

UNIVERSITÀ POLITECNICA DELLE MARCHE
FACOLTÀ DI INGEGNERIA
Dipartimento di Ingegneria dell'Informazione
Corso di Laurea Magistrale in Ingegneria Informatica e dell'Automazione



TESI DI LAUREA

**Diagnosi guasti in macchine centrifughe ad asse orizzontale tramite
segnali di vibrazione**

**Fault diagnosis in horizontal axis centrifugal machines using
vibration signals**

Relatore

Prof. Alessandro Freddi

Correlatore

Ing. Alessio Zimar

Candidato

Eris Prifti

ANNO ACCADEMICO 2022-2023

*“L’innovazione è lo strumento specifico dell’imprenditoria.
L’atto che favorisce il successo con una nuova capacità
di creare benessere.”*

Peter Ferdinando Drucker

Sommario

Negli ultimi anni l'Analisi Vibrazionale ha assunto un'importanza sempre maggiore, in quanto, attraverso le misure provenienti da una macchina rotante e alla loro successiva elaborazione è possibile prendere decisioni strategiche riguardo la gestione e la manutenzione della macchina. In particolare, le vibrazioni giocano un ruolo importantissimo nella manutenzione dei macchinari e soprattutto nella manutenzione predittiva. In questa tesi sono stati analizzati i dati relativi ai cuscinetti FAG utilizzati su macchine centrifughe dell'azienda Pieralisi. In particolare, è stata realizzata una fase di Extract, Transform and Load (ETL) dei dati, seguita da un'analisi descrittiva; successivamente è stata effettuata una fase di analisi del segnale e di classificazione, per valutare le differenti tipologie di guasto subite dai cuscinetti.

Keyword: Data Analytics, Analisi Vibrazionale, Trasformata di Fourier, Cuscinetti, Extract Transform and Load, Exploratory Data Analysis, Cluster, Classificazione

Introduzione	1
1 Introduzione alla manutenzione	3
1.1 Strategie di manutenzione	3
1.2 Manutenzione predittiva	5
1.3 Tecnologie predittive	7
2 Analisi vibrazionale	8
2.1 Introduzione all'analisi vibrazionale	8
2.2 Concetti di base dell'analisi delle vibrazioni	8
2.2.1 Vibrazione semplice	8
2.3 Vibrazione macchinari industriali	10
2.3.1 Vibrazione complessa	10
2.3.2 Altri tipi di vibrazione	11
2.4 Studio delle vibrazioni	12
2.4.1 Trasformata di Fourier	12
2.4.2 Spostamento, velocità e accelerazione	13
2.4.3 Analisi spettrale	15
3 Macchine rotanti e cuscinetti volventi	16
3.1 Macchine rotanti	16
3.2 Cuscinetti volventi	16
3.3 Tecniche di rilevamento dei guasti dei cuscinetti volventi	17
3.3.1 Tecniche basate sull'analisi di parametri semplificati	17
3.3.2 Tecniche basate sull'analisi delle forme d'onda spettrali e temporali	18
3.4 Componenti dei cuscinetti volventi e frequenze di guasto	18
3.4.1 Formule per il calcolo delle frequenze di cedimento dei cuscinetti	19
3.4.2 Influenza della variazione dell'angolo di contatto nelle frequenze di cedimento del cuscinetto	20
3.5 Previsione del livello di gravità del danno al cuscinetto	20
3.6 Difetti tipici del cuscinetto e identificazione spettrale	23
3.7 Fasi di deterioramento	25
3.8 Demodulazione o analisi dell'inviluppo	26

4	Strumenti e componenti utilizzati	29
4.1	Sistema di acquisizione dati	29
4.2	Macchina ad asse orizzontale e canotto	32
4.3	Cuscinetti a rullo FAG e a sfere SKF	33
4.3.1	Cuscinetto a rullo FAG	33
4.3.2	Cuscinetto a sfere SKF 6207	34
4.4	Accelerometri piezoelettrici PCB	35
4.5	Matlab	35
4.5.1	Diagnostic Feature Design	37
4.5.2	Classification Learner	37
4.6	Python	38
4.6.1	TensorFlow	38
4.6.2	Keras	39
5	Implementazione	40
5.1	Creazione dei fault e panoramica degli esperimenti	40
5.2	Creazione del dataset	44
5.3	Analisi in frequenza Matlab	47
5.3.1	Diagnostic Feature Designer e Classification Learner in Matlab	48
5.4	Classificazione in Python	56
5.4.1	TensorFlow e Keras	56
6	Risultati	59
6.1	Risultati analisi in frequenza	59
6.1.1	Analisi segnale Yc	59
6.1.2	Analisi segnale Za	65
6.2	Risultati Diagnostic Feature Designer e Classification Learner	70
6.2.1	Analisi segnale Yc	70
6.2.2	Analisi segnale Za	78
6.3	Risultati Python	86
6.3.1	Risultati segnale Yc	86
6.3.2	Risultati segnale Za	89
7	Conclusioni	92
	Bibliografia	95

Elenco delle figure

1.1	Schema manutenzione correttiva [Mathworks, 2019]	3
1.2	Schema manutenzione preventiva [Mathworks, 2019]	4
1.3	Schema manutenzione predittiva [Mathworks, 2019]	4
1.4	Schema manutenzione proattiva [Power-MI, 2022]	4
1.5	Costo totale di un guasto [Power-MI, 2022]	6
2.1	Somma nel dominio del tempo della vibrazione semplice [Power-MI, 2022]	10
2.2	Vibrazione casuale [Power-MI, 2022]	11
2.3	Battiti intermittenti [Power-MI, 2022]	12
2.4	Modulazione in ampiezza [Power-MI, 2022]	12
3.1	Componenti dei cuscinetti [Power-MI, 2022]	18
3.2	Composizione cuscinetti volventi [Power-MI, 2022]	19
3.3	Fault anello esterno [Power-MI, 2022]	24
3.4	Fault anello interno [Power-MI, 2022]	24
3.5	Fault nelle sfere o rulli [Power-MI, 2022]	24
3.6	Fault nella gabbia [Power-MI, 2022]	24
3.7	Fase 1 di deterioramento del cuscinetto [Power-MI, 2022]	26
3.8	Fase 2 di deterioramento del cuscinetto [Power-MI, 2022]	26
3.9	Fase 3 di deterioramento del cuscinetto [Power-MI, 2022]	26
3.10	Fase 4 di deterioramento del cuscinetto [Power-MI, 2022]	26
3.11	Esempio di forma d'onda di un cuscinetto difettoso [Power-MI, 2022]	27
4.1	Sistema di acquisizione nell'ambiente operativo	30
4.2	Schermata di configurazione accelerometri	31
4.3	Schermata AUTO CYCLE	32
4.4	Rotovariatore e canotto	33
4.5	Posizione cuscinetti FAG e SKF	33
4.6	Cuscinetto a rullo FAG N207	34
4.7	Cuscinetto a rullo FAG NU207	34
4.8	Cuscinetto SKF 6207	35
4.9	Accelerometro triassiale PCB	35
5.1	Cuscinetto FAG N207 senza guasti	41
5.2	Cuscinetto FAG NU207 senza guasti	41
5.3	Presenza del fault nell'anello esterno	42

5.4	Presenza del fault nell'anello interno	42
5.5	Presenza del fault nel rullo	43
5.6	Disegno posizionamento accelerometri piezoelettrici PCB	43
5.7	Posizionamento dei sensori nel banco da prova	44
5.8	Visualizzazione formato tdms in Excel	45
5.9	Dataset formato ".mat" utilizzato per Matlab	46
5.10	Dataset formato ".txt" utilizzato per Python	47
5.11	Interfaccia e selezione variabili per il Diagnostic Feature Designer	49
5.12	Frame policy	49
5.13	Principali caratteristiche esportate	50
5.14	Interfaccia Diagnostic Feature Designer dove effettuare diverse azioni	51
5.15	Interfaccia Diagnostic Feature Designer per il signal trace	51
5.16	Calcolo delle principali feature	51
5.17	Esempio di creazione nuova sessione Classification Learner	52
5.18	Menù per eseguire diverse azioni per la classificazione	53
5.19	Tipi di tecniche di classificazione	54
5.20	Esempio di scatter plot	54
5.21	Esempio di matrice di confusione	55
5.22	Esempio di curva ROC e AUC	56
6.1	Andamento temporale 16 esperimenti del segnale Yc divisi per tipo	61
6.2	Confronto 4 segnali del dataset	62
6.3	Trasformata di Fourier 16 esperimenti del segnale Yc divisi per tipo	63
6.4	Risposta in frequenza del segnale Yc senza fault	64
6.5	Risposta in frequenza del segnale Yc con fault presente nell'anello esterno	64
6.6	Risposta in frequenza del segnale Yc con fault presente nell'anello interno	65
6.7	Risposta in frequenza del segnale Yc con fault presente nella sfera o rullo	65
6.8	Andamento temporale 16 esperimenti del segnale Za divisi per tipo	66
6.9	Confronto andamento temporale segnali Za	67
6.10	Trasformata di Fourier 16 esperimenti del segnale Za divisi per tipo	68
6.11	Risposta in frequenza di 4 segnali campione di Za	69
6.12	Andamento temporale segnale Yc	70
6.13	Andamento temporale segnale Yc con TSA	70
6.14	Spettro del segnale Yc dopo l'applicazione del TSA	71
6.15	Istogrammi Feature Yc	71
6.16	Istogramma Feature Yc	72
6.17	Istogrammi Feature Yc con TSA	72
6.18	Istogramma Feature Yc con TSA	73
6.19	Calcolo ranking feature di Yc con e senza TSA	73
6.20	Risultato allenamento per la classificazione del segnale Yc con diversi algoritmi	74
6.21	Risultato dei test per la classificazione del segnale Yc con diversi algoritmi	74
6.22	Informazioni train e test del segnale Yc con il modello trilayered neural network	75
6.23	Matrici di confusione numero osservazioni train	75
6.24	Matrici di confusione valutazione true positive/false negative train	75
6.25	Matrici di confusione positive predictive values/false discovery rate train	76
6.26	Matrici di confusione numero osservazioni test	76
6.27	Matrici di confusione valutazione true positive/false negative test	76
6.28	Matrici di confusione positive predictive values/false discovery rate test	76
6.29	Curva ROC no fault del train del segnale Yc	77
6.30	Curva ROC fault esterno del train del segnale Yc	77
6.31	Curva ROC fault interno del train del segnale Yc	77

6.32	Curva ROC fault rullo del train del segnale Yc	77
6.33	Curva ROC no fault del test del segnale Yc	77
6.34	Curva ROC fault esterno del test del segnale Yc	77
6.35	Curva ROC fault interno del test del segnale Yc	78
6.36	Curva ROC fault rullo del test del segnale Yc	78
6.37	Andamento temporale segnale Za	78
6.38	Andamento temporale segnale Za con TSA	79
6.39	Spettro del segnale Za dopo l'applicazione del TSA	79
6.40	Istogrammi Feature Za	80
6.41	Istogramma Feature Za	80
6.42	Istogrammi Feature Za con TSA	81
6.43	Istogramma Feature Za con TSA	81
6.44	Calcolo ranking feature di Za con e senza TSA	82
6.45	Risultato allenamento della classificazione con diversi algoritmi del segnale Za	82
6.46	Risultato dei test della classificazione con diversi algoritmi del segnale Za	83
6.47	Informazioni allenamento e test del modello quadratic SVM del segnale Za	83
6.48	Matrici di confusione numero osservazioni train	84
6.49	Matrici di confusione valutazione true positive/false negative train	84
6.50	Matrici di confusione positive predictive values/false discovery rate train	84
6.51	Matrici di confusione numero osservazioni test	84
6.52	Matrici di confusione valutazione true positive/false negative test	85
6.53	Matrici di confusione positive predictive values/false discovery rate test	85
6.54	Curva ROC no fault del train del segnale Za	85
6.55	Curva ROC fault esterno del train del segnale Za	85
6.56	Curva ROC fault interno del train del segnale Za	85
6.57	Curva ROC fault rullo del train del segnale Za	85
6.58	Curva ROC no fault del test del segnale Za	86
6.59	Curva ROC fault esterno del test del segnale Za	86
6.60	Curva ROC fault interno del test del segnale Za	86
6.61	Curva ROC fault rullo del test del segnale Za	86
6.62	Andamento temporale del segnale Yc con i suoi 16 esperimenti	87
6.63	Scatter plot del segnale Yc con le sue 16 prove	87
6.64	Andamento del segnale Yc per tipologia di prova	87
6.65	Valori accuratezza delle epoche	88
6.66	Matrice di confusione	89
6.67	Andamento temporale del segnale Za con i suoi 16 esperimenti	89
6.68	Scatter plot del segnale Za con le sue 16 prove	90
6.69	Andamento del segnale Za per tipologia di prova	90
6.70	Valori accuratezza delle epoche segnale Za	90
6.71	Matrice di confusione Za	91

Elenco delle tabelle

3.1	Bande spettrali per il rilevamento di problemi ai cuscinetti volventi	25
4.1	Tabella parametri cuscinetti	34
5.1	Descrizione delle prove eseguite	41

Negli ultimi decenni, i dati sono diventati sempre più importanti per le aziende e rappresentano un vantaggio competitivo e significativo se utilizzati correttamente. Per sfruttare appieno questo vantaggio, i dati devono essere analizzati e studiati approfonditamente. L'analisi dei dati è un processo che comprende l'ispezione, la pulizia, la trasformazione e la modellazione dei dati al fine di identificare informazioni utili che supportano le decisioni strategiche aziendali. Se le decisioni aziendali vengono prese senza l'utilizzo delle conoscenze derivanti dai dati, non possiamo parlare di vere decisioni strategiche, ma piuttosto di semplici ipotesi. Integrare l'analisi dei dati nel processo decisionale consente di basare le scelte su informazioni concrete, aumentando la probabilità di successo e consentendo all'azienda di adattarsi in modo più tempestivo e efficace al contesto competitivo in cui opera.

Seguendo quanto proposto da [Paolo Atzeni, 2023], attraverso una fase accurata di analisi dei dati, è possibile estrarre informazioni preziose. L'analisi dei dati consente di identificare modelli, tendenze e correlazioni che potrebbero essere altrimenti sfuggiti all'attenzione. Queste informazioni possono essere utilizzate per prendere decisioni strategiche più informate, con l'obiettivo di intervenire, quando si presenta un problema in un macchinario, in modo mirato per riuscire infine ad ottimizzare i processi aziendali.

Per sfruttare appieno il potenziale dei dati, è importante adottare approcci analitici adeguati, utilizzare strumenti e tecnologie avanzate, e avere a disposizione competenze specializzate nell'analisi dei dati. Investire nella capacità di analizzare i dati può aprire nuove prospettive per le aziende e consentire loro di cogliere le opportunità nascoste nei dati stessi, conducendo al successo nel mercato competitivo di oggi.

Attraverso l'analisi vibrazionale, i dati acquisiti tramite accelerometri o altri strumenti di misura dalle macchine industriali sono di fondamentale importanza per la manutenzione di esse, e per mezzo della loro lavorazione è possibile sviluppare algoritmi per la manutenzione predittiva.

L'analisi delle vibrazioni, come citato da [Antonio Gulli, 2019], rappresenta uno strumento potente per gestire i dati relativi alle vibrazioni meccaniche che possono interessare macchinari o componenti in generale. La misurazione delle vibrazioni offre numerosi vantaggi ai progettisti e agli ingegneri che si occupano del monitoraggio dello stato di salute di strutture e componenti.

Le moderne apparecchiature elettroniche sono spesso dotate di sensori in grado di rilevare urti e vibrazioni eccessive. Una volta che i dati vengono acquisiti dai sensori, è necessario elaborarli per rilevare eventuali problemi. Questo processo di elaborazione dei dati consente di identificare anomalie nelle vibrazioni, come frequenze insolite o ampiezze sproporzionate, che possono indicare guasti imminenti o problemi di funzionamento.

L'analisi delle vibrazioni consente quindi di individuare precocemente potenziali difetti, migliorando la sicurezza e l'affidabilità dei macchinari. Inoltre, fornisce informazioni utili per programmare la manutenzione in modo predittivo, consentendo interventi preventivi mirati e riducendo al minimo i tempi di fermo imprevisti. Ciò si traduce in una maggiore efficienza operativa, una prolungata durata delle attrezzature e una riduzione dei costi di manutenzione complessivi. In questo modo si sta cercando di muoversi in una direzione che riguarda l'industria 4.0 dove attraverso tecniche di diagnosi e di prognosi, viste da [G. Vachtsevanos, 2006], si vuole allungare il tempo di vita delle entità e intervenire tempestivamente appena un problema si presenta.

Complessivamente, l'analisi delle vibrazioni è uno strumento prezioso per il monitoraggio dello stato di salute dei macchinari e dei componenti, consentendo agli operatori di prendere decisioni informate e tempestive per garantire il funzionamento ottimale e affidabile delle attrezzature industriali. Inoltre i sensori intelligenti che sono stati adoperati per l'analisi delle vibrazioni sono una componente essenziale nell'industria 4.0.

In questo elaborato sono stati analizzati i dati degli accelerometri ottenuti da un sistema di acquisizioni dati riguardanti le vibrazioni derivanti dai cuscinetti della FAG installati su una macchina centrifuga per l'attività olearia, operante sia in modalità nominale sia quando essa è affetta da guasti sui cuscinetti. In una prima fase viene effettuata l'acquisizione dati, successivamente viene fatta una elaborazione dei dati per estrapolare l'informazione utile. L'obiettivo finale è quello di capire quando è presente un guasto in modo da poter conoscere in anticipo un potenziale pericolo, tramite un'analisi del segnale nei domini del tempo e della frequenza, seguita da una fase di classificazione utilizzata per distinguere le varie tipologie di fault che avvengono sul cuscinetto.

La presente tesi è composta da sette capitoli strutturati come di seguito specificato:

- Nel Capitolo 1 sarà introdotto il mondo della manutenzione.
- Nel Capitolo 2 verrà introdotta l'analisi vibrazionale e le sue varie caratteristiche.
- Nel Capitolo 3 verranno introdotti i cuscinetti volventi.
- Nel Capitolo 4 saranno introdotte gli strumenti utilizzati con una spiegazione esaustiva.
- Nel Capitolo 5 verrà analizzata l'implementazione di tutto il progetto che comprende al suo interno diverse fasi come la creazione del dataset, la creazione dei fault e così via.
- Nel Capitolo 6 saranno riportati i risultati ottenuti.
- Nel Capitolo 7 verranno tratte le conclusioni e verranno delineati alcuni possibili sviluppi futuri.

Introduzione alla manutenzione

Il campo della manutenzione nell'industria ha subito un notevole progresso grazie agli avanzi tecnologici nel controllo e nella misurazione. Questo capitolo fornisce una panoramica esaustiva sull'evoluzione di tale settore, con particolare focus sulla manutenzione predittiva delle macchine (CBM) e sui criteri essenziali per la sua implementazione. Vengono inoltre analizzati i vantaggi tecnici ed economici offerti dalla CBM, che vengono confrontati con i tradizionali metodi di manutenzione. Infine, il capitolo si dedica all'individuazione e alla valutazione delle diverse tecnologie di manutenzione predittiva disponibili.

1.1 Strategie di manutenzione

La manutenzione all'interno del settore industriale ha subito un'evoluzione significativa, principalmente grazie ai progressi nella tecnologia di controllo e misurazione. Questo capitolo fornisce un riepilogo di tale evoluzione, focalizzandosi su quattro fasi chiave della manutenzione e fornendo una descrizione dettagliata su ciascuna di esse.

In passato, la manutenzione veniva eseguita solamente in seguito a guasti, generando costi di riparazione, tempi di inattività e altre spese correlate; Questo approccio è noto come **manutenzione correttiva**.

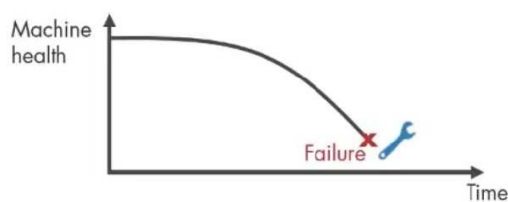


Figura 1.1: Schema manutenzione correttiva [Mathworks, 2019]

Per affrontare i costi derivanti dalla scarsa disponibilità delle macchine e i tempi di fermo della produzione, i responsabili della manutenzione hanno iniziato ad implementare programmi di ispezioni periodiche al fine di mantenere le macchine nelle migliori condizioni possibili e ridurre la probabilità di guasti. Tuttavia, questa strategia, conosciuta come **manutenzione preventiva**, presenta incertezze in termini di costo e solleva interrogativi riguardo all'adeguatezza degli intervalli di manutenzione stabiliti e la possibilità di ridurli senza conseguenze negative per le macchine. Queste incertezze limitano l'efficacia della manutenzione preventiva.

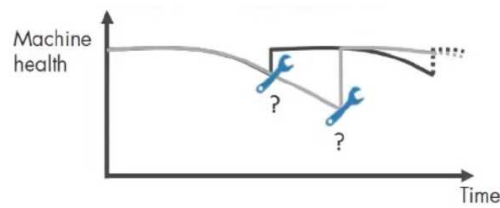


Figura 1.2: Schema manutenzione preventiva [Mathworks, 2019]

Grazie allo sviluppo tecnologico, è emerso un nuovo concetto di manutenzione basato sul monitoraggio delle condizioni o dello stato delle macchine. Questo approccio è noto come **manutenzione predittiva**. Si basa sulla capacità di anticipare i guasti attraverso la conoscenza del comportamento normale e previsto delle macchine, consentendo di programmare interventi senza interrompere il processo produttivo. Ciò ottimizza i costi di produzione, manodopera e ricambi. La manutenzione predittiva consente di evitare guasti importanti che possono essere costosi e si traduce in tempi di intervento più rapidi e tempestivi.

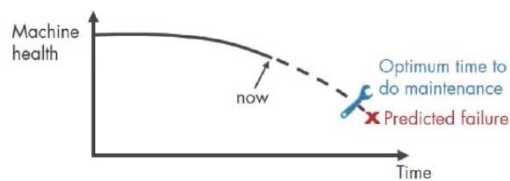


Figura 1.3: Schema manutenzione predittiva [Mathworks, 2019]

La **manutenzione proattiva** è stata sviluppata come un complemento all'evoluzione della manutenzione predittiva. Mentre la manutenzione predittiva può determinare la probabilità di guasto di un componente della macchina senza indagare sulla causa principale del guasto stesso. La manutenzione proattiva affronta questa incertezza analizzando la causa principale dei guasti ripetuti e risolvendo gli aspetti tecnici correlati, come l'utilizzo di componenti della macchina inadeguati. In questo modo, la manutenzione proattiva mira a prevenire i guasti futuri affrontando direttamente le cause radice del problema, aumentando la durata e l'affidabilità complessiva della macchina.

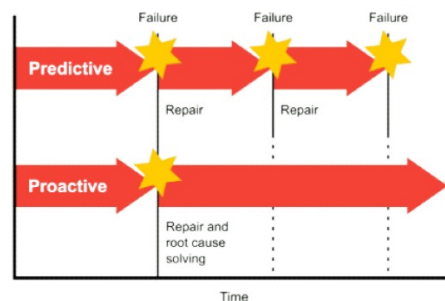


Figura 1.4: Schema manutenzione proattiva [Power-MI, 2022]

1.2 Manutenzione predittiva

Come introdotto da [Luciano Furlanetto, 2022] e [Power-MI, 2022] nella manutenzione predittiva, l'obiettivo è valutare lo stato dei componenti meccanici o elettrici attraverso il monitoraggio e l'analisi di vari parametri: come le vibrazioni, la temperatura, i contaminanti dell'olio e l'isolamento. Questo approccio, inoltre consente di programmare le operazioni di manutenzione solo quando necessario, riducendo i costi operativi e di manutenzione e allo stesso tempo aumentando l'affidabilità delle apparecchiature. La base della manutenzione predittiva è il monitoraggio continuo delle apparecchiature in funzione. Ciò consente di valutare il loro stato senza doverle fermare, fornendo informazioni sul funzionamento delle macchine e consentendo di rilevare problemi legati ai componenti, alla progettazione o all'installazione. L'implementazione di un programma di manutenzione predittiva richiede una strategia chiara e l'implementazione graduale su risorse selezionate. In genere, si inizia con l'applicazione di tecniche di analisi delle vibrazioni su un gruppo selezionato di macchine. Attraverso un programma pilota, si identificano i difetti, si analizzano le cause alla radice e si forniscono feedback una volta che i difetti sono stati eliminati. I risultati delle analisi vengono registrati in un sistema di gestione delle informazioni, consentendo di alimentare il processo decisionale con informazioni chiare e prioritarie.

L'utilizzo di soluzioni basate su cloud per la collaborazione e la consegna dei dati, come Power-MI, semplifica la creazione di report e facilita l'accesso alle informazioni necessarie per il processo decisionale. L'integrazione dei dati di monitoraggio delle condizioni in un repository comune migliora la capacità diagnostica e rompe le tradizionali barriere organizzative tra i diversi gruppi di competenze.

Una volta ottenuti risultati positivi dal programma pilota, si procede all'espansione graduale del programma ad altre risorse, integrando diverse tecnologie e gestendo efficacemente i dati raccolti. L'obiettivo finale è un'implementazione completa della manutenzione predittiva che fornisca benefici significativi in termini di efficienza operativa, durata delle apparecchiature e riduzione dei costi di manutenzione.

La giustificazione tecnica della manutenzione predittiva è abbastanza ovvia, perché consente di confrontare le azioni intraprese durante la manutenzione preventiva o correttiva con quelle che sarebbero necessarie con il monitoraggio continuo e la sorveglianza dei parametri misurati.

Quando ci limitiamo alla sola manutenzione correttiva, il termine "manutenzione" diventa sinonimo di "riparazione". Tuttavia, ciò non significa che questo tipo di intervento debba essere evitato. Infatti, l'obiettivo finale di ogni intervento di manutenzione è riparare o sostituire componenti danneggiati per prolungare la vita della macchina. Gli interventi correttivi sono inevitabili. Tuttavia, concentrare tutte le azioni di manutenzione solo su questo aspetto è un errore. Questo approccio offre solo un vantaggio, un basso costo di investimento iniziale, ma comporta notevoli svantaggi.

- Gli arresti imprevisti di solito si verificano quando la produzione deve essere maggiore, cioè nel momento peggiore.
- Questi arresti sono spesso catastrofici, in quanto spesso portano a danni significativi. A volte per la perdita di un cuscinetto, si possono danneggiare seriamente le trasmissioni.
- I costi per ripararli diventano elevati.
- La necessità di avere pezzi di ricambio in magazzino aumenta con l'incertezza di ciò che può fallire.
- Il rischio di incidenti aumenta.

- Si è totalmente all'oscuro delle reali condizioni delle macchine.
- Si accettano fallimenti senza indagare la causa principale; ad esempio, se un cuscinetto si deteriora prematuramente e lo sostituiamo semplicemente, potremmo trascurare un problema di disallineamento o squilibrio che potrebbe causare il danno prematuro.

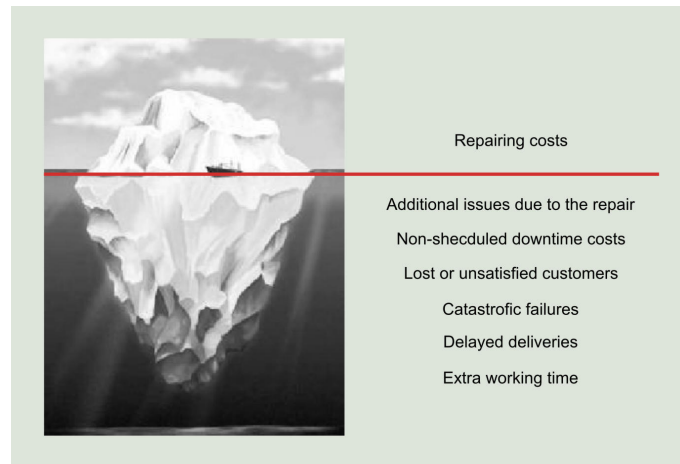


Figura 1.5: Costo totale di un guasto [Power-MI, 2022]

Un altro approccio alla manutenzione è quello preventivo, detto anche programmato, il quale si basa sulla sostituzione dei componenti in base alla presunta fine della loro vita utile. La durata prevista di ciascun componente è solitamente fornita dal produttore dell'apparecchiatura originale, che fornisce anche linee guida per la manutenzione preventiva, compresi gli intervalli consigliati per la sostituzione e la lubrificazione dei componenti. Tuttavia, la manutenzione preventiva ha anche i suoi svantaggi:

- Il problema principale è intervenire su una macchina che funziona in modo corretto solo perché è stato programmato di intervenire. Le macchine in funzione stabiliscono un equilibrio operativo che può essere difficile da ripristinare una volta che vengono apportate modifiche preventive.
- Quando si interviene sulla macchina, si cambiano componenti che in genere sono in buone condizioni, ma si conosce lo stato effettivo solo dopo lo smontaggio.
- E' possibile cambiare componenti che hanno esaurito la loro vita teorica e non essere consapevoli di altri in condizioni di deterioramento. Non si ha alcun controllo sui danni nascosti.
- Per quanto riguarda il costo, può crescere rapidamente a causa dei frequenti interventi sulle macchine, molti dei quali non necessari.

La manutenzione predittiva può comportare sfide iniziali come costi di investimento più elevati per la tecnologia e la formazione, ma alla fine produce un ritorno sull'investimento a medio e lungo termine. I vantaggi sono evidenti:

- Si conoscono sempre le condizioni della macchina.
- Si eliminano la grande maggioranza dei difetti.
- Si ferma la macchina o si interviene su di essa solo quando è veramente necessario.

- Si conosce la fase di danneggiamento dei componenti partendo da una fase iniziale, programmando in questo modo la loro sostituzione nel momento più opportuno.
- Quando si interviene sulla macchina, si conosce già il problema, riducendo i tempi di riparazione.
- E' possibile identificare guasti nascosti e la causa di essi.
- Si riduce il numero di ricambi dal magazzino, acquistandoli anticipatamente quando si rileva il problema in una fase iniziale.
- Vengono ricevuti bonus sui premi assicurativi.
- Si aumenta la sicurezza dell'impianto.

1.3 Tecnologie predittive

Un aspetto chiave di qualsiasi programma di manutenzione predittiva efficace è la capacità di utilizzare più tecnologie per coprire i guasti più comuni o verificare un guasto utilizzando più tecnologie. Ciò non solo conferma la diagnosi, ma consente anche raccomandazioni correttive più accurate e precise. Diverse tecnologie possono essere utilizzate nella manutenzione basata sulle condizioni, ma le più comunemente utilizzate secondo [Power-MI, 2022] sono:

- Analisi delle vibrazioni
- Termografia
- Ispezione visuale
- Analisi dell'olio
- Ultrasuoni aerei
- Allineamento dell'albero
- Analisi della corrente motoria

Tra tutte queste l'analisi delle vibrazioni verrà approfondito nel Capitolo 2.

L'analisi delle vibrazioni è la più diffusa tra le diverse tecnologie applicabili alla manutenzione predittiva. La ragione di ciò è la sua versatilità nel rilevare molti difetti in una vasta gamma di macchine, insieme al suo ragionevole investimento economico iniziale.

2.1 Introduzione all'analisi vibrazionale

Come descritto da [Power-MI, 2022] e [Augusto Di Benedetto, 2007] il monitoraggio delle vibrazioni è un metodo fondamentale per valutare lo stato di salute e il comportamento delle macchine rotanti industriali, come motori, compressori e turbine. La vibrazione caratteristica di ogni macchina è unica e dipende dalla progettazione, fabbricazione, uso e usura dei suoi componenti. Un meccanico o ingegnere di manutenzione responsabile di un asset industriale può ottenere significativi risparmi sui costi operativi e di manutenzione, dedicando tempo e impegno per comprendere la natura dell'impronta di vibrazione. Attraverso il monitoraggio costante, è possibile rilevare anomalie o cambiamenti nel comportamento della macchina, consentendo interventi tempestivi.

2.2 Concetti di base dell'analisi delle vibrazioni

La vibrazione è un movimento di oscillazione di piccola ampiezza. Ogni oggetto emette un segnale di vibrazione che riflette le sue caratteristiche. Nelle macchine, questo segnale di vibrazione è unico e contiene informazioni da ciascun componente della macchina. In altre parole, il segnale vibrazionale di una macchina è la somma delle vibrazioni di tutte le sue parti. Questo fenomeno consente ai meccanici o ingegneri di manutenzione, di identificare e comprendere la salute della macchina, individuando eventuali problemi o anomalie che possono richiedere interventi preventivi.

2.2.1 Vibrazione semplice

La vibrazione semplice è un movimento oscillatorio periodico che coinvolge un oggetto o un sistema meccanico che si muove avanti e indietro attorno a una posizione di equilibrio. Questo tipo di vibrazione è caratterizzata da un singolo grado di libertà, il che significa che il movimento può avvenire lungo una sola direzione.

Un esempio di vibrazione semplice è quello di una molla collegata a un oggetto. Se si spinge l'oggetto in una certa direzione dalla sua posizione di equilibrio e poi si rilascia, l'oggetto inizierà a oscillare avanti e indietro attorno alla posizione di equilibrio fino a quando l'energia meccanica viene dissipata. Questo tipo di vibrazione segue una legge matematica nota come moto armonico semplice, che è caratterizzato da una frequenza costante e un'ampiezza costante nel tempo.

Le vibrazioni semplici sono importanti in molti contesti, sia nella scienza che nella tecnologia. I tre parametri che definiscono questo tipo di vibrazione sono frequenza, ampiezza e fase, le quali verranno descritte di seguito.

Ampiezza

L'ampiezza delle vibrazioni si riferisce alla massima escursione o all'estensione di un movimento oscillatorio da una posizione di equilibrio o punto centrale. In altre parole, è la distanza massima percorsa dall'oggetto o dal sistema durante il suo movimento di andata e ritorno.

Per capire meglio, si deve prendere come esempio un'onda sinusoidale, che è una rappresentazione grafica comune delle vibrazioni. L'ampiezza di un'onda sinusoidale corrisponde all'altezza massima della cresta dell'onda rispetto alla sua posizione di equilibrio (valore zero) o, nel caso di un'onda con ampiezza negativa, la profondità massima del cavo rispetto alla posizione di equilibrio.

I diversi modi per misurare l'ampiezza delle vibrazioni sono:

- **Picco-picco (peak-to-peak):** indica l'escursione massima dell'onda (sollecitazione massima, gioco meccanico).
- **Picco (peak):** indica l'escursione massima positiva o negativa della vibrazione (urti di breve durata).
- **RMS (Root Mean Square = valore medio efficace):** è la misura più significativa di ampiezza in quanto tiene conto della storia dell'onda nel tempo e dà un valore di ampiezza direttamente correlata al contenuto energetico della vibrazione:

$$RMS = \sqrt{\frac{1}{T} \int_0^T x^2(t) dt} \quad (2.2.1)$$

dove con $x(t)$ si è indicato l'andamento del segnale in funzione del tempo e T è la durata del segnale stesso. Nel caso di una grandezza sinusoidale di ampiezza A il suo valore efficace RMS vale:

$$RMS = \frac{A}{\sqrt{2}} \approx 0.707A \quad (2.2.2)$$

Nel caso di segnali digitali costituiti da N campioni il valore RMS si valuta mediante la seguente espressione:

$$RMS = \sqrt{\frac{1}{N} \sum_{i=1}^N x_i^2} \quad (2.2.3)$$

- **Valor medio (Average):** tiene conto della storia dell'onda nel tempo ma ha scarso interesse pratico non essendo correlato ad alcuna grandezza fisica.

Per i segnali continui:

$$Avg = \frac{1}{T} \int_0^T x(t) dt \quad (2.2.4)$$

Per segnali discreti:

$$Avg = \frac{1}{N} \sum_{i=1}^N x_i \quad (2.2.5)$$

Frequenza

La frequenza delle vibrazioni si riferisce al numero di cicli completi di oscillazione che un oggetto o un sistema compie in un determinato periodo di tempo. È una misura dell'andamento ripetitivo del movimento oscillatorio e viene espressa in Hertz (Hz), che corrisponde a un ciclo al secondo.

La frequenza delle vibrazioni è direttamente correlata alla velocità con cui avviene l'oscillazione. Vibrazioni ad alta frequenza si verificano quando il ciclo di oscillazione viene completato rapidamente, mentre vibrazioni a bassa frequenza richiedono più tempo per completare un ciclo.

Fase

La fase delle onde sinusoidali è una misura della differenza temporale tra due di esse, espressa in termini di angolo, come gradi o radianti. Questa normalizzazione del tempo utilizza come riferimento un ciclo d'onda completo senza considerare il suo vero periodo di tempo. La differenza di fase tra due forme d'onda è chiamata "offset di fase" o "spostamento di fase". Uno sfasamento di 360° gradi rappresenta un ritardo di un ciclo o un intero periodo d'onda, che in realtà è solo uno sfasamento. In modo simile, uno sfasamento di 90° indica uno spostamento di $\frac{1}{4}$ del periodo dell'onda, e così via. La fase può essere considerata sia positiva che negativa, indicando se una forma d'onda è in ritardo o in anticipo rispetto a un'altra. Questi fenomeni sono rispettivamente chiamati "ritardo di fase" e "anticipo di fase".

2.3 Vibrazione macchinari industriali

2.3.1 Vibrazione complessa

Una vibrazione complessa è il risultato della combinazione di diverse vibrazioni semplici, con frequenze e ampiezze diverse, ognuna delle quali è associata ai componenti interni in movimento di una macchina. La forma d'onda di vibrazione di una macchina non è un segnale sinusoidale semplice, ma può diventare molto complessa nel tempo. Nella Figura 2.1 è possibile osservare l'effetto della somma di due segnali di vibrazione con frequenze diverse, creando una vibrazione complessa. Anche in situazioni relativamente semplici come questa, può essere difficile estrarre le frequenze e le ampiezze delle due componenti dal risultante segnale complesso. Tuttavia, nella realtà, la stragrande maggioranza dei segnali di vibrazione è molto più complessa, rendendo l'interpretazione e l'analisi di tali forme d'onda estremamente difficili.

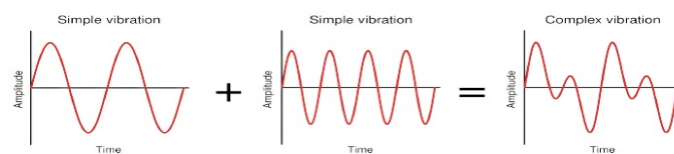


Figura 2.1: Somma nel dominio del tempo della vibrazione semplice [Power-MI, 2022]

La situazione più comune nelle macchine rotanti è che l'ottenimento di informazioni dettagliate sul funzionamento interno della macchina attraverso l'analisi diretta della forma d'onda delle vibrazioni risulta spesso difficile e limitato. Le forme d'onda delle vibrazioni sono tipicamente complesse e contengono una varietà di componenti, rendendo complicata l'identificazione delle specifiche sorgenti di vibrazione.

Nonostante ciò, ci sono casi in cui l'analisi della forma d'onda delle vibrazioni può risultare molto utile e informativa. Ad esempio, in situazioni dove si conoscono le frequenze di risonanza della macchina o di componenti specifici, è possibile rilevare l'amplificazione delle vibrazioni a queste frequenze, segnalando potenziali problemi o la necessità di interventi correttivi. Inoltre, l'analisi della forma d'onda può essere utilizzata per identificare eventuali cambiamenti nel comportamento della macchina nel tempo, suggerendo l'insorgenza di usura o guasti.

Tuttavia, per ottenere una comprensione approfondita del funzionamento interno della macchina e diagnosticare con precisione eventuali problematiche, è spesso necessario integrare l'analisi della forma d'onda con altre tecniche di monitoraggio delle vibrazioni, come l'analisi spettrale, l'analisi modale e l'uso di sensori specifici.

2.3.2 Altri tipi di vibrazione

Oltre alla semplice vibrazione, esistono anche altri tipi di vibrazione come: la vibrazione casuale, battiti intermittenti e la modulazione.

