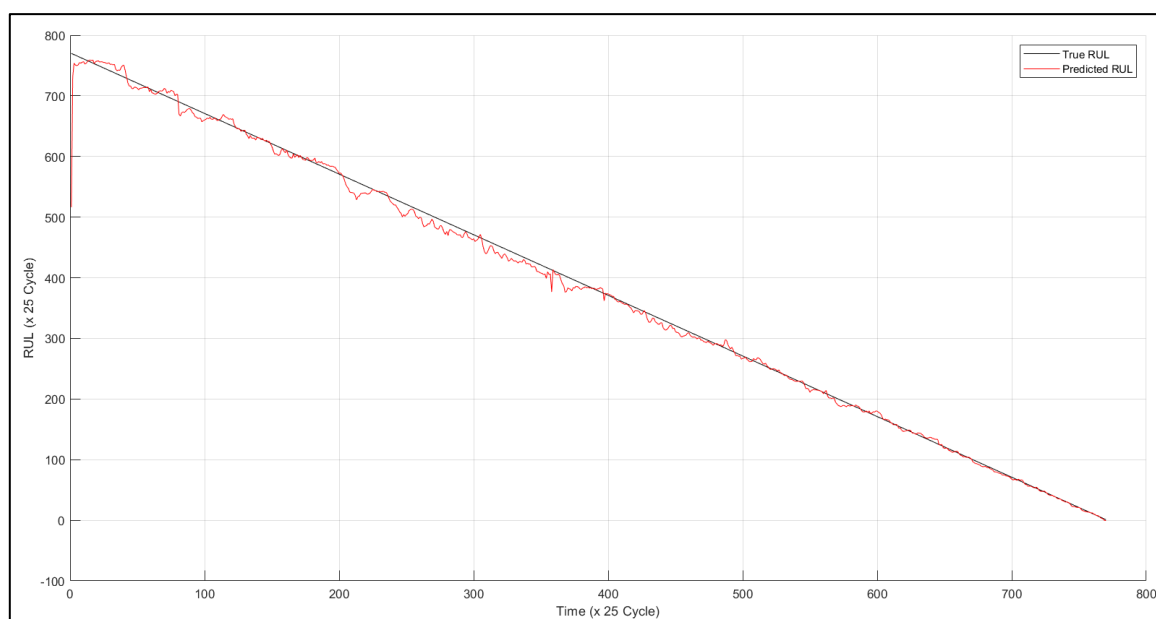
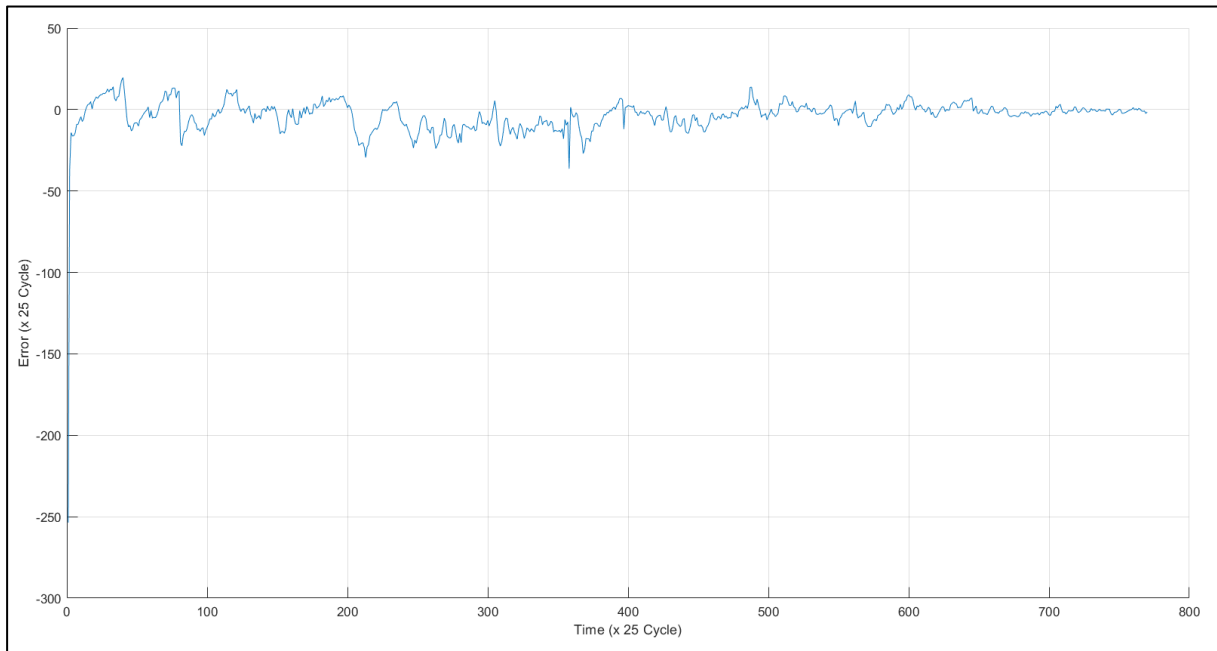


