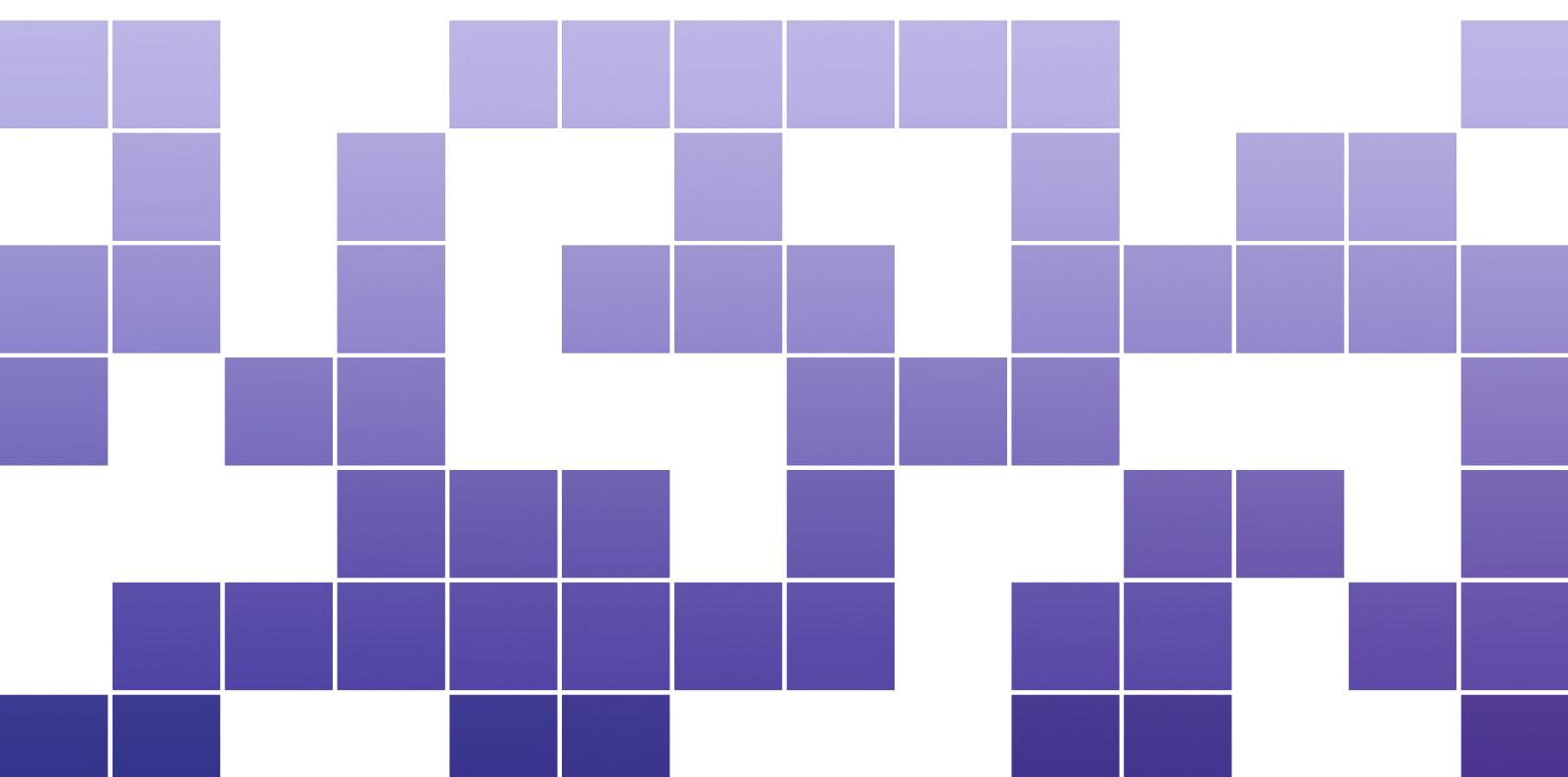


Modulo di diagnosi per il sistema di propulsione PHM Asia Pacific 2023

Progetto svolto da:

Diego Santarelli, Andrea Marini, Simone Recinelli



Università Politecnica delle Marche
Facoltà di Ingegneria
Dipartimento di Ingegneria dell'Informazione
Corso di Laurea Magistrale in Ingegneria Informatica e dell'Automazione



Progetto del corso di:
Manutenzione Preventiva per l'Automazione e la Robotica Intelligente

Tenuto dal Prof. Alessandro Freddi, durante l'anno accademico 2024-2025

9 CFU

Redazione del progetto e del documento a cura di:

- **Dott. Diego Santarelli** (Matr. 1118746) - diegosantarelli37@gmail.com
- **Dott. Andrea Marini** (Matr. 1118778) - andreaa.mariniii@gmail.com
- **Dott. Simone Recinelli** (Matr. 1118757) - simonerecinelli@gmail.com

Anno Accademico 2024-2025
Anno II - Semestre I



Indice

Introduzione	5
1 Task 1	11
1.1 Task 1: Classificazione delle situazioni normali e anomali	11
1.2 Visualizzazione degli Spettri di Potenza	11
1.3 Generazione e selezione delle Feature	13
1.4 Modello di Classificazione Scelto per il Task 1	14
1.4.1 Valutazione dell'Accuratezza sul Test Set Reale	16
2 Task 2	17
2.1 Task 2: Classificazione delle anomalie note e ignote	17
2.1.1 Motivazioni dell'approccio in due step	17
2.2 Metodologia	18
2.2.1 Primo classificatore	18
2.2.2 Secondo classificatore	21
3 Task 3	24
3.1 Task 3: Localizzazione della bolla	24
3.2 Metodologia	24
3.2.1 Preparazione del dataset	24
3.2.2 Generazione e selezione delle feature	24
3.3 Modello di classificazione	25
3.3.1 Valutazione dell'accuratezza del modello	25

4	Task 4	27
4.1	Localizzazione del guasto sulla valvola	27
4.2	Metodologia	27
4.2.1	Preparazione del dataset	27
4.2.2	Generazione e selezione delle feature	27
4.3	Modello di classificazione	28
4.3.1	Valutazione dell'accuratezza del modello	28
5	Task 5	30
5.1	Predizione della percentuale di apertura delle valvole	30
5.2	Metodologia	30
5.2.1	Preparazione del dataset	30
5.2.2	Data Augmentation	30
5.2.3	Generazione e selezione delle feature	31
5.3	Scelta della regressione come tecnica di apprendimento	31
5.4	Valutazione delle prestazioni del modello	32



Introduzione

Negli ultimi anni, l'affidabilità dei sistemi di propulsione spaziale ha assunto un ruolo sempre più cruciale, poiché eventuali guasti durante una missione possono compromettere irrimediabilmente il successo dell'operazione. Tuttavia, il monitoraggio continuo di tali sistemi risulta spesso complesso a causa delle difficoltà di acquisizione e trasmissione dei dati nello spazio. Per affrontare questa sfida, la Japan Aerospace Exploration Agency (JAXA) ha sviluppato un simulatore avanzato in grado di riprodurre con precisione il comportamento di un sistema di propulsione sia in condizioni normali sia in presenza di diversi scenari di guasto. Questo simulatore si è rivelato uno strumento di grande valore per lo sviluppo di sistemi diagnostici e per la manutenzione predittiva, permettendo di migliorare l'affidabilità e la sicurezza delle missioni spaziali.

Obiettivi di progetto

Il progetto trattato di seguito si inserisce nell'ambito della competizione PHM Asia Pacific 2023, finalizzata allo sviluppo di un modulo di diagnosi automatica capace di individuare e classificare anomalie nei dati di telemetria simulati di un sistema di propulsione spaziale. In particolare, il sistema diagnostico richiesto deve essere in grado di:

- Distinguere tra condizioni di funzionamento normale e anomalo;
- Identificare la tipologia di guasto (contaminazione da bolle, malfunzionamento di una valvola elettromagnetica, guasto sconosciuto);
- Localizzare la posizione della bolla nell'impianto di propulsione;
- Individuare quale valvola elettromagnetica è guasta;
- Stimare il rapporto di apertura della valvola guasta.

Descrizione del sistema di propulsione simulato

Il sistema di propulsione considerato è schematizzato in Figura 1. Esso è costituito da un serbatoio ad alta pressione (2.0 MPa) che alimenta una rete di tubazioni, valvole e sensori di pressione. Le elettrovalvole (SV1-SV4) regolano il flusso del propellente, mentre la pressione

è misurata da sette sensori di pressione distinti (P1-P7). L'inserimento di bolle d'aria e il malfunzionamento delle valvole rappresentano le principali fonti di anomalia simulate.

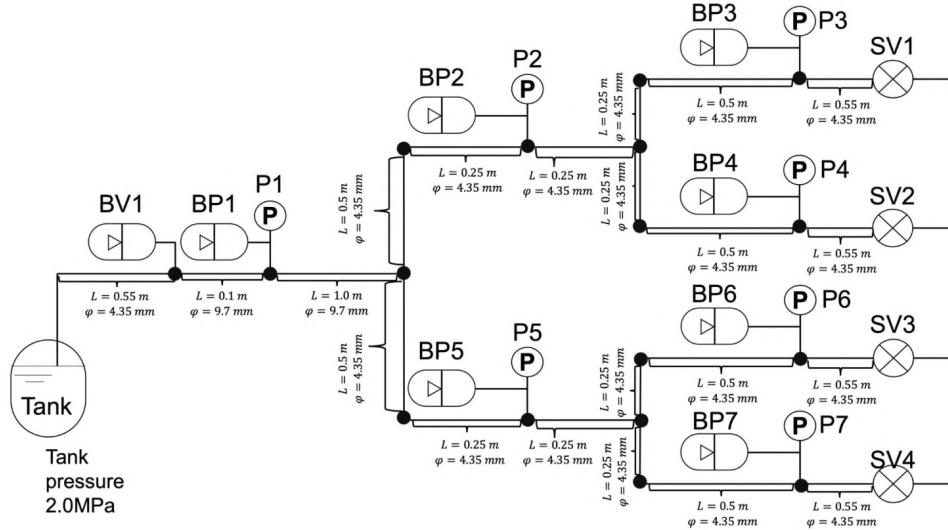


Figura 1: Schema del sistema di propulsione sperimentale.

