

POLITECNICO DI TORINO

DIGEP

Laurea Magistrale
In
Ingegneria Gestionale

TESI DI LAUREA

**Sviluppo di un modello di manutenzione predittiva basato sulla
combinazione di metodi physics-based e data-driven**



Relatori:

Prof.ssa Giulia Bruno
Prof. Franco Lombardi
Dott. Emiliano Traini

CANDIDATO
Santino Telesca
265210

Anno accademico 2019-2020

This work is subject to the Creative Commons Licence

All Rights Reserved

ABSTRACT

Sviluppo di un modello di manutenzione predittiva basato sulla combinazione di metodi physics-based e data-driven

Il processo di manutenzione è cruciale in tutti i sistemi soggetti a guasti o degradazione, in particolare nelle attività manifatturiere. Infatti, i costi di manutenzione possono raggiungere fino al 40% del costo di produzione in determinati settori. Nella piena evoluzione del fenomeno dell'industria 4.0 si stanno diffondendo dei metodi di manutenzione che consentono di effettuare interventi più efficaci che, massimizzando l'utilizzo dei componenti nella loro vita utile, riducono i costi operativi e i tempi di inattività. Tali metodi sono identificati con la categoria *manutenzione predittiva* e includono diverse tecnologie innovative, come l'*IoT* per la diffusione dei sensori che monitorano le macchine e l'*AI* che fornisce gli algoritmi per interpretare i dati raccolti.

Un processo specifico di manutenzione è il *Tool Condition Monitoring*, il quale ha l'obiettivo di monitorare le condizioni dell'utensile di una macchina affinché possa produrre dei pezzi lavorati che rispettino i requisiti di progettazione. Ciò è necessario poiché, a causa delle sollecitazioni a cui è sottoposto l'utensile nel processo di lavorazione, esso è soggetto a diversi fenomeni di degradazione i quali ne compromettono la qualità superficiale realizzata dall'utensile.

La ricerca di tesi si basa sul monitoraggio dell'utensile della fresatura, uno dei processi più diffusi nell'ambito delle lavorazioni meccaniche. Di tale utensile il processo di degradazione che influenza maggiormente la qualità di lavorazione è la *flank wear*, la quale è difficile da monitorare direttamente a causa delle misurazioni onerose. Da qui nasce la necessità di un metodo di manutenzione predittiva, con lo scopo di prevedere l'andamento del processo di usura con l'utilizzo di sensori che misurano parametri indiretti, come le vibrazioni e le emissioni acustiche.

Sulla base del dataset sperimentale *milling* (NASA-PCoE) sono stati implementati e confrontati diversi metodi *data-driven*, tra cui il modello di serie storiche *ARIMA* e i principali algoritmi *machine learning* (*ML*), e sono stati confrontati con un metodo, basato sulla relazione di *Taylor* sulla vita utile dell'utensile, più tradizionale *physics-based*. Dalle peculiarità dell'algoritmo basato sulle reti neurali, modello *ML* risultato migliore, e del modello di *Taylor* è stato sviluppato un modello ibrido che sfrutta entrambe le potenzialità.

INDICE

Pagina

ABSTRACT	II
INDICE	III
LISTA DELLE FIGURE	V
LISTA DELLE TABELLE	VI
LISTA DELLE EQUAZIONI	VII
LISTA DEGLI SCRIPT	VIII
INTRODUZIONE	1
1.1 L'IMPATTO DEI PROCESSI DI MANUTENZIONE NEI SETTORI PRODUTTIVI	1
1.2 LA RELAZIONE CON L'INDUSTRIA 4.0	2
1.3 INTRODUZIONE ALLA TESI	2
STATO DELL'ARTE	4
2.1 CLASSIFICAZIONE DEI METODI DI MANUTENZIONE	4
2.2 METODI TRADIZIONALI	8
2.3 METODI STATISTICI	10
2.3.1 <i>Serie storiche</i>	11
2.4 PCA	12
2.5 METODI BASATI SU ALGORITMI DI MACHINE LEARNING	13
2.6 METODI IBRIDI	16
2.6.1 <i>Combinazione tra ARIMA e ANN</i>	17
FENOMENI DI USURA DELL'UTENSILE NEI PROCESSI DI LAVORAZIONE PER ASPORTAZIONE DI MATERIALE	19
3.1 INTRODUZIONE	19
3.2 CENNI SUL PROCESSO DI FRESATURA	19
3.3 TIPI DI USURA	20
3.4 MECCANISMI DI USURA	23
SERIE STORICHE	26
4.1 INTRODUZIONE	26
4.2 CLASSIFICAZIONE DEI METODI DI PREVISIONE BASATI SULLE SERIE STORICHE	26
4.3 METODO ARIMA	28
4.3.1 <i>Modello autoregressivo</i>	28
4.3.2 <i>Differenziazione</i>	29
4.3.3 <i>Media mobile sull'errore</i>	29
4.3.4 <i>Modello generalizzato</i>	30
4.4 IMPLEMENTAZIONE IN R	30

MODELLI PHYSICS-BASED	33
5.1 INTRODUZIONE.....	33
5.2 MODELLI DI DEGRADAZIONE.....	34
5.2.1 <i>Degradazione per frattura</i>	34
5.2.2 <i>Degradazione per fatica</i>	35
5.2.3 <i>Degradazione per usura</i>	35
5.3 METODO DI IMPLEMENTAZIONE	36
5.4 INTERPRETAZIONE FISICA DELL'EQUAZIONE DI TAYLOR	38
5.5 MODELLO DI TAYLOR ESTESO.....	39
5.5.1 <i>Stima dei coefficienti</i>	39
5.6 IMPLEMENTAZIONE IN R	40
STRUMENTI UTILIZZATI	42
6.1 R.....	42
6.2 LIBRERIE	42
CASO DI STUDIO	45
7.1 FRAMEWORK.....	45
7.2 DATASET	48
METODI DI VALUTAZIONE DEI MODELLI PREDITTIVI.....	52
8.1 VALIDAZIONE DEL MODELLO DI PREVISIONE.....	52
8.2 CROSS VALIDATION.....	53
8.3 METRICHE DI VALUTAZIONE	54
IMPLEMENTAZIONE MODELLI DI PREVISIONE DELL'USURA	56
9.1 DATA CLEANING.....	56
9.2 ANALISI ANDAMENTO DELLA VARIABILE DI CONTROLLO (VB).....	57
9.3 APPROCCIO DI CROSS VALIDATION.....	58
9.4 MODELLI ARIMA.....	61
9.4.1 <i>Modello ARIMA dinamico</i>	65
9.5 MODELLO DI TAYLOR.....	67
9.5.1 <i>Modello Taylor con dataset ridotto</i>	71
9.6 ALGORITMI DI MACHINE LEARNING.....	72
9.6.1 <i>Confronto performance al variare della dimensione del training set</i>	75
9.6.2 <i>Sensibilità alla qualità dei dati degli algoritmi di ML</i>	77
9.6.3 <i>Ottimizzazione algoritmo NN</i>	80
9.7 ANALISI ERRORE DI PREVISIONE ALL'AVANZARE DEI RUN	81
9.8 MODELLO IBRIDO.....	84
CONCLUSIONI E LAVORI FUTURI.....	89
10.1 CONCLUSIONI.....	89
10.2 LAVORI FUTURI.....	90
APPENDICE 1	92
APPENDICE 2	93
RIFERIMENTI.....	94

LISTA DELLE FIGURE

	Pagina
FIGURA 2.1.1 RAPPRESENTAZIONE TEMPI DI INTERVENTO NELLA MANUTENZIONE PREVENTIVA [5]	5
FIGURA 2.1.2 RAFFIGURAZIONE TIPI DI MANUTENZIONE [9]	6
FIGURA 2.3.1 CURVE DI GUASTO [34]	10
FIGURA 3.2.1 RAPPRESENTAZIONE FRESATURA [74]	19
FIGURA 3.3.1 ANDAMENTO NEL TEMPO DELLA FLANK WEAR [72]	21
FIGURA 3.3.2 TIPI DI USURA: (A) FLANK WEAR; (B) CRATER WEAR; (C) NOTCH WEAR; (D) NOSE RADIUS (E) COMB CRACKS; (F) PARALLEL (MECHANICAL) CRACKS; (G) BUILT-UP EDGE; (H) GROSS PLASTIC DEFORMATION; (I) EDGE CHIPPING; (J) CHIP HAMMERING; (K) GROSS FRACTURE [72]	22
FIGURA 4.2.1 CLASSIFICAZIONE METODI DI SERIE STORICHE [50]	27
FIGURA 4.4.1 PROCEDURA IMPLEMENTAZIONE MODELLO ARIMA [60]	32
FIGURA 5.4.1 RELAZIONE EMPIRICA TRA T E V [70]	38
FIGURA 7.1.1 SCHEMA DEL FRAMEWORK UTILIZZATO [100]	45
FIGURA 7.2.1 SETUP DELLA RACCOLTA DATI [98]	50
FIGURA 8.1.1 SCOMPOSIZIONE DATASET PER TESTARE MODELLI DI PREVISIONE	52
FIGURA 8.2.1 RAPPRESENTAZIONE METODO DI CROSS VALIDATION [110]	53
FIGURA 9.1.1 ANDAMENTO DI VB (DA MISURAZIONI) PER SCENARIO DOC=1,5, FEED=0,5, MATERIALE=CAST IRON.....	57
FIGURA 9.2.1 ANDAMENTO DI VB ALL'AVANZARE DEI RUN CON LE MISURE DEL DATASET	58
FIGURA 9.3.1 RAPPRESENTAZIONE APPROCCIO DI CROSS VALIDATION UTILIZZATO	59
FIGURA 9.4.1 PERFORMANCE MODELLI ARIMA	63
FIGURA 9.4.2 CONFRONTO PREVISIONI VB MODELLO ARIMA CON VALORI EFFETTIVI	64
FIGURA 9.4.3 VARIAZIONE DELL'ERRORE % SULLA PREVISIONE DEL LIVELLO DI VB DELL'ULTIMO RUN (T=74) CON MODELLO ARIMA	66
FIGURA 9.4.4 PREVISIONI ARIMA DINAMICO: (A) PREVISIONI EFFETTUATE IN T=0 (B) PREVISIONI EFFETTUATE IN T=24 (DOPO L'OSSERVAZIONE DI 3 RUN)	66
FIGURA 9.5.1 PERFORMANCE MODELLO DI TAYLOR PER PREVISIONI VB SUL TEST SET	69
FIGURA 9.5.2 CONFRONTO PREVISIONI VB MODELLO TAYLOR CON VALORI EFFETTIVI	70
FIGURA 9.5.3 ANDAMENTO RMSE ALL'AUMENTARE DEI RUN OSSERVATI NEL MODELLO DI TAYLOR	71
FIGURA 9.5.4 PRESTAZIONI MODELLO TAYLOR RIDOTTO	72
FIGURA 9.6.1 BOX-PLOT CONFRONTO PERFORMANCE MODELLI DI TAYLOR E NN	75
FIGURA 9.6.2 PREVISIONI CASE 7 CON RIMOZIONE FEATURES AE_TABLE	79
FIGURA 9.6.3 RISULTATO OTTIMIZZAZIONE IPERPARAMETRI ALGORITMO NN	81
FIGURA 9.7.1 DISTRIBUZIONE ERRORE ALL'AVANZARE DEI RUN NEI MODELLI DI TAYLOR E NN	82
FIGURA 9.7.2 CONFRONTO ANDAMENTO ERRORI DEI MODELLI TAYLOR E NN ALL'AVANZARE DEI RUN CON CURVA TEORICA DI PROPAGAZIONE DELLA <i>FLANK WEAR</i>	83
FIGURA 9.8.1 PESI OTTENUTI NEL MODELLO IBRIDO	86
FIGURA 9.8.2 PRESTAZIONI MODELLO IBRIDO CONFRONTATE CON MODELLI SINGOLI	87
FIGURA 9.8.3 CONFRONTO PRESTAZIONI MODELLO IBRIDO AL VARIARE DELLA DIMENSIONE DEL TRAINING SET	88

LISTA DELLE TABELLE

	Pagina
TABELLA 7.2.1 DESCRIZIONE CASE DEL DATASET [98]	48
TABELLA 7.2.2 DESCRIZIONE CAMPI DATASET [98]	49
TABELLA 9.4.1 PERFORMANCE ARIMA CON DIVERSI REGRESSORI	64
TABELLA 9.5.1 RISULTATO REGRESSIONE MODELLO TAYLOR.....	69
TABELLA 9.5.2 DATASET RIDOTTO PER MODELLO TAYLOR (MATERIALE CAST IRON)	71
TABELLA 9.6.1 PERFORMANCE ALGORITMI DI ML CON 10/14 TOOLS NEL TRAINING SET	74

LISTA DELLE EQUAZIONI

	Pagina
EQUAZIONE 2.2.1 RELAZIONE BASE DI TAYLOR.....	8
EQUAZIONE 2.2.2 COEFFICIENTI DI TAYLOR BASE	8
EQUAZIONE 2.2.3 RELAZIONE DI COLDING CON 5 COSTANTI.....	9
EQUAZIONE 2.6.1 SUDDIVISIONE COMPONENTI DI USA SERIE STORICA.....	18
EQUAZIONE 2.6.2 RELAZIONE RESIDUI CON COMPONENTE LINEARE	18
EQUAZIONE 2.6.3 FUNZIONE RESIDUI DI REGRESSIONE.....	18
EQUAZIONE 4.2.1 SMORZAMENTO ESPONENZIALE.....	27
EQUAZIONE 4.2.2 REGRESSIONE LINEARE SEMPLICE	28
EQUAZIONE 4.3.1 MODELLO AUTOREGRESSIVO.....	29
EQUAZIONE 4.3.2 RELAZIONE DIFFERENZIAZIONE DEL PRIMO ORDINE.....	29
EQUAZIONE 4.3.3 MEDIA MOBILE SU ERRORE DI GRADO Q	29
EQUAZIONE 4.3.4 MODELLO ARIMA GENERALIZZATO.....	30
EQUAZIONE 5.2.1 DEFINIZIONE CREEP RATE	34
EQUAZIONE 5.2.2 LEGGE DI NORTON SUL CREEP RATE.....	34
EQUAZIONE 5.2.3 MODELLO DI WOHLER	35
EQUAZIONE 5.2.4 LEGGE DI ARCHARD	36
EQUAZIONE 5.2.5 LEGGE DI COULOMB SULLA FORZA DI FRIZIONE	36
EQUAZIONE 5.4.1 RELAZIONE DELLA COSTANTE DELLA LEGGE DI TAYLOR.....	38
EQUAZIONE 5.4.2 RELAZIONE N DELL'EQUAZIONE DI TAYLOR	38
EQUAZIONE 5.5.1 MODELLO DI TAYLOR ESTESO PER UTENSILI DA TAGLIO	39
EQUAZIONE 5.5.2 RELAZIONE INVERSA DI TAYLOR PER STIMA VB.....	39
EQUAZIONE 5.5.3 TRASFORMAZIONE LOGARITMICA DELL'EQUAZIONE DI TAYLOR ESTESA.....	40
EQUAZIONE 5.5.4 EQUAZIONE MATRICIALE STIMA COEFFICIENTI CON OLS.....	40
EQUAZIONE 5.5.5 RAPPORTO DI DUARTE E ARRUDA.....	40
EQUAZIONE 7.2.1 RMS (ROOT MEAN SQUARE)	51
EQUAZIONE 8.3.1 ERRORE DI PREVISIONE.....	54
EQUAZIONE 8.3.2 ME: ERRORE MEDIO DI PREVISIONE	54
EQUAZIONE 8.3.3 MAPE: ERRORE ASSOLUTO MEDIO PERCENTUALE	54
EQUAZIONE 8.3.4 RMSE: RADICE DELL'ERRORE QUADRATICO MEDIO	55
EQUAZIONE 8.3.5 RRSE: RADICE DELL'ERRORE QUADRATICO RELATIVO	55
EQUAZIONE 9.8.1 CALCOLO PREVISIONI DI VB DEL MODELLO IBRIDO.....	84
EQUAZIONE 9.8.2 CONDIZIONE PER CALCOLARE I PESI DEL MODELLO IBRIDO	84

LISTA DEGLI SCRIPT

	Pagina
SCRIPT 1 ALGORITMO DI CROSS VALIDATION	60
SCRIPT 2 DEFINIZIONE ATTRIBUTO <i>CHANGE</i>	61
SCRIPT 3 MODELLO ARIMA (P,D,Q).....	62
SCRIPT 4 MODELLI ARIMA AUTOMATICI.....	62
SCRIPT 5 MODELLO DI TAYLOR PER PREVISIONI VB	67
SCRIPT 6 ALGORITMI DI MACHINE LEARNING	73
SCRIPT 7 OTTIMIZZAZIONE IPERPARAMETRI ALGORITMO NN.....	80
SCRIPT 8 ALGORITMO MODELLO IBRIDO TRA TAYLOR E NN.....	85

1 CAPITOLO

INTRODUZIONE

Il seguente capitolo, nei primi due paragrafi, introduce il ruolo dei processi di manutenzione nelle attività produttive e l'evoluzione che stanno subendo tali processi nel contesto più ampio dell'industria 4.0. Infine, nell'ultimo paragrafo, sarà introdotta la struttura della tesi e l'obiettivo di ricerca che si pone.

1.1 L'impatto dei processi di manutenzione nei settori produttivi

Si stima che i costi di manutenzione variano tra il 15%, per il settore manifatturiero in generale, e fino al 40% per l'industria delle lavorazioni metalliche, del costo di produzione [1]. L'attenzione verso gli investimenti nei processi di manutenzione mostra un trend crescente, infatti nel 1981 sono stati spesi circa 600 miliardi di dollari negli Stati Uniti nella manutenzione degli impianti produttivi, cifra che è più che raddoppiata negli ultimi anni [2], puntando a sistemi di manutenzione sempre più efficienti. Con la corretta implementazione di sistemi di manutenzione predittiva tali costi possono essere ridotti fino al 30% [3], riducendo l'attività richiesta per effettuare il monitoraggio ed effettuando la sostituzione del sistema monitorato solo quando è necessario. Infatti, lo studio [93] riporta che con la manutenzione predittiva si può ridurre l'incidenza dei guasti fino al 70%, consentendo di aumentare il tempo produttivo dei sistemi fino al 30% [94]. Si stima che i metodi di manutenzione predittiva più recenti, basati su algoritmi di *machine learning*, possono ridurre un ulteriore 30% i costi di manutenzione attuali, incrementando la vita operativa delle macchine e riducendo i periodi di inattività [4].

L'efficienza dei metodi di prevenzione predittiva rispetto a quella dei metodi tradizionali di manutenzione preventiva è stata dimostrata da uno studio della Emerson Rosemount Company [93] comparando le attività di manutenzione dei trasmettitori in diversi settori. È emerso che nei settori in cui si adottavano approcci di manutenzione tradizionali, il 30% dei trasmettitori presentavano problemi critici al momento della sostituzione e che quindi necessitavano di essere sostituiti in anticipo rispetto al periodo di intervento schedato. Mentre nel settore degli impianti di produzione di energia nucleare, in cui si adottano sistemi di manutenzione predittiva, solo il 10% dei trasmettitori era in condizioni critiche dimostrando le potenzialità di tali metodi [34].

1.2 La relazione con l'industria 4.0

Il fenomeno che si sta assistendo negli ultimi anni nei processi produttivi è quello denominato con il termine Industria 4.0. All'interno di tale fenomeno sono incluse diverse tecnologie, tra cui l'Internet delle cose (*Internet of Things - IoT*) e lo sviluppo dell'intelligenza artificiale (*AI*), quali sono fattori abilitanti per lo sviluppo dei recenti metodi di manutenzione predittiva. Infatti, essi si basano sui dati raccolti in tempo reale dai sensori, che quindi necessitano della tecnologia *IoT*, e di software sofisticati per l'analisi della mole di dati raccolti che includono algoritmi di intelligenza artificiale [95].

Il paradigma emergente andrà a sostituire quello tradizionale in cui il controllo dei processi è affidato a codici scritti manualmente per PLC (*programmable logic controller*) i cui feedback sono poi interpretati dagli operatori con procedure standardizzate [96]. Nuovi sviluppi in determinati ambiti come quello della matematica applicata e in software di *statistical learning*, con la disponibilità crescente di strumenti *open source*, offrono un buon incentivo per incrementare la raccolta dei dati nei processi produttivi e il loro utilizzo in modelli basati sui dati raccolti con le nuove tecnologie. Uno degli sviluppi più interessanti degli ultimi anni è nell'ambito degli algoritmi di *machine learning*, i quali possono evolvere in modo autonomo migliorando le prestazioni nel tempo sfruttando i dati raccolti dai sensori che monitorano i sistemi di produzione. Tuttavia, la ricerca ha proposto diversi metodi basati sull'intelligenza artificiale, rendendo paradossalmente difficile la scelta del modello da adottare da parte dei responsabili dei processi produttivi. Ma date le loro potenzialità, gli algoritmi di *machine learning*, tra cui le reti neurali, troveranno sempre più applicazioni nell'ambito della produzione industriale [97].

1.3 Introduzione alla tesi

La ricerca di tesi ha l'obiettivo di analizzare i principali metodi di manutenzione, in particolare dei metodi *physics-based* e di quelli *data-driven* che abilitano la manutenzione predittiva, per poi implementare un modello ibrido che migliora le prestazioni complessive rispetto a quelle dei modelli singoli. Includendo il capitolo introduttivo, la tesi è composta da dieci capitoli, i quali verranno brevemente descritti. Il secondo capitolo contiene uno studio dell'arte che fornisce prima gli elementi teorici dei processi di manutenzione in generale e poi introduce alcune applicazioni dei principali metodi presenti in letteratura. I metodi di *machine learning* verranno descritti con maggiore dettaglio poiché sono quelli su cui si stanno verificando i maggiori sviluppi negli ultimi anni. Mentre, il terzo capitolo tratta sinteticamente il processo di lavorazione della fresatura poiché è quello su cui si basa il processo di *tool condition monitoring* trattato nel caso studio. Inoltre, saranno descritti i principali tipi di usura che si verificano nelle lavorazioni delle macchine utensili e i meccanismi fisici che li generano.

Nei capitoli 4 e 5 sono descritti, da una prospettiva principalmente teorica, i principali modelli dei metodi basati sulle serie storiche e di quelli *physics-based* rispettivamente. Mentre, non saranno trattati gli algoritmi *machine learning* in un capitolo dedicato poiché sono già descritti ampiamente nel capitolo dello stato dell'arte e nel corso dell'analisi dei risultati. In particolare, nel capitolo 4 sono descritti i modelli che compongono il metodo ARIMA generalizzato il quale è il metodo di riferimento per le serie storiche. Nel capitolo 5, invece, sono riportate le relazioni fisiche dei principali fenomeni di usura che si possono verificare sugli utensili di lavorazione. Inoltre, nell'ultimo paragrafo di ognuno dei due capitoli è presentato un algoritmo di implementazione dei modelli con il linguaggio di programmazione R. Quest'ultimo è lo strumento di riferimento per tutte le analisi svolte durante questa ricerca, il quale sarà introdotto nel capitolo 6 insieme alle librerie che sono state utilizzate negli script riportati in seguito.

I capitoli successivi descrivono la metodologia adottata nel caso di studio e i risultati ottenuti. Seguendo l'ordine, nel capitolo 7 è descritto sinteticamente il framework adottato, sviluppato da un gruppo di ricerca esterno, per utilizzare i dati dei sensori nei metodi *data-driven*. Dove, inoltre, sarà descritto il dataset sperimentale, costruito dal centro di ricerca NASA-PCoE, utilizzato in tutti i modelli presentati. Nel capitolo 8 è introdotta la teoria alla base utilizzata per valutare i modelli di *tool condition monitoring* implementati. In particolare, sono descritti l'approccio di *cross validation* utilizzato per testare i modelli e le metriche di misurazione dell'errore di previsione. Il capitolo 9 contiene la descrizione di tutti i modelli implementati, dove saranno discussi i risultati ottenuti in diverse condizioni. Dalle considerazioni sui risultati dei modelli singoli e sulla distribuzione dei valori del parametro monitorato sarà sviluppato un modello ibrido tra un modello *physics-based* e uno *data-driven*. Infine, nel capitolo 10, sono riportate le conclusioni sui risultati ottenuti dall'applicazione dei diversi metodi sottolineandone i punti di forza e le limitazioni di ognuno di essi.

2 CAPITOLO

STATO DELL'ARTE

Il seguente capitolo, diviso in sei sezioni (classificazione dei metodi, metodi tradizionali, metodi statistici, PCA, machine learning e metodi ibridi), delinea le principali caratteristiche dei metodi di manutenzione con applicazione industriale presenti in letteratura, distinguendo i metodi tradizionali da quelli innovativi. Nella prima sezione verranno presentati brevemente i possibili metodi di manutenzione in ordine di complessità, riportando le caratteristiche peculiari e le esigenze che hanno indotto la loro evoluzione. Nella seconda sono riportati i metodi tradizionali, che si sono sviluppati sulla base delle conoscenze della meccanica classica, i quali in alcuni contesti svolgono ancora un ruolo determinante. Nella terza si propone una descrizione generale dei metodi statistici applicati ai processi di manutenzione, i primi ad utilizzare i dati registrati sulle macchine per effettuare previsioni nell'ottica di una manutenzione predittiva. Seguono dei cenni alla PCA, una particolare tecnica di analisi dei dati che ha alla base specifiche funzioni statistiche, che permette di gestire con migliori risultati i dataset di grandi dimensioni. Mentre, nel quinto paragrafo si presta maggiore attenzione ai metodi basati su algoritmi di machine learning, mostrando come negli ultimi anni sono al centro dell'interesse della ricerca nell'ambito della manutenzione predittiva e riportando una breve descrizione dei principali algoritmi che sono stati sviluppati. Infine, sono presentati i metodi ibridi, i quali combinano le caratteristiche di diversi metodi per ottenere un modello complessivo che migliora le performance rispetto ai modelli singoli.

2.1 Classificazione dei metodi di manutenzione

La prima forma di manutenzione avveniva solo in caso di guasto, riparando o sostituendo il componente danneggiato o usurato. A partire dal 1950 è stata introdotta la manutenzione preventiva al fine di limitare gli effetti di un guasto, che con l'approccio precedente portava spesso a situazioni di inattività dell'intero processo produttivo [5]. I primi metodi di manutenzione preventiva si basavano solo su pianificazioni temporali, senza considerare le condizioni della macchina o componente soggetti a manutenzione. In questo contesto si cercava di minimizzare i costi totali di manutenzione, considerando i costi di sostituzione e le perdite operative della macchina, e un primo modello matematico fu proposto da Bazovsky nel 1961 [6].

Tuttavia, tali approcci non riuscivano a prevedere guasti anomali e spesso portavano a interventi non necessari. Come mostrato nella figura 2.1.1, a partire dall'implementazione

della macchina o sostituzione del componente (t_0), con la manutenzione preventiva si fissa un tempo (t_r) in cui effettuare la sostituzione. Tra l'istante di sostituzione e quello in cui avverrebbe effettivamente il guasto (t_f) intercorre un intervallo che può essere significativo, con la manutenzione predittiva si cerca di minimizzare tale differenza in modo da ridurre i costi di sostituzione. È stato stimato che il 99% dei guasti meccanici può essere previsto con l'ausilio di specifici indicatori, su questa base nasce la *Condition Based Maintenance* (CBM) [7].

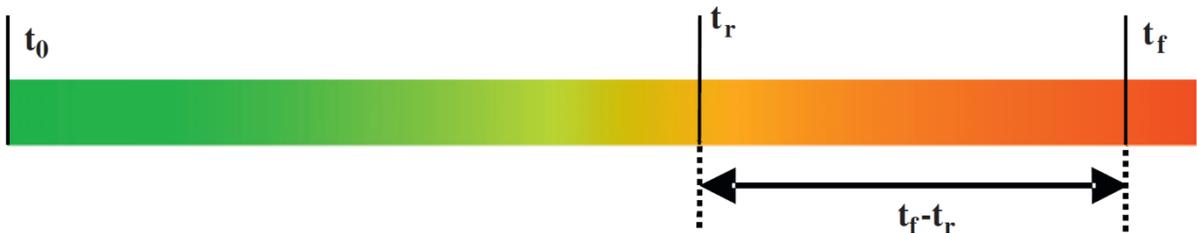


Figura 2.1.1 Rappresentazione tempi di intervento nella manutenzione preventiva [5]

La motivazione che blocca le organizzazioni a sviluppare metodi prognostici può essere trovata nel conservatorismo che è presente nelle politiche di manutenzione tradizionale. Ciò significa che nei sistemi in cui si applica una manutenzione preventiva, i componenti sono sostituiti con largo anticipo rispetto alla loro vita utile rimanente. È stato dimostrato che questo conservatorismo è dovuto alle incertezze nell'evoluzione dell'usura del componente [106]. Pertanto, non conoscendo il reale stato di usura del componente, si sceglie di intervenire in anticipo con un intervallo di sicurezza basando le stime sui casi di lavoro estremi.

Una classificazione dei metodi di manutenzione è stata proposta in letteratura [8] e individua quattro gruppi che saranno brevemente descritti in seguito. Il primo è quello della manutenzione reattiva, chiamati anche metodi *Run-to-Failure* (R2F), la quale prevede di intervenire solo quando un guasto occorre, pertanto non è adatto nei processi dove si vuole massimizzare il tempo di attività delle macchine e le lavorazioni richiedono basse tolleranze. Il secondo gruppo è rappresentato dalla manutenzione preventiva, che ha l'obiettivo di intervenire prima che il guasto occorre, seppure con pianificazioni piuttosto approssimative. Il terzo gruppo è quello della manutenzione predittiva, il cui scopo è quello di rendere più efficiente il processo di intervento rispetto al gruppo precedente, e sarà maggiormente discusso in quanto oggetto dello studio. Infine, ci sono i metodi di manutenzione avanzata, che sono inseriti in un conteso più ampio del sistema produttivo. I metodi di manutenzione hanno seguito un'evoluzione nel tempo parallelamente al progresso tecnologico. Infatti, a partire dai primi modelli di manutenzione preventiva si è cercato di migliorare le performance in termini di riduzione di costi operativi, dovuti principalmente ai costi degli utensili, e minimizzazione dei tempi di inattività delle macchine. Una rappresentazione schematica è

proposta in [9], riportata nella figura 2.1.2, dalla quale si possono cogliere facilmente le differenze dei diversi approcci di manutenzioni, che saranno brevemente descritti in seguito.

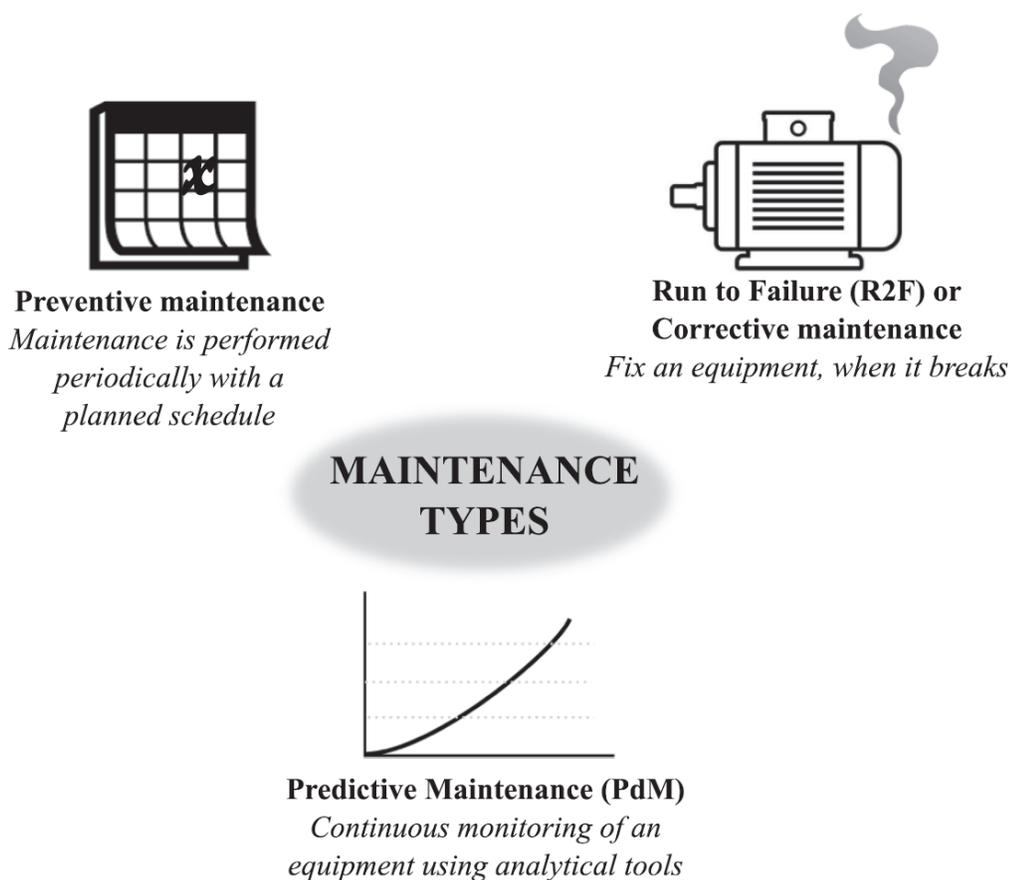


Figura 2.1.2 Raffigurazione tipi di manutenzione [9]

I modelli di manutenzione più semplici si basano solo sul tempo di attività della macchina, schedulando programmi di sostituzione predefiniti in specifici periodi [10], essi sono facili da implementare ma possono condurre a sovrastimare l'attività di manutenzione. Modelli più accurati utilizzano i dati in tempo reale provenienti dalle macchine in attività, essi sono i cosiddetti *Condition Based Maintenance*. Seppure anche tali metodi si basano su schedulazioni pianificate ex-ante, gli interventi effettivi avvengono solo quando è necessario sulla base delle condizioni di monitoraggio, rendendo il processo più efficiente [11]. Essi prevedono due processi principali: diagnostici e prognostici [12]. I primi hanno l'obiettivo di identificare un guasto (tipo, posizione e gravità) [13] il più presto possibile e l'azione che generalmente segue è quella di fermare la macchina. I secondi invece, usano i dati dei sensori per prevedere i guasti così da evitare di fermare la macchina. Tuttavia, se da una parte essi consentono di ridurre i costi operativi e migliorano la reliability del sistema, dall'altra richiedono un consistente investimento iniziale relativo alla strumentazione necessaria per le misurazioni.