Figura 7.25 – Confronto RUL reale e RUL predetta

Il grafico mostra che, per la prima parte, la predizione della RUL non è molto precisa, a differenza dell'ultima parte in cui, ad ogni istante, è prossima alla RUL reale. Il motivo è che, andando avanti con le acquisizioni, la rete apprende maggiormente sull'andamento della vita del coltello, regolando i pesi delle connessioni tra i layer e migliorando le predizioni. A conferma di ciò, la figura seguente riporta l'andamento della differenza tra RUL reale e stimata nel corso delle acquisizioni.



*Figura 7.26 – Andamento dell'errore tra RUL vera e predetta*

I risultati ottenuti con la rete neurale LSTM, quindi, non si discostano di molto dalla RUL reale; tuttavia, anche in questo caso, l'algoritmo per essere applicato dovrebbe essere addestrato sullo storico dei dati reali dei coltelli, dalla condizione di sano fino alla condizione di usurato.

## **7.2 Osservazioni sui risultati ottenuti**

I due metodi utilizzati si differenziano per il tipo di dati a disposizione e per la tecnica impiegata per la stima. Infatti, nel primo algoritmo è stato utilizzato il Degradation Model, tramite la funzione `predictAndUpdateState` proposta da Matlab: il modello viene impiegato quando si ha l'andamento dell'indicatore di usura del componente in esame ed è nota la soglia che individua la condizione di usura; quindi, sulla base dell'andamento del degrado predice la RUL, ovvero il tempo rimanente prima di raggiungere la soglia. Il secondo algoritmo, invece, si basa sul Similarity Model, che stima la RUL del componente confrontando l'andamento dell'usura con gli andamenti di componenti simili; per esso si è utilizzato il deep learning, con l'addestramento della rete neurale LSTM.



Per entrambi gli algoritmi i problemi incontrati sono stati relativi al fatto che, non avendo a disposizione la vita intera del coltello, è stata considerata come vita la sequenza delle acquisizioni relative a diversi tipi di usura crescente: non essendo questo l'andamento reale dell'usura, inevitabilmente le predizioni non sono state del tutto soddisfacenti. Inoltre, soprattutto per quanto riguarda la prima tecnica, la stima è errata per i primi cicli, in quanto il modello, non conoscendo ancora il sistema in esame, ha parametri di predizione non ancora corretti.

## Conclusioni

Il presente elaborato è stato sviluppato nell'ambito dell'attività di tirocinio presso la Schnell S.p.A., produttrice di macchine per la lavorazione del tondino d'acciaio da costruzione.

A seguito delle diverse problematiche riscontrate, connesse con l'usura del coltello (prisma che causa il taglio del tondino d'acciaio), l'azienda persegue il duplice obiettivo di individuare il tipo di degrado sul coltello e di prevedere la vita utile rimanente, per cercare di intervenire prima che l'utilizzo del macchinario con un coltello eccessivamente usurato porti a rotture impreviste, e ben più gravi, del macchinario stesso. Scopo del presente studio è stato, quindi, quello di realizzare modelli per la classificazione dell'usura e di sviluppare algoritmi per la predizione della vita utile rimanente.

Inizialmente sono stati effettuati migliaia di cicli di taglio, sotto diverse condizioni di usura, acquisendo ogni volta diversi segnali provenienti dell'organo del taglio come quelli del driver motore (temperatura, posizione, velocità e intensità di corrente), la forza rilevata dall'estensimetro posizionato sulla biella e le vibrazioni rilevate dagli accelerometri. A questi dati è stata applicata la tecnica dell'extrapolazione delle features, che permette di trasformare dataset di grandi dimensioni, in dati con più contenuto informativo e di dimensione minore.