Profilo tipico della pressione

Il profilo tipico della pressione misurata durante un esperimento è mostrato in Figura 2. Le elettrovalvole vengono azionate ciclicamente secondo una sequenza di apertura e chiusura, causando variazioni caratteristiche della pressione. Un'analisi accurata di tali variazioni consente di identificare eventuali comportamenti anomali e diagnosticarne la causa.

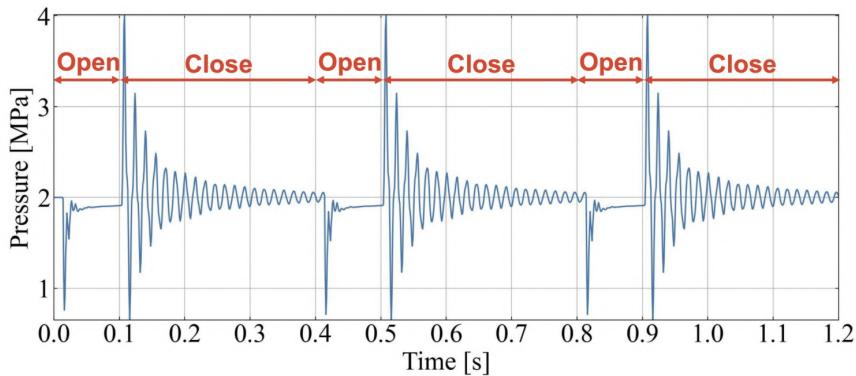


Figura 2: Profilo tipico della pressione nel sistema di propulsione.

Dataset di riferimento

Il dataset fornito per la competizione è stato sviluppato in collaborazione con la Japan Aerospace Exploration Agency (JAXA) ed è composto da 223 esperimenti simulati. Ogni esperimento è rappresentato da una serie temporale che misura la pressione in 7 punti distinti dell'impianto, con una frequenza di campionamento di 1 kHz per una durata totale di 1200 ms. I dati sono suddivisi in un training set etichettato (Case 1 - 177) e in un test set non

etichettato (Case 178 - 223), utilizzato per valutare le prestazioni del modulo diagnostico sviluppato.

Le anomalie simulate nel dataset includono:

- Contaminazione da bolle: la presenza di bolle d'aria nelle tubazioni altera la velocità del suono nel fluido, causando variazioni nelle fluttuazioni di pressione. Le bolle possono trovarsi in otto posizioni possibili: BV1 e BP1 - BP7.
- Guasti alle valvole elettromagnetiche: in condizioni normali, le valvole elettromagnetiche (SV1 - SV4) operano con un rapporto di apertura del 100% (aperte) o 0% (chiuse). In caso di guasto, una valvola potrebbe aprirsi parzialmente, con un rapporto di apertura compreso tra 0% e 100%, riducendo il flusso del propellente.
- Anomalie sconosciute: il dataset di test include anche anomalie non previste, per valutare la capacità del sistema diagnostico di distinguere tra guasti noti e condizioni anomale non anticipate.

È importante notare che, per simulare le differenze individuali tra componenti spaziali reali, sono state considerate quattro varianti del sistema di propulsione (denominate No.1 - No.4, corrispondenti agli spacecraft). I dati di addestramento includono le varianti No.1 - No.3, mentre il set di test contiene dati anche della variante No.4, non presente nel training set.

Strumenti utilizzati per l'analisi dei dati

Per affrontare le sfide diagnostiche proposte dalla competizione, è stato adottato un approccio basato su MATLAB, sfruttando applicazioni specifiche per l'analisi dei segnali e l'apprendimento automatico. Gli strumenti principali utilizzati includono:

- Diagnostic Feature Designer: quest'applicazione consente di progettare e confrontare interattivamente diverse caratteristiche (feature) diagnostiche, facilitando l'identificazione di quelle più efficaci nel discriminare tra condizioni normali e di guasto. Permette l'importazione di dati misurati o simulati, l'elaborazione dei segnali e l'analisi statistica delle feature. Inoltre, include strumenti per l'analisi in frequenza, come la trasformata di Fourier (FFT) e lo spettrogramma, che consentono di estrarre feature spettrali significative per il riconoscimento delle anomalie. La possibilità di visualizzare spettri di ampiezza e densità spettrale di potenza facilita l'individuazione di pattern caratteristici associati a differenti condizioni operative.
- Classification Learner: utilizzato per addestrare modelli di classificazione supervisionata, questo strumento offre una varietà di algoritmi, tra cui alberi decisionali, macchine a vettori di supporto e metodi *ensemble*. Consente di esplorare i dati, selezionare le feature più rilevanti per l'addestramento, specificare schemi di validazione e ottimizzare gli iperparametri per migliorare le prestazioni del modello.
- Regression Learner: strumento che permette di addestrare modelli di regressione per prevedere variabili continue. Offre una gamma di algoritmi, tra cui regressione lineare, alberi di regressione e macchine a vettori di supporto. Consente di esplorare i dati, selezionare le feature, specificare schemi di validazione e valutare le prestazioni dei modelli attraverso metriche opportune per la regressione.

L'integrazione di questi strumenti ha facilitato un flusso di lavoro efficiente, dalla progettazione delle feature diagnostiche all'addestramento e alla valutazione dei modelli, supportando lo sviluppo di un sistema diagnostico robusto e accurato.

Strategia ed organizzazione del progetto

La presente relazione è strutturata in modo da riflettere le diverse fasi del progetto, articolate nei cinque task principali richiesti dalla competizione:

1. Task 1 (rilevazione di anomalie): distinguere tra condizioni di funzionamento normale e anomalo.
2. Task 2 (classificazione del tipo di guasto): identificare se l'anomalia è dovuta a contaminazione da bolle, malfunzionamento di una valvola elettromagnetica o un guasto sconosciuto.
3. Task 3 (localizzazione della bolla): determinare la posizione della bolla nell'impianto di propulsione.
4. Task 4 (identificazione della valvola guasta): individuare quale valvola elettromagnetica presenta un malfunzionamento.
5. Task 5 (stima del rapporto di apertura della valvola guasta): determinare il grado di apertura della valvola malfunzionante.

La strategia seguita si sviluppa in una serie di passaggi strutturati per affrontare la challenge in modo efficace. Il primo step consiste nella generazione di feature diagnostiche utili per il rilevamento e la classificazione dei guasti. Tali feature rappresentano la base per l'addestramento dei modelli dedicati ai vari task. Per l'estrazione delle feature dai segnali analizzati, è stata adottata una strategia di suddivisione temporale, nota come *frame policy*. Quando si lavora con i segnali temporali, suddividerli in finestre più piccole (frame) permette di applicare specifiche operazioni su ciascuna porzione del segnale. In un contesto di classificazione, ogni frame viene analizzato separatamente e successivamente si combina l'insieme delle predizioni per ottenere una valutazione complessiva dell'intero segnale. A tal fine, è stato implementato un algoritmo di voting, che aggrega le predizioni di ciascun frame per determinare la classe finale del segnale. Il sistema di votazione considera le etichette previste per ogni frame e assegna loro un peso uniforme, garantendo che tutte le porzioni del segnale contribuiscano equamente alla decisione finale. La classe assegnata al segnale complessivo corrisponde a quella che riceve la maggioranza dei voti, con una soglia di decisione adattata per regolare il livello di affidabilità della classificazione.

Successivamente, attraverso la k-fold cross validation, il training set è stato suddiviso in un set di dati di training e in un set di dati di validation, in modo tale da consentire la validazione dei modelli. Questo approccio si è reso necessario poiché, nel contesto della challenge, il set di test non forniva informazioni sulle classi di guasto, rendendo indispensabile una validazione interna per stimare l'efficacia dei modelli.

Questa strategia permette di ridurre il rischio che il modello si adatti eccessivamente ai dati di validazione, migliorando l'affidabilità delle sue prestazioni su dati mai visti prima. Infine, grazie alla pubblicazione dei risultati ufficiali da parte degli organizzatori della competizione (file `answers.csv`), è stato possibile confrontare le predizioni dei modelli con i valori reali per calcolare le performance e stimare la reale capacità predittiva degli algoritmi sviluppati.

Preprocessing ed etichettatura

Prima di affrontare i singoli task, è stata svolta un'attività preliminare di preprocessing. Questa fase iniziale, comune a tutti i task, è risultata essenziale per garantire la coerenza dei dati e la corretta impostazione delle attività di classificazione e stima previste dai cinque task.

Importazione e gestione dei file

Sono stati eseguiti i seguenti passaggi:

- Importazione delle label dal file `labels.xlsx`, con rinomina dei campi principali (Case, Spacecraft, Condition);
- Conversione dei valori testuali 'Yes'/'No' in valori numerici binari (0/1) per facilitare il processamento dei dati;

- Caricamento dei file contenenti i segnali di pressione relativi ai 177 esperimenti dalla cartella dataset/train/data.