L'evoluzione successiva è rappresentata dai modelli di manutenzione predittiva (PdM), in cui le misurazioni sulla macchina sono usate in combinazione con i dati di performance del processo misurati da altri dispositivi. L'uso di tali dati congiuntamente permette ai modelli statistici di analizzare i trend storici al fine di predire l'istante in cui la macchina necessita un intervento [14]. I metodi PdM si definiscono *offline* quando il monitoraggio è periodico e avviene tramite misurazioni dirette che non possono essere effettuate quando la macchina è operativa, mentre sono detti *online* quando invece esso avviene analizzando un flusso di dati continuo proveniente da sensori che effettuano misure indirette durante tutta la vita operativa della macchina. Negli ultimi anni grazie all'evoluzione delle tecnologie di analisi dei *Big Data* si stanno diffondendo maggiormente i metodi *online*, poiché con la corretta interpretazione e analisi dei dati si possono migliorare notevolmente le performance e scoprire tendenze non osservabili con i metodi tradizionali [15]. Un altro approccio che si sta adottando nell'ambito della manutenzione predittiva è la *log-based PdM*. Infatti, le macchine moderne utilizzano software più evoluti che in precedenza e i *log* che rilasciano per attività di *debugging* possono diventare dati utili per algoritmi di intelligenza artificiale per prevedere lo stato della macchina.

Un fattore determinante nei processi di manutenzioni soggetti ad usura dell'utensile è il *Remaining Useful Life* (RUL), pertanto è il parametro di cui generalmente si vuole prevedere con i metodi proposti. I metodi di predizione del RUL possono essere *physics-based* o *data-driven*. I primi sono metodi basati su modelli fisici o empirici che descrivono i processi di degradazione delle macchine, il limite principale di questi modelli è che trattandosi di processi stocastici necessitano la gestione di diverse fonti di incertezza. Mentre i secondi usano i dati forniti dalle misurazioni sulla macchina, in cui la dispersione è insita nei dati stessi, processati da tecniche statistiche o di *machine learning*. Quindi le performance dei metodi *physics-based* dipendono dalla variabilità, le cui principali fonti sono: variabilità temporale, variabilità unit-to-unit, variabilità non-lineare e variabilità di misurazione. Un modello per raggiungere una buona accuratezza dovrebbe includere tutte le fonti di variabilità elencate, tuttavia molti modelli in letteratura ne considerano solo alcune. In particolare, è scarsamente considerata la variabilità non-lineare, che tuttavia gioca un ruolo importante a causa della variazione di velocità di degradazione dell'utensile con l'avanzamento della degradazione stessa. Sono stati proposti dei modelli esponenziali per considerare questo fenomeno, ma sono poco efficaci in quanto essi possono essere linearizzati attraverso una trasformazione logaritmica [14]. Mentre l'accuratezza dei metodi *data-driven* dipende dalla quantità e qualità dei dati storici, i quali sono difficili da replicare per delle analisi di ricerca [16].

2.2 Metodi tradizionali

I metodi della meccanica classica riguardanti l'usura di utensili di macchine rotanti si basano sulla relazione non lineare che c'è tra due principali parametri: velocità di taglio (V) e vita dell'utensile (T). A questa relazione influiscono i parametri di profondità di taglio (b) e di velocità di avanzamento (t). Una prima equazione è stata proposta da Taylor nel 1906 ed ha la seguente forma:

$$VT^n = C$$

Equazione 2.2.1 Relazione base di Taylor

dove (A, B e D sono costanti empiriche):

$$n = 1/B \quad e \quad C = At^{-C/B} b^{-D/B}$$

Equazione 2.2.2 Coefficienti di Taylor base

Sono state proposte diverse forme estese, in cui sono state implementare altre variabili nel modello (come la velocità di avanzamento, che nel modello base rientra nella costante), con l'obiettivo di ridurre l'incidenza della variabilità, che tuttavia richiedono la stima di un numero maggiore di coefficienti [17]. Sebbene l'equazione assume una forma deterministica, le costanti empiriche contengono le incertezze derivate dai dati usati per stimarle, quindi tale modello può essere considerato un processo stocastico.

Diversi studi dimostrano che tali modelli presentano buone performance solo in limitati range di condizioni di lavoro, soprattutto per i parametri di profondità di taglio e per velocità di avanzamento. Si è dimostrato che l'errore standard del modello varia di oltre il 6% con l'incremento dei parametri di lavorazione citati [18]. Ciò rappresenta un importante limite poiché la flessibilità del metodo è una caratteristica importante per consentire il passaggio alle nuove tecnologie, le quali consentono di variare i parametri di lavorazione molto più velocemente rispetto alle macchine meno recenti. Inoltre, sono richiesti molti dati ed esperimenti per calibrare il modello, che lo rendono costoso da implementare nonostante la semplicità della forma. Infine, essi non forniscono ulteriori informazioni sullo stato dell'utensile, come il profilo e l'avanzamento dell'usura, che possono essere input importanti per migliorare le performance del sistema. Ciò rende i modelli basati sulla relazione di Taylor poco robusti per contesti applicativi dinamici, per cui data la loro diffusione possono limitarsi a coprire un ruolo di baseline per sviluppare modelli più complessi [19].

Essi sono ancora usati in combinazione con modelli più evoluti, come nel modello basato sulla funzione *recursive MLE* per stimare la funzione di densità della probabilità di usura proposto da Z. Yan [20], in cui con la legge di Taylor è possibile espandere il modello al variare delle condizioni di lavorazione, in particolare la velocità di taglio e le caratteristiche

dell'utensile. Tuttavia, i parametri della legge di Taylor necessitano di essere stimati opportunamente e un metodo per farlo può essere la regressione basata sui minimi quadrati log-log [21].

Per le lavorazioni di fresatura, di cui si focalizza maggiormente la ricerca di tesi, l'equazione basica di Taylor richiede di stimare quattro esponenti oltre alla costante C, pertanto già con due differenti materiali sono necessarie 32 misurazioni per ottenere tali stime, rendendo il metodo lungo e costoso. Ciononostante, l'equazione basica ha un basso indice di affidabilità (<73%) per utensili con inclusioni di solfuri, i quali sono molto diffusi in questi processi di lavorazione per incrementare la vita utile dell'utensile. È stata proposta un'equazione che include altri due parametri, uno per la percentuale di solfuri e l'altro la *flank wear*, la quale ha mostrato performance nettamente superiori arrivando a un livello di affidabilità del 96%, che possono giustificare il numero ancora più elevato di misurazioni necessarie per stimare i parametri [22].

Date le limitazioni del modello di Taylor nell'applicabilità in contesti con un range più elevato di condizioni di lavoro, Colding (1961) ha proposto relazioni polinomiali che includono più variabili. La prima proposta non ha trovata applicabilità industriale poiché contiene nove costanti, che richiedono un numero eccessivo di misurazioni per essere stimate. La seconda equazione proposta ha ridotto il numero di costanti a cinque elementi e presenta la seguente forma:

$$k + y + bx + cx^2 + dz + hxz = 0$$

Equazione 2.2.3 Relazione di Colding con 5 costanti

dove y è la velocità di taglio, z la vita utile (come già nell'equazione di Taylor) e x è lo spessore del truciolo teorico, tutte espresse in scala log-log. Mentre k, b, c, d, h sono le costanti empiriche da stimare.

Il modello di Colding riesce a descrivere il processo di usura dell'utensile in un range di condizioni di lavoro più ampio rispetto al modello di Taylor. Tuttavia, esso richiede minimo cinque serie di esperimenti separati per stimare le costanti, e ogni serie deve contenere un numero di replicazioni adeguato a ottenere una sufficiente attendibilità statistica. È stato dimostrato [23] che le costanti e l'errore del modello dipendono fortemente dai criteri di usura adottati, quindi anche questo modello presenta dei limiti di flessibilità. Inoltre, tale equazione andrebbe usata solo nel range dei punti in cui sono state effettuate le stime delle costanti, poiché le previsioni al di fuori del range presentano un errore notevolmente più elevato.

Una ricerca successiva [24] ha dimostrato che combinando due modelli, quello di Colding già citato e quello di Archard¹, si può ottenere un modello risultante più robusto. Esso, infatti permette di controllare oltre ai parametri di taglio quelli relativi al materiale di lavorazione (resistenza al taglio) e della geometria dell'utensile (angoli di taglio). Inoltre, esso può essere utilizzato con diversi criteri di *flank wear*, al contrario del modello base di Colding che è molto rigido su questo aspetto. Tuttavia, esso presenta anche alcuni svantaggi, come una minore precisione per bassi valori di criteri di usura e per una maggiore sensibilità al materiale di rivestimento dell'utensile.

2.3 Metodi statistici

I metodi basati sulle previsioni storiche del tempo di vita dei componenti sono tra i più diffusi tra i processi industriali nonostante le loro limitazioni. Infatti, uno studio [1] ha dimostrato che a parità di condizioni operative, i componenti soggetti ad usura possono presentare una vita operativa nettamente diversa. L'esperimento è stato effettuato su un campione di 30 cuscinetti che sono stati portati a rottura, ottenendo valori di durata prima della rottura altamente variabili, da 15 a 300 ore. Per alcuni tipi di componenti, in particolare quelli di natura elettronica, i metodi *time-based* non possono essere adottati. Infatti, come descritto nello studio [34] i guasti possono presentare sei tipi di profili, come riportato in figura 2.3.1,

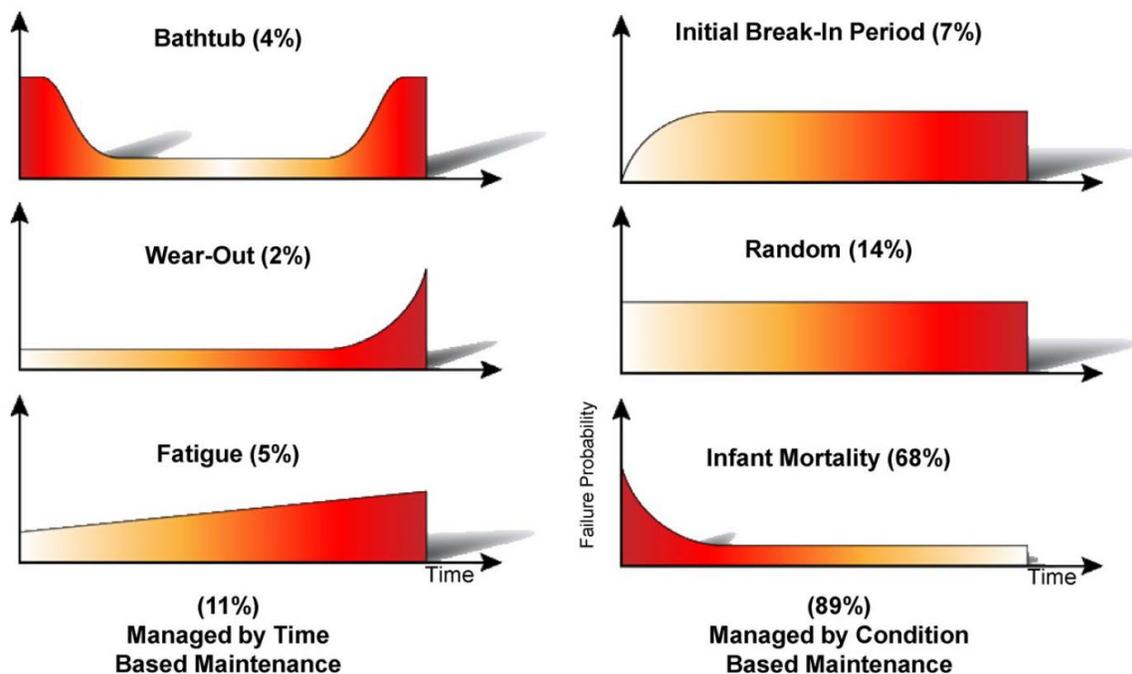


Figura 2.3.1 Curve di guasto [34]

¹ $V=k_0D_2e$, dove k_0 è la costante della funzione di usura, D_2 è la componente di forza, e rappresenta le condizioni geometriche dell'utensile.

e solo la metà di essi possono essere gestiti con tali metodi. Infatti, dallo studio [25] analizzando un campione di componenti, i tre profili di guasti a sinistra (relative ai metodi *time-based*) sono seguiti solo nell'11% dei casi, mentre il restante 89% può essere gestito con approcci CBM.

Gli approcci tradizionali di stima della *reliability* si basano sulla distribuzione delle variabili, di cui si assumono unità identiche nelle popolazioni. Sono stati sviluppati numerosi modelli statistici che seguono questo approccio, con l'utilizzo di diverse distribuzioni come quella di Poisson, l'esponenziale e la *Log-Normal*. Ma la distribuzione più diffusa in questo conteso è la Weibull poiché si adatta meglio ai fenomeni caratteristici di un processo di guasto, come l'andamento di rottura della curva *bathtub*. Tali metodi possono essere utili per particolari contesti dove può essere accettata l'assunzione riguardante l'identità delle unità della popolazione, come nella produzione in larga scala di componenti in serie. Ma in un contesto di manutenzione più sofisticato, come nel caso del processo di usura degli utensili delle macchine rotanti, sono necessarie informazioni più dettagliate sullo stato attuale del componente da monitorare. I metodi basati su serie storiche hanno il vantaggio di non richiedere dati di *condition monitoring* (CM), quindi possono essere usati senza l'ausilio di specifici sensori, ma riescono a fornire solo un valore approssimativo di *mean-time-to-failure* dell'intera popolazione senza fornire informazioni sul componente specifico.

2.3.1 Serie storiche

Con gli approcci *data-driven* si costruiscono modelli previsionali direttamente dai dati forniti da sistemi di CM, costruiti sulla base dei dati storici. In particolare, si usano i dati raccolti di osservazioni passate della variabile da analizzare per sviluppare dei modelli che ne descrivono la relazione nel tempo. L'obiettivo del modello è quindi quello di estrapolare dalle serie storiche le previsioni future. Questo approccio metodologico è particolarmente utile quando non si ha sufficiente conoscenza fisica del fenomeno da monitorare da costruire un modello matematico che mette in relazione le variabili osservate. I metodi più basilari sono i modelli di proiezione semplice, come l'*exponential smoothing*² e il modello lineare autoregressivo³. Essi hanno il vantaggio di essere semplici da implementare e adattabili a qualsiasi calcolatore programmabile. Tuttavia, essi si basano sull'assunzione di base che il processo da monitorare è stabile per poter applicare i trend passati su proiezioni future. Ciò comporta delle stime inaccurate quando si verificano dei cambiamenti, cogliendo il cambio di trend con almeno un periodo temporale di ritardo. Il modello *exponential smoothing* è caratterizzato da una buona robustezza e sono presenti studi per determinati modelli in cui si

² La formulazione più semplice, in assenza di stagionalità, assume la seguente forma:

$$S_t = \alpha X_t + (1 - \alpha)S_{t-1}, \quad \hat{X}_t(m) = S_t$$

³ Formula generale: $z_t = \varphi z_{t-1} \dots + \varphi_p z_{t-p} + \alpha_t$

è dimostrata l'accuratezza delle stime, come nell'ambito delle previsioni di vendita, potendo adattarsi a fenomeni specifici come trend crescenti e stagionalità [26]. Mentre nel contesto della manutenzione predittiva ci sono scarse evidenze delle performance del metodo a causa delle limitazioni citate, che giustificano gli sforzi della ricerca verso modelli basati sull'intelligenza artificiale che verranno discussi in seguito.

Un'applicazione basata sul modello *exponential smoothing* è stata effettuata su un campione di dati relativi alle caratteristiche di cuscinetti a due corone di rulli cilindrici [27]. Esso è stato sviluppato con una sola variabile di osservazione: l'intensità di vibrazione. Il campione contiene 200 valori registrati con un intervallo di 10 minuti. L'intensità di vibrazione parte da valori superiori allo 0 e presenta un trend crescente, a cui è fissata una soglia critica a 0,22 A/g. Le variazioni dei valori registrati presentano dei picchi ripidi, pertanto sono stati scelti valori di α e β elevati (entrambi pari a 0,6) per rendere il modello sufficientemente reattivo. Sono stati testati diversi esperimenti variando il numero di periodi e il periodo iniziale del modello, su cui sono emersi i seguenti risultati: l'accuratezza delle previsioni è migliore per periodi brevi, in cui i valori stimati si avvicinano molto ai valori effettivi. Già a partire da periodi di 10 unità temporali l'accuratezza scende sotto il 50%, perdendo completamente di utilità arrivando a 14 unità. Inoltre, la bontà delle previsioni dipende fortemente dal periodo iniziale in cui parte il modello. Se esso parte da uno stadio avanzato dove sono già presenti delle fluttuazioni tra i valori, le previsioni sono fortemente deviate. Si può concludere che tale metodo può essere applicato per fare delle previsioni a breve termine quando si hanno a disposizione serie storiche complete e accurate, che saranno sfruttate dal modello con pesi crescenti verso i dati più recenti.

2.4 PCA

Nei processi di monitoraggio, quale il CBM, si genera una grande quantità di dati che vanno opportunamente trattati. Infatti, solo una parte di essi porta dell'informazione utile, mentre il resto è di scarso valore a causa delle alte correlazioni tra diversi valori. I metodi basati su PCA (*Principal Component Analysis*) consentono di utilizzare i dati raccolti da diversi sensori, uno scenario che si sta affermando sempre di più con l'industria 4.0, al fine di ridurre lo spazio effettivo di informazioni da analizzare. Il principio alla base di tale metodo è l'analisi, tramite opportune trasformazioni di vettori ortogonali nello spazio dei valori, della correlazione tra i dati provenienti dai sensori. Lo scopo è quello di migliorare la comprensione complessiva dei dati disponibili, difficile da interpretare in modo diretto a causa dell'eterogeneità e quantità dei dati stessi. L'approccio basato sulla PCA può essere usato sia per l'identificazione delle condizioni di intervento, definite in base allo scostamento dalle normali condizioni operative, che per la previsione del tipo e del tempo di rottura di un componente [28].

Un'evoluzione recente dei metodi di previsione sono i *similarity-based life prediction method*, i quali confrontano lo stato di usura misurato con quello di una serie storica di valori, utilizzando tecniche PCA per analizzare i dati. Il principio alla base è che per livelli di usura simili si prevedono RUL simili, così si può stimare il RUL dell'utensile, sulla base delle misurazioni attuali, con una media pesata dei valori di RUL del campione di dati storici [24]. Un'applicazione del modello descritto si trova in letteratura per la stima della vita residua dei cuscinetti a sfera. Ci sono due approcci: il primo prevede di confrontare, seguendo determinate regole, i valori misurati con tutti i dati storici e la stima della vita residua sarà rappresentata dal dato storico con la migliore corrispondenza; il secondo invece, prevede di confrontare i valori misurati con la stessa sequenza di osservazione dei dati storici e da cui con l'analisi di similitudine si calcola la differenza di RUL per ogni coppia, la quale verrà usata per affinare i dati storici per poi effettuare le previsioni desiderate. I metodi per stimare la vita rimanente sono diversi, tra cui si cita quello proposto da H. Wang [30] che prevede l'uso della PCA per sintetizzare le diverse variabili misurate sul componente e successivamente una funzione di correlazione incrociata per calcolare le similitudini con i dati storici. Un altro esempio di applicazione riguarda la stima del RUL per i motori aerei, in cui si effettua la *feature selection* con l'algoritmo Relief⁴, si usa la PCA per la *feature extraction* e infine si calcola la *feature trajectory smoothing* con l'algoritmo Kernel⁵ [29]. Tuttavia, va sottolineato che il metodo delle similitudini per ottenere una buona accuratezza necessita di disporre dei dati storici sull'intero ciclo vita del componente da monitorare [30].

2.5 Metodi basati su algoritmi di Machine Learning

Gli algoritmi di *machine learning* (ML) sono un'applicazione di intelligenza artificiale, i quali sono caratterizzati dalla capacità di evolvere autonomamente a partire da un training iniziale. Tali approcci consentono di gestire dati multivariata e di grandi dimensioni, estrarre relazioni nascoste all'interno dei dati e di lavorare in ambienti dinamici e complessi. Essi sono usati in molti ambiti, dal riconoscimento vocale alla robotica, oltre che nel caso in esame della manutenzione predittiva. Tali algoritmi possono essere classificati in tre categorie: supervisionati, in cui vengono fornite sia le variabili di risposta che le etichette a cui appartengono; non supervisionati, che si basano solo sulle variabili di risposta di cui devono

⁴ Algoritmo progettato in origine per l'applicazione a problemi di classificazione binarie con caratteristiche discrete o numerici. Esso calcola un punteggio per ogni caratteristica al fine di effettuare un ranking.

⁵ I metodi *kernel* si avvicinano al problema mappando i dati in uno spazio di caratteristiche multidimensionali, dove ogni coordinata corrisponde a una caratteristica dei dati dell'elemento, trasformando i dati in un insieme di punti dello spazio euclideo.

formare dei cluster; per rinforzo, i quali sono implementati direttamente nel sistema ed evolvono con i cambiamenti dell'ambiente. La disponibilità di informazioni dipende dal metodo di manutenzione preesistente su cui va sviluppato il modello di ML. Nel caso di un approccio R2F è possibile adottare un algoritmo supervisionato grazie alla disponibilità di dati derivanti da un processo ciclico. Infatti, un algoritmo supervisionato richiede un dataset composto da n iterazioni di una coppia $\{x_i, y_i\}$ di osservazioni che contengono le informazioni di ogni iterazione i -esima del processo. Il primo termine, x_i , è un vettore che contiene i valori delle variabili in esame del processo, mentre y_i è il vettore degli output di tali variabili. Se y assume valori continui si è in presenza di un problema di regressione, come nel caso della previsione del RUL. Mentre per valori discreti di y si ha un problema di classificazione, tipico dei processi che possono assumere due o più stati di attività [31]. Le fasi che sono ricorrenti nei metodi ML sono: selezione dei dati storici, chiamato anche acquisizione dei dati in ambito PdM, che ha lo scopo di ottenere i dati necessari per l'analisi; segue la fase di pre-processamento dei dati, che include la trasformazione, la pulizia e la riduzione dei dati stessi; dopo aver ottenuto i dati nella forma desiderata va selezionato il modello adeguato, che quindi dopo la fase di training richiede una fase di validazione; infine, quando è stato scelto il modello da adottare segue la fase di manutenzione durante l'intero periodo operativo [32].

Dallo studio [9] si evince che gli approcci basati sull'intelligenza artificiale (AI) stanno sostituendo i metodi tradizionali. Infatti, gli approcci *model-based* richiedono conoscenze meccaniche del fenomeno, mentre quelli statistici necessitano di avanzate conoscenze matematiche del processo. Mentre con l'AI si superano questi limiti e uno studio [33] ha dimostrato che riesce a ottenere performance migliori rispetto ai metodi tradizionali. Di conseguenza la ricerca sta avendo un interesse crescente verso gli approcci ML in ambito PdM, infatti fino al 2013 sono stati pubblicati⁶ solo due articoli a riguardo, mentre tra il 2013 e il 2018 ne sono stati pubblicati oltre 30. Tuttavia, sono state effettuate ancora poche applicazioni industriali a causa della complessità di integrazione con gli ambienti di produzione attuali [34]. Dagli articoli pubblicati emerge che l'algoritmo più diffuso (33% dei casi) è il Random Forest (RF), seguito dai metodi basati sulle reti neurali (27%), e la restante parte dagli algoritmi Support Vector Machine (SVM) e k-means [9]. Le principali caratteristiche di tali algoritmi saranno presentate singolarmente in seguito.

Gli algoritmi RF sono di tipo supervisionato e possono essere implementati sia per problemi di regressione che di classificazione. Il principio alla base è lo sviluppo di un insieme di alberi decisionali casualizzati, che costituiscono una "foresta", e le previsioni sono date da una media degli output forniti da ognuno. Essi presentano buone performance quando il numero di variabili è più grande rispetto al numero di osservazioni disponibili. Infatti,

⁶ Sui portal IEEEXplore Digital Library e Science Direct.

rispetto agli approcci *deep decision trees*, con i metodi RF si evita di incorrere in overfitting perché lavorano con alberi più piccoli formati da sottoinsiemi casuali [35]. Tali metodi sono tra i più usati perché rispetto agli altri metodi di ML essi consentono di generalizzare il modello e fornire delle stime robuste [36]. I principali svantaggi sono relativi alla complessità del metodo e al tempo computazionale richiesto per eseguirlo.

Le tecniche basate su *Artificial Neural Network* (ANN) sono quelle più studiate negli ultimi anni per lo sviluppo di modelli di manutenzione predittiva per le macchine utensili in rotazione [5]. Un ANN consiste in uno strato di nodi input e uno di nodi output, uno o più strati di nodi nascosti e da pesi di collegamento tra i vari nodi. Il network deduce la funzione incognita che relaziona i dati regolando i pesi su osservazioni ripetute di input e output. In letteratura sono presenti numerosi studi che dimostrano le potenzialità delle ANN, tra cui si citano: la maggiore rapidità funzionamento rispetto alle tecniche multivariata tradizionali; performance di accuratezza migliori rispetto ai metodi statistici tradizionali, con la possibilità di operare senza assunzioni sulla distribuzione statistica della funzione; capacità di catturare fenomeni complessi anche in assenza di conoscenza a priori [37]. Mentre una delle principali limitazioni delle tecniche ANN è la scarsa trasparenza di come evolve il network e di conseguenza di come prende decisioni. Ciò è dovuto dalla complessità della struttura del network necessaria per modellizzare fenomeni altrettanto complessi, e come riporta [34] la trasparenza di un modello è inversamente proporzionale alla sua complessità. Sono state sviluppate diverse tecniche basate su ANN, come il *deep learning* e la *Convolutional Neural Network* (CNN). Il principio della prima riguarda il *learn* dei dati su diversi livelli gerarchici, che permette di implementare funzioni complesse che mappano i dati di input direttamente sull'output. Mentre la tecnica CCN è una classe particolare di *deep learning* che integra i dati provenienti da sensori che misurano diversi parametri. Seppure quest'ultime tecniche sofisticate presentano ottime performance in determinati ambiti, esse necessitano di conoscenze avanzate nella selezione dei dati [38].

I metodi SVM sono algoritmi supervisionati usati per analisi di regressione e per *pattern recognition*. Il principale vantaggio dei metodi SVM risiede nell'elevata accuratezza nella separazione di diverse classi di dati e nell'individuazione del punto ottimale di separazione. Tuttavia, essi non sono di facile interpretazione e dipendono molto dalla funzione *kernel* scelta, oltre che a richiedere un elevato tempo di training proporzionale alla dimensione del campione [39]. Un esempio di applicazione è stato proposto da Praveenkumar [40] per identificare i guasti nei sistemi di trasmissione delle autovetture sulla base dei segnali di vibrazione. Il modello è in grado di classificare il tipo di guasto con una precisione superiore al 90%.

I modelli *k-means* invece sono molto diffusi nelle tecniche non supervisionate per determinare un set di partizioni. L'obiettivo di tale algoritmo è di trovare k partizioni in cui sono raggruppati i dati affini e sono separati quelli diversi. Esso è facile da implementare e

presenta delle buone performance anche con grandi quantità di dati, minimizzando la varianza all'interno delle classi e massimizzando quella tra esse. Mentre le criticità principali dell'algoritmo in esame sono la difficoltà di determinazione il numero di partizioni (k) e la sensibilità nell'ordine di inserimento dei dati [41].

Uno degli algoritmi più usati per l'analisi di dataset di grandi dimensioni è *gradient boosting* (GB). Esso è un modello che prevede una fase di training progressiva con un aumento di complessità, in modo da massimizzare l'accuratezza delle revisioni. Con le previsioni fatte in sequenza il modello è in grado di evolvere sugli errori precedenti anziché basarsi su un processo *bootstrap*. Gli step previsti dall'algoritmo sono i seguenti:

1. Assegnare a ogni osservazione x_i un peso $w_i=1/n$, n pari al totale delle osservazioni.
2. Effettuare un training "debole" con i pesi iniziali.
3. Per ogni osservazione: se il modello prevede x_i correttamente aumentare w_i , altrimenti diminuirlo.
4. Effettuare un training con i nuovi pesi.
5. Reiterare da step 3 fino a raggiungere tutte le previsioni x_i corrette.

Esso è particolarmente utile per analizzare dati ordinati e dati categorici quando è difficile descrivere il fenomeno con modelli matematici [42].

Spesso i metodi *condition-based* dispongono solo di misurazioni non invasive, che nel caso degli utensili di taglio sono generalmente dei sensori che misurano l'intensità di vibrazione. Tuttavia, tali misurazioni sono difficili da interpretare poiché le vibrazioni prodotte dall'utensile in esame sono di intensità molto inferiore rispetto a quelle prodotte da altre fonti della macchina (come il motore). Quindi si ottengono dei dati molto distorti dal rumore di fondo e per far emergere il trend monotono relativo all'usura dell'utensile è necessaria una fase preliminare di trattamento dei dati. È stato proposto un algoritmo basato sulla funzione di autocorrelazione (ACF), sfruttando l'intuizione che i valori delle vibrazioni seguono un andamento ciclico con la frequenza periodica di sostituzione. Tale algoritmo può essere integrato con un processo di manutenzione basato su ML, quindi non periodico, così da migliorare le performance complessive grazie alla rimozione del rumore dai dati [43].

2.6 Metodi ibridi

Come è stato mostrato nei paragrafi precedenti ogni metodo è caratterizzato da punti di forza e di debolezza, la cui scelta dipende fortemente dal contesto, tra cui il livello di conoscenza del fenomeno e la quantità di dati a disposizione, in cui deve essere applicato. I metodi che

si prestano meglio ai processi di manutenzione predittiva sono quelli basati sulle serie storiche e i più recenti basati sulle reti neurali di intelligenza artificiale.

Tra i modelli di serie storiche il più diffuso è il modello ARIMA, il quale assumendo una struttura di correlazione di tipo lineare tra i valori delle serie storiche non è adatto a interpretare pattern di tipo non lineare. Gli algoritmi basati sulle reti neurali (ANN) invece non richiedendo particolari assunzioni sulle relazioni dei dati, sono molto flessibili nelle previsioni di pattern non lineari [44]. Tuttavia, nel caso di pattern lineari i metodi ANN mostrano risultati contrastanti ragione per cui si usano ancora i metodi di serie storiche come benchmark [45].

Ci sono molte ragioni per adottare un approccio ibrido, in particolare tra i due modelli sopracitati. La prima, è che nella pratica non è sempre facile determinare se una serie storica segue un pattern lineare o non lineare, rendendo ardua la scelta del metodo. Tipicamente per scegliere il metodo da adottare si testano diversi modelli su un campione di dati storici, per poi scegliere quello che ha mostrato le performance migliori con le previsioni sul campione di test. Tuttavia, la scelta del modello è fortemente influenzata dal campionamento dei dati utilizzati, dall'incertezza dei modelli e dai cambiamenti della struttura. Inoltre, va considerato che i dati reali raramente seguono un trend puramente lineare o non lineare, rendendo inaccurati sia i metodi ARIMA che quelli ANN. Infatti, anche se quest'ultimi sono più flessibili nella modellazione, non possono ottenere risultati ottimali in presenza di dati che presentano una componente lineare e una non lineare.

Si è dimostrato, per le ragioni citate, che nell'ambito della previsione non esiste un metodo migliore che si adatti ad ogni situazione [46]. Mentre altri studi [47] dimostrano che combinando diversi metodi è possibile migliorare l'accuratezza delle previsioni rispetto ai modelli individuali, creando un metodo che più si avvicina all'ottimo ricercato. Nel caso in esame combinando il metodo ARIMA con quello ANN è possibile ottenere un metodo che sfrutta le caratteristiche del primo per l'analisi della componente lineare della serie di dati e quelle del secondo per le componenti non lineari.

2.6.1 Combinazione tra ARIMA e ANN

Sulla base delle caratteristiche dei due modelli, ARIMA e ANN, è stato proposto da Zhang [48] un metodo ibrido che sfrutta le entrambe le potenzialità. L'intuizione su cui si basa il metodo è stata quella di scomporre i dati delle serie storiche in due componenti, come mostra la seguente equazione:

$$y_t = L_t + N_t$$

Equazione 2.6.1 Suddivisione componenti di una serie storica

dove per ogni osservazione y è data dalla somma della componente lineare L con quella non lineare N . Con la regressione lineare è possibile stimare L , da cui si possono calcolare i residui, indicati con la lettera e , come segue:

$$e_t = y_t - \hat{L}_t$$

Equazione 2.6.2 Relazione residui con componente lineare

Dai residui calcolati con le previsioni del metodo ARIMA è possibile analizzare la relazione non lineare, se è presente, nei dati della serie storica. L'analisi può essere fatta a partire da un insieme di residui e_i di dimensione n , il quale andrà a costituire i nodi input del modello ANN:

$$e_t = f(e_{t-1}, e_{t-2}, \dots, e_{t-n}) + \varepsilon_t$$

Equazione 2.6.3 Funzione residui di regressione

dove f è la funzione non lineare da determinare con il metodo ANN e ε_t è l'errore casuale.

Il metodo ibrido è stato testato su diversi dataset, relativi a serie storiche con andamenti difficili da classificare come lineari o non lineari. Applicando i metodi ARIMA e ANN sono state osservate dei limiti nelle previsioni da entrambi, il primo nel lungo periodo e il secondo nel breve, dimostrando le debolezze dei metodi singoli per serie storiche con andamenti complessi. Parte di tali limiti sono stati superati con il metodo ibrido, che ha registrato un miglioramento in termini di errore percentuale del 16,13% rispetto al metodo ARIMA e del 9,89% rispetto a quello ANN, con una buona accuratezza sia nel breve che nel lungo periodo.

3 CAPITOLO

FENOMENI DI USURA DELL'UTENSILE NEI PROCESSI DI LAVORAZIONE PER ASPORTAZIONE DI MATERIALE

3.1 Introduzione

I processi di lavorazione meccanica per taglio prevedono l'asportazione di materiale dal pezzo di lavorazione. Generalmente la porzione di materiale asportato è relativamente bassa rispetto al pezzo di lavorazione e viene chiamata comunemente truciolo [71]. Tra i processi principali di questo tipo di lavorazione ci sono la tornitura e la fresatura, di cui quest'ultima verrà approfondita maggiormente essendo quella a cui fa riferimento il dataset utilizzato per le analisi della ricerca di tesi in esame.