Le **vibrazioni casuali** non seguono schemi speciali che si ripetono costantemente, rendendo difficile individuare dove inizia e finisce un ciclo. Questo tipo di vibrazione è comunemente associato a fenomeni di turbolenza in soffianti e pompe, problemi di lubrificazione e contatto metallo-metallo negli elementi rotanti o cavitazione nelle pompe (vedi Figura 2.2). La natura caotica di queste vibrazioni le rende generalmente più adatte ad essere analizzate nello spettro piuttosto che nella forma d'onda. L'analisi spettrale consente di estrarre informazioni rilevanti sulle diverse componenti di frequenza presenti nelle vibrazioni casuali, facilitando la comprensione delle cause sottostanti e agevolando la diagnosi e la manutenzione delle apparecchiature industriali.

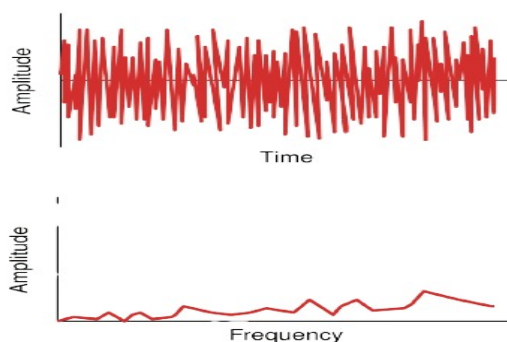


Figura 2.2: Vibrazione casuale [Power-MI, 2022]

I battiti intermittenti sono causati dalla presenza di battiti continui che generano un segnale ripetitivo. Questi fenomeni sono più comunemente riscontrati negli ingranaggi, nei cuscinetti, nel passaggio della girante o delle pale del ventilatore e così via. Il segnale generato da questi battiti continui tende a diminuire nel tempo a causa dello smorzamento presente nel sistema. Questo processo di attenuazione è chiaramente illustrato nella Figura 2.3.

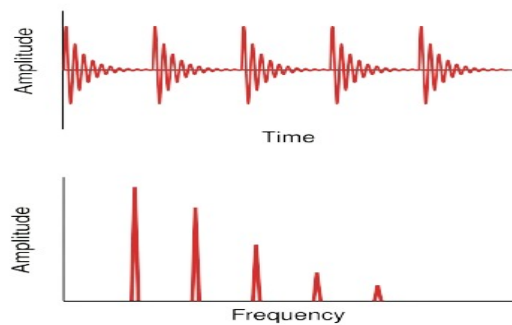


Figura 2.3: Battiti intermittenti [Power-MI, 2022]

La modulazione di ampiezza (AM) è un metodo di modulazione che coinvolge la variazione dell'ampiezza di un segnale portante a causa dell'influenza di un altro segnale, tipicamente di frequenza inferiore. Nello spettro mostrato nella Figura 2.4, la componente più significativa è la portante, mentre le altre componenti, che sembrano simili ad armoniche, sono note come bande laterali. Queste ultime si trovano simmetricamente su entrambi i lati della portante, e la loro distanza è uguale alla frequenza di modulazione.

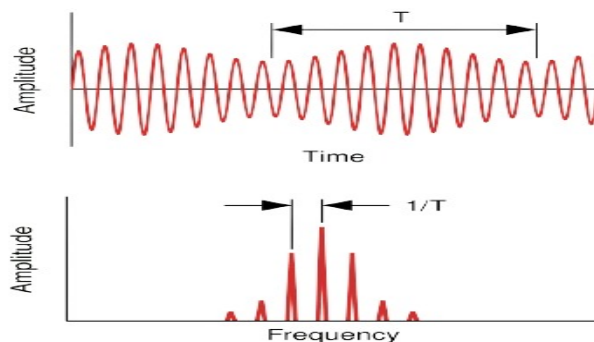


Figura 2.4: Modulazione in ampiezza [Power-MI, 2022]

2.4 Studio delle vibrazioni

2.4.1 Trasformata di Fourier

L'analisi delle vibrazioni nel dominio della frequenza è un approccio estremamente utile per comprendere il comportamento delle macchine e individuare possibili guasti. Questo metodo si basa sulla trasformata di Fourier, una tecnica sviluppata dal matematico francese Jean Baptiste Fourier (1768 - 1830).

La trasformata di Fourier è un concetto matematico fondamentale nell'analisi delle onde e dei segnali. Essa trasforma una funzione dal dominio del tempo a quello delle frequenze, consentendo una rappresentazione più dettagliata delle sue componenti, in modo da poter rilevare le frequenze fondamentali e le armoniche presenti nel segnale originale.

Analizzando un segnale di vibrazione mediante la trasformata di Fourier, è possibile ottenere informazioni dettagliate sulle frequenze caratteristiche di ogni componente della macchina. Questo ci consente di visualizzare i dati sotto forma di un grafico ampiezza-frequenza, noto come "spettro", il quale è uno strumento fondamentale per l'analisi delle macchine e la diagnosi dei problemi. Grazie a questo approccio, è possibile identificare pattern

specifici e anomalie nelle vibrazioni, rendendo più facile distinguere il comportamento caratteristico dei vari componenti della macchina e individuare eventuali malfunzionamenti.

Come introdotto da [Isermann, 2006] la trasformata di Fourier può essere applicata anche a segnali a tempo discreto, a patto che questo segnale possa essere e venga interpretato come il segnale ottenuto dal campionamento di un segnale a tempo continuo di cui esista la trasformata di Fourier.

La DFT (Discrete Fourier Transform) si applica a segnali a tempo discreto che possono essere approssimati da un numero finito di campioni, così che anche la corrispondente trasformata di Fourier a tempo discreto, con frequenza normalizzata nel continuo $[0, 2\pi]$, possa essere campionata in frequenza, cioè essere rappresentata da un numero finito di frequenze piuttosto che nell'intervallo continuo $[0, 2\pi]$.

La DFT è utilizzata per analizzare segnali discreti, come ad esempio segnali digitali campionati da un sistema di acquisizione di dati. Prende in input una sequenza finita di dati e restituisce una sequenza di coefficienti che rappresentano le componenti sinusoidali presenti nel segnale originale. Il risultato della DFT è un vettore di coefficienti complessi, in cui ogni coefficiente rappresenta la componente sinusoidale corrispondente a una specifica frequenza.

$$Y(n) = \sum_{h=0}^{N-1} y(h)e^{-j\frac{2\pi}{N}nh} \quad 0 \leq n \leq N-1 \quad (2.4.1)$$

Tuttavia, la DFT tradizionale può essere computazionalmente onerosa, soprattutto quando si tratta di analizzare segnali di grandi dimensioni. Per ovviare a questo problema, è stata sviluppata una variante della DFT chiamata trasformata di Fourier veloce (FFT - Fast Fourier Transform).

La FFT è un algoritmo efficiente per calcolare la DFT di una sequenza di dati. È molto più rapida rispetto alla DFT tradizionale, rendendo possibile l'analisi di segnali di grandi dimensioni in tempi ragionevoli. La FFT è basata sull'idea che la DFT può essere scomposta in sotto-trasformate più piccole, riducendo notevolmente il numero di calcoli necessari. Questo approccio divide ricorsivamente la sequenza di dati in blocchi di dimensioni minori, calcola le sotto-trasformate e combina i risultati per ottenere la DFT completa.

La FFT è ampiamente utilizzata in diverse applicazioni, come: elaborazione del segnale, analisi spettrale, elaborazione delle immagini, compressione dati e altro ancora. Consente di ottenere rapidamente informazioni sulle frequenze presenti in un segnale, facilitando l'identificazione di comportamenti periodici e la risoluzione di problemi di analisi diagnostica in diverse discipline, inclusa la vibrazione meccanica e l'ingegneria delle macchine.

2.4.2 Spostamento, velocità e accelerazione

Nell'ambito delle analisi vibrazionali, velocità, spostamento e accelerazione sono tre grandezze fondamentali per descrivere il comportamento dinamico di un sistema soggetto a vibrazioni.

- Lo **spostamento** è la distanza percorsa da un punto all'interno del sistema rispetto alla sua posizione di equilibrio durante le vibrazioni. Indica il movimento oscillatorio del punto rispetto alla sua posizione di riposo. Lo spostamento è misurato in unità di lunghezza, ad esempio millimetri (mm) o pollici (in).
- La **velocità** indica quanto velocemente il punto si sposta in una determinata direzione durante un periodo di tempo specifico. La velocità di spostamento è misurata in unità di lunghezza per unità di tempo, ad esempio millimetri al secondo (mm/s) o pollici al secondo (in/s).

- **L'accelerazione** è la variazione della velocità di un punto all'interno del sistema rispetto al tempo. Rappresenta la rapidità con cui la velocità di spostamento cambia durante le vibrazioni. L'accelerazione è misurata in unità di lunghezza per unità di tempo al quadrato, ad esempio metri al secondo quadrato (m/s^2) o g-forze (accelerazione gravitazionale).

Queste grandezze sono strettamente correlate tra loro e sono utilizzate per caratterizzare le vibrazioni e valutare la risposta di un sistema a sollecitazioni dinamiche. Nell'analisi vibrazionale, gli accelerometri e altri sensori sono comunemente utilizzati per misurare queste grandezze. I dati raccolti vengono quindi analizzati mediante tecniche di spettrografia e altre metodologie per identificare le frequenze di risonanza, le componenti transitorie e altre caratteristiche delle vibrazioni.

In generale, nella maggior parte delle analisi vibrazionali, l'accelerazione è preferibile per misurare frequenze alte, mentre la velocità viene utilizzata per le frequenze basse. Questa scelta è motivata dalle caratteristiche e dai vantaggi di ciascuna grandezza rispetto alle diverse frequenze di vibrazione.

Frequenze alte:

- Per le alte frequenze, le ampiezze delle vibrazioni tendono ad essere maggiori, rendendo l'accelerazione una misura più sensibile ed efficace. L'accelerazione offre una risposta più lineare e una maggiore precisione nei range di frequenza elevata, poiché la sua risposta dinamica è proporzionale alla frequenza.
- Molti guasti e problemi a frequenze elevate sono indicativi di condizioni critiche o di cedimento strutturale, e misurare l'accelerazione è utile per rilevare queste situazioni in modo più accurato.

Frequenze basse:

- Per le basse frequenze, come quelle associate a guasti meccanici o problemi di rotazione di macchinari industriali, l'ampiezza delle vibrazioni tende ad essere più bassa. In queste situazioni, la velocità è spesso preferita perché offre una migliore discriminazione tra le vibrazioni del sistema e il rumore di fondo.
- La velocità è meno sensibile alle frequenze alte, ma è più stabile e meno soggetta alle interferenze di rumore, rendendola ideale per misurare le vibrazioni a basse frequenze.

In alcuni casi, è comune utilizzare entrambe le grandezze simultaneamente: l'accelerazione per misurare le frequenze alte e la velocità per le frequenze basse. Questo approccio consente di ottenere una visione completa del comportamento vibrazionale del sistema, coprendo tutto lo spettro di frequenze rilevanti.

Tuttavia, è importante notare che la scelta della grandezza di misura dipende anche dal tipo di sistema o struttura che si sta analizzando e dalle specifiche esigenze dell'applicazione.

2.4.3 Analisi spettrale

Quando si effettua l'analisi delle vibrazioni di una macchina, si ottengono dati preziosi che richiedono un'attenta analisi per comprenderne il significato. Il successo di questa analisi dipende dalla corretta interpretazione degli spettri vibrazionali rispetto alle condizioni operative della macchina. Come descritto da [Power-MI, 2022] e [Augusto Di Benedetto, 2007] le fasi tipiche dell'analisi delle vibrazioni sono le seguenti:

- **Raccolta dei dati vibrazionali:** Il processo inizia con la raccolta dei dati mediante l'utilizzo di sensori di vibrazione come accelerometri. Questi sensori misurano l'accelerazione, la velocità o lo spostamento delle vibrazioni generate dalla macchina.
- **Trasformazione dei dati nel dominio delle frequenze:** I dati vibrazionali raccolti vengono solitamente convertiti dal dominio del tempo a quello delle frequenze mediante l'applicazione di una Trasformata di Fourier o di altre tecniche simili. Questo passaggio è essenziale per ottenere lo spettro di vibrazione, che mostra l'ampiezza delle vibrazioni in funzione delle diverse frequenze presenti nel segnale.
- **Identificazione dei picchi nello spettro:** Nel successivo passo, si individuano i picchi nello spettro vibrazionale. Il picco principale del primo ordine rappresenta la frequenza di rotazione dell'albero della macchina. Nei casi di macchine con più alberi, ogni albero avrà il suo picco. Oltre al picco di primo ordine, possono essere presenti armoniche e multipli interi di quest'ultimo, indicando fenomeni come squilibri, disequilibri e altre condizioni meccaniche anomale.
- **Analisi e interpretazione degli spettri:** Una volta individuati i picchi nello spettro, si procede all'analisi e all'interpretazione dei dati. Questo processo richiede competenze specifiche per identificare le cause delle vibrazioni e stabilire il livello di criticità dei problemi rilevati. L'analisi può coinvolgere la correlazione dei picchi con le caratteristiche della macchina, la conoscenza delle frequenze di risonanza e delle caratteristiche strutturali.
- **Diagnosi delle condizioni della macchina:** Sulla base delle informazioni ottenute dall'analisi, si procede con la diagnosi delle condizioni operative della macchina. Questo implica valutare la gravità dei problemi riscontrati e determinare se tali vibrazioni sono entro i limiti di tolleranza accettabili o se richiedono interventi correttivi.
- **Raccomandazioni e azioni correttive:** Infine, in base alla gravità dei problemi diagnosticati, si formulano raccomandazioni per le riparazioni o le azioni correttive necessarie. Queste possono includere interventi di manutenzione preventiva, bilanciamento delle parti rotanti, allineamento accurato delle componenti o altre misure per migliorare la stabilità e l'affidabilità della macchina.

Macchine rotanti e cuscinetti volventi

In questo capitolo si discuterà delle macchine elettriche rotanti e principalmente verrà introdotta una descrizione dettagliata dei cuscinetti volventi, i quali sono elementi meccanici essenziali presenti nella maggior parte di queste macchine.

3.1 Macchine rotanti

Come introdotto da [Verolino, 2019] le macchine rotanti costituiscono un elemento cruciale nel panorama dell'ingegneria meccanica, rivestendo un ruolo di rilievo in una vasta gamma di settori industriali. Le macchine rotanti sono dispositivi meccanici il cui elemento principale, il rotore, compie un movimento di rotazione attorno a un asse centrale. Queste macchine sono caratterizzate dalla loro capacità di trasferire energia da un punto all'altro attraverso il movimento rotativo del rotore.

Le macchine rotanti possono essere classificate in base a diversi criteri, tra cui la fonte di alimentazione, la configurazione del rotore e l'applicazione specifica. Le principali categorie includono motori elettrici, turbine, compressori e pompe.

Le macchine rotanti sono ovunque nelle moderne applicazioni tecnologiche. Dalle turbine utilizzate nella generazione di energia elettrica alle pompe impiegate nei processi industriali, il contributo delle macchine rotanti è onnipresente nei settori dell'energia, dell'industria manifatturiera e dei trasporti. In questo capitolo verrà trattato una componente molto importante delle macchine rotanti, i cuscinetti volventi, i quali hanno innumerevoli applicazioni negli impianti industriali.

3.2 Cuscinetti volventi

I cuscinetti volventi rappresentano una componente critica nell'ingegneria meccanica, rivestendo un ruolo di fondamentale importanza nelle macchine industriali e nei dispositivi meccanici. Essi sono dispositivi meccanici progettati per ridurre l'attrito tra le superfici di contatto in movimento relativo. La loro struttura basilare comprende sfere o rulli alloggiati in anelli esterni ed interni, consentendo il movimento rotativo o lineare con minima resistenza. I cuscinetti volventi sono classificati in diverse categorie in base al tipo di elemento rotante utilizzato (sfere o rulli) e alla disposizione delle piste di rotolamento. Le varianti includono cuscinetti a sfere, cuscinetti a rulli cilindrici, cuscinetti a rulli conici e cuscinetti orientabili.

Questi cuscinetti sono degli elementi che negli impianti industriali si guastano maggiormente e ci sono diverse tecniche per rilevare i fault in essi.

3.3 Tecniche di rilevamento dei guasti dei cuscinetti volventi

Come descritto da [Power-MI, 2022] e [Tedric A. Harris, 2006] l'affidabilità e la durata dei cuscinetti volventi dipende da una complessa combinazione di fattori, come: carichi, velocità di rotazione, lubrificazione, montaggio, temperatura e forze esterne come disallineamenti e squilibri. Data la complessità di questi fattori, diventa praticamente impossibile determinare con precisione la vita utile dei cuscinetti utilizzando metodi analitici convenzionali.

A causa dell'importanza e della criticità di questi componenti, è fondamentale adottare moderne tecniche di manutenzione predittiva basate sull'analisi delle vibrazioni. Questo approccio consente di monitorare in modo continuo e non invasivo il comportamento dei cuscinetti, rilevando eventuali anomalie o difetti in fase iniziale.

Gli obiettivi principali di questa analisi sono due: in primo luogo, identificare eventuali difetti nei cuscinetti volventi attraverso una diagnosi basata sugli spettri e le forme d'onda delle vibrazioni; in secondo luogo, determinare i fattori critici da considerare per prevedere il grado di deterioramento o la gravità del danno ai cuscinetti, al fine di stimarne la vita utile residua.

L'analisi spettrale delle vibrazioni gioca un ruolo cruciale in questo contesto, poiché non solo permette di monitorare costantemente lo stato dei cuscinetti, ma anche di individuare la causa principale di eventuali cedimenti o guasti cronici dei cuscinetti, fornendo così informazioni fondamentali per eliminare definitivamente le cause di degrado e migliorare la durata e l'affidabilità dei cuscinetti volventi.

Le tecniche per determinare i guasti ai cuscinetti si basano sul rilevamento di forti impatti o impulsi che si verificano durante la rotazione dell'albero. Verranno distinti in due gruppi principali: tecniche basate sull'analisi di parametri semplificati e tecniche basate sull'analisi delle forme d'onda spettrali e temporali.

3.3.1 Tecniche basate sull'analisi di parametri semplificati

Le tecniche basate sull'analisi di parametri semplificati sono:

- **RMS globale o valore di vibrazione della velocità di picco:** di solito viene applicato l'ISO 10816.
- **Fattore di cresta:** è il rapporto tra il valore di picco e il valore RMS.
- **Kurtosis:** è un parametro calcolato statisticamente dalla forma d'onda temporale che informa della quantità di picchi presenti nella forma d'onda.
- **Skewness:** è un parametro statistico calcolato dalla forma d'onda del tempo, che fornisce informazioni su come il segnale è simmetrico rispetto al suo valore medio.
- **Banda ad alta frequenza variabile:** tipicamente definita tra 1 kHz e 20 kHz.
- **HFD:** banda ad alta frequenza tra 5 e 30 kHz in accelerazione.
- **Spike-energy:** banda ad alta frequenza tra 5 e 60 kHz.
- **SPM o Shock pulses:** misura le vibrazioni ad alta frequenza tra 30 e 40 kHz.
- **Acoustic crack detection/IFD:** banda tra 80 e 120 kHz.

- **Emissione acustica:** banda con portata superiore a 200 kHz.

Il limite principale di queste tecniche è la loro incapacità di discernere tra problemi ai cuscinetti e altri problemi della macchina che generano anche vibrazioni ad alta frequenza come: problemi agli ingranaggi, cavitazione, frequenze elettriche, ecc. Un altro limite, non meno importante, sono i limitati intervalli di risposta in frequenza dei sensori che utilizzano e il metodo di montaggio utilizzato per fissarli alla macchina.

3.3.2 Tecniche basate sull'analisi delle forme d'onda spettrali e temporali

L'analisi della forma d'onda spettrale e temporale è determinante nella diagnosi del danneggiamento di un cuscinetto volvente, permettendo, attraverso il confronto dell'evoluzione delle ampiezze spettrali, di prevedere il futuro grado di deterioramento e pianificarne la sostituzione con largo anticipo. In questo capitolo verrà approfondito l'uso di queste tecniche.

3.4 Componenti dei cuscinetti volventi e frequenze di guasto

I cuscinetti volventi sono costituiti da diversi componenti osservabili nella Figura 3.1.

- **Inner ring (o anello interno):** è la parte del cuscinetto che è generalmente montata in modo fisso su un alloggiamento o una struttura fissa.
- **Outer ring (o anello esterno):** è la parte del cuscinetto che viene collegata all'albero rotante o all'elemento mobile della macchina.
- **Balls (o elementi volventi):** questi costituiscono il nucleo del cuscinetto e sono progettati per ridurre l'attrito tra l'anello esterno e quello interno consentendo una rotazione regolare. Gli elementi volventi possono essere di diversi tipi, tra cui sfere, rulli o aghi, a seconda del tipo di cuscinetto.
- **Cage (o gabbia):** è una struttura che mantiene gli elementi volventi separati e regolarmente distribuiti all'interno del cuscinetto. La gabbia aiuta a ridurre l'attrito tra gli elementi volventi e a mantenere la loro corretta posizione, contribuendo a garantire un funzionamento stabile e una durata prolungata del cuscinetto.

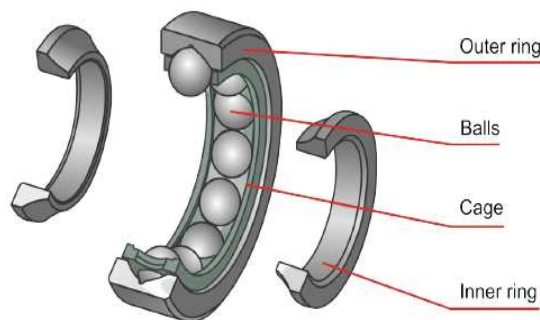


Figura 3.1: Componenti dei cuscinetti [Power-MI, 2022]

Il deterioramento di ciascuno di questi elementi genererà una o più frequenze caratteristiche negli spettri di frequenza che può consentire una rapida e facile identificazione. Come introdotto da [Power-MI, 2022] e [Augusto Di Benedetto, 2007] le quattro possibili frequenze di guasto dei cuscinetti sono:

- **BPFO (Ball Pass Frequency Outer):** la frequenza di guasto dell'anello esterno corrisponde fisicamente al numero di sfere o rulli che attraversano un dato punto dell'anello esterno ogni volta che l'albero compie un giro completo.
- **BPFI (Ball Pass Frequency Inner):** la frequenza di cedimento dell'anello interno corrisponde fisicamente al numero di sfere o rulli che attraversano un dato punto dell'anello interno ogni volta che l'albero compie un giro completo.
- **BSF (Ball Spin Frequency):** la frequenza di rottura dell'elemento rotante corrisponde fisicamente al numero di giri che una sfera o un rullo del cuscinetto compie ogni volta che l'albero compie un giro completo.
- **FTF (Fundamental Train Frequency):** la frequenza di guasto della gabbia corrisponde fisicamente al numero di giri che compie la gabbia del cuscinetto ogni volta che l'albero compie un giro completo.

3.4.1 Formule per il calcolo delle frequenze di cedimento dei cuscinetti

Le caratteristiche geometriche di ciascun cuscinetto determinano le sue frequenze di cedimento, che possono essere rilevate tramite le firme spettrali quando il cuscinetto è danneggiato. Le formule per calcolare queste frequenze sono illustrate nelle equazioni 3.4.1-3.4.4. Poiché le frequenze di guasto per la maggior parte dei cuscinetti non sono solitamente numeri interi, la vibrazione dominante causata da un difetto sarà asincrona, ovvero non corrisponderà alle armoniche della frequenza di rotazione dell'albero o della macchina. L'analisi delle firme spettrali e delle vibrazioni è una pratica comune per il monitoraggio dello stato dei cuscinetti e per intervenire tempestivamente in caso di problemi, evitando così guasti gravi.

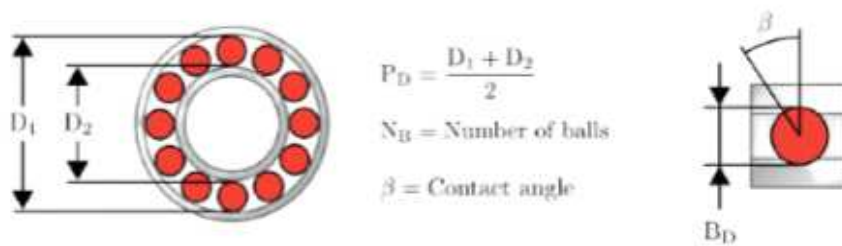


Figura 3.2: Composizione cuscinetti volventi [Power-MI, 2022]

$$BPFO = RPM \frac{N_B}{2} \left(1 - \frac{B_D}{P_D} \cos(\beta) \right) \quad (3.4.1)$$

$$BPFI = RPM \frac{N_B}{2} \left(1 + \frac{B_D}{P_D} \cos(\beta) \right) \quad (3.4.2)$$

$$BSF = RPM \frac{P_D}{B_D} \left[1 - \left(\frac{B_D}{P_D} \cos(\beta) \right)^2 \right] \quad (3.4.3)$$

$$FTF = RPM \frac{1}{2} \left(1 - \frac{B_D}{P_D} \cos(\beta) \right) \quad (3.4.4)$$

Quando uno dei componenti del cuscinetto subisce danni, nella firma spettrale delle vibrazioni possiamo distinguere la frequenza fondamentale corrispondente all'elemento danneggiato, accompagnata da armoniche. In situazioni in cui non si disponga di informazioni precise sui parametri fisici del cuscinetto, esistono formule empiriche che consentono di determinare le frequenze di cedimento delle piste del cuscinetto e della gabbia basandosi unicamente sul numero di corpi volventi e sulla velocità di rotazione.

$$BPFO = 0.4 * N_B * RPM \quad (3.4.5)$$

$$BPFI = 0.6 * N_B * RPM \quad (3.4.6)$$

$$FTF = 0.4 * RPM \quad (3.4.7)$$

Queste formule empiriche sono strumenti utili per stimare le frequenze di cedimento senza disporre di dati dettagliati sui parametri del cuscinetto. L'analisi di queste frequenze e le loro armoniche è essenziale per diagnosticare eventuali problemi nel cuscinetto e poter prendere misure correttive in modo tempestivo per prevenire guasti gravi.

3.4.2 Influenza della variazione dell'angolo di contatto nelle frequenze di cedimento del cuscinetto

L'angolo di contatto dei cuscinetti volventi si riferisce all'angolo formato tra la linea che congiunge il centro del cuscinetto con il punto di contatto tra il rullo e la pista e la linea perpendicolare alla pista stessa nel punto di contatto. Questo concetto è particolarmente rilevante nei cuscinetti a rullo o a sfere.

In generale, l'angolo di contatto è una delle caratteristiche geometriche fondamentali dei cuscinetti volventi ed è direttamente correlato alle prestazioni del cuscinetto stesso. Esso influenza la capacità del cuscinetto di sopportare carichi radiali e assiali, l'efficienza di rotolamento, la resistenza alle sollecitazioni e la durata del cuscinetto. Esistono cuscinetti con diversi tipi di angoli di contatto, a seconda dell'applicazione e dei requisiti di carico.

Le frequenze di cedimento dei cuscinetti, calcolate secondo le formule matematiche 3.4.1-3.4.4, sono strettamente legate all'angolo di contatto e ad altre caratteristiche geometriche specifiche del cuscinetto. Di conseguenza, anche una minima variazione dell'angolo di contatto può comportare una deviazione dalle frequenze ideali di cedimento, rendendo difficile l'identificazione precisa di tali frequenze nello spettro vibratorio.

Ci sono diverse possibili cause che possono generare una variazione dell'angolo di contatto, tra cui il disallineamento, la dilatazione termica, un eccessivo serraggio dei bulloni o danni alle piste del cuscinetto.

Queste variazioni influenzeranno le frequenze di cedimento pre-calcolate (frequenze ideali), portando a una mancata coincidenza con le frequenze effettive rilevate nello spettro vibratorio. Pertanto, in alcuni casi, è necessario ammettere un certo margine di tolleranza nell'identificazione delle frequenze difettose al fine di effettuare una corretta diagnosi del problema e intraprendere azioni di manutenzione appropriate.

3.5 Previsione del livello di gravità del danno al cuscinetto

Come descritto da [Power-MI, 2022], quando si effettua una diagnosi spettrale di un problema nel cuscinetto, è necessario tenere conto delle seguenti considerazioni:

- I guasti più frequenti dei componenti dei cuscinetti si verificano solitamente nel seguente ordine: pista esterna, pista interna, elementi volventi e infine nella gabbia.

- Riconoscere se un cuscinetto presenta un difetto nella pista interna o esterna riveste una grande importanza poiché ciò consente di stimare la vita utile residua del cuscinetto e adottare misure adeguate per la manutenzione. In generale, i cuscinetti con difetti nella pista esterna tendono ad avere una durata più lunga rispetto a quelli con difetti nella pista interna. Questo fatto è dovuto a diverse ragioni legate alla geometria e al modo in cui i carichi sono distribuiti nel cuscinetto. Identificare correttamente il tipo di difetto aiuta gli operatori a valutare l'urgenza della sostituzione o della manutenzione del cuscinetto e a pianificare meglio le attività di manutenzione preventiva per garantire un funzionamento sicuro ed efficiente delle macchine.
- I difetti identici presenti nelle due piste del cuscinetto si manifestano con ampiezze di frequenza di guasto differenti. Questa disparità è dovuta alla configurazione della struttura portante del cuscinetto e al percorso che la vibrazione deve seguire per raggiungere il sensore. Se il sensore viene posizionato vicino alla zona di carico del cuscinetto, dove solitamente si verificano i difetti nell'anello esterno, la trasmissione dell'energia di vibrazione al sensore avviene attraverso l'anello esterno e l'involucro. Questo circuito di trasmissione fornisce una misurazione affidabile della vibrazione generata dal difetto nell'anello esterno. Al contrario, se il guasto si trova nella zona dell'anello interno, che nella maggior parte dei casi è quella che ruota, l'energia di vibrazione viene trasmessa al sensore attraverso le sfere o i rulli, la gabbia, la pista esterna e l'involucro. Questo tratto di trasmissione è solitamente più debole e influisce sulla riduzione dell'ampiezza del segnale di vibrazione rilevato. Inoltre, bisogna considerare che la pista interna gira e spesso si trova al di fuori della zona di carico, il che implica un'ampiezza di impatto significativamente inferiore rispetto all'anello esterno. Affinché i dati vengano acquisiti correttamente, è importante garantire che il campionamento delle misurazioni deve corrispondere ad un giro completo dell'albero. In questo modo, viene assicurato di rilevare il segnale proveniente dagli impatti prodotti quando il difetto attraversa la zona di carico. Il corretto posizionamento dei sensori e la scelta del giusto intervallo di campionamento sono fondamentali per una diagnosi accurata delle condizioni del cuscinetto e per pianificare le azioni di manutenzione necessarie.
- Le letture delle vibrazioni ad alta frequenza costituiscono un indicatore di partenza dell'inizio del deterioramento del cuscinetto. Quando il cuscinetto presenta piccoli difetti, come pitting o micro-crepe, gli impatti generati da questi difetti tendono a eccitare le frequenze naturali delle piste del cuscinetto. Per rilevare queste vibrazioni ad alta frequenza, le misurazioni vengono effettuate in unità di accelerazione, tipicamente nella banda compresa tra 1 e 20 kHz. Questa gamma di frequenza è particolarmente sensibile all'identificazione dei primi segni di problemi nel cuscinetto. Monitorare le vibrazioni ad alta frequenza consente agli operatori di intervenire tempestivamente, eseguendo una diagnosi anticipata, adottando azioni di manutenzione preventive per evitare danni più gravi e prolungare la vita utile del cuscinetto.
- Durante la fase iniziale del deterioramento della pista del cuscinetto, è comune osservare negli spettri di vibrazione la presenza di armoniche ad alta frequenza e bassa ampiezza. Identificare queste caratteristiche è essenziale per rilevare il deterioramento iniziale del cuscinetto e monitorarne l'evoluzione nel tempo. Ciò consente di pianificare con largo anticipo eventuali interventi di sostituzione o manutenzione. Man mano che il danno progredisce, le ampiezze delle frequenze di guasto inizialmente individuate aumenteranno e appariranno le armoniche intermedie di tali frequenze di guasto. Nel corso del tempo, durante la fase finale in cui il danno diventa più grave, diventeranno chiaramente evidenti le prime armoniche delle frequenze di guasto, rendendo ancora più evidente la necessità di interventi tempestivi.

- Il verificarsi di nuove frequenze di cedimento dei cuscinetti indicherà una maggiore gravità del difetto.
- Le frequenze di cedimento della pista del cuscinetto sono talvolta accompagnate da bande laterali la cui differenza di frequenza rispetto alla fondamentale coincide con la frequenza di rotazione dell'albero. Un'altra frequenza che può modulare le frequenze di cedimento delle piste dei cuscinetti è la frequenza di cedimento della gabbia del cuscinetto o FTF, che indica un danno maggiore del cuscinetto. L'aumento del numero e dell'ampiezza di queste bande laterali indicherà la progressione del danno.
- Quando si verifica un danno significativo al cuscinetto, le singole frequenze possono scomparire e possono comparire bande di energia significative, che spesso indicano cambiamenti nella geometria del cuscinetto.
- Se la lubrificazione non è adeguata, il deterioramento del cuscinetto accelererà, quindi è buona norma che quando il danno viene identificato, il cuscinetto sia opportunamente lubrificato per cercare di prolungarne la vita.
- L'analisi della forma d'onda temporale può essere utile per la diagnosi dei cuscinetti. Il cuscinetto deteriorato sarà caratterizzato da forti impatti in accelerazione e una differenza di frequenza tra i picchi che coincidono approssimativamente con una delle frequenze di guasto.
- In che modo la velocità di rotazione influenza la progressione del deterioramento dei cuscinetti? È stato verificato in pratica che per velocità di rotazione di 3000RPM o superiori il degrado è notevolmente accelerato, mentre per velocità inferiori a 300RPM la progressione del difetto è molto lenta e può evolvere per diversi mesi.

L'analisi di casi storici è un prezioso strumento per valutare la gravità dei problemi nei futuri casi. Ad esempio, se si riuscisse ad identificare un difetto del cuscinetto e prevedere che la sua durata sarà di tre settimane, si avrà un'opportunità eccellente per confrontare il difetto previsto con i dati reali di vibrazione ottenuti. Questo consentirà di stabilire quanto il difetto si sia effettivamente sviluppato.

Quando si rileva un danno significativo, si possono utilizzare queste informazioni storiche per valutare la gravità dei futuri problemi. Se i livelli di vibrazione raggiungono quelli ottenuti durante l'ultima lettura, si saprà che il danno è molto grave e richiederà un'azione immediata. D'altra parte, se il cuscinetto precedentemente cambiato mostrava solo una regolare desquamazione, si potrebbe prevedere che, se in futuro il nuovo cuscinetto mostrerà gli stessi sintomi, potrà continuare a funzionare per un periodo prolungato.

Come descritto da [G. Vachtsevanos, 2006], anche se le macchine possono non essere identiche, è possibile estrapolare i risultati di cuscinetti simili, tenendo conto di piccole differenze di comportamento.

Questo approccio basato su dati storici ci aiuta a prendere decisioni più informate riguardo alla manutenzione preventiva e ci consente di pianificare interventi tempestivi per evitare guasti catastrofici e prolungare la vita utile dei cuscinetti in modo efficiente.

3.6 Difetti tipici del cuscinetto e identificazione spettrale

Di seguito sono riportati i difetti più tipici dei cuscinetti e la loro identificazione nello spettro di frequenza:

- **Difetti della pista esterna:** lo spettro è caratterizzato dalla presenza di picchi armonici della frequenza di rottura della pista esterna (tra 8 e 10 armoniche del BPFO, ciò significa multipli interi della frequenza di rottura dell'anello esterno). Vedere Figura 3.3.
- **Difetti della pista interna :** lo spettro mostra diversi picchi armonici della frequenza di rottura della pista interna (di solito tra 8 e 10 armoniche BPFI, ciò significa multipli interi della frequenza di rottura dell'anello interno) modulati da bande laterali a $1RPM$. Vedere Figura 3.4.
- **Difetti delle sfere o dei rulli:** sono caratterizzati dalla presenza nello spettro di armoniche alla frequenza di deterioramento degli elementi volventi (BSF). Nella maggior parte dei casi, l'armonica di maggiore ampiezza indica il numero di sfere o rulli danneggiati. Questi difetti sono di solito accompagnati da difetti sulle piste, cioè le superfici interne ed esterne del cuscinetto sulle quali scorrono le sfere o i rulli. Vedere Figura 3.5.
- **Difetti della gabbia:** sono caratterizzati dalla presenza nello spettro della frequenza di cedimento della gabbia (FTF) e delle sue armoniche. In generale, un difetto nella gabbia è accompagnato da difetti sulle piste e la FTF di solito modula una di queste frequenze di cedimento delle piste, dando luogo a somme e/o differenze di frequenze. Vedere Figura 3.6.
- **Difetti di componenti multipli:** è piuttosto comune trovare cuscinetti con più componenti deteriorati e, in tal caso, si verificheranno molteplici frequenze di cedimento e le relative armoniche corrispondenti.
- **Looseness (gioco):** nei cuscinetti può essere distinto in vari tipi:
 - **Eccessivo gioco interno del cuscinetto:** solitamente presenta una firma spettrale caratterizzata dalla presenza di vibrazioni sincrone (armoniche della velocità di rotazione), vibrazioni sotto-sincrone ($0.5 \times RPM$) e non sincrone ($1.5 \times RPM$, $2.5 \times RPM$, $3.5 \times RPM$, ecc.). A volte queste frequenze possono essere modulate dalla FTF.
 - **Gioco tra cuscinetto e albero:** si osservano diverse armoniche della frequenza di rotazione e di solito quella dominante è la $3 \times RPM$ (frequenza di rotazione moltiplicata per 3).
 - **Gioco tra cuscinetto e involucre:** si presentano diverse armoniche della frequenza di rotazione, e i picchi a $1 \times RPM$ e $4 \times RPM$ sono quelli con ampiezze più elevate.
- **Allineamento errato del cuscinetto:** come già accennato nel capitolo riguardante l'allineamento, le firme spettrali sono caratterizzate dalla presenza di vibrazioni a varie armoniche della frequenza di rotazione, con l'ampiezza più significativa a $N_B \times RPM$, dove N_B è il numero di elementi volventi nel cuscinetto.
- **Inadeguata lubrificazione:** i problemi di lubrificazione sono caratterizzati da vibrazioni ad alta frequenza (tra 1 kHz e 20 kHz), con bande di picchi spaziate tra loro, a causa dell'eccitazione delle frequenze di risonanza dei cuscinetti in questi intervalli di frequenza.

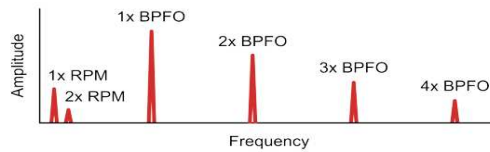


Figura 3.3: Fault anello esterno [Power-MI, 2202]

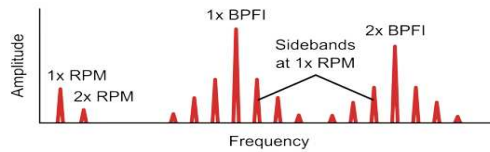


Figura 3.4: Fault anello interno [Power-MI, 2022]

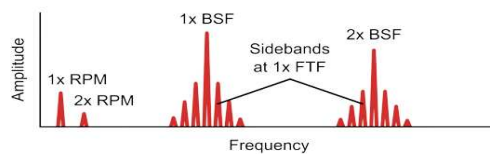


Figura 3.5: Fault nelle sfere o rulli [Power-MI, 2022]

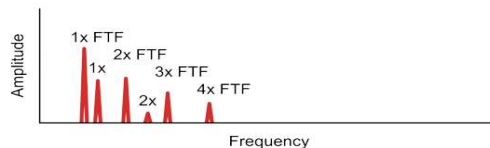


Figura 3.6: Fault nella gabbia [Power-MI, 2022]

La possibilità di suddividere il valore complessivo dello spettro in diverse bande di frequenza consente di individuare in anticipo le aree in cui di solito si manifestano i problemi più tipici, aiutando ad identificarli anche prima di poter visualizzare lo spettro nelle frequenze e nel dominio temporale. Per le macchine tipiche, dove l'obiettivo è monitorare problemi a bassa e media frequenza (disequilibrio, disallineamento, gioco, ecc.) e problemi ad alta frequenza (cuscinetti), viene raccomandato di utilizzare le bande di frequenza indicate nella tabella 3.1.