Le features sono state prima ordinate secondo i valori restituiti dal criterio One-Way ANOVA, che indica quanto la feature in esame è significativa della presenza dell'usura, e poi selezionate considerando quelle con i valori maggiori. In seguito al ranking delle features, è stato conseguito un primo importante risultato, ovvero individuare le grandezze più sensibili della presenza dell'usura sul coltello (quelle con i valori maggiori), quali l'intensità della corrente del motore e la forza rilevata dall'estensimetro sulla biella: questo evita l'utilizzo di ulteriori sensori sulla macchina, riducendo costi.

In seguito, le features selezionate sono state utilizzate come training-set per addestrare il classificatore tramite il toolbox *ClassificationLearner* di Matlab, al fine di individuare le tre classi di usura del coltello: coltello nuovo, coltello semi-usurato e coltello usurato. Sia per il tondino da 12 mm sia per quello da 16 mm i classificatori migliori sono stati i Support Vector Machine (SVM). La metrica seguita nella scelta dei modelli migliori è stata la precisione sul validation-set:

- per il tondino da 12 mm, la precisione del modello ottenuto è stata del 94,3%;

- per il tondino da 16 mm, la precisione del modello ottenuto è stata del 99,2%.

Entrambi i modelli, quindi, riescono bene a classificare il degrado sulla lama, tuttavia potrebbero essere migliorati ulteriormente considerando come training-set le acquisizioni relative all'usura reale che si forma sul coltello. Infatti, nell'addestramento sono stati considerati dati di coltelli in condizioni di assenza di usura, di coltelli in presenza di usura applicata manualmente e di coltelli di materiali nettamente più morbidi di quello standard al fine di accelerare la formazione del degrado.

Infine, i due classificatori ottenuti sono stati utilizzati per l'individuazione dell'usura real-time, mettendo in comunicazione Matlab e il PLC, che gestisce la macchina, tramite cavo Ethernet: al termine di ogni ciclo di taglio, viene mostrato nel Command Window di Matlab il valore indicativo del tipo di usura presente nel coltello.

Nella seconda parte dell'attività, sono stati implementati due algoritmi per la predizione della vita utile rimanente (RUL) del coltello. Lo scopo non era quello di realizzare modelli di predizione da applicare subito al caso pratico, come effettuato per i modelli di classificazione, ma comprendere il funzionamento di due tecniche di predizione e proporre due algoritmi. Il primo algoritmo si basa sul Degradation Model, in cui viene predetta la RUL a partire dall'andamento dell'indicatore di condizione dell'usura del coltello e dalla conoscenza della soglia di sicurezza. Poiché non si aveva a disposizione l'andamento reale del degrado del coltello dalla condizione di sano e a quella di usurato, è stato ipotizzato che la vita del coltello fosse quella costituita dalla sequenza delle acquisizioni relative a diversi tipi di usura crescente. Sono state considerate le features estrapolate precedentemente e sono state ordinate secondo il livello di monotonicità, applicando il "Metodo del coefficiente di correlazione di Spearman", che calcola la monotonicità tramite la correlazione monotona tra la feature e la variabile tempo. Sono state selezionate le features con i valori maggiori e su di esse è stata applicata la Principle Component Analysis. A questo punto è stata presa la prima componente principale ottenuta come indicatore di condizione. Per testare la tecnica è stato supposto di essere al 40% della vita e di predire la RUL, ovvero di predire tra quanti ciclo di taglio l'indicatore avrebbe superato la soglia di sicurezza. A tale scopo, Matlab propone due tipologie di modelli di degradazione:

- il modello lineare, che descrive il comportamento di degradazione come un processo stocastico lineare, da utilizzare quando il sistema non presenta un degrado cumulativo;

- il modello esponenziale, che descrive il comportamento di degrado come un processo stocastico esponenziale, da utilizzare quando il componente di test subisce un degrado cumulativo.