In questo capitolo, nel paragrafo 3.2, sarà introdotto brevemente il processo di lavorazione della fresatura, il quale è rappresentativo e di supporto per la comprensione dei sistemi in cui si vogliono studiare i metodi di manutenzione predittiva. Nei due paragrafi successivi, 3.3 e 3.4, sono riportati i principali tipi di usura che si possono osservare in questi processi di lavorazione (paragrafo 3.3) e quali sono i meccanismi fisici e chimici che portano a questi tipi di usura (paragrafo 3.4).

3.2 Cenni sul processo di fresatura

La fresatura è un processo di lavorazione meccanica per asportazione di truciolo, in cui dal pezzo di lavorazione il materiale è rimosso mediante un utensile rotante detto fresa come rappresentato nella figura 3.2.1. La fresa è composta da inserti multipli, i quali asportano del

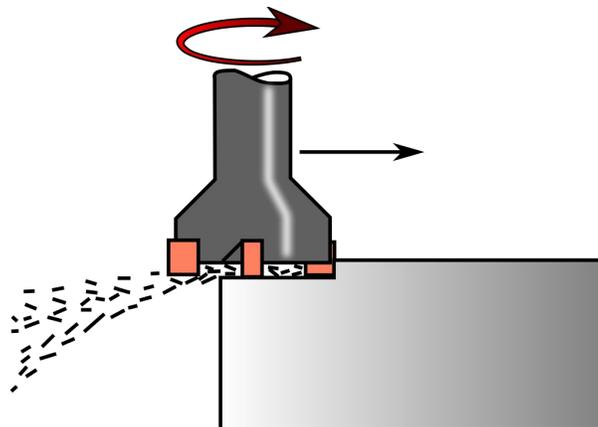


Figura 3.2.1 Rappresentazione fresatura [74]

materiale dal pezzo di lavorazione con tagli ripetuti per ogni giro dell'utensile [74]. A seconda della posizione del materiale rispetto all'asse di rotazione le due operazioni principali sono la fresatura periferica e quella frontale. Mentre si distingue tra fresatura concorde e discorde a seconda del verso di avanzamento dell'utensile, nel primo caso nella stessa direzione della rotazione e nel secondo nella direzione opposta [72].

Nella fresatura periferica realizzata in discordanza la formazione del truciolo è ottenuta con avanzamento del pezzo in direzione contraria al senso di rotazione dell'utensile. La sezione del truciolo, rispetto all'azione progressiva di taglio, è nulla all'inizio e massima al momento del distacco. Questo metodo di fresatura richiede una centratura della fresa molto accurata ed assorbe, specialmente nel caso di larghe superfici da fresare, una notevole quantità di energia. La fresatura periferica inoltre sollecita sensibilmente il mandrino della fresatrice, perché la resistenza opposta all'azione di taglio è in continua oscillazione [73].

Il sistema di fresatura frontale sfrutta, invece, le proprietà taglienti delle frese aventi l'asse di rotazione normale al piano da fresare. I taglienti frontali della fresa svolgono normalmente solo la funzione di lisciare la superficie già fresata dai taglienti periferici. Il rendimento del sistema di fresatura frontale in genere è superiore a quello periferico, sia come qualità della superficie lavorata, sia come quantità di materiale asportabile nella unità di tempo. Infine, quest'ultimo metodo risulta più economico perché permette di utilizzare, anche per larghe passate, frese relativamente piccole leggere [73].

L'elemento caratterizzante del processo di fresatura è l'utensile, poiché è la parte che interagisce direttamente con il materiale. Esistono diversi tipi di frese, che si distinguono per il tipo di materiale e per la geometria degli inserti. I materiali usati per gli inserti presentano elevata durezza e resistenza alle alte temperature, poiché durante il processo di lavorazione si genera una grande quantità di calore. Gli inserti più economici sono costruiti con acciaio ad alta velocità, mentre quelli più resistenti all'usura usano metalli con carburi cementati [74]. Gli inserti sono fissati seguendo una forma elicoidale a corpo cilindrico dell'utensile, il quale può essere inserito nel mandrino del motore che genera il movimento di rotazione. Infine, l'utensile esegue parte del movimento orizzontale (asse Z) per la regolazione della profondità di taglio, mentre i movimenti nel piano (assi X e Y) sono generalmente eseguiti dal piano in cui è bloccato il pezzo di lavorazione [71].

3.3 Tipi di usura

Il modo in cui avviene l'usura dell'utensile, e la parte che si usura di più di esso, varia molto a seconda delle condizioni di taglio, del materiale e delle specifiche di lavorazione. In seguito, facendo riferimento alla classificazione proposta da Cook [70], saranno descritte brevemente

i principali tipi e meccanismi di usura per i processi di lavorazione per asportazione di materiale mediante il taglio con utensili rotanti. Comprendere i tipi di usura che influenzano maggiormente la qualità resa dall'utensile è fondamentale per definire il processo di manutenzione predittiva e stimare la vita rimanente dell'utensile. Tuttavia, è importante fare la distinzione tra l'usura dell'utensile e la vita utile dell'utensile. Infatti, a partire dai dati storici e dalle relazioni fisiche, come si vedrà nei capitoli successivi, generalmente è possibile predire lo stato di usura dell'utensile. Ma ciò non si può tradurre automaticamente in una stima della vita utile poiché quest'ultima dipende fortemente dai requisiti di lavorazione e, nella pratica, un utensile è sostituito quando esso non è più in grado di produrre delle parti che soddisfino i requisiti progettuali. Quest'ultimi possono variare molto a seconda del tipo di pezzo, i quali includono le tolleranze sulle dimensioni, sulla rugosità superficiale ed altri parametri progettuali. Per questi motivi la vita utile dell'utensile è un indicatore che deriva dalle considerazioni sullo stato di usura, ma è relazionato alle condizioni di lavoro e ai requisiti esterni. Esso non può essere stimato a priori e viene comunemente usato per comparare diversi materiali e condizioni di lavoro, quantificando le differenze in termini di lavorazioni effettuabili con lo stesso utensile [72].

L'usura sul fianco (*flank wear*) dell'utensile è il tipo di usura di maggiore interesse per i processi di lavorazione analizzati. Essa è una zona di usura che si verifica in modo piuttosto uniforme sul fianco del tagliente dell'utensile come mostrato nella figura 3.3.2 (a). Il meccanismo principale che causa questo tipo di usura è l'abrasione tra l'utensile e il materiale. La propagazione della banda di usura segue l'andamento rappresentato nella figura 3.3.1, dove nella prima fase si verifica un aumento accelerato, per poi aumentare in modo

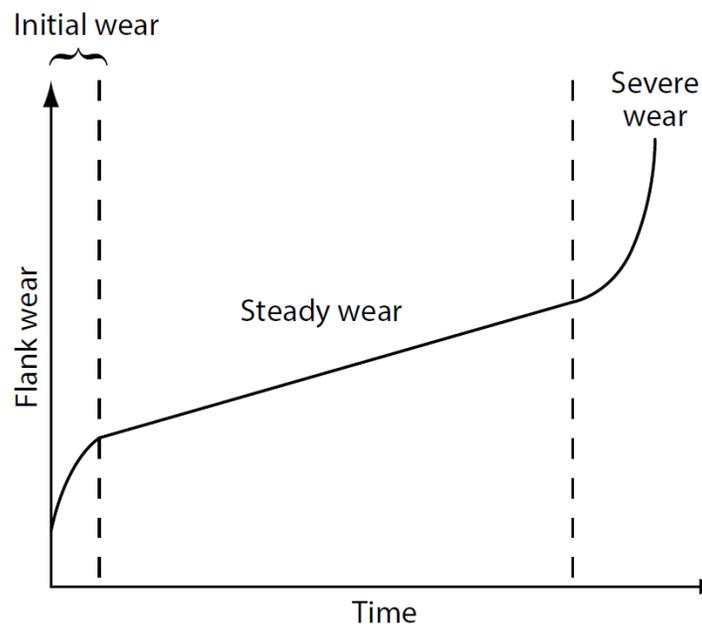


Figura 3.3.1 Andamento nel tempo della flank wear [72]

costante e lentamente per gran parte del fenomeno e infine, dopo aver superato una determinata soglia critica, l'aumento della banda di usura diventa esponenziale. In condizioni di basse velocità di lavorazione è il processo di usura che predomina e quindi regola la vita utile dell'utensile. Essa può essere misurata solo quando la macchina è ferma, mediante metodi ottici, come il microscopio, che rivelano la lunghezza della banda di usura. Generalmente sono misurate due grandezze, VB_{avg} che rappresenta lo spessore medio della banda di usura e VB_{max} che è definito dalla normativa ISO 3685 come il massimo spessore della sezione centrale della banda di usura [75]. A seconda del tipo di utensile e di tolleranza richiesta dalla lavorazione il livello massimo di usura, misurato con la lunghezza della banda VB_{max} , è compreso tra 0,25 mm e 1,3 mm. Mentre per applicazioni che richiedono basse tolleranze si impiegano interferometri laser per ottenere una mappatura 3-D dell'usura in modo da assicurare una finitura superficiale con alta accuratezza [76].

Dalla figura 3.3.2 (b) è possibile vedere anche il secondo tipo di usura, il cosiddetto *cratere* (*crater wear*). Esso è una sezione concava all'interno della faccia dell'utensile, la quale si forma a causa dei contatti caratterizzati da elevati stress meccanici e alte temperature. Esso si presenta con una curvatura che corrisponde al raggio di curvatura del truciolo che raggiunge la massima profondità in un nella zona interna della faccia dell'utensile, lontano dalla parte tagliente a contatto con il materiale. A basse velocità è trascurabile rispetto all'usura sul fianco, tuttavia all'aumentare della velocità il *cratere* può avanzare molto rapidamente e intersecare l'usura sul fianco compromettendo lo stato dell'utensile. Infatti, l'avanzare di questo tipo di usura può indebolire la struttura meccanica dell'inserto fino al

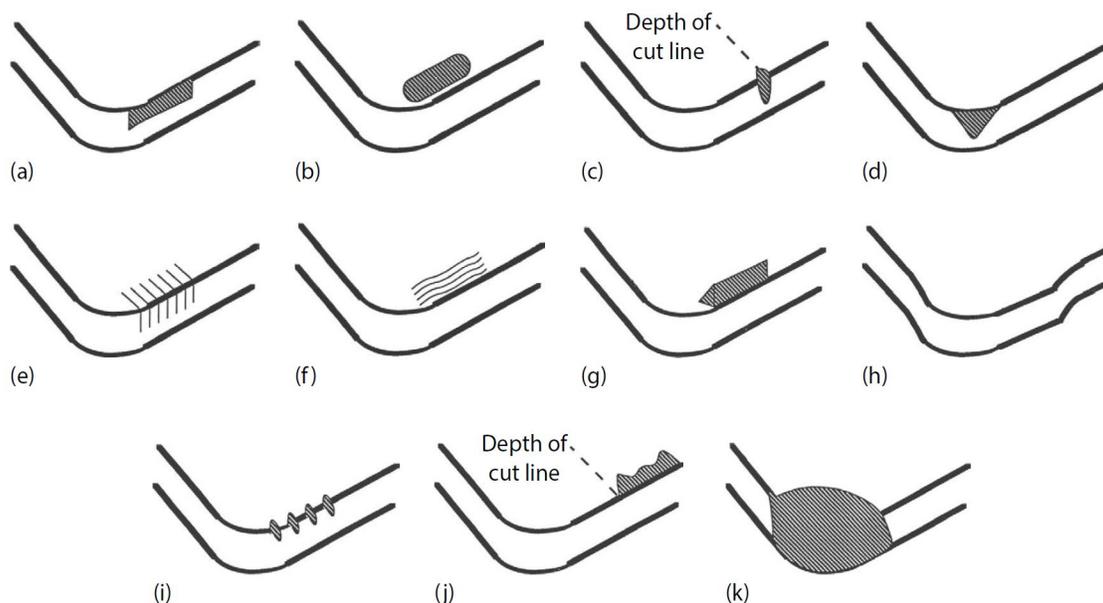


Figura 3.3.2 Tipi di usura: (a) flank wear; (b) crater wear; (c) notch wear; (d) nose radius (e) comb cracks; (f) parallel (mechanical) cracks; (g) built-up edge; (h) gross plastic deformation; (i) edge chipping; (j) chip hammering; (k) gross fracture [72].

verificarsi di deformazioni e fratture dell'utensile. L'andamento della dimensione del *cratere* è simile a quello dell'usura sul fianco, il cui avanzamento può essere ridotto stabilizzando chimicamente il materiale dell'utensile.

Un altro tipo di usura frequente negli utensili da taglio è la formazione di crepe (*cracks*) a causa di stress meccanico e termici. Essa si verifica all'avanzare dei cicli ripetuti di carico dell'utensile e quando tra il tagliente e il materiale si generano alte temperature. Le crepe possono generarsi sia in modo perpendicolare al bordo tagliente, soprattutto quando si utilizza un liquido di raffreddamento, che in modo parallelo ad esso come rappresentato nella figura 3.3.2 (e ed f). Quando si verifica questo fenomeno è importante monitorarlo poiché la presenza di crepe può portare a una rapida frattura dell'utensile o alla scheggiatura del materiale.

Infine, ci sono altri tipi di usura che impattano solo marginalmente sulla qualità di lavorazione dell'utensile. Tra queste si citano l'usura del raggio del naso (*nose radius wear*), la quale si verifica per propagazione dell'usura sul fianco dall'inserto fino al corpo dell'utensile principalmente a causa di meccanismi di abrasione e di corrosione, e la formazione di tacche sulla faccia dell'utensile (*notch wear*). Quest'ultimo tipo si verifica soprattutto in presenza di materiali che generano truciolo duro, come l'acciaio inox e le leghe contenenti nichel, ed è accentuato in caso di lubrificanti corrosivi. Mentre gli altri tipi di usura, riportati solo graficamente nella figura 3.3.2, tipicamente interessano solo particolari tipi di lavorazione e pertanto non sono stati approfonditi come quelli presenti in questo paragrafo.

3.4 Meccanismi di usura

I tipi di usura descritti nel paragrafo precedente sono facilmente osservabili nei comuni processi di lavorazione, nonostante essi derivino da meccanismi fisici che spesso avvengono a livello microscopico. Infatti, in determinate condizioni di lavoro, in particolare a causa delle alte velocità di taglio, si raggiungono temperature fino a 1000° C da cui si possono verificare diversi fenomeni che influenzano la qualità dell'utensile. In seguito, saranno elencati e brevemente descritti i principali meccanismi che generano l'usura dell'utensile nei processi di taglio:

- Usura per adesione: fenomeno in cui dei piccoli frammenti del pezzo di lavorazione possono fondersi e aderire sull'utensile alterandone la forma. Ancora più dannoso è il caso in cui i frammenti in precedenza aderiti si staccano, portando via anche parte del materiale dell'utensile. È un meccanismo che progredisce molto lentamente, tuttavia contribuisce nella

formazione dell'usura del *cratere*. Nel caso di lavorazioni a basse velocità di taglio di materiali morbidi, come le leghe di alluminio, tale meccanismo può portare al tipo di usura chiamato *edge build-up* rappresentato nella figura 3.3.2 (g) dove si presenta un grande accumulo di materiale sull'utensile. Quando lo strato raggiunge determinate dimensioni l'utensile è da sostituire perché l'alterazione della forma derivante dallo strato di accumulo altera le capacità di taglio in profondità dell'utensile.

- Abrasione: meccanismo di usura che avviene a causa della frizione tra le particelle dure di materiale che si forma durante la lavorazione che si frappongono tra il materiale stesso e l'utensile rimuovendo particelle dal tagliente. Tale fenomeno può essere ridotto mediante l'uso di lubrificanti, che oltre a ridurre la temperatura rimuovono la presenza di particelle sulle superfici di contatto [77]. L'abrasione si verifica soprattutto sulla superficie del fianco dell'utensile ed è pertanto la causa principale dell'usura sul fianco (*flank wear*). Nel caso in cui le particelle dure che causano l'abrasione sono trascinate dal fluido di taglio tale meccanismo è chiamato *usura erosiva*.
- Usura per diffusione: ad elevate temperature gli atomi delle strutture cristalline, del materiale e dell'utensile, possono migrare da un metallo all'altro alterandone le strutture. In particolari circostanze, tale fenomeno può portare a una riduzione della durezza dell'utensile aumentando il rischio di frattura. Inoltre, con il meccanismo della diffusione si indebolisce molto l'area in cui si è formato il *cratere*, aumentando il rischio di frattura dell'utensile.
- Ossidazione: fenomeno che avviene per i componenti dell'utensile che sono esposti all'aria atmosferica. Generalmente le zone più sensibili all'ossidazione sono quelle ad elevata temperatura vicine al bordo del tagliente e si identificano facilmente dalla tonalità cromatica che appare meno vivace. Tale fenomeno si può verificare anche sul materiale di lavorazione, con la conseguente generazione di particelle di ossido che possono contribuire al meccanismo dell'abrasione.
- Corrosione: meccanismo di usura che avviene a causa delle reazioni chimiche che si possono generare tra le componenti dell'utensile, il fluido lubrificante e il materiale di lavoro. Esso contribuisce alla propagazione dell'usura sul fianco e a quella sul *cratere*, tuttavia per gli utensili comuni è un fenomeno generalmente trascurabile. Mentre il fenomeno può diventare di maggiore

importanza per utensili costituiti da materiali altamente reattivi come le leghe in titanio [78].

4 CAPITOLO

SERIE STORICHE

4.1 Introduzione

Una serie storica è un insieme di osservazioni ordinate cronologicamente. Generalmente, i punti di tali serie sono raccolti con intervalli regolari di tempo. L'analisi delle serie storiche è il processo di estrazione di pattern e informazioni utili dalle serie temporali, mentre la previsione dalle serie storiche ha l'obiettivo di predire i valori futuri sulla base delle osservazioni passate.

I metodi basati sulle serie storiche si differenziano dagli altri modelli predittivi supervisionati per due aspetti. Il primo è la presenza del tempo come principale predittore, e l'analisi si propone di interpretare come la variabile di interesse varia nel tempo. Il secondo è che l'analisi delle serie storiche può essere applicata anche in assenza di dati su altri attributi, come nel caso delle previsioni con singola variata che si basa solo sul tempo. Tuttavia, con l'implementazione di altre variate anche le previsioni delle serie storiche diventano più efficaci, tali metodi sono detti serie storiche multivariata [49].

Il capitolo è composto da tre paragrafi oltre all'introduzione: il primo presenta una classificazione generale dei principali modelli usati per le serie storiche, con le relative caratteristiche; nel secondo è presentato nel dettaglio il metodo ARIMA, tecnica che verrà presa come riferimento nella ricerca di tesi; infine, nel terzo paragrafo, è proposta una procedura di implementazione in R e le basi statistiche su cui si fonda.

4.2 Classificazione dei metodi di previsione basati sulle serie storiche

I metodi di previsione basati sulle serie storiche possono essere classificati in quattro categorie: previsioni basati sulla decomposizione delle serie storiche, tecniche *smoothing-based*, tecniche *regression-based* e infine le tecniche *machine learning-based* come mostrato nella figura 4.2.1 [50].

La decomposizione delle serie storiche è il processo di suddivisione della serie temporale nei componenti principali che la compongono e che possono spiegare il fenomeno rappresentato. I componenti rappresentativi delle serie storiche sono: il trend, la stagionalità e il rumore. I primi due, trend e stagionalità, sono anche chiamati componenti sistematici

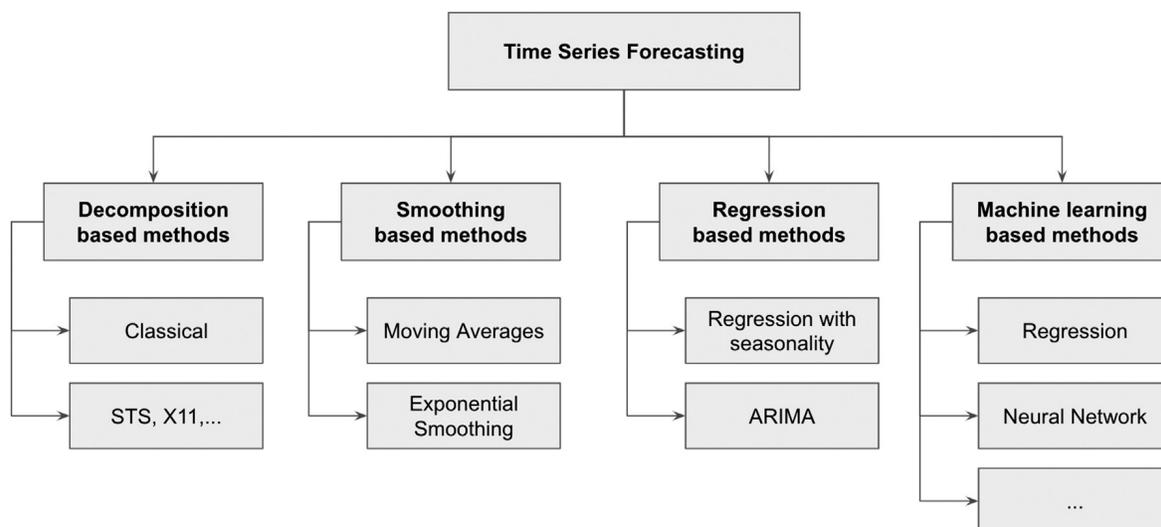


Figura 4.2.1 Classificazione metodi di serie storiche [50]

perché possono essere predetti. Mentre il rumore è per definizione casuale, pertanto è un componente non sistematico. I componenti descritti possono essere stimati singolarmente mediante altre tecniche, come la regressione, per poi essere aggregati e costituire il modello di previsione per decomposizione.

Le tecniche *smoothing-based* si basano sul principio di effettuare le previsioni utilizzando le osservazioni passate in ordine di importanza decrescente, dando maggiore peso a quelle più recenti. Lo smorzamento può essere una semplice combinazione lineare, come nel caso della media pesata, che può essere mobile in modo da includere ad ogni iterazione un numero fisso di osservazioni partendo sempre dalla più recente. Il modello basilico più diffuso di questa categoria è lo smorzamento esponenziale, il quale necessita di un solo parametro per mettere in relazione tutte le osservazioni passate. La relazione è la seguente:

$$F_{t+1} = \alpha y_t + \alpha(1 - \alpha)y_t + \alpha(1 - \alpha)^2 y_{t-2} + \dots + \alpha(1 - \alpha)^T y_{t-T}$$

Equazione 4.2.1 Smorzamento esponenziale

dove la previsione F_{t+1} è data dalla combinazione delle osservazioni y_{t-i} passate, pesate da un coefficiente α che diminuisce esponenzialmente [3]. Maggiore è α e minore è il peso attribuito alle osservazioni più lontane nel tempo, fino ad arrivare al caso limite con $\alpha=1$ in cui la previsione futura sarà uguale a l'ultima osservazione presente. Tuttavia, con un solo parametro non è possibile descrivere una serie storica che presenta trend o stagionalità. In tali casi può essere utilizzata il modello dello smorzamento esponenziale a due parametri proposto da Holt che include il componente del trend, oppure a tre parametri per includere anche il componente della stagionalità.

I metodi basati sulle regressioni hanno le caratteristiche dei modelli predittivi supervisionati, con la particolarità che hanno il tempo tra le variabili indipendenti [4]. La forma più semplice è la regressione lineare, con una forma del tipo:

$$y_t = a * t + b$$

Equazione 4.2.2 Regressione lineare semplice

in cui a e b sono i coefficienti da stimare, attraverso i quali è possibile mettere in relazione la variabile da prevedere con il tempo. Esistono diverse forme funzionali più complesse, tra cui le più diffuse sono quella esponenziale e quella polinomiale. Tali metodi sono generalmente preferibili quando la serie storica mostra dei pattern globali, che possono essere catturati dai coefficienti del modello. Mentre non è possibile ottenere lo stesso risultato quando i pattern sono locali, infatti il modello di regressione necessita come input le informazioni su quando si verifica un cambiamento di pattern. Mentre con gli approcci *smoothing-based* tale problema non si pone poiché riescono ad estrapolare i cambi di trend e stagionalità sulla base delle osservazioni più recenti.

4.3 Metodo ARIMA

L'acronimo ARIMA sta per *Autoregressive Integrated Moving Average*, ed è uno dei modelli più diffusi per la previsione da serie storiche. La prima metodologia ARIMA è stata sviluppata da Box e Jenkins negli anni '70 [53] e fa parte della categoria dei metodi di regressione e si differenzia per l'inclusione dell'aspetto dell'autocorrelazione tra i dati della serie storica. L'autocorrelazione è una misura di quanto due variabili sono dipendenti tra di loro ed è molto alta nel caso vi sussista una relazione lineare. Nel caso di regressione multipla, la correlazione può essere calcolata tra tutte le variabili presenti così da costruire una matrice detta *funzione di autocorrelazione*.

Nei prossimi paragrafi saranno presentati i tre modelli che integra il metodo ARIMA: il modello autoregressivo, la differenziazione e il modello della media mobile sull'errore.

4.3.1 Modello autoregressivo

I modelli autoregressivi sono particolari modelli di regressione multipla applicati sulle serie temporali a cui è stata aggiunta una componente di *lag* a partire dalla serie originale. La finestra di *lag* è denominata con la lettera p e indica da quante osservazioni passate è composta la combinazione lineare della previsione [54]. Infatti, i regressori sono i valori della serie storica della stessa variabile di cui si vuole fare la previsione, e la regressione ha lo

scopo di stimare i coefficienti a da attribuire a ciascuno punto per trovare la combinazione lineare ottima. Tale regressione può essere rappresentata dall'equazione:

$$y_t = l + a_1 y_{t-1} + a_2 y_{t-2} + \dots + a_p y_{t-p} + e$$

Equazione 4.3.1 Modello autoregressivo

Dove l è il livello del dataset ed e è il rumore. Un modello autoregressivo con *lag* p può essere espresso brevemente con la formulazione $AR(p)$.

4.3.2 Differenziazione

La differenziazione nell'ambito delle serie storiche è una tecnica che permette di convertire una serie non-stazionaria in una stazionaria. Per definizione, una serie stazionaria è una serie di dati che non dipende dal tempo, come può esserlo una variabile casuale che non è correlata al fenomeno in esame. L'obiettivo di ottenere una serie stazionaria è quello di avere una stima dei valori dell'errore presente nella serie storica, che idealmente è una variabile casuale con media pari a zero, da utilizzare nella tecnica successiva [55].

La tecnica è molto semplice e prevede di ottenere una serie di valori dalla sottrazione di punti consecutivi, così da eliminare il fenomeno del trend che rende la serie non stazionaria. La differenziazione del primo ordine prevede la seguente operazione:

$$y'_t = y_t - y_{t-1}$$

Equazione 4.3.2 Relazione differenziazione del primo ordine

Successivamente si verifica se i valori ottenuti dalla differenziazione sono stazionari. Nel caso in cui non si soddisfa questa condizione si procede in un'ulteriore differenziazione dai valori ottenuti in precedenza. Il numero di differenziazioni effettuate è detto *grado di differenziazione* e viene comunemente indicato con la lettera d .

4.3.3 Media mobile sull'errore

Dalle previsioni effettuate con il modello autoregressivo si possono calcolare gli errori per i punti passati della serie storica, per poi essere inclusi come regressori nel modello finale. Infatti, il loro peso può essere stimato tramite una regressione dall'equazione:

$$y_t = I + e_t + \theta_1 e_{t-1} + \theta_2 e_{t-2} + \dots + \theta_q e_{t-q}$$

Equazione 4.3.3 Media mobile su errore di grado q

dove e_i è l'errore della previsione nel punto i , mentre per il periodo t di cui non si dispone ancora del valore realizzato si considera e_t come rumore bianco. Il risultato del modello sono le stime dei coefficienti θ_i da utilizzare come pesi nella media mobile. Il modello può essere brevemente indicato con la formulazione $MA(q)$ in cui q indica il numero di previsioni e i relativi errori da includere [56].

4.3.4 Modello generalizzato

Dalla combinazione dei tre modelli presentati si ottiene la formulazione del modello ARIMA:

$$y'_t = I + a_1 y'_{t-1} + a_2 y'_{t-2} + \dots + a_p y'_{t-p} + e_t + \theta_1 e_{t-1} + \theta_2 e_{t-2} + \dots + \theta_q e_{t-q}$$

Equazione 4.3.4 Modello ARIMA generalizzato

Come è possibile osservare dall'equazione, il modello è un autoregressione sui valori differenziati di grado d a cui si aggiungono gli errori come ulteriori regressori. Il metodo usato per stimare i coefficienti è il *Maximum Likelihood Estimation (MLE)*, che si caratterizza per massimizzare la probabilità che i dati ottenuti sono uguali ai dati effettivi [57].

La tecnica ARIMA è quindi un modello generale che può essere identificato con la terna di parametri (p, d, q) . Ad esempio, con ARIMA $(0, 1, 0)$ si ottiene il modello *Random walk*, mentre con ARIMA $(p, 0, 0)$ si ottiene un modello autoregressivo semplice di grado p [49].

4.4 Implementazione in R

Nonostante la complessità della forma del modello ARIMA, il quale richiede il calcolo di diverse equazioni iterative, con R è sufficiente l'esecuzione di pochi comandi per ottenere le previsioni grazie alle funzioni della libreria *forecast*.

La funzione più facile da utilizzare è *auto.arima()*, la quale usa l'algoritmo Hyndman-Khandakar [58] che è in grado di scegliere il modello ARIMA ottimale combinando il metodo *root tests*, la minimizzazione AICc⁷ e il metodo MLE. L'algoritmo prevede degli specifici step per scegliere i parametri p, d e q del modello ARIMA:

⁷ AIC è l'acronimo di Akaike Information Criterion, uno stimatore dell'errore sulle previsioni sul campione di test, usato quindi come una stima della qualità di un modello statistico. La

1. Il numero di differenziazione (d), generalmente compreso tra 0 e 2, è determinando mediante test KPSS, in modo iterativo.
2. I parametri p e q sono scelti minimizzando lo stimatore dell'errore AICc, in seguito alla differenziazione di grado d dal punto precedente. L'algoritmo prevede di procedere in modo graduale nello spazio del modello, evitando di provare ogni combinazione di p e q per ridurre il carico computazionale.
 - a. Alla prima iterazione l'algoritmo è inizializzato con quattro modelli standard;
 - b. Tra i quattro modelli si sceglie quello con AICc minore.
3. Successivamente si effettuano delle variazioni sul modello di partenza, in particolare:
 - a. si variano p e/o q di ± 1 ;
 - b. si include/esclude una costante c .
4. Si ripete il punto 3 finché non è più possibile ridurre lo stimatore AICc.

Se invece si vuole impostare il modello manualmente, è possibile usare la funzione *arima ()*. In tale funzione è possibile selezionare il metodo tramite l'argomento *order=c(p, d, q)*. Inoltre, in entrambi i metodi è possibile includere dei regressori con l'argomento *xreg*. Come mostrato nella figura 4.4.1, l'applicazione del metodo ARIMA può essere eseguita seguendo la procedura suggerita dal manuale [60]. Il primo passo prevede di rappresentare graficamente la serie storica così da valutare il tipo di andamento e la presenza di valori anomali. Successivamente, se è necessario, bisogna effettuare una trasformazione Box-Cox⁸ sui dati per stabilizzare la varianza. Poi si può scegliere se usare il metodo manuale oppure quello automatico basato sull'algoritmo descritto in precedenza. Se si sceglie quello manuale è necessario eseguire degli ulteriori test per valutare la bontà del modello. Infine, in entrambi i casi è prevista un'analisi dei residui, che è possibile effettuare con il *Ljung-Box test*, il quale ha per ipotesi nulla che gli errori sono indipendentemente distribuiti e come ipotesi alternativa che invece è presente una correlazione sistematica [62]. Infine, quando sono soddisfatti tutti i test è possibile calcolare le previsioni.

variante AICc considera anche la correlazione tra i dati, per evitare di selezionare modelli con troppi parametri i quali sono affetti da overfitting per campioni di piccole dimensioni. [59]

⁸ Trasformazione che permette di ottenere una distribuzione normale attraverso un elevamento a potenza [61]

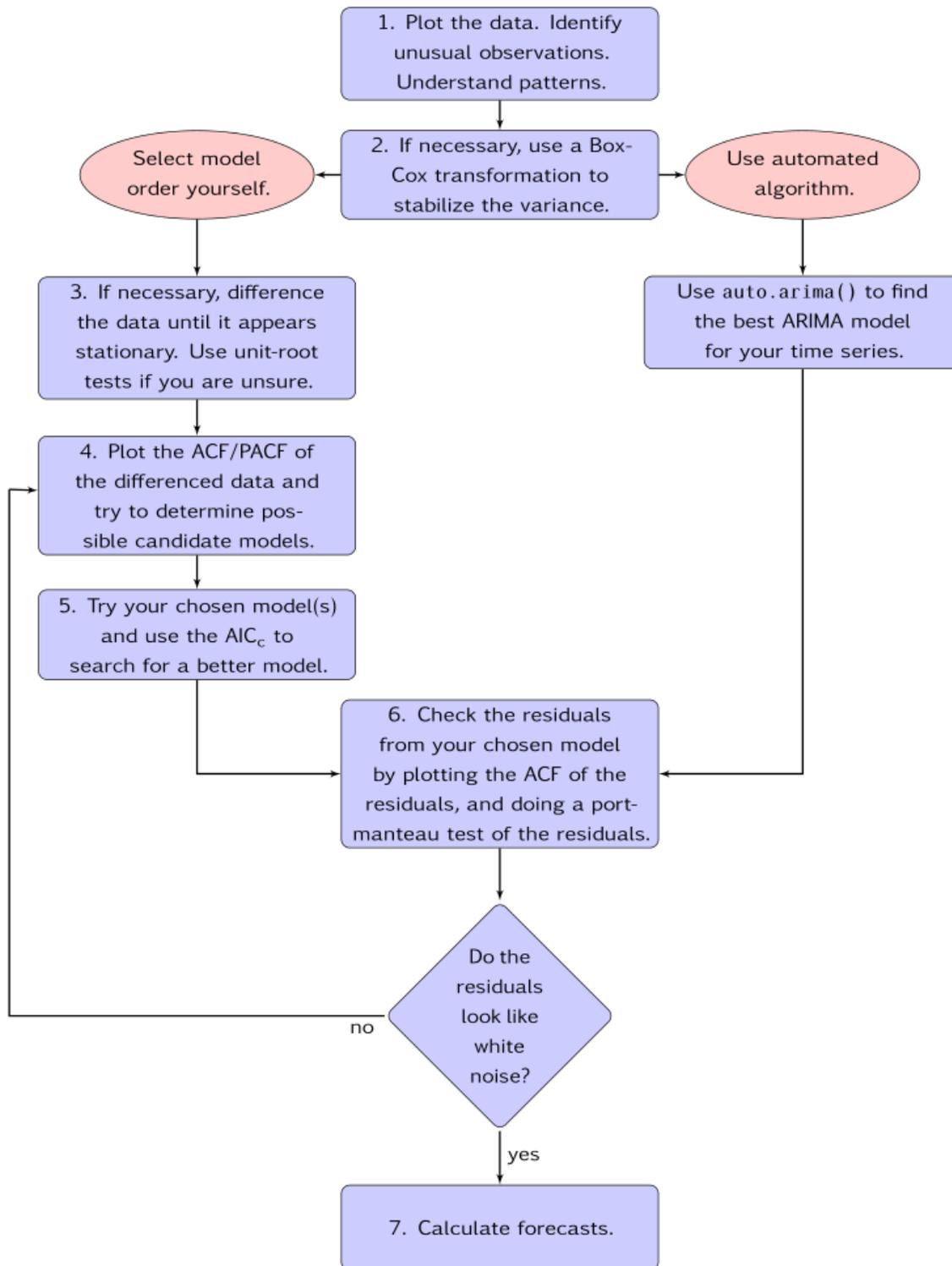


Figura 4.4.1 Procedura implementazione modello ARIMA [60]

5 CAPITOLO

MODELLI PHYSICS-BASED

5.1 Introduzione

Sebbene negli ultimi anni siano stati pubblicati studi prevalentemente sui metodi *data-driven*, in alcuni ambiti operativi, tra cui si cita il monitoraggio delle turbine offshore, la scelta di modelli *physics-based* (PbMs) può essere più vantaggiosa [63]. La caratteristica principale di tali metodi è l'abilità di relazionare matematicamente il fenomeno di usura con il tempo di utilizzo, parametro che può essere facilmente monitorato.