Suddividendo lo spettro complessivo in queste bande di frequenza, è possibile individuare le specifiche aree in cui si verificano comunemente i problemi. Questo approccio consente di identificare le frequenze rilevanti associate a specifici componenti della macchina o a possibili guasti, contribuendo in modo proattivo alla manutenzione delle macchine:

- Identificazione anticipata dei problemi.
- Diagnosi proattiva.

- Ottimizzazione della manutenzione.
- Monitoraggio continuo.
- Riduzione dei costi di manutenzione.

Parametro spettrale	Larghezza di banda
Squilibrio, disallineamento, allentamento, sfregamento e usura della cinghia	Da 1,5 a 2,5 RPM
Disallineamento e allentamento	Da 1,5 a 2,5 RPM
Looseness, disallineamento e BSF	Da 2,5 a 4,5 RPM
Prima armonica delle frequenze di guasto del cuscinetto BPFO, BPFI e BSF.	Da 4,5 a 20,5 RPM
Armoniche con frequenze di guasto dei cuscinetti più elevate e presenza di frequenze legate a problemi elettrici	Da 20,5 a 50 RPM
Banda spettrale ad alta frequenza, in unità di accelerazione (G). Parametro di allarme in caso di precoce deterioramento del cuscinetto, presenza di cavitazione o mancanza di lubrificazione.	Da 1 a 20 kHz

Tabella 3.1: Bande spettrali per il rilevamento di problemi ai cuscinetti volventi

3.7 Fasi di deterioramento

Le fasi del deterioramento di un cuscinetto sono quattro e sono le seguenti:

- **Fase 1:** in questa stadio, il cuscinetto è in perfette condizioni e nello spettro sono visibili solo la frequenza di rotazione ed eventualmente alcune delle sue armoniche.
- **Fase 2:** le letture delle vibrazioni si verificano ad alta frequenza e sono il primo sintomo dell'inizio del deterioramento di un cuscinetto. Queste letture sono dovute ad urti, causati da un piccolo difetto, che tendono ad eccitare le frequenze naturali ad alta frequenza delle piste del cuscinetto. Queste misurazioni sono prese dallo spettro di accelerazione in una banda compresa tra 1 kHz e 20 kHz.
- **Fase 3:** compaiono le frequenze di guasto caratteristiche e le loro armoniche. Man mano che il danno progredisce, aumenta sia l'ampiezza delle frequenze di guasto e delle loro armoniche, sia la vibrazione ad alta frequenza in accelerazione. Il monitoraggio della loro evoluzione ci consente di pianificare una sostituzione con largo anticipo.
- **Fase 4:** questa è la stadio finale del cuscinetto. Quando è notevolmente danneggiato, ci sono sintomi simili all'allentamento e all'attrito. C'è anche un rumore di fondo ben visibile in accelerazione ad alta frequenza. Aumenta l'ampiezza di 1RPM e le sue armoniche e diminuisce o scompare l'ampiezza delle frequenze di guasto man mano che vengono mascherate dal rumore di fondo.

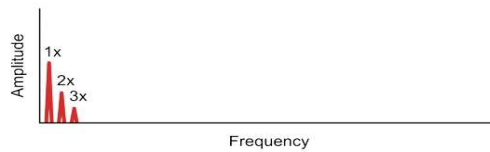


Figura 3.7: Fase 1 di deterioramento del cuscinetto [Power-MI, 2022]

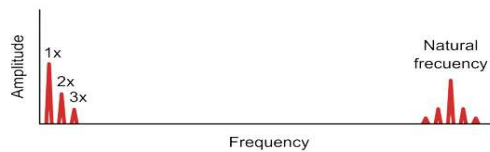


Figura 3.8: Fase 2 di deterioramento del cuscinetto [Power-MI, 2022]

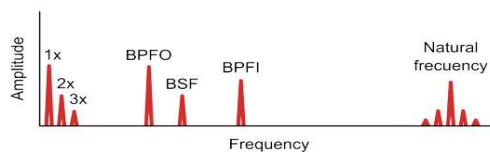


Figura 3.9: Fase 3 di deterioramento del cuscinetto [Power-MI, 2022]

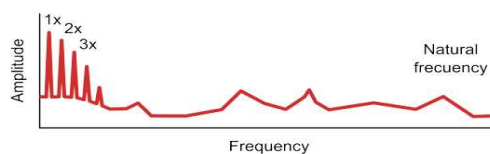


Figura 3.10: Fase 4 di deterioramento del cuscinetto [Power-MI, 2022]

3.8 Demodulazione o analisi dell'involuppo

L'uso della demodulazione, o dell'analisi dell'involuppo, rappresenta un eccellente strumento complementare per migliorare l'analisi spettrale nella rilevazione di difetti nei cuscinetti e nei riduttori. Questa sezione descrive cos'è la demodulazione e le sue applicazioni. L'involuppo di un segnale è una tecnica utilizzata per estrarre la componente a bassa frequenza o l'andamento generale di un segnale complesso. In termini semplici, l'involuppo rappresenta una "fascia" che segue i punti più alti del segnale originale. Questo può essere utile in situazioni in cui si desidera concentrarsi sugli aspetti a bassa frequenza di un segnale e trascurare le oscillazioni ad alta frequenza. L'involuppo di un segnale è spesso utilizzato

nell'analisi delle vibrazioni per rilevare la presenza di modulazione di ampiezza e aiuta ad evidenziare le variazioni nell'ampiezza del segnale dovute a difetti o altre anomalie. La demodulazione è il processo di estrazione di un segnale modulante o di bassa frequenza da un segnale modulato, che può avere componenti ad alta frequenza. In altre parole, la demodulazione è il processo di "decodificare" un segnale modulato per recuperare l'informazione originale. In sintesi, l'involuppo di un segnale è il processo di estrazione dell'andamento generale da un segnale complesso, mentre la demodulazione è il processo di estrazione di un segnale modulante o di bassa frequenza da un segnale modulato.

La demodulazione può essere spiegata con il seguente esempio: supponiamo di avere un cuscinetto con un difetto nella pista esterna. Ogni volta che una sfera passa sul difetto, genera un piccolo impatto che percepiamo come un "click". Se misuriamo le vibrazioni in questo cuscinetto durante una rotazione completa dell'albero, otterremo una forma d'onda come rappresentato in colore rosso nella Figura 3.11.

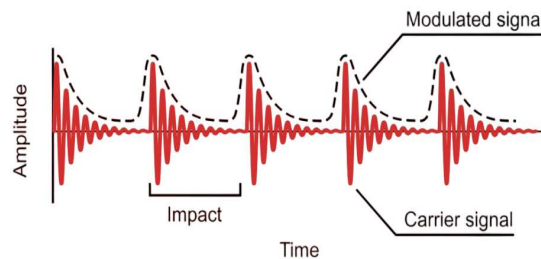


Figura 3.11: Esempio di forma d'onda di un cuscinetto difettoso [Power-MI, 2022]

L'impatto generato quando ciascuna sfera del cuscinetto incontra una discontinuità è simile a quello che il cuscinetto subisce quando viene colpito con un martello, somigliante al suono di una campana colpita dallo stesso. La struttura formata dal cuscinetto, dal suo supporto e dall'albero della macchina agisce insieme come una campana di risonanza. Il suono che emette quando una sfera colpisce il difetto corrisponde a una certa frequenza, chiamata frequenza naturale o frequenza di risonanza, che di solito è una frequenza elevata, compresa tra 1 e 20 kHz a seconda della massa e della rigidità del cuscinetto. Questa frequenza di risonanza non è concentrata solo su alcune frequenze particolari, ma su bande di esse. La frequenza di risonanza è una proprietà della struttura e non viene influenzata da quante volte essa viene colpita o dall'intensità dell'impatto.

Con questa comprensione, è possibile guardare da un altro punto di vista le vibrazioni associate a questo difetto: il cuscinetto o la "campana" suona costantemente e quando il difetto viene colpito dalla sfera, il suono semplicemente diventa più forte. Questo effetto di variazione dell'ampiezza della frequenza naturale è noto come modulazione di ampiezza ed è chiaramente visibile nella Figura 3.11, in cui la frequenza naturale funge da segnale portante e il segnale BPFO funge da segnale modulato.

La scelta dell'utilizzo della demodulazione nella diagnosi per valutare fault è estremamente utile per fornire all'analista le seguenti capacità:

- **Lecture con molta vibrazione casuale:** una ragione per utilizzare la demodulazione è l'esistenza di macchine che a causa del loro normale funzionamento generano una grande quantità di vibrazione casuale che aumenta il rumore di fondo nello spettro. Questo livello di rumore può coprire le armoniche associate alle frequenze di rottura dei cuscinetti nello spettro standard. Inoltre, il rumore di fondo prodotto dalla vibrazione

casuale è spesso simile a quello causato dall'usura grave dei cuscinetti, il che potrebbe far passare inosservata tale consumo. La demodulazione può essere estremamente utile in questa situazione.

- **Lecture a bassa velocità:** la demodulazione è anche molto utile per diagnosticare problemi nei cuscinetti a rulli di macchine a velocità estremamente basse. Se si ha una macchina che ruota a $60RPM$ ($1Hz$) e si desidera individuare le frequenze dei cuscinetti in un intervallo compreso tra $3RPM$ e $10RPM$, le frequenze di queste armoniche saranno inferiori a $10Hz$. Quando si lavora con lo spettro di vibrazione in velocità, l'integrazione necessaria dalla accelerazione alla velocità funge da filtro passa-alto a $10Hz$, rendendo queste basse frequenze quasi indistinguibili. Fortunatamente, i dati demodulati rimangono validi poiché la frequenza di risonanza del cuscinetto è ad alta frequenza.
- **Rilevamento veloce:** la demodulazione consente di rilevare molto rapidamente un possibile guasto al cuscinetto, poiché le frequenze di rottura dei cuscinetti che appaiono nello spettro di demodulazione sono il primo indicatore dell'inizio di un problema in essi. La demodulazione può rilevare queste frequenze di rottura prima che diventino abbastanza evidenti da apparire nello spettro standard.
- **Miglioramento delle capacità diagnostiche:** se si verifica un picco a una determinata frequenza nello spettro di demodulazione, è molto probabile che si tratti di una frequenza di rottura del cuscinetto. Quindi, si può aspettare che essa appaia anche nello spettro standard man mano che il deterioramento progredisce. Al contrario, se un picco anomalo, non sincronizzato con la frequenza di rotazione dell'albero, appare nello spettro standard e appare anche nello spettro di demodulazione, allora è possibile essere sicuri che si tratti di una frequenza di rottura del cuscinetto. Infine, se si notano frequenze di rottura del cuscinetto nello spettro standard in due diverse posizioni della macchina e queste frequenze appaiono nello spettro di demodulazione di una posizione ma non nell'altra, si può concludere che la posizione in cui appaiono è quella dove si ha un problema al cuscinetto.

Purtroppo, non è possibile utilizzare esclusivamente lo spettro di demodulazione come unico strumento diagnostico perché non consente di correlare l'ampiezza delle frequenze di guasto con lo stato di deterioramento del cuscinetto. Il motivo di ciò è dovuto al fatto che man mano che il deterioramento del cuscinetto progredisce, il numero di difetti nel cuscinetto aumenta, causando una casualità sempre maggiore nel segnale demodulato. Pertanto, all'evolversi del danneggiamento del cuscinetto, le armoniche nello spettro di demodulazione perderanno ampiezza e potrebbero addirittura scomparire.

Inoltre, il livello di picco nello spettro di demodulazione è semplicemente relativo a quanto sono rumorosi i "click" rispetto al rumore di fondo del cuscinetto. Questo rapporto, da solo, non è sufficiente per correlare la gravità del guasto al cuscinetto.

Strumenti e componenti utilizzati

In questo capitolo verranno descritti sinteticamente gli strumenti e i componenti utilizzati per svolgere l'analisi.

Inizialmente è stata attuata un'analisi per capire quali cuscinetti utilizzare con i macchinari a disposizione dell'azienda per effettuare le prove. Di seguito verranno elencati gli strumenti e i componenti, i quali sono correlati da una descrizione.

- Sistema di acquisizione dati
- Macchina ad asse orizzontale rotovariatore
- Cannotto
- Cuscinetti a rullo FAG N207 e FAG NU207
- Cuscinetto a sfere SKF 6207 2RS
- Accelerometri piezoelettrici PCB HT356A33
- Matlab per elaborazione dei dati
- Python per elaborazione dei dati

4.1 Sistema di acquisizione dati

Il sistema di acquisizione è stato creato dall'azienda Idea dopo un'analisi effettuata considerando le necessità dell'impresa PIERALISI MAIP. Il sistema di acquisizione comprensivo di monitor e PC è in grado di pilotare ed acquisire i seguenti dispositivi:

- n.8 accelerometri triassiali (IEPE).
- n.16 sensori di prossimità (PNP o NPN).
- n.16 sonde di temperatura PT100 (2, 3 o 4 fili).
- n.8 estensimetri (a ponte intero o a mezzo ponte).

Tali acquisizioni vengono effettuate tramite la FPGA del Compact RIO della NI e tutti i valori ottenuti sono caricati su un server NAS (circa 8 Gbyte di memoria + 8 Gbyte di backup). Il sistema è anche dotato di un UPS indipendente per permetterne il funzionamento stabile in mancanza di rete o sotto sbalzi di corrente indesiderati. Nella Figura 4.1 si può osservare il sistema di acquisizione.



Figura 4.1: Sistema di acquisizione nell'ambiente operativo

Principalmente il sistema di acquisizione è stato usato per apprendere i dati da due accelerometri e per questo non è stato possibile sfruttare tutta la potenza di calcolo che può fornire il sistema in se, infatti i canali possono campionare a 10kHz che è un valore superiore a quello che effettivamente serve. Quando viene avviato il sistema è necessario attraverso l'interfaccia software impostare il modo di acquisizione dei dati, in questo caso accelerometri. La schermata per la configurazione degli accelerometri la si può vedere in Figura 4.2.

Da questa sezione è possibile configurare i parametri relativi all'acquisizione degli accelerometri: selezionare i canali da acquisire e la configurazione dei parametri per la conversione della grandezza acquisita. Nella schermata sono presenti tre tabelle ognuna relativa ad un modulo di acquisizione degli accelerometri (ogni modulo gestisce 8 canali). Ciascuna tabella è composta da diverse colonne (En, AI Channel, Unity, BiasVoltage, Sensitivity). Di seguito il significato di ogni colonna:

- **EN:** in questa colonna è possibile abilitare o meno l'acquisizione di ciascun canale.
- **AI CHANNEL:** rappresenta il canale fisico sul quale è connesso quel segnale (questa etichetta non è modificabile).
- **CHANNEL LABEL:** è un nome simbolico che è possibile attribuire a ciascun canale. Questo nome sarà quello riportato nella tabella dei grafici e sui file "*tdms (Technical Data Management Streaming)*" di salvataggio.

- **UNITY:** è un parametro riportato sul certificato di calibrazione dell'accelerometro. Va inserito solo se l'acquisizione è effettuata in DC. In questo caso la misura è effettuata in AC e quindi il parametro viene lasciato a 0.
- **SENSITIVITY:** è un parametro presente sul certificato di calibrazione dell'accelerometro. Per ogni asse è specificato un valore di sensitivity. Questo valore va sempre inserito ed è necessario per la conversione della misura da Volt in accelerazione.

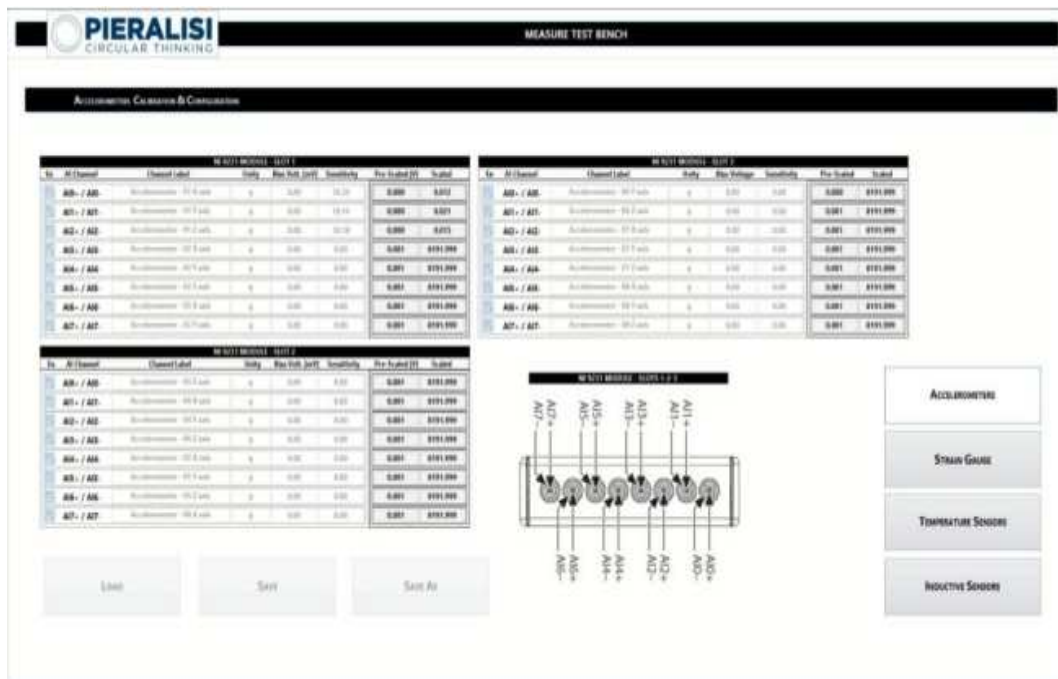


Figura 4.2: Schermata di configurazione accelerometri

Dopo che è stata impostata la configurazione dei canali è possibile salvare la stessa in un file formato json per richiamarla in altre prove senza dover ripetere il procedimento ogni volta.

Una volta fatto tutto ciò, è possibile passare al passo successivo, cioè si può procedere nella parte dove è possibile iniziare il test e salvare i risultati. Questa parte dell'interfaccia viene chiamata "Schermata di AUTO CYCLE".

Dalla schermata in Figura 4.3 è possibile accedere ai comandi per dare lo start e lo stop al test ed eseguire il grafico di 4 grandezze acquisite per ciascuna tipologia di sensore, selezionando dal menù a tendina presente su ogni grafico, il canale che si vuole visualizzare. Nella lista dei canali, saranno presenti solo quelli che in fase di configurazione sono stati abilitati all'acquisizione. Scorrendo con i tab in alto è possibile visualizzare i grafici relativi a ciascuna tipologia di sensore. Nella parte laterale è possibile (prima di dare il via al test) editare il campo test description che sarà salvato insieme ai file *tdms*. In esso può essere inserita una descrizione dettagliata del tipo di test effettuato. Al di sotto di test description sono presenti due campi che indicano, rispettivamente, la cartella locale temporanea (sul PC) e la cartella remota (sul NAS) in cui sono salvati i file di acquisizione. Premendo i rispettivi tasti sotto ciascun campo è possibile accedere a ciascuna cartella. Più in basso si trova un visualizzatore dei file scritti in locale e l'indicazione dell'ultimo file trasferito sul NAS.

Una volta premuto lo stop test, tutti i file vengono trasferiti e la cartella locale temporanea sul PC viene eliminata. Dopo che è stato schiacciato il tasto stop test, attendere che tutti i file siano trasferiti prima di chiudere l'applicativo. Le cartelle salvate sul NAS, per

ogni test effettuato, sono nominate in funzione della data e dell'ora di inizio del test (Es. Test_19.07.2021 09.35.39.268). All'interno di questa cartella sono contenute delle sottocartelle delle acquisizioni nominate con un numero crescente (Es. Acquisition_00000000000000, Acquisition_00000000000001, ecc..) ed una cartella Acquisition info che contiene il file di configurazione dei canali per quella prova, la configurazione delle rampe di velocità, un file con il test description e 2 file con, rispettivamente, la descrizione di tutti i warning di temperatura e di RPM avvenuti durante la prova.



Figura 4.3: Schermata AUTO CYCLE

4.2 Macchina ad asse orizzontale e canotto

La macchina utilizzata è un rotovariatore ad asse orizzontale il quale è un dispositivo progettato, costruito ed industrializzato internamente dall'azienda Pieralisi MAIP, coperto da brevetto internazionale che permette la regolazione continua ed automatica del decanter centrifugo in funzione delle caratteristiche del prodotto alimentato, grazie al controllo elettronico che regola i giri differenziali della coclea in relazione al carico istantaneo; consente il risparmio energetico poiché restituisce al motore principale la coppia assorbita.

I giri del rotovariatore vengono forniti attraverso un quadro elettrico dove è possibile modificare essi a piacimento, per questi esperimenti è stato deciso di far girare il rotovariatore a $562.5RPM$ in modo il canotto girasse a $1000RPM$ secondo l'equazione 4.2.1 dove RPM_C rappresenta i giri al minuto del canotto, RPM_R rappresenta i giri al minuto del rotovariatore, D_C i denti del canotto (80 in questo caso), D_R i denti del rotovariatore (45 in questo caso).

Attraverso un cinghia il rotovaratiatore è stato collegato al canotto, il quale funge da albero motore e dove i cuscinetti con e senza guasti vengono posti; il canotto è stato modificato preventivamente per fare in modo che i cuscinetti potessero essere inseriti ed estratti correttamente senza uno sforzo eccessivo. In Figura 4.4 si può osservare a sinistra e di colore blu il rotovariatore e a destra e di colore verde il canotto.

$$RPM_C = RPM_R \frac{D_C}{D_R} \quad (4.2.1)$$



Figura 4.4: Rotovariatore e canotto

4.3 Cuscinetti a rullo FAG e a sfere SKF

I cuscinetti che sono stati scelti per questi esperimenti sono i FAG N207 e FAG NU207. Inizialmente la scelta era caduta sul marchio SKF, ma per questione economiche e di tempistiche la scelta è ricaduta sui cuscinetti citati sopra. Inoltre, essendo che in azienda erano presenti dei cuscinetti SKF a sfere è stato deciso di utilizzare anche essi. In una parte del canotto sono stati montati gli SKF (colore blu nella Figura 4.5), mentre dall'altra parte sono stati montati i FAG (colore rosso nella Figura 4.5).

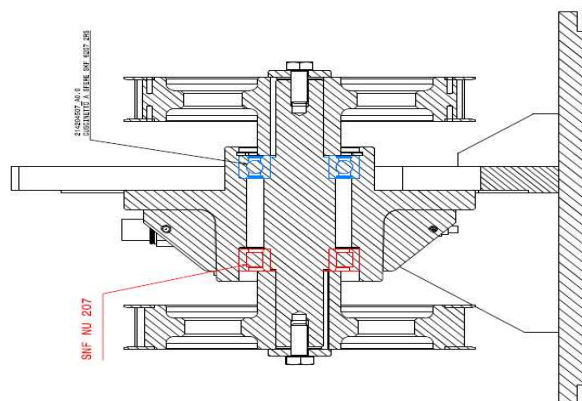


Figura 4.5: Posizione cuscinetti FAG e SKF

4.3.1 Cuscinetto a rullo FAG

I cuscinetti a rulli cilindrici sono disponibili in una vasta gamma di design, serie, varianti e dimensioni. In termini di design, si differenziano principalmente per il numero di corone di rulli, gli orletti dell'anello interno/esterno e i design e materiali per le gabbie.

Questi cuscinetti possono operare in applicazioni con pesanti carichi radiali e velocità elevate. Inoltre, consentono il disallineamento assiale (eccetto i tipi con orletti sia sull'anello interno che esterno), quindi offrono elevata rigidezza, basso attrito e lunga durata di esercizio.

I cuscinetti a rulli cilindrici sono disponibili anche in versioni schermate o in due metà. Nei cuscinetti schermati, i rulli sono protetti da agenti contaminanti, acqua e polvere e le tenute realizzano anche le funzioni di ritenzione del lubrificante ed esclusione degli agenti contaminanti. Ciò consente di ridurre l'attrito e prolungare la durata di esercizio. I

cuscinetti in due metà sono destinati principalmente a disposizioni difficilmente accessibili, come ad esempio quelle degli alberi a gomiti, per semplificare le attività di manutenzione e sostituzione.

Sono stati scelti i cuscinetti del marchio FAG N207 e NU207 per le dimensioni che hanno, il numero di rulli e perché sono complementari l'uno con l'altro. Ciò vuol dire che è possibile realizzare, per quanto riguarda i FAG N207, dei fault esterni e nei rulli grazie alla possibilità di estrarre l'anello esterno; mentre per quanto riguarda i FAG NU207 è possibile realizzare i fault nell'anello interno e nei rulli grazie alla sua estrazione. E' possibile effettuare diverse prove. In Figura 4.6 e 4.7 si possono osservare i due cuscinetti.



Figura 4.6: Cuscinetto a rullo FAG N207



Figura 4.7: Cuscinetto a rullo FAG NU207

Attraverso un calibro sono state prese le misure dei due cuscinetti dopo che sono stati spaccati, per calcolare particolari frequenze di guasto. Nella Tabella 4.1 vengono riportati i parametri presi attraverso il calibro.

Cuscinetto	N_b	D_1	D_2	B_d	β
FAG N207	14	63.92mm	44.79mm	9.95mm	0
FAG NU207	14	63.92mm	44.79mm	9.95mm	0

Tabella 4.1: Tabella parametri cuscinetti

Dove N_b indica il numero di palle o rulli, D_1 indica il diametro dell'anello esterno, D_2 indica il diametro dell'anello interno, B_d indica il diametro dei rulli e β indica l'angolo di contatto che per convenzione viene assunto 0.

4.3.2 Cuscinetto a sfere SKF 6207

I cuscinetti radiali a una corona di sfere sono particolarmente versatili, assicurano basso attrito ed essendo ottimizzati per ridurre i livelli di rumorosità e vibrazioni consentono velocità di rotazione elevate. Possono sopportare carichi assiali e radiali in ambo le direzioni, sono semplici da montare e richiedono meno attività di manutenzione rispetto a molti altri tipi di cuscinetti.

Sono stati scelti i cuscinetti SKF 6207 perché già presenti in azienda, in tal modo è stato possibile un risparmio sia di tempo sia economico. Il cuscinetto SKF 6207 si può osservare in Figura 4.8.



Figura 4.8: Cuscinetto SKF 6207

4.4 Accelerometri piezoelettrici PCB

Gli accelerometri usati sono i piezoelettrici triassiali PCB modello HT356A33, e la scelta è ricaduta su di essi perché sono stati forniti insieme al sistema di acquisizione dati. L'accelerometro ICP a taglio ceramico triassiale PCB Piezotronics 356A33 è alimentato da condizionatori di segnale a corrente costante semplici ed economici. Questi sensori sono facili da utilizzare e si interfacciano con strumenti di analisi del segnale, acquisizione dati e registrazione. Le sue principali caratteristiche sono :

- Sensibilità alla tensione fissa
- Segnale di uscita a bassa impedenza
- Funzionamento a due fili
- Funzione intrinseca di autotest



Figura 4.9: Accelerometro triassiale PCB

4.5 Matlab

Matlab è una piattaforma di programmazione e calcolo numerico utilizzata da milioni di ingegneri e scienziati per l'analisi di dati, lo sviluppo di algoritmi e la creazione di modelli.

Matlab combina un ambiente desktop ottimizzato per l'analisi iterativa e i processi di progettazione con un linguaggio di programmazione che esprime le operazioni matematiche con matrici e array in modo diretto. Le caratteristiche principali di Matlab sono:

- **É stato sviluppato professionalmente**, infatti dispone di toolbox rigorosamente testati e interamente documentati.
- **Poi dispone di app interattive**, in effetti le sue applicazioni consentono di vedere come algoritmi differenti lavorano con i dati. Esegue iterazioni finché non vengono ottenuti i risultati desiderati, quindi può generare automaticamente un programma Matlab per riprodurre o automatizzare il lavoro.
- **Infine la scalabilità**, dove si può dimensionare le analisi per eseguirle su cluster, GPU e cloud solo con lievi modifiche al codice. Non è necessario riscrivere il codice o apprendere la programmazione di big data e tecniche out-of-memory.

Le funzionalità principali di Matlab sono :

- **Analisi dati:** ingegneri e scienziati utilizzano Matlab per organizzare, pulire e analizzare serie di dati complesse provenienti da campi diversi come climatologia, manutenzione predittiva, ricerca medica e finanza.
- **Grafica:** Matlab utilizza i grafici integrati per visualizzare i dati, ottenere informazioni e identificare sottostanti pattern e trend.
- **Programmazione:** creazione di script, funzioni e classi.
- **Creazione di App:** App Designer consente di creare applicazioni professionali senza dover essere uno sviluppatore software professionista. Per progettare la propria interfaccia grafica utente (GUI) sarà sufficiente trascinare e rilasciare componenti di visualizzazione, quindi utilizzare l'editor integrato per programmarne rapidamente il comportamento.
- **Utilizzo di Matlab con altri linguaggi:** utilizzo di MATLAB con Python, C/C++, Fortran, Java e altri linguaggi.
- **Hardware:** connessione di Matlab all'hardware.
- **Calcolo parallelo:** esegue calcoli su larga scala utilizzando computer desktop multicore, GPU, cluster, grid e cloud.
- **Distribuzione su desktop e via web:** possono essere condivisi i programmi MATLAB.
- **Cloud computing:** lavora in ambienti cloud da MathWorks Cloud a cloud pubblici, inclusi AWS e Azure.

Matlab è stato usato per la parte di pre-processamento e la creazione del dataset in formato ".mat" e ".txt" e principalmente per tre motivi.

- Il primo è la trasformazione dei dati e la pulizia di essi.
- Il secondo è l'analisi in frequenza.
- Il terzo è l'utilizzo di due applicazioni di Matlab, Diagnostic Feature Design e Classification Learner, dove il primo è stato usato per un'ulteriore pulizia e per il calcolo di alcune caratteristiche importanti; mentre il secondo è stato utilizzato per fare classificazione su queste caratteristiche.

4.5.1 Diagnostic Feature Design

L'applicazione Diagnostic Feature Designer consente di completare la fase di progettazione delle caratteristiche all'interno del flusso di lavoro per la manutenzione predittiva mediante un'interfaccia grafica multifunzione. È possibile progettare e confrontare le caratteristiche in modo interattivo e successivamente determinare quali di esse sono più efficaci nel discriminare tra i dati provenienti da sistemi nominali e da sistemi difettosi. Le caratteristiche più efficaci diventano infine gli indicatori di condizione per la diagnosi dei guasti e la prognosi.

Usando questa applicazione è possibile:

- Importare dati misurati o simulati da file individuali, un file di insieme o un datastore di insiemi che fa riferimento a file esterni all'applicazione.
- Visualizzare interattivamente i dati per tracciare le variabili di insieme importate o calcolate all'interno dell'applicazione. Raggruppare i dati in base all'etichetta di condizione nei grafici in modo da poter vedere chiaramente se i dati provengono da sistemi nominali o difettosi.
- Derivare nuove variabili come segnali con media temporale sincrona. L'applicazione esegue l'elaborazione su tutti i membri dell'insieme con un solo comando.
- Generare caratteristiche dalle variabili e visualizzare la loro efficacia utilizzando istogrammi. Le caratteristiche includono statistiche del segnale, metriche non lineari, metriche per macchinari rotanti e metriche spettrali. È inoltre possibile creare caratteristiche personalizzate.
- Classificare le caratteristiche per determinare quali sono le più adatte a discriminare le differenze comportamentali nei dati.
 - Utilizzare la classificazione supervisionata con caratteristiche etichettate per determinare quali caratteristiche sono con più probabilità in grado di discriminare tra comportamenti nominali e difettosi.
 - Utilizzare la classificazione non supervisionata quando i dati non presentano variabili o etichette di condizione, per determinare quali caratteristiche mostrano il miglior clustering con altre caratteristiche, le quali hanno più probabilità di indicare differenti condizioni di guasto o di esercizio.
 - Utilizzare la classificazione prognostica con caratteristiche estratte da dati relativi a guasti, per determinare quali caratteristiche sono in grado di indicare la vita utile residua (RUL).
- Esportare direttamente le caratteristiche più efficaci in Classification Learner per ottenere una maggiore comprensione dell'efficacia delle caratteristiche e per l'addestramento di particolari algoritmi.
- Generare il codice per le caratteristiche scelte in modo da poter riprodurre, personalizzare e automatizzare i calcoli delle caratteristiche in una funzione Matlab. È possibile generare anche codice e blocchi Simulink che supportano i dati in streaming.

4.5.2 Classification Learner

L'applicazione Classification Learner addestra modelli per classificare dati. Utilizzando questa app, è possibile esplorare il machine learning supervisionato attraverso vari classificatori. È possibile esplorare i dati, selezionare le caratteristiche, specificare schemi di convalida,

addestrare modelli e valutare i risultati. Inoltre, è possibile eseguire l'addestramento automatico per cercare il miglior tipo di modello di classificazione, tra cui alberi decisionali, analisi discriminante, support vector machines, regressione logistica, nearest neighbors, naive Bayes, approssimazione del kernel, ensemble e classificazione con reti neurali.

È possibile eseguire il machine learning supervisionato fornendo un insieme noto di dati di input (osservazioni o esempi) e risposte conosciute ai dati (etichette o classi). Si utilizzano i dati per addestrare un modello che genera previsioni per le risposte a nuovi dati. Per utilizzare il modello con nuovi dati, o per apprendere sulla classificazione programmata, è possibile esportare il modello nello spazio di lavoro o generare codice Matlab per ricreare il modello addestrato.

4.6 Python

Il secondo strumento utilizzato, oltre a Matlab, è Python. Python è un linguaggio di programmazione dinamico orientato agli oggetti utilizzabile per molti tipi di sviluppo software. Offre un forte supporto all'integrazione con altri linguaggi e programmi, è fornito di una estesa libreria standard e può essere appreso in pochi giorni. Python è un linguaggio pseudocompilato, infatti un interprete si occupa di analizzare il codice sorgente (semplici file testuali con estensione ".py") e, se sintatticamente corretto, di eseguirlo. In Python, non esiste una fase di compilazione separata (come avviene in C, per esempio) che generi un file eseguibile partendo dal sorgente.

L'esser pseudointerpretato rende Python un linguaggio portabile. Una volta scritto un sorgente, può essere interpretato ed eseguito sulla gran parte delle piattaforme attualmente utilizzate, siano esse di casa Apple (Mac) che PC (Microsoft Windows e GNU/Linux). Semplicemente, basta la presenza della versione corretta dell'interprete.

Infine, Python è free software: non solo il download dell'interprete per la propria piattaforma, così come l'uso di Python nelle proprie applicazioni, è completamente gratuito; ma oltre a questo Python può essere liberamente modificato e così ridistribuito, secondo le regole di una licenza pienamente open-source.

Principalmente Python è stato utilizzato per classificare i dati grezzi così come vengono ottenuti dagli accelerometri e per fare ciò sono state utilizzate due librerie open source: TensorFlow e Keras.

4.6.1 TensorFlow

Come introdotto da [Antonio Gulli, 2019] TensorFlow è una libreria open-source di machine learning sviluppata da Google. Essa è ampiamente utilizzata per la creazione e l'addestramento di modelli di machine learning, in particolare per reti neurali. TensorFlow fornisce una piattaforma flessibile e efficiente per costruire e sperimentare con modelli di intelligenza artificiale su una vasta gamma di dispositivi, dai server di grandi dimensioni ai dispositivi mobili e dispositivi embedded.

Alcune delle caratteristiche principali di TensorFlow sono:

- **Grafico di Computazione:** TensorFlow rappresenta le operazioni di calcolo come un grafo diretto aciclico (DAG, Directed Acyclic Graph). Questo permette una maggiore flessibilità e ottimizzazione nell'esecuzione di operazioni di calcolo parallele.
- **Tensori:** TensorFlow manipola i dati utilizzando tensori, che sono strutture dati simili a matrici multidimensionali. Questi tensori sono fondamentali per rappresentare i dati in input, i pesi del modello, e gli output.

- **Architettura Flessibile:** si può utilizzare TensorFlow per sviluppare una vasta gamma di modelli di machine learning, compresi modelli per il riconoscimento di immagini, il riconoscimento del linguaggio naturale, la traduzione automatica e altro ancora.
- **Supporto per GPU e TPU:** TensorFlow offre supporto per l'utilizzo di unità di elaborazione grafica (GPU) e unità di elaborazione tensoriale (TPU) per accelerare il processo di addestramento dei modelli.
- **TensorBoard:** TensorFlow include uno strumento di visualizzazione chiamato TensorBoard, che permette di tracciare e visualizzare graficamente le metriche di addestramento, esplorare il grafo di computazione e analizzare le prestazioni del modello.
- **Community attiva:** grazie alla sua popolarità, TensorFlow ha una vasta community di sviluppatori che contribuiscono a migliorare la libreria e forniscono supporto attraverso forum e risorse online.
- **TensorFlow Lite:** una versione di TensorFlow progettata per dispositivi mobili e dispositivi edge, consente l'esecuzione di modelli di machine learning su risorse limitate.

TensorFlow è utilizzato in molte applicazioni del mondo reale e ha contribuito significativamente alla diffusione e all'avanzamento del machine learning e dell'intelligenza artificiale.

4.6.2 Keras

Come descritto da [Antonio Gulli, 2019] Keras è una libreria open-source di machine learning altamente modulare e user-friendly, progettata per semplificare il processo di creazione e sperimentazione di reti neurali. Inizialmente sviluppata come wrapper su Theano e TensorFlow, Keras è diventata parte integrante di TensorFlow a partire dalla versione 2.0. Questa integrazione ha contribuito a consolidare Keras come uno degli strumenti principali per lo sviluppo di modelli di deep learning.

Caratterizzata da una sintassi chiara e intuitiva, Keras è particolarmente apprezzata per la sua facilità d'uso, che la rende accessibile anche ai principianti nel campo del machine learning. Essa offre un'ampia gamma di strati (layers) predefiniti, funzioni di attivazione e ottimizzatori, semplificando notevolmente la progettazione di reti neurali complesse. Inoltre, Keras facilita la gestione delle operazioni comuni nel machine learning, come la suddivisione dei dati, l'addestramento del modello e la valutazione delle prestazioni.

La versatilità di Keras è evidente nella sua capacità di supportare diversi tipi di modelli, tra cui reti neurali sequenziali, modelli funzionali e condivisi. Questa flessibilità consente agli sviluppatori di implementare soluzioni adatte alle specifiche esigenze del problema affrontato.

L'integrazione di Keras con TensorFlow fornisce ulteriori vantaggi, come il supporto per l'accelerazione hardware tramite GPU e TPU, contribuendo così a migliorare l'efficienza computazionale durante le fasi di addestramento e inferenza.

In sintesi, Keras rappresenta una scelta consigliata per gli sviluppatori impegnati nella progettazione e nell'implementazione di modelli di deep learning, grazie alla sua natura modulare, intuitiva e alla stretta integrazione con TensorFlow, una delle librerie di machine learning più ampiamente utilizzate.

In questo capitolo verrà descritto l'implementazione suddividendola in diverse fasi

Come descritto precedentemente nell'introduzione, lo stato dell'implementazione comprenderà al suo interno altre fasi che verranno elencate di seguito e approfondite nel seguente paragrafo.

- Creazione dei fault ed esperimenti eseguiti
- Creazione del dataset
- Analisi in frequenza MATLAB
- Diagnostic Feature Design e Classificazione in MATLAB
- Classificazione in Python

5.1 Creazione dei fault e panoramica degli esperimenti

All'inizio il numero di esperimenti e il numero di fault da eseguire era molto maggiore rispetto a quelli effettivamente eseguiti, ciò è stato dovuto al fatto che dopo un'analisi attenta e per questioni di tempistiche è stato deciso di ridurre il tutto. Alla fine è stato scelto di considerare un unico fault, ovvero quello di un'incisione lineare, il quale è stato applicato sia sull'anello interno, sia sull'anello esterno e sia sul rullo. Gli esperimenti sono stati eseguiti su 5 cuscinetti differenti della marca FAG dove in ognuno di questi era presente o meno il guasto. In totale il numero di esperimenti eseguiti sono 16, quindi 4 esperimenti per classe di guasto, la frequenza di campionamento del sistema di acquisizione è stata di 10kHz e la velocità del motore era di 1000 RPM (giri al minuto).