Assegnazione delle etichette per i task di classificazione

Per ciascun task, sono state definite e assegnate le seguenti etichette ai dati:

- Task 1: Distinzione tra condizioni di funzionamento normale (0) e anormale (1);
- Task 2: Classificazione dei guasti tra Bubble Anomaly (2), Solenoid Valve Fault (3) e Unknown Fault (1);
- Task 3: Identificazione della posizione della bolla tra i punti {BP1-BP7, BV1} nei casi di Bubble Anomaly. A ciascun esperimento viene assegnata un'etichetta numerica corrispondente alla posizione della bolla: 1 per bolla in BP1, 2 per BP2, 3 per BP3, 4 per BP4, 5 per BP5, 6 per BP6, 7 per BP7 e 8 per BV1. Se non è presente alcuna bolla, viene assegnata l'etichetta 0.
- Task 4: Identificazione della valvola guasta tra SV1-SV4 nei casi di Solenoid Valve Fault;
- Task 5: Predizione della percentuale di apertura della valvola guasta (valori continui tra 0 e 100).

Struttura dei dati

La tabella principale, denominata `labeledData`, associa a ciascun esperimento, ovvero un Case, le relative etichette per i cinque task descritti. Un esempio della struttura della tabella è riportato nella Figura 3.

	1 Case	2 Task1	3 Task2	4 Task3	5 Task4	6 Task5
1	1201x8 t...	0	0	0	0	100
2	1201x8 t...	0	0	0	0	100
3	1201x8 t...	0	0	0	0	100
4	1201x8 t...	0	0	0	0	100
5	1201x8 t...	0	0	0	0	100
6	1201x8 t...	0	0	0	0	100
7	1201x8 t...	0	0	0	0	100
8	1201x8 t...	0	0	0	0	100
9	1201x8 t...	0	0	0	0	100
10	1201x8 t...	0	0	0	0	100
11	1201x8 t...	0	0	0	0	100
12	1201x8 t...	0	0	0	0	100
13	1201x8 t...	0	0	0	0	100
14	1201x8 t...	0	0	0	0	100
15	1201x8 t...	0	0	0	0	100
16	1201x8 t...	0	0	0	0	100
17	1201x8 t...	0	0	0	0	100
18	1201x8 t...	0	0	0	0	100
19	1201x8 t...	0	0	0	0	100
20	1201x8 t...	0	0	0	0	100
21	1201x8 t...	0	0	0	0	100
22	1201x8 t...	0	0	0	0	100
23	1201x8 t...	0	0	0	0	100
24	1201x8 t...	0	0	0	0	100
25	1201x8 t...	0	0	0	0	100
26	1201x8 t...	0	0	0	0	100

Figura 3: Tabella `labeledData` contenente i Case ID e le etichette dei vari task.

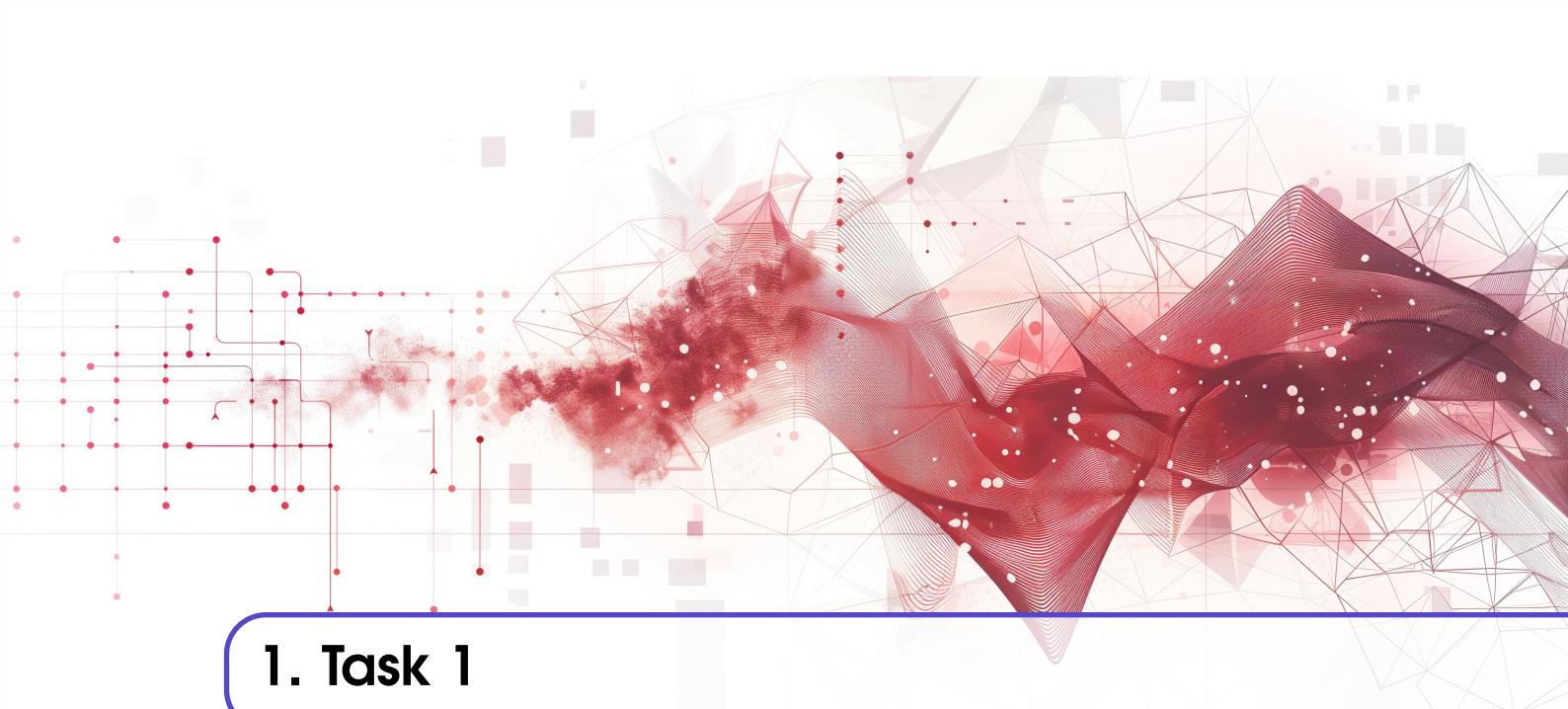
Cliccando su ciascun Case ID, è possibile accedere a una tabella contenente i segnali di pressione rilevati nei sette sensori di pressione (P1-P7) durante l'esperimento. Ogni tabella contiene misurazioni campionate nel tempo, come mostrato in Figura 4.

	1 TIME	2 P1	3 P2	4 P3	5 P4	6 P5	7 P6	8 P7
1	0	2	2	2	2	2	2	2
2	1.0000e-...	2	2	2	2	2	2	2
3	0.0020	2	2	2	2	2	2	2
4	0.0030	2	2	2	2	2	2	2
5	0.0040	2	2	2	2	2	2	2
6	0.0050	2	2	2	2	2	2	2
7	0.0060	2	2	2	2	2	2	2
8	0.0070	2	2	2	2	2	2	2
9	0.0080	2	2	2	2	2	2	2
10	0.0090	2	2	2	2	2	2	2
11	0.0100	2	2	2	2	2	2	2
12	0.0110	2	2	2	2	2	2	2
13	0.0120	2	2.0000	0.2155	0.1310	2	0.1310	0.1372
14	0.0130	2.0000	0.5218	0.8185	0.6050	1.5775	0.6081	0.4910
15	0.0140	1.7617	0.8746	0.0018	0.1221	5.6267e-...	0.9022	0.3542
16	0.0150	0.4622	1.2087	0.3023	0.3654	1.2089	0.4875	0.1733
17	0.0160	0.9581	0.0096	1.4608	0.7992	0.8488	0.7555	0.5586
18	0.0170	1.3559	0.7887	0.1726	0.2872	0.3328	0.5983	0.5585
19	0.0180	1.4153	0.7909	0.5665	0.3783	1.2534	0.1200	0.2593
20	0.0190	1.8631	1.0574	1.5779	0.7667	1.2579	0.3203	0.7465
21	0.0200	1.7928	1.6564	0.9394	1.2827	1.1617	0.8488	1.0131
22	0.0210	1.5809	1.3881	1.2196	1.7012	1.5167	0.9924	1.1293
23	0.0220	1.5409	1.6581	1.7371	1.7679	1.6109	1.2285	1.5095
24	0.0230	1.6185	1.7502	1.2834	1.5591	1.5538	1.4229	1.8938
25	0.0240	1.7351	1.6605	2.0077	1.6359	1.5998	1.4143	1.9274
26	0.0250	1.8843	1.7586	1.5674	1.7681	1.9062	1.4437	1.8687

Figura 4: Tabella dei dati di pressione per un singolo esperimento. Misurazioni nel tempo per i sensori P1-P7.

Conclusioni

Questa fase di preprocessing ed etichettatura ha costituito la base per tutte le successive attività di analisi e modellazione, risultando cruciale per garantire la qualità dei dati e la corretta esecuzione degli algoritmi di classificazione e stima richiesti nei vari task.



1. Task 1

1.1 Task 1: Classificazione delle situazioni normali e anormali

Il Task 1 ha come obiettivo l'individuazione delle condizioni di funzionamento normale e anormale del sistema di propulsione. È stato quindi affrontato come un problema di classificazione binaria, in cui ciascun esperimento viene etichettato come:

- Normal: funzionamento regolare (etichetta 0);
- Abnormal: presenza di un'anomalia o guasto (etichetta 1).

1.2 Visualizzazione degli Spettri di Potenza

Per lavorare efficacemente con segnali temporali, il segnale di ciascun esperimento è stato suddiviso in finestre temporali (frame) utilizzando la seguente configurazione:

- Frame Size (FS): 0.400s;
- Frame Rate (FR): 0.400s.

Di conseguenza, le finestre risultano contigue e non sovrapposte, garantendo così una suddivisione continua del segnale nel tempo. Tali parametri sono stati selezionati in seguito a sperimentazioni preliminari e hanno dimostrato di garantire il miglior compromesso tra granularità dell'analisi e stabilità dei risultati.