I PbMs descrivono i fenomeni che regolano un sistema, con lo scopo di identificare le condizioni operative in esso cui si trova. Essi sono in grado di diagnosticare le condizioni e prevedere la vita utile rimanente sulla base dei meccanismi di degradazione, che necessitano di essere identificati, che modellano. L'implementazione di modelli di degradazione richiede degli input diretti sulle condizioni di deterioramento del sistema. Tuttavia, gli input che raccolgono i sensori sono delle misure indirette delle variabili che si vuole monitorare. Quindi in ultima analisi, è necessario correlare gli input registrati dai sensori con le variabili controllate dai modelli di degradazione [71]. Seppure i PbMs non siano scalabili per sistemi diversi, poiché richiedono modelli specifici che spiegano il fenomeno in esame, essi sono particolarmente utili da integrare nei processi prognostici per avere una prima stima dello stato del sistema [72].

Il capitolo è strutturato in cinque paragrafi oltre a quello introduttivo: il primo descrive i principali modelli di degradazione (per frattura, per fatica e per usura) che è possibile monitorare in un processo produttivo; il secondo descrive una procedura generale da adottare per implementare un modello *physics-based* in un processo di manutenzione predittiva; il terzo approfondisce la relazione del modello di Taylor, già presentato nel capitolo 2, facendo un focus sul significato fisico dei coefficienti che compaiono; nel quarto è proposto il modello esteso della relazione di Taylor, che includendo ulteriori parametri raggiunge una migliore accuratezza nelle previsioni e sarà pertanto il modello di riferimento di questo studio; infine si propone un metodo di implementazione in R e le assunzioni teoriche da considerare.

5.2 Modelli di degradazione

Come è stato descritto in precedenza, i PbMs si basano sui modelli di degradazione che regolano un sistema. Pertanto, è necessario comprendere i tipi di modelli che regolano i sistemi da studiare per poterli applicare. Nell'ambito delle macchine rotanti esistono tre principali meccanismi di degradazione: per frattura, per fatica e per usura [72]. In seguito, saranno presentati brevemente facendo riferimento ai principali modelli presentati in letteratura.

5.2.1 Degradazione per frattura

La degradazione per frattura consiste in una deformazione permanente che può essere causata da bassi carichi prolungati nel tempo ad alte temperature. La causa fisica è la formazione di spazi vuoti tra i bordi di grano del materiale che si espandono. Tale fenomeno dipende dal tempo e dalla temperatura, come mostrano le seguenti equazioni:

$$\text{Creep rate} = \frac{\Delta \varepsilon}{\Delta t}$$

Equazione 5.2.1 Definizione Creep rate

dove la deformazione ε è messa in correlazione con il tempo t . Mentre la legge di Norton lega l'avanzamento della frattura (*creep rate*) con il livello di stress σ a cui è sottoposto il materiale e la temperatura T :

$$\text{Creep rate} = AT^n \sigma^m$$

Equazione 5.2.2 Legge di Norton sul Creep rate

dove A , n e m sono costanti empiriche relative al materiale. Dallo stato di avanzamento della frattura possono essere identificati tre fasi di deformazione:

- La prima è caratterizzata da un *creep rate* decrescenti, chiamata *frattura primaria*.
- La seconda presenta un *creep rate* costante, la quale dura più tempo ed è chiamata *frattura secondaria*.
- La terza è caratterizzata da un *creep rate* crescente, chiamata *frattura terziaria*, ed è la fase in cui la frattura si verifica a livello macroscopico.

Un ambito in cui il processo diagnostico è particolarmente incentrato su questo tipo di deterioramento è quello del monitoraggio delle turbine, in particolare le pale soggette ad alte temperature. In letteratura [105] è stato presentato un processo prognostico che stima il RUL della turbina, a partire dai dati di tempo e temperatura, stimando l'avanzamento della frattura.

5.2.2 Degradazione per fatica

La degradazione per fatica è un fenomeno che occorre quando il sistema è soggetto a sollecitazioni cicliche, le quali causano stress termici e meccanici. Tale fenomeno si presenta sotto forme differenti a seconda dell'intensità dei cicli. In particolare, nel caso di cicli ad alta intensità, il fenomeno di degradazione si presenta al di sotto della soglia di snervamento del materiale e il principale fattore che regola il meccanismo di degradazione è la variazione di tensione ($\Delta\sigma$) che si applica nei vari cicli. Mentre per i cicli a bassa intensità il fenomeno di degradazione per fatica si verifica al di sopra della soglia di snervamento e il fenomeno è regolato dalla variazione di deformazione ($\Delta\varepsilon$) a cui è sottoposto il materiale [71].

Tra i modelli riconosciuti in letteratura, è presente quello di Wohler [73] che dimostra a partire dai dati sperimentali dei binari ferroviari, i quali sono soggetti a fatica con elevata ciclicità, che la variazione di tensione è responsabile del fenomeno di degradazione. L'equazione che modella tale relazione è la seguente:

$$\log(N_f) = \log(C_0) - d * SD - m * \log(\sigma_a)$$

Equazione 5.2.3 Modello di Wohler

in cui il numero di cicli N_f dipende dall'ampiezza, della tensione σ_a e dalla deviazione standard consentita (SD) oltre a dei parametri (C_0 , d , m) che dipendono dalla geometria del materiale. Tramite il numero di cicli ottenuti dal modello è possibile determinare il RUL del materiale confrontandolo con il numero di cicli già effettuati. Tale stima è di particolare importanza, poiché la conseguenza del processo di degradazione per fatica è la rottura del materiale [71].

5.2.3 Degradazione per usura

L'ultimo meccanismo frequente di degradazione è quello per usura, il quale è trattato più volte nell'elaborato di tesi poiché è quello su cui si basano le analisi del dataset scelto. Il meccanismo di degradazione per usura avviene a causa della frizione tra due superfici, in modo diretto o mediante la presenza di un fluido, che genera perdita di materiale. Tale meccanismo è difficile da modellare accuratamente a causa dell'elevata influenza dei fattori esterni sul sistema [74]. Nei paragrafi 5.4 e 5.5 sarà trattato il modello di Taylor, il quale mette in relazione il livello di usura direttamente con i parametri di lavorazione della macchina. Mentre esistono dei modelli che relazionano il livello di usura con parametri fisici del materiale, che saranno presentati brevemente in seguito.

Il principale modello che tratta l'usura adesiva è la legge di Archard (equazione 5.2.4), la quale lega il livello di usura con i parametri fisici e meccanici del materiale: la durezza (H), il carico a cui è sottoposto (W) e la distanza di scorrimento (L), oltre a un coefficiente del materiale (K). Mentre ci sono modelli che stimano l'usura in modo indiretto tramite il calcolo della forza di frizione che si genera dal materiale. Il più basilico di tali modelli è la legge di Coulomb (equazione 5.2.5) [75] che stima la forza di frizione a partire dalla forza normale a cui è soggetto il materiale mediante un coefficiente di frizione. Le equazioni dei due modelli sono le seguenti:

$$V_{wear} = K \left(\frac{WL}{3H} \right)$$

Equazione 5.2.4 Legge di Archard

$$F_T = \mu F_N$$

Equazione 5.2.5 Legge di Coulomb sulla forza di frizione

5.3 Metodo di implementazione

In letteratura [64] è stato proposto uno schema generale con gli step necessari per implementare un metodo prognostico basato su un modello meccanico. Essi saranno presentati brevemente in seguito, facendo riferimento al caso di studio della ricerca di tesi:

1. Selezione del componente critico del sistema: il metodo è applicato al livello del singolo componente, pertanto è necessario selezionare quello maggiormente soggetto ai fenomeni di guasti o usura. Considerando il tempo richiesto per l'implementazione dell'intero metodo, questa fase è molto importante per focalizzare gli sforzi nella giusta direzione. Nel caso della fresatura, il componente critico è senza dubbio il tagliente dell'utensile, il quale è soggetto a forti stress meccanici durante il processo di lavorazione per asportazione di truciolo.
2. Determinazione dei meccanismi di usura e associazione di un modello fisico: una volta individuato il componente da monitorare è necessario comprendere le dinamiche che portano al deterioramento delle sue caratteristiche meccaniche. Nel caso di componenti complessi e unici tale fase può essere svolta con la tecnica *root cause analysis*, un processo sistematico per identificare le cause che generano un determinato fenomeno [65]. Nel caso degli utensili delle macchine rotanti, come la fresa, il meccanismo di usura da tenere maggiormente in considerazione è la *flank wear* (VB). In letteratura, come si cita nel capitolo di stato dell'arte, sono stati proposti diversi modelli matematici che mettono in relazione l'usura con il tempo di utilizzo. Nel paragrafo successivo sarà approfondita l'equazione di Taylor generalizzata.

3. Definizione delle condizioni di lavoro: per poter implementare correttamente un modello matematico basato sulle leggi fisiche è richiesta una definizione rigorosa dei parametri fisici e meccanici che possono influire sul processo di usura del componente analizzato. Alcune sono variabili di contorno, come la temperatura e la pressione ambientale, che possono essere trascurate se rientrano in range circoscritti. Di maggiore importanza sono i parametri di lavorazione della macchina o il carico di lavoro in un contesto più ampio. Infatti, la validità dei modelli matematici dipende dalle condizioni di applicazione, le quali devono essere monitorate costantemente. Nel caso della fresatura i parametri principali che influiscono sull'avanzamento dell'usura sono la velocità di taglio, il tipo di utensile (materiale e geometria) e il materiale di lavorazione. Di grande impatto sono anche altri parametri di lavorazione, in particolare la velocità di avanzamento e la profondità di taglio.
4. Raccolta dati su diverse condizioni di lavoro: dopo aver stabilito quali sono i parametri e le condizioni di lavorazione e i rispettivi intervalli in cui possono variare, è necessario raccogliere dei dati con misurazioni della variabile che quantifica l'usura per una vasta varietà di condizioni di lavoro. Se non sono disponibili dati di lavorazioni passate che soddisfano le esigenze del modello si possono definire degli esperimenti da effettuare, nel paragrafo 5.5.1 sarà presentata una tecnica per la scelta del campione. Per il caso della fresatura è necessario avere un dataset con almeno una coppia di osservazioni di tutte le combinazioni dei parametri di lavorazione elencati nel punto precedente.
5. Stima dei coefficienti empirici: grazie ai dati raccolti è possibile stimare i coefficienti empirici presenti nella relazione matematica. Il metodo da usare in questa fase dipende dal tipo di relazione, spesso di tipo non lineare. Nel caso della relazione di Taylor per gli utensili rotanti può essere utilizzata una regressione lineare con il metodo dei minimi quadrati.
6. Valutazione del metodo: con i coefficienti stimati il modello è in grado di effettuare le previsioni richieste, stimando i livelli di usura per determinate condizioni di lavoro e di tempo di utilizzo. Le prime previsioni da effettuare sono sui dati raccolti non usati per stimare i coefficienti, così da confrontare le previsioni con i valori effettivi che si sono realizzati. Tale step è cruciale per determinare la correttezza e l'accuratezza del metodo, che tuttavia dipende dal numero di dati storici disponibili. Nel caso in esame verranno usati indicatori statistici consolidati nell'ambito delle previsioni, in particolare l'RMSE, mentre la scelta dei dati da usare per effettuare le valutazioni può essere effettuata mediante il metodo di *cross validation*.

5.4 Interpretazione fisica dell'equazione di Taylor

L'equazione di Taylor è la relazione più diffusa nell'ambito dei processi di lavorazioni meccaniche con macchine rotanti e utensili soggetti ad usura. Dall'equazione basica (2.2.1) è evidente la relazione inversa che lega la vita utile dell'utensile con la velocità di taglio. Tale relazione è stata dimostrata anche empiricamente [70] come mostra la figura 5.4.1.

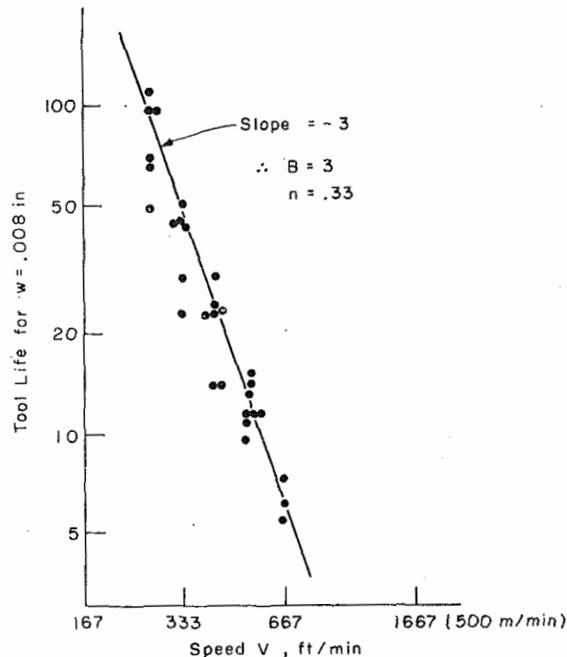


Figura 5.4.1 Relazione empirica tra T e V [70]

Inoltre, è possibile dare un significato fisico anche ai due coefficienti che compaiono nell'equazione. Per spiegare il significato della costante C basta porre $T=1$ ottenendo:

$$V(1)^n = C \rightarrow V = C$$

Equazione 5.4.1 Relazione della costante della legge di Taylor

il che mostra che C corrisponde alla velocità di taglio con la quale si otterrebbe un minuto di vita utile. Mentre per esplicitare il significato di n è necessario differenziare l'equazione 2.2.1 fino a raggiungere la seguente relazione:

$$\frac{d}{dn} \left(-\frac{dT}{dV} \right) < 0$$

Equazione 5.4.2 Relazione n dell'equazione di Taylor

che mette n in relazione con il termine $(-dT/dV)$, che può essere definito come la sensitività alla velocità di taglio della vita utile dell'utensile. Quindi n è un indicatore di quanto la vita

utile dell'utensile risente della variazione della velocità di taglio [66]. Da dati empirici è stato definito un range di tale esponente per i materiali dell'utensile più comune:

- $n \in [0,1;0,15]$ per utensili in acciaio HS;
- $n \in [0,2;0,25]$ per utensili con carburi;
- $n \in [0,6;1.0]$ per utensili in ceramica.

5.5 Modello di Taylor esteso

Per le analisi presentate nel capitolo 9 è stata utilizzata l'equazione di Taylor estesa per gli utensili rotanti, la quale include tutti i parametri di lavorazione presenti nel dataset. L'equazione presenta la seguente forma:

$$T = K * v^R f^S a^W VB_{max}^Z$$

Equazione 5.5.1 Modello di Taylor esteso per utensili da taglio

Dove:

- T è la vita utile dell'utensile calcolata in t_0 , espressa in [min];
- K, R, S, W, e Z sono costanti empiriche da stimare. Verranno chiamate anche con il termine "coefficienti", in quanto rappresentano i coefficienti nel modello di regressione multipla;
- v è la velocità di taglio espressa in [mm/min];
- f è la velocità di avanzamento, denominata *feed* nel dataset, espressa in [mm/rev];
- a è la profondità di taglio, denominata *DOC* nel dataset, espressa in [mm];
- VB_{max} è la larghezza massima di *flank wear* che può essere misurata in relazione al tempo di attività T, espressa in [mm].

Tale equazione può essere facilmente invertita per stimare il livello di VB atteso dopo un periodo di utilizzo t, come mostra la seguente equazione:

$$VB(t) = e^{\frac{\ln(t) - \ln K - R \ln v - S \ln f - W \ln a}{Z}}$$

Equazione 5.5.2 Relazione inversa di Taylor per stima VB

5.5.1 Stima dei coefficienti

La stima dei coefficienti dell'equazione 5.5.1 può essere effettuata con una regressione lineare multipla effettuando una trasformazione logaritmica, come mostra la seguente equazione:

$$\ln(T) = \ln(K) + R\ln(v) + S\ln(f) + W\ln(a) + Z\ln(VB_{max})$$

Equazione 5.5.3 Trasformazione logaritmica dell'equazione di Taylor estesa

Che può essere scritta in forma matriciale:

$$Y = Xb$$

$$\begin{bmatrix} \ln T_1 \\ \ln T_2 \\ \ln T_3 \\ \vdots \\ \ln T_N \end{bmatrix} = \begin{bmatrix} 1 & \ln v_1 & \ln f_1 & \ln a_1 & \ln VB_{max_1} \\ 1 & \ln v_2 & \ln f_2 & \ln a_2 & \ln VB_{max_2} \\ 1 & \ln v_3 & \ln f_3 & \ln a_3 & \ln VB_{max_3} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & \ln v_N & \ln f_N & \ln a_N & \ln VB_{max_N} \end{bmatrix} \begin{bmatrix} K \\ R \\ S \\ W \\ Z \end{bmatrix}$$

Dove X è la matrice di sensitività [67] e b contiene i coefficienti che possono essere stimati con il metodo dei minimi quadrati, che può essere ottenuto con le seguenti operazioni matriciali:

$$b = (X^T X)^{-1} X^T Y = V X^T Y$$

Equazione 5.5.4 Equazione matriciale per la stima dei coefficienti con OLS

Dall'equazione 5.5.4 è possibile notare che la precisione con cui i coefficienti sono stimati dipende dall'inverso della matrice $X^T X$, termine su cui si basano i criteri di ottimizzazione. Quello proposto da Duarte e Arruda [68] prevede la minimizzazione del rapporto (NC) del massimo valore singolare (MaxVSX) con il minimo valore singolare (MinVSX) della matrice X:

$$NC = \frac{MaxVSX}{MinVSX}$$

Equazione 5.5.5 Rapporto di Duarte e Arruda

Secondo tale criterio, il campione di dati da raccogliere per il training del metodo deve soddisfare la condizione di min (NC) per ottenere la stima ottimale dei coefficienti.

5.6 Implementazione in R

Come è stato descritto nei paragrafi precedenti, i metodi *physics-based* si basano su delle equazioni matematiche già ben definite, in cui i dati empirici hanno solo il ruolo di stimare i coefficienti che legano le relazioni. Se la relazione è di tipo lineare, oppure linearizzabile, la stima dei coefficienti con la regressione lineare multipla può essere effettuata tramite la

funzione $lm()$ già presente nell'ambiente R. Per verificare la linearità dei dati è possibile prima verificare il tipo di andamento rappresentando i punti graficamente.

Prima di applicare i coefficienti stimati nel modello è importante che siano verificate le seguenti assunzioni [69]:

- I residui sono normalmente distribuiti;
- Tra i regressori non vi è multicollinearità: può essere testata mediante il test VIF^9 (*Variance Inflation Factor*);
- Presenza di omoschedasticità: va verificato che gli errori non presentino una correlazione sistematica con le variabili indipendenti, come ad esempio con il tempo.

Infine, quando sono state verificate le assunzioni e la regressione raggiunge un livello statisticamente significativo, le previsioni possono essere effettuate integrando parametricamente i coefficienti stimati nel modello matematico. Nel caso in cui alcuni coefficienti non raggiungono la soglia di significatività stabilita, è possibile effettuare la regressione modificando il campione di dati usato per stimare i coefficienti.

⁹ Il VIF è il rapporto tra la varianza di un modello con regressori multipli con la varianza di un modello con singolo regressore. È usato per quantificare il grado di multicollinearità in una regressione effettuata con il metodo OLS, restituendo un indice che misura quanto la varianza di una regressione aumenta a causa della presenza di collinearità tra i regressori. Per ogni regressore i incluso nel modello si può calcolare l'indice con la seguente relazione:

$$VIF_i = \frac{1}{1-R_i^2}.$$

6 CAPITOLO

STRUMENTI UTILIZZATI

Il seguente capitolo riporta gli strumenti utilizzati per effettuare le analisi sui modelli presentati in seguito, fornendo una breve descrizione delle loro caratteristiche. Nel primo paragrafo è descritto il linguaggio di programmazione (R) utilizzato per implementare i modelli ed effettuare delle analisi statistiche su di essi. Mentre nel secondo, sono descritte le librerie utilizzate a cui fanno riferimento le funzioni utilizzate negli script mostrati nel capitolo 9.

6.1 R

Le analisi dei dati e le applicazioni dei modelli sono state effettuate tramite l'ambiente R. R è un linguaggio di programmazione che è incluso nell'omonimo software open source supportato dalla *R Foundation for Statistical Computing*. Il linguaggio è largamente utilizzato per lo sviluppo di software statistici e di analisi di dati. Esso si basa sui linguaggi C e Fortran e in parte dallo stesso R, infatti è parzialmente *self-hosting* [79].

R tramite le numerose librerie messe a disposizione dalla fondazione consente di effettuare una grande varietà di analisi statistiche. Infatti, rende possibile implementare facilmente modelli lineari e non lineari, analisi delle serie storiche, clustering e altri modelli. Inoltre, R consente di rappresentare i dati con strumenti grafici avanzati che soddisfano le esigenze dei dati statistici [80].

6.2 Librerie

I modelli utilizzati per le analisi presenti in questa tesi non hanno richiesto lo sviluppo poiché sono già stati sviluppati in apposite librerie rese disponibili dalla *R Foundation*. Di seguito sono riportate le librerie utilizzate:

- *R.matlab*: consente di leggere dati con estensione *.mat* derivanti dal linguaggio di programmazione Matlab [81].
- *Readxl*: consente di leggere e scrivere file con estensioni *.csv* e *.xlms* in modo da interfacciarsi con Excel [82].

- Forecast: scritta da Rob J. Hyndman, contiene i metodi e gli strumenti per visualizzare e analizzare le serie storiche, includendo i modelli di *exponential smoothing* e ARIMA. Da tale libreria sono state utilizzate le funzioni [83]:
 - Arima();
 - Auto.arima();
 - Forecast().
- Ggplot2: sistema per la creazione dichiarativa di grafica, basato su *The Grammar of Graphics*. Contiene tutti i tipi di grafici utili per la descrizione statistica dei risultati ottenuti [84].
- Dplyr: strumento utile per la manipolazione dei dati, consentendo di eseguire tutte le operazioni basi tipiche del linguaggio SQL [85].
- Neuralnet: consente di eseguire la formazione di reti neurali utilizzando la *backpropagation* o la *backpropagation resiliente* con o senza peso. Le funzioni sono flessibili poiché consentono di personalizzare l'errore e la funzione di attivazione [86].
- Nnet: pacchetto che abilita algoritmi di reti neurali *feed-forward* con un solo *layer* nascosto e modelli log-lineari multinomiali [87].
- Rstanarm: stima i modelli di regressione compilati utilizzando il pacchetto *rstan*, che fornisce l'interfaccia R alla libreria *Stan C ++* per la stima bayesiana [88].
- Metrics: implementazione delle metriche di valutazione in R comunemente utilizzate nel training supervisionato. Implementa metriche per problemi di regressione, serie temporali, classificazione binaria, classificazione e recupero delle informazioni [89].
- Caret: fornisce diverse funzioni utili per il training dei modelli di regressione e strumenti per la rappresentazione grafica dei risultati [90].
- GridExtra: fornisce una serie di funzioni *user-level* per lavorare con la grafica a "griglia", in particolare per disporre più grafici basati su griglia su una pagina e disegnare tabelle [91].
- Plyr: insieme di strumenti che risolve una serie comune di problemi, in particolare consente di suddividere una struttura complessa di dati e lavorare sulle singole componenti in modo omogeneo [92].

- RandomForest: implementa l'algoritmo *random forest* proposto da Breiman (basato sul codice Fortran originale di Breiman e Cutler) per la classificazione e la regressione dei dati [113].
- Gbm: pacchetto sviluppato originariamente da Greg Ridgeway, il quale contiene un'implementazione delle estensioni all'algoritmo *AdaBoost* di Freund e Schapire che includono il *gradient boosting machine* di Friedman [114].
- Bolstad: insieme di funzioni R e data set che consentono di implementare le statistiche bayesiane [115].

7 CAPITOLO

CASO DI STUDIO

Il seguente capitolo descrive un metodo con il quale processare i dati ottenuti dai sensori e l'origine dei dati utilizzati in questa ricerca. Nel primo paragrafo sono riportate le fasi principali del framework proposto da un gruppo di ricercatori del Politecnico di Torino [100], al quale si fa riferimento in tutte le analisi successive della ricerca di tesi. Mentre nel secondo paragrafo è descritto il dataset utilizzato, facendo riferimento alla descrizione fornita dagli autori che hanno condotto l'esperimento [98].

7.1 Framework

Lo schema generale del framework proposto da [100] è rappresentato graficamente nella figura 7.1.1. Essa mostra come tra il processo di acquisizione dei dati e quello di monitoraggio c'è uno stadio intermedio di pre-processamento dei dati. Tale stadio è quello che determina la qualità delle analisi successive, pertanto verrà descritto con maggiore dettaglio in questo paragrafo.

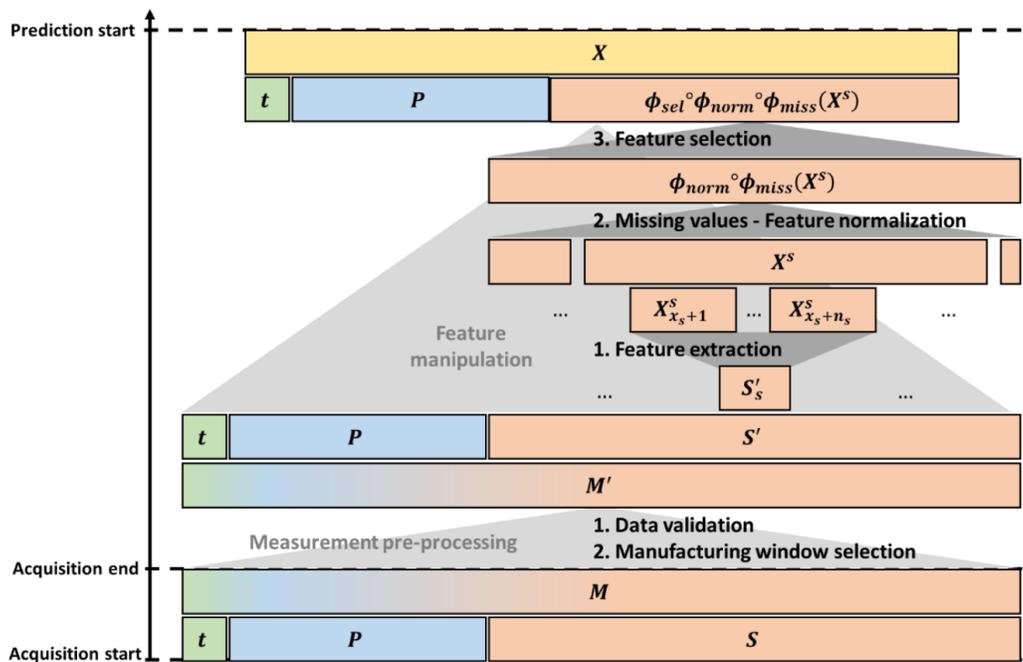


Figura 7.1.1 Schema del framework utilizzato [100]

La prima fase di pre-processamento dei dati, la *data cleaning*, prevede la rimozione dei valori che non soddisfano determinati requisiti e che si presentano incoerenti con il dataset in generale [101]. Il primo criterio utilizzato per il dataset *milling* è stato quello di rimuovere i record incompleti, cioè che presentavano valori mancanti in alcuni campi. Il secondo criterio prevede di eliminare i record con valori che non rientrano nel dominio previsto per un determinato parametro. Sono stati eliminati in totale 22 record ottenendo al termine della fase di *cleaning* un dataset composto da 146 osservazioni.

Segue l'analisi di rimozione degli outlier, che rispetto alla fase precedente segue dei criteri più specifici per individuare i dati da eliminare. Gli outlier sono per definizione dei valori che si discostano fortemente dagli altri punti del campione e tali scostamenti non sono spiegati dai normali processi del sistema [102]. Nel framework proposto è considerata la distinzione tra la ricerca degli outlier all'interno di una serie temporale e tra serie temporali, in modo da riconoscere segnali estremi che indicano un'instabilità del processo. La loro rimozione dal campione è pertanto necessaria per evitare un effetto di distorsione nelle successive analisi. Un primo metodo per cui si può applicare per l'individuazione di outlier è il Z-value test, metodo basato sull'assunzione che i dati seguano una distribuzione normale [103]. Sotto tale condizione, il 99,73% dei valori standardizzati è distribuito nel range (-3,3), e possono essere considerati outlier i valori che si presentano al di fuori di tale intervallo. Nel caso in esame è stato utilizzato il metodo basato sul range interquartile (*IQR*) [104]. Da questa analisi sono stati identificati 33 outlier su un totale di 1020 di valori registrati dai sensori.

Per le serie temporali, in particolare per i valori registrati dai sensori, è necessaria una fase di *stationary window selection*, la quale permette di selezionare la fase stazionaria della macchina tramite la tecnica di *Change Point Detection (CPD)* [105]. Infatti, per le analisi di previsione sono richiesti i dati registrati nel periodo che intercorre tra l'avvio e la fine del processo di lavorazione. I criteri di selezione della serie stazionaria dipendono dal tipo di segnale: nel caso di segnali asimmetrici non-negativi è il valore massimo della media, mentre per i segnali simmetrici è il massimo valore della deviazione standard. Nel caso in esame il *change point* è stato fissato con il parametro $\alpha = 0,1\%$, usando il test statistico Bartlett per il sensore della corrente alternata e il test *Generalized Likelihood Ratio* per gli altri sensori.

Mentre le prime fasi hanno l'obiettivo di eliminare le righe con valori mancanti o distorti, quelle che seguono hanno lo scopo di definire le colonne contenenti le variabili che portano più informazione nel descrivere il fenomeno da analizzare. La prima tra queste è la *feature extraction*, che a partire da un insieme complesso di variabili

multidimensionali, combina e seleziona le variabili maggiormente correlate alle variabili di interesse. Tale operazione consente di ridurre il carico computazionale richiesto degli algoritmi da implementare in seguito e migliora la velocità di apprendimento dei processi di *machine learning*. Il primo metodo utilizzato è di tipo *time domain statistics* il quale valuta la magnitudine del segnale attraverso un set di statistiche, tra cui media e deviazione standard, che descrivono la fase stazionaria. Segue il metodo di tipo *frequency domain statistics*, il quale descrive la distribuzione dello spettro di potenza della fase stazionaria. Il dominio di frequenza può essere ottenuto tramite la trasformazione discreta di Fourier, mentre le statistiche sono simili all'estrazione precedente. Infine, viene applicata la tecnica *time polynomial regression coefficients*, in cui si effettuano regressioni polinomiali su tutta la serie temporale. A valle delle tre tecniche di estrazione sono state estratte 162 features¹⁰.

Dalle features ottenute nella fase precedente è previsto un processo di normalizzazione dei dati. Infatti, per implementare con efficacia gli algoritmi di machine learning è necessario che i valori si presentino su scale omogenee. Tale fase è anche detta *feature scaling*, e nel caso in esame è stata effettuata con una normalizzazione Min-Max ottenendo così valori compresi tra 0 e 1.

L'ultima fase è la *feature selection*, che ha lo scopo di ridurre il numero ancora elevato di attributi derivato dalla fase di *feature extraction*. Quindi si selezionano solo le feature che sono sufficientemente correlate con le variabili di risposta e si rimuovono le feature che sono altamente correlate tra di loro, scegliendone solo una di esse per ridurre la ridondanza. In questo caso la variabile di risposta è VB e le correlazioni sono state quantificate e ordinate tramite il coefficiente di varianza, la cui soglia è stata fissata a 0,25. Tuttavia, la selezione avviene prevalentemente mediante metodi non supervisionati. I principali metodi si basano sui coefficienti di variazione e correlazione tra le feature, che selezionano solo quelle con valori maggiori. Nel caso in esame sono stati usati i metodi che calcolano i parametri di monotonicità¹¹ e prevedibilità, con un peso di 0,8 per il primo e di 0,2 per il secondo a causa delle dimensioni ridotte del dataset, fissando la soglia del test d'ipotesi con $\alpha = 0,1$. A valle di tutti i metodi di selezione sono state rimosse 105

¹⁰ Per ognuno dei sei sensori sono state estratte 8 features di dominio temporale (massimo, minimo, media, deviazione standard, errore quadratico medio, simmetria, curtosi e fattore di cresta), 7 del dominio di frequenza replicate sul modulo e sull'argomento (massimo, minimo, media, deviazione standard, simmetria, curtosi e il picco dello spettro relativo) e 5 coefficienti polinomiali di regressione.