Le classi di guasto in totale sono 4:

- La prima classe è quella senza guasti dove il comportamento del cuscinetto e quindi dell'intero sistema è in condizioni operativo.
- La seconda classe è quella dove il guasto avviene nell'anello esterno.
- La terza classe è quella dove il fault si trova nell'anello interno.
- L'ultima classe è quella dove il guasto è presente nel rullo.

Nella Tabella 5.1 è possibile osservare in modo sintetico il numero di prove, il guasto, il tipo di guasto correlato con una breve descrizione.

Tipo di prova	Numero di prove	Tipo di guasto	Descrizione
Cuscinetto senza fault	4	No	Singola prova eseguita per 2 minuti
Cuscinetto con fault nell'anello esterno	4	Linea	Singola prova eseguita per 2 minuti
Cuscinetto con fault nell'anello interno	4	Linea	Singola prova eseguita per 2 minuti
Cuscinetto con fault nel rullo	4	Linea	Singola prova eseguita per 2 minuti

Tabella 5.1: Descrizione delle prove eseguite

Inizialmente il primo esperimento che è stato deciso di eseguire è stato quello con i cuscinetti FAG N207 e NU207 privi di guasto per vedere il loro comportamento e trarre le prime osservazioni; in Figura 5.1 e 5.2 si possono osservare i cuscinetti in buono stato.

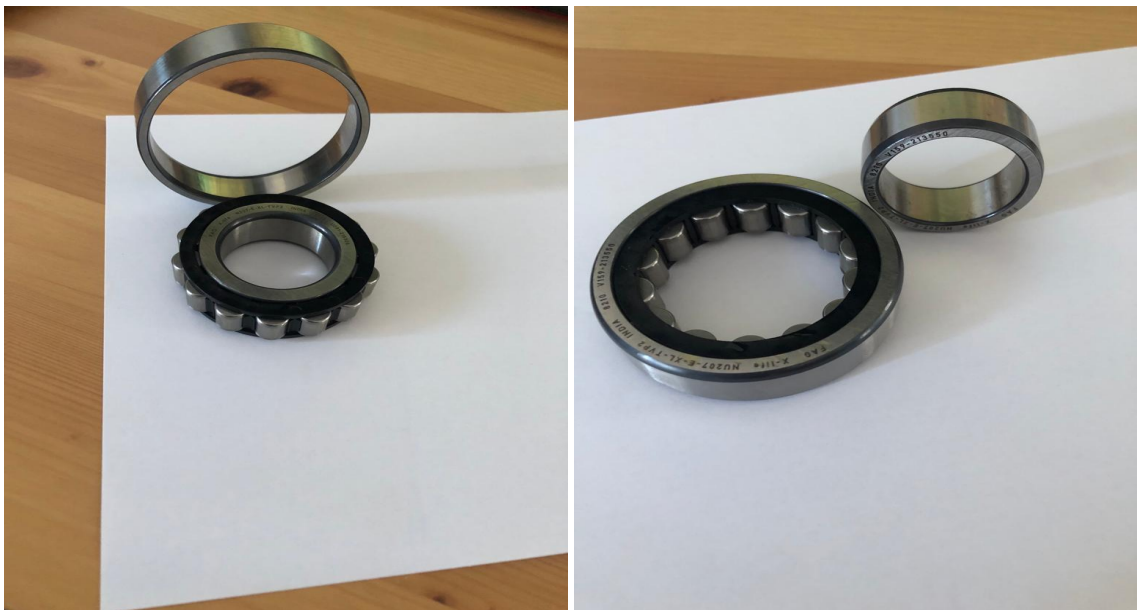


Figura 5.1: Cuscinetto FAG N207 senza guasti **Figura 5.2:** Cuscinetto FAG NU207 senza guasti

Successivamente sono partite le prime prove dove è stato deciso di applicare un guasto, più precisamente un'incisione lineare nell'anello esterno come mostrato in Figura 5.3. Sono stati eseguiti 4 esperimenti dalla durata di 2 minuti come citato nella Tabella 5.1. Un'osservazione ottenuta da questa prima prova è stata che il danno generato nell'anello esterno si è ripercosso in tutto il cuscinetto andando a creare danni anche nell'anello interno e nei rulli, ciò ha reso l'analisi vibrazionale ancora più complessa di come ci si potesse aspettare. Inoltre, questa rilevazione è stata riscontrata in seguito quando sono state effettuate le altre prove per il guasto creato nell'anello interno e nel rullo.

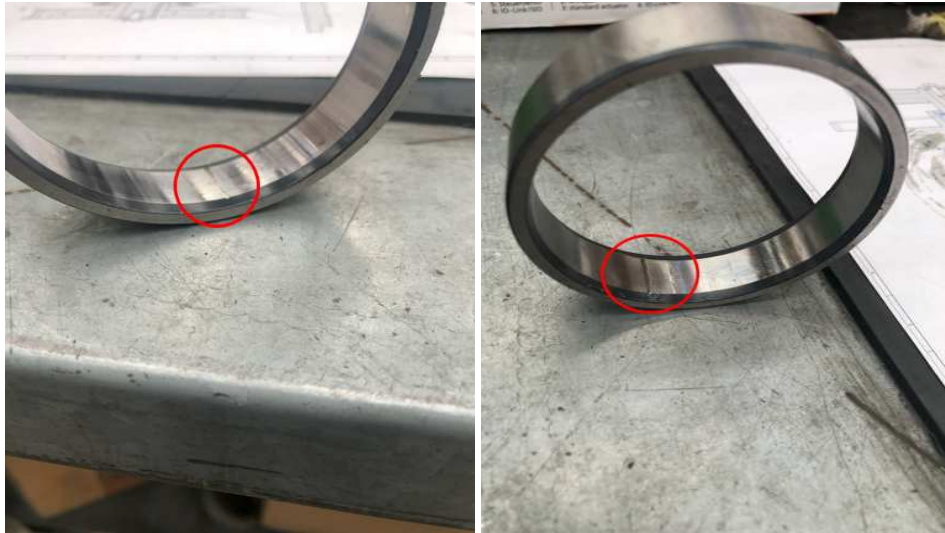


Figura 5.3: Presenza del fault nell'anello esterno

In seguito sono stati creati i fault in un altro cuscinetto nell'anello interno per andare ad osservare il suo andamento; il tipo di guasto è sempre un'incisione lineare e la durata è sempre di 2 minuti per un totale di 4 prove. Il danno applicato nel cuscinetto, precisamente nell'anello interno si può osservare nella Figura 5.4.



Figura 5.4: Presenza del fault nell'anello interno

Dopo è stato preso un cuscinetto nuovo, è stato indifferente quale si prendesse tra il FAG N207 e il FAG NU207 in quanto complementari, ed è stato applicato un guasto, come negli altri casi un'incisione lineare. In seguito è stata effettuata la prova sempre per il numero di volte descritto precedentemente sempre della stessa durata. Il guasto al rullo lo si può osservare nella Figura 5.5.



Figura 5.5: Presenza del fault nel rullo

Prima di effettuare gli esperimenti dei cuscinetti precedentemente mostrati, un ruolo molto importante ha avuto la scelta del posizionamento dei due accelerometri piezoelettrici PCB. Infatti, il loro posizionamento è stato fondamentale perché più sono vicini al cuscinetto nel canotto e più precise sono le misure; infatti come citato precedentemente nel Capitolo 4, il canotto è stato modificato appositamente per fare in modo che uno dei due accelerometri stesse il più vicino possibile al cuscinetto. La scelta del loro posizionamento, la si può notare in Figura 5.6 dove il colore rosso rappresenta dove sono stati posati gli accelerometri piezoelettrici PCB, mentre il colore verde rappresenta gli accelerometri monoassiali di SKF utilizzati come sarà descritto in seguito per un confronto in frequenza.

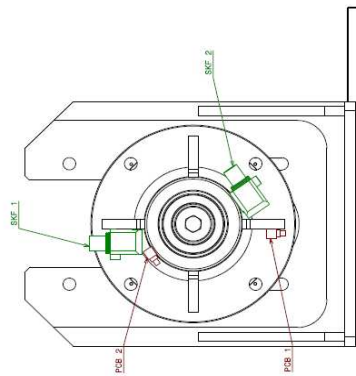


Figura 5.6: Disegno posizionamento accelerometri piezoelettrici PCB

Nella Figura 5.7 si può esaminare il posizionamento dei sensori nel banco da prova effettivo.



Figura 5.7: Posizionamento dei sensori nel banco da prova

5.2 Creazione del dataset

Un nodo molto importante è stato la creazione del dataset, infatti tutti gli esperimenti effettuati venivano salvati nel NAS del sistema di acquisizione in un formato "tdms" (Technical Data Management Streaming).

Il formato di file tdms è uno standard di file sviluppato da National Instruments per la gestione di dati tecnici e scientifici. Tdms è progettato per consentire la memorizzazione di dati acquisiti da strumenti di misura, dispositivi di acquisizione dati e altri sistemi di monitoraggio in un formato strutturato e efficiente. Questo formato di file è particolarmente adatto per la gestione di dati temporali, poiché supporta la memorizzazione di dati a più canali con informazioni temporali associate.

I file tdms contengono dati organizzati in gerarchie di gruppi e canali, consentendo una facile navigazione e interpretazione dei dati. Oltre alla memorizzazione dei dati stessi, i file tdms possono includere metadati descrittivi che forniscono informazioni aggiuntive sul contesto e sulla provenienza dei dati registrati.

L'utilizzo diffuso di tdms nel contesto degli strumenti e delle applicazioni di acquisizione dati industriali è testimone della sua affidabilità e flessibilità come formato di file per la gestione avanzata di dati tecnici complessi.

Nella Figura 5.8 si può osservare come viene visualizzato un formato tdms con Excel, il quale è composto da:

- **Root Name:** rappresenta il nome con cui identificare il file tdms.
- **Group:** un gruppo è una struttura che contiene uno o più canali e può anche includere altri gruppi. All'interno di un gruppo, è possibile definire ulteriori sottogruppi o canali, creando una gerarchia che riflette la struttura dei dati registrati.
- **Channels:** un canale rappresenta una singola sequenza di dati, tipicamente associata a una grandezza fisica o a una variabile specifica. I canali in questo caso sono due, uno per ogni accelerometro e ogni canale è composto a sua volta da:

- **Channel:** identifica il nome del canale e può essere impostato dal software già installato nel sistema di acquisizione.
- **Datatype:** il tipo di dati.
- **Length:** indica la lunghezza del canale, probabilmente la dimensione del vettore di dati.
- **Minimum:** definisce i valori minimi che possono essere assunti dai dati del canale.
- **Maximum:** definisce i valori massimi che possono essere assunti dai dati del canale.
- **Description:** fornisce una descrizione del canale, spiegando la natura o la funzione della grandezza misurata.
- **Wf_increment:** viene utilizzato per incrementare il tempo di 0.0001 secondi.
- **Wf_start_time:** data in cui inizia la misurazione precisando sia il tempo che il giorno in cui è stata eseguita.
- **Start Index:** rappresenta l'indice iniziale dei dati all'interno del file tdms.

Root Name	Title	Author	Date/Time	Groups	Description						
Accelerometers				1							
Group	Channels	Description									
Channels	2										
Channel	Datatype	Unit	Length	Minimum	Maximum	Description	wf_increment	wf_start_time	Start Index		
YC []	DT_FLOAT		1293000				0,0001	28/07/2023 12:30:08,013 PM	1		
ZA []	DT_FLOAT		1293000				0,0001	28/07/2023 12:30:08,013 PM	1		

Figura 5.8: Visualizzazione formato tdms in Excel

Quindi una volta ottenuti i file tdms dei 16 esperimenti è stato deciso di unificare questi file in modo da costruire il dataset sia in un formato *“.mat”* utilizzato appunto in Matlab sia in un formato *“.txt”* utilizzato in Python. In entrambi i formati il dataset è composto da diverse colonne di cui ognuna contiene un'informazione fondamentale.

Il dataset utilizzato per Matlab è stato creato in un formato tabella ed è composto da 12 colonne e 16 righe dove ogni riga identifica un esperimento ed è ogni colonna è composta da uno dei seguenti parametri:

- **Yc:** rappresenta un formato di dato *“Timetable”* che contiene al suo interno due valori, uno è il tempo e l'altro è il valore effettivo dell'accelerometro; Yc viene utilizzato per indicare l'asse di misurazione dell'accelerometro triassiale posto nella posizione più vicina al cuscinetto nel canotto. Viene utilizzata solo una misura delle 3 ottenute dall'accelerometro perchè le altre è come se non misurassero niente, infatti viene collegato al sistema di acquisizione solo il cavo che riguarda l'asse Y.
- **Za:** come Yc, rappresenta un formato di dato *“Timetable”* che contiene al suo interno due valori, uno è il tempo e l'altro è il valore effettivo dell'accelerometro; Za viene utilizzato per indicare l'asse di misurazione dell'accelerometro triassiale posto nella posizione più lontana al cuscinetto nel canotto. Viene utilizzata solo una misura delle 3 ottenute dall'accelerometro perchè le altre è come se non misurassero niente, infatti viene collegato al sistema di acquisizione solo il cavo che riguarda l'asse Z.
- **Id_esperimento:** viene utilizzato per riconoscere quale esperimento è stato eseguito ed è composto da una lettera (indica il no-fault, il fault anello esterno, interno o rullo) e un numero (indica il numero di esperimento di una classe di guasto che può andare da 1 a 4).

- **FA**: frequenza calcolata del motore quando il cuscinetto non presenta guasti.
- **BPFO**: frequenza calcolata del motore quando il cuscinetto presenta un guasto nell'anello esterno.
- **BPFI**: frequenza calcolata del motore quando il cuscinetto presenta un guasto nell'anello interno.
- **BSF**: frequenza calcolata del motore quando il cuscinetto presenta un guasto nel rullo.
- **RPM**: questo valore rappresenta i giri al minuto che compie il canotto (è quindi l'albero motore) durante la prova, infatti il valore è stato deciso che rimanesse invariato (stazionario) per tutte le prove, per non complicare l'analisi in frequenza.
- **Fs**: frequenza di campionamento.
- **Wr_increment**: incremento temporale.
- **Cod_Fault**: identificativo del fault che va da 0 a 3:
 - 0: Indica la classe senza guasto.
 - 1: Indica la classe di guasto dell'anello esterno.
 - 2: Indica la classe di guasto dell'anello interno.
 - 3: Indica la classe di guasto del rullo.
- **Pos_Fault**: descrizione del fault, simile al "Cod_Fault" sopra citato.

Tutti questi parametri descritti si possono riscontrare nella Figura 5.9.

	1	2	3	4	5	6	7	8	9	10	11	12
	Yc	Za	Id_esperimento	FA	BPFO	BPFI	BSF	RPM	Fs	Wr_increment	Cod_Fault	Pos_Fault
1	50000x1 ti...	50000x1 ti...	"N_1"	16.6667	93.3333	140	87.9960	1000	10000	1.0000e-04	"0"	"No_fault"
2	50000x1 ti...	50000x1 ti...	"N_2"	16.6667	93.3333	140	87.9960	1000	10000	1.0000e-04	"0"	"No_fault"
3	50000x1 ti...	50000x1 ti...	"N_3"	16.6667	93.3333	140	87.9960	1000	10000	1.0000e-04	"0"	"No_fault"
4	50000x1 ti...	50000x1 ti...	"N_4"	16.6667	93.3333	140	87.9960	1000	10000	1.0000e-04	"0"	"No_fault"
5	50000x1 ti...	50000x1 ti...	"E_1"	16.6667	93.3333	140	87.9960	1000	10000	1.0000e-04	"1"	"Fault_anell..."
6	50000x1 ti...	50000x1 ti...	"E_2"	16.6667	93.3333	140	87.9960	1000	10000	1.0000e-04	"1"	"Fault_anell..."
7	50000x1 ti...	50000x1 ti...	"E_3"	16.6667	93.3333	140	87.9960	1000	10000	1.0000e-04	"1"	"Fault_anell..."
8	50000x1 ti...	50000x1 ti...	"E_4"	16.6667	93.3333	140	87.9960	1000	10000	1.0000e-04	"1"	"Fault_anell..."
9	50000x1 ti...	50000x1 ti...	"I_1"	16.6667	93.3333	140	87.9960	1000	10000	1.0000e-04	"2"	"Fault_anell..."
10	50000x1 ti...	50000x1 ti...	"I_2"	16.6667	93.3333	140	87.9960	1000	10000	1.0000e-04	"2"	"Fault_anell..."
11	50000x1 ti...	50000x1 ti...	"I_3"	16.6667	93.3333	140	87.9960	1000	10000	1.0000e-04	"2"	"Fault_anell..."
12	50000x1 ti...	50000x1 ti...	"I_4"	16.6667	93.3333	140	87.9960	1000	10000	1.0000e-04	"2"	"Fault_anell..."
13	50000x1 ti...	50000x1 ti...	"R_1"	16.6667	93.3333	140	87.9960	1000	10000	1.0000e-04	"3"	"Fault_rullo"
14	50000x1 ti...	50000x1 ti...	"R_2"	16.6667	93.3333	140	87.9960	1000	10000	1.0000e-04	"3"	"Fault_rullo"
15	50000x1 ti...	50000x1 ti...	"R_3"	16.6667	93.3333	140	87.9960	1000	10000	1.0000e-04	"3"	"Fault_rullo"
16	50000x1 ti...	50000x1 ti...	"R_4"	16.6667	93.3333	140	87.9960	1000	10000	1.0000e-04	"3"	"Fault_rullo"

Figura 5.9: Dataset formato ".mat" utilizzato per Matlab

Mentre il dataset utilizzato per Python è composto da un formato ".txt" ed ha alcune piccole differenze con il formato ".mat", infatti qui non si hanno più le Timetable, ma il valore del tempo e dell'accelerometro sono divisi, ed inoltre si hanno alcuni campi in più rispetto al precedente dataset:

- **Time**: Indica l'ora e la data di inizio dell'esperimento.
- **Tempo**: Indica il tempo dell'esperimento in secondi partendo da 0.
- **Num_esperimento**: indica il numero dell'esperimento che può andare a 1 a 16.

Nella Figura 5.10 si può osservare il formato ".Txt".

Time	Tempo	Yc	Za	Id_esperimento	Num_esperimento	FA	BPFO	BPFI	BSF	RPM	Fs	Wr_increment	Cod_fault	Pos Fault
2023-08-03 09:42:02.093816757	0.0001 sec	0.121094	-0.546875	N_1	1	16.666667	93.333333	140	87.995971	1000	10000	0.0001	0	No_fault
2023-08-03 09:42:02.093916757	0.0002 sec	0.186523	0.074219	N_1	1	16.666667	93.333333	140	87.995971	1000	10000	0.0001	0	No_fault
2023-08-03 09:42:02.094016757	0.0003 sec	0.233398	0.112305	N_1	1	16.666667	93.333333	140	87.995971	1000	10000	0.0001	0	No_fault
2023-08-03 09:42:02.094116757	0.0004 sec	0.235352	0.746094	N_1	1	16.666667	93.333333	140	87.995971	1000	10000	0.0001	0	No_fault
2023-08-03 09:42:02.094216757	0.0005 sec	0.194336	-0.884766	N_1	1	16.666667	93.333333	140	87.995971	1000	10000	0.0001	0	No_fault
...
2023-08-03 16:30:50.378782474	4.9996 sec	0.181641	0.151367	R_4	16	16.666667	93.333333	140	87.995971	1000	10000	0.0001	3	Fault_rullo
2023-08-03 16:30:50.378882474	4.9997 sec	-1.280273	-1.682617	R_4	16	16.666667	93.333333	140	87.995971	1000	10000	0.0001	3	Fault_rullo
2023-08-03 16:30:50.378982474	4.9998 sec	-1.877930	-2.551758	R_4	16	16.666667	93.333333	140	87.995971	1000	10000	0.0001	3	Fault_rullo
2023-08-03 16:30:50.379082474	4.9999 sec	-0.164062	-0.414062	R_4	16	16.666667	93.333333	140	87.995971	1000	10000	0.0001	3	Fault_rullo
2023-08-03 16:30:50.379182474	5 sec	-1.096680	-1.700195	R_4	16	16.666667	93.333333	140	87.995971	1000	10000	0.0001	3	Fault_rullo

Figura 5.10: Dataset formato ".txt" utilizzato per Python

Una volta che sono stati costruiti i due dataset è stato possibile passare alla fase successiva, cioè quella in cui vengono utilizzati dei filtri per fare analisi in frequenza in Matlab; inoltre i due dataset sono stati ridotti per questione di tempo di esecuzione dei programmi creati.

5.3 Analisi in frequenza Matlab

L'analisi in frequenza non è stata fatta solo su Matlab, ma anche in Python, poiché i risultati sono uguali si riporta quella eseguita in Matlab per una lettura migliore.

Innanzitutto sono state calcolate differenti frequenze a partire dai parametri che si trovano nella Tabella 4.1 e dalle Formule 3.4.1-3.4.7:

$$FA = \frac{RPM}{60} = 16.66Hz \quad (5.3.1)$$

$$BPFO = \frac{0.4 \cdot RPM \cdot N_R}{60} = 93.33Hz \quad (5.3.2)$$

$$BPFI = \frac{0.6 \cdot RPM \cdot N_R}{60} = 140Hz \quad (5.3.3)$$

$$BSF = \frac{RPM}{60} \cdot \left(\frac{D_1 + D_2}{B_D} \right) \cdot \left[1 - \left(\frac{B_D}{\frac{D_1 + D_2}{2}} \right)^2 \right] = 87.99Hz \quad (5.3.4)$$

Dove RPM rappresenta i giri al minuto del motore che sono 1000. Queste sono le 4 frequenze che servono per fare analisi in frequenza; la frequenza associato al guasto nella gabbia del cuscinetto non è stata usata perché è un evento raro e quindi è stato trascurato.

Successivamente una volta ottenute queste frequenze, all'interno del dataset è stato applicato un filtro passa basso per eliminare in parte il rumore di fondo situato nell'ambiente dove sono state eseguite le varie prove. Un filtro passa basso è un dispositivo o un algoritmo che consente al passaggio di segnali a frequenza bassa mentre attenua le frequenze più elevate; questo tipo di filtro è comunemente utilizzato in elettronica, telecomunicazioni, elaborazione del segnale e altre discipline per rimuovere o ridurre le componenti ad alta frequenza da un segnale. Il filtro utilizzato lo si può osservare nel codice sottostante:

```
Yc=lowpass(double(Tabella_temporale.Yc),5000,fs);
Za=lowpass(double(Tabella_temporale.Za),5000,fs);
```


Il filtro è stato applicato a tutte e due le misure ottenute dai due accelerometri, al filtro vengono date le misure, 5000Hz indica la frequenza oltre la quale vogliamo che le frequenze vengano attenuate e fs indica la frequenza di campionamento. Successivamente è stata creata una funzione per implementare la Fast Fourier Transform (FFT) per passare dal dominio del tempo al dominio in frequenza. Per applicarla si deve verificare se la lunghezza della misura è una potenza di 2, in caso non fosse si procede troncando il vettore per fare in modo che diventi una potenza di 2.

```
function [f,P,t] = fft_teorico(y,fs)

% Trovare la lunghezza del segnale
L = numel(y);
% Si controlla che la lunghezza del segnale è pari in caso
% si fa in modo che esso lo diventi
if rem(L,2)~=0
y = y(1:end-1);
L = L-1;
end

% Si applica la fft
Y = fft(y);

% Calcola lo spettro bidirezionale P2
P2 = abs(Y/L);
% Calcola lo spettro singolo P
P = P2(1:L/2+1);
P(2:end-1) = 2*P(2:end-1);

% Definisci il dominio delle frequenze f e plotta
% lo spettro singolo, P
f = fs*(0:(L/2))/L;

% Plotta il segnale nel dominio del tempo
t = (0:L-1)*1/fs;
```

La funzione prende in ingresso il vettore delle misure a cui si vuole applicare la FFT e la frequenza di campionamento, e restituisce il dominio del tempo, lo spettro singolo e il dominio delle frequenze. Successivamente la funzione viene richiamata più volte nel Main per calcolare la FFT dei 16 esperimenti eseguiti e quindi si avrà una FFT per ogni esperimento, in seguito i grafici in frequenza vengono confrontati tra loro per vedere se si riscontra una corrispondenza con i valori delle frequenze dei fault calcolati precedentemente. Da questa prima analisi è venuto fuori che si riesce ad individuare il fault ma non a classificarlo; questo verrà approfondito nel Capitolo 6, per ciò sono stati utilizzati due strumenti di Matlab, cioè il Diagnostic Feature Design e il Classification Learner, per classificare gli esperimenti pulendo ulteriormente i dati e calcolando alcune caratteristiche principali ed eseguire la classificazione su di essi.

5.3.1 Diagnostic Feature Designer e Classification Learner in Matlab

Il dataset ridotto è stato caricato nell'applicazione Diagnostic Feature Designer attraverso la seguente procedura:

- Si carica il dataset all'interno dell'ambiente di lavoro Matlab.
- Successivamente su New Session si importa il dataset nell'applicazione.
- Si selezionano le variabili che saranno utilizzate per calcolare le feature e la variabile per la classificazione.

Un esempio di interfaccia dell'applicazione e selezione di variabili lo si può osservare in Figura 5.11.

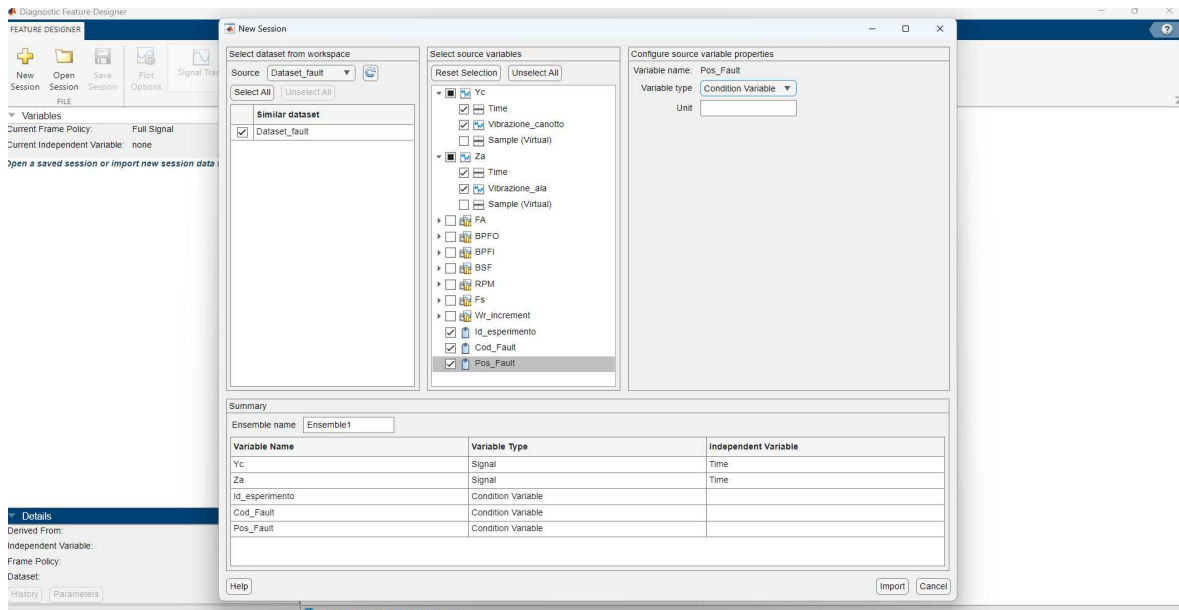


Figura 5.11: Interfaccia e selezione variabili per il Diagnostic Feature Designer

Prima di plottare il segnale è stato deciso di attuare una frame policy, altrimenti il numero delle feature che sono state calcolate non erano abbastanza e ciò rendeva difficile la classificazione successivamente. Si specifica una politica del frame quando si desidera eseguire l'elaborazione dei dati su segmenti sequenziali di un segnale anziché sull'intero segnale in una sola volta. Una frame policy è costituita da una dimensione del frame e da una frequenza del frame. La dimensione del frame è l'intervallo durante il quale vengono raccolti i dati di esso. La frequenza del frame è l'intervallo di tempo tra gli inizi dei frame. La frame policy scelta è stata di 0.2048s. La politica del frame si può osservare in Figura 5.12.

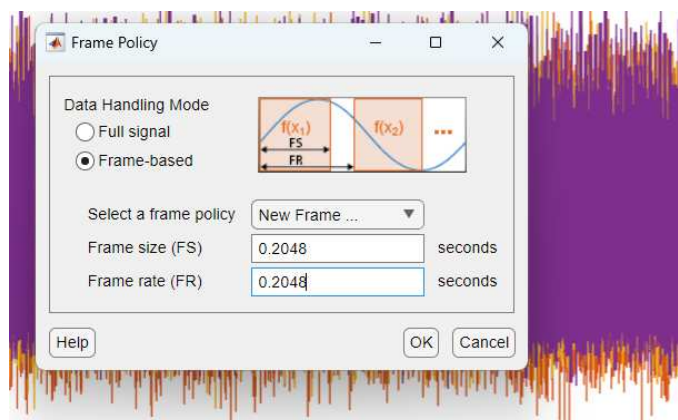


Figura 5.12: Frame policy

Una volta scelta la politica del frame sono stati plottati tutti i 16 esperimenti e raggruppati per "Pos_Fault" e successivamente sono state eseguite delle azioni su questi dati:

- La prima azione ad essere attuata è stata quella di andare a eseguire un' analisi in frequenza pure qui, attraverso "Spectral Estimation" e successivamente "Welch's Method" che è un metodo, il quale rappresenta una tecnica avanzata nell'ambito dell'analisi spettrale, progettata per gestire efficacemente la varianza nei dati. A differenza della tradizionale trasformata di Fourier, il metodo di Welch suddivide il segnale in segmenti sovrapposti, calcolando poi la potenza spettrale di ciascun segmento. Questo approccio consente una maggiore risoluzione in frequenza, rendendo il metodo di Welch particolarmente adatto per segnali non stazionari o soggetti a variazioni nel tempo.
- Successivamente sono state calcolate attraverso "Signal Feature" alcune caratteristiche importanti per fare classificazione, queste sono:
 - Media
 - RMS
 - Deviazione standard
 - Fattore di forma (Shape Factor)
 - Kurtosi (Kurtosis)
 - Assimetria (Skewness)
 - Fattore di cresta (Crest Factor)
 - Fattore di impulso (Impulse Factor)
 - Fattore di sicurezza (Clearance Factor)
 - Valori di picco (Peak Value)
- Dopo è stata applicata l'azione di classificare le caratteristiche per importanza con la tecnica "One way ANOVA"(analisi unidirezionale della varianza); è una tecnica statistica utilizzata per confrontare le medie di tre o più gruppi indipendenti al fine di determinare se esistono differenze significative tra di essi. Questa analisi valuta se la variazione tra le medie dei gruppi è superiore alla variazione all'interno di ciascun gruppo, basandosi sulla presunzione che i dati seguano una distribuzione normale e abbiano varianze omogenee.
- Attraverso l'utilizzo dell' ANOVA per osservare le caratteristiche più importanti, quest'ultime sono state esportate nell'altra applicazione di Matlab, Classification Learner, per attuare la classificazione vera e propria. Per la scelta delle feature da classificare vedere Figura 5.13.

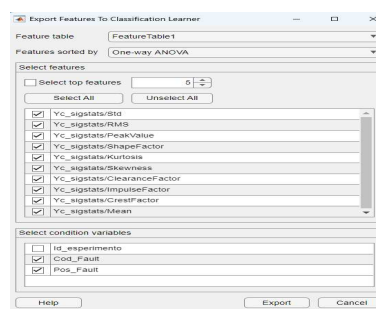


Figura 5.13: Principali caratteristiche esportate

- Infine il processo è stato riprodotto nel medesimo modo, ma attuando prima un ulteriore pulizia dei dati con il metodo TSA (Time-Synchronous averaging (TSA)). L'obiettivo principale del TSA è migliorare il rapporto segnale-rumore di un segnale periodico, rendendo più evidenti i componenti periodici e riducendo l'effetto del rumore non correlato. Questa tecnica è spesso impiegata in contesti in cui si desidera analizzare segnali che si ripetono ciclicamente nel tempo. Funzionamento del TSA:
 - **Sincronizzazione temporale:** si identifica un ciclo temporale nel segnale, come un periodo di oscillazione o un evento ricorrente.
 - **Divisione in segmenti:** il segnale viene diviso in segmenti, o "finestre," in corrispondenza dei cicli temporali identificati.
 - **Allineamento temporale:** i segmenti vengono allineati temporalmente in modo che gli eventi ricorrenti si sovrappongano.
 - **Averaging:** i segmenti vengono sommati o mediati insieme. Poiché gli eventi ricorrenti sono allineati, le componenti periodiche del segnale si sommano, mentre il rumore non correlato si media verso zero.
 - **Risultati:** l'output finale è un segnale "migliorato", le cui componenti periodiche sono amplificate rispetto al rumore non correlato.

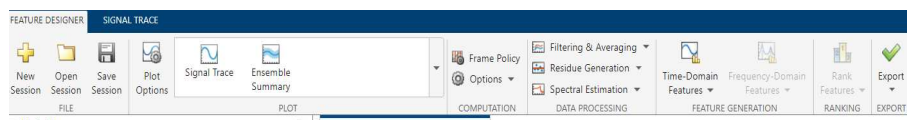


Figura 5.14: Interfaccia Diagnostic Feature Designer dove effettuare diverse azioni

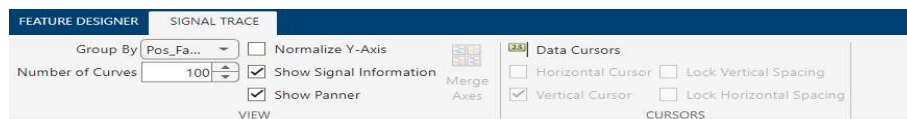


Figura 5.15: Interfaccia Diagnostic Feature Designer per il signal trace

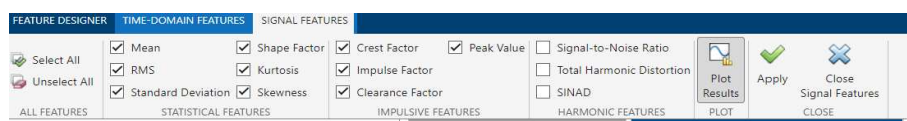


Figura 5.16: Calcolo delle principali feature

Una volta che le variabili sono state esportate nel Classification Learner è necessario aprire una nuova sessione ed è necessario selezionare alcune variabili importanti come:

- **Data Set Variable:** rappresenta il dataset.
- **Response:** rappresenta la variabile per indicare le classi in questo caso sono 4.
- **Predictors:** possono essere scelte le variabili attraverso le quali si vuole effettuare la predizione per la classificazione.
- **Validation Scheme:** lasciato di default cross-validation.
- **Test:** attraverso il quale è possibile lasciare da parte una percentuale del dataset per testare la classificazione.

Negli esperimenti eseguiti sono state provate molti tipi di divisione dei dati di train e di test per osservare come si comportavano gli algoritmi. Principalmente verranno riportati nel Capitolo 6 i risultati ottenuti con il 80% di train e il 20% di test.

Una volta definite tutti questi parametri come in Figura 5.17 si può iniziare la sessione vera e propria.

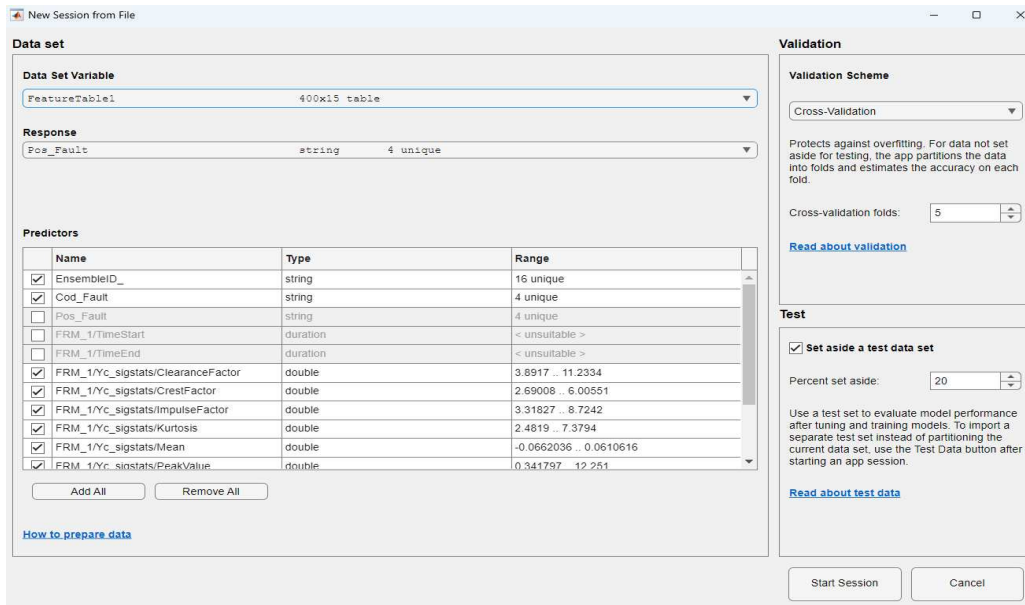


Figura 5.17: Esempio di creazione nuova sessione Classification Learner

E' stato allenato tutto il dataset delle feature con l'opzione "Train All", vedere Figura 5.18, mentre osservare Figura 5.19 dove vengono utilizzate varie tecniche di classificazione, come:

- **Alberi Decisionali:** è uno strumento che partendo dal dataset iniziale, facendo domande sugli attributi (sintomi), si sceglie una direzione in cui muoversi fino ad arrivare alla decisione finale, cioè la classe di guasto (foglia). La biforcazione dell'albero dipende dall'algoritmo che esso implementa.
- **Discriminant Analysis:** si presuppone che classi diverse generino dati basati su distribuzioni gaussiane diverse. Per addestrare (creare) un classificatore, la funzione di adattamento stima i parametri di una distribuzione gaussiana per ciascuna classe. Per prevedere le classi di nuovi dati, il classificatore addestrato trova la classe con il costo di classificazione errata più basso. L'analisi discriminante lineare è nota anche come discriminante di Fisher, dal nome del suo inventore.
- **Logistic Regression Classifier:** è un modello di regressione utilizzato per la classificazione di dati binari, ovvero situazioni in cui la variabile di risposta è di tipo binario (ad esempio, 0 o 1, vero o falso, positivo o negativo). Nonostante il nome, la regressione logistica è comunemente utilizzata per problemi di classificazione anziché per problemi di regressione.
- **Naive Bayes Classifier:** è un tipo di classificatore probabilistico basato sul teorema di Bayes, da cui deriva il nome. È "naive" (ingenuo) perché assume indipendenza condizionale tra i predittori, il che significa che la presenza di una caratteristica in una classe non è influenzata dalla presenza di altre caratteristiche. Il Naive Bayes Classifier è spesso utilizzato per problemi di classificazione, in cui l'obiettivo è assegnare un'istanza a una delle categorie predefinite. Questo modello si basa su principi probabilistici e

calcola la probabilità di appartenenza a una classe dato un insieme di caratteristiche osservate.

- **Support Vector Machines (SVM):** è un tipo di algoritmo di apprendimento automatico utilizzato sia per problemi di classificazione che per problemi di regressione. L'obiettivo principale delle SVM è trovare un iperpiano (una superficie decisionale) che massimizzi la separazione tra diverse classi.
- **Nearest Neighbor Classifier:** il Classificatore del Vicino più Prossimo è un tipo di algoritmo di classificazione nell'ambito dell'apprendimento automatico. La sua idea fondamentale è che gli oggetti che sono simili dovrebbero appartenere alla stessa classe o categoria. Questo classificatore è spesso chiamato anche "k-Nearest Neighbors" (k-NN), dove "k" rappresenta il numero di vicini più prossimi considerati.
- **Kernel Approximation:** è una tecnica utilizzata nelle Support Vector Machines (SVM) e in altri algoritmi di apprendimento automatico basati sul kernel. L'obiettivo di questa tecnica è ridurre la complessità computazionale associata al calcolo del prodotto scalare nello spazio delle caratteristiche indotto dal kernel, consentendo un addestramento più efficiente.
- **Ensemble Classifier:** è un tipo di modello di machine learning che combina le previsioni di diversi modelli più semplici, noti come "base learners" (modelli di base), al fine di migliorare le prestazioni complessive del sistema. L'obiettivo è sfruttare la diversità e la complementarità dei singoli modelli di base per ottenere una previsione più accurata e robusta rispetto a ciascun modello preso singolarmente.
- **Neural Network Classifier:** questo classificatore è un tipo di modello di machine learning basato su reti neurali artificiali. Le reti neurali sono modelli ispirati al funzionamento del cervello umano e sono particolarmente adatte per apprendere modelli complessi e non lineari dai dati. Questi classificatori possono essere utilizzati per affrontare una vasta gamma di problemi di classificazione, da quelli relativamente semplici a quelli più complessi e multidimensionali. L'architettura, la dimensione e altri parametri della rete neurale possono essere adattati in base alle specifiche esigenze del problema. Le reti neurali profonde (deep neural networks o deep learning) hanno dimostrato particolare successo in molte applicazioni, spesso superando altri approcci in problemi complessi.