Questi valori sono stati selezionati in seguito a sperimentazioni preliminari e hanno dimostrato di garantire il miglior compromesso tra granularità dell'analisi e stabilità dei risultati.

Per una migliore comprensione del comportamento dei segnali, sono stati analizzati gli spettri di potenza delle pressioni rilevate nei sensori P1 - P7. Le seguenti Figure 1.1–1.7 mostrano gli spettri relativi ai diversi canali di pressione:

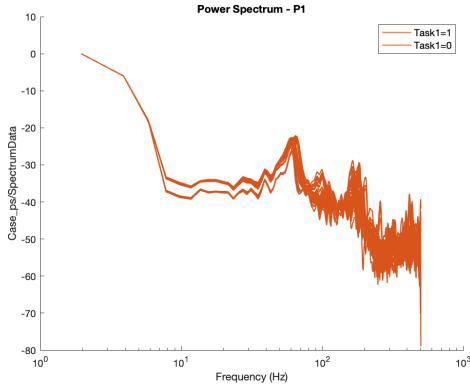


Figura 1.1: Spettro di potenza relativo al sensore P1.

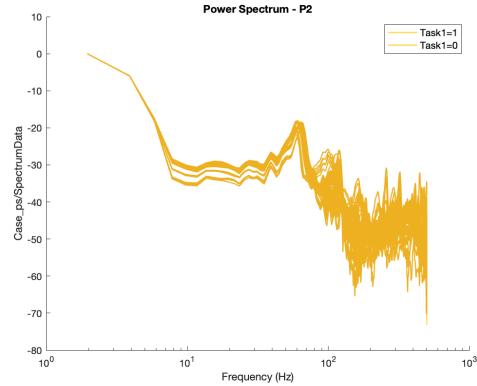


Figura 1.2: Spettro di potenza relativo al sensore P2.

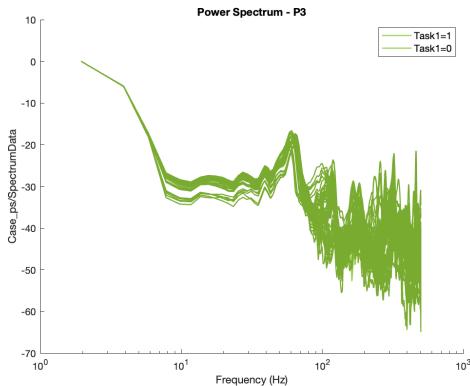


Figura 1.3: Spettro di potenza relativo al sensore P3.

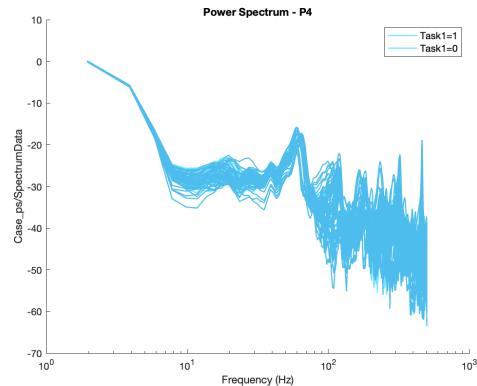


Figura 1.4: Spettro di potenza relativo al sensore P4.

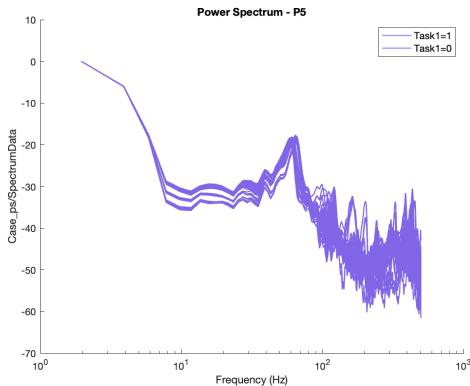


Figura 1.5: Spettro di potenza relativo al sensore P5.

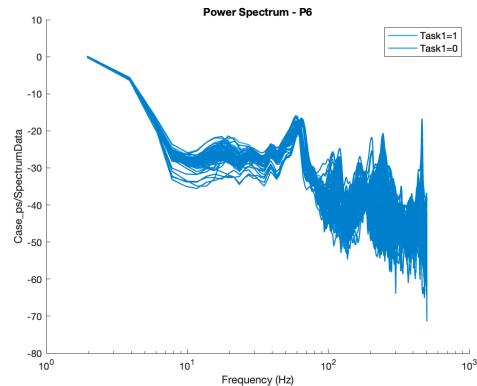


Figura 1.6: Spettro di potenza relativo al sensore P6.

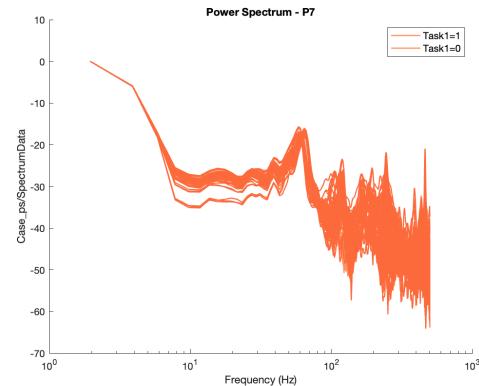


Figura 1.7: Spettro di potenza relativo a P7

1.3 Generazione e selezione delle Feature

La fase iniziale ha riguardato la generazione manuale di feature diagnostiche utili per caratterizzare il comportamento dei segnali di pressione misurati nei sette punti del sistema (P1-P7). Le feature sono state generate sia nel dominio del tempo che in quello della frequenza. In particolare, le feature utilizzate sono:

- **Dominio del tempo:**

- Varianza: misura la dispersione dei valori del segnale attorno alla media, evidenziando variazioni e fluttuazioni.
- Integrale lineare: rappresenta l'area sottesa dal segnale, utile per analizzare il comportamento accumulato nel tempo.
- Massimo: valore massimo raggiunto dal segnale, indicativo di picchi anomali o condizioni estreme.
- Minimo: valore minimo del segnale, utile per individuare eventuali cadute improvvise di pressione.
- Media: valore medio del segnale, fornisce un'indicazione della tendenza centrale del comportamento del sistema.
- Mediana: valore centrale del segnale, meno sensibile ai picchi rispetto alla media e utile per descrivere la distribuzione dei dati.
- 25° e 75° percentile: valori che delimitano il primo e il terzo quartile della distribuzione dei dati, utili per valutare la dispersione e la presenza di outlier.

- **Dominio della frequenza:**

- Peak Value: valore massimo dello spettro di frequenza, utile per identificare la presenza di risonanze o componenti dominanti.
- Peak Frequency: frequenza corrispondente al valore massimo dello spettro, indicativa della componente principale del segnale.
- RMS: valore quadratico medio nel dominio della frequenza, misura la potenza del segnale.
- Power Spectrum Sum: somma delle ampiezze dello spettro di potenza, rappresenta l'energia totale del segnale nel dominio della frequenza.
- Deviazione standard: misura la variabilità delle ampiezze dello spettro, utile per valutare la presenza di rumore o segnali distribuiti su più frequenze.

Per valutare l'importanza delle feature e selezionare quelle più rilevanti ai fini della classificazione, è stato utilizzato il test ANOVA (Analysis of Variance). Questo metodo statistico ha come obiettivo verificare se i dati provenienti da diverse classi di guasto condividono la stessa media. L'ANOVA consente di identificare le feature che contribuiscono maggiormente alla separazione tra le categorie, riducendo così la dimensionalità del problema e migliorando l'efficacia del modello di classificazione. Dopo l'analisi ANOVA, le feature più significative sono state selezionate per l'addestramento del modello, migliorando la robustezza e la generalizzazione del classificatore.

In Figura 1.8 è riportata la classifica di importanza delle feature ottenuta tramite questo approccio. In particolare, utilizzando l'Anova, sono state considerate 22 feature.

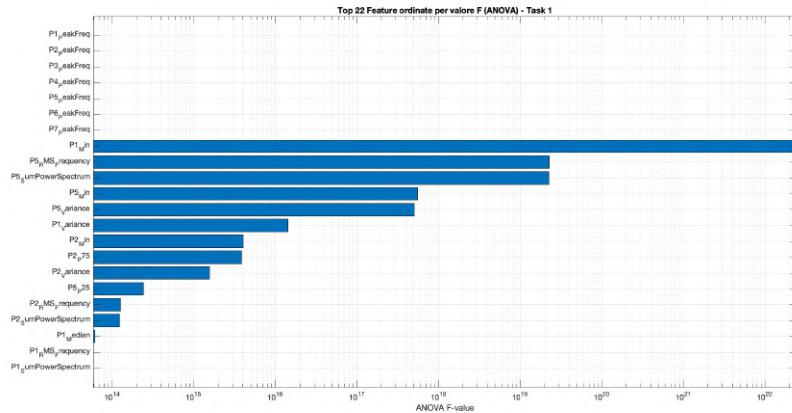


Figura 1.8: Feature importance ottenuta tramite Anova.

1.4 Modello di Classificazione Scelto per il Task 1

Dopo aver effettuato diverse sperimentazioni con vari algoritmi di classificazione, il modello che ha mostrato le migliori prestazioni è risultato essere il CoarseTree. Questo modello si è rivelato particolarmente robusto ed efficace per il compito di individuare le condizioni di funzionamento normale e anormale del sistema di propulsione.

Il modello è stato addestrato utilizzando le 22 feature ordinate tramite Anova nella fase precedente e validato tramite K-fold cross-validation con $K = 10$, al fine di garantire la robustezza dei risultati.

In figura 1.9, è riportata la matrice di confusione ottenuta sui dati di validazione.