Non essendo noto il comportamento effettivo dell'usura (se lineare o esponenziale), in quanto l'indicatore di condizione è stato considerato come la sequenza di usure diverse, sono stati applicati entrambi i modelli di degradazione. Il modello lineare ha ottenuto risultati migliori rispetto a quello esponenziale, poiché la predizione risulta più vicina alla vera RUL, tuttavia, i risultati non sono molto precisi, in quanto l'indicatore di condizione in questione fa riferimento non al vero degrado, ma all'unione di diverse usure. Per migliorare l'algoritmo si dovrebbe osservare l'andamento reale del coltello e avere a disposizione lo storico del degrado di ulteriori coltelli, al fine di addestrare il modello su parametri reali. Inoltre, la predizione della RUL con questa tecnica impiega del tempo prima di avere stime precise, in quanto all'inizio delle predizioni (ovvero solamente dopo pochi cicli di lavoro) la conoscenza del sistema in esame è minima.

Il secondo algoritmo, invece, si basa sul Similarity Model, tecnica che stima la RUL del coltello sulla base dello storico di degradazione di componenti simili. La predizione utilizza la rete neurale LSTM, una rete ricorrente che tiene memoria di sequenze passate. Come per il primo algoritmo, è stato ipotizzato che la vita del coltello fosse quella costituita dalla sequenza di acquisizioni relative a diversi tipi di usura crescente. Dal dataset sono stati ottenuti venticinque gruppi di dati, considerati come le vite di altrettanti coltelli: ventiquattro di essi sono stati utilizzati per il training, mentre il rimanente per il test della rete. La rete, quindi, è stata addestrata sul training-set e poi testata sul test-set, ottenendo buoni risultati: la RUL predetta in ogni istante è risultata vicina alla RUL vera, in particolare nell'ultima parte, ovvero quando la rete aveva ormai appreso l'andamento dell'usura del coltello in esame. Anche in questo caso, l'algoritmo deve essere migliorato considerando la vita reale dei componenti per poter essere applicato.

In conclusione, il lavoro eseguito ha consentito di individuare, fra tutte le grandezze monitorate, la forza rilevata dall'estensimetro sulla biella e l'intensità di corrente del motore come le più significative della presenza dell'usura sul coltello. I classificatori ottenuti riescono a determinare il tipo di usura, ma dovrebbero essere perfezionati con l'addestramento su dati relativi alla vera usura, per essere utilizzati. I modelli di predizione della RUL, invece,

necessitano dell'andamento reale del degrado del coltello e rappresentano, quindi, solo una base di partenza per ulteriori studi.

## Bibliografia

- [1] “Progetto Simple”, documento Schnell S.p.A.
- [2] L. Cavalletti, “Manutenzione preventiva di macchine per taglio acciaio, il caso Schnell S.p.A”, Laurea magistrale in Ingegneria Meccanica, Università Politecnica delle Marche, anno accademico 2017-2018
- [3] Matlab, “Predictive Maintenance, Part 2: Feature Extraction for Identifying Condition Indicators”, 2019 Gennaio 15, video online in *YouTube*  
<https://www.youtube.com/watch?v=pcXr8I2QvHw>
- [4] “Signal Features”, in *MathWorks* <https://it.mathworks.com/help/predmaint/ug/signal-features.html> (ultimo accesso: 2021 Novembre 29)
- [5] S. Heo, J. H. Lee, “Fault detection and classification using artificial neural networks”, 2018, in *ScienceDirect*, <https://www.sciencedirect.com/>
- [6] G. James, D. Witten, T. Hastie, R. Tibshirani, “An Introduction to Statistical Learning with Applications in R”, Springer, 2013
- [7] T. K. Kim, “Understanding one-way ANOVA using conceptual figures”, *Korean Journal of Anesthesiology*, 70, pp. 22-26, 2017  
<https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5296382/>
- [8] M. G. Larson, “Analysis of Variance”, *Circulation*, Vol.17, N. 1, 2008
- [9] MATLAB helper ®, “*Classification Learner APP | MATLAB Helper Blog*”, video online in *YouTube* <https://www.youtube.com/watch?v=3YGtxY67bq0> (ultimo accesso: 2021 Novembre 29)
- [10] R. Kohavi, F. Provost, “Glossary of terms”, pp. 271–274, 1998
- [11] “Assess Classifier Performance in Classification Learner”, in *Mathworks* <https://it.mathworks.com/help/stats/assess-classifier-performance.html> (ultimo accesso: 2021 Novembre 29)
- [12] S. Pulkit, “The Most Comprehensive Guide to K-Means Clustering You’ll Ever Need”, 2019, in *AnalyticsVidhya*  
<https://www.analyticsvidhya.com/blog/2019/08/comprehensive-guide-k-means-clustering/> (ultimo accesso: 2021 Novembre 29)
- [13] “TC1000 | TwinCAT 3 ADS”, in *Beckhoff*  
<https://www.beckhoff.com/it-it/products/automation/twincat/tc1xxx-twincat-3-base/tc1000.html> (ultimo accesso: 2021 Novembre 29)