¹¹ Calcolata per ogni case con la formula $m = \frac{1}{N-1} |n_{>} - n_{<}|$, dove N è il numero totale di osservazioni, $n_{>}$ è il numero di variazioni positive e $n_{<}$ è quello di variazioni negative.

features dalle 162 estratte in precedenza, ottenendo così 57 features da utilizzare nelle analisi successive.

7.2 Dataset

Le analisi effettuate nella ricerca di tesi sono basate sul dataset pubblico *Milling* (Agogino & Goebel) messo a disposizione dal Prognostic Center of Excellence (NASA-PCoE) [98]. Esso contiene i valori registrati da sei sensori durante tutto il ciclo vita di 16 tools (*case*), sotto diverse condizioni di lavoro identificate con 8 scenari, per un totale di 170 lavorazioni (*run*). Ogni *case* è caratterizzato da tre parametri di lavorazione, che seguono le raccomandazioni del produttore dell'utensile, e dal tipo di materiale lavorato di dimensioni fisse (483 mm x 178 mm x 51 mm). Eccetto la velocità di taglio, che resta invariata a 200 m/min, le altre variabili sono dicotomiche e in particolare: la velocità di avanzamento (*feed*) è stata fissata a 0,25 mm/s o a 0,5 mm/s; la profondità di taglio (DOC) è stata fissata a 0,75 mm o a 1,5 mm; mentre i materiali del pezzo di lavorazione sono ghisa (*cast iron*) o acciaio inossidabile 145 (*stainless steel*). La tabella 7.2.1 riassume l'insieme degli esperimenti effettuati.

Tabella 7.2.1 Descrizione Case del dataset [98]

Case	#Runs	Material	Feed	DOC
1	17	Cast iron	0.5	1.5
2	14	Cast iron	0.5	0.75
3	16	Cast iron	0.25	0.75
4	7	Cast iron	0.25	1.5
5	6	Stainless steel	0.5	1.5
6	1	Stainless steel	0.25	1.5
7	8	Stainless steel	0.25	0.75
8	6	Stainless steel	0.5	0.75
9	9	Cast iron	0.5	1.5
10	10	Cast iron	0.25	1.5
11	23	Cast iron	0.25	0.75
12	15	Cast iron	0.5	0.75
13	15	Stainless steel	0.25	0.75
14	10	Stainless steel	0.5	0.75
15	7	Stainless steel	0.25	1.5
16	6	Stainless steel	0.5	1.5

Per effettuare le lavorazioni è stata impiegata la *Matsuura MC-510V*, una macchina a controllo numero verticale con una fresa per spianare a sei inserti da 70 mm. Sono stati utilizzati inserti KC710, caratterizzati da diversi rivestimenti in sequenza di carburo di titanio, carbonitruro di titanio e nitrato di titanio, che conferiscono una buona resistenza al *cratering* e all'usura degli spigoli così che il fenomeno di usura principale è la *flank wear*.

Mentre i sensori hanno raccolto dati in modo continuo, sia durante l'attività che durante il fermo macchina, le misurazioni della *flank wear* (VB) sono state effettuate in modo periodico. Infatti, per ottenere tale misurazione è stato necessario rimuovere l'inserito dalla fresa e misurare con un apposito microscopio la distanza dal tagliente all'estremità dell'usura abrasiva sulla faccia laterale dell'utensile. Data la diversità dei parametri di lavorazione, sono stati ottenuti cicli con tempo di attività e numero di lavorazioni differenti per ogni utensile. Inoltre, data la discontinuità delle misurazioni, non è stata rispettata una soglia massima fissa di *flank wear* ottenendo dei valori massimi piuttosto eterogenei.

I sensori hanno raccolto dati definiti su una serie temporale di 9000 unità per ogni lavorazione. Le misurazioni sono state effettuate sulla corrente alternata (smcAC) e continua (smsDC) del motore del mandrino e sulle vibrazioni (vib_table e vib_spindle) ed emissioni acustiche (AE_table e AE_spindle) sul piano di lavoro e sul mandrino rispettivamente. Tutti i parametri appena descritti, presenti nel dataset, sono riportati in modo riassuntivo nella tabella 7.2.2.

Tabella 7.2.2 Descrizione campi dataset [98]

NOME CAMPO	DESCRIZIONE
CASE	Numero del case (1-16)
RUN	Contatore delle lavorazioni effettuate nel case
VB	Flank wear, misurata dopo i run (non tutti)
TIME	Durata totale della lavorazione (all'interno del case)
DOC	Profondità di taglio (costante all'interno del case)
FEED	Velocità di taglio (costante all'interno del case)
MATERIAL	Tipo di materiale lavorato (costante all'interno del case)
SMCAC	Corrente AC motore mandrino
SMCDC	Corrente DC motore mandrino
VIB_TABLE	Vibrazioni sul piano di lavoro
VIB_SPINDLE	Vibrazioni sul mandrino
AE_TABLE	Emissioni acustiche sul piano di lavoro
AE_SPINDLE	Emissioni acustiche sul mandrino

Il dataset si può quindi riassumere con l'insieme di *monitoring factors* $M(i, j)$ per il tool i -esimo alla lavorazione (run) j -esima, così definiti:

$$M(i, j) := P \cup S \cup T \cup O$$

Dove gli insiemi P, S, T e O rappresentano:

- P: parametri di produzione;
- S: serie storiche delle misurazioni dei sensori;
- T: istante temporale in cui si registra il run j-esimo;
- O: altri fattori relativi alla misurazione.

I dati raccolti dai sensori sono stati inviati al sistema di processamento attraverso una scheda di acquisizione dati ad alta velocità con frequenza di campionamento massima di 100 KHz. Per il campionamento dei dati in uscita è stata utilizzato il software LabVIEW¹² per l'elaborazione del segnale generato dai sensori. Inoltre, i segnali dei sensori, eccetto quelli della corrente (sia alternata che continua) del motore del mandrino, sono stati sottoposti a un processo di pre-elaborazione come mostrato schematicamente nella figura 7.2.1.

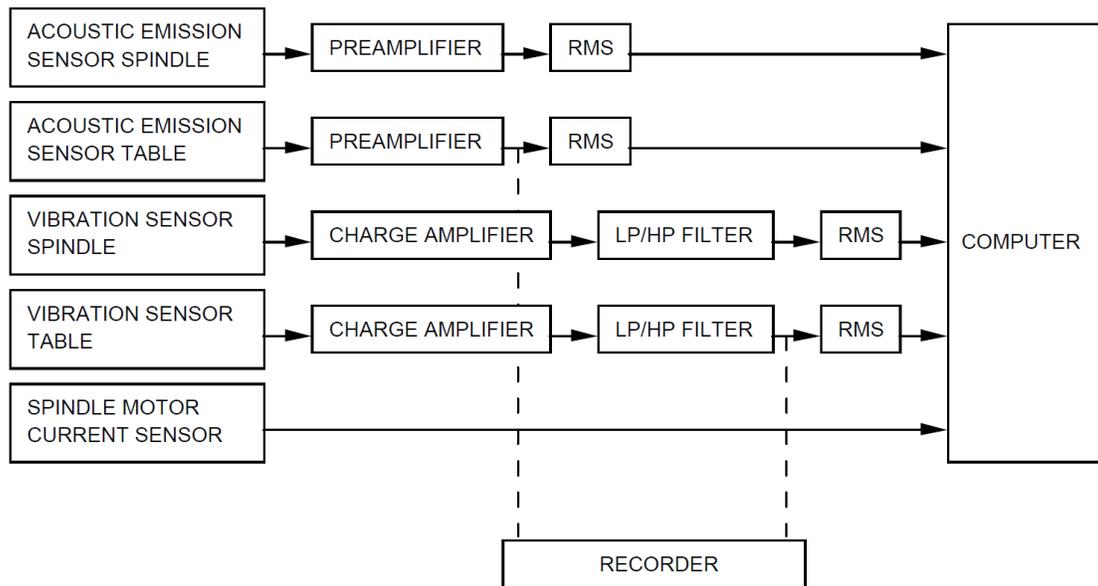


Figura 7.2.1 Setup della raccolta dati [98]

Sia per i sensori di emissioni acustiche che per quelli di vibrazione, il segnale è stato amplificato per soddisfare i requisiti di soglia minima richiesto dalle apparecchiature di processamento dei segnali. Tali valori amplificati sono stati poi codificati tramite la trasformazione RMS come riportato nell'equazione 7.2.1. Tale passaggio consente di attenuare il segnale dalle oscillazioni misurate, così da renderlo più accessibile

¹² LabVIEW è un software di programmazione grafica che consente di gestire la configurazione hardware, i dati di misura e il debug. [99]

all'elaborazione del processo. L'RMS è proporzionale al contenuto energetico del segnale, secondo la formula:

$$RMS = \sqrt{\frac{1}{\Delta T} \int_0^{\Delta T} f^2(t) dt}$$

Equazione 7.2.1 RMS (Root Mean Square)

Dove ΔT è un intervallo di tempo costante (fissato a 8,00 ms) e $f(t)$ è la funzione del segnale proveniente dai sensori a una data frequenza (250 Hz nel caso in esame).

Inoltre, i dati dei sensori di emissione acustica e di vibrazione del piano di lavoro sono stati registrati anche senza l'elaborazione RMS. Tale scelta è stata fatta per consentire di effettuare delle comparazioni direttamente sui *raw data*. Infatti, può essere vantaggioso effettuare la fase di *feature extraction* sui valori forniti direttamente dai sensori.

8 CAPITOLO

METODI DI VALUTAZIONE DEI MODELLI PREDITTIVI

Il seguente capitolo ha lo scopo di definire le basi teoriche su cui si basano le valutazioni dei modelli implementati nelle analisi successive. Nel primo paragrafo è descritto il principio adottato per effettuare la fase di validazione del modello. Infatti, soprattutto quando il dataset da utilizzare contiene pochi dati, è fondamentale utilizzare un criterio per suddividere i dati a disposizione per effettuare sia l'addestramento che la validazione del modello di previsione. Un criterio di suddivisione dei dati, con il metodo di cross validation, è presentato nel secondo paragrafo. Nel terzo paragrafo invece, sono presentate le metriche utilizzate per valutare le performance dei modelli, riportando brevemente le principali proprietà di essi.

8.1 Validazione del modello di previsione

Come rappresentato nella figura 8.1.1, per validare un modello è necessario suddividere il dataset in due sottoinsiemi: il primo per addestrare il modello (*training set*) e il secondo per testarlo (*validation test* o *test set*). Tale suddivisione è necessaria per verificare le performance del modello senza influenzarlo con i dati che effettivamente sono stati già osservati. Dal confronto delle previsioni del modello addestrato con il *training set* (linea blu

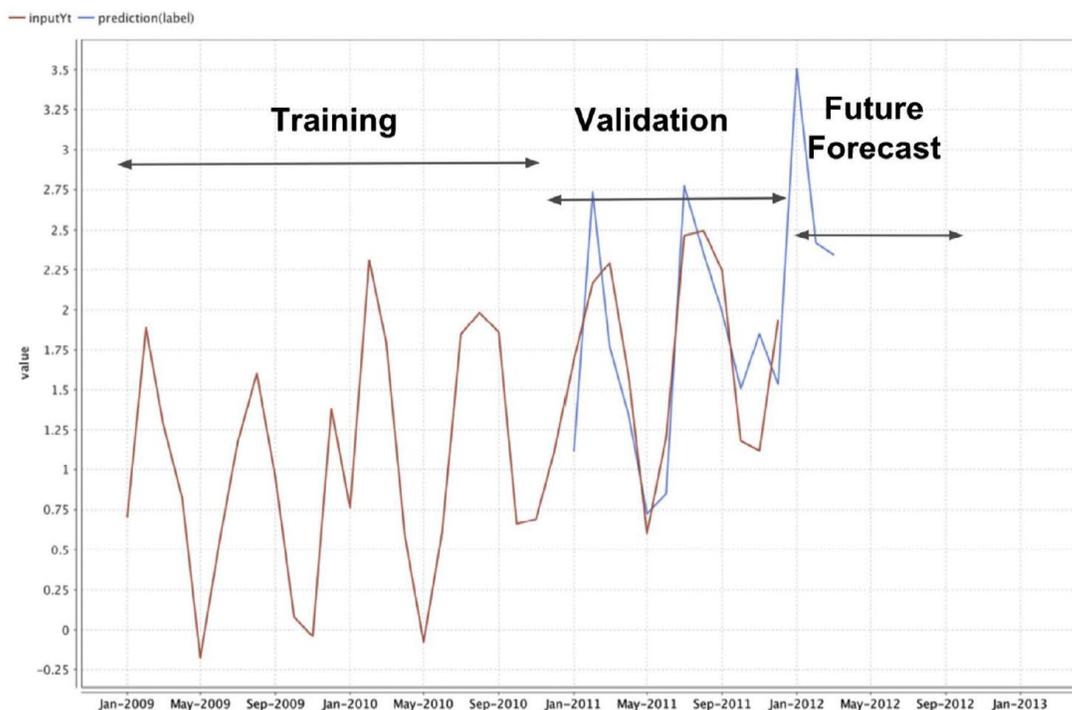


Figura 8.1.1 Scomposizione dataset per testare modelli di previsione

nella figura) e con i valori reali (linea rossa) si può valutare l'accuratezza del modello di previsione.

Con il modello validato si possono poi effettuare le previsioni future, dove l'efficacia può essere misurata solo a posteriori [50]. La fase di validazione è necessaria poiché con l'analisi dei residui, che è fatta sugli stessi dati usati per l'addestramento del modello, si rischia di valutare un modello affetto da *overfitting* in cui le previsioni sui dati non osservati possono presentare un'accuratezza molto inferiore rispetto a quella osservata con l'analisi dei residui.

8.2 Cross validation

Il metodo base di validazione dei dati prevede di scomporre il dataset in due partizioni ed eseguire, come descritto nel paragrafo precedente, con una l'addestramento e con l'altra la validazione. Tuttavia, nel caso di campioni con pochi dati a disposizione, una scelta arbitraria delle partizioni può influenzare fortemente le valutazioni. Da questa considerazione nasce l'approccio di *cross validation*, il quale prevede di partizionare il dataset più volte in modo iterativo. Come mostrato nella figura 8.2.1, un metodo che utilizza questo approccio prevede di suddividere il dataset in k partizioni, utilizzando $k-1$ di esse per l'addestramento e la rimanente per la validazione. Questo processo viene replicato k volte, così da avere k misure di performance con lo stesso dataset. Al termine delle iterazioni si possono aggregare le misure ottenute con un operatore statistico, il più utilizzato è la media, per avere una misura più robusta delle prestazioni del modello di previsione [109]. Infatti, quest'ultima non dipende più dalla scelta dei dati utilizzati per l'addestramento del metodo grazie alle iterazioni che consentono di alternare diverse volte i dati da destinare alle due partizioni.



Figura 8.2.1 Rappresentazione metodo di cross validation [110]

Il valore di k dipende, con una relazione inversa, dalle dimensioni del campione. Infatti, per campioni piccoli k dovrebbe essere grande per aumentare il numero di iterazioni e ridurre il bias sull'errore calcolato. Convenzionalmente si utilizza $k = 5$ o $k = 10$, sulla base dei risultati empirici di alcuni studi in letteratura [111].

8.3 Metriche di valutazione

Al fine di valutare le performance dei diversi metodi implementati si utilizzano le metriche ampiamente consolidate in letteratura, come riportato nell'articolo [107]. La prima metrica che può essere adottata è l'errore di previsione (equazione 8.3.1), che rappresenta lo scarto tra la previsione effettuata e il valore effettivo per ogni punto. Una limitazione di tale metrica è la dipendenza dalla scala dei valori, non consentendo di paragonare previsioni su valori di grandezze differenti.

$$Error(e_i) = Y_{i\text{ effettivo}} - Y_{i\text{ calcolato}}$$

Equazione 8.3.1 Errore di previsione

Quelle che seguono sono metriche aggregate, le quali consentono di valutare le prestazioni del modello su tutto l'orizzonte del campione di validazione. La prima metrica di tale famiglia è l'errore medio (equazione 8.3.2), che verrà raramente utilizzato a causa delle sue limitazioni. Infatti, mediando gli errori con i rispettivi segni si verifica una compensazione tra essi, portando a sottostimare l'errore del modello. Inoltre, tale misura non considera l'ordine di grandezza del fenomeno che si sta monitorando, rendendo difficile paragonare il modello in diversi scenari.

$$Mean\ Error\ (ME) = \frac{1}{N} \sum_{i=1}^N (Y_{i\text{ effettivo}} - Y_{i\text{ calcolato}})$$

Equazione 8.3.2 ME: Errore medio di previsione

Per superare le limitazioni dell'errore medio spesso si usa l'errore medio assoluto in forma percentuale (equazione 8.3.3). Con la forma assoluta dell'errore si evita il fenomeno di compensazione, mentre dividendo per i valori effettivi si ottiene una misura relativa. Grazie alle sue proprietà il MAPE può essere usato per confrontare diversi modelli in diversi scenari.

$$Mean\ Absolute\ Percentage\ Error\ (MAPE) = \frac{1}{N} \frac{|\sum_{i=1}^N Y_{i\text{ effettivo}} - Y_{i\text{ calcolato}}|}{\sum_{i=1}^N Y_{i\text{ effettivo}}}$$

Equazione 8.3.3 MAPE: Errore assoluto medio percentuale

Le successive metriche sono quelle più ampiamente utilizzate per valutare le performance dei modelli di previsione e saranno pertanto sempre calcolate nelle analisi effettuate in questa ricerca di tesi. Come è stato descritto, già il MAPE presenta delle buone proprietà di comparazione di diversi modelli. Tuttavia, a parità di MAPE si possono presentare dei modelli che effettuano stime con buona accuratezza in tutti i punti e modelli che presentano ottima accuratezza in alcuni punti e scarsa in altri. Per superare questa criticità si usano gli errori quadratici, che hanno la proprietà di penalizzare fortemente gli errori di maggiore entità e di penalizzare meno quelli più piccoli. Su questo principio si basa l'ampiamente conosciuto RMSE che presenta la seguente equazione:

$$\text{Root Mean Squared Error (RMSE)} = \sqrt{\frac{1}{N} \sum_{i=1}^N (Y_{i \text{ effettivo}} - Y_{i \text{ calcolato}})^2}$$

Equazione 8.3.4 RMSE: Radice dell'errore quadratico medio

Come l'errore medio di previsione anche l'RMSE è dipendente dalla scala. Per questo motivo si affianca anche una misura relativa: l'RRSE (equazione 8.2.5). Quest'ultima metrica oltre a permettere di valutare diversi metodi da già un'indicazione relativa della bontà del modello. Infatti, il denominatore della formula può essere considerato un modello *baseline* dove le previsioni future sono pari alla media dei valori osservati. Quindi si può concludere che un modello apporta informazione se l'RRSE è strettamente minore di 1, così come per tutte le altre metriche presentate il verso di preferenza è verso il basso [108].

$$\text{Root Relative Squared Error (RRSE)} = \sqrt{\frac{\sum_{i=1}^N (Y_{i \text{ effettivo}} - Y_{i \text{ calcolato}})^2}{\sum_{i=1}^N (Y_{i \text{ effettivo}} - \bar{Y})^2}}$$

Equazione 8.3.5 RRSE: Radice dell'errore quadratico relativo

9 CAPITOLO

IMPLEMENTAZIONE MODELLI DI PREVISIONE DELL'USURA

Il seguente capitolo rappresenta la parte centrale della ricerca di tesi. Infatti, contiene i risultati dell'implementazione, utilizzando il dataset *milling*, dei metodi descritti nei capitoli precedenti. Esso è diviso in otto paragrafi che seguono l'evoluzione dello sviluppo dei modelli. Nel primo paragrafo si descrive la fase preliminare, di *data cleaning* in aggiunta al framework descritto nel capitolo 7, che è stata effettuata per migliorare la qualità del dataset. Nel secondo paragrafo è analizzata, mediante una statistica descrittiva, la distribuzione della variabile da monitorare per comprendere meglio il fenomeno da modellizzare. Nel terzo paragrafo è descritta la metodologia applicata per implementare l'approccio di *cross validation*, già descritto dal punto di vista teorico nel paragrafo 8.2, rispettando le caratteristiche del caso di studio *milling*. Nei paragrafi 4, 5 e 6 sono analizzati i risultati ottenuti dei metodi di *tool condition monitoring* presentati nei capitoli precedenti: modelli ARIMA (metodo delle serie storiche), modello di Taylor (metodo *physics-based*) e infine algoritmi di machine learning (metodi *data-driven*). Per ognuno dei metodi analizzati sono state sviluppate delle varianti dei modelli per analizzare le caratteristiche specifiche della categoria. Dai metodi che hanno presentato le prestazioni migliori, nel paragrafo 7, sono stati analizzati gli errori di previsione per singolo run (numero di lavorazioni effettuate dall'utensile) per analizzare la distribuzione dell'errore all'avanzare del fenomeno di usura. Infine, dalle considerazioni dell'analisi degli errori, nel paragrafo 8 è presentato un modello ibrido tra il metodo *physics-based* (modello di Taylor) e un metodo *data-driven* (algoritmo basato sulle reti neurali) con l'obiettivo di presentare un approccio più robusto per modellizzare un fenomeno di usura caratterizzato da andamenti lineari e non lineari.

9.1 Data cleaning

Il dataframe di partenza, "mill_df_for_models.rda" come si può vedere nella prima riga dello script 1 del paragrafo 9.3, è l'output della fase di *feature selection*. Tale fase non verrà approfondita ulteriormente da come è stata presentata nel paragrafo 7.1 in quanto non è l'oggetto principale della ricerca di tesi. Tale dataframe contiene tutti i dati che rispettano le condizioni di coerenza richieste per i valori dei sensori, con la rimozione degli outlier e dei dati mancanti.

Tuttavia, sono presenti ancora delle anomalie per i valori di VB. In particolare, sono stati rimossi i seguenti *case*:

- *Case 1*: le misurazioni del livello di usura non rispettano la condizione di monotonia crescente. Esso appartiene allo scenario {DOC=1,5, feed=0,5, materiale=*cast iron*}, la cui replicazione è il *case 9*. Le due curve di livello di usura, mostrate nella figura

9.1.1, dovrebbero mostrare un andamento simile, mentre è evidente che le misurazioni degli ultimi punti del case 1 si discostano molto dall'andamento previsto.

- *Case 6*: i valori di VB sono mancanti, non consentendo di implementare nessun modello di previsione.

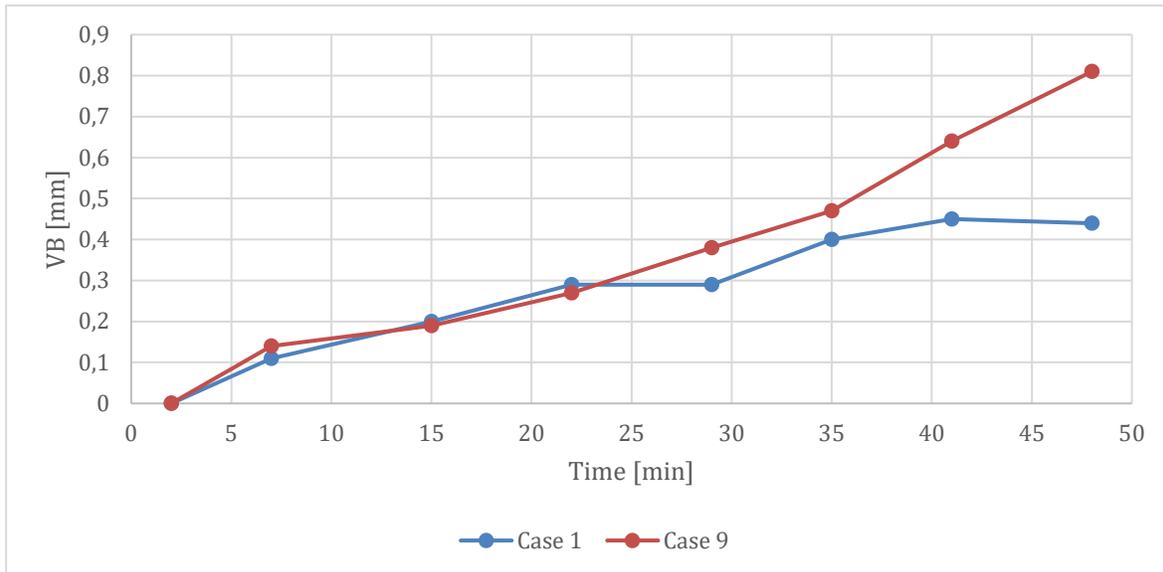
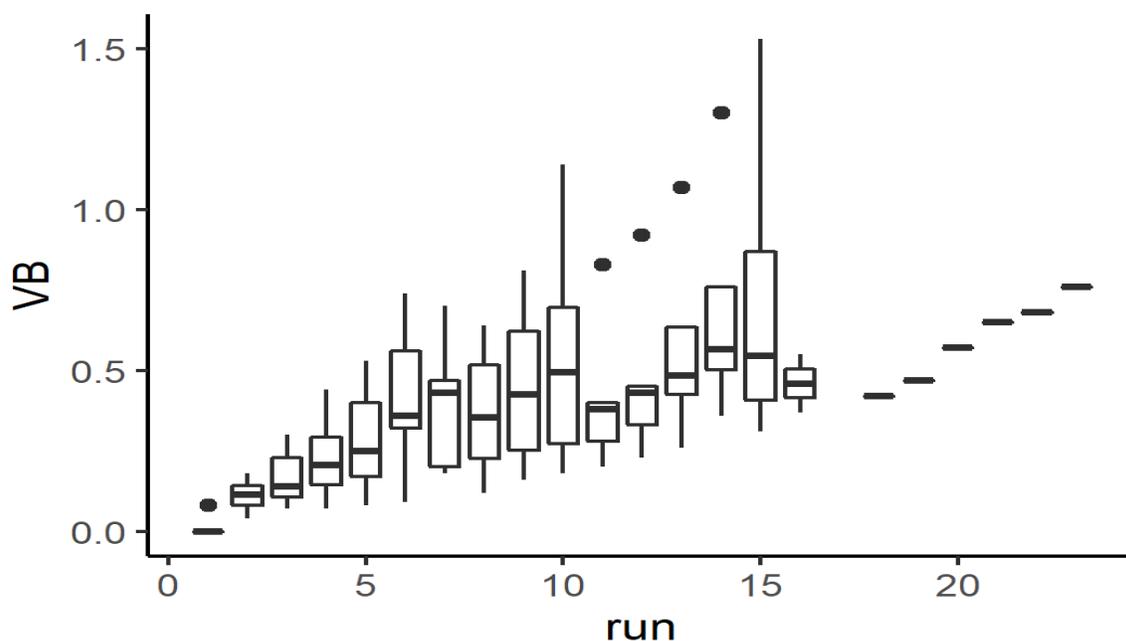


Figura 9.1.1 Andamento di VB (da misurazioni) per scenario DOC=1,5, feed=0,5, materiale=cast iron.

9.2 Analisi andamento della variabile di controllo (VB)

Prima di analizzare i modelli implementati è interessante osservare l'andamento della variabile di controllo del processo di usura all'avanzare dei run. I box-plot rappresentati nella figura 9.2.1 sono stati calcolati sulla base delle misure di VB dei diversi *tools* per ogni determinato numero di run effettuati. L'obiettivo di tale analisi è quello di dimostrare che la variabilità del livello di usura aumenta all'avanzare dei run. Infatti, nelle prime lavorazioni esso segue un andamento costante e pressoché simile per tutti gli utensili, mentre quando l'usura raggiunge uno stato avanzato la sua propagazione assume un carattere maggiormente aleatorio. Dalla figura citata tale fenomeno lo si può dedurre dall'aumento dell'ampiezza dei box-plot all'aumentare dei run. Tuttavia, tale trend è possibile osservarlo solo in parte a causa delle limitate osservazioni presenti nel dataset in esame. Infatti, il numero di *tools* osservati per ogni livello di run effettuati è variabile e rende il confronto tra i vari run in parte approssimativo. Tale limitazione è maggiormente evidente per i numeri di run che presentano

una sola osservazione, i cui box-plot sono rappresentati con delle linee, in cui non è possibile nemmeno approssimare la variabilità presente. Mentre è necessario porre maggiore attenzione per i run nella parte intermedia (da 11 a 13), dove la variabilità diminuisce a causa del ridotto numero di osservazioni (di circa 5 contro una media di 10 dei run precedenti) e quindi la contrazione della loro ampiezza non è significativa. Si può concludere che l'ipotesi di partenza, riguardante l'aumentare della variabilità di VB all'avanzare dei run, è stata in parte confermata e sarà di supporto per le analisi successive.



Run (j)	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	18	19	20	21	22	23
# tools	9	12	14	10	13	13	9	8	8	6	5	5	4	4	4	2	1	1	1	1	1	1

Figura 9.2.1 Andamento di VB all'avanzare dei run con le misure del dataset

9.3 Approccio di cross validation

Come presentato nel paragrafo 8.2, l'approccio del metodo di *cross validation* consente di effettuare delle misure di performance più robuste generando diverse partizioni di *training set* e *validation set*. Nel caso in esame tale metodo è stato applicato al dataset *milling*. Tale dataset necessita di alcune precisazioni. La prima riguarda la presenza di 8 scenari ¹³ replicati

¹³ Dati dalle combinazioni dei parametri binari: materiale, velocità di avanzamento e profondità di taglio.

due volte (2 *case*¹⁴ sono stati esclusi, ottenendo un totale di 14 *case*), quindi è necessario che nel *training set* sia presente almeno un case di ogni scenario per addestrare i modelli a prevedere tutti i possibili scenari futuri. La seconda riguarda la diversità del fenomeno di usura per i due materiali di lavorazione, che determinano una velocità di usura dell'utensile nettamente differente. Ciò rende necessario inserire un numero uguale di case per ogni materiale per addestrare i modelli con la stessa efficacia per le previsioni dei due diversi andamenti di usura.

Nella figura 9.3.1 è rappresentato graficamente il partizionamento dei dati effettuato con il dataset *mill*. Tutti i *case* (tools) presenti nel dataset costituiscono l'insieme *supervised historical data* (I_{SUP}) il quale è partizionato negli insiemi di *training set* (I_{TR}) e *test set* (I_{TEST}) tramite estrazioni casuali. Nella figura sono estratti cinque *tools* per ogni materiale, nelle analisi seguenti sono stati testati i modelli anche con quattro e sei estrazioni per materiale per valutare la sensibilità dei modelli alla dimensione dei dati di addestramento.

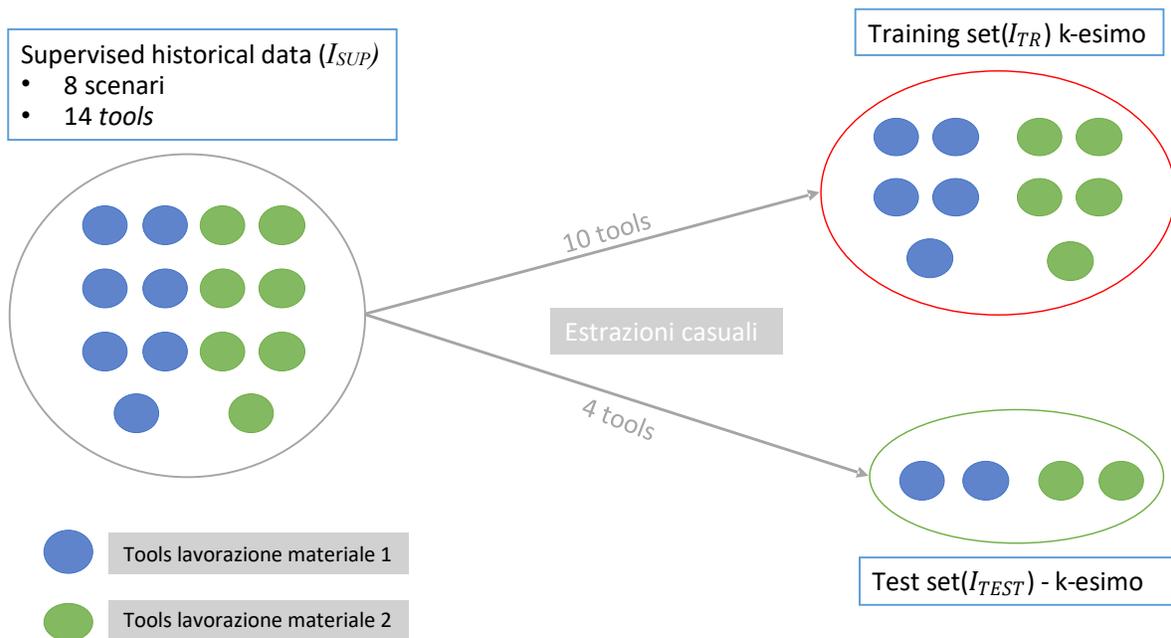


Figura 9.3.1 Rappresentazione approccio di cross validation utilizzato

Nello script 1 è presente l'algoritmo implementato per selezionare i *case* da destinare nelle due partizioni: di addestramento e validazione. Il numero di iterazioni è stato fissato a 10 per le ragioni illustrate nel paragrafo 8.2. L'algoritmo consente di scegliere il numero di *case* per ogni materiale da impiegare per l'addestramento, con la variabile denominata "n_case_train_for_m" che può assumere i valori {4;5;6}, in modo da testare i modelli con

¹⁴ Ogni case si riferisce ad un utensile, in seguito sarà usato il termine *tools* con lo stesso significato.

diverse condizioni. Nello script è riportato solo la sezione per la selezione dei case del primo materiale, del tutto analoga per il secondo, che segue tre step:

1. Inizializzazione del vettore che andrà a contenere i *tools* della partizione di addestramento con un tool del materiale in esame estratto casualmente;
2. Estrazione casuale di altri tre *tools* con la condizione che appartengono a scenari che non sono già presenti nella partizione di addestramento, in modo da selezionare un *case* per ognuno dei quattro scenari;
3. Se il numero di tools richiesto per l'addestramento è superiore a 4 si estraggono in modo casuale altri *tools* fino a raggiungere il numero impostato.

Al termine della selezione dei *tools* da destinare all'addestramento per ogni materiale, essi sono aggregati in un vettore unico. Mentre i tools da utilizzare per la validazione sono i complementari di quest'ultimo.