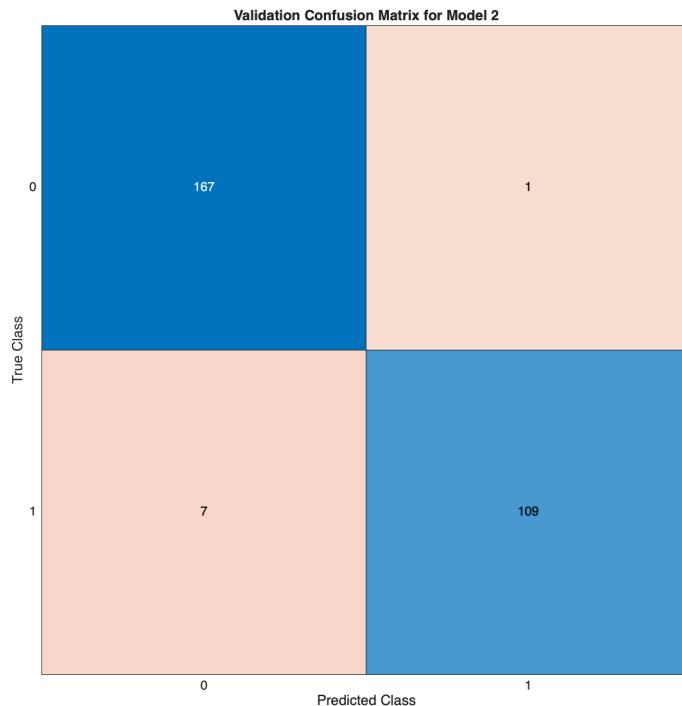


Figura 1.9: Matrice di confusione sul validation set del modello per il Task 1.

Voting per Decisione Finale

Poiché il modello produce una predizione per ciascun frame, è stato implementato un algoritmo di voting per stabilire la classe finale dell'esperimento. La decisione complessiva è stata presa secondo la regola di maggioranza dei frame: l'etichetta assegnata al caso corrisponde a quella predetta più frequentemente tra i suoi frame. In altre parole, la classe finale assegnata a ciascun esperimento corrisponde alla classe più frequente tra tutte quelle predette per i frame appartenenti a quello specifico caso.

In particolare, ciascun esperimento (Case) viene suddiviso in più finestre temporali (frame), ognuna delle quali è analizzata singolarmente dal modello di classificazione. Poiché ogni frame può essere classificato in modo indipendente come normale o anomalo, per ottenere un'unica predizione finale per l'intero esperimento è necessario aggregare i risultati delle singole finestre. Questo viene fatto applicando la regola della maggioranza semplice: la classe assegnata all'esperimento è quella che compare con maggiore frequenza tra le predizioni dei suoi frame.

L'adozione di questa strategia consente di mitigare l'effetto di eventuali errori di classificazione sui singoli frame e ottenere una valutazione più robusta sullo stato complessivo del sistema durante l'esperimento.

Matematicamente, il voting è stato implementato calcolando la moda (`mode`) delle predizioni dei frame corrispondenti a ciascun esperimento, garantendo così che l'etichetta più ricorrente venga assegnata come output finale per il task.

Questa tecnica si è dimostrata particolarmente efficace, in quanto consente di bilanciare l'incertezza delle singole predizioni riducendo il rischio di falsi positivi e negativi, contribuendo così all'affidabilità complessiva del modello Coarse Tree applicato al Task 1.

Di seguito, la Figura 1.10 mostra la matrice di confusione ottenuta per il Task 1, evidenziando la distribuzione degli errori di classificazione tra le diverse classi.

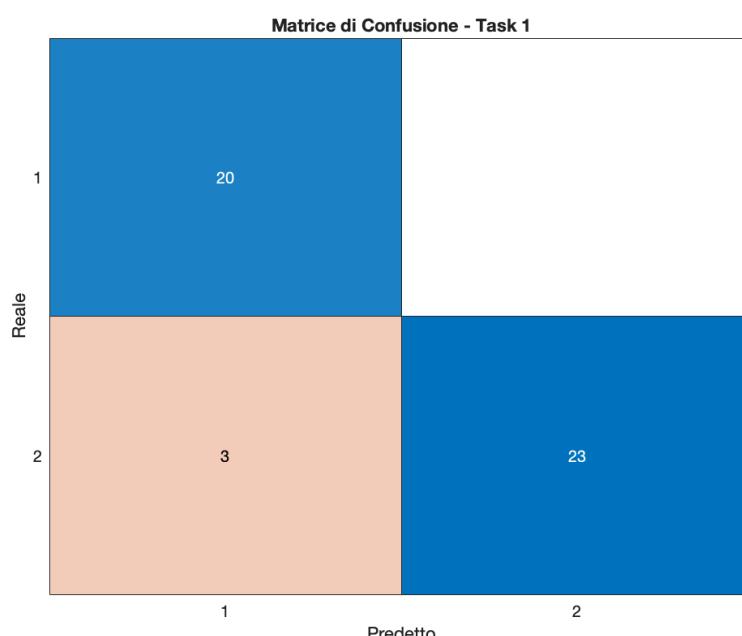


Figura 1.10: Matrice di confusione del modello Coarse Tree per il Task 1.

1.4.1 Valutazione dell'Accuratezza sul Test Set Reale

Dopo aver addestrato il modello Coarse Tree e ottenuto le predizioni sui dati di test non etichettati, è stata condotta una verifica finale delle prestazioni del modello confrontando le predizioni con le etichette reali presenti nel file `answer.csv`, fornito dagli organizzatori della competizione. Tale file contiene le risposte corrette relative ai Task proposti, tra cui il Task 1, permettendo così di calcolare l'accuratezza effettiva del modello.

Per effettuare questa valutazione, sono stati eseguiti i seguenti passaggi:

- Importazione dei dati di test etichettati (test set reale);
- Estrazione delle predizione dei valori di Task 1 per ciascun caso di test;
- Confronto tra le predizioni e le etichette effettive del Task 1;
- Calcolo dell'accuratezza come rapporto tra il numero di predizioni corrette e il numero totale di casi nel test set.

Al termine dell'esecuzione del codice, è stata calcolata l'accuratezza finale del modello sul test set reale, risultata pari a **93.48%**.

Questo valore conferma l'efficacia del modello Coarse Tree nel discriminare tra condizioni di funzionamento normale e anormale, dimostrando una buona capacità predittiva anche sui dati di test ufficiali forniti dalla competizione (Figura 1.11).

Case	PredictedLabel	ActualLabel
178	1	1
179	1	1
180	0	0
181	1	1
182	0	0
183	0	0
184	1	1
185	0	0
186	1	1
187	0	0
188	1	1
189	0	0
190	1	1
191	0	0
192	1	1
193	1	1
194	0	0
195	0	0
196	1	1
197	1	1
198	0	0
199	0	1
200	1	1
201	0	0
202	1	1
203	0	0
204	1	1
205	1	1
206	0	0
207	1	1
208	0	0
209	1	1
210	0	0
211	0	1
212	0	1
213	0	0
214	1	1
215	0	0
216	1	1
217	0	0
218	1	1
219	1	1
220	0	0
221	1	1
222	1	1
223	0	0

Figura 1.11: Confronto etichette predette con etichette reali



2. Task 2

2.1 Task 2: Classificazione delle anomalie note e ignote

L'obiettivo del Task 2 è distinguere tra i diversi tipi di guasto, identificando la natura delle anomalie classificate come condizioni anormali nel Task 1 e assegnando loro una specifica etichetta.

Per affrontare il problema in modo efficace, il processo è stato suddiviso in due step sequenziali:

1. Primo classificatore: prende in input le situazioni anormali classificate dal Task 1 e distingue tra anomalie note e anomalie ignote (a cui viene assegnata un'etichetta pari ad 1);
2. Secondo classificatore: prende in input solo le anomalie note identificate dal primo classificatore e ne determina la natura, classificandole come Bubble Anomaly (etichetta 2) o Valve Fault (etichetta 3).

2.1.1 Motivazioni dell'approccio in due step

L'adozione di un approccio gerarchico e sequenziale rappresenta la strategia più efficace per risolvere il Task 2, poiché garantisce un processo di classificazione più robusto e interpretabile. Di seguito sono riportati i principali vantaggi di questa scelta:

- Maggiore capacità di identificare le anomalie ignote: poiché nel training set non sono presenti etichette per le anomalie ignote, il primo classificatore ha il compito di individuare tutto ciò che non corrisponde ai guasti noti, basandosi sulla distribuzione delle feature. Questo step consente di isolare le anomalie ignote senza che il secondo classificatore debba occuparsi della loro gestione, migliorando così la capacità di generalizzazione del sistema;
- Maggiore accuratezza nella classificazione dei guasti noti: una volta separati i guasti noti dalle anomalie ignote, il secondo classificatore lavora su un dataset più pulito e bilanciato, contenente solo i dati già riconosciuti come anomalie note. Questo consente al modello di concentrarsi esclusivamente sulla distinzione tra Bubble Anomaly e Valve Fault, senza il rischio di interferenze dovute alla presenza di anomalie ignote.

2.2 Metodologia

Il processo di classificazione è stato strutturato in più fasi, dalla selezione del training set alla scelta delle feature più rilevanti per il modello.

2.2.1 Primo classificatore

Preparazione del dataset

Per quanto concerne il primo classificatore, il training set è stato costruito utilizzando esclusivamente i Case contenenti guasti noti, ovvero quelli in cui almeno una delle condizioni Bubble Anomaly o Valve Fault è stata identificata (è stato effettuato il filtraggio della Tabella *labeledData* presente in Figura 3). Questi Case sono stati etichettati con il valore 4. Tuttavia, nel training set non sono presenti dati etichettati come guasti ignoti, rendendo necessario un approccio che permetta al modello di generalizzare sulla loro possibile presenza.