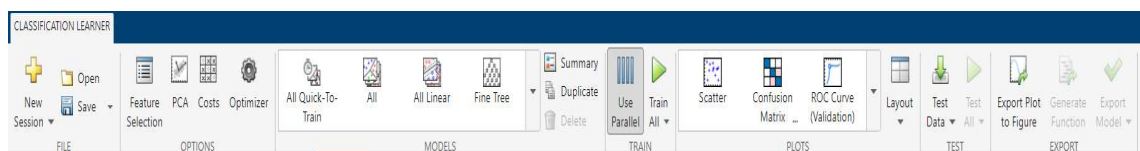


Figura 5.18: Menù per eseguire diverse azioni per la classificazione

1 Tree
Last change: Fine Tree
2.1 Tree
Last change: Fine Tree
2.2 Tree
Last change: Medium Tree
2.3 Tree
Last change: Coarse Tree
2.4 Naive Bayes
Last change: Gaussian Naive Bayes
2.5 Naive Bayes
Last change: Kernel Naive Bayes
2.6 SVM
Last change: Linear SVM
2.7 SVM
Last change: Quadratic SVM
2.8 SVM
Last change: Cubic SVM
2.9 SVM
Last change: Fine Gaussian SVM
2.10 SVM
Last change: Medium Gaussian SVM
2.11 SVM
Last change: Coarse Gaussian SVM
2.12 Ensemble
Last change: Boosted Trees
2.13 Ensemble
Last change: Bagged Trees
2.14 Ensemble
Last change: RUSBoosted Trees
2.15 Neural Network
Last change: Narrow Neural Network
2.16 Neural Network
Last change: Medium Neural Network
2.17 Neural Network
Last change: Wide Neural Network
2.18 Neural Network
Last change: Bilayered Neural Network
2.19 Neural Network
Last change: Trilayered Neural Network
2.20 Kernel
Last change: SVM Kernel
2.21 Kernel
Last change: Logistic Regression Kernel

Figura 5.19: Tipi di tecniche di classificazione

Una volta allenato e testato il dataset è possibile eseguire altre azioni per osservare la correlazione che ci può essere tra i dati in modo grafico, come per esempio:

- **Scatter plot:** è un tipo di grafico che rappresenta dati bidimensionali su un piano cartesiano. In uno scatter plot, i dati sono visualizzati come una serie di punti, ognuno dei quali rappresenta una coppia di valori. L'asse orizzontale (asse delle x) rappresenta una variabile, mentre l'asse verticale (asse delle y) rappresenta un'altra variabile. Questo tipo di grafico è spesso utilizzato per mostrare la relazione o il modello di dispersione tra due variabili e per identificare eventuali pattern o tendenze nei dati.

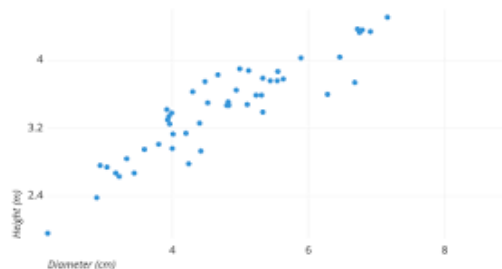


Figura 5.20: Esempio di scatter plot

- **Matrice di confusione:** è una tabella utilizzata per valutare le prestazioni di un modello di classificazione. Questa matrice confronta le predizioni fatte dal modello con le classi reali delle osservazioni nel dataset. La matrice è organizzata in modo che ogni riga rappresenti la classe reale, mentre ogni colonna rappresenta la classe prevista dal modello. Le quattro caselle principali di una matrice di confusione sono:
 - **True Positive (TP):** Osservazioni correttamente classificate come positive.

- **True Negative (TN):** Osservazioni correttamente classificate come negative.
- **False Positive (FP):** Osservazioni erroneamente classificate come positive (falsi positivi).
- **False Negative (FN):** Osservazioni erroneamente classificate come negative (falsi negativi).

Questi valori possono essere utilizzati per calcolare diverse metriche di valutazione delle prestazioni del modello, come l'accuratezza, la precisione, il richiamo (recall), la F1-score, ecc. La matrice di confusione è particolarmente utile quando si lavora con problemi di classificazione multiclasse in cui ci sono più di due classi ed è possibile notare se c'è correlazione tra le diverse classi, ciò si può osservare dalla diagonale della matrice di confusione dove i valori, di solito, sono più elevati rispetto alle altre parti della matrice.

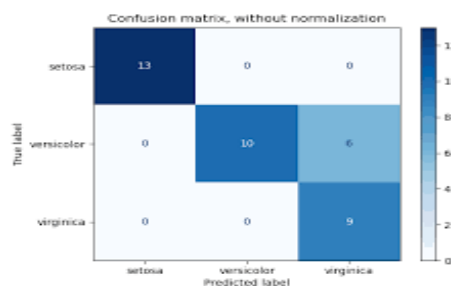


Figura 5.21: Esempio di matrice di confusione

- **Curva ROC:** la curva ROC (Receiver Operating Characteristic) è un grafico che mostra la performance di un modello di classificazione variando la soglia di decisione. La curva ROC è costruita tracciando il tasso di veri positivi (True Positive Rate, TPR) al variare del tasso di falsi positivi (False Positive Rate, FPR) per diversi valori di soglia di decisione. Idealmente, un modello di classificazione con prestazioni eccellenti avrà una curva ROC che si sposterà rapidamente verso l'angolo in alto a sinistra del grafico, il che indica un'alta TPR e un basso FPR. In generale, un'area sotto la curva ROC (AUC) maggiore indica una migliore capacità discriminativa del modello. La curva ROC è spesso utilizzata per confrontare le prestazioni di diversi modelli di classificazione e per selezionare il miglior modello in base alla sua capacità di discriminare tra le classi positive e negative. Inoltre è utilizzata per calcolare l'AUC (Area Under The Curve) che fornisce una singola cifra che sintetizza le prestazioni complessive del modello. L'AUC misura l'area sottesa alla curva ROC. Un valore di AUC pari a 1 indica un modello perfetto, in cui si ha una TPR di 1 e un FPR di 0 per qualsiasi soglia di decisione. Al contrario, un valore di AUC pari a 0.5 suggerisce che il modello non è in grado di distinguere tra classi positive e negative in modo significativo, comportandosi in modo simile a una scelta casuale. In sostanza, maggiore è l'AUC, migliore è la capacità del modello di classificazione di discriminare tra le classi. La valutazione dell'AUC fornisce una panoramica della qualità del modello su un'intera gamma di soglie di decisione, rispetto a considerare solo un singolo punto di operazione. In molti casi, un valore di AUC compreso tra 0.7 e 0.9 è considerato buono, mentre un valore superiore a 0.9 è eccellente. La curva ROC e l'AUC sono strumenti comuni per valutare e confrontare le prestazioni dei modelli di classificazione.

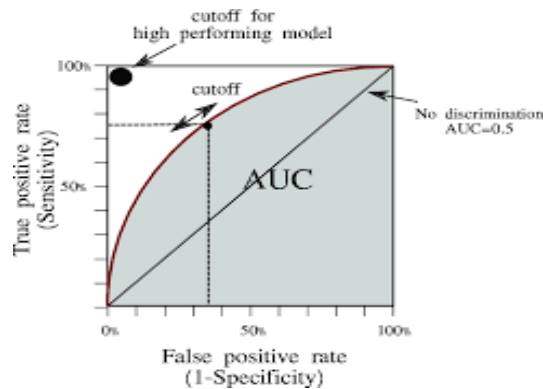


Figura 5.22: Esempio di curva ROC e AUC

I risultati ottenuti da questa analisi e con l'utilizzo di codesti strumenti verranno approfonditi nel Capitolo 6.

5.4 Classificazione in Python

Quello che è stato fatto in Matlab è stato replicato in Python, anche in questo caso è stato considerato un dataset ridotto senza effettuare su di esso alcuna pulizia o rimozione di outliers; è stato considerato il dataset grezzo così come viene dalle misurazioni. Questo è stato effettuato per eseguire un confronto e osservare se i risultati ottenuti con Matlab potessero essere ottenuti con i dati non lavorati in Python. Gli esiti come già detto precedentemente verranno discussi nel Capitolo 6, mentre qui verrà spiegata la procedura per ottenere quest'analisi. Per effettuare l'analisi sono stati seguiti i seguenti passi:

- Il primo passo è stato una visualizzazione dei dati nel dominio del tempo di tutti gli esperimenti attraverso dei plot dinamici e statici. Inoltre sono stati utilizzati degli scatter plot per esaminare la rappresentazione dei dati nel tempo.
- Il secondo, è stato l'utilizzo della FFT per osservare graficamente le frequenze dei guasti, però questi risultati essendo uguali a quelli effettuati in Matlab non sono riportati. Pure qui è stato provato ad effettuare qualche filtro o qualche tecnica per eliminare alcune frequenze inutili con scarsi risultati.
- Il terzo, è utilizzare TensorFlow e Keras per classificare i dati grezzi.

5.4.1 TensorFlow e Keras

Il codice utilizzato in Python viene inizializzato con una parte di un processo di preparazione dei dati per l'addestramento di un modello di machine learning, un modello di reti neurali utilizzando "TensorFlow" e "Sklearn". La prima parte del codice si può dividere nei seguenti passaggi:

- **Importazione delle librerie:** viene importato *LabelEncoder* da Sklearn per codificare le etichette in forma numerica. Viene importata la funzione *to_categorical* da TensorFlow.keras.utils per eseguire la codifica one-hot delle etichette, dove questa codifica categorica è una tecnica che converte i dati categorici in binari. Poiché il modello di apprendimento automatico non può elaborare dati categorici, è necessario convertirli in numerici o binari, a seconda dell'algoritmo utilizzato.

- **Definizione di parametri:** *"win_len"* è la lunghezza della finestra o del kernel, mentre *"stride"* è l'intervallo di spostamento del kernel.
- **Inizializzazione delle liste X e Y:** X è utilizzato per memorizzare i dati, mentre Y è utilizzato per memorizzare le etichette.
- **Loop sulle etichette uniche nella colonna "Pos_Fault" del dataframe:** viene creato un dataframe temporaneo contenente solo le righe con un valore specifico di *"Pos_Fault"*.
- **Creazione di finestre scorrevoli:** viene utilizzato un doppio loop per creare finestre scorrevoli dai dati nel dataframe temporaneo. Ogni finestra ha una lunghezza di *win_len* e si sposta con un intervallo di spostamento di *stride*. Le finestre vengono aggiunte alla lista X e le etichette corrispondenti vengono aggiunte alla lista Y.
- **Preparazione dei dati per l'addestramento:** le finestre sono convertite in un array numpy e modificate nella forma appropriata. Le etichette sono convertite in un array numpy.
- **Codifica delle etichette:** viene creato un oggetto *"LabelEncoder"* e addestrato sulle etichette. Le etichette vengono trasformate in valori interi tramite il metodo *transform* dell'oggetto *LabelEncoder*.
- **Codifica one-hot:** gli interi codificati vengono convertiti in vettori one-hot utilizzando la funzione *to_categorical* di Keras.

In sintesi, il codice prepara le finestre scorrevoli dai dati in base alle etichette specificate. Le etichette vengono poi codificate sia come interi che come vettori one-hot per essere utilizzate come target durante l'addestramento di un modello di machine learning.

Il passo successivo è dividere il dataset in due parti, una parte per il train e una parte per il test e ciò viene fatto attraverso il seguente codice, il quale rappresenta un esempio di prova eseguita:

```
from sklearn.model_selection import train_test_split
X_train, X_test, Y_train, Y_test =
train_test_split(X, OHE_Y, test_size=0.8, shuffle=True)
```

Per la divisione del dataset sono stati sperimentati diversi partizionamenti e i risultati che verranno riproposti nel Capitolo 6 hanno scomposizione del seguente tipo: 20% di train e 80% di test.

Successivamente viene utilizzata un'altra parte di codice per creare un modello di rete neurale convoluzionale (CNN) usando l'API di Keras in TensorFlow; il codice utilizzato si può riepilogare nei seguenti passi:

- **Importazione delle librerie:** vengono importate diverse classi e funzioni da TensorFlow per costruire il modello.
- **Configurazione dell'esecuzione eager di TensorFlow:** l'esecuzione eager è un approccio di esecuzione immediata delle operazioni, utile durante lo sviluppo e il debug del modello.
- **Definizione del numero di classi**
- **Definizione del modello CNN:** vengono definite tre teste (heads) separate per il modello, ciascuna con uno strato di convoluzione, uno strato di dropout, uno strato di max pooling e uno strato di appiattimento. Ogni testa ha parametri diversi come il

numero di filtri, la dimensione del kernel e la dimensione del pooling. Questo consente alla rete di apprendere diverse caratteristiche a diverse scale spaziali. Le uscite di tutte e tre le teste sono concatenate insieme in un unico vettore attraverso uno strato di concatenazione (merged). Un layer denso (dense1) con attivazione ReLU è utilizzato per l'interpretazione delle caratteristiche concatenate. L'ultimo strato (outputs) è uno strato denso con attivazione softmax, che è comune negli output di classificazione multiclasse. La funzione softmax assegna probabilità alle diverse classi. Il modello completo viene creato utilizzando l'API Model di Keras.

- **Compilazione del modello**
- **Visualizzazione del modello:** viene stampato un riepilogo del modello che include informazioni sulle dimensioni degli strati e il numero totale di parametri del modello.

In sintesi il codice definisce e configura un modello CNN utilizzato per la risoluzione di problemi di classificazione multiclasse.

In seguito è presente una parte di codice, successiva a quella descritta sopra, che utilizza il modello CNN per addestrare la rete neurale con i dati di allenamento e valutare le prestazioni sul set di dati di test. Di seguito si può vedere cosa fa ogni parte del codice:

- **"batch_size = 32"**: specifica la dimensione del batch, ovvero il numero di campioni di addestramento utilizzati in una singola iterazione durante l'allenamento. In questo caso, il batch size è impostato a 32 di default.
- **"epochs = 10"**: definisce il numero di epoche, ovvero il numero di volte che l'intero set di dati di addestramento viene presentato alla rete neurale per il training. In questo caso, sono definite 10 epoche.
- **"history = cnn_model.fit(...)"**: questa linea di codice addestra il modello utilizzando i dati di allenamento e registra la storia di esso. La funzione "fit" è un metodo di Keras che addestra il modello per un numero specificato di epoche utilizzando i dati di training.

Il risultato dell'addestramento viene memorizzato nella variabile "history", che può essere utilizzata per analizzare le prestazioni del modello e tracciare le curve di apprendimento. Infine si ha l'ultima parte del codice utilizzato per l'analisi delle prestazioni del modello, in particolare alla creazione di una matrice di confusione. Il codice confronta le previsioni del modello con le etichette reali utilizzando una matrice di confusione e visualizza i risultati mediante una "heatmap". Questo tipo di visualizzazione è utile per valutare le prestazioni di un modello di classificazione, specialmente in problemi con più classi.

In questo capitolo verranno riportati tutti i risultati ottenuti con una descrizione dettagliata di essi

Il seguente capitolo si dividerà in 3 parti principali in cui saranno esposti i risultati ottenuti dal dataset ridotto ed ogni sezione avrà un parte di visualizzazione e una parte di analisi. Inoltre verrà esposto perché dopo una prima analisi è stato deciso di utilizzare strumenti differenti e anche tipi di procedimento diversi per arrivare al risultato finale.

- Nella prima fase si esporrà una rappresentazione delle misure nel tempo e una delle misure in frequenza; questo sottocapitolo prenderà il nome di "Risultati analisi in frequenza".
- Nella seconda, si presenteranno i risultati ottenuti dalle due applicazioni di Matlab; verrà chiamato "Risultati Diagnostic Feature Designer e Classification Learner".
- Infine nella terza, si avrà una prima parte di analisi descrittiva e una seconda parte di analisi e classificazione attraverso Python e le sue librerie citate precedentemente, codesto sottocapitolo prenderà il nome di "Risultati Python".

6.1 Risultati analisi in frequenza

Il seguente paragrafo si dividerà in due sezioni ed in entrambe si attuerà una visualizzazione dei dati per osservare il loro andamento temporale e una visualizzazione in frequenza per verificare se è possibile individuare il fault oppure no.

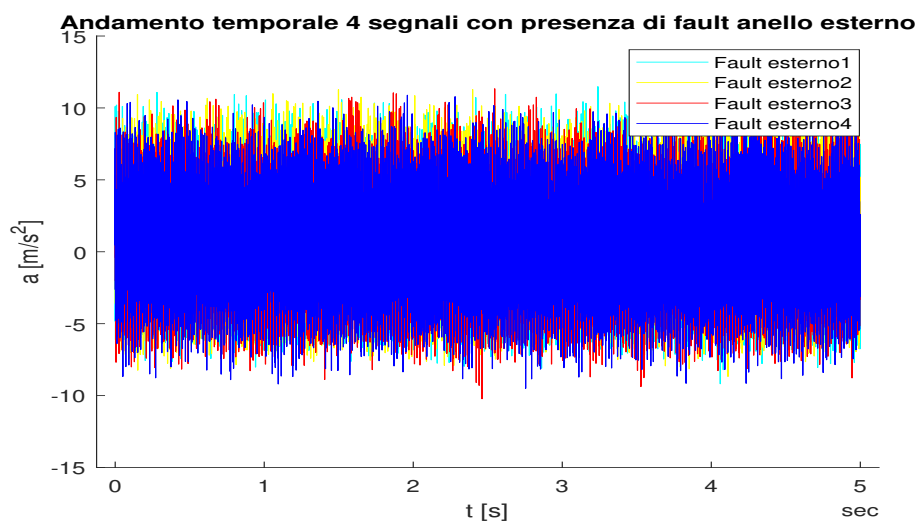
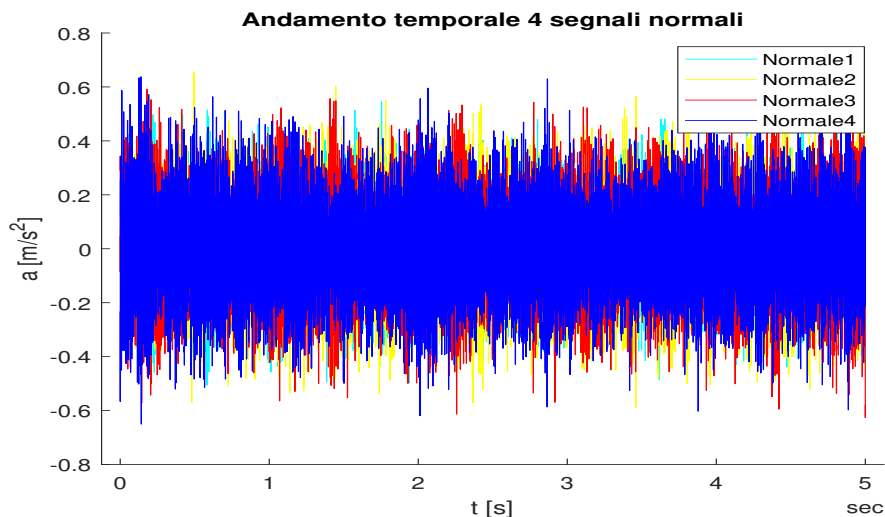
- Nella prima parte si discuterà degli esiti degli esperimenti ottenuti, tramite l'analisi in frequenza, considerando la misura dell'accelerometro ottenuta posizionandolo il più vicino possibile al cuscinetto.
- Nella seconda parte si parlerà dei risultati degli esperimenti ottenuti considerando il sensore posto a una distanza maggiore dell'accelerometro sopra indicato.

6.1.1 Analisi segnale Yc

Il segnale ottenuto dal posizionamento più vicino al cuscinetto è stato chiamato Yc ed è stata effettuata sia un' analisi descrittiva sia un'analisi per il rilevamento del guasto. Nella

Figura 6.1 può essere osservato l'andamento temporale dei 16 esperimenti effettuati dove l'asse delle ordinate di ogni grafico rappresenta l'accelerazione in m/s^2 , il quale riproduce il comportamento del cuscinetto, mentre nell'asse delle ascisse viene rappresentato il tempo in s (secondi) che rappresenta una piccola percentuale dell'effettivo tempo di misurazione effettuato, questo per questioni di complessità di esecuzione del codice che racchiude al suo interno anche il tempo di esecuzione; 5 secondi sono sufficienti per notare una distinzione tra un cuscinetto buono e uno che presenta un guasto. La Figura 6.1 può essere divisa in 4 sezioni:

- Il primo grafico rappresenta l'andamento dei 4 segnali rispetto al tempo dove non è presente nessun difetto.
- Il secondo grafico mostra l'andamento temporale dei 4 segnali, dove è stato eseguito un'incisione lineare come guasto nell'anello esterno del cuscinetto.
- Il terzo grafico considera l'andamento temporale dei 4 segnali, dove è stato eseguito un'incisione lineare come fault nell'anello interno del cuscinetto.
- Mentre il quarto grafico considera l'andamento temporale dei 4 segnali, dove è stato eseguito un'incisione lineare come guasto nella sfera o rullo del cuscinetto.



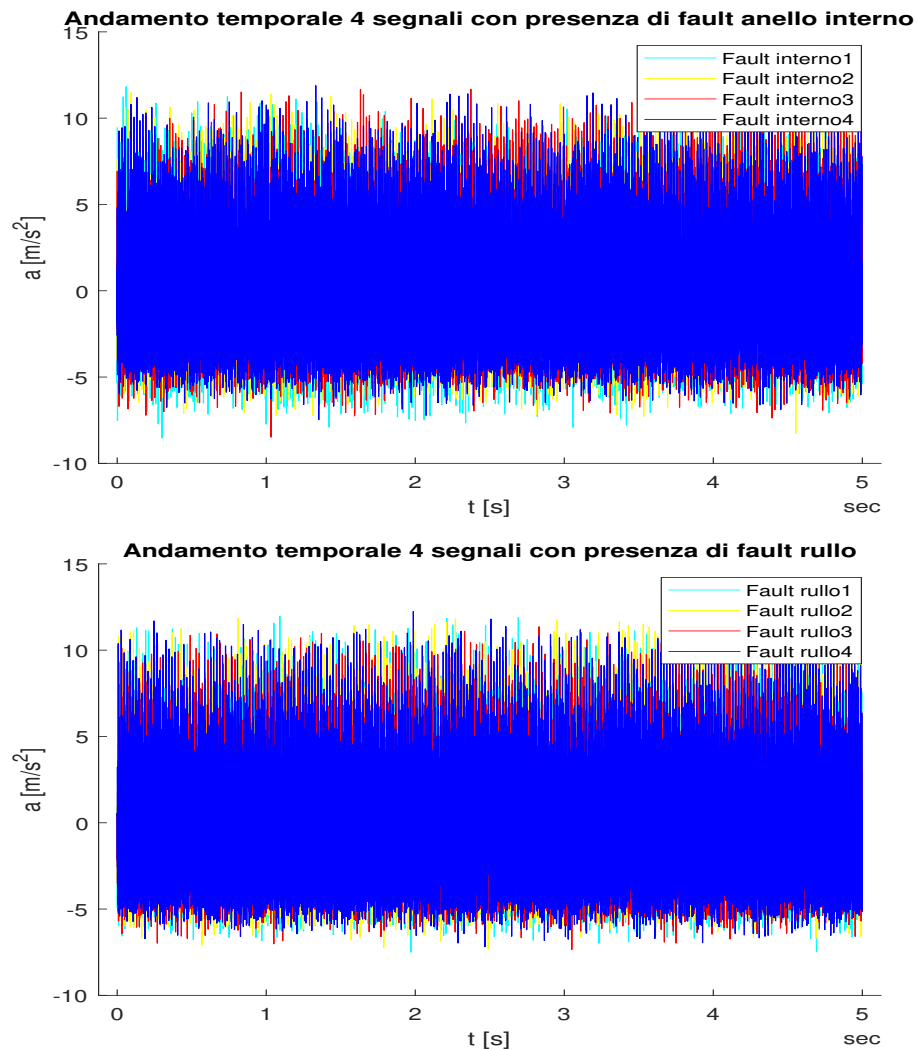


Figura 6.1: Andamento temporale 16 esperimenti del segnale Yc divisi per tipo

Già da questa prima rappresentazione si può osservare che esiste una differenza tra le ampiezze, osservando l'asse delle ordinate, tra i diversi segnali; cioè tra il segnale che non presenta nessun fault e i segnali che presentano un guasto, quest'ultimi hanno un'ampiezza maggiore rispetto al segnale del cuscinetto senza difetto. Per accentuare maggiormente questa osservazione è stato deciso di prendere un segnale per tipo e rappresentare il tutto graficamente per effettuare un primo confronto. Questo viene eseguito nella Figura 6.2 dove:

- Il segnale di colore nero rappresenta il segnale senza guasti.
- Il segnale di colore rosso, anche se si vede solo nello sfondo, mostra il fault applicato all'anello esterno.
- Il segnale di colore blu riproduce il fault presente nell'anello interno.
- Il segnale di colore giallo descrive il fault presente nella sfera o rullo.

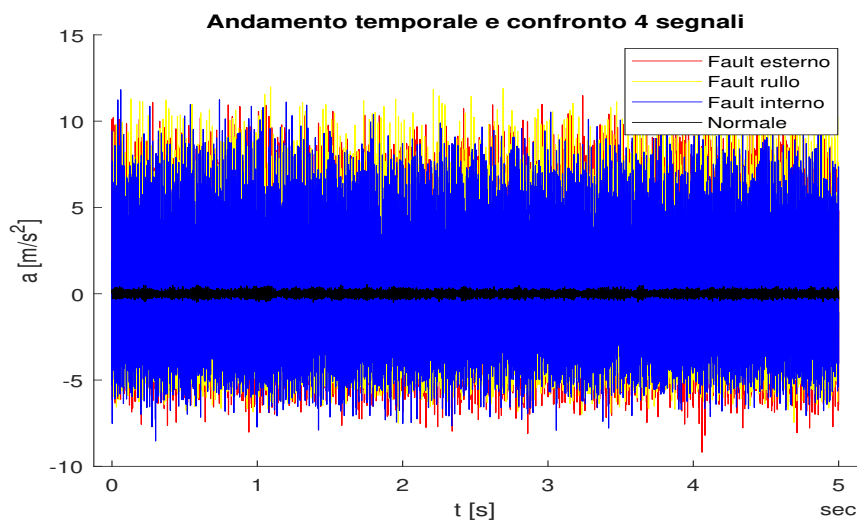
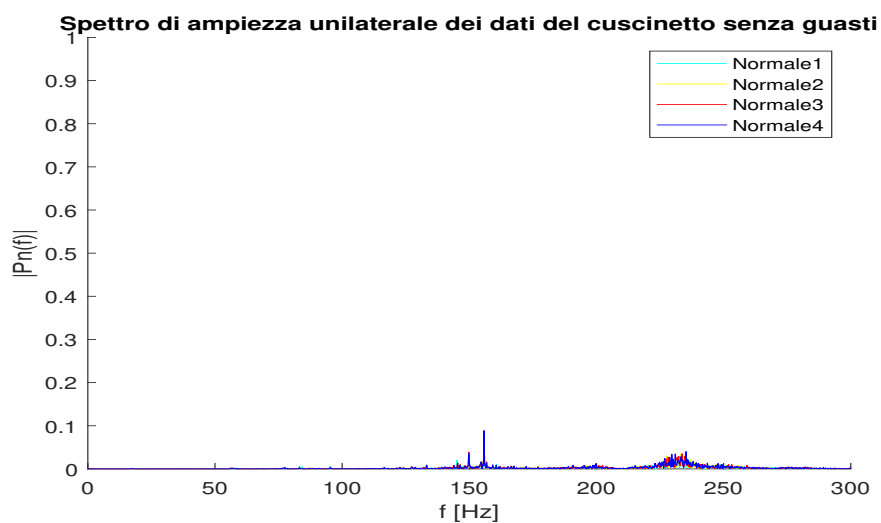


Figura 6.2: Confronto 4 segnali del dataset

Dopo questa prima visualizzazione dei segnali si voleva approfondire la loro analisi attraverso la Fast Fourier Transform, per scoprire se era possibile notare le frequenze dei fault e la frequenza di normale funzionamento del macchinario dai grafici ottenuti attraverso la trasformata. Come sopra è stata fatta una rappresentazione della FFT raggruppando per tipo di classe e successivamente è stato riproposto una riproduzione della trasformata di un singolo segnale per tipologia. Nella Figura 6.3, come sopra si ha:

- Il primo grafico rappresenta l'ampiezza dello spettro del segnale senza guasto.
- Il secondo grafico riproduce l'ampiezza dello spettro del segnale con fault presente nell'anello esterno.
- Il terzo grafico descrive l'ampiezza dello spettro del segnale con fault presente nell'anello interno.
- Il quarto grafico mostra l'ampiezza dello spettro del segnale con fault presente nella sfera o rullo.



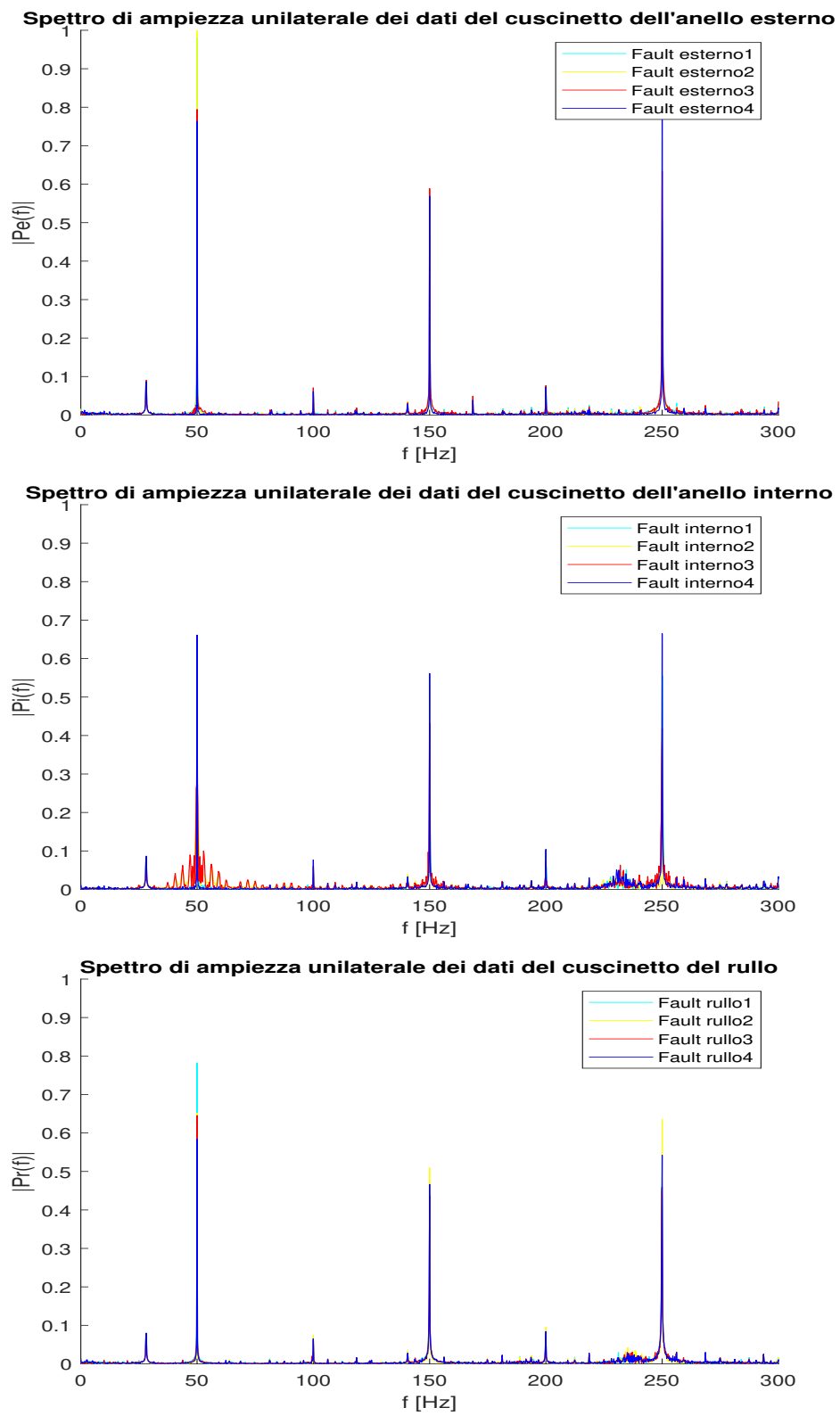


Figura 6.3: Trasformata di Fourier 16 esperimenti del segnale Yc divisi per tipo

Per avere una rappresentazione migliore si considererà un grafico alla volta per tipo di classe. Il primo grafico che verrà visto è quello che riguarda il comportamento nominale, dove il cuscinetto è in buono stato. Nella Figura 6.4 si può osservare l'andamento in frequenza del segnale normale, dove per normale si intende nessuna creazione di guasti nel cuscinetto. Dal grafico si può osservare come nelle ascisse viene rappresentata la frequenza mentre nelle ordinate l'ampiezza, come si può vedere non si riesce a distinguere la frequenza normale del macchinario di 16.66Hz e le sue armoniche mentre si presenta qualcosa sui 150Hz .

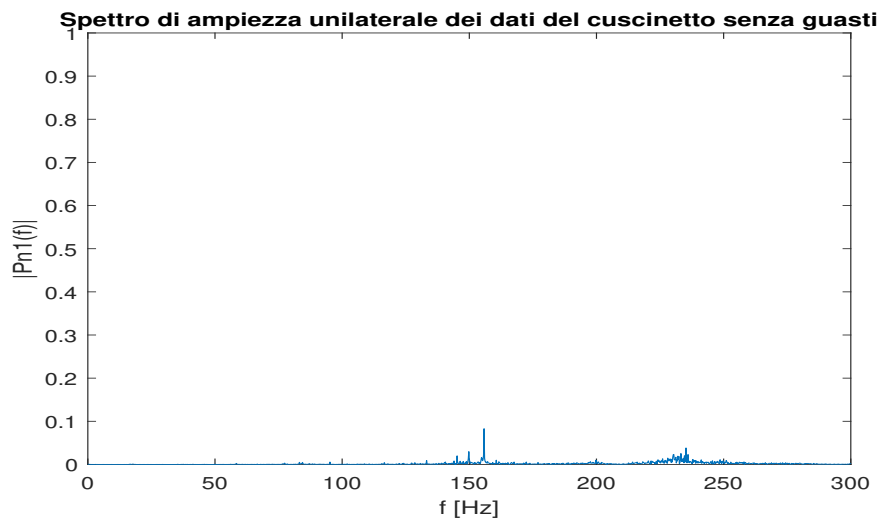


Figura 6.4: Risposta in frequenza del segnale Y_c senza fault

Successivamente è stato considerato la risposta in frequenza del segnale che presenta un guasto nell'anello esterno del cuscinetto, è stato osservato che la frequenza del fault dell'anello esterno di 93.33Hz non si riscontra nel grafico, ma si nota qualche picco a 50Hz , 150Hz e così via con le armoniche dispari come si vede in Figura 6.5.

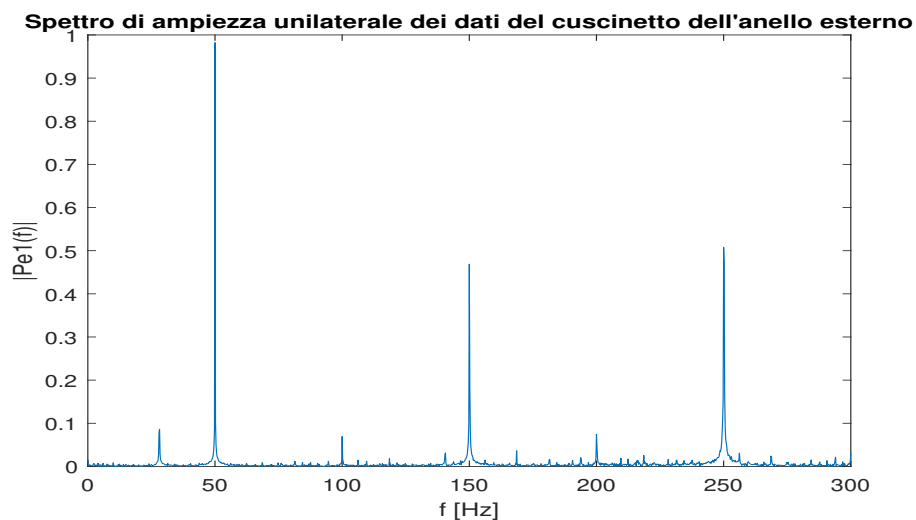


Figura 6.5: Risposta in frequenza del segnale Y_c con fault presente nell'anello esterno

In seguito è stato esaminato il segnale che presentava un fault nell'anello interno e come si nota in Figura 6.6 si hanno sempre dei picchi verso 50Hz , 150Hz e così via con le armoniche dispari, mentre per quanto riguarda la frequenza del fault dell'anello interno di 140Hz non si vede nessun picco, né in quella frequenza né nelle sue armoniche.

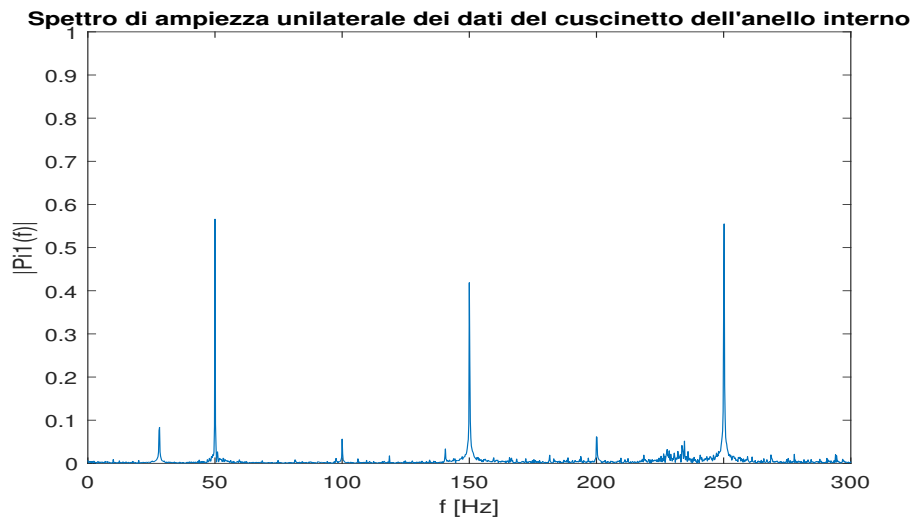


Figura 6.6: Risposta in frequenza del segnale Yc con fault presente nell'anello interno

Come ultimo è stato preso in considerazione il segnale che presentava un fault nella sfera (o rullo) e come i segnali precedenti non si riesce a visualizzare un picco della frequenza del fault nella sfera di 87.99Hz, ma si hanno picchi di 50Hz, 150Hz e così via come si può rilevare dal grafico in Figura 6.7

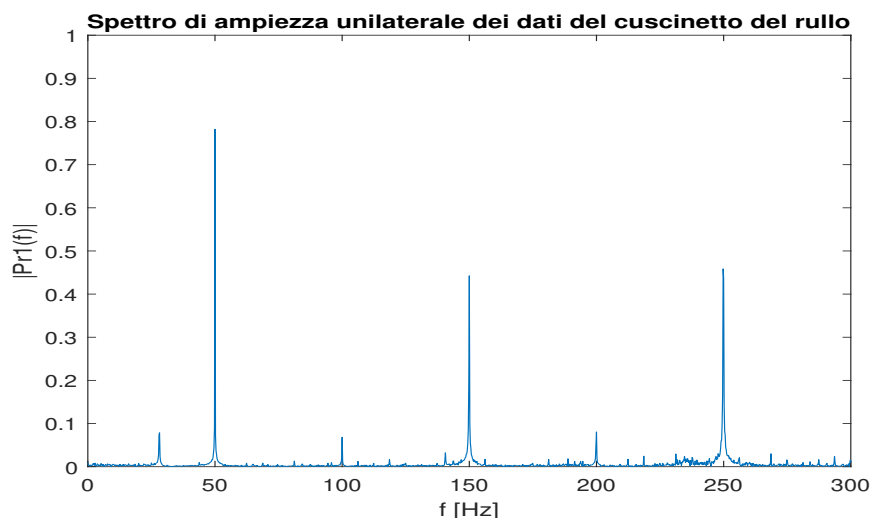


Figura 6.7: Risposta in frequenza del segnale Yc con fault presente nella sfera o rullo

In seguito a questa prima analisi in frequenza si può osservare che si riesce a distinguere se c'è guasto oppure no, ma non si riesce a classificare la tipologia di fault, questo lo si può dedurre dal cambiamento che si ha in ampiezza dei 4 grafici delle risposte in frequenza riportate sopra. Per correttezza è stato deciso di riportare anche i risultati conseguiti dal segnale Z_a, cioè il segnale ottenuto dalle misurazioni dell'accelerometro posto un po' più lontano dal posizionamento del cuscinetto.