- [14] “Manual: MATLAB®/Simulink® TwinCAT 3”, Vers. 1.2, 2017  
<https://docplayer.net/58530683-Manual-matlab-simulink-twincat-3-version-date-order-no-te14xx.html> (ultimo accesso: 2021 Novembre 29)
- [15] “Evoluzione del concetto di manutenzione”, in *diacs* <https://www.diacs.it/evoluzione-della-manutenzione/> (ultimo accesso: 2021 Novembre 29)
- [16] MATLAB, “Predictive Maintenance, Part 1: Introduction”, 2018 Dicembre 19, video online in *YouTube* [Predictive Maintenance, Part 1: Introduction - YouTube](https://www.youtube.com/watch?v=...)
- [17] “Predictive Maintenance: Estimating Remaining Useful Life with MATLAB”  
<https://opencadd.com.br/wp-content/uploads/2020/01/predictive-maintenance-ebook-part3-estimating-rul-with-matlab.pdf> (ultimo accesso: 2021 Novembre 29)
- [18] A. Baru, “Three Ways to Estimate Remaining Useful Life for Predictive Maintenance”, in *MathWorks* <https://it.mathworks.com/company/newsletters/articles/three-ways-to-estimate-remaining-useful-life-for-predictive-maintenance.html> (ultimo accesso: 2021 Novembre 29)
- [19] “Wind Turbine High-Speed Bearing Prognosis” in *MathWorks*  
<https://it.mathworks.com/help/predmaint/ug/wind-turbine-high-speed-bearing-prognosis.html> (ultimo accesso: 2021 Novembre 29)
- [20] Y. Dodge (2010), “The Concise Encyclopedia of Statistics”, Springer-Verlag New York, 2010, pp. 502
- [21] S. Chakraborty, N. Gebraeel, M. Lawley, and H. Wan, "Residual-Life Estimation for Components with Non-Symmetric Priors.", *IIE Transactions* Vol. 41, Number 4, 2009, pp. 372–387
- [22] N. Gebraeel, "Sensory-Updated Residual Life Distributions for Components with Exponential Degradation Patterns.", *IEEE Transactions on Automation Science and Engineering*. Vol. 3, Number 4, 2006, pp. 382–393
- [23] K. Suzuki, “Artificial Neural Networks - Methodological Advances and Biomedical Applications”, InTech, 2011
- [24] O. Calligaris, “Introduzione alle reti neurali”,  
<https://matematica.unibocconi.it/articoli/introduzione-alle-reti-neurali> (ultimo accesso: 2021 Novembre 29)
- [25] Data Science Team, “Introduzione al concetto LSTM”, 2019, in *datascience*  
<https://datascience.eu/it/apprendimento-automatico/comprendimento-delle-reti-lstm/>  
 (ultimo accesso: 2021 Novembre 29)

- [26] “Guida illustrata a LSTM e GRU: una spiegazione passo passo” in *ichi.pro*  
<https://ichi.pro/it/guida-illustrata-a-lstm-e-gru-una-spiegazione-passo-passo-75771469479713> (ultimo accesso: 2021 Novembre 29)
- [27] “trainingOptions” in *MathWorks*  
[https://it.mathworks.com/help/deeplearning/ref/trainingoptions.html#bu80qkw-3\\_head](https://it.mathworks.com/help/deeplearning/ref/trainingoptions.html#bu80qkw-3_head)  
(ultimo accesso: 2021 Novembre 29)

### **Bibliografia non citata**

R. Iserman, “Fault-Diagnosis Systems: An Introduction from Fault Detection to Fault Tolerance”, Berlin: Springer Verlag, 2006.