Il test set del primo classificatore è stato costruito filtrando i risultati del Task 1, selezionando solo i Case che erano stati classificati come anormali (etichettati come 1 nella tabella risultante del Task 1). Questo implica che il primo classificatore del Task 2 dovrà determinare se questi Case anomali appartengono ai guasti noti (Bubble Anomaly o Valve Fault) o se devono essere considerati come guasti ignoti.

Generazione e selezione delle feature

Per estrarre informazioni significative dai dati, il training set è stato elaborato, utilizzando una frame policy con i seguenti parametri:

- Frame Size (FS) = 0.400s;
- Frame Rate (FR) = 0.400s.

Questo ha permesso di ottenere una suddivisione dei dati temporali adeguata per l'analisi e la generazione delle feature.

Per garantire un'elevata capacità discriminativa del modello, è stata effettuata una selezione delle feature più rilevanti. Sono state scelte 12 feature (Figura 2.1), ordinate in base al criterio Variance. Questa scelta è stata motivata dal fatto che nel training set non erano presenti esempi di guasti ignoti, quindi non era possibile un approccio supervisionato diretto.

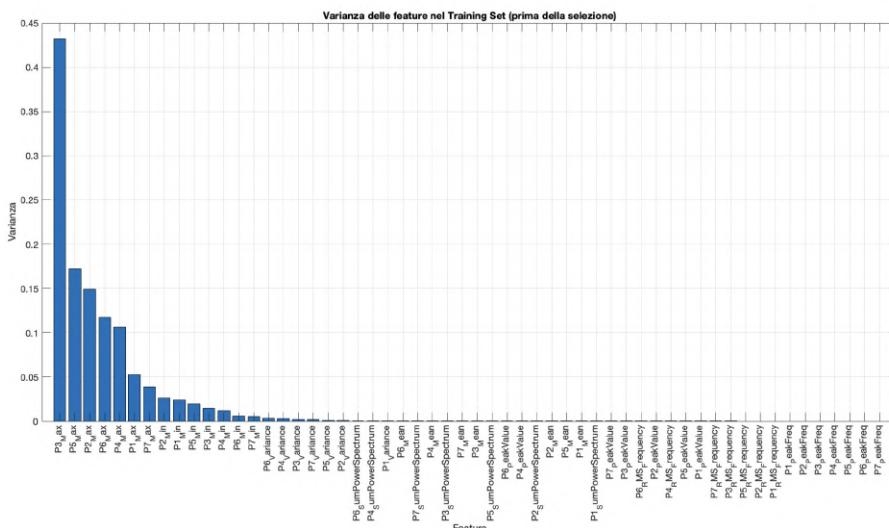


Figura 2.1: Selezione delle feature tramite Variance.

Modello di classificazione

Per separare i guasti noti da quelli ignoti è stato utilizzato un modello one class classifier di tipo Isolation Forest. Questo algoritmo è particolarmente adatto all'individuazione di anomalie, in quanto costruisce una struttura ad albero in grado di isolare rapidamente gli outlier (unknown anomaly) con un numero ridotto di suddivisioni.

Per ottimizzare le prestazioni del modello, è stata adottata una strategia di Grid-Search per la selezione degli iperparametri. Questo metodo ha permesso di esplorare diverse combinazioni di parametri per identificare quelli che garantissero la massima efficacia nella separazione tra guasti noti e ignoti. Nello specifico, i valori di NumLearners (numero di alberi dell'Isolation Forest) sono stati scelti in modo da coprire un ampio range: 100, 300, 500, 1000. Questo intervallo è stato selezionato considerando la letteratura sull'Isolation Forest, dove il numero di alberi solitamente varia tra 100 e 1000, bilanciando stabilità e costo computazionale. Analogamente, i valori di ContaminationFraction (percentuale di dati che il modello assume come anomali) sono stati testati tra 1% e 5%, poiché nei problemi di anomaly detection il tasso di anomalie reali rientra tipicamente in questo intervallo.

Il miglior valore di NumLearners e ContaminationFraction è stato poi scelto in base al numero di falsi positivi generati nel training set, ottimizzando così la capacità del modello di distinguere tra guasti noti e ignoti senza introdurre un'eccessiva quantità di errori.

È stato implementato un meccanismo di voting che aggrega i risultati di più istanze del classificatore, stabilendo una soglia minima pari a 1 per confermare la classificazione di un'anomalia. Questo approccio riduce la probabilità di falsi positivi e migliora l'affidabilità del sistema.

Riepilogo primo classificatore

Dopo la preparazione dei dati, che include la selezione dei casi rilevanti e la generazione delle feature, viene addestrato un modello Isolation Forest ottimizzato tramite ricerca su griglia e validazione k-fold cross-validation. Il modello viene poi applicato ai dati di test, assegnando una previsione a ciascuna finestra temporale e determinando l'etichetta finale di ogni caso attraverso un meccanismo di voting, la cui soglia minima è impostata ad 1. I risultati vengono infine raccolti in una tabella che assegna a ciascun caso un'etichetta di anomalia.

Come già accennato, per ottimizzare il modello, è stata implementata una ricerca grid search che esplora diverse combinazioni di iperparametri, tra cui il numero di alberi nel modello Isolation Forest e la frazione di contaminazione. La selezione del miglior modello avviene minimizzando il numero di falsi positivi sulla base della validazione incrociata.

I Case corrispondenti a guasti noti sono etichettati con valore pari a 4, mentre quelli ignoti con valore pari ad 1.

Valutazione dell'accuratezza del primo classificatore

L'accuratezza del primo classificatore è stata valutata confrontando le sue predizioni con le etichette reali presenti nel dataset di test fornito dalla competizione.

Per determinare la qualità delle previsioni, è stato seguito il seguente processo:

- Caricamento delle etichette reali, filtrando solo i Case che presentano anomalie note, ovvero quelli etichettati come 2 (Bubble Anomaly) e 3 (Valve Fault). Per uniformare il confronto, questi valori sono stati ricodificati con l'etichetta 4, corrispondente ai guasti noti;
- Le previsioni del primo classificatore sono state abbinate alle etichette reali utilizzando la colonna Case, garantendo una corrispondenza diretta tra i dati reali e i risultati del modello;
- È stata quindi calcolata l'accuratezza del modello come la percentuale di previsioni corrette rispetto al totale delle osservazioni analizzate.

L'accuratezza ottenuta dal primo classificatore è pari a **100%**, indicando che il modello ha perfettamente classificato i Case identificati come anomalie dal Task 1 rispetto alle etichette reali (Figura 2.2).

Case	TrueLabel	PredictedLabel
"178"	4	4
"179"	4	4
"181"	4	4
"184"	1	1
"186"	4	4
"188"	4	4
"190"	4	4
"192"	1	1
"193"	4	4
"196"	4	4
"197"	4	4
"200"	1	1
"202"	4	4
"204"	4	4
"205"	4	4
"207"	1	1
"209"	4	4
"214"	4	4
"216"	4	4
"218"	1	1
"219"	4	4
"221"	4	4
"222"	1	1

Figura 2.2: Accuratezza del primo classificatore del Task 2

Di seguito, la Figura 2.3 mostra la matrice di confusione ottenuta per il primo classificatore del Task 2, evidenziando la distribuzione degli errori di classificazione tra le diverse classi.

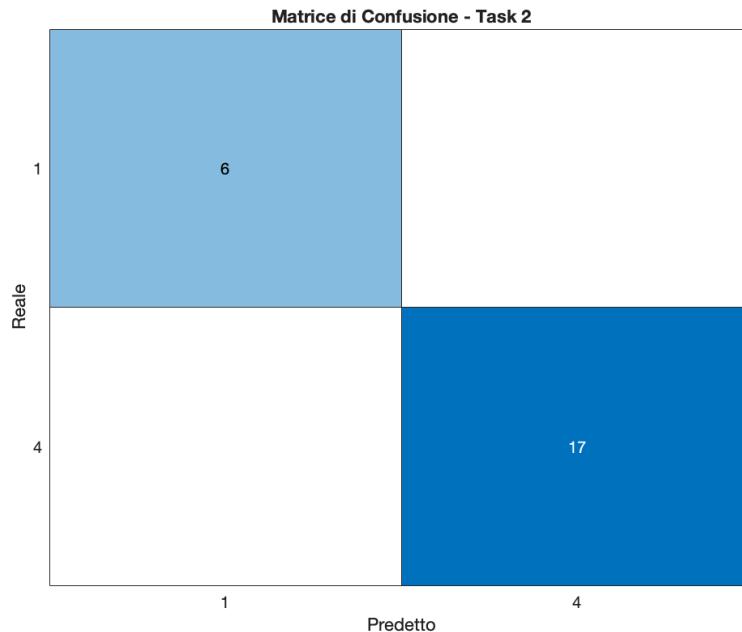


Figura 2.3: Matrice di confusione del primo classificatore del Task 2.

2.2.2 Secondo classificatore

Il secondo classificatore ha il compito di identificare la natura delle anomalie note individuate dal primo classificatore del Task 2, distinguendo tra Bubble Anomaly (etichetta 2) e Valve Fault (etichetta 3).

Preparazione del dataset

Per costruire il dataset del secondo classificatore, sono state applicate diverse operazioni: il training set è stato costruito filtrando esclusivamente i Case contenenti guasti noti, ovvero quelli con etichetta 2 (Bubble Anomaly) e 3 (Valve Fault) dalla Tabella *labeledData* presente in Figura 3. Ovviamente, sono stati mantenuti gli stessi Case del training set utilizzato per il primo classificatore, in modo da garantire coerenza nella rappresentazione dei guasti noti.

Il test set del secondo classificatore è stato costruito filtrando i risultati del primo classificatore, selezionando solo i Case che erano stati classificati come guasti noti (etichetta 4). Questi dati, inizialmente etichettati come guasti noti senza distinzione tra Bubble Anomaly e Valve Fault, vengono ora rianalizzati dal secondo classificatore per assegnare loro l'etichetta corretta, distinguendoli tra anomalie di tipo 2 e 3.