6.1.2 Analisi segnale Z_a

In questa sezione si opererà allo stesso modo come per il segnale Yc. Infatti si avrà una parte di visualizzazione e una parte di analisi. Nella prima parte è stata presentata l'andamento temporale dei segnali come si nota in Figura 6.8.

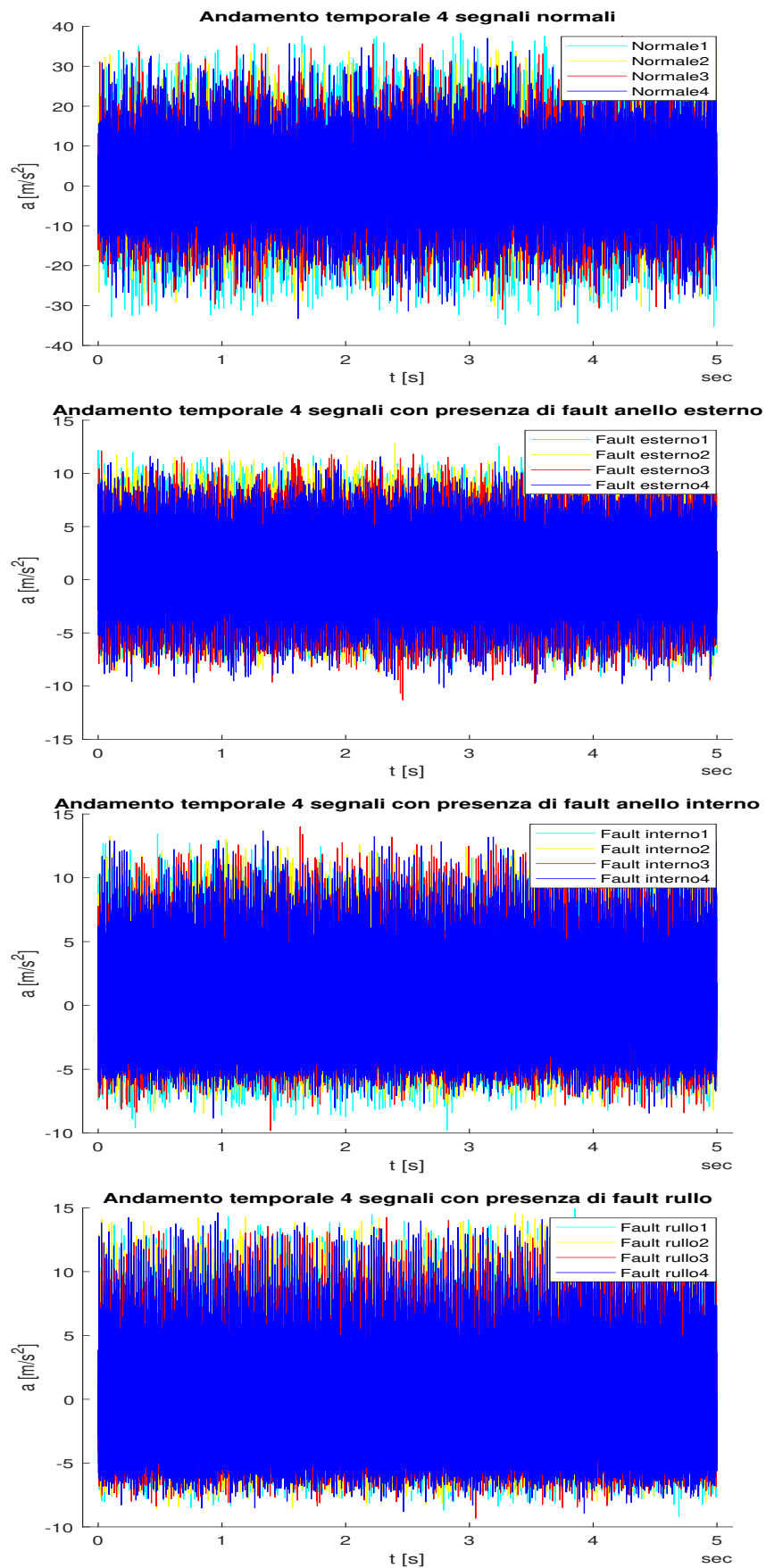


Figura 6.8: Andamento temporale 16 esperimenti del segnale Z_a divisi per tipo

La Figura 6.8 è divisa nel seguente modo:

- Il primo grafico rappresenta l'andamento temporale del segnale senza fault.
- Il secondo grafico riproduce l'andamento temporale del segnale con il fault presente nell'anello esterno.
- Il terzo grafico descrive l'andamento temporale del segnale con il fault posto nell'anello interno.
- Il quarto grafico raffigura l'andamento temporale del segnale con il fault creato nella sfera (o rullo).

Successivamente è stato scelto 1 segnale per tipologia di classe e sono stati confrontati come riportato in Figura 6.9.

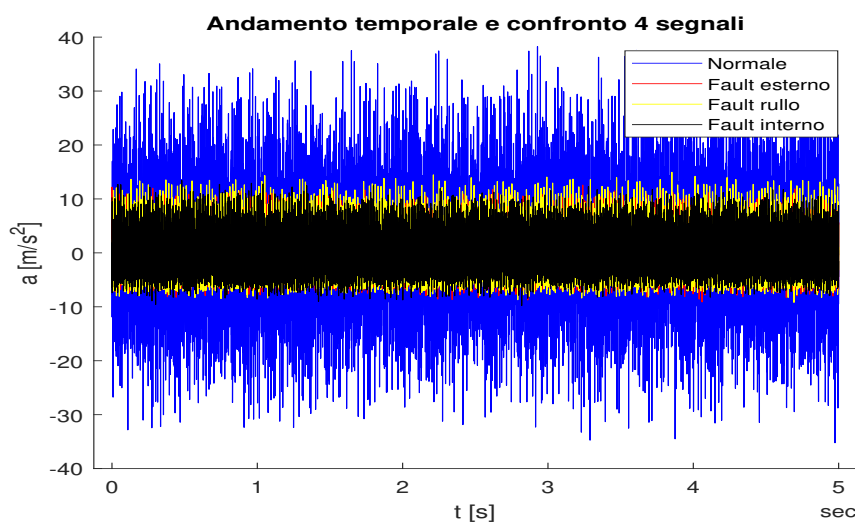


Figura 6.9: Confronto andamento temporale segnali Za

Nella Figura 6.9 si ha:

- Il colore nero rappresenta il fault dell'anello interno.
- Il colore blu raffigura il segnale senza fault.
- Il colore rosso riproduce il fault all'anello esterno.
- Il colore giallo descrive il fault alla sfera (o rullo).

Da questa prima rappresentazione si nota che il segnale normale ha un'ampiezza maggiore rispetto agli altri e ciò probabilmente è dovuto al fatto che questo sensore è stato posizionato un po' più lontano rispetto all'altro sensore e quindi gli effetti dei fault non si identificano molto bene. In seguito è stata eseguita la Trasformata di Fourier e nella Figura 6.10 viene riportato ciò. Come è successo con il segnale Yc, anche nel segnale Za ci sono delle frequenze che si ripetono, ma non sono quelle quelle calcolate in precedenza. La cosa che si nota maggiormente è che il comportamento normale ha un'ampiezza maggiore rispetto agli altri segnali e quindi, non si distingue il comportamento fault-free da quello con guasti. Inoltre per correttezza sono state riportate anche le FFT singole dei segnali in Figura 6.11 anche se non si osserva niente di più da quello detto precedentemente.

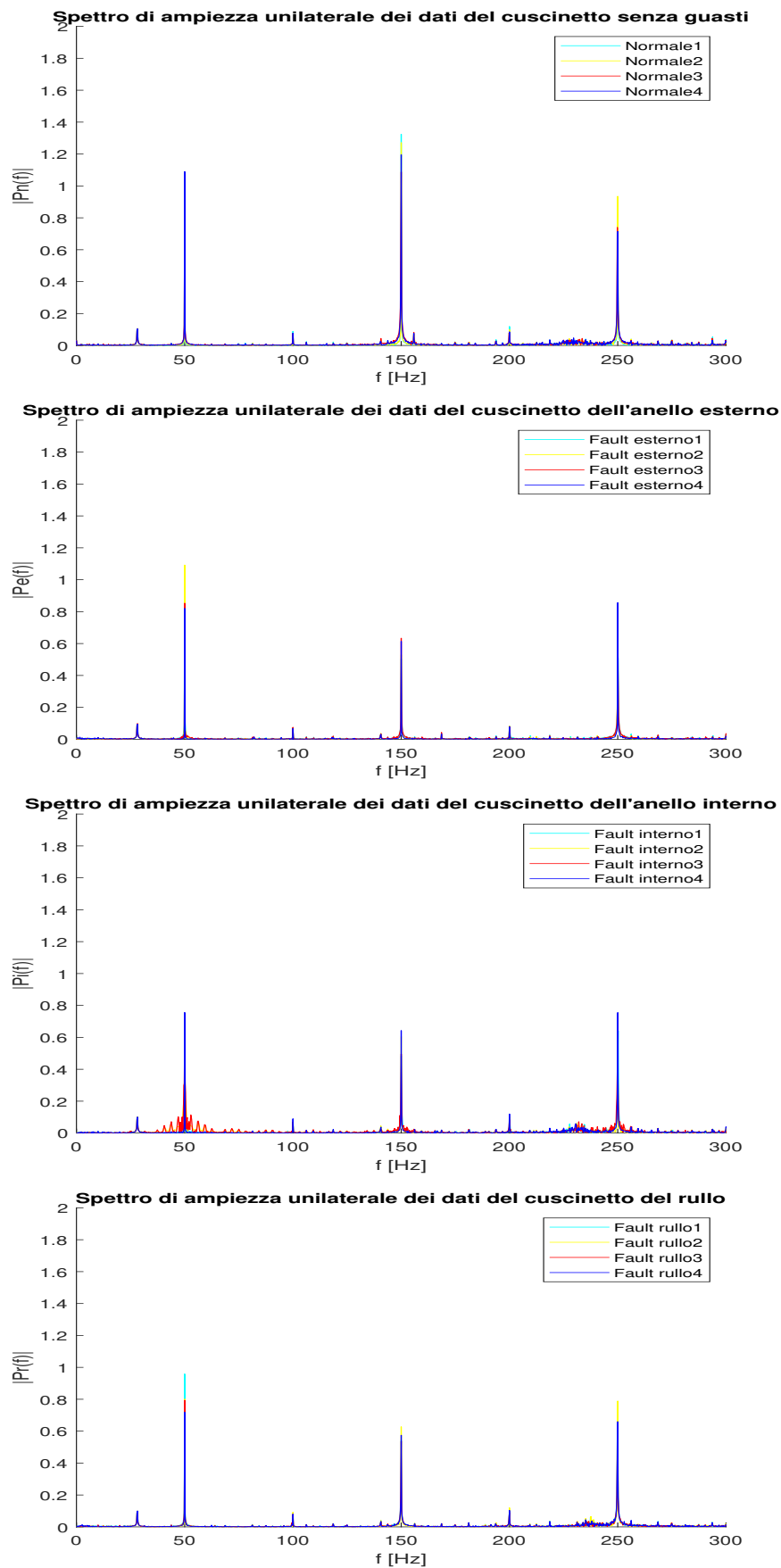


Figura 6.10: Trasformata di Fourier 16 esperimenti del segnale Z_a divisi per tipo

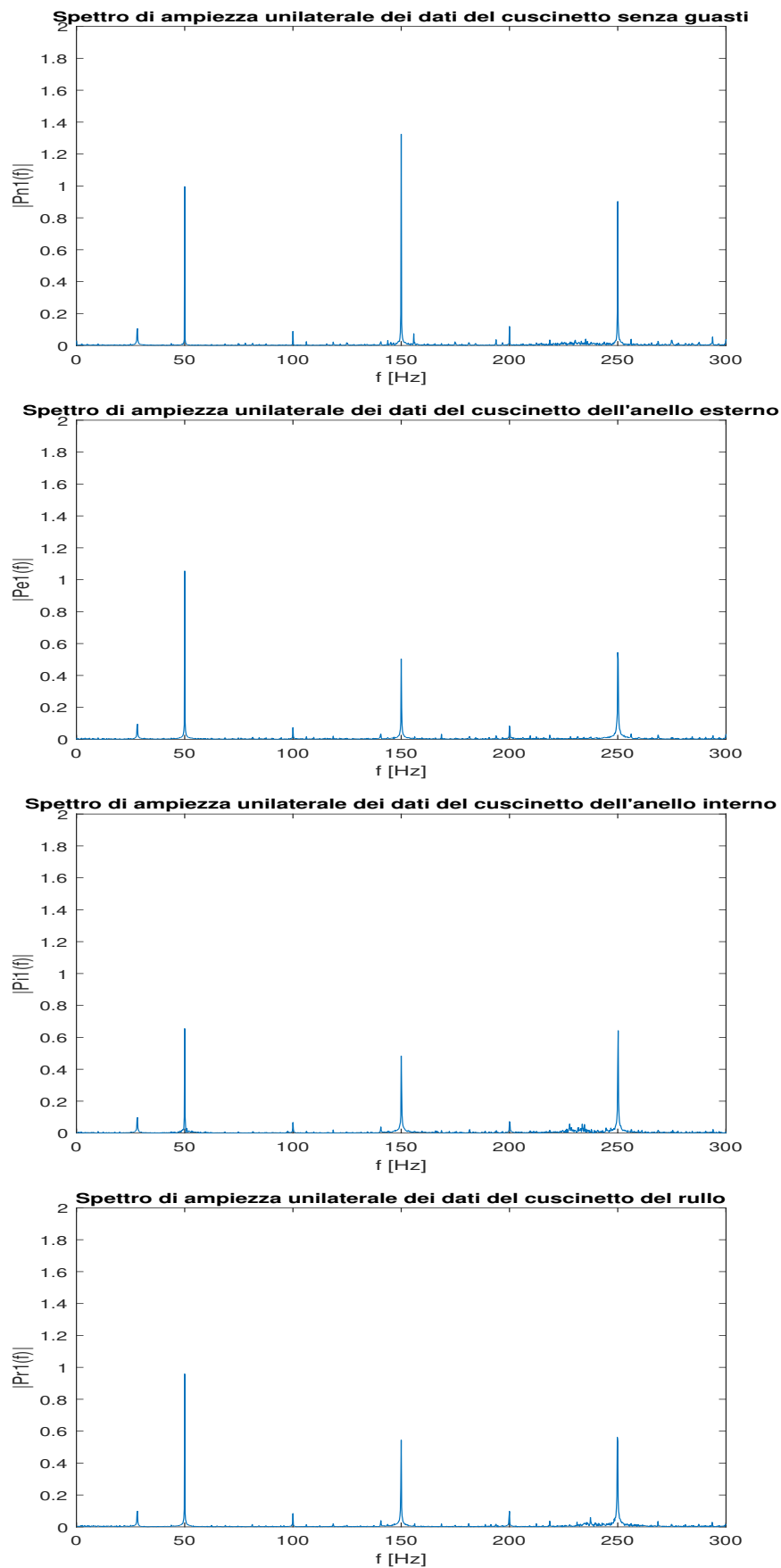


Figura 6.11: Risposta in frequenza di 4 segnali campione di Z_a

6.2 Risultati Diagnostic Feature Designer e Classification Learner

In seguito ai limitati risultati ottenuti dall'analisi in frequenze, è stato deciso di procedere attraverso algoritmi di classificazione attraverso due applicazioni di Matlab. Il seguente sottocapitolo si dividerà in due parti:

- La prima parte è utilizzata per discutere dei risultati ottenuti tramite il segnale Yc.
- La seconda parte è impiegata per trattare degli esiti ottenuti dal segnale Za.

6.2.1 Analisi segnale Yc

Come per il sottocapitolo precedente, anche qui ci sarà una parte di visualizzazione e una parte di analisi, inoltre in una prima fase ci saranno i risultati ottenuti dal Diagnostic Feature Designer e in una seconda fase quelli ottenuti dal Classification Learner. In Figura 6.12 si può osservare l'andamento temporale del segnale Yc.

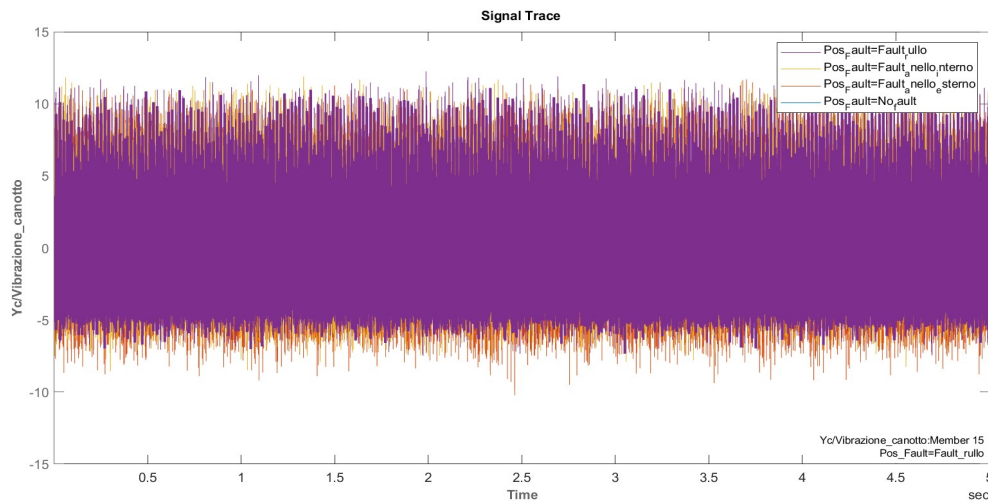


Figura 6.12: Andamento temporale segnale Yc

Successivamente in Figura 6.13 è stato provato ad eseguire il TSA per vedere se il segnale diventava più pulito.

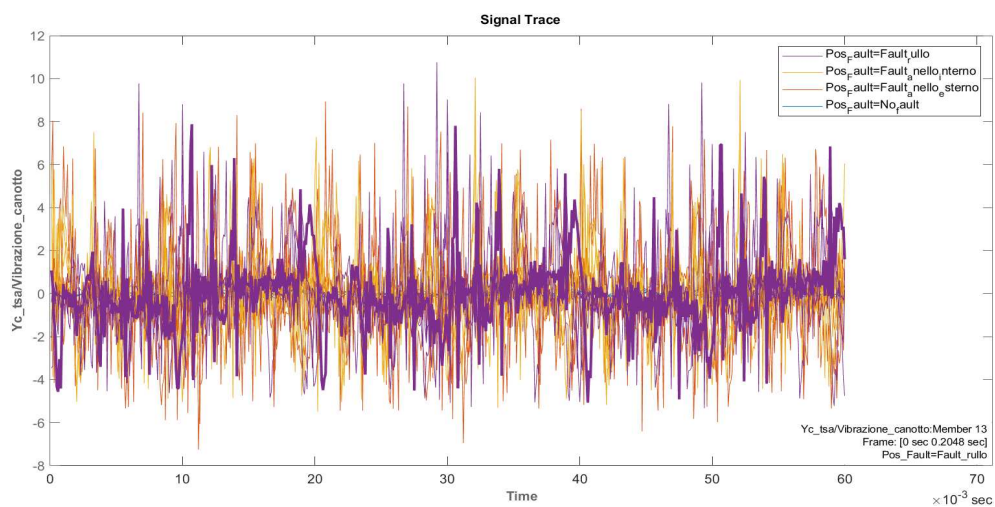


Figura 6.13: Andamento temporale segnale Yc con TSA

La Figura 6.14 mostra un tentativo di utilizzo dello spettro del segnale, dopo l'applicazione del TSA, con limitati risultati anche qui.

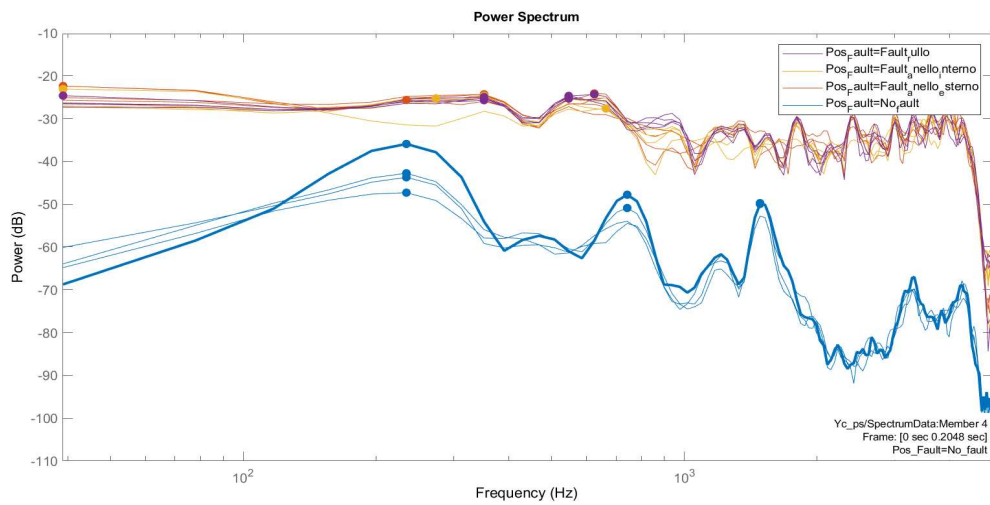


Figura 6.14: Spettro del segnale Yc dopo l'applicazione del TSA

Successivamente sono state calcolate le feature del segnale Yc senza TSA inizialmente e i risultati si possono osservare in Figura 6.15 e Figura 6.16.

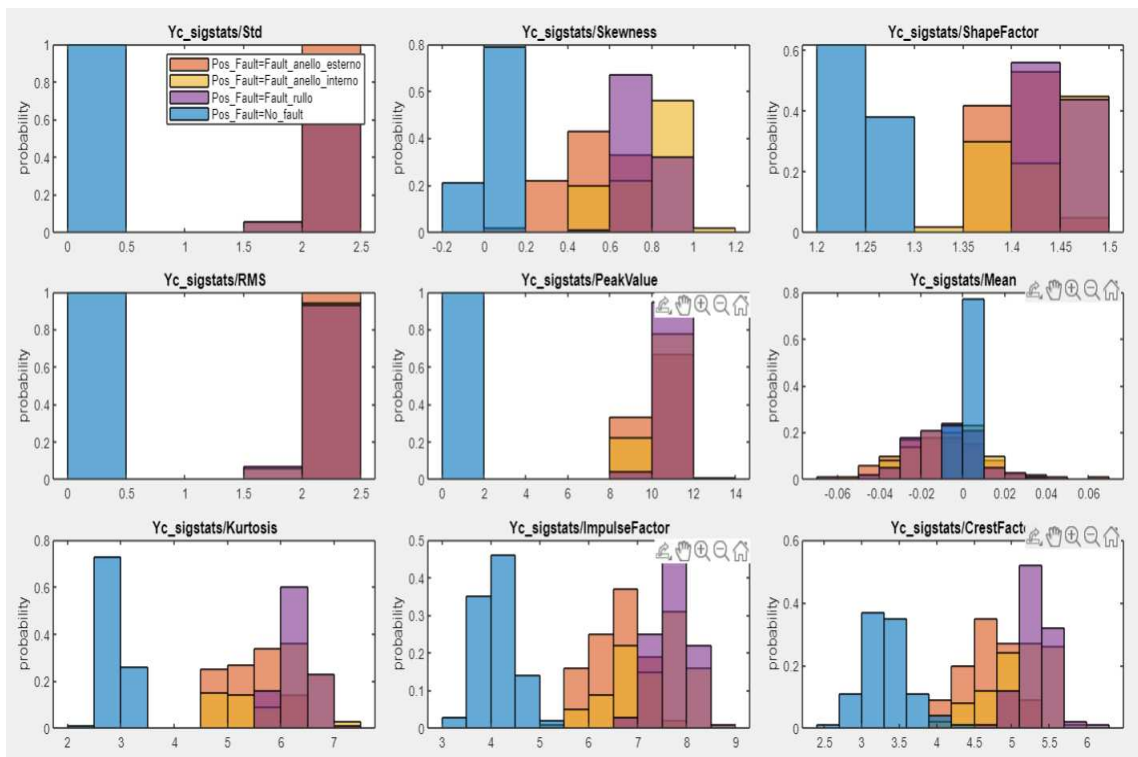


Figura 6.15: Istogrammi Feature Yc

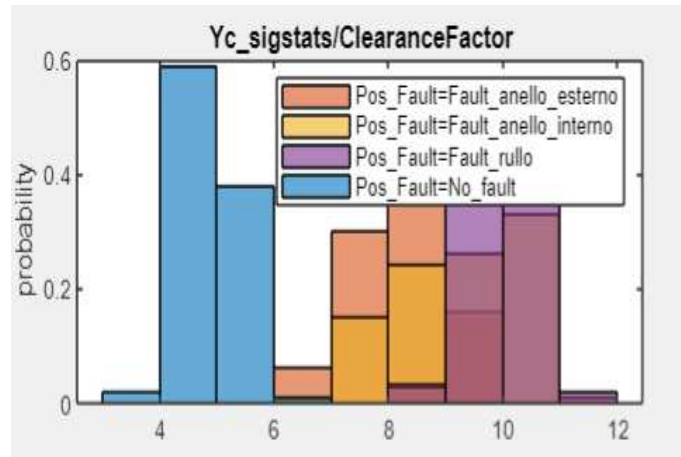


Figura 6.16: Istogramma Feature Yc

Questi istogrammi servono per valutare la probabilità che una feature appartiene ad un range di valori, infatti si vuole che le colonne con diversi colori siano separate tra di loro per avere una totale separazione tra gli istogrammi di un colore, quindi di una classe, dagli altri istogrammi di colore diverso, quindi le altre classi; inoltre è importante che non ci sia sovrapposizione. Quanto fatto per il segnale Yc è stato replicato anche per il segnale Yc con TSA come mostrato in Figura 6.17 e Figura 6.18.

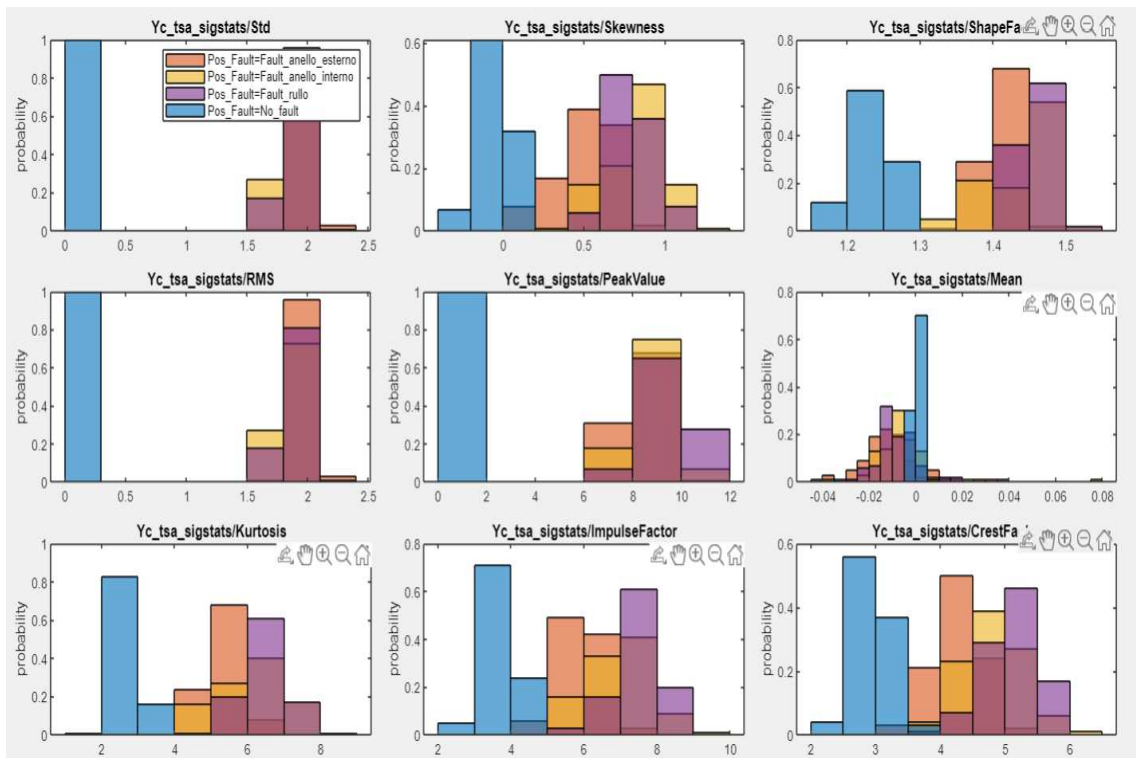


Figura 6.17: Istogrammi Feature Yc con TSA

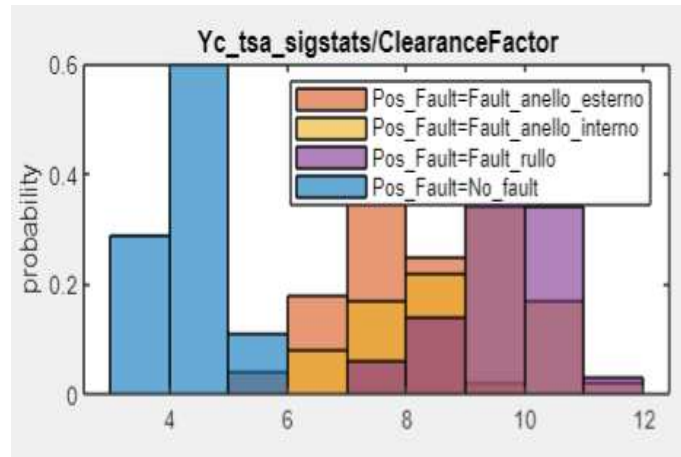


Figura 6.18: Istogramma Feature Yc con TSA

Successivamente è stato calcolato il ranking delle feature del segnale Yc con e senza TSA come mostrato in Figura 6.19, dove vengono elencate le feature per ordine di importanza; a sinistra della figura si ha il risultato del ranking normalizzato che va da 0 a 1 dove il valore 1 ha maggiore importanza per la classificazione, mentre a destra si ha lo stesso risultato ma non normalizzato.

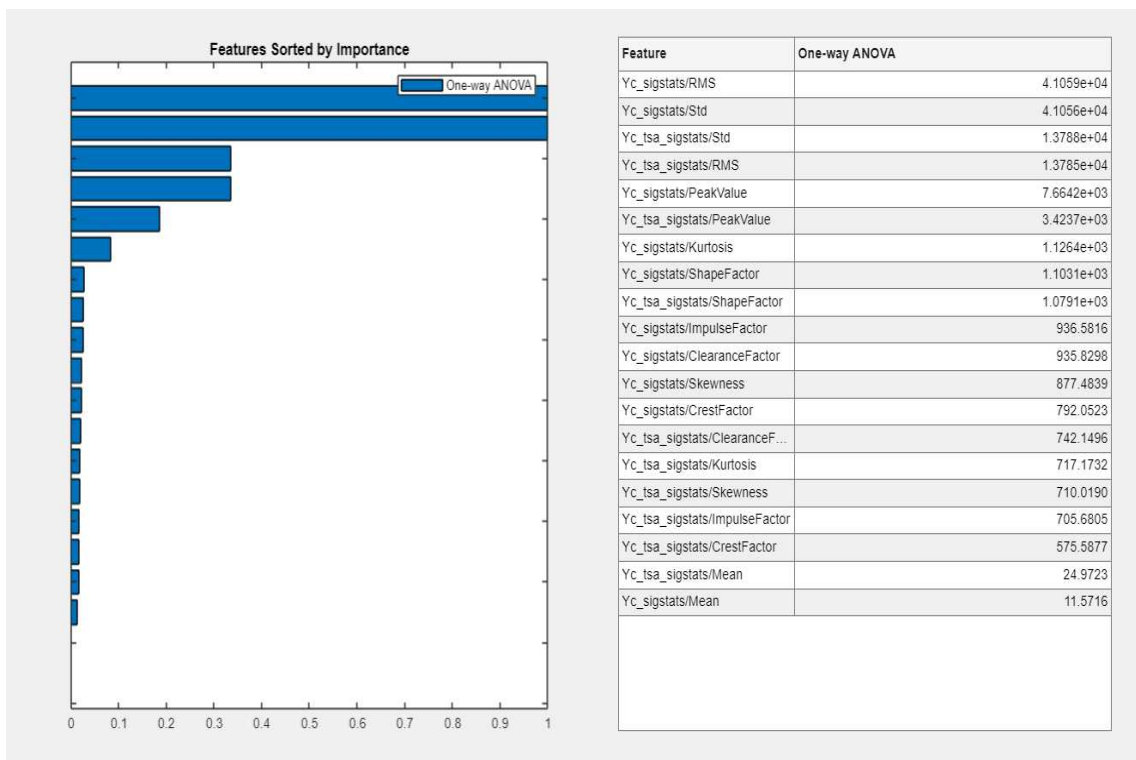


Figura 6.19: Calcolo ranking feature di Yc con e senza TSA

Le 5 feature più importanti (rms, shape factor, standard deviation, kurtosis, peak value) sono state importate nel Classification Learner ed è stata eseguita la vera e propria classificazione.

Successivamente sono stati allenati e testati i modelli come mostrano le Figure 6.20 e 6.21; dove l'80% di dati sono stati utilizzati per il train e il 20% di test in questo esempio riportato.

4.1 Tree Last change: Fine Tree	Accuracy (Validation): 77.2% 5/5 features	4.20 Ensemble Last change: Boosted Trees	Accuracy (Validation): 77.2% 5/5 features
4.2 Tree Last change: Medium Tree	Accuracy (Validation): 77.5% 5/5 features	4.21 Ensemble Last change: Bagged Trees	Accuracy (Validation): 80.0% 5/5 features
4.3 Tree Last change: Coarse Tree	Accuracy (Validation): 70.6% 5/5 features	4.22 Ensemble Last change: Subspace Discriminant	Accuracy (Validation): 75.3% 5/5 features
4.4 Linear Discriminant Last change: Linear Discriminant	Accuracy (Validation): 75.9% 5/5 features	4.23 Ensemble Last change: Subspace KNN	Accuracy (Validation): 71.2% 5/5 features
4.5 Quadratic Discriminant Last change: Quadratic Discriminant	Accuracy (Validation): 78.1% 5/5 features	4.24 Ensemble Last change: RUSBoosted Trees	Accuracy (Validation): 77.8% 5/5 features
4.6 Naive Bayes Last change: Gaussian Naive Bayes	Accuracy (Validation): 72.2% 5/5 features	4.25 Neural Network Last change: Narrow Neural Network	Accuracy (Validation): 81.9% 5/5 features
4.7 Naive Bayes Last change: Kernel Naive Bayes	Accuracy (Validation): 74.7% 5/5 features	4.26 Neural Network Last change: Medium Neural Network	Accuracy (Validation): 81.6% 5/5 features
4.8 SVM Last change: Linear SVM	Accuracy (Validation): 77.2% 5/5 features	4.27 Neural Network Last change: Wide Neural Network	Accuracy (Validation): 79.4% 5/5 features
4.9 SVM Last change: Quadratic SVM	Accuracy (Validation): 80.3% 5/5 features	4.28 Neural Network Last change: Bilayered Neural Network	Accuracy (Validation): 81.9% 5/5 features
4.10 SVM Last change: Cubic SVM	Accuracy (Validation): 81.6% 5/5 features	4.29 Neural Network Last change: Trilayered Neural Network	Accuracy (Validation): 82.8% 5/5 features
4.11 SVM Last change: Fine Gaussian SVM	Accuracy (Validation): 81.9% 5/5 features	4.30 Kernel Last change: SVM Kernel	Accuracy (Validation): 72.2% 5/5 features
4.12 SVM Last change: Medium Gaussian SVM	Accuracy (Validation): 81.9% 5/5 features	4.31 Kernel Last change: Logistic Regression Kernel	Accuracy (Validation): 69.4% 5/5 features
4.13 SVM Last change: Coarse Gaussian SVM	Accuracy (Validation): 75.9% 5/5 features		

Figura 6.20: Risultato allenamento per la classificazione del segnale Yc con diversi algoritmi

4.1 Tree Last change: Fine Tree	Accuracy (Test): 70.0% 5/5 features	4.20 Ensemble Last change: Boosted Trees	Accuracy (Test): 73.8% 5/5 features
4.2 Tree Last change: Medium Tree	Accuracy (Test): 72.5% 5/5 features	4.21 Ensemble Last change: Bagged Trees	Accuracy (Test): 73.8% 5/5 features
4.3 Tree Last change: Coarse Tree	Accuracy (Test): 63.7% 5/5 features	4.22 Ensemble Last change: Subspace Discriminant	Accuracy (Test): 76.2% 5/5 features
4.4 Linear Discriminant Last change: Linear Discriminant	Accuracy (Test): 75.0% 5/5 features	4.23 Ensemble Last change: Subspace KNN	Accuracy (Test): 68.8% 5/5 features
4.5 Quadratic Discriminant Last change: Quadratic Discriminant	Accuracy (Test): 80.0% 5/5 features	4.24 Ensemble Last change: RUSBoosted Trees	Accuracy (Test): 73.8% 5/5 features
4.6 Naive Bayes Last change: Gaussian Naive Bayes	Accuracy (Test): 75.0% 5/5 features	4.25 Neural Network Last change: Narrow Neural Network	Accuracy (Test): 81.2% 5/5 features
4.7 Naive Bayes Last change: Kernel Naive Bayes	Accuracy (Test): 78.8% 5/5 features	4.26 Neural Network Last change: Medium Neural Network	Accuracy (Test): 73.8% 5/5 features
4.8 SVM Last change: Linear SVM	Accuracy (Test): 77.5% 5/5 features	4.27 Neural Network Last change: Wide Neural Network	Accuracy (Test): 78.8% 5/5 features
4.9 SVM Last change: Quadratic SVM	Accuracy (Test): 78.8% 5/5 features	4.28 Neural Network Last change: Bilayered Neural Network	Accuracy (Test): 76.8% 5/5 features
4.10 SVM Last change: Cubic SVM	Accuracy (Test): 78.8% 5/5 features	4.29 Neural Network Last change: Trilayered Neural Network	Accuracy (Test): 81.2% 5/5 features
4.11 SVM Last change: Fine Gaussian SVM	Accuracy (Test): 72.5% 5/5 features	4.30 Kernel Last change: SVM Kernel	Accuracy (Test): 70.0% 5/5 features
4.12 SVM Last change: Medium Gaussian SVM	Accuracy (Test): 78.8% 5/5 features	4.31 Kernel Last change: Logistic Regression Kernel	Accuracy (Test): 72.5% 5/5 features
4.13 SVM Last change: Coarse Gaussian SVM	Accuracy (Test): 78.8% 5/5 features		

Figura 6.21: Risultato dei test per la classificazione del segnale Yc con diversi algoritmi

Come mostrano le seguenti figure si hanno buoni risultati con diversi algoritmi di classificazione sia per quanto riguarda l'allenamento che per i test, ma non sono ottimi.