Script 1 Algoritmo di cross validation

```
mill <- readRDS(file=" mill_df_for_models.rda")
#esclusione Case 1 e 6 (anomalie VB)
mill<-dplyr::filter(mill,case!=1 & case!=6)
case<-unique(dplyr::select(mill,case,DOC,feed,material1))
case_m1<-dplyr::filter(case, material1==1)

#scelta del numero di case per materiale per il train set (da 4 a 6)
n_case_train_for_m<-5

for (i in 1:10) {
  #inizializzazione variabili con i Case per il training set di ogni materiale
  train_m1<-rbind(train_m1,dplyr::filter(mill,case==sample(case_m1$case,1)))
  train_m2<-rbind(train_m2,dplyr::filter(mill,case==sample(case_m2$case,1)))

  #scelta successivi 3 case in modo casuale: si ottengono 4 case per materiale (tutti gli 8 scenari)
  #con la condizione di non avere case con coppia (DOC,feed) uguale nel train set
  for (j in 2:4) {
    repeat{
      c<-sample(case_m1$case,1)
      parametri<-select(dplyr::filter(case_m1,case==c), DOC,feed)
      if(nrow(match_df(select(train_m1,DOC,feed),parametri))==0){
        break}
      }
    train_m1<-rbind(train_m1,dplyr::filter(mill,case==c))
  }
}
```

```

#aggiunge i case successivi al 4 al train set
if (n_case_train_for_m-4>0) {for (j in 1:(n_case_train_for_m-4)){
  repeat{
    c<-sample(case_m1$case,1)
    if(!c%in%train_m1$case){
      break}}
  train_m1<-rbind(train_m1,dplyr::filter(mill,case==c))}

#costruzione training set dall'unione dei training set dei due materiali
#test set con i case complementari
train<-rbind(train_m1,train_m2)
test <- dplyr::filter(mill, ! case %in% unique(train$case))

```

9.4 Modelli ARIMA

I primi modelli implementati sono quelli di serie storiche che si basano solo sui valori di VB nel tempo e sui parametri di lavorazione, non considerando i valori dei sensori. Data la presenza di diversi scenari sono stati utilizzati modelli ARIMA multivariata, pertanto il primo step è stato quello della definizione dei regressori. In questa fase è stato definito l'attributo "change" per fornire al modello l'informazione che la serie storica è composta da *case* differenti tra loro. L'attributo *change* è quindi così definito:

$$Change = \begin{cases} 1 & \text{per il primo run di ogni case} \\ 0 & \text{altrimenti} \end{cases}$$

Tale attributo è stato aggiunto nel modello con una colonna al dataframe *mill*, il quale si rivelerà significativo nei modelli di previsione, come riportato nello script 2.

Script 2 Definizione attributo *change*

```

change<-vector(length = nrow(mill))
change[1]<-1
for(i in 2:nrow(mill)){
  if(mill$case[i]==mill$case[i-1]){
    change[i]=0
  }else{
    change[i]=1
  }
}
mill<-cbind(mill,change)

```

L'insieme complessivo dei regressori considerati è: *DOC, feed, time, run, material1*¹⁵ e *change*. Inoltre, sono stati testati diversi tipi di modelli ARIMA, compreso l'algoritmo per la scelta automatica presentato nel paragrafo 4.4 con la funzione *auto.arima()*.

I tipi di modelli ARIMA testati manualmente sono (1,0,0), (1,1,0), (2,0,0), (2,1,0), il cui modello generico (p,d,q) è riportato nello script. mentre per l'algoritmo automatico sono stati seguiti due approcci. Il primo, denominato "v1", è addestrato con tutti i case indistintamente dal materiale. Mentre il secondo, denominato "v2," anziché riconoscere il materiale con l'apposito regressore, esegue due modelli separati per i materiali.

Script 3 Modello ARIMA (p,d,q)

```
#ARIMA (p,d,q) -----
xtrain<-dplyr::select(train, time, feed, DOC,material1, change, run)
xtest<-dplyr::select(test, time, feed, DOC,material1, change, run)
model<-Arima(train$VB,xreg=as.matrix(xtrain), order = c(p, d, q))
fcst<- forecast(model, xreg=as.matrix(xtest))
est_VB<-fcst[["mean"]]
```

Script 4 Modelli ARIMA automatici

```
#Auto ARIMA v1 -----
xtrain<-dplyr::select(train, time, feed, DOC,material1, change, run)
xtest<-dplyr::select(test, time, feed, DOC,material1, change, run)
model<-auto.arima(train$VB,xreg=as.matrix(xtrain))
fcst<- forecast(model, xreg=as.matrix(xtest))
est_VB<-fcst[["mean"]]

#Auto ARIMA v2 (Modelli separati per materiale)-----
xtrain_m1<-dplyr::select(train_m1, time, feed, DOC, change, run)
xtrain_m2<-dplyr::select(train_m2, time, feed, DOC, change, run)
xtest<-dplyr::select(test, time, feed, DOC, change, run)
model_m1<-auto.arima(train_m1$VB,xreg=as.matrix(xtrain_m1))
model_m2<-auto.arima(train_m2$VB,xreg=as.matrix(xtrain_m2))

est_VB2<-vector(length = nrow(test))
for (i in 1:nrow(test)) {
  if(test$material1[i]){
    est_VB2[i]<-as.vector(forecast(model_m1, xreg=as.matrix(xtest[i,]))[["mean"]])
  }
}
```

¹⁵ Attributo che definisce il tipo di materiale: è pari a 1 per *cast iron* e pari a 0 per *steel*.

```

}else{
  est_VB2[i]<-as.vector(forecast(model_m2, xreg=as.matrix(xtest[i,]))[["mean"]])
}
}
}

```

Le analisi sono state effettuate con il metodo di *cross validation* presentato nel capitolo precedente utilizzando 5 *case* per materiale scelti in modo casuale nella partizione di addestramento, le cui prestazioni sono state misurate con le metriche RMSE e RRSE. Includendo tutti i regressori il modello che ha registrato le migliori performance, come mostrato nella figura 9.4.1, è quello denominato “auto ARIMA v2”, che dimostra come è più vantaggioso separare i tipi di materiali di lavorazione da analizzare. Questo deriva dalla differente propagazione dell’usura dell’utensile per i diversi materiali lavorati, che seppure sia caratterizzata da un pattern simile, è difficile da modellare con lo stesso metodo di serie storiche. Infatti, con la funzione *auto.arima()* è emerso che il modello ottimale per il primo materiale è dell’ordine (2,0,0) mentre per il secondo è dell’ordine (2,1,0).

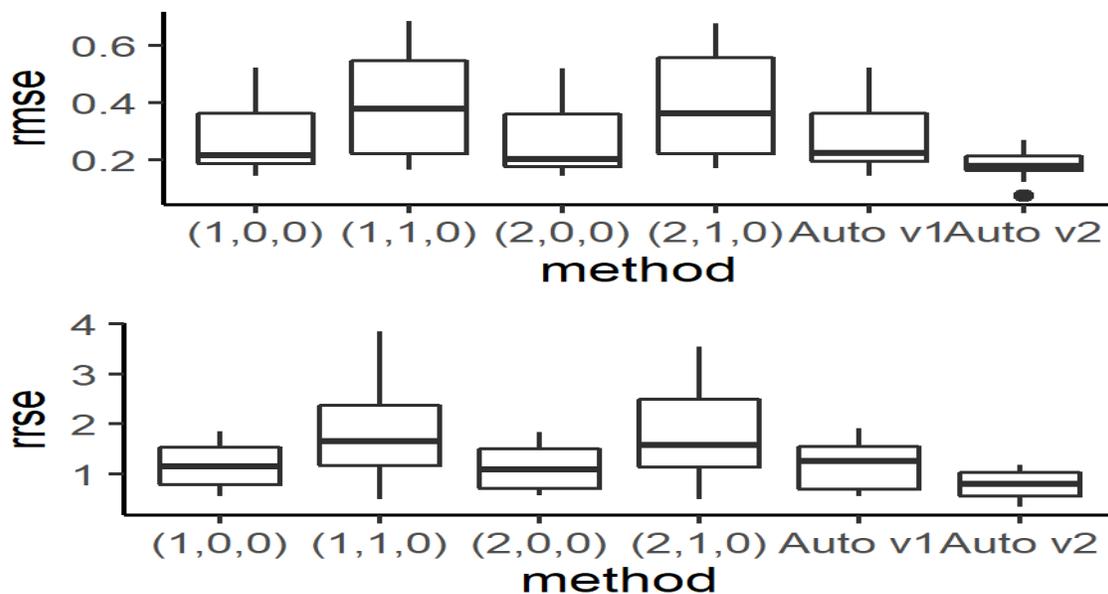


Figura 9.4.1 Performance modelli ARIMA

Prendendo come riferimento il modello “auto ARIMA v2” sono stati implementati ulteriori modelli per validare l’insieme di regressori scelto. In particolare, sono state effettuate le seguenti analisi:

- Verifica della significatività dell’attributo aggiungo *change*: omettendo l’attributo dai regressori le performance di previsione peggiorano, pertanto è significativo.

- Verifica dei regressori temporali: nei modelli soprariportati sono stati inclusi tra i regressori sia il tempo che il numero di run eseguiti, entrambi riconducibili a parametri di tipo temporale. È stato verificato se fossero ridondanti implementando due modelli con un singolo regressore temporale. È emerso che con un singolo regressore le performance peggiorano in entrambi i casi. Tale risultato può essere spiegato dalla considerazione che tra il tempo e il numero di run non vi è una relazione lineare, assumendo che ogni run abbia una durata differente.

I risultati di tali analisi sono riportati sinteticamente nella seguente tabella:

Tabella 9.4.1 Performance ARIMA con diversi regressori

MODELLO	RMSE	RRSE
TUTTI I REGRESSORI	0,180297	0,790363
TUTTI ESCLUSO CHANGE	0,268386	1,103367
TUTTI ESCLUSO RUN	0,191758	0,860839
TUTTI ESCLUSO TIME	0,186589	0,869894

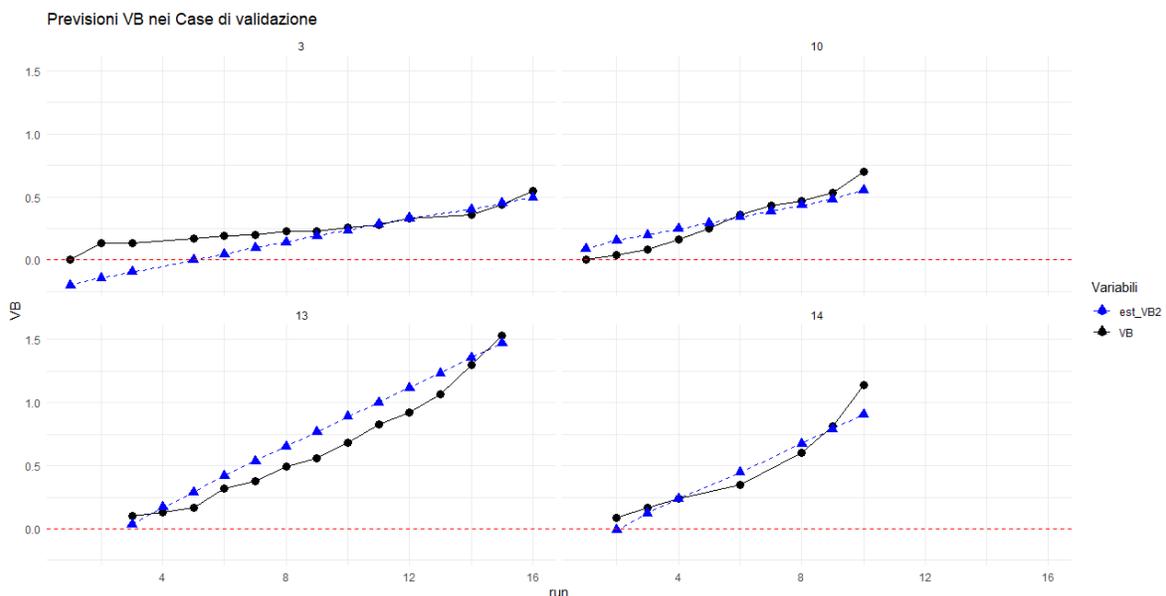


Figura 9.4.2 Confronto previsioni VB modello ARIMA con valori effettivi

Infine, con il modello che ha presentato le prestazioni migliori sono state fatte le previsioni sui *case* della partizione di validazione. Come mostrato nella figura 9.4.2, le previsioni di VB degli ultimi run sono più accurate per i *case* con un numero maggiore di run (*case* 3 e 13). Il numero maggiore di run è dato dalla propagazione più lenta dell'usura che si verifica a basse velocità di avanzamento e profondità di taglio, determinando un andamento più lineare della curva. Infatti, i modelli ARIMA, come descritto nel capitolo 4, sono adatti a modellare serie storiche con andamento lineare. Mentre la propagazione dell'usura,

soprattutto nella fase finale, segue un andamento esponenziale come è evidente nel *case* 14. Quest'ultimo *case* mostra i limiti dei metodi di previsione basati su modelli lineari, in cui le previsioni tendono a sottostimare i livelli di usura nelle ultime fasi del ciclo vita dell'utensile (*case* 10 e 14 della figura sopraccitata).

9.4.1 Modello ARIMA dinamico

Nel paragrafo precedente è stato dimostrato che i metodi di serie storiche possono essere applicati per prevedere l'andamento di usura di un utensile di lavorazione. L'indicatore RRSE minore di uno (circa 0,8 nel modello migliore) mostra che tali modelli sono in grado di spiegare il fenomeno, seppure solo in parte. Tuttavia, le potenzialità dei metodi di serie storiche si raggiungono quando si forniscono delle informazioni aggiornate sul fenomeno da monitorare, così da generare delle previsioni sulla base dei trend già osservati.

Un modello ARIMA che aggiorna le previsioni di usura su un determinato *case*, che simula l'utensile corrente che si sta utilizzando su una macchina, può essere analizzato a partire da un dataset di due *case*. Il primo si usa interamente per l'addestramento del modello, mentre il secondo viene usato sia per l'addestramento che per la validazione. In particolare, per un *case* di n run si possono effettuare n set di previsioni aggiornate. Definito m come il numero di run osservati, ogni set di previsioni si effettua con il modello addestrato con le m nuove osservazioni del secondo *case* (in aggiunta a quelle del primo), stimando gli $(n-m)$ run rimanenti. Tale algoritmo è stato applicato alla coppia di *case* 2 e 12, appartenenti allo scenario con il materiale *cast iron*, $DOC = 0,75$ e $feed = 0,5$. Il *case* 2 è stato utilizzato per l'addestramento mentre il *case* 12 per il monitoraggio.

I risultati ottenuti sono riportati nell'appendice 1 dove nelle colonne sono riportati gli errori delle previsioni aggiornate ad ogni run. Nello specifico, la prima colonna contiene le previsioni effettuate prima del primo run, la seconda quelle effettuate prima del secondo run (avendo già osservato il primo), fino ad arrivare all'ultima colonna che contiene la previsione di usura dell'ultimo run prima che esso si verifichi. Si può notare dalle colonne che, generalmente, le previsioni per i livelli di usura dei run successivi sono più accurate rispetto a quelle sugli ultimi run previsti per l'utensile. Tuttavia, le previsioni di usura più utili sono quelle sui livelli più avanzati, quindi sugli ultimi run dell'utensile. Il risultato più importante di quest'analisi è rappresentato nella figura 9.4.3, dove si dimostra che l'accuratezza delle previsioni del livello di usura all'ultimo run del modello ARIMA migliora notevolmente già dopo tre run osservati.

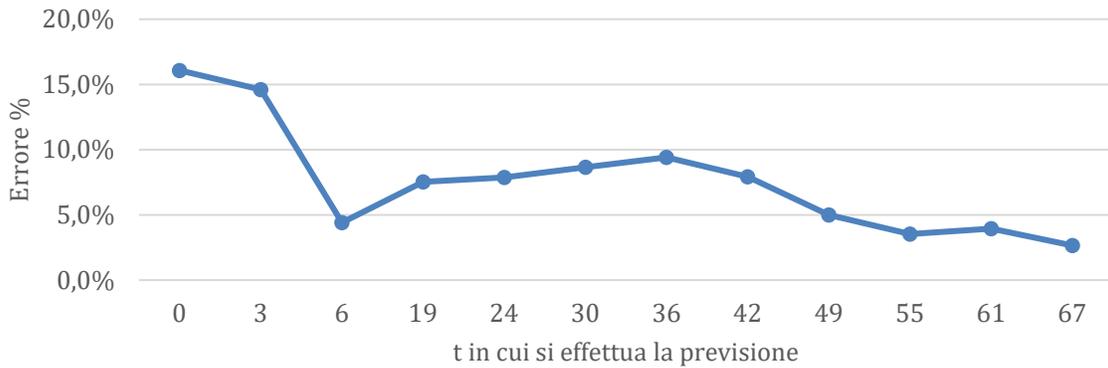


Figura 9.4.3 Variazione dell'errore % sulla previsione del livello di VB dell'ultimo run ($t=74$) con modello ARIMA

Nella figura 9.4.4 è possibile vedere come sono aggiornate le previsioni dopo 3 run osservati. Dalla figura (a) si vede come le previsioni effettuate senza nessuna osservazione dell'utensile da monitorare tende a sovrastimare il livello di VB nella prima fase del ciclo vita dell'utensile mentre la sottostima nell'ultima fase, a causa della natura lineare del metodo. Mentre nella figura (b) è evidente come le previsioni diventano più accurate in seguito all'osservazione dei livelli di usura registrati nei primi 3 run. Si può concludere che l'approssimazione del modello ARIMA può essere accettata solo localmente, mentre rende il metodo poco accurato per modellare l'andamento di usura dell'intero ciclo vita dell'utensile.

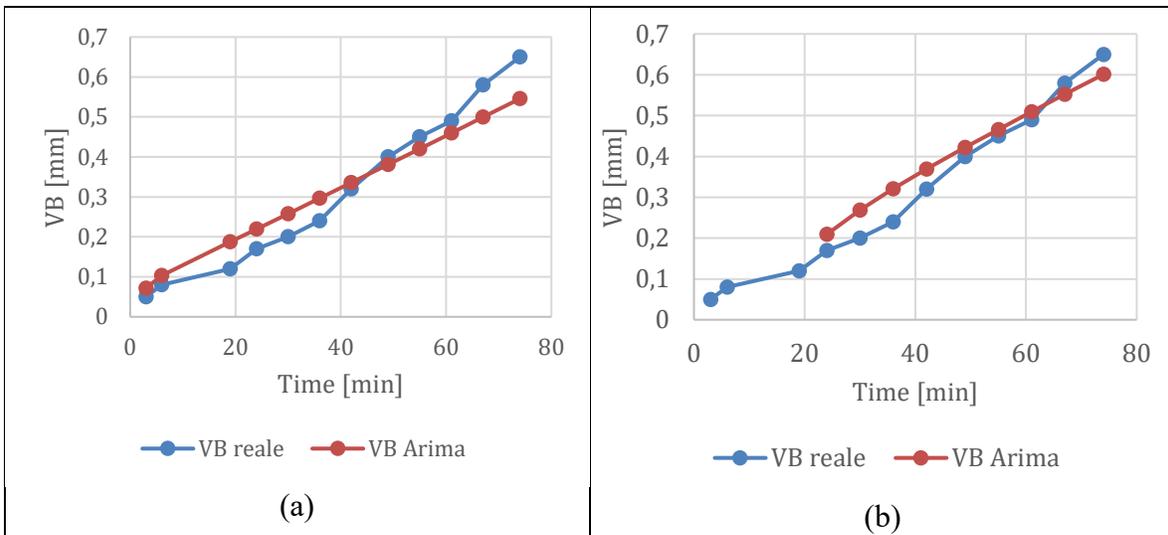


Figura 9.4.4 Previsioni ARIMA dinamico: (a) previsioni effettuate in $t=0$ (b) previsioni effettuate in $t=24$ (dopo l'osservazione di 3 run)

Nonostante il buon miglioramento dell'accuratezza delle previsioni dei livelli di usura del modello ARIMA va precisato che tale approccio dinamico non è sempre applicabile nei processi industriali. Infatti, esso richiede la misurazione della variabile di controllo, in questo caso il livello VB_{max} , del fenomeno di degradazione monitorato. Spesso tali misure possono essere effettuate solo in modalità *offline*, quando la macchina è ferma, risultando molto onerose in termini sia di tempo che di costi. Questo aspetto limita l'applicazione del modello presentato solo ai processi il cui fenomeno di degradazione è caratterizzato da una variabile di controllo direttamente misurabile con dei sensori in modo periodico e automatizzabile.

9.5 Modello di Taylor

Il modello di Taylor è stato implementato come riferimento ai metodi *physics-based*, in quanto il processo di degradazione analizzato può essere modellizzato con una legge fisica in termini matematici. È stata scelta la relazione estesa del modello di Taylor poiché è largamente conosciuta e nel caso in esame include tutti i parametri di lavorazione presenti nel dataset *mill*. Infatti, i parametri f e a dell'equazione 5.5.1 corrispondono agli attributi del dataset *feed* e *DOC* rispettivamente. Mentre il parametro VB_{max} corrisponde alla notazione sintetica VB utilizzata nel dataset, che si riferisce al tipo di usura *flank wear* descritto nel paragrafo 3.3.

Per effettuare le previsioni del livello di VB nel tempo è stata usata la relazione inversa dell'equazione 5.5.2. Come riportato nello script 5, i valori del dataset sono stati prima trasformati con la funzione logaritmica per poi effettuare una regressione lineare multipla con la funzione $lm()$ di R. Il risultato della regressione è la stima dei coefficienti empirici che compaiono nell'equazione 5.5.2, la quale può essere usata per prevedere il livello di VB dopo un determinato tempo di attività dell'utensile. Tali coefficienti sono relativi ad uno specifico utensile e materiale di lavorazione, ragione per cui è stato necessario impostare due differenti regressioni per i due materiali presenti nel dataset. Infine, le previsioni di VB sono state effettuate per ogni materiale con i relativi coefficienti stimati.

Script 5 Modello di Taylor per previsioni VB

```
#definizione coefficienti per relazione  $time=k*feed^s*DOC^w*VB^z$ 
coef_m1<- data.frame(k= NA, s= NA, w= NA, z= NA)
coef_m2<- data.frame(k= NA, s= NA, w= NA, z= NA)
fcst<- data.frame(case= NA, time= NA, VB= NA)

#trasformazione logaritmica dei valori VB, time, DOC e feed escludendo i valori 0
(log(0)=-inf)
Intrain_m1<-log(dplyr::filter(train_m1,VB!=0&time!=0)                                %>%
dplyr::select(VB,time,DOC,feed))
```

```

Intrain_m2<-log(dplyr::filter(train_m2,VB!=0&time!=0)
dplyr::select(VB,time,DOC,feed)) %>%

#stima coefficienti con regressione lineare multipla
f_m1<-lm(formula=time~feed+DOC+VB,data=Intrain_m1)
f_m2<-lm(formula=time~feed+DOC+VB,data=Intrain_m2)
coef_m1[1,]<-f_m1$coefficients
coef_m1[1,1]<-exp(coef_m1[1,1])
coef_m2[1,]<-f_m2$coefficients
coef_m2[1,1]<-exp(coef_m2[1,1])

#previsioni sul test set
for (i in 1:nrow(test)) {
  fcst[i,]$case<-test[i,]$case
  fcst[i,]$time<-test[i,]$time
  if(test[i,]$material1){
    fcst[i,]$VB<-exp((log(test[i,]$time)-log(coef_m1$k)-coef_m1$s*log(test[i,]$feed)-
coef_m1$w*log(test[i,]$DOC))/coef_m1$z)
  }else{
    fcst[i,]$VB<-exp((log(test[i,]$time)-log(coef_m2$k)-coef_m2$s*log(test[i,]$feed)-
coef_m2$w*log(test[i,]$DOC))/coef_m2$z)
  }}
  result[nrow(result)+1,] <- list("Taylor", rmse(test$VB, fcst$VB), rrse(test$VB,
fcst$VB))
}

```

Prima di procedere con le previsioni dei livelli di VB è necessario analizzare il risultato della regressione usata per stimare i coefficienti. Infatti, per essere valida la relazione fisica prevista per il fenomeno di degradazione, l'analisi di regressione deve presentare un'alta significatività statistica. I risultati della regressione per la stima dei coefficienti del primo materiale, che sono rappresentativi anche del secondo, sono riportati nella tabella 9.5.1. Si può notare che è stato omesso il parametro della velocità di taglio tra i regressori, poiché, seppure sia il parametro che influenzi maggiormente la vita utile dell'utensile, negli esperimenti del dataset è tenuto costante a 200 m/min non consentendo di stimarne l'incidenza sulla propagazione dell'usura. Tuttavia, tutti gli altri parametri di lavorazione sono altamente significativi (al simbolo *** in tabella corrisponde un $\alpha=0$) e contribuiscono a spiegare il fenomeno in esame. Anche l'indicatore R^2 conferma che il modello scelto ha una forma funzionale adeguata al fenomeno di degradazione sottostante all'andamento del livello di VB, consentendo di spiegare oltre il 93% della varianza dei dati.

Tabella 9.5.1 Risultato regressione modello Taylor

Regressore	Coefficiente	Std. Error	t value	Pr(> t)	Significatività
(Intercetta)	4,3282	0,12882	33.599	0	***
Feed	-0,4155	0,10117	-4.107	0.000145	***
DOC	-1,0154	0,09814	-10.347	0	***
VB	1,0802	0.04190	25.782	0	***
					$R^2 = 0,9309$

Anche in questo caso è stato validato il modello di previsione con il metodo di *cross validation* con 10 iterazioni, dove per ognuna sono stati selezionati casualmente 5 case per materiale per la partizione di addestramento. Le performance ottenute dal modello sono riportate nella figura 9.5.1, dove è possibile vedere come il modello abbia ottenuto dei buoni risultati in entrambi gli indicatori. Infatti, l'RMSE ha una media di circa 0,096 (contro 0,18 ottenuto dal modello ARIMA) e l'RRSE una media di circa 0,42 (contro 0,79 del modello ARIMA) il che implica che il modello è in grado di apportare molte informazioni circa le stime future dei valori di VB. Tuttavia, l'ampiezza dei box-plot indica che il modello è fortemente influenzato dai dati di addestramento e di validazione selezionati. Questo fenomeno, che è possibile notarlo soprattutto dall'RRSE che va da un minimo di 0,26 a un massimo di 0,73, dimostra l'importanza delle fasi precedenti a quella di modellazione. In particolare, esso può essere controllato dalla fase di *data cleaning*, con l'obiettivo di rimuovere tutti i valori che possono influenzare negativamente sulla stima dei coefficienti del modello di previsione.

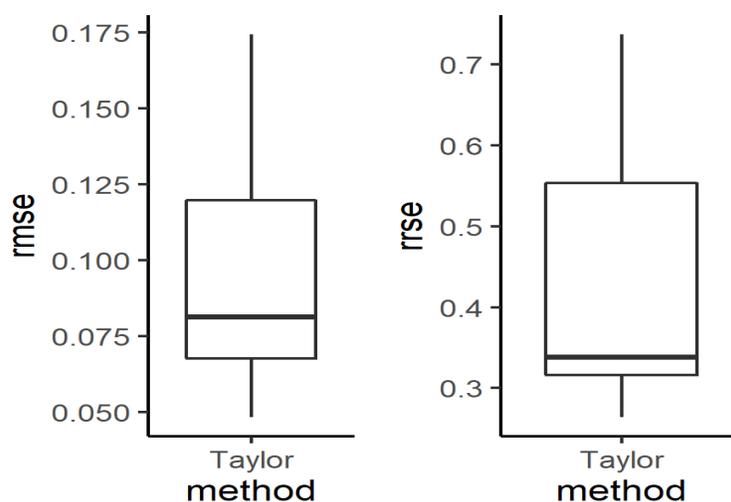


Figura 9.5.1 Performance modello di Taylor per previsioni VB sul test set

Dalla figura 9.5.2 è possibile osservare le previsioni effettuate dal modello di Taylor con le stesse condizioni del modello ARIMA nella figura 9.4.2. Si può notare come valgano le stesse considerazioni fatte per il modello ARIMA sulle limitazioni nel modellare le fasi esponenziali della curva di degradazione. Infatti, anche se la relazione di Taylor ha una forma esponenziale, i coefficienti sono stimare sulla forma linearizzata per poter effettuare la regressione lineare multipla. Rispetto al modello ARIMA, il modello di Taylor assicura di non effettuare previsioni inferiori a 0 e generalmente interpola meglio i punti dell'intero ciclo vita dell'utensile.

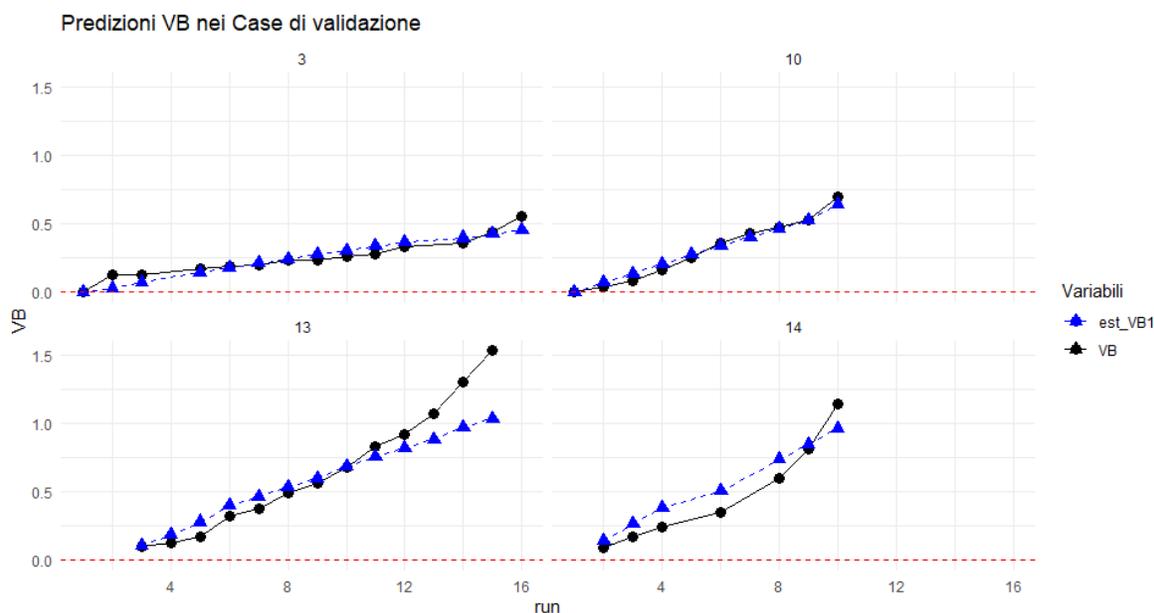


Figura 9.5.2 Confronto previsioni VB modello Taylor con valori effettivi

È stato testato un modello dinamico, come è stato fatto nel paragrafo 9.4.1 per il modello ARIMA, per valutare le variazioni di performance del modello di Taylor all'avanzare dei valori di usura osservati. È emerso che quest'ultimo, al contrario del modello ARIMA, tende a peggiorare l'accuratezza delle previsioni come è mostrato nella figura 9.5.3. Questo risultato determina che per incrementare l'accuratezza di un modello *physics-based* è necessario osservare un intero ciclo vita del componente (un *case* dell'utensile nel caso in esame) poiché, generalmente, la relazione fisica modella l'intero processo di degradazione.

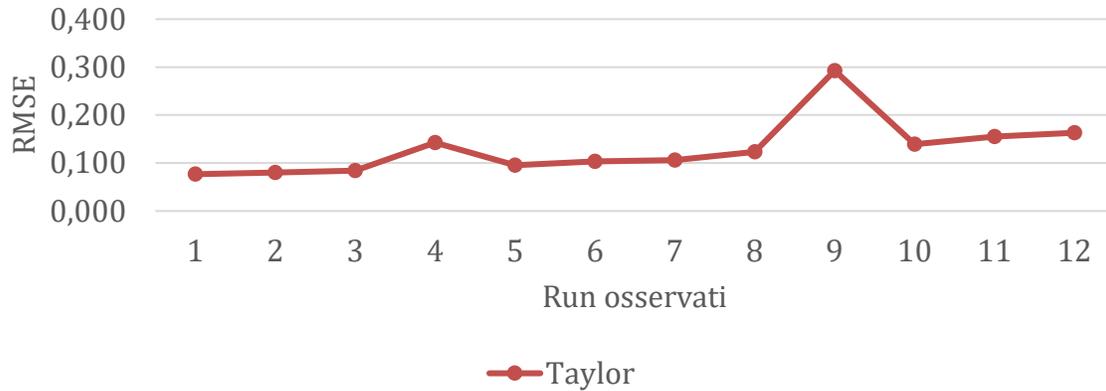


Figura 9.5.3 Andamento RMSE all'aumentare dei run osservati nel modello di Taylor

9.5.1 Modello Taylor con dataset ridotto

Il modello di Taylor descritto in precedenza è stato addestrato includendo i valori di VB di tutti i run, dei *case* dell'apposita partizione, presenti nel dataset. Tuttavia, la potenzialità dei metodi *physics-based* risiede nella possibilità di essere implementati con pochi dati a disposizione rispetto ai metodi *data-driven*, come descritto nel capitolo 5. In questo caso, esso è stato dimostrato, addestrando il modello con un solo valore di VB per ogni *case*, riducendo i dati per l'addestramento di ogni materiale da oltre 50 a sole 5 osservazioni. È stato scelto l'ultimo valore di VB per ogni *case*, come riportato nella tabella 9.5.2 per il primo materiale, poiché è nell'ultima fase che si verifica maggiormente l'andamento esponenziale del fenomeno di degradazione.

Tabella 9.5.2 Dataset ridotto per modello Taylor (materiale cast iron)

Case	T [min]	V[m/min]	f [mm/rev]	DOC [mm]	VB _{max} [mm]
2	72	200	0,5	0,75	0,55
3	81	200	0,25	0,75	0,55
4	39	200	0,25	1,5	0,49
9	46	200	0,5	1,5	0,81
10	57	200	0,25	1,5	0,7
11	105	200	0,25	0,75	0,76
12	74	200	0,5	0,75	0,65

Successivamente sono stati stimati i coefficienti della relazione fisica del modello con una regressione lineare multipla come è stato fatto in precedenza nello script 5. I risultati del metodo addestrato con una sola osservazione per *case*, modello denominato "Taylor ridotto", sono riportati nella figura 9.5.4. Come da aspettative, le performance del modello addestrato

con pochi dati sono inferiori a quelle del modello addestrato con tutti i dati. Nello specifico, il peggioramento delle prestazioni per le due metriche considerate è:

- L'RMSE peggiora da una media di 0,096 con deviazione standard pari a 0,0136 a una media di 0,155 e deviazione standard pari a 0,052;
- L'RRSE peggiora da una media di 0,42 con deviazione standard pari a 0,057 a una media di 0,66 e deviazione standard pari a 0,195.