Generazione e selezione delle feature

Per estrarre informazioni significative dai dati, il training set è stato elaborato, utilizzando una frame policy con i seguenti parametri:

- Frame Rate (FR) = 0.400s;
- Frame Size (FS) = 0.400s.

Per migliorare l'efficacia del modello, abbiamo deciso di selezionare tutte le feature disponibili, in quanto il modello risultava ugualmente stabile e non si osservavano peggioramenti significativi nelle prestazioni dovuti a overfitting.

Modello di classificazione

Per la classificazione, sono stati addestrati diversi modelli validati tramite K-fold cross-validation con $K = 10$. Dopo aver testato diverse configurazioni di modelli sul test set non etichettato, il **Quadratic SVM** è risultato il più efficace.

I risultati finali sono stati aggregati attraverso un voto di maggioranza e salvati nel file results.csv, aggiornando la colonna Task2 con le nuove etichette assegnate ai Case.

Valutazione dell'accuratezza del secondo classificatore

L'accuratezza del secondo classificatore è stata valutata confrontando le sue predizioni con le etichette reali presenti nel test set fornito dalla competizione (Figura 2.4).

Il processo di valutazione ha seguito questi passaggi:

- Caricamento delle etichette reali dal file answer.csv, con rinominazione delle colonne e filtraggio dei Case con etichette 2 e 3;
- Caricamento delle predizioni del file results.csv, mantenendo solo i Case con etichette 2 e 3;
- Confronto tra le etichette reali e le predizioni del secondo classificatore per calcolare la percentuale di predizioni corrette.

Il secondo classificatore ha raggiunto un'accuratezza del **100%**, indicando che tutte le anomalie note sono state correttamente classificate.

Case	Task2	CaseLabel
178	2	2
179	3	3
180	0	0
181	3	3
182	0	0
183	0	0
184	1	1
185	0	0
186	2	2
187	0	0
188	3	3
189	0	0
190	3	3
191	0	0
192	1	1
193	2	2
194	0	0
195	0	0
196	2	2
197	2	2
198	0	0
199	0	3
200	1	1
201	0	0
202	3	3
203	0	0
204	2	2
205	3	3
206	0	0
207	1	1
208	0	0
209	2	2
210	0	0
211	0	3
212	0	3
213	0	0
214	3	3
215	0	0
216	2	2
217	0	0
218	1	1
219	2	2
220	0	0
221	2	2
222	1	1
223	0	0

Figura 2.4: Accuratezza del secondo classificatore del Task 2

Di seguito, la Figura 2.5 mostra la matrice di confusione ottenuta per il secondo classificatore del Task 2, evidenziando la distribuzione degli errori di classificazione tra le diverse

classi.

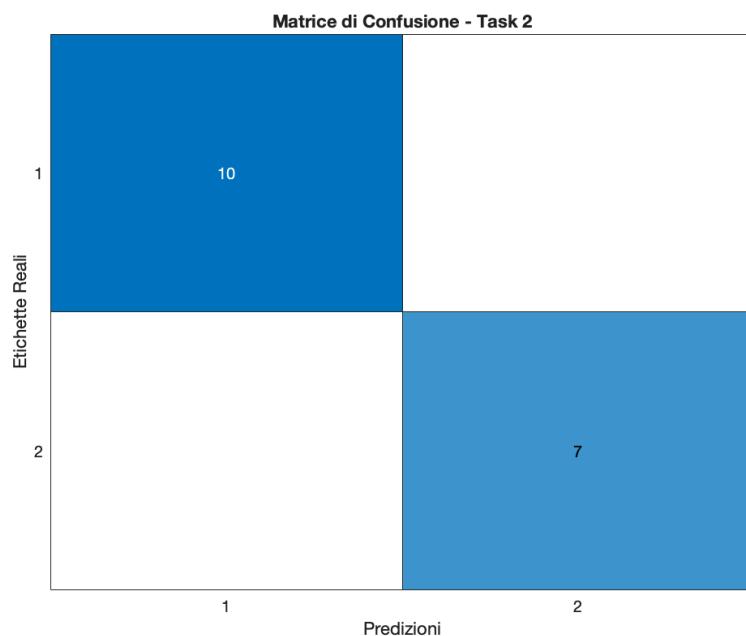


Figura 2.5: Matrice di confusione del secondo classificatore del Task 2.

3. Task 3

3.1 Task 3: Localizzazione della bolla

L'obiettivo del Task 3 è determinare la posizione della bolla nei casi di guasto identificati come bubble anomaly nel Task 2. La posizione della bolla deve essere assegnata a una delle otto etichette disponibili, ovvero BV1 e da BP1 a BP7.

3.2 Metodologia

La metodologia adottata viene spiegata di seguito.

3.2.1 Preparazione del dataset

Per costruire il dataset di training, sono stati selezionati i casi dalla tabella labeledData aventi un'etichetta diversa da zero nella colonna Task3, includendo informazioni relative al nome del caso e all'etichetta associata.

Il test set è stato ottenuto filtrando i risultati del Task 2, includendo solo i Case identificati come bubble anomaly (Task2 = 2). Il modello predittivo dovrà quindi assegnare ad ogni Case una delle etichette da 1 a 8 in base alla posizione della bolla.

3.2.2 Generazione e selezione delle feature

Per estrarre informazioni significative dai dati, il dataset è stato elaborato con una frame policy che utilizza finestre temporali con i seguenti parametri:

- Frame Size (FS) = 0.400s;
- Frame Rate (FR) = 0.400s.

Le finestre temporali sono state analizzate estrapolando un insieme di feature statistiche e frequenziali per ogni segnale misurato. Le feature estratte comprendono:

- Media, mediana, percentili 25 e 75, varianza, integrale numerico, minimo e massimo;
- Spettro di potenza, frequenza di picco, somma dello spettro di potenza, deviazione standard e RMS nel dominio della frequenza.

Successivamente, per la selezione delle feature più rilevanti, è stato utilizzato un test ANOVA, che misura l'importanza statistica di ciascuna feature rispetto alla classificazione della posizione della bolla. Al termine del processo, sono state selezionate le 9 feature più significative, mostrate in Figura 3.1.

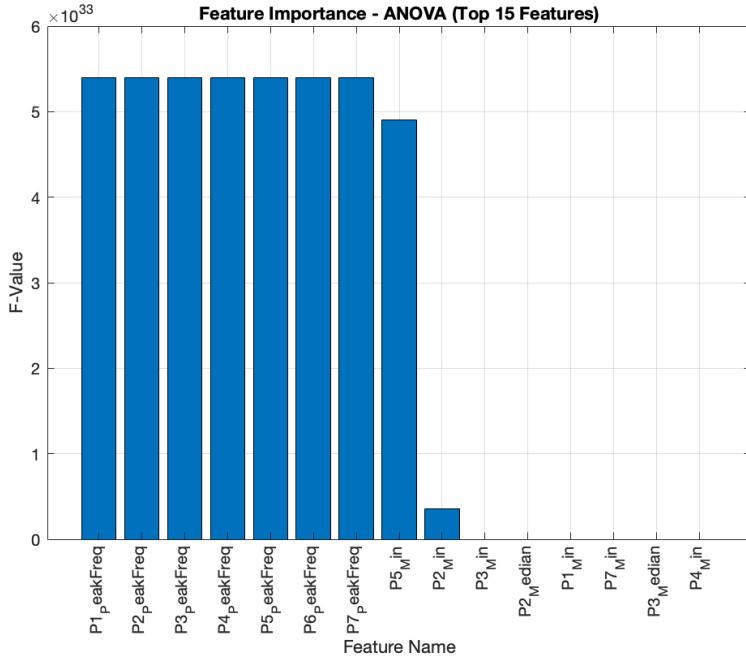


Figura 3.1: Selezione delle feature tramite test ANOVA.

3.3 Modello di classificazione

Dopo aver testato diverse configurazioni di modelli, per la classificazione della posizione della bolla è stato utilizzato un modello di Support Vector Machine con kernel lineare (**Linear SVM**). Il modello è stato validato tramite K-fold cross-validation con $K = 10$, addestrato sui Case con posizione nota e successivamente testato sui Case etichettati come bubble anomaly.

La predizione della posizione della bolla nel test set è stata ottenuta mediante il modello SVM, assegnando un'etichetta a ciascuna finestra temporale. Successivamente, per ciascun Case, è stato applicato il majority voting, selezionando come predizione finale l'etichetta più frequente nelle finestre temporali associate al caso.

3.3.1 Valutazione dell'accuratezza del modello

L'accuratezza del modello è stata valutata confrontando le predizioni con le etichette reali, fornite dalla competizione, attraverso i seguenti passaggi:

- Caricamento delle etichette corrette dal file `answer.csv`;
- Filtraggio dei Case con etichette diverse da zero;
- Caricamento delle predizioni dal file `results.csv`;
- Aggregazione delle predizioni per Case con majority voting;
- Confronto tra etichette reali e predizioni per calcolare l'accuratezza.

Il modello ha raggiunto un'accuratezza dell'**100%**, dimostrando un'elevata capacità di localizzare la posizione della bolla.

Di seguito, la Figura 3.2 mostra la matrice di confusione ottenuta per il Task 3, evidenziando la distribuzione degli errori di classificazione tra le diverse classi.