Come campione è stato deciso di riportare il modello migliore, cioè il **Trilayered Neural Network**.

In Figura 6.22 si può osservare una prima descrizione dell’algoritmo dove vengono rappresentati i livelli di accuratezza sia per quanto riguarda il training che per il testing.

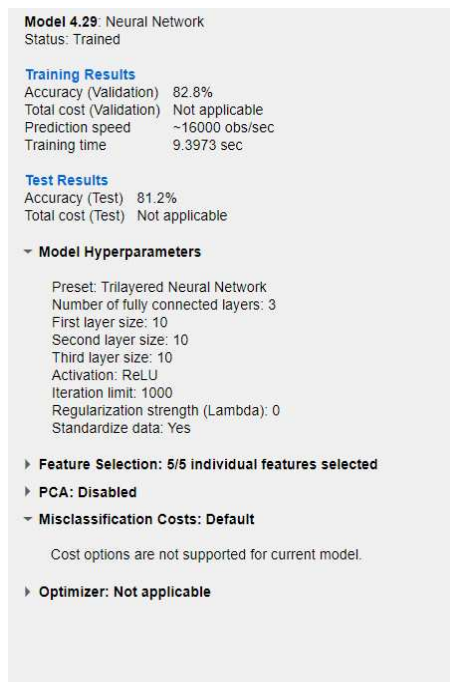


Figura 6.22: Informazioni train e test del segnale Yc con il modello trilayered neural network

Sempre del modello trilayered neural network sono state riportate la matrice di confusione e la curva ROC di train e di test. La matrice di confusione e le sue diverse valutazioni si possono osservare nelle figure sottostanti.

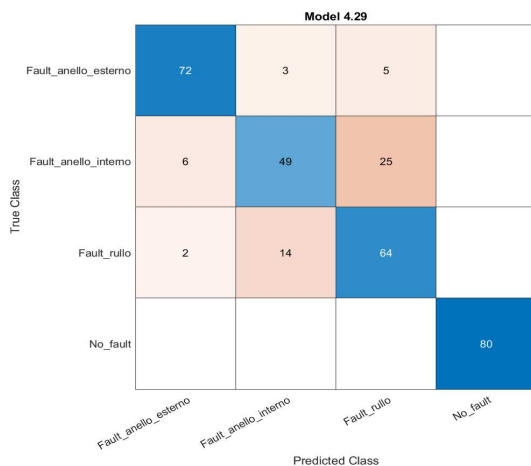


Figura 6.23: Matrici di confusione numero osservazioni train

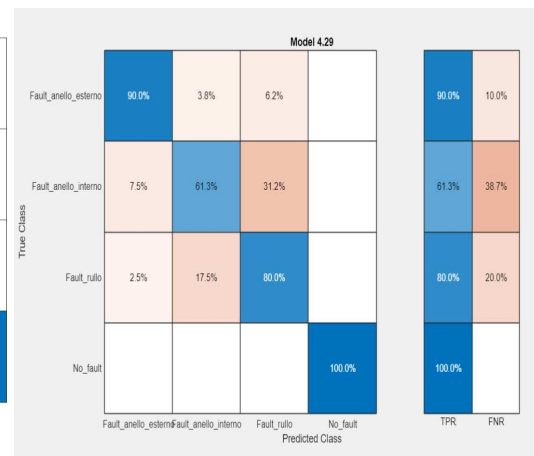


Figura 6.24: Matrici di confusione valutazione true positive/false negative train

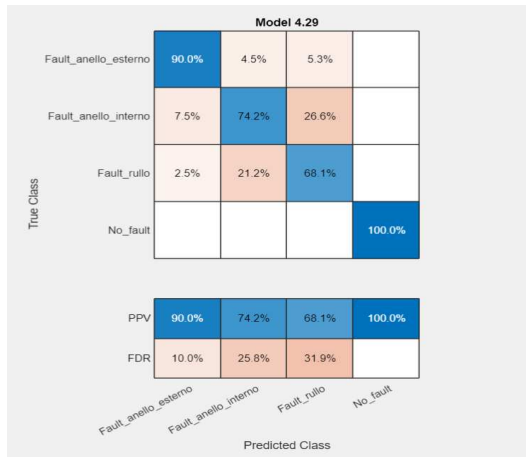


Figura 6.25: Matrici di confusione positive predictive values/false discovery rate train

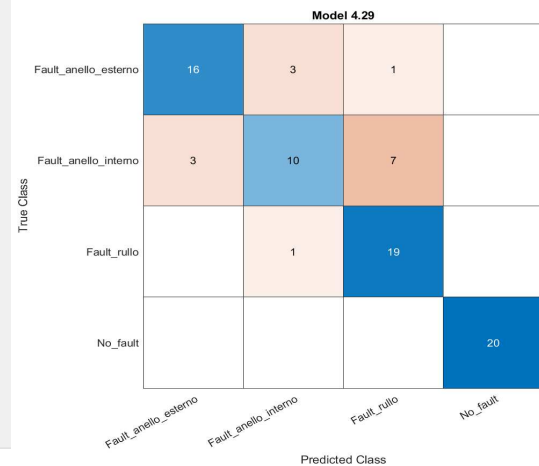


Figura 6.26: Matrici di confusione numero osservazioni test

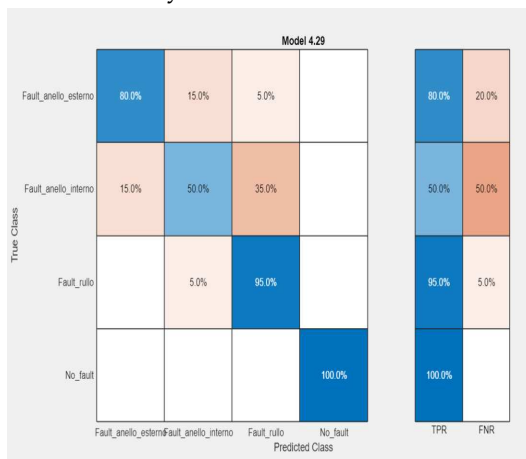


Figura 6.27: Matrici di confusione valutazione true positive/false negative test

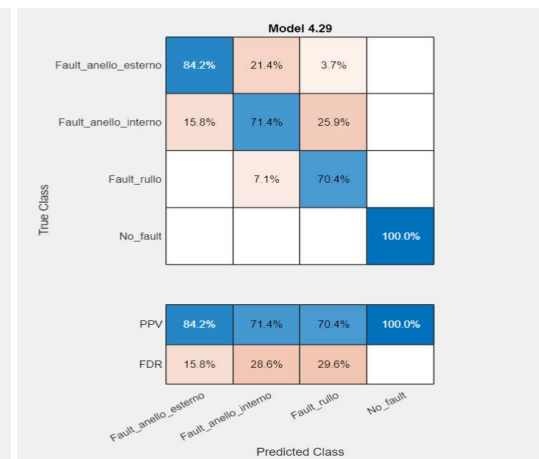


Figura 6.28: Matrici di confusione positive predictive values/false discovery rate test

I risultati ottenuti con le matrici di confusione sia per quanto riguarda il train che per il test, non sono ottime, ma sono comunque accettabili. Ciò in parte è dovuto alla frame policy che si è scelta, infatti scegliendola di 0.2048, si ha che all'incirca i membri delle feature sono 25 che moltiplicato per 4, il numero delle classi, porta come valore 100.

Da questo è stato deciso di dividere i dati, come detto precedentemente in questa casistica, in 80% di train e il 20% di test.

Quello che si nota subito è il fatto che, l'algoritmo non riesce a discriminare molto bene le classi intermedie come la classe del fault nell'anello interno e quella rullo, mentre la classe del fault nell'anello esterno e quella senza fault vengono isolate in modo adeguato.

Mentre per quanto riguarda la valutazione della curva ROC può essere notata nelle figure qui sotto riportate dove si osserva che il valore AUC è compreso tra 0.7 e 0.9 sia per quanto riguarda il test che per il train, il che vuol dire che le prestazioni del classificatore in questione sono buone.

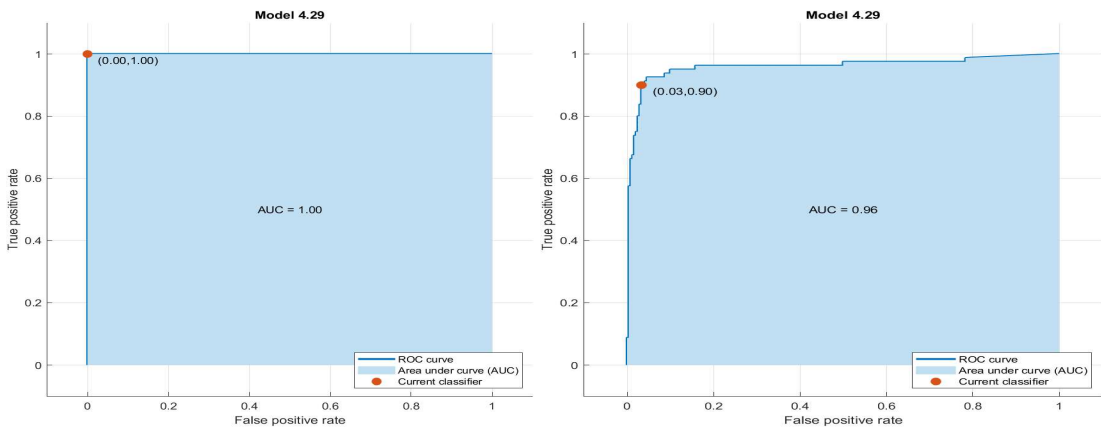


Figura 6.29: Curva ROC no fault del train del segnale Yc

Figura 6.30: Curva ROC fault esterno del train del segnale Yc

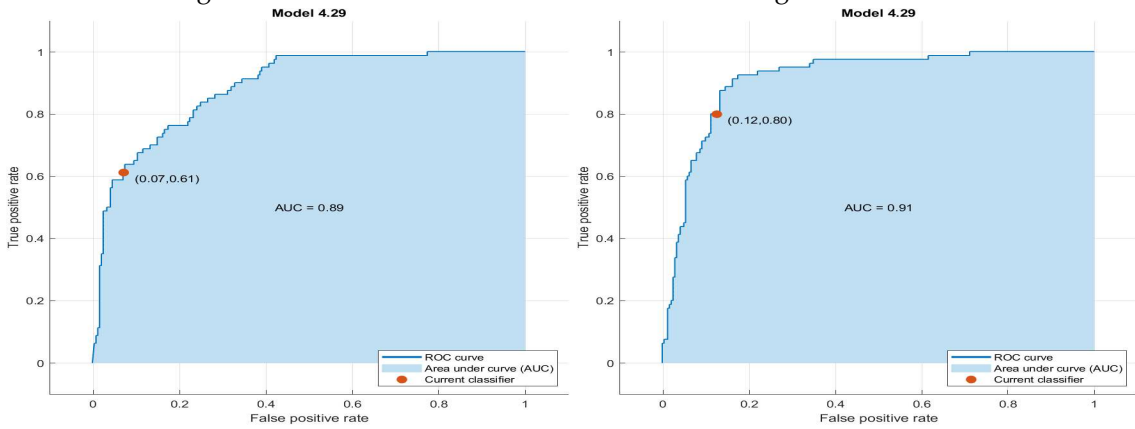


Figura 6.31: Curva ROC fault interno del train del segnale Yc

Figura 6.32: Curva ROC fault rullo del train del segnale Yc

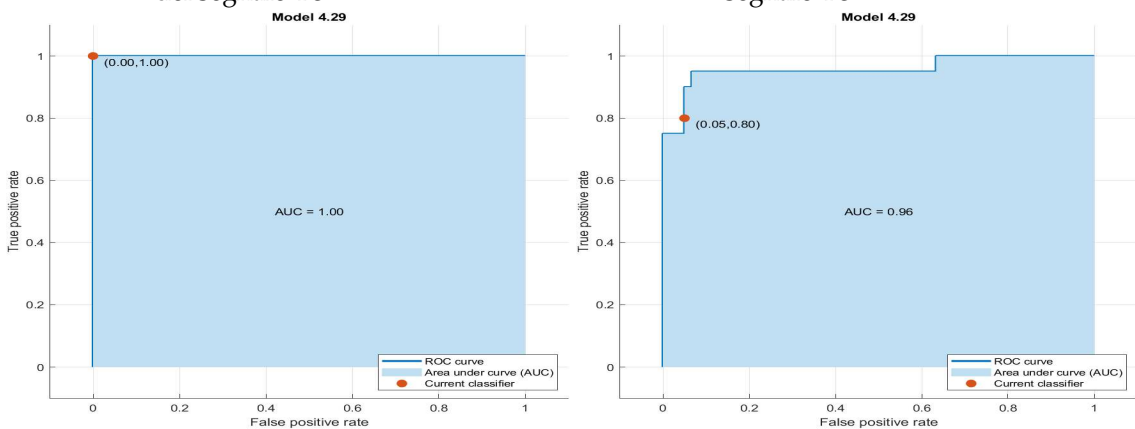


Figura 6.33: Curva ROC no fault del test del segnale Yc

Figura 6.34: Curva ROC fault esterno del test del segnale Yc

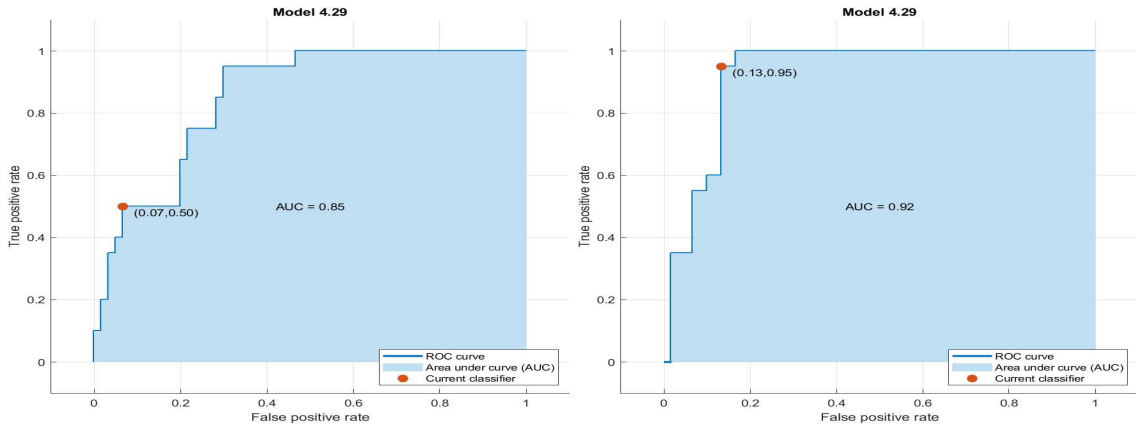


Figura 6.35: Curva ROC fault interno del test del segnale Yc **Figura 6.36:** Curva ROC fault rullo del test del segnale Yc

6.2.2 Analisi segnale Za

Come descritto per il segnale Yc, anche qui il sottocapitolo verrà diviso in due sezioni, una parte descrittiva e una di analisi, inoltre ci sarà una divisione a seconda dello strumento utilizzato. I risultati visivi ottenuti sono simili al sottocapitolo 6.2.1, cambia leggermente, però la sorpresa è stata nel fatto che, per quanto la classificazione, i risultati sono migliori usando un classificatore differente.

Si parte dai risultati ottenuti dal Diagnostic Feature Designer dove in Figura 6.37 si può osservare l'andamento nel tempo del segnale Za.

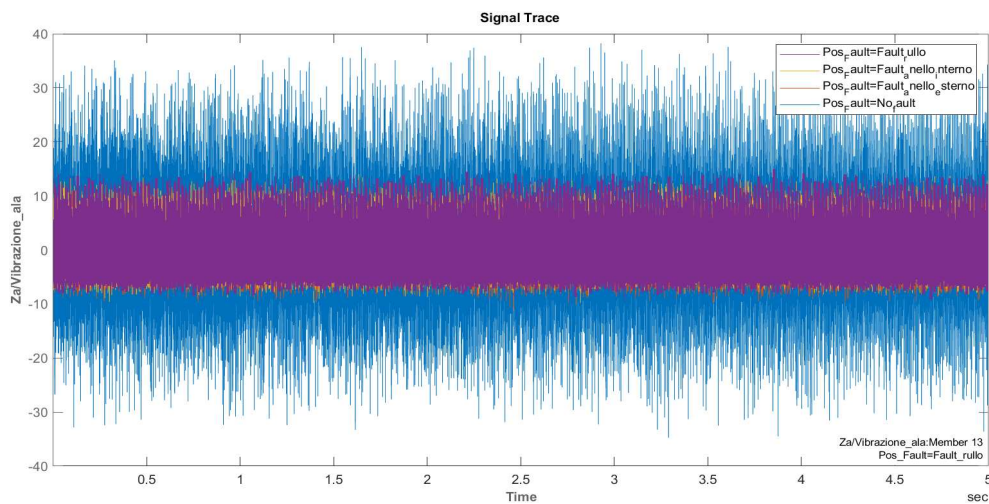


Figura 6.37: Andamento temporale segnale Za

In Figura 6.38 si può osservare una frazione del segnale, dopo una pulizia con la tecnica del TSA, attraverso la frame policy indicata precedentemente.

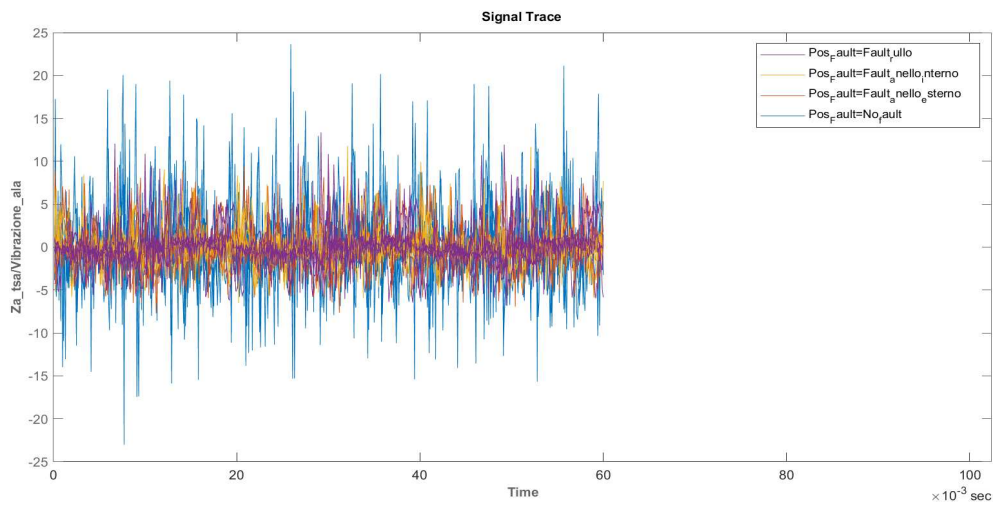


Figura 6.38: Andamento temporale segnale Za con TSA

In Figura 6.39 è stato riprodotto lo spettro del segnale, dopo aver applicato il TSA, nel tentativo di riuscire ad osservare le frequenze calcolate con risultati limitati.

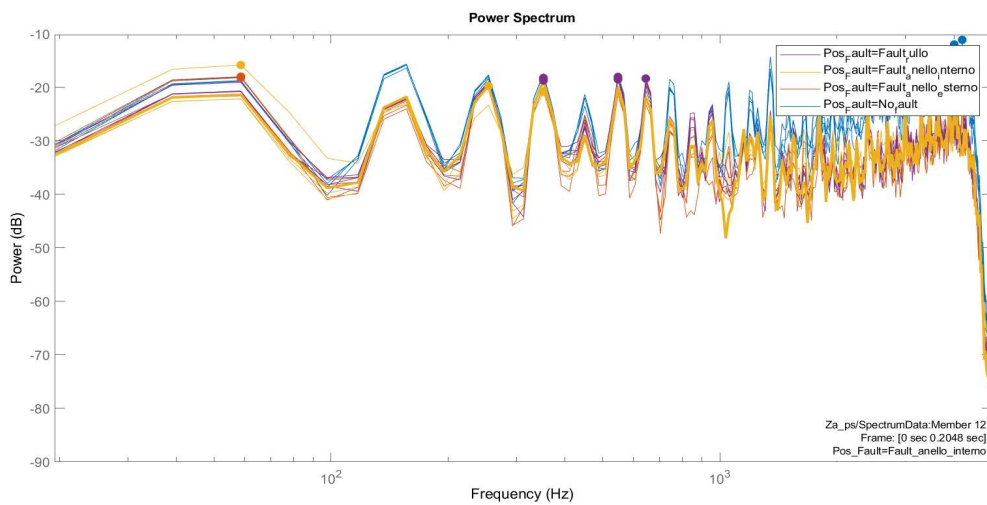


Figura 6.39: Spettro del segnale Za dopo l'applicazione del TSA

In Figura 6.40 e 6.41 si possono osservare gli istogrammi delle feature raggruppate per classe di guasto.

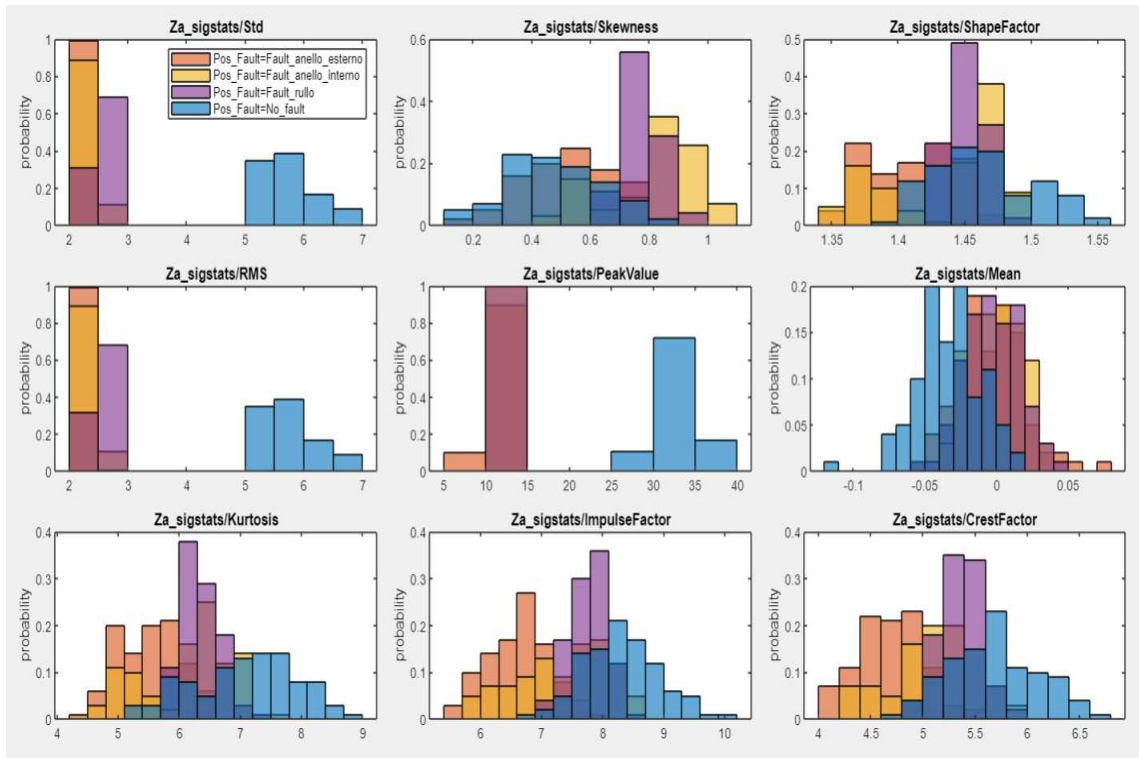


Figura 6.40: Istogrammi Feature Za

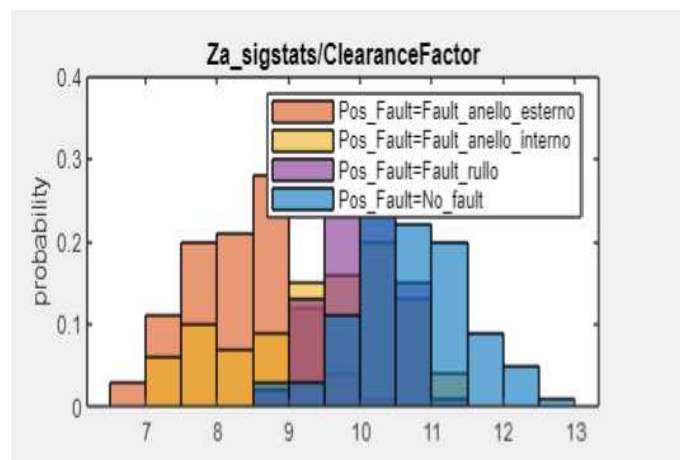


Figura 6.41: Istogramma Feature Za

Come è stato fatto per il segnale Yc, anche nel segnale Za vengono riportate gli istogrammi delle feature raggruppate per classe di guasto in Figura 6.42 e 6.43.

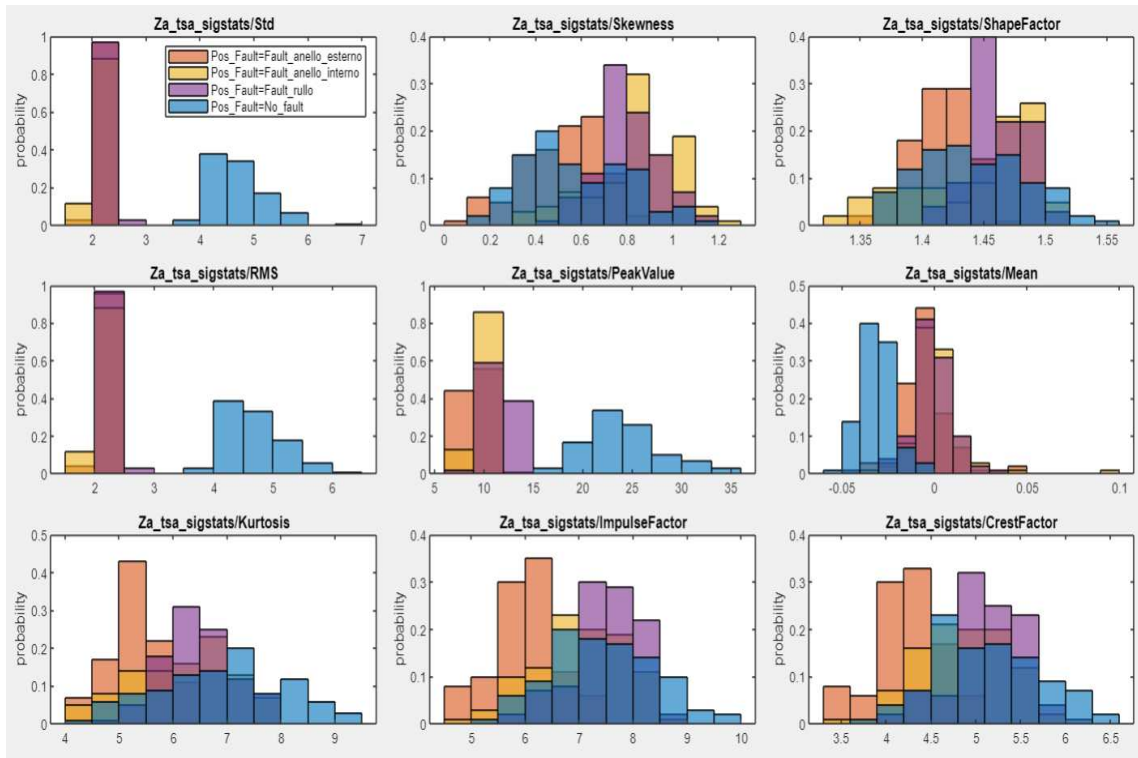


Figura 6.42: Istogrammi Feature Za con TSA

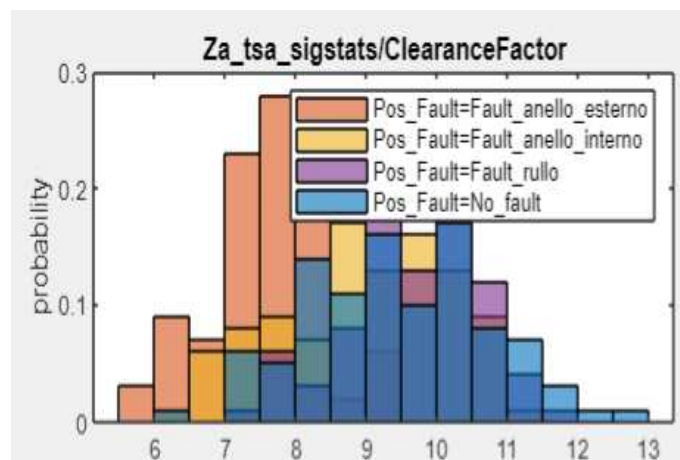


Figura 6.43: Istogramma Feature Za con TSA

Infine nella Figura 6.44 viene calcolato il ranking delle feature, sia con TSA che senza, con il test ANOVA.

Attraverso questo ranking è possibile esportare le feature che hanno valore maggiore e che sono più significative per la classificazione.

Come stato eseguito con il segnale Yc, anche per il segnale Za vengono riportare le 5 principali caratteristiche che hanno avuto il punteggio maggiore; esse sono:

- Impulse factor
- Peak value
- RMS

- Skewness
- Standard deviation

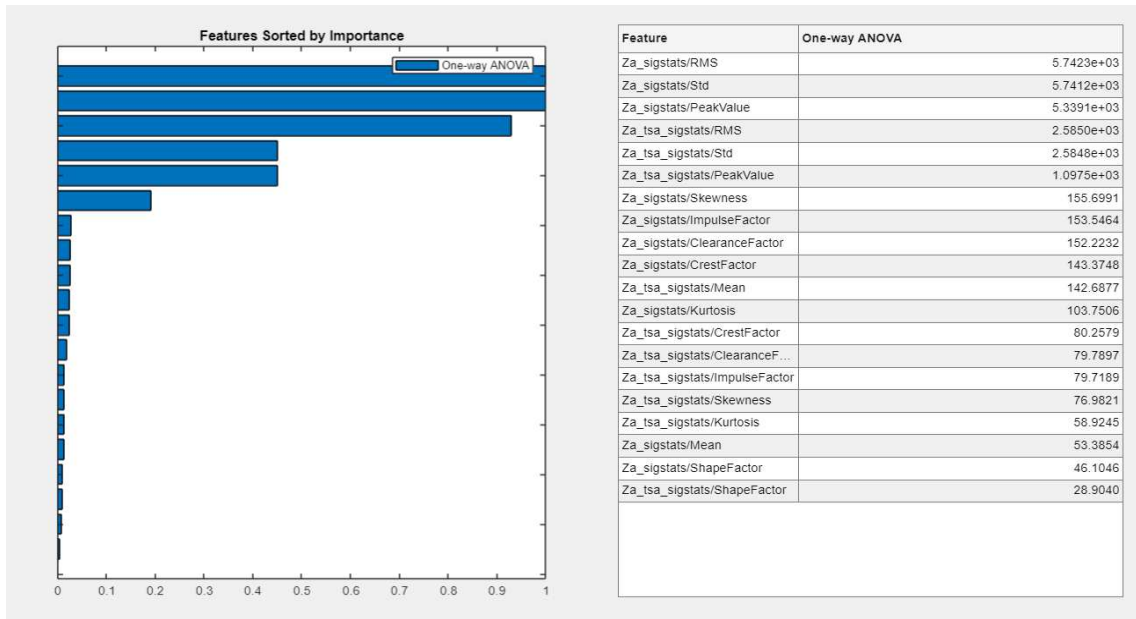


Figura 6.44: Calcolo ranking feature di Za con e senza TSA

Di seguito vengono riportati i risultati ottenuti dal Classification Learner, utilizzando le 5 caratteristiche sopra elencate per la classificazione.

<p>2.1 Tree Accuracy (Validation): 90.9% Last change: Fine Tree 5/5 features</p> <p>2.2 Tree Accuracy (Validation): 90.9% Last change: Medium Tree 5/5 features</p> <p>2.3 Tree Accuracy (Validation): 92.2% Last change: Coarse Tree 5/5 features</p> <p>2.4 Linear Discriminant Accuracy (Validation): 80.0% Last change: Linear Discriminant 5/5 features</p> <p>2.5 Quadratic Discriminant Accuracy (Validation): 93.8% Last change: Quadratic Discriminant 5/5 features</p> <p>2.6 Naive Bayes Accuracy (Validation): 89.4% Last change: Gaussian Naive Bayes 5/5 features</p> <p>2.7 Naive Bayes Accuracy (Validation): 90.9% Last change: Kernel Naive Bayes 5/5 features</p> <p>2.8 SVM Accuracy (Validation): 91.6% Last change: Linear SVM 5/5 features</p> <p>2.9 SVM Accuracy (Validation): 94.7% Last change: Quadratic SVM 5/5 features</p> <p>2.10 SVM Accuracy (Validation): 95.3% Last change: Cubic SVM 5/5 features</p> <p>2.11 SVM Accuracy (Validation): 92.8% Last change: Fine Gaussian SVM 5/5 features</p> <p>2.12 SVM Accuracy (Validation): 92.2% Last change: Medium Gaussian SVM 5/5 features</p> <p>2.13 SVM Accuracy (Validation): 88.8% Last change: Coarse Gaussian SVM 5/5 features</p>	<p>2.20 Ensemble Accuracy (Validation): 25.0% Last change: Boosted Trees 5/5 features</p> <p>2.21 Ensemble Accuracy (Validation): 93.8% Last change: Bagged Trees 5/5 features</p> <p>2.22 Ensemble Accuracy (Validation): 80.9% Last change: Subspace Discriminant 5/5 features</p> <p>2.23 Ensemble Accuracy (Validation): 90.6% Last change: Subspace KNN 5/5 features</p> <p>2.24 Ensemble Accuracy (Validation): 25.0% Last change: RUSBoosted Trees 5/5 features</p> <p>2.25 Neural Network Accuracy (Validation): 92.8% Last change: Narrow Neural Network 5/5 features</p> <p>2.26 Neural Network Accuracy (Validation): 93.4% Last change: Medium Neural Network 5/5 features</p> <p>2.27 Neural Network Accuracy (Validation): 93.1% Last change: Wide Neural Network 5/5 features</p> <p>2.28 Neural Network Accuracy (Validation): 93.8% Last change: Bilayered Neural Network 5/5 features</p> <p>2.29 Neural Network Accuracy (Validation): 93.4% Last change: Trilayered Neural Network 5/5 features</p> <p>2.30 Kernel Accuracy (Validation): 90.0% Last change: SVM Kernel 5/5 features</p> <p>2.31 Kernel Accuracy (Validation): 89.4% Last change: Logistic Regression Kernel 5/5 features</p>
--	---

Figura 6.45: Risultato allenamento della classificazione con diversi algoritmi del segnale Za

2.1 Tree Last change: Fine Tree	Accuracy (Test): 93.8% 5/5 features	2.20 Ensemble Last change: Boosted Trees	Accuracy (Test): 93.8% 5/5 features
2.2 Tree Last change: Medium Tree	Accuracy (Test): 93.8% 5/5 features	2.21 Ensemble Last change: Bagged Trees	Accuracy (Test): 95.0% 5/5 features
2.3 Tree Last change: Coarse Tree	Accuracy (Test): 95.0% 5/5 features	2.22 Ensemble Last change: Subspace Discriminant	Accuracy (Test): 86.2% 5/5 features
2.4 Linear Discriminant Last change: Linear Discriminant	Accuracy (Test): 82.5% 5/5 features	2.23 Ensemble Last change: Subspace KNN	Accuracy (Test): 87.5% 5/5 features
2.5 Quadratic Discriminant Last change: Quadratic Discriminant	Accuracy (Test): 95.0% 5/5 features	2.24 Ensemble Last change: RUSBoosted Trees	Accuracy (Test): 93.8% 5/5 features
2.6 Naive Bayes Last change: Gaussian Naive Bayes	Accuracy (Test): 92.5% 5/5 features	2.25 Neural Network Last change: Narrow Neural Network	Accuracy (Test): 96.2% 5/5 features
2.7 Naive Bayes Last change: Kernel Naive Bayes	Accuracy (Test): 91.2% 5/5 features	2.26 Neural Network Last change: Medium Neural Network	Accuracy (Test): 92.5% 5/5 features
2.8 SVM Last change: Linear SVM	Accuracy (Test): 87.5% 5/5 features	2.27 Neural Network Last change: Wide Neural Network	Accuracy (Test): 95.0% 5/5 features
2.9 SVM Last change: Quadratic SVM	Accuracy (Test): 96.2% 5/5 features	2.28 Neural Network Last change: Bilayered Neural Network	Accuracy (Test): 95.0% 5/5 features
2.10 SVM Last change: Cubic SVM	Accuracy (Test): 91.2% 5/5 features	2.29 Neural Network Last change: Trilayered Neural Network	Accuracy (Test): 92.5% 5/5 features
2.11 SVM Last change: Fine Gaussian SVM	Accuracy (Test): 91.2% 5/5 features	2.30 Kernel Last change: SVM Kernel	Accuracy (Test): 87.5% 5/5 features
2.12 SVM Last change: Medium Gaussian SVM	Accuracy (Test): 92.5% 5/5 features	2.31 Kernel Last change: Logistic Regression Kernel	Accuracy (Test): 88.8% 5/5 features
2.13 SVM Last change: Coarse Gaussian SVM	Accuracy (Test): 90.0% 5/5 features		

Figura 6.46: Risultato dei test della classificazione con diversi algoritmi del segnale Za

Dalle informazioni delle Figura 6.45 e 6.46 è stato deciso di riportare i dati ottenuti dal classificatore con il valore di accuratezza maggiore, il quale è il **Quadratic SVM**. Nell'immagine 6.47 si possono osservare le informazioni per quanto riguarda il valore di accuratezza del train e del test; inoltre è presente una tabella che rappresenta una finestra di dialogo, che mostra i costi di classificazione errata predefiniti (matrice dei costi) come una tabella con etichette di righe e colonne determinate dalle classi nella variabile di risposta. Le righe della tabella corrispondono alle classi reali e le colonne corrispondono alle classi previste. È possibile interpretare la matrice dei costi in questo modo: la voce nella riga i e nella colonna j è il costo della classificazione errata delle osservazioni della i -esima classe nella j -esima classe. Le voci diagonali della matrice dei costi devono essere 0 e le voci fuori diagonale devono essere numeri reali non negativi.

Model 2.9: SVM
Status: Trained

Training Results
Accuracy (Validation) 94.7%
Total cost (Validation) 17
Prediction speed ~7700 obs/sec
Training time 3.1353 sec

Test Results
Accuracy (Test) 96.2%
Total cost (Test) 3

Model Hyperparameters
Preset: Quadratic SVM
Kernel function: Quadratic
Kernel scale: Automatic
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection: 5/5 individual features selected

PCA: Disabled

Misclassification Costs: Default

		Predicted Class			
		Fault_anello_esterno	Fault_anello_interno	Fault_rullo	No_fault
True Class	Fault_anello_esterno	0	1	1	1
	Fault_anello_interno	1	0	1	1
	Fault_rullo	1	1	0	1
	No_fault	1	1	1	0

Optimizer: Not applicable

Figura 6.47: Informazioni allenamento e test del modello quadratic SVM del segnale Za

Nell'immagini sottostanti vengono riportate le matrici di confusione sia del training che del test del segnale Za.

Come si può notare fin da subito, i risultati sono migliori come già introdotto precedentemente. Questo classificatore riesce a discriminare meglio le classi rispetto a quello utilizzato nel segnale Yc, il che non vuol dire che un classificatore è meglio dell'altro, ma che i dati sono distribuiti in modo migliore.

Le classi intermedie dove era difficile classificarle nel segnale Yc, con questo classificatore vengono classificate in modo adeguato.