Se i valori medi delle performance sono peggiorati di quasi due volte, la dispersione dei risultati nelle diverse iterazioni del processo di *cross validation* sono peggiorate di oltre tre volte. Ciò significa che il modello “ridotto” è molto più sensibile ai dati e quindi è in grado di effettuare previsioni meno robuste che dipendono fortemente dai dati di addestramento. Ciò è dovuto proprio alla bassa numerosità del campione utilizzato per effettuare la regressione, in cui basta un solo valore affetto da errori di misurazione che può influenzare l'intero modello di previsione. Tuttavia, le performance restano migliori rispetto a quelle registrate con il modello ARIMA nonostante siano stati impiegati solo un decimo dei dati per addestrare il modello. Quindi si può concludere che nei processi in cui si hanno pochi dati a disposizione, e si può modellare con un'equazione matematica il processo di degradazione, un metodo *physics-based* può essere una buona approssimazione per il monitoraggio del componente/utensile.

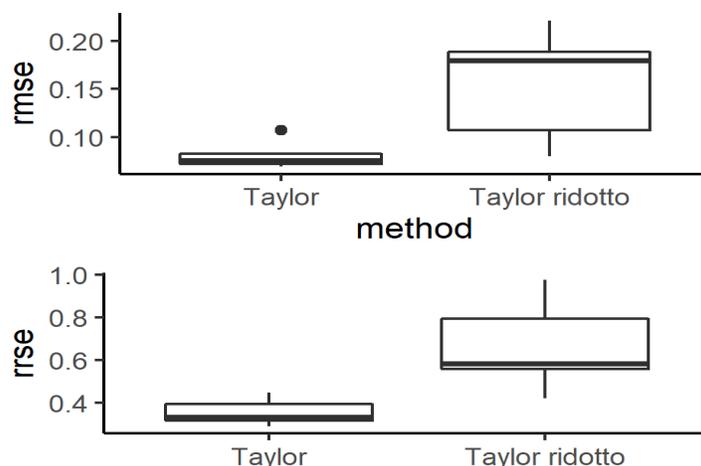


Figura 9.5.4 Prestazioni modello Taylor ridotto

9.6 Algoritmi di Machine Learning

L'implementazione degli algoritmi di *machine learning* ha permesso di includere nei modelli anche i dati registrati dai sensori. Tali dati non possono essere forniti ai modelli in modo diretto poiché sono stati registrati su istanti temporali multipli per ogni run. Pertanto, si è utilizzato l'output delle fasi di *feature extraction* e *feature selection* presentati nel paragrafo

7.1. I modelli scelti sono quelli maggiormente diffusi nell'ambito della manutenzione predittiva com'è emerso nello studio dello stato dell'arte presente nel capitolo 2. A questi è stato aggiunto il modello di regressione lineare per avere una baseline su cui confrontare le prestazioni. Le funzioni utilizzati sono riportati in sequenza nello script 6. In particolare:

- Per la regressione lineare (LM) è stata utilizzata la funzione *lm()* della libreria *stats*;
- Per l'algoritmo *neural network* (NN) è stata utilizzata la funzione *nnet()* dell'omonima libreria;
- La regressione lineare di *bayes* (BLM) è stata effettuata con la funzione *bayes.lm()* della libreria *bolstad*;
- L'algoritmo *random forest* (RF) è stato implementato tramite la funzione *randomForest()* dell'omonima libreria;
- Infine, l'algoritmo *boosted decision tree* (BDT) è stato implementato tramite la funzione *gbm()* dell'omonima libreria.

Script 6 Algoritmi di Machine Learning

```
#linear regression(LM) -----
model <- lm(f, data = train)
est_VB <- as.numeric(predict(model, test[X]))
result[nrow(result)+1,] <- list(k,"LM", rmse(test$VB, est_VB), rrse(test$VB, est_VB))

# neural network (NN)-----
train_control <- trainControl(method = "cv", number = 4)
model <- train(f, data = train, trControl = train_control, method = "nnet")
model <- nnet(f, train, size = 10, decay = 0.01, maxit = 2000)
est_VB_NN <- as.numeric(predict(model, test[X]))
result[nrow(result)+1,] <- list(k,"NN", rmse(test$VB, est_VB_NN), rrse(test$VB,
est_VB_NN))

#bayesian linear regression (BLM)-----
model <- bayes.lm(f,data = train)
est_VB <- as.numeric(predict(model, test[X]))
result[nrow(result)+1,] <- list(k,"BLM", rmse(test$VB, est_VB), rrse(test$VB, est_VB))

#random forest (RF)-----
model <- randomForest(f, data = train)
est_VB <- as.numeric(predict(model, test[X]))
result[nrow(result)+1,] <- list(k,"RF", rmse(test$VB, est_VB), rrse(test$VB, est_VB))

#boosted decision tree (BDT)-----
```

```

model <- gbm(f, data = train)
est_VB <- as.numeric(predict(model, test[X]))
result[nrow(result)+1,] <- list(k,"BDT", rmse(test$VB, est_VB), rrse(test$VB, est_VB))

```

Nella tabella 9.6.1 sono riportate le performance medie dei modelli, addestrati con 5 *tools* per materiale, sulle previsioni di tutti i run presenti nel *test set*. La media è stata calcolata sulle prestazioni ottenute ad ogni iterazione e nella tabella sono riportati in ordine di accuratezza. È stato testato anche il modello di Taylor così da confrontare i modelli di *machine learning* con uno tradizionale. È emerso che, tra gli algoritmi di *machine learning*, quello basato sulle reti neurali (NN) presenta le prestazioni migliori in entrambi gli indicatori. Mentre gli algoritmi basati sulla regressione lineare, regressione lineare multipla (LM) e regressione lineare Bayesiana (BLM), presentano un errore maggiore di oltre tre volte, a causa della forma lineare che non si adatta bene al fenomeno in esame. Inoltre, tali algoritmi presentano difficoltà nell'includere un numero elevato di regressori rispetto a un numero limitato di osservazioni [112].

Tuttavia, come si può notare dalla prima riga della tabella 9.6.1, è il modello di Taylor a presentare un errore di previsione minore. Nonostante esso sia addestrato solo con i parametri di produzione e sia un metodo *offline*. Infatti, effettua le previsioni del livello di usura ricevendo come input solo l'istante di tempo in cui si vuole stimare VB, ottenendo un'accuratezza migliore dei metodi *online* che possono aggiornare le loro previsioni sulla base degli input dei sensori, i quali monitorano indirettamente il processo di usura.

Tabella 9.6.1 Performance algoritmi di ML con 10/14 tools nel training set

	RMSE MEDIO	RRSE MEDIO
TAYLOR	0,1178	0,4439
NN	0,1389	0,5036
BDT	0,1874	0,6771
RF	0,1953	0,6918
LM	0,2378	0,9257
BLM	0,6668	2,6040

Dalla figura 9.6.1 è possibile confrontare con maggiore dettaglio le prestazioni dei due modelli risultati migliori. I box-plot rappresentano la dispersione delle metriche di performance per i due metodi al variare del partizionamento dei dati, in *training* e *test set* nelle 10 iterazioni di *cross validation*, al fine di trarre delle considerazioni più robuste. In valore medio, come presentato nella tabella precedente, il modello di Taylor è preferito in entrambe le misure degli errori. Ma osservando la distribuzione delle misure, la differenza è più marcata solo per l'RRSE mentre la distribuzione dell'RMSE tende in parti a sovrapporsi.

L'RMSE presenta una dispersione maggiore e, in particolare, il valore mediano è molto simile tra i due modelli, così come tende ad esserla la distribuzione inferiore della misura dell'errore. Ciò indica che la valutazione dei modelli è fortemente influenzata dal *dataset* e dal partizionamento scelto, limitazione superata in parte con l'approccio di *cross validation* in cui il partizionamento è ogni volta casuale il che consente di rimuovere l'influenza arbitraria sulla scelta dei dati da testare.

Dalla dispersione dell'RMSE si può dedurre che in condizioni ottimali, ovvero con dei dati di addestramento che apportano sufficiente informazione per il fenomeno da monitorare, i due modelli presentano un'accuratezza di previsione simile nonché migliore rispetto a tutti gli altri modelli analizzati. Mentre, il modello NN è maggiormente influenzato dalla qualità dei dati di addestramento, risultando meno robusto rispetto al modello di Taylor il quale presenta una minore dispersione dell'errore e una maggiore simmetria rispetto al valore mediano. Come sarà discusso nel paragrafo 9.6.2 il modello NN è fortemente dipendente dalle *feature* inserite, il che consente di ottenere un ampio margine di miglioramento del metodo fino a superare le performance del modello di Taylor il quale non può essere ottimizzato ulteriormente.

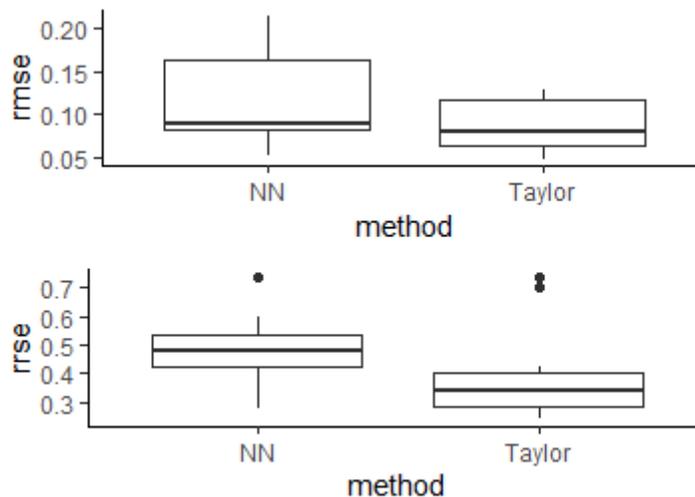


Figura 9.6.1 Box-plot confronto performance modelli di Taylor e NN

9.6.1 Confronto performance al variare della dimensione del training set

I diversi metodi presentati nel paragrafo precedente sono stati testati al variare del *training set* al fine di valutare l'influenza della quantità di dati sull'accuratezza del metodo. Nella tabella 9.6.2 sono riportate la media e la deviazione standard dell'RMSE, la prima per misurare le performance complessive nelle K=10 iterazioni di validazioni, la seconda per valutare la dispersione di tali misure.

Come da ipotesi, l'accuratezza nelle previsioni è migliorata per tutti i modelli. Tuttavia, tale variazione non è della stessa entità per tutti i modelli. Infatti, per i metodi che mostravano già delle buone performance, come quello di Taylor e l'algoritmo NN, con il passaggio dal *training set* composto da 8 tools a quello da 12 tools hanno mostrato un miglioramento di circa il 10%. Mentre, gli algoritmi LM e BLM che mostravano delle performance nettamente inferiori con il *training set* di piccole dimensioni raggiungono, diminuendo l'errore di oltre tre volte, delle performance quasi paragonabili agli altri metodi con il *training set* contenete l'85% dei tools. La dispersione delle misure, invece, segue un andamento meno condiviso tra i diversi metodi. Per il modello di Taylor e gli algoritmi LM e BLM essa diminuisce all'aumentare del *training set*, mentre per gli algoritmi NN, RF e BDT essa aumenta. Tale andamento conferma la robustezza del metodo *physics-based* che oltre a presentare l'errore medio complessivo minore è anche quello meno sensibile alle variazioni dei dati. L'aumento della dispersione per gli algoritmi di *ML* indica la loro dipendenza dalla qualità dei dati, in cui con grandi quantità di dati l'algoritmo più semplice, la regressione lineare, sembra essere più robusto rispetto agli altri analizzati.

Tabella 9.6.2 Statistiche RMSE algoritmi ML e modello di Taylor al variare del training set

RMSE	Tools usati nel training set					
	8/14 (57%)		10/14 (71%)		12/14 (85%)	
Metodo	Media	Dev. St.	Media	Dev. St.	Media	Dev. St.
Taylor	0,1044	0,0278	0,1178	0,0324	0,0922	0,0273
NN	0,1384	0,0385	0,1389	0,0447	0,1214	0,0523
RF	0,1968	0,0591	0,1953	0,0727	0,1874	0,0918
BDT	0,1856	0,0419	0,1874	0,0569	0,1875	0,0709
LM	0,7510	0,8202	0,2378	0,0510	0,2003	0,0382
BLM	2,2042	1,5231	0,6668	0,3866	0,3981	0,1684

Nella tabella 9.6.3 sono riportate le stesse statistiche relative alla misura dell'errore RRSE. Per quanto riguarda i valori medi si possono trarre delle considerazioni analoghe a quelle fatte per l'RMSE, in cui diminuisce l'errore per tutti i metodi all'aumentare delle dimensioni del *training set* e il modello di Taylor presenta sempre le migliori prestazioni. Tuttavia, per questa metrica i metodi basati sugli algoritmi di *ML*, in particolare NN, RF e BDT, presentano una dispersione minore risultando leggermente più robusti rispetto al modello di Taylor seppure con dei valori medi dell'errore maggiori. È interessante notare come in alcuni casi l'RRSE aumenti all'aumentare delle dimensioni del *training set* andando in controtendenza con l'RMSE. Questo fenomeno si può spiegare dalla forma dell'indicatore mostrata nel paragrafo 8.3 (equazione 8.3.5). Esso, infatti, confronta le previsioni effettuate dai modelli con il valore medio dei valori effettivi da prevedere. Quindi l'RMSE può diminuire se migliora l'accuratezza del metodo in esame (errore al numeratore) oppure se peggiora il metodo baseline (errore al denominatore), il quale dipende solo dai dati presenti

nel *test set*. Il valore medio può diventare una buona stima se tali dati sono omogenei tra loro, come nel caso di tools appartenenti allo stesso scenario. Così al diminuire del *test set* diminuisce anche l'eterogeneità dei tools da monitorare, rendendo più accurata la previsione baseline rappresentata dal valore medio. Per le caratteristiche descritte, è da preferire l'RMSE come misura dell'errore dei metodi poiché esso non dipende dalla distribuzione dei dati presenti nel *test set*.

Tabella 9.6.3 Statistiche RRSE algoritmi ML e modello di Taylor al variare del training set

RRSE	Tools presenti nel training set					
	8/14 (57%)		10/14 (71%)		12/14 (85%)	
Metodo	Media	Dev. St.	Media	Dev. St.	Media	Dev. St.
Taylor	0,4046	0,1215	0,4439	0,1453	0,3882	0,1655
NN	0,5132	0,0607	0,5036	0,1256	0,4667	0,1262
RF	0,7253	0,0876	0,6918	0,1217	0,7023	0,1390
BDT	0,7008	0,0856	0,6771	0,0956	0,7304	0,1145
LM	2,9195	2,8904	0,9257	0,3540	0,8584	0,2885
BLM	8,5860	6,3422	2,6040	1,7405	1,7509	0,9731

9.6.2 Sensibilità alla qualità dei dati degli algoritmi di ML

Come è stato discusso nel paragrafo precedente, il metodo *physics-based* presenta delle previsioni più robuste dei metodi *data-driven* basati sugli algoritmi di ML. Tuttavia, dalle ultime considerazioni è emerso che quest'ultimi presentano ampi margini di miglioramento che dipendono dalla qualità dei dati forniti per l'addestramento e degli input utilizzati per effettuare le previsioni.

Per supportare le conclusioni dell'analisi precedente, è stata fatta una comparazione puntuale delle previsioni ottenute con il metodo di Taylor e il metodo *data-driven* basato sull'algoritmo NN. Per non penalizzare il metodo *data-driven*, il quale è più sensibile alla quantità di dati usati per l'addestramento, il confronto è stato effettuato utilizzando un *training set* composto da 12 tools, 6 per materiale selezionati casualmente, lasciando solo un tool per materiale per testare le previsioni. In tali condizioni il metodo di Taylor risulta leggermente migliore come riportato nella seguente tabella:

Tabella 9.6.4 Confronto errore di previsione tra i metodi Taylor e NN

	TAYLOR	NN
RMSE	0,0402	0,0418
RRSE	0,2203	0,2289

Tuttavia, le performance di previsione del metodo *data-driven*, come riportato nella tabella 9.6.5, sono fortemente influenzate dai dati forniti dal sensore *AE_table* nel run 4 del case 7. Infatti, tali valori seguono un trend generalmente monotono crescente che segue l'andamento di VB, mentre nel run 4 le features di dominio temporale riportano dei valori nulli, in netto contrasto con i valori riportati nei run adiacenti. I valori di tali features sono probabilmente dovuti a un'anomalia registrata dal sensore *AE_table*, ma esse non sono state rimosse nella fase di *feature selection* perché complessivamente presentano una correlazione significativa con la variabile di controllo VB.

Tabella 9.6.5 Features AE Table Case 7

RUN	VB	AE_TABLE_CO	AE_TABLE_MIN	AE_TABLE_MEAN	AE_TABLE_SD
1	0	0,3058	0,3211	0,2367	0,1089
2	0,09	0,3241	0,3368	0,2682	0,1570
3	0,13	0,2675	0,2789	0,2007	0,0627
4	0,22	0,0000	0,0000	0,0000	0,0000
5	0,24	0,3967	0,4184	0,3545	0,1758
6	0,34	0,4043	0,4579	0,3127	0,1169
7	0,46	0,4054	0,4289	0,3241	0,1227

L'influenza del valore anomalo descritto nelle previsioni generate dall'algorithm NN si può osservare nella figura 9.6.2. Nel grafico a sinistra sono rappresentate le previsioni effettuate con l'inclusione del valore anomalo, che registra dei valori decrescenti nel run 4, in cui l'algorithm NN prevede un valore di VB nel run 4 addirittura inferiore di quello previsto nel run 3. Mentre nel grafico a destra sono rappresentate le previsioni ottenute dallo stesso algorithm escludendo, nell'intero dataset, le features del sensore anomalo. Si può subito notare che la previsione di VB per il run 4 è crescente rispetto al run precedente e l'errore di previsione, rappresentato con una freccia che misura la distanza tra il valore predetto e il valore effettivo, si riduce notevolmente.

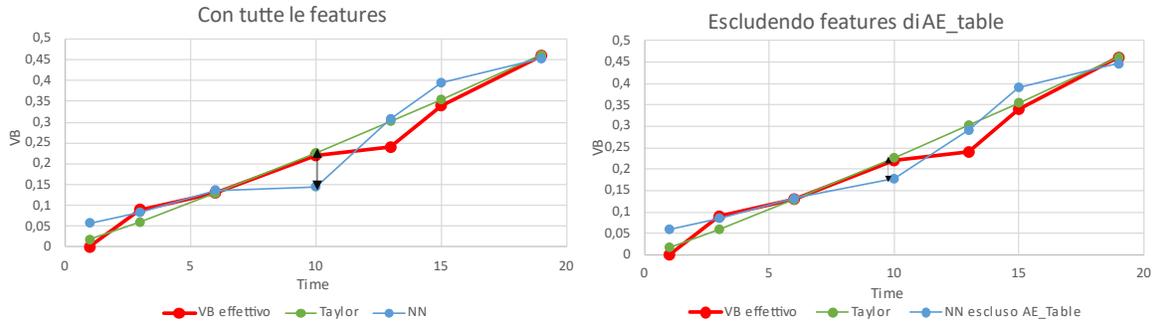


Figura 9.6.2 Previsioni Case 7 con rimozione features AE_table

L'errore di previsione complessivo ottenuto con l'algoritmo NN escludendo le feature anomale è risultato inferiore a quello ottenuto dal metodo di Taylor come riportato nella tabella 9.6.6. Tale risultato, ottenuto individuando singolarmente le *features* che influenzavano negativamente le performance di previsione del metodo *data-driven*, è stato ottenuto con lo scopo di dimostrare le potenzialità degli algoritmi di ML e la loro dipendenza dalla qualità dei dati forniti.

Tabella 9.6.6 Errore di previsione su test set composto da 2 tools con metodi Taylor e NN (esclusione feature AE_table)

	TAYLOR	NN
RMSE	0,0402	0.0354
RRSE	0,2203	0,1939

I valori della feature AE_table nel *case 7* rappresentano un'anomalia locale dei dati, che in un processo di manutenzione predittiva *online* verrebbero identificati come *outlier* e quindi esclusi dagli input per effettuare le stime. Tuttavia, si possono introdurre dei criteri per migliorare la qualità globale del dataset da usare per addestrare i modelli. Un primo criterio può essere quello di rimuovere le *features* che presentano un elevato numero di valori estremi, che corrispondono a 0 e 1 nel caso di dati normalizzati, poiché non apportano sufficiente informazione. Con il solo scopo di dimostrare la sensibilità di tali algoritmi alla qualità dei dati forniti sono state individuate le seguenti *features* nel dataset che presentavano dei valori estremi:

- *Vib_spindle_f_arg_rspb* con 30 valori pari a 1;
- *AE_table_f_arg_rspb* con 14 valori pari a 1;
- *AE_spindle_f_arg_rspb* con 16 valori pari a 1.

In seguito alla rimozione di tali *features* le performance del modello NN sono sensibilmente migliorate, come riportato nella tabella 9.6.7, raggiungendo un'accuratezza simile a quella del modello di Taylor. Tale miglioramento può essere spiegato anche con la riduzione del fenomeno di *overfitting*, in quanto un numero eccessivo di *features*, se sono disponibili pochi dati nel *training set*, può influenzare negativamente nella fase di addestramento. Infatti, all'aumentare dei regressori, i metodi *data-driven* necessitano di maggiori dati per modellare il fenomeno e generalizzare il trend per effettuare previsioni future con nuovi input [116].

Tabella 9.6.7 Confronto performance tra metodi Taylor e NN con esclusione feature anomale

MODELLO	RMSE	RRSE
TAYLOR	0,1037	0,4481
NN	0,1389	0,5036
NN CON ESCLUSIONE FEATURE	0,1099	0,4801

9.6.3 Ottimizzazione algoritmo NN

I modelli presentati nei paragrafi precedenti sono stati implementati, come suggerito dagli autori delle librerie, utilizzando i parametri di default delle rispettive funzioni. In seguito, è stata effettuata un'analisi sui parametri dell'algoritmo NN il quale verrà preso come riferimento dei metodi *data-driven*. È stata scelta la funzione *nnet()* che implementa una rete neurale su un singolo *layer*, la quale è risultata nettamente migliore rispetto alla funzione *neuralNet()* che consente di sviluppare la rete su più *layer*. I parametri che influenzano maggiormente l'algoritmo scelto, i cosiddetti *iperparametri*, sono il numero di nodi nascosti e i *weight decay* [117]. La ricerca dei parametri ottimali è stata effettuata tramite l'approccio *grid search* come riportato nel seguente script:

Script 7 Ottimizzazione iperparametri algoritmo NN

```
nnetGrid <- expand.grid(size = seq(from = 4, to = 12, by = 1),
                        decay = seq(from = 0.003, to = 0.011, by = 0.002))

nnetFit <- train(f,
                 data = train,
                 method = "nnet",
                 metric = "RRSE",
                 trControl = train_control,
                 tuneGrid = nnetGrid,
                 verbose = FALSE)
```

I risultati della *grid search* sono rappresentati nella figura 9.6.3, dove si può dedurre che il numero ottimale di nodi nascosti è tra 10 e 12 mentre il *weight decay* deve avere valori superiori a 0,005. Il punto di minimo con il dataset in esame si raggiunge con 10 nodi nascosti e con *weight decay* pari a 0,007, molto vicino ai parametri di default (10 e 0,01 rispettivamente). Addestrando l'algoritmo con i parametri ottimali si ottengono, rispetto alle performance ottenute con i parametri di default, dei miglioramenti in termini di accuratezza delle previsioni di circa il 2%. Considerando i miglioramenti ottenibili, in un'applicazione industriale specifica tale fase andrebbe approfondita ulteriormente. Tuttavia, le variazioni a questo livello di dettaglio non sono significative ai fini della ricerca in esame e l'analisi è stata effettuata con lo scopo di confermare la validità dei parametri usati in precedenza.



Figura 9.6.3 Risultato ottimizzazione iperparametri algoritmo NN

9.7 Analisi errore di previsione all'avanzare dei run

Dall'analisi dei valori di VB presentata nel paragrafo 9.2 è emerso che la variabile di controllo del fenomeno di usura mostra una variabilità crescente all'avanzare dei run. Da questa caratteristica del fenomeno da monitorare sono state effettuate delle analisi degli errori sulle previsioni del livello di usura per ogni run. La metrica utilizzata per misurare l'errore aggregato sui livelli di VB per un determinato run (j) su un insieme di tools è l'RMSE. Sono stati analizzati i modelli che hanno presentato le performance complessive migliori: quello di Taylor (*physics-based*) e quello basato sull'algoritmo *neural network* (NN, appartenente ai modelli di *machine learning*).

Come mostrato nella figura 9.7.1, è emerso che nei due modelli gli errori mostrano un trend differente all'avanzare dei run. In particolare, il modello di Taylor è caratterizzato da un andamento parabolico verso il basso dove la distribuzione dell'RMSE raggiunge il minimo nei run intermedi e tende a crescere nei run iniziali e finali. Il modello NN, al

contrario, mostra un andamento parabolico verso l'alto con la presenza del massimo dell'RMSE nei run centrali.

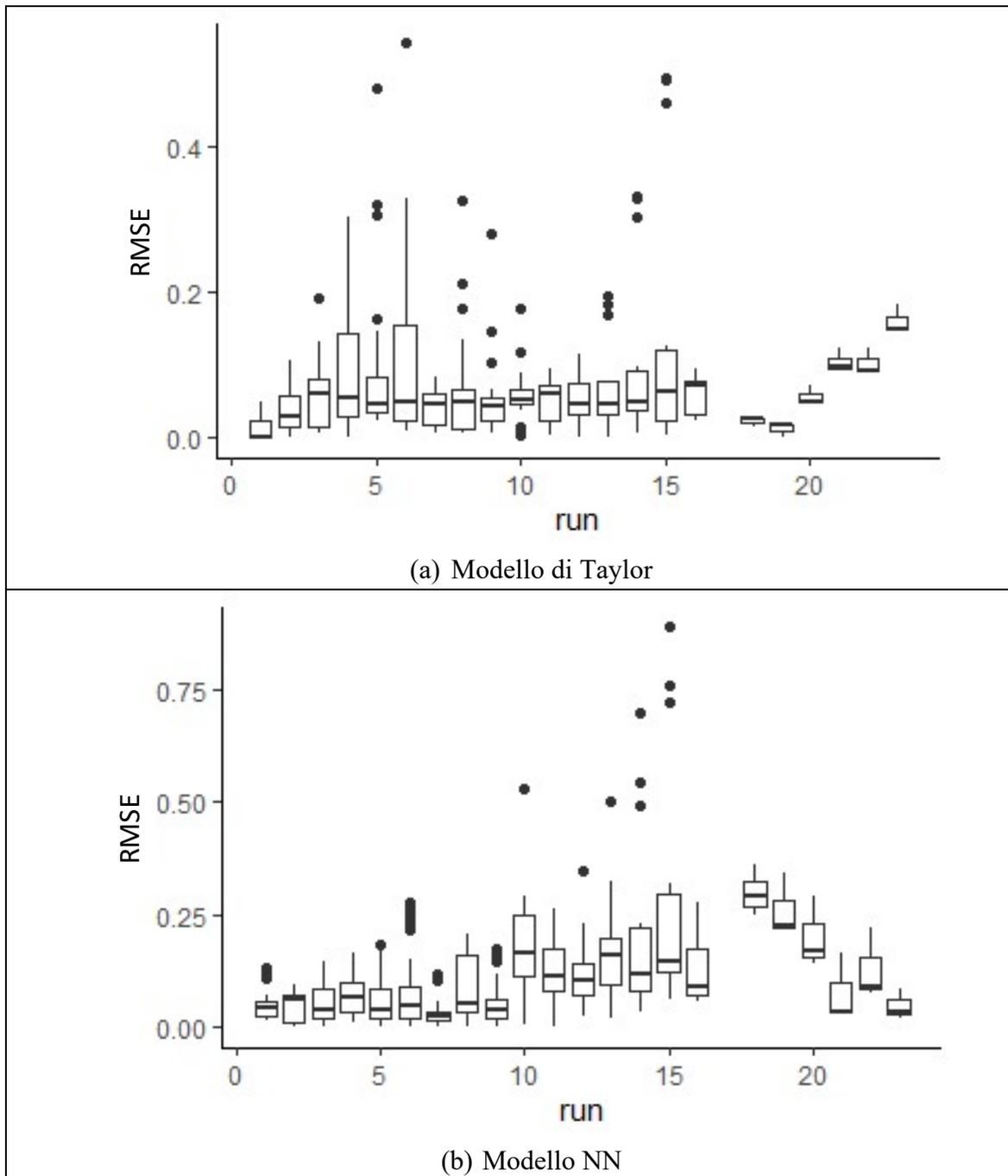


Figura 9.7.1 Distribuzione errore all'avanzare dei run nei modelli di Taylor e NN

Tale fenomeno può essere spiegato con la figura 9.7.2, dove l'andamento dell'errore di previsione all'avanzare dei run è confrontato con la curva tipica di diffusione del fenomeno

di usura, la *flank wear* (VB), analizzato. Dalla figura è evidente che l'errore di previsione dei due modelli è correlato con la fase di usura, determinata dal numero di run, dell'utensile. Tale risultato è in linea con le aspettative teoriche presentate nei capitoli precedenti, in particolare nel paragrafo 2.6 in cui sono descritte le potenzialità di diversi metodi. In questo caso, le fasi di usura che presentano andamenti non lineari (usura iniziale e usura esponenziale) sono modellate meglio da modelli che non hanno assunzioni di linearità, come dimostra l'andamento dell'errore del modello NN. Mentre nella fase centrale, in cui il processo di usura segue un andamento lineare, performa meglio il modello di Taylor, il quale è stato implementato tramite una regressione lineare multipla.

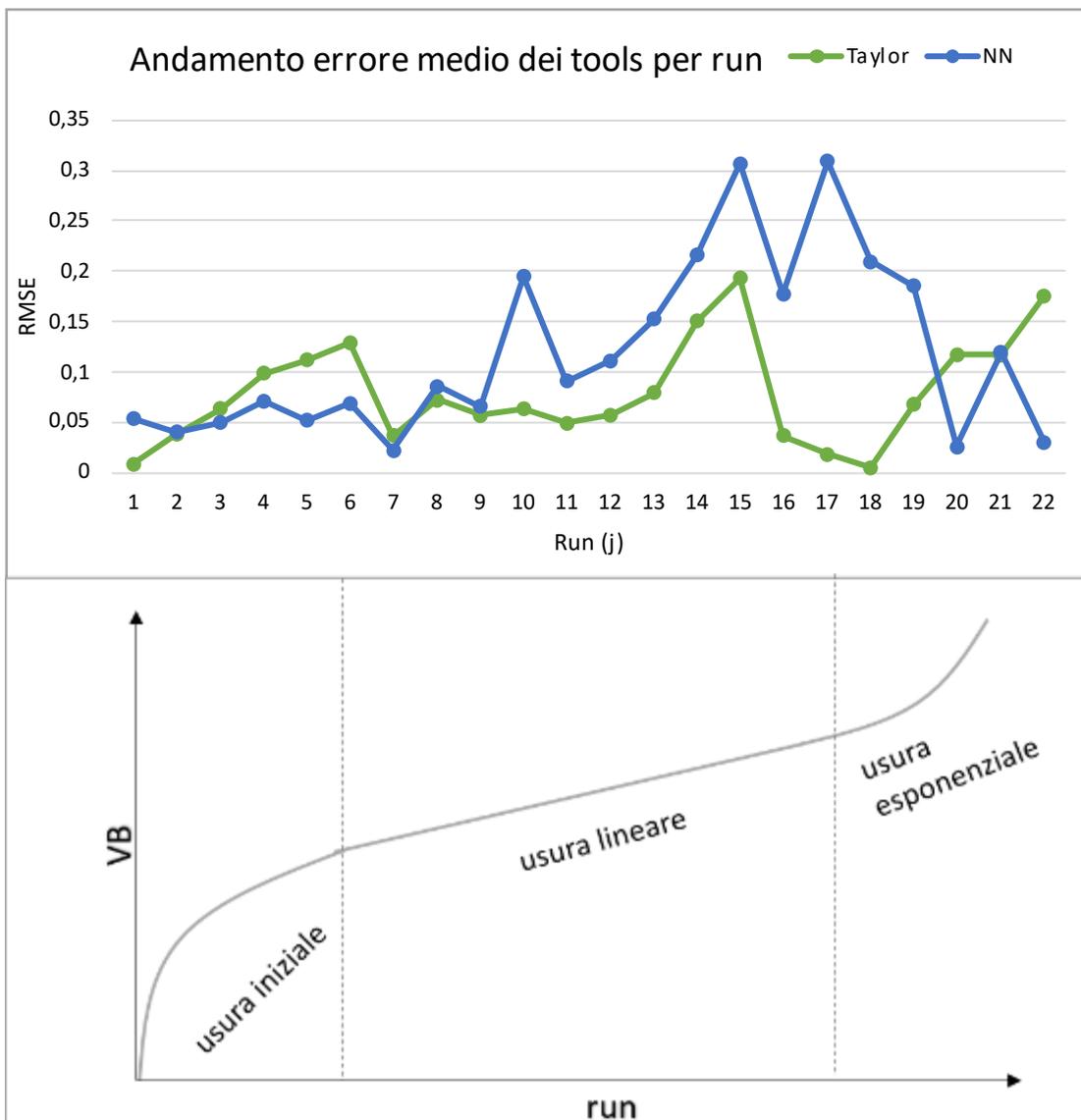


Figura 9.7.2 Confronto andamento errori dei modelli Taylor e NN all'avanzare dei run con curva teorica di propagazione della *flank wear*

9.8 Modello ibrido

Come è emerso dall'analisi degli errori di previsione all'avanzare dei run, l'accuratezza dei metodi *physics-based* e di quelli *data-driven* varia nel corso del fenomeno di usura. Dove, in particolare, nei tratti lineari sono più accurati i primi e in quelli non lineari i secondi. Da questo risultato è stato ideato un modello ibrido tra i due approcci che per ogni run genera una previsione dalla combinazione lineare delle previsioni dei modelli singoli. In questo modo si sfruttano le potenzialità di ogni metodo per il tratto dell'andamento dell'usura dove è più efficace. Il modello ibrido sviluppato presenta i seguenti step:

1. Addestramento dei modelli di Taylor e NN singolarmente con *training set* I_{TR} come mostrato nei paragrafi 9.5 e 9.6;
2. Generazione di un set di previsioni di VB, per ogni run (j), sul *training set* con entrambi i modelli singoli: $VB_{Taylor,j}$ e $VB_{NN,j}$ con $j = \{1, 2, \dots, \max_i(\text{run}_{max,i})\}$ ¹⁶, dove il primo pedice si riferisce al modello con quale sono state fatte le previsioni e il secondo indica il run j-esimo di cui si effettuano le previsioni per tutti i tools che presentano l'osservazione di tale run in I_{TR} .¹⁷
3. Calcolo pesi $w_j \in [0,1]$ ¹⁸ per ogni run j, per generare le combinazioni lineari delle previsioni come riportato nella seguente equazione:

$$VB_{hybrid,j}(w_j) = w_j * VB_{Taylor,j} + (1-w_j) * VB_{NN,j}$$

Equazione 9.8.1 Calcolo previsioni di VB del modello ibrido

Dove il peso w_j è il peso che minimizza l'RMSE delle previsioni del modello ibrido per il run j-esimo sui tools di I_{TR} , dove la condizione può essere scritta nel seguente modo:

$$\min_{w_j} RMSE_j (VB_{effettivi,j} ; VB_{hybrid,j}(w_j))$$

Equazione 9.8.2 Condizione per calcolare i pesi del modello ibrido

¹⁶ $run_{max,i}$ si riferisce all'ultimo run effettuato dal tool i-esimo. Il max su i è per individuare il tool che ha effettuato il numero maggiore di run.