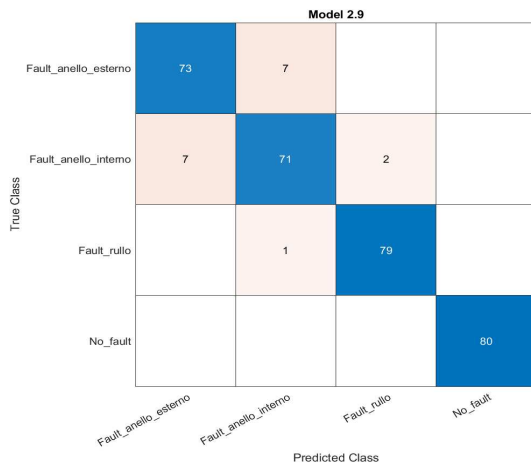


Figura 6.48: Matrici di confusione numero osservazioni train

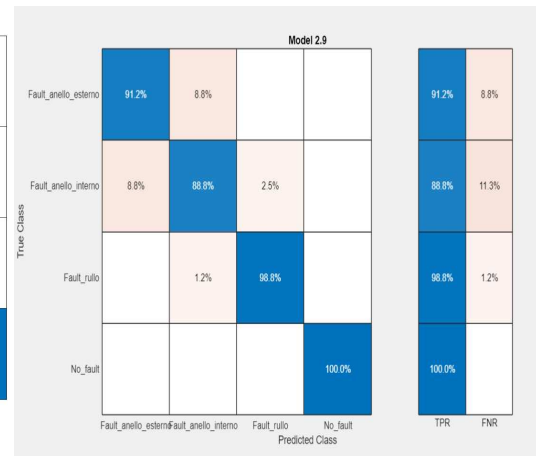


Figura 6.49: Matrici di confusione valutazione true positive/false negative train

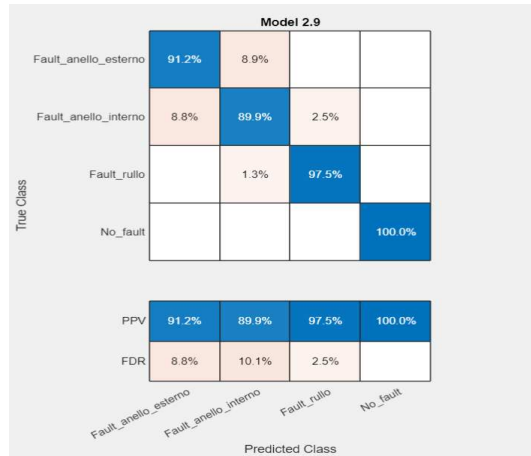


Figura 6.50: Matrici di confusione positive predictive values/false discovery rate train

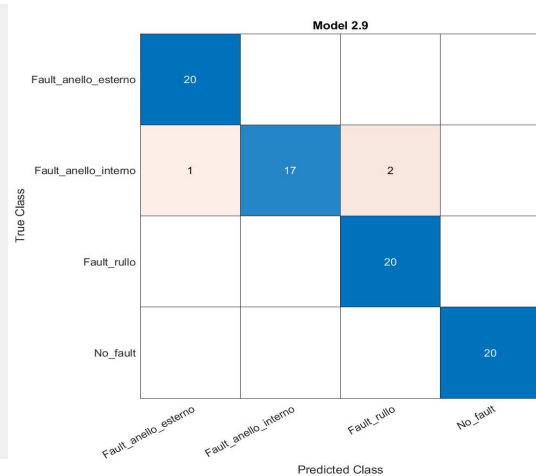


Figura 6.51: Matrici di confusione numero osservazioni test

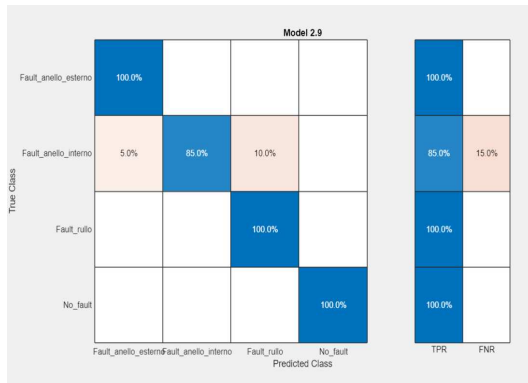


Figura 6.52: Matrici di confusione valutazio-
ne true positive/false negative test

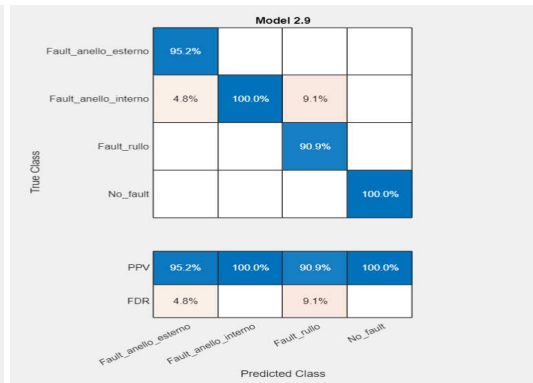


Figura 6.53: Matrici di confusione positive predictive values/false discovery rate test

Nelle figure sottostanti vengono riportate i valori ottenuti per quanto riguarda la curva ROC e AUC.

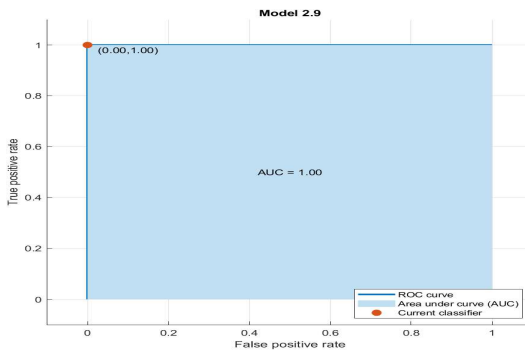


Figura 6.54: Curva ROC no fault del train del segnale Za

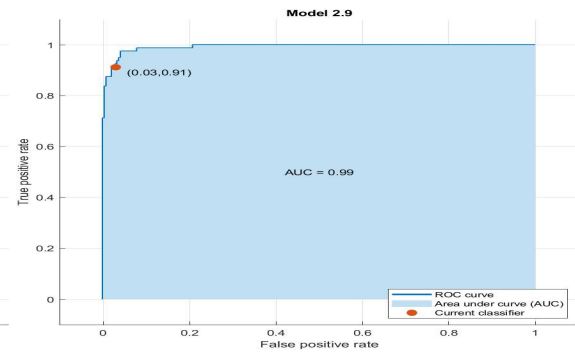


Figura 6.55: Curva ROC fault esterno del train del segnale Za

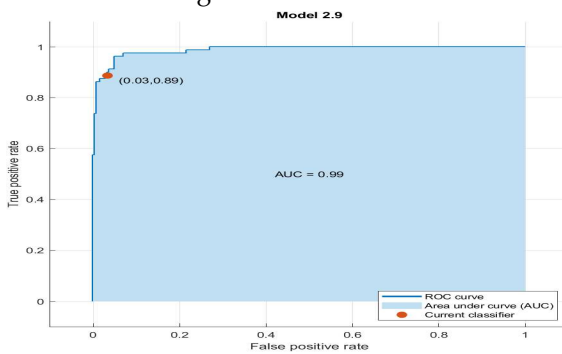


Figura 6.56: Curva ROC fault interno del train del segnale Za

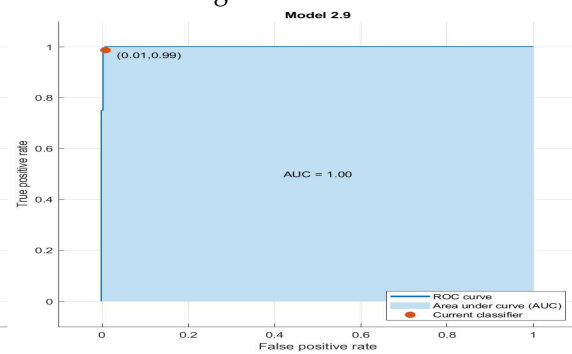


Figura 6.57: Curva ROC fault rullo del train del segnale Za

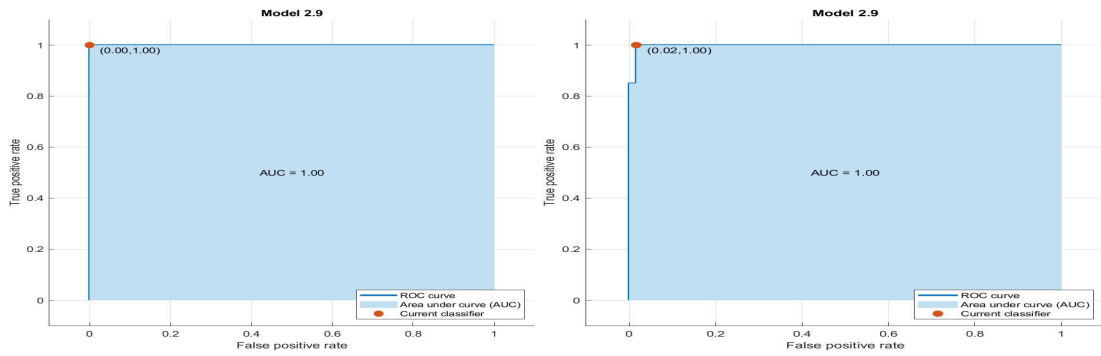


Figura 6.58: Curva ROC no fault del test del segnale Za

Figura 6.59: Curva ROC fault esterno del test del segnale Za

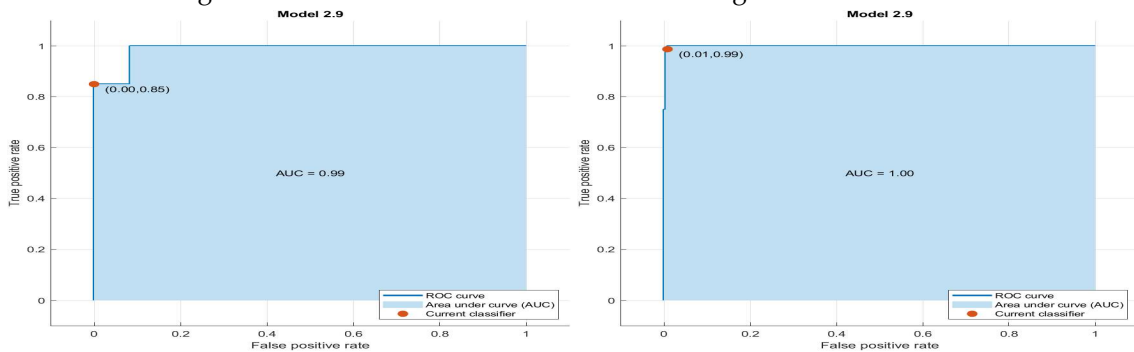


Figura 6.60: Curva ROC fault interno del test del segnale Za

Figura 6.61: Curva ROC fault rullo del test del segnale Za

Come accennato inizialmente nel sottocapitolo, anche per quanto riguarda il valore che si della metrica AUC (area sotto la curva) si ha un notevole miglioramento e i valori sono vicini al numero 1, il che vuol dire che la performance del classificatore utilizzato è ottima.

6.3 Risultati Python

In questa sezione è stato considerato il dataset grezzo e quindi non è stata calcolata alcuna feature che poteva derivare da una conoscenza teorica ed è stata eseguita la classificazione dei dati grezzi. Come per le sezioni trattate precedentemente è divisa in due parti:

- Nella prima parte si tratterà dell'analisi del segnale Yc.
- Nella seconda parte si parlerà dei risultati del segnale Za.

Come descritto precedentemente si avrà una breve parte di visualizzazione e descrizione e poi l'analisi effettiva, questo per ciascuno segnale.

6.3.1 Risultati segnale Yc

Inizialmente è stata realizzata una visualizzazione dei dati grezzi anche in Python. La Figura 6.62 rappresenta l'andamento nel tempo dei 16 esperimenti eseguiti ed ognuno è rappresentato da un colore differente.

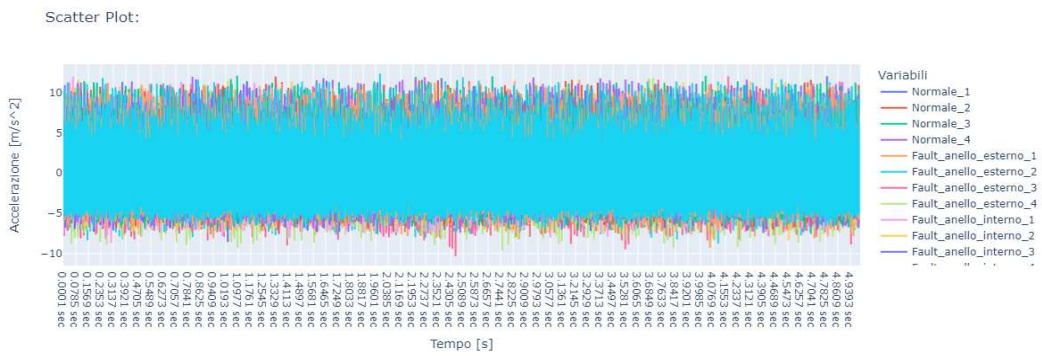


Figura 6.62: Andamento temporale del segnale Yc con i suoi 16 esperimenti

La Figura 6.63 rappresenta uno scatter plot dove nelle ordinate si ha l’accelerazione, mentre nelle ascisse si ha la dimensionalità del dataset, cioè viene rappresentato la dimensione totale del dataset, in questo caso 800000, diviso per classe di guasto dove il colore blu rappresenta l’assenza di fault, il colore arancione il guasto all’anello esterno, il colore verde il difetto all’anello interno e il colore rosso il fault alla sfera. Già da questa analisi, come nelle altre effettuate in Matlab, si nota una differenza tra il segnale senza fault e i segnali con i fault.

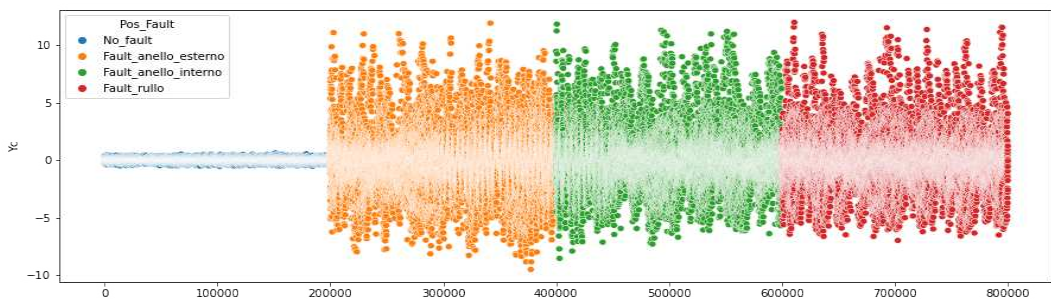


Figura 6.63: Scatter plot del segnale Yc con le sue 16 prove

Nella Figura 6.64 viene rappresentato un grafico interattivo in Python.

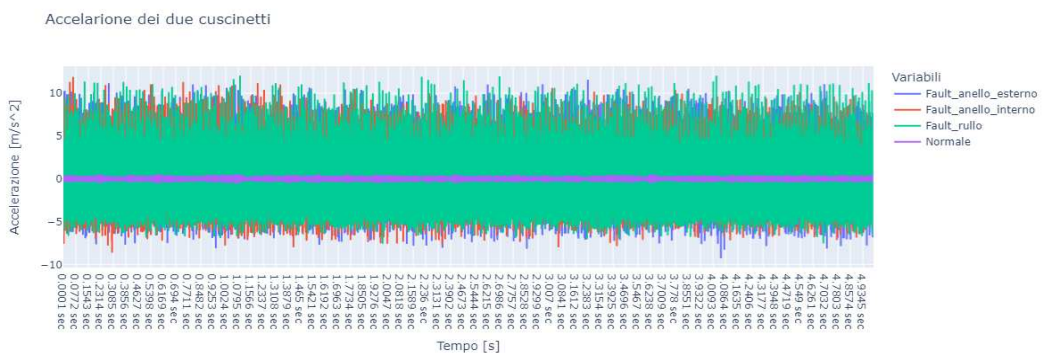


Figura 6.64: Andamento del segnale Yc per tipologia di prova

Successivamente è stata eseguita la classificazione del dataset nelle 4 tipologie di classi differenti. In Figura 6.65 vengono riportati i risultati dove:

- **Loss:** indica il training loss e risalta quanto bene il modello si adatta ai dati di train.

- **Accuracy:** mostra l'accuratezza del train, la quale è l'accuratezza di un modello rispetto ai dati su cui è stato addestrato.
- **Val_loss :** rappresenta il validation loss e segnala quanto bene il modello si adatta ai nuovi dati.
- **Val_accuracy:** delinea la validation accuracy e denota l'accuratezza di un modello sui nuovi dati.

Come si nota dalla figura, in questa prova sono state considerate 10 epoche e una dimensione del batch di 32, quella di default di Keras. Inoltre, il dataset è stato testato in diversi modi tra dati di training e di test. I risultati che vengono riportati, utilizzano un 20% dei dati per il training e un 80% dei dati per il testing, per osservare se ci potesse essere il problema dell'overfitting, dove quest'ultimo si verifica quando il modello non può essere generalizzato e si adatta in modo eccessivo al set di dati di addestramento.

Dalla figura si nota che all'aumentare del valore delle epoche aumenta anche il valore di accuratezza del train e del test.

```
Epoch 1/10
c:\Users\eris4\AppData\Local\Programs\Python\Python39\lib\site-packages\tensorflow\python\data\ops\structured_function.py:265: UserWarning: warnings.warn(
17/17 [=====] - 4s 215ms/step - loss: 4.5347 - accuracy: 0.2500 - val_loss: 1.4878 - val_accuracy: 0.2505
Epoch 2/10
17/17 [=====] - 3s 191ms/step - loss: 1.7358 - accuracy: 0.3214 - val_loss: 1.1885 - val_accuracy: 0.5949
Epoch 3/10
17/17 [=====] - 4s 217ms/step - loss: 1.0342 - accuracy: 0.7049 - val_loss: 0.8039 - val_accuracy: 0.8111
Epoch 4/10
17/17 [=====] - 4s 220ms/step - loss: 0.8184 - accuracy: 0.8383 - val_loss: 0.6757 - val_accuracy: 0.9070
Epoch 5/10
17/17 [=====] - 4s 225ms/step - loss: 0.5621 - accuracy: 0.9361 - val_loss: 0.5708 - val_accuracy: 0.9398
Epoch 6/10
17/17 [=====] - 4s 249ms/step - loss: 0.4328 - accuracy: 0.9774 - val_loss: 0.4757 - val_accuracy: 0.9478
Epoch 7/10
17/17 [=====] - 4s 271ms/step - loss: 0.2371 - accuracy: 0.9906 - val_loss: 0.3229 - val_accuracy: 0.9539
Epoch 8/10
17/17 [=====] - 4s 248ms/step - loss: 0.0950 - accuracy: 0.9925 - val_loss: 0.2310 - val_accuracy: 0.9450
Epoch 9/10
17/17 [=====] - 4s 235ms/step - loss: 0.0606 - accuracy: 0.9850 - val_loss: 0.2167 - val_accuracy: 0.9652
Epoch 10/10
17/17 [=====] - 5s 280ms/step - loss: 0.0856 - accuracy: 0.9774 - val_loss: 0.2163 - val_accuracy: 0.9417
```

Figura 6.65: Valori accuratezza delle epoche

Infine è stata riportata la matrice di confusione per avere una rappresentazione migliore tra le classi che sono state predette in modo corretto. Ciò lo si può notare in Figura 6.66, dove nella diagonale si hanno le classi predette correttamente, mentre fuori si hanno le classi predette in modo non corretto.

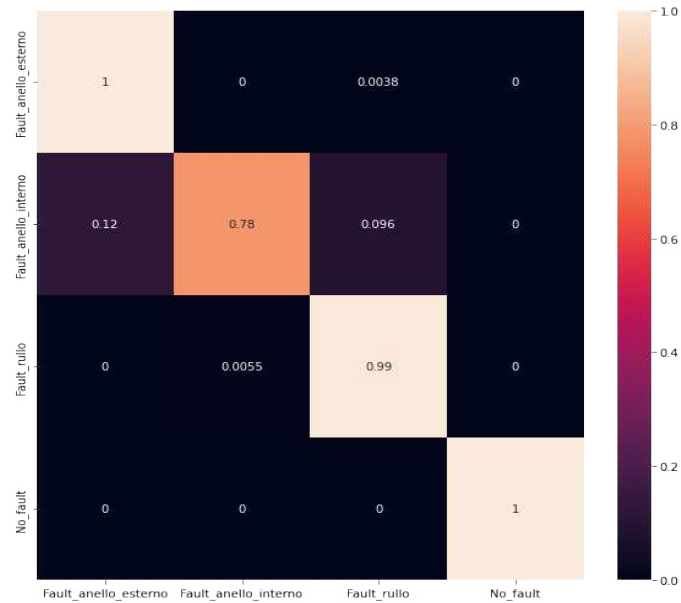


Figura 6.66: Matrice di confusione

6.3.2 Risultati segnale Za

Come è stato effettuato per il segnale Yc, anche per Za vengono riportati i risultati dell’analisi dove una prima parte è stata dedicata alla visualizzazione e una secondo parte ai risultati della classificazione.

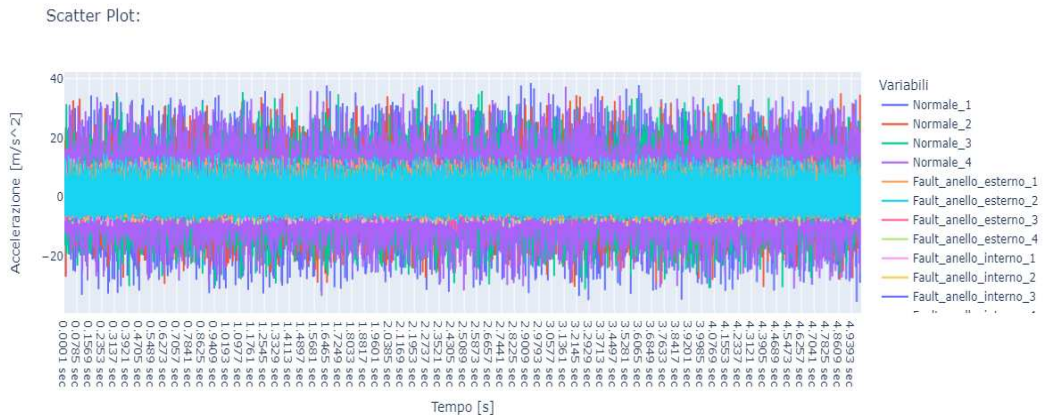


Figura 6.67: Andamento temporale del segnale Za con i suoi 16 esperimenti

Quello che si nota di diverso dal segnale Yc dalla Figura 6.68 è che l’ampiezza del segnale quando non si ha fault è maggiore rispetto alle ampiezze dei segnali con la presenza di fault; mentre per il resto, cioè i valori di accuratezza e la matrice di confusione, i risultati ottenuti sono più o meno simili al segnale Yc.

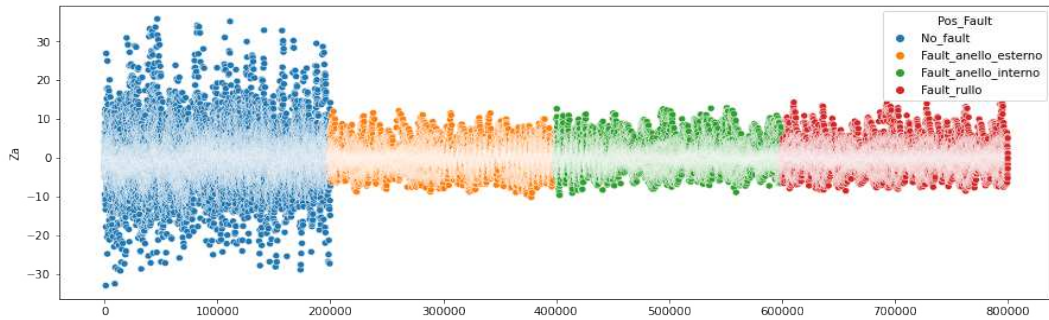


Figura 6.68: Scatter plot del segnale Z_a con le sue 16 prove

Nella Figura 6.69 tramite un grafico interattivo viene riportato l’andamento del segnale Z_a per tipologica di prova.

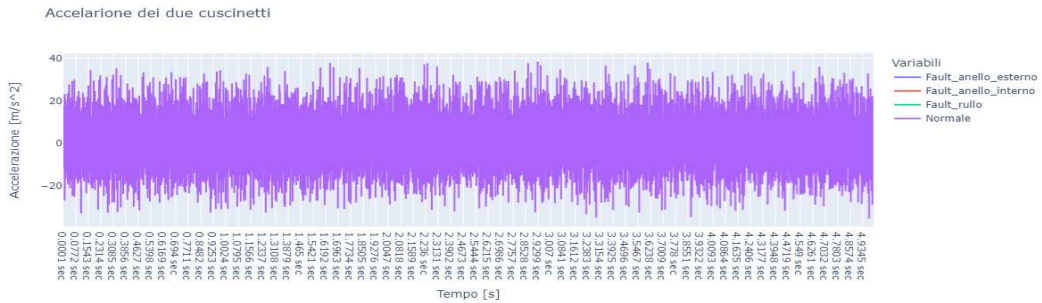


Figura 6.69: Andamento del segnale Z_a per tipologia di prova

La Figura 6.70 rappresenta i valori di accuratezza, che aumentano all’aumentare delle epoche.

```
Epoch 1/10
1/17 [>.....] - ETA: 1s - loss: 1.6691 - accuracy: 0.2500
c:\Users\eris4\AppData\Local\Programs\Python\Python39\lib\site-packages\tensorflow\python\data\ops\structured_function.py:265: UserWarning: warnings.warn(
17/17 [=====] - 4s 217ms/step - loss: 4.6356 - accuracy: 0.3365 - val_loss: 1.2486 - val_accuracy: 0.4135
Epoch 2/10
17/17 [=====] - 4s 215ms/step - loss: 0.9584 - accuracy: 0.5526 - val_loss: 0.9041 - val_accuracy: 0.6734
Epoch 3/10
17/17 [=====] - 4s 217ms/step - loss: 0.5807 - accuracy: 0.8158 - val_loss: 0.6810 - val_accuracy: 0.7951
Epoch 4/10
17/17 [=====] - 3s 200ms/step - loss: 0.3753 - accuracy: 0.8947 - val_loss: 0.5185 - val_accuracy: 0.8576
Epoch 5/10
17/17 [=====] - 3s 195ms/step - loss: 0.2884 - accuracy: 0.9305 - val_loss: 0.4293 - val_accuracy: 0.9051
Epoch 6/10
17/17 [=====] - 4s 220ms/step - loss: 0.2120 - accuracy: 0.9549 - val_loss: 0.3773 - val_accuracy: 0.9215
Epoch 7/10
17/17 [=====] - 4s 235ms/step - loss: 0.1586 - accuracy: 0.9774 - val_loss: 0.3272 - val_accuracy: 0.9403
Epoch 8/10
17/17 [=====] - 4s 233ms/step - loss: 0.1432 - accuracy: 0.9737 - val_loss: 0.3178 - val_accuracy: 0.9159
Epoch 9/10
17/17 [=====] - 4s 238ms/step - loss: 0.0916 - accuracy: 0.9887 - val_loss: 0.3032 - val_accuracy: 0.9046
Epoch 10/10
17/17 [=====] - 4s 234ms/step - loss: 0.0867 - accuracy: 0.9850 - val_loss: 0.2295 - val_accuracy: 0.9671
```

Figura 6.70: Valori accuratezza delle epoche segnale Z_a

In Figura 6.71 viene riportata, come fatto per il segnale Yc, la matrice di confusione del segnale Za.

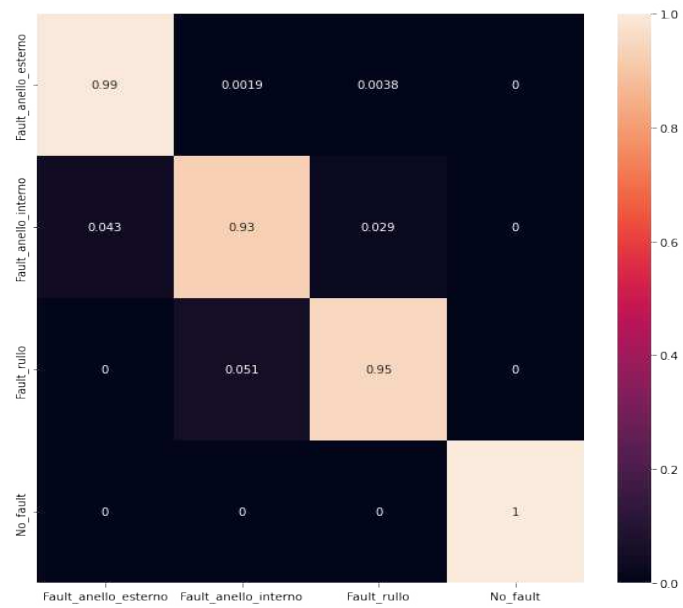


Figura 6.71: Matrice di confusione Za

Conclusioni

In questa parte finale verrà trattato le conclusioni, i problemi che si sono riscontrati e implementazioni future

Il progetto sviluppato è stato utile nell'apprendimento di nuove tecniche per riuscire a rilevare un guasto su un componente, in modo da dare la possibilità all'azienda di intervenire, diminuendo i costi sia per quanto riguarda le tempistiche sia per il valore economico. Infatti, intervenendo anticipatamente, è possibile che lo stato di degradazione del componente non arrivi ad un valore critico, di rottura, il che potrebbe causare danni maggiori non solo al macchinario, ma anche all'intero sistema. I risultati ottenuti sono di fondamentale importanza per sviluppi ulteriori in futuro, infatti gli obiettivi fissati inizialmente, cioè quello di classificare le differenti classi di fault, hanno avuto esito positivo. Inizialmente l'obiettivo era quello di osservare direttamente le frequenze dei fault e del normale andamento senza guasti, tutto questo nel dominio della frequenza; ciò è stato possibile in parte, ovvero discriminare solo quando il guasto era presente o meno, ma a causa di alcuni problemi non è stato fattibile identificare la classe di guasto. Principalmente, sono state formulate due ipotesi:

- La prima, consiste nella possibilità che il sistema di acquisizione non abbia un circuito di condizionamento adeguato per le prove effettuate, infatti dai grafici risultano rappresentate differenti frequenze, tra cui troviamo anche la frequenza di rete a 50Hz e le sue armoniche dispari. Inoltre, poiché gli esperimenti sono stati svolti in officina si hanno differenti disturbi che influenzano il segnale utile e sommandosi ad esso, non è possibile distinguere visivamente tra frequenze utili e non. Quindi con un circuito adeguato, forse si potevano trovare risultati migliori nell'analisi in frequenza.
- La seconda ipotesi consiste nel fatto che il canotto su cui sono stati montati i cuscinetti, che funge da albero motore, è stato modificato. Infatti, dopo i lavori di ritocco, come la creazione di alcuni fori per posizionare i sensori e alcuni lavori di tornitura per posizionare i cuscinetti, il canotto sia diventato troppo equilibrato e quindi le frequenze di interesse non comparivano nei grafici sviluppati.

Sarebbe stato interessante capire il motivo di quanto detto precedentemente, ma a causa del tempo limitato e delle tempistiche prolungate, per quanto riguarda l'ottenimento dei materiali e delle risorse umane per eseguire il montaggio e smontaggio dei cuscinetti, ciò non è stato possibile. Per colmare questo problema si è pensato di trovare soluzioni alternative, in modo da non restare inattivi, di conseguenza l'obiettivo che era inizialmente la classificazione

tramite la visualizzazione delle frequenze è cambiato in classificazione attraverso algoritmi data driven. Le soluzioni alternative prevedono l'utilizzo di strumenti diversi, anche per osservare se al termine ci potesse essere una similarità tra i risultati ottenuti con strumenti diversi. Le soluzioni alternative sono due:

- La prima consiste nell'utilizzare le applicazioni di Matlab come Diagnostic Feature Designer e Classification Learner per calcolare determinate feature del segnale, sono state scelte le 5 con valore maggiore secondo il test ANOVA, e classificare quest'ultime. Infatti, dopo diverse prove, si è visto che classificare direttamente il segnale grezzo su Matlab non era fattibile perché venivano ottenuti percentuali di accuratezza inferiori al 50%. Mentre calcolando le feature è stato possibile avere, con alcuni algoritmi di classificazione, valori importanti di accuratezza e per cui era possibile distinguere le varie classi. In Matlab inizialmente, si era pensato che il segnale Yc avrebbe fornito dei risultati migliori rispetto al segnale Za, essendo posizionato più vicino al cuscinetto, ma ciò non si è verificato. Con il segnale Yc si ha avuto un valore di accuratezza poco sopra l'80%, mentre con il segnale Za, il valore di accuratezza superava il 90%. Infatti, con quest'ultimo le classi venivano classificate in modo corretto.
- L'altra opzione è stata utilizzare Python, attraverso l'importazione di alcune librerie come Keras e TensorFlow è stato possibile eseguire la classificazione dei dati grezzi utilizzando le reti neurali. I risultati ottenuti già dopo poche epoche sono di notevole importanza. Successivamente le informazioni ottenute in Python con i dati grezzi sono state confrontate con le informazioni ottenute in Matlab delle feature per un confronto.

Confrontando i due risultati, quelli di Matlab e quelli di Python, si ottiene che quelli con il calcolo delle caratteristiche sono più trasparenti perché si basano su una conoscenza fisica e matematica, dato che si utilizzano equazioni per calcolare le feature. Nonostante ciò si possono utilizzare entrambi metodi senza alcun problema.

Essendo l'argomento così attuale e così interessante, può avere sviluppi futuri innumerevoli, ma ne verranno elencate sono alcune:

- Sviluppare una scheda di acquisizione dati, da posizionare insieme ad un sensore vicino al macchinario e attraverso la quale è possibile mandare dati in tempo reale, applicando diversi algoritmi con caratteristiche diverse per classificare i tipi di guasto che possono presentarsi nell'entità. In questo modo si può calcolare il tempo di vita rimanente del componente per svolgere la manutenzione predittiva. Tutto questo, per ridurre sia i costi di manutenzione che di riparazione, ma anche i tempi, infatti un macchinario più sta fermo per la manutenzione e più il costo da parte dell'azienda aumenta.
- Miglioramento del codice per integrarlo con i software che l'azienda già possiede in modo da eseguire analisi differenti e intervenire appena un problema viene riscontrato.
- Sviluppo di un'applicazione, sia per pc che per dispositivi mobile, con codice integrato in modo da gestire i diversi macchinari di un'azienda da remoto, dove per gestione si intende la possibilità di avere differenti grafici sull'andamento temporale della vita del macchinario e la possibilità di intervenire in modo tempestivo anche da applicazione, per interrompere il processo in casi di emergenza.
- Acquisizione dei dati da aggiungerne in più al set di addestramento. L'aumento dei dati può contribuire a migliorare la capacità del modello di generalizzare su una vasta gamma di condizioni di funzionamento dei cuscinetti, inoltre si potrebbero avere più dati di diversi macchinari così da riuscire ad effettuare manutenzione predittiva su diversi componenti. Questo offre la possibilità, di effettuare prognosi quando si utilizzano macchinari simili anche se non si possiedono dati di quella specifica entità.

- Espansione a diverse condizioni di carico e vibrazioni. Se possibile, ampliare il set di dati per includere diverse condizioni di carico e vibrazioni, ciò può migliorare la capacità del modello di generalizzare su un'ampia gamma di situazioni operative. Questo ampliamento può essere fatto con diversi componenti, per avere un dataset il più completo possibile.
- Utilizzo di tecniche specifiche per rilevare nuove anomalie, in modo da poter identificare pattern inusuali, i quali potrebbero indicare nuovi fault non precedentemente classificati.
- Una volta che vengono acquisiti dati in grande numero, lavorare su di essi diventa oneroso sia dal punto di vista computazionale che di memoria, quindi sarebbe utile implementare degli algoritmi per la riduzione della dimensionalità, come l'analisi delle componenti principali (PCA), per semplificare la complessità dei dati e migliorare l'efficienza computazionale.
- Inoltre, quando si hanno tante informazioni e il sistema è in stato di funzionamento nominale si potrebbero monitorare le prestazioni nel tempo, infatti si possono implementare dei sistemi di monitoraggio delle prestazioni nel tempo, per rilevare eventuali decadimenti delle prestazioni del modello nel corso del tempo e aggiornare il modello di conseguenza.

- ANTONIO GULLI, S. P., AMITA KAPOOR (2019), *Deep Learning with TensorFlow 2 and Keras: Regression, ConvNets, GANs, RNNs, NLP, and more with TensorFlow 2 and the Keras API*, Packt Publishing. (Cited at pages 1, 38 e 39)
- AUGUSTO DI BENEDETTO, N. P. B. (2007), *Fondamenti di teoria delle vibrazioni meccaniche. Sistemi ad un grado di libertà*, CEA. (Cited at pages 8, 15 e 19)
- G. VACHTSEVANOS, M. R. A. H. B. W., F. L. LEWIS (2006), *Intelligent fault diagnosis and prognosis for engineering systems*, Wiley. (Cited at pages 2 e 22)
- ISERMANN, R. (2006), *Fault Diagnosis Systems An Introduction from Fault Detection to Fault Tolerance*, Springer Verlag Berlin. (Cited at page 13)
- ISERMANN, R. (2011), *Fault Diagnosis Applications Model based Conditon Monitoring Actuators Drives, Machinery Plants Sensors and Fault tolerant Systems*, Springer.
- L. H. CHIANG, E. L. R., R. D. BRAATZ (2000), *Fault Detection and Diagnosis in Industrial Systems*, Springer-Verlag London.
- LIU RUICHEN, C. S. H. H., TANG ZHENHAO (2021), «Fast Fault Identification Method of Wind Turbine Rolling Bearing based on VMD and Time Domain and Frequency Domain Feature Selection», *2021 33rd Chinese Control and Decision Conference (CCDC)*.
- LUCIANO FURLANETTO, M. M., MARCO GARETTI (2022), *Ingegneria della manutenzione. Strategie e metodi*, Franco Angeli. (Cited at page 5)
- MANUEL BAUERI, J. W. F. W. M. K., NETHARASAN BALARATNAM (2022), «Comparison of envelope demodulation methods in the analysis of rolling bearing damage», *Journal of Vibration and Control*.
- MARCO COPPALONI, M. P. (2010-2011), *Diagnostica di cuscinetti volventi mediante signal processing di segnali di vibrazione, corrente e tensione*, Tesi di laurea, Facoltà di Ingegneria Meccanica dell'Università di Milano.
- MARCO LUISE, G. M. V. (2009), *Teoria dei segnali*, McGraw-Hill Education.
- MATHWORKS (2019), «Predictive Maintenance», Rap. tecn., MatLab. (Cited at pages iv, 3 e 4)
- PAOLO ATZENI, P. F. S. P. R. T., STEFANO CERI (2023), *Basi di dati*, McGraw-Hill Education. (Cited at page 1)

- POWER-MI (2022), «Introduction to maintenance», Rap. tecn., Powe-MI. (Cited at pages iv, 4, 5, 6, 7, 8, 10, 11, 12, 15, 17, 18, 19, 20, 24, 26 e 27)
- TEDRIC A. HARRIS, M. N. K. (2006), *Essential Concepts of Bearing Technology*, CRC Press. (Cited at page 17)
- VEROLINO, L. (2019), *Elementi di macchine elettriche*, Edises. (Cited at page 16)
- XIAOMING HAN, S. S. J. Z., JIN XU (2022), «Crack fault diagnosis of vibration exciter rolling bearing based on genetic algorithm–optimized Morlet wavelet filter and empirical mode decomposition», *International Journal of Distributed Sensor Networks*.
- YA HE, K. F. Z. J., MINGHUI HU (2021), «Bearing Condition Evaluation Based on the Shock Pulse Method and Principal Resonance Analysis», *IEEE Transactions on Instrumentation and Measurement*.
- ZENGQIANG MA, S. L. X. L., FEIYU LU (2020), «An Adaptive Generalized Demodulation Method for Multimedia Spectrum Analysis is Applied in Rolling Bearing Fault Diagnosis», *IEEE Access (Volume: 8)*.

Siti Web consultati

- Wikipedia – www.wikipedia.org
- SKF – www.skf.com/it
- MatLab – www.mathworks.com
- Python – www.python.org
- Keras – www.keras.io
- TensorFlow – www.tensorflow.org
- Power-MI – www.power-mi.com
- Mobius Institute – www.mobiusinstitute.com
- GitHub – www.github.com
- Kaggle – www.kaggle.com