¹⁷ Per esempio, $VB_{Taylor,1}$ è il vettore di previsioni di VB effettuare con il modello di Taylor sui tools in I_{TR} al run 1.

¹⁸ Calcolati su un insieme discreto con passo di 0,01, quindi più nello specifico w_j appartiene all'insieme $\{0; 0,01; 0,02; \dots 0,99; 1\}$.

4. Calcolo previsioni sul *test set* con media pesata (equazione 9.8.1) delle previsioni di Taylor e del modello NN su I_{TEST} mediate i pesi w_j calcolati nella fase di addestramento (punto 3).

La procedura descritta è stata implementata mediante l’algoritmo riportato nello script 8. In esso il vettore dei pesi, contenuto nel dataframe “weights” che lo collega al vettore dei numeri di run, è inizializzato con il valore 1, corrispondente al modello di Taylor poiché è quello che ha presentato l’accuratezza complessiva migliore (paragrafo 9.6). Anche l’RMSE di riferimento, denominato “rmse_best”, è inizializzato con le previsioni relative al modello di Taylor (“est_VB_T_train”) per il medesimo motivo. L’algoritmo procede con il confronto dell’RMSE ottenuto con il modello ibrido per ogni peso tra 0 e 1 con passo di 0,01 con l’RMSE di riferimento. Quest’ultimo è aggiornato ogni volta che si ottiene un valore migliore di quello ottenuto nelle iterazioni precedenti. Per ogni run il peso che viene salvato nel vettore dei pesi è quello corrispondente alla combinazione lineare dei modelli singoli che ottiene l’RMSE minore al termine di tutte le iterazioni. Nel caso in cui per un determinato run non ci siano osservazioni nel *training set*, il peso corrispondente è fissato pari a quello del run precedente poiché, com’è emerso dall’analisi degli errori nel paragrafo precedente, l’accuratezza dei modelli tende a polarizzarsi in determinati intervalli. Nella seconda parte dell’algoritmo, dopo aver calcolato i pesi, si effettuano le previsioni sul *test set* con la formula dell’equazione 9.8.1.

Script 8 Algoritmo modello ibrido tra Taylor e NN

```
# calcolo pesi per run
Weights <- data.frame(run = NA, w = NA) %>% dplyr::filter(!is.na(run))
weights[1,] <- list(1,1)
for(r in 1:max(mill$run)){
  weights[r,1] <- r
  if(nrow(dplyr::filter(train, run==r))>0){
    vb_run <- dplyr::filter(train, run==r) %>% select(VB)
    rmse_best <- rmse(vb_run$VB,dplyr::filter(est_VB_T_train, run==r)$VB)
    for(p in 0:100){
      p <- p/100
      est_VB <- p*(dplyr::filter(est_VB_T_train, run==r) %>% select(VB))+
      (1-p)*(dplyr::filter(est_VB_NN_train, run==r) %>% select(VB))
      if(rmse(vb_run$VB,est_VB$VB)<=rmse_best){
        rmse_best <- rmse(vb_run$VB,est_VB$VB)
        weights[r,2] <- p
      }
    }
  }
}
```

```

}else
  weights[r,2] <- weights[r-1,2]
}

#calcolo previsioni con modello ibrido
est_VB_h <- data.frame(case=NA,run=NA,VB=NA) %>% dplyr::filter(!is.na(case))
for(t in 1:nrow(test)){
  p <- as.numeric(dplyr::filter(weights, run==test[t,]$run)$w)
  est_VB_h[t,]$case <- test[t,]$case
  est_VB_h[t,]$run <- test[t,]$run
  est_VB_h[t,]$VB <- p*est_VB_T[t] + (1-p)*est_VB_NN[t]
}

```

Nella figura 9.8.1 sono rappresentati i pesi ottenuti per il modello ibrido con un *training set* composto da 10 tools, dove con valori tendenti ad 1 prevale il modello di Taylor e per valori tendenti a 0 prevale NN. Per il primo run prevale Taylor poiché esso si riferisce a un tempo di utilizzo dell'utensile vicino a 0, in cui anche il livello di VB presenta valori nulli, e la relazione matematica prevede un livello di VB nullo per $T = 0$ per definizione. Mentre nei run successivi i pesi rispecchiano i risultati ottenuti con l'analisi degli errori, in cui si verifica una fase lineare intermedia (dal run 13 al run 19) in cui è più accurato il modello di Taylor e quindi i pesi sono tendenti a 1, mentre nelle fasi esterne prevale il modello *data-driven* basato sull'algorithm NN.

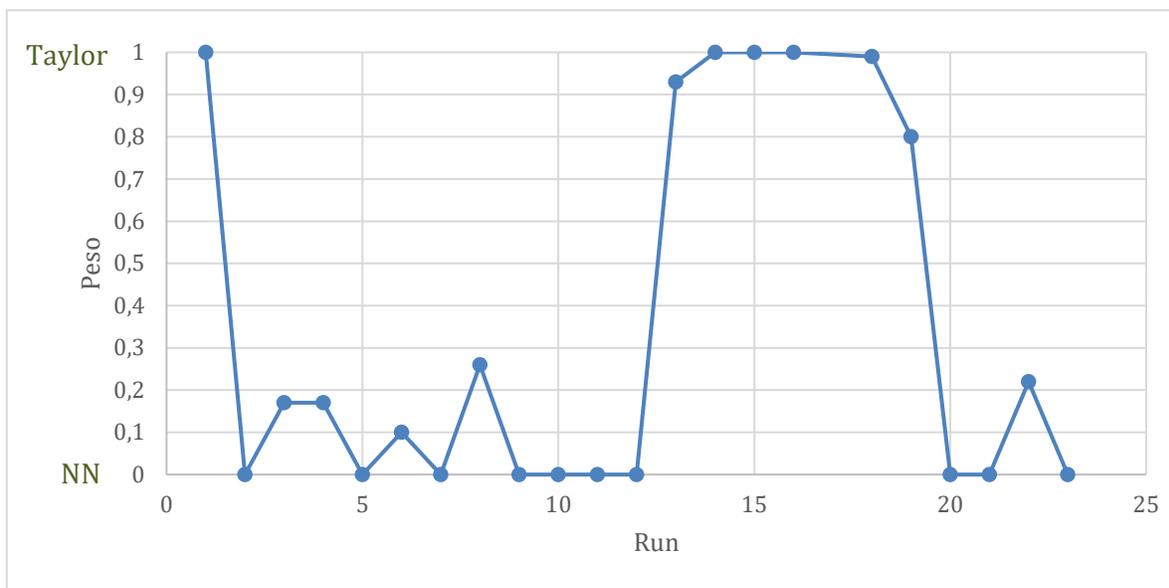


Figura 9.8.1 Pesi ottenuti nel modello ibrido

Gli errori di previsione sul test set ottenuti nelle $K = 10$ iterazioni del processo di *cross validation* con il *training set* composto da 10 tools sono riportate nell'appendice 2, mentre nella figura 9.8.2 sono rappresentati gli errori complessivi del modello ibrido e di quelli singoli. Sotto il profilo dell'RMSE il modello ibrido presenta una distribuzione simile a quella del modello NN, leggermente spostata verso il basso e con un outlier in meno. Mentre analizzando la distribuzione dell'RRSE, la distribuzione del modello ibrido ha un valore mediano simile ai modelli singoli ma con una dispersione molto minore. Si può dedurre che il modello ibrido migliora solo leggermente l'accuratezza complessiva delle previsioni, mentre il vantaggio principale di tale approccio è quello di ottenere un metodo più robusto la cui bontà delle previsioni è meno influenzata dai dati di addestramento rispetto ai modelli singoli.

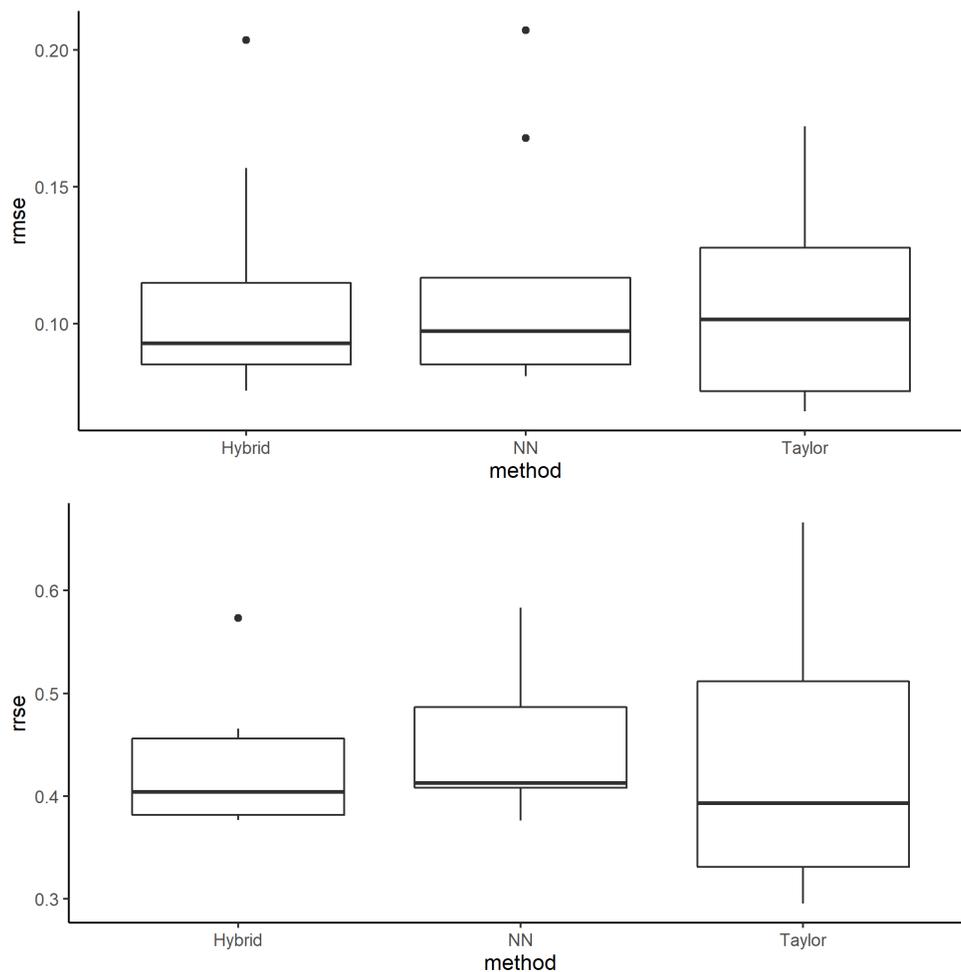


Figura 9.8.2 Prestazioni modello ibrido confrontate con modelli singoli

Come mostrato nella figura 9.8.3, sono state analizzate le prestazioni del modello ibrido e dei relativi modelli singoli al variare della dimensione dell'insieme dei dati di addestramento. L'errore misurato con la metrica RRSE è sempre minore con il modello ibrido rispetto a quelli singoli, seppure le differenze sono più marcate con *training set* di maggiori dimensioni. Mentre, analizzando i valori della metrica RMSE, è emerso che con un *training set* di piccole dimensioni (composto da 8/14 tools) è più accurato il modello di Taylor perché non ci sono abbastanza dati per addestrare la rete neurale, la quale presenta un errore nettamente superiore rispetto al metodo *physics-based*, e quindi il modello ibrido presenta delle prestazioni intermedie tra essi. Mentre all'aumentare della dimensione del *training set* l'errore del modello NN diventa molto simile a quello del modello di Taylor e di conseguenza il modello ibrido ottiene delle prestazioni migliori dei modelli singoli. Inoltre, si può osservare che con il modello ibrido l'incremento dell'accuratezza all'aumentare dei tools usati nel *training set* è maggiore rispetto a quello ottenuti dai modelli singoli come attesta la maggiore ripidità della curva relativa ad esso nella figura.

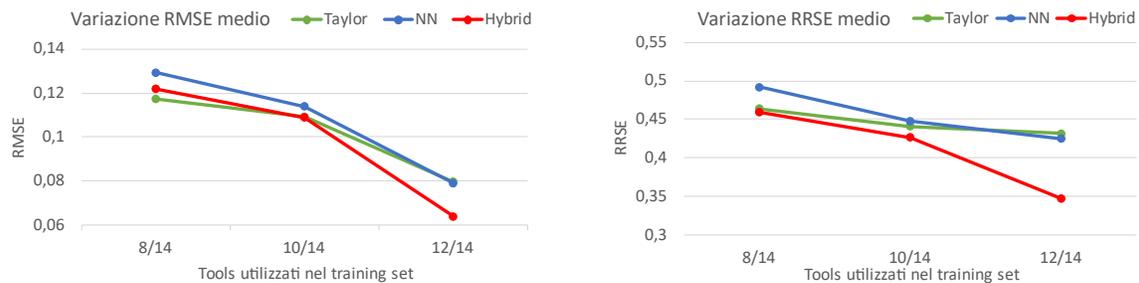


Figura 9.8.3 Confronto prestazioni modello ibrido al variare della dimensione del training set

10 CAPITOLO

CONCLUSIONI E LAVORI FUTURI

Quest'ultimo capitolo riporta nel primo paragrafo, in modo sintetico, i risultati ottenuti dalla ricerca sperimentale sul caso di studio *milling* descrivendo le caratteristiche dei diversi metodi analizzati con i relativi vantaggi e svantaggi. Mentre, nel secondo paragrafo, sono proposti dei possibili lavori futuri da effettuare per validare i risultati ottenuti e propone l'implementazione di un altro approccio di modello ibrido.

10.1 Conclusioni

I risultati ottenuti, dall'implementazione dei modelli nel capitolo precedente, rispettano le caratteristiche teoriche descritte per ogni categoria di metodi per il monitoraggio dell'usura degli utensili analizzata. Ogni categoria ha presentato delle prestazioni migliori delle altre in determinate circostanze, pertanto si può affermare che non è possibile identificare un metodo ottimo per descrivere tutti gli andamenti di usura, così come per l'ambito delle previsioni in generale non esiste un unico modello di riferimento [118].

In presenza di fenomeni di usura il cui parametro da monitorare consente di effettuare agevolmente delle misurazioni dirette, anche in modo periodico, i metodi di serie storiche possono essere una soluzione facile da implementare che garantisce buone prestazioni di previsione. Infatti, com'è stato mostrato con i modelli ARIMA, aggiornando i valori della variabile monitorata nel corso dell'avanzamento del fenomeno di usura essi sono in grado di aggiornare le previsioni con una buona flessibilità. Il vantaggio di tali metodi è che richiedono un numero limitato di sensori e necessitano di un basso carico computazionale per essere implementati. Tuttavia, spesso misurare direttamente la variabile da monitorare è troppo oneroso ed è necessario implementare un metodo che si basa su misurazioni indirette.

La categoria successiva analizzata è quella dei metodi *physics-based*. Tale approccio si è rilevato essere il più accurato e robusto nel caso in esame. Infatti, con il modello di Taylor sono state ottenute le migliori prestazioni complessive di previsione con diverse condizioni. Esso ha il vantaggio di poter essere implementato anche in assenza di sensori a bordo della macchina e con poche misurazioni per addestrare il modello. Infatti, è stato dimostrato che anche con un numero molto limitato di dati è possibile implementare un modello *physics-based* che fornisce una prima approssimazione delle previsioni da effettuare. La principale limitazione di tale metodo è che può essere applicato solo per i fenomeni di usura di cui si conosce una legge matematica che ne descrive l'andamento. Inoltre, seppure ha ottenute le migliori performance complessive, in determinate fasi del fenomeno di usura può essere più accurato un metodo *data-driven*.

Mentre la categoria dei metodi *data-driven* basata sugli algoritmi di *machine learning* ha dimostrato di avere delle buone potenzialità se ha a disposizione un numero sufficiente di dati. Infatti, addestrando adeguatamente tali algoritmi è possibile ottenere delle prestazioni simili a quelle dei metodi *physics-based* senza introdurre nessuna relazione matematica che descrive il fenomeno di usura. È emerso che l'algoritmo migliore per descrivere il fenomeno della *flank wear* con i *monitoring factors* del dataset *milling* è quello basato sulle reti neurali (denominato NN). Tale metodo ha ottenuto delle prestazioni complessive solo leggermente inferiori a quelle del metodo *physics-based*, mentre si è rivelato essere migliore nel modellizzare i tratti di andamento non lineare. Le limitazioni che riguardano questi metodi sono la necessità di installare un elevato numero di sensori sulle macchine, predisporre di apposite apparecchiature per l'elaborazione delle misurazioni e infine di raccogliere una grande quantità di dati per addestrare gli algoritmi. Il tutto si traduce in un elevato investimento iniziale, che offre la possibilità di monitorare con efficacia anche fenomeni di cui non si dispone di una legge fisica specifica.

Infine, l'approccio ibrido definito dalla combinazione lineare tra un metodo *physics-based* e uno *data-driven* ha consentito di ottenere delle prestazioni migliori di entrambi i metodi singoli in determinate condizioni. Infatti, se si impiega un numero limitato di dati per l'addestramento del metodo *data-driven* è preferibile il metodo *physics-based* singolarmente. Mentre, quando i modelli singoli ottengono delle prestazioni simili tra loro, l'approccio ibrido consente di incrementare sensibilmente l'accuratezza complessiva e soprattutto di ottenere dei risultati molto più robusti. Ciò è possibile grazie ai pesi della combinazione lineare calcolati per ogni lavorazione dell'utensile che consentono di selezionare il metodo migliore per ogni fase, lineare o non lineare, dell'andamento dell'usura.

10.2 Lavori futuri

I risultati più interessanti ottenuti da questa ricerca di tesi sono le prestazioni ottenute con i metodi *physics-based* e con gli algoritmi di *machine learning* e con il relativo modello ibrido. Una prima ricerca per validare i risultati è quella di applicare gli stessi metodi su un dataset con gli stessi *monitoring factors* di dimensioni maggiori, data la necessità della grande quantità di dati necessari per addestrare i modelli. Con un'ulteriore ricerca si potrebbe validare questi risultati su un altro caso di studio, possibilmente su un fenomeno di usura differente.

Inoltre, è possibile implementare un approccio ibrido tra un metodo di serie storiche (come il modello ARIMA) e un algoritmo di *machine learning*. Infatti, quest'ultimi consentono di effettuare una stima dello stato di avanzamento dell'usura nel momento in cui effettuano la stessa sulla base dei dati dei sensori, mentre con i metodi di serie storiche si

possono effettuare delle stime future sulla base dei dati osservati. Quindi, dalla combinazione dei due metodi si possono ottenere delle previsioni future sul livello di usura sulla base delle misurazioni indirette fornite dai sensori. Si possono implementare due approcci speculari: il primo prevede di stimare i valori futuri dei sensori con il metodo delle serie storiche e con essi stimare il livello di usura con l'algoritmo di *machine learning*; con il secondo invece si effettuano le stime del livello di usura con l'algoritmo di *machine learning* con i dati dei sensori presenti e si effettuano le previsioni future dell'andamento di usura con il metodo delle serie storiche.

APPENDICE 1

Previsioni modello ARIMA dinamico

Time [min]	VBreale [mm]	Errore previsioni effettuate in t (VBreale – VBstimato) [mm]												
3	0,05	0,02												
6	0,08	0,02	0,01											
19	0,12	0,07	0,06	0,09										
24	0,17	0,05	0,04	0,08	0,04									
30	0,2	0,06	0,05	0,10	0,07	0,06								
36	0,24	0,06	0,05	0,11	0,08	0,08	0,05							
42	0,32	0,02	0,01	0,07	0,05	0,05	0,03	0,01						
49	0,4	-0,02	-0,02	0,04	0,02	0,02	0,01	-0,01	0,01					
55	0,45	-0,03	-0,03	0,04	0,02	0,01	0,00	-0,01	0,01	0,03				
61	0,49	-0,03	-0,02	0,04	0,02	0,02	0,01	0,00	0,02	0,04	0,04			
67	0,58	-0,08	-0,07	-0,01	-0,03	-0,03	-0,04	-0,04	-0,03	-0,01	0,00	-0,01		
74	0,65	-0,10	-0,09	-0,03	-0,05	-0,05	-0,06	-0,06	-0,05	-0,03	-0,02	-0,03	0,02	
	ME%	14,8%	12,4%	17,0%	10,7%	9,6%	6,3%	4,4%	4,7%	5,0%	3,8%	3,0%	2,6%	
	RMSE	0,0535	0,0490	0,0696	0,0467	0,0453	0,0342	0,0308	0,0286	0,0289	0,0260	0,0198	0,0172	

APPENDICE 2

Performance di accuratezza dei metodi singoli e del modello ibrido nelle $K = 10$ iterazioni di *cross validation*.

ITERAZIONE	METHOD	RMSE	RRSE
1	Taylor	0,0694	0,2955
1	NN	0,0972	0,4138
1	Hybrid	0,0901	0,3834
2	Taylor	0,1722	0,6662
2	NN	0,1170	0,4526
2	Hybrid	0,1198	0,4634
3	Taylor	0,0731	0,3274
3	NN	0,0882	0,3952
3	Hybrid	0,0848	0,3801
4	Taylor	0,1298	0,3849
4	NN	0,1678	0,4979
4	Hybrid	0,1569	0,4655
5	Taylor	0,1009	0,5006
5	NN	0,0830	0,4121
5	Hybrid	0,0759	0,3766
6	Taylor	0,1722	0,6662
6	NN	0,0972	0,3763
6	Hybrid	0,1006	0,3892
7	Taylor	0,1218	0,3427
7	NN	0,2073	0,5832
7	Hybrid	0,2037	0,5733
8	Taylor	0,1022	0,5157
8	NN	0,0808	0,4078
8	Hybrid	0,0756	0,3816
9	Taylor	0,0680	0,3077
9	NN	0,1163	0,5267
9	Hybrid	0,0957	0,4334
10	Taylor	0,0825	0,4016
10	NN	0,0841	0,4093
10	Hybrid	0,0860	0,4186

RIFERIMENTI

- [1] R.K. Mobley, An Introduction to Predictive Maintenance, Van Nostrand Reinhold, 1990.
- [2] Ferret, The basics of predictive/preventive maintenance, 2006.
- [3] G. Hale, "InTech survey: Predictive maintenance top technology challenge in 2007," in *InTech*, 2007.
- [4] B. Cline, Predictive maintenance applications for machine learning, 2017.
- [5] A. Heng, Rotating machinery prognostics: State of the art, challenges and opportunities, 2009.
- [6] I. Bazovsky, Reliability Theory and Practice, Prentice-Hall, Englewood Cliffs, 1961.
- [7] H.P. Bloch, Machinery failure analysis and trouble shooting, 1997.
- [8] F. Trojan, Proposal of Maintenance-types Classification to Clarify Maintenance Concepts in Production and Operations, 2017.
- [9] T. P., Carvalho, A systematic review of machine learning methods applied to predictive maintenance, 2019.
- [10] K. F Gallimore, A framework for developing maintenance strategies, 1988.
- [11] J. Moubray, Reliability-centered Maintenance, 1997.
- [12] Q. Zhang, Remaining useful life estimation for mechanical systems based on similarity of phase space trajectory, 2015.
- [13] Y. Lei, A review on empirical mode decomposition in fault diagnosis of rotating machinery, 2013.
- [14] N. Li, An improved exponential model for predicting remaining useful life of rolling element bearings, 2015.
- [15] J. Daily, Predictive Maintenance: How Big Data Analysis Can Improve Maintenance, 2016.
- [16] M. S. Kan, A review on prognostic techniques for non-stationary and non-linear rotating systems, 2015.
- [17] B. Colding, A wear relationship for turning, milling and grinding, Doctoral Thesis, KTH, Stockholm, 1959.
- [18] D. Johansson, Assessment of Commonly Used Tool Life Models in Metal Cutting, 2017.
- [19] B. Li, A review of tool wear estimation using theoretical analysis and numerical simulation technologies, 2012.
- [20] Z. Yan, Remaining useful life prediction for aero-engine based on the similarity of degradation characteristics, 2019.
- [21] R. Serra, Cutting tool life management in turning process: a new approach based on a stochastic wear process and the Cox model, 2017.
- [22] G. Poulachon, Sulphide inclusion effects on tool-wear in high productivity milling of tool steels, 2002
- [23] D. Johansson, Tool life and wear model in metal cutting, Influence of varying flank wear criterion on colding's tool life equation, 2014.

- [24] D. Johansson, Tool life and wear modelling in metal cutting, based on combining the archard and the colding equations, 2016.
- [25] F. S. Nowlan, Reliability centered maintenance, 1978.
- [26] S. Gardner, Exponential smoothing: The state of the art—Part II, 2006.
- [27] Y. Hu, Exponential Smoothing Model for Condition Monitoring: A Case Study, 2013.
- [28] D. Lewin, Predictive maintenance using PCA, 1995.
- [29] B. Zhang, Residual Useful Life Prediction for Slewing Bearing Based on Similarity under Different Working Conditions, 2018.
- [30] W. Heng, Research on rolling bearing state health monitoring and life prediction based on PCA and Internet of things with multi-sensor, 2020.
- [31] G. Susto, Machine Learning for Predictive Maintenance: A Multiple Classifier Approach, 2015.
- [32] G. Soares, An on-line weighted ensemble of regressor models to handle concept drifts, 2015.
- [33] Baptista, Forecasting fault events for predictive maintenance using data-driven techniques and ARMA modeling, 2018.
- [34] H. Hashemian, State-of-the-art predictive maintenance techniques, 2011.
- [35] G. Biau, A random forest guided tour, 2016.
- [36] I. Amihai, An industrial case study using vibration data and machine learning to predict asset health, 2018.
- [37] J. Lee, A systematic approach for developing and deploying advanced prognostics technologies and tools: methodology and applications, 2007.
- [38] Z.Pan, Cognitive acoustic analytics service for internet of things, 2017.
- [39] T. Sexton, Hybrid datafication of maintenance logs from AI-assisted human tag, 2017.
- [40] T. Praveenkumar, Fault diagnosis of automobile gearbox based on machine learning techniques, 2014.
- [41] G. Hamerly, Partitional clustering algorithms, 2015.
- [42] K. Liulys, Machine Learning Application in Predictive Maintenance, 2019.
- [43] D. Maeda, Remaining Useful Life Prediction of Industrial Consumables Using Wideband Vibration Signals, 2019.
- [44] G. Zhang, Forecasting with artificial neural networks: the state of the art, 1998
- [45] M.M. Elkateb, A comparative study of medium-weather-dependent load forecasting using enhanced artificial fuzzy neural network and statistical techniques, 1998
- [46] C. Chateld, What is the ‘best’ method of forecasting, 1988
- [47] R. Clemen, Combining forecasts: a review and annotated bibliography with discussion, 1989
- [48] G. Zhang, Time series forecasting using a hybrid ARIMA and neural network model, 2003
- [49] Jan G. De Gooijer, 25 years of time series forecasting, 2006
- [50] V. Kotu, Data Science: Concepts and Practice, 2019
- [51] Pegels, Exponential smoothing: Some new variations, 1969

- [52] Newbold, On exponential smoothing and the assumption of deterministic trend plus white noise datagenerating models, 1989
- [53] G.E.P. Box, G. Jenkins, Time Series Analysis, Forecasting and Control, 1970
- [54] W. Wang, Autoregressive Model-Based Gear Fault Diagnosis, 2002
- [55] J. Smith, Forecasting costs incurred from unit differencing fractionally integrated processes, 1994
- [56] D. Pierce, Least squares estimation in the regression model with autoregressive-moving average errors, 1971
- [57] V. Godambe, An optimum property of regular maximum likelihood estimation, 1960
- [58] Rob J. Hyndman, Yeasmin Khandakar, Automatic Time Series Forecasting: The forecast Package for R, 2008
- [59] www.en.wikipedia.org/wiki/Akaike_information_criterion
- [60] www.otexts.com/fpp2/arima-r.html
- [61] www.en.wikipedia.org/wiki/Power_transform
- [62] www.en.wikipedia.org/wiki/Ljung%E2%80%93Box_test
- [63] Lee, Prognostics and health management design for rotary machinery systems – reviews, methodology and applications, 2014
- [64] E. Lughofer, Predictive Maintenance in Dynamic System, 2019
- [65] Tinga, T.: Principles of Loads and Failure Mechanisms: Applications in Maintenance, Reliability and Design, 2013
- [66] Z. Liang, Cutting Speed Sensitivity of Tool Life, 2002
- [67] J.V. Beck, Parameter estimation in engineering problems, 1985
- [68] V. Duarte, F. Arruda, Updating rotor-bearing finite element models using experimental frequency response functions, 1993
- [69] www.statisticssolutions.com/assumptions-of-multiple-linear-regression
- [70] N. H Cook, Tool Wear and Tool Life, 1993
- [71] Metal Cutting, University of St. Thomas, 2014
- [72] D. Stephenson, Metal Cutting Theory and Practice, 2016
- [73] www.tecnoutensilidecca.it
- [74] [www.en.wikipedia.org/wiki/Milling_\(machining\)](http://www.en.wikipedia.org/wiki/Milling_(machining))
- [75] Tool life testing with single-point turning tools, ISO Standard 3685:1993(E), 1993.
- [76] A. J. DeArdo, A better way to assess machinability, 1993
- [77] H. Chen , Wear of aluminum matrix composites reinforced with nickel-coated carbon fibres, 1996
- [78] B. M. Kramer, On tool materials for high speed machining, 1987
- [79] www.r-project.org/about.html
- [80] [www.it.wikipedia.org/wiki/R_\(software\)](http://www.it.wikipedia.org/wiki/R_(software))
- [81] www.rdocumentation.org/packages/R.matlab/versions/3.6.2
- [82] www.rdocumentation.org/packages/readxl/versions/1.3.1
- [83] www.rdocumentation.org/packages/forecast/versions/8.13
- [84] www.rdocumentation.org/packages/ggplot2/versions/3.3.2

- [85] www.rdocumentation.org/packages/dplyr/versions/0.7.8
- [86] www.rdocumentation.org/packages/neuralnet/versions/1.44.2
- [87] www.rdocumentation.org/packages/nnet/versions/7.3-14
- [88] www.rdocumentation.org/packages/rstanarm/versions/2.21.1
- [89] www.rdocumentation.org/packages/Metrics/versions/0.1.4
- [90] www.rdocumentation.org/packages/caret/versions/6.0-86
- [91] www.rdocumentation.org/packages/gridExtra/versions/2.3
- [92] www.rdocumentation.org/packages/plyr/versions/1.8.6
- [93] G. Sullivan, Operations & maintenance best practices: a guide to achieving operational efficiency, 2010
- [94] J. Jimenez, Towards multi-model approaches to predictive maintenance: A systematic literature survey on diagnostics and prognostics, 2020
- [95] L. Xu, Industry 4.0: state of the art and future trends, 2018
- [96] K. Liulys, Machine Learning Application in Predictive Maintenance, 2019
- [97] A. Kanawaday, Machine learning for predictive maintenance of industrial machines using IoT sensor data, 2017
- [98] A. Agogino and K. Goebel, Milling dataset (NASA-PCoE), 2007
- [99] www.ni.com
- [100] E. Traini, G. Bruno, G. D'Antonio, F. Lombardi, Machine Learning Framework for Predictive Maintenance in Milling, 2020
- [101] X. Chu, Data Cleaning: Overview and Emerging Challenges, 2016
- [102] <https://www.itl.nist.gov/div898/handbook/prc/section1/prc16.htm>
- [103] D. N. Lawley, A Generalization of Fisher's z Test, 1938
- [104] S Ahmad, On efficient monitoring of process dispersion using interquartile range, 2012
- [105] S. Liu, Change-point detection in time-series data by relative density-ratio estimation, 2013
- [106] T. Tinga, Application of physical failure models to enable usage and load based maintenance, 2010
- [107] A. Botchkarev, Performance Metrics (Error Measures) in Machine Learning Regression, Forecasting and Prognostics: Properties and Typology, 2018
- [108] R. Hyndman, J. Koehler, Another look at measures of forecast accuracy, 2016
- [109] M. Browne, Cross-Validation Methods, 2000
- [110] [www.en.wikipedia.org/wiki/Cross-validation_\(statistics\)](http://www.en.wikipedia.org/wiki/Cross-validation_(statistics))
- [111] T. Hastie, The Elements of Statistical Learning (Springer Series in Statistics), 2009
- [112] R.B. Darlington, Regression and linear models. 1990
- [113] www.rdocumentation.org/packages/randomForest/versions/4.6-14/topics
- [114] www.rdocumentation.org/packages/gbm/versions/2.1.8
- [115] W.M. Bolstad, Introduction to Bayesian Statistics, 2017
- [116] H. Jabbar, Methods to avoid over-fitting and under-fitting in supervised machine learning (comparative study), 2015

- [117] Leslie N. Smith, A Disciplined Approach To Neural Network Hyper-Parameters: Part 1, 2018
- [118] D. W. Bunn, Combining forecasts, 1988