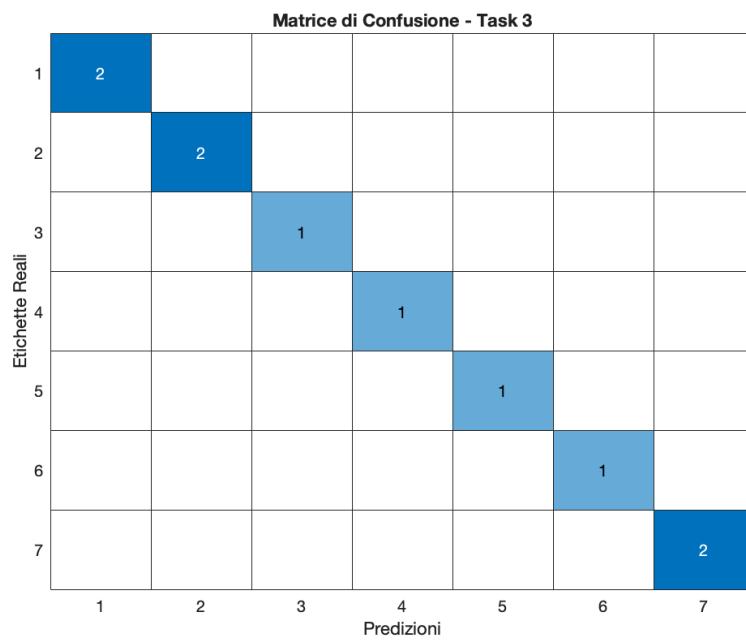


Figura 3.2: Matrice di confusione del classificatore per il Task 3.

4. Task 4

4.1 Localizzazione del guasto sulla valvola

L'obiettivo del Task 4 è individuare la valvola in cui si verifica il guasto per i casi identificati come Valve Fault nel Task 2. L'etichetta del guasto deve corrispondere a una delle quattro posizioni disponibili: SV1, SV2, SV3 e SV4.

4.2 Metodologia

Di seguito viene descritta la metodologia adottata.

4.2.1 Preparazione del dataset

Per costruire il dataset di training del modello, la tabella labeledData è stata filtrata selezionando solo i Case con etichette diverse da zero nella colonna Task4, ovvero quelli in cui il guasto su una valvola è noto.

Il test set è stato ottenuto filtrando i risultati del secondo classificatore del Task 2, selezionando esclusivamente i Case identificati come Valve Fault (etichetta pari a 3). Il modello predittivo ha il compito di assegnare a ogni Case una delle etichette da 1 a 4, corrispondenti alla valvola guasta.

4.2.2 Generazione e selezione delle feature

Il dataset di training è stato elaborato utilizzando una frame policy basata su finestre temporali con i seguenti parametri:

- Frame Size (FS) = 0.400 s;
- Frame Rate (FR) = 0.400 s.

Dalle finestre temporali sono state estratte diverse feature statistiche e frequenziali, tra cui:

- Media, mediana, percentili 25 e 75, varianza, integrale numerico, minimo e massimo;
- Spettro di potenza, frequenza di picco, somma dello spettro di potenza, deviazione standard e RMS nel dominio della frequenza.

Le feature selezionate sono state utilizzate per addestrare il modello di classificazione. La Figura 4.1 mostra un esempio della selezione delle feature più rilevanti.

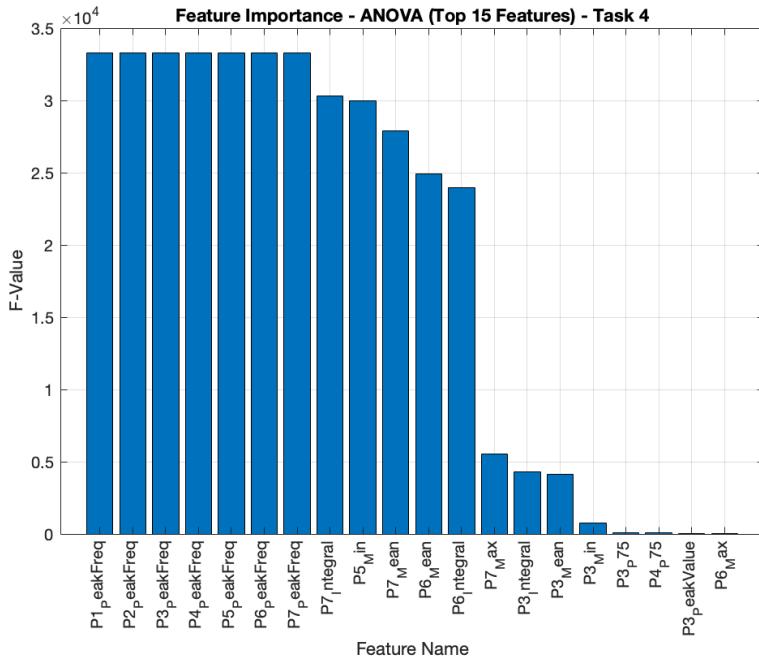


Figura 4.1: Selezione delle feature per il Task 4.

4.3 Modello di classificazione

Per la classificazione della posizione del guasto è stato adottato un modello di apprendimento supervisionato, validato tramite K-fold cross-validation con $K = 10$. Il modello è stato addestrato sui Case con posizione nota e testato sui Case etichettati come Valve Fault. Dopo aver valutato diverse configurazioni di modelli, il più efficace si è rivelato essere il modello **Bagged Trees**.

La predizione della posizione del guasto nel test set è stata ottenuta applicando il modello alle feature estratte per ogni finestra temporale. Successivamente, il majority voting è stato utilizzato per assegnare l'etichetta finale a ciascun Case, basandosi sulla predizione più frequente tra le finestre temporali associate allo stesso caso.

4.3.1 Valutazione dell'accuratezza del modello

L'accuratezza del modello è stata valutata confrontando le predizioni con le etichette reali, fornite dalla competizione, attraverso i seguenti passaggi:

- Caricamento delle etichette corrette dal file `answer.csv`;
- Filtraggio dei Case con etichette diverse da zero;
- Caricamento delle predizioni dal file `results.csv`;
- Aggregazione delle predizioni per Case con majority voting;
- Confronto tra etichette reali e predizioni per calcolare l'accuratezza.

Il modello ha raggiunto un'accuratezza dell'**85,71%**, dimostrando una buona capacità di localizzazione del guasto sulle valvole. Sebbene questo valore possa sembrare relativamente basso, è importante considerare che il modello commette un errore su sette casi disponibili nel test set. Questo livello di accuratezza è comunque adeguato per il problema affrontato, dato che le classi sono bilanciate e la difficoltà intrinseca della predizione è elevata.

La Figura 4.2 mostra la matrice di confusione ottenuta per il Task 4, evidenziando la distribuzione degli errori di classificazione tra le diverse classi.

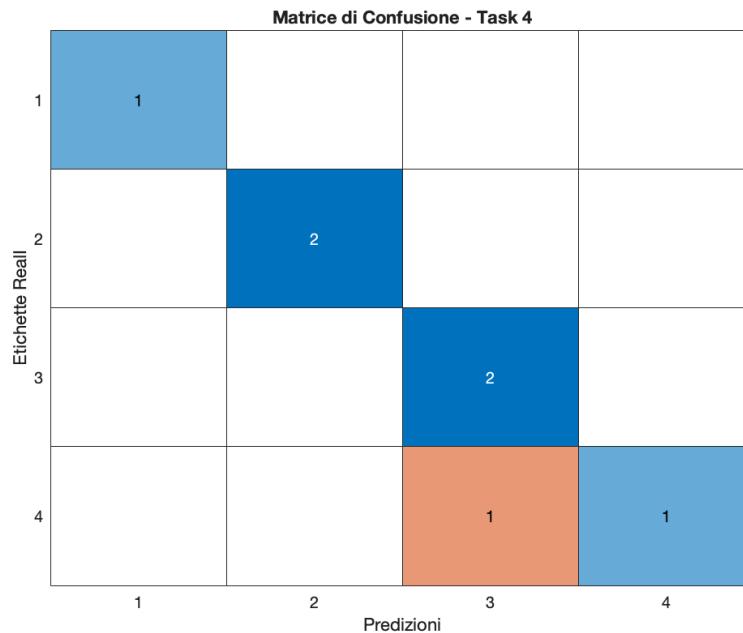


Figura 4.2: Matrice di confusione per il Task 4.

5. Task 5

5.1 Predizione della percentuale di apertura delle valvole

L'obiettivo del Task 5 è stimare, attraverso tecniche di regressione, la percentuale di apertura delle valvole (SV1, SV2, SV3, SV4) per i casi identificati come *Valve Fault* nel Task 4. A differenza del Task 4, che era un problema di classificazione, in questo caso l'obiettivo è predire un valore continuo compreso tra 0 e 100, corrispondente alla percentuale di apertura della valvola guasta.

Una previsione accurata di questo valore è cruciale per comprendere il livello di compromissione del sistema. Un'apertura parziale della valvola potrebbe indicare un malfunzionamento lieve, mentre un'apertura molto ridotta potrebbe suggerire un problema grave, richiedendo un intervento di manutenzione urgente.

5.2 Metodologia

La metodologia adottata per affrontare il problema della regressione segue i seguenti passaggi.

5.2.1 Preparazione del dataset

Per costruire il dataset di training, sono stati selezionati esclusivamente i casi in cui il valore relativo al Task 5 fosse diverso da 100. Questo implica che il dataset di training è costituito solo da valvole che non si sono completamente aperte, e quindi presentano un guasto che ne influenza il funzionamento.

Analogamente, il test set è stato ottenuto selezionando i Case etichettati con valore 3 dal Task 2, rappresentando guasti su una valvola. Questo approccio assicura che il modello venga allenato su dati effettivamente rappresentativi del fenomeno da prevedere.

5.2.2 Data Augmentation

Per migliorare la capacità di generalizzazione del modello e ridurre il rischio di overfitting, è stata applicata una tecnica di *Data Augmentation*. Il dataset di training è stato triplicato, e su ogni nuova istanza è stato aggiunto del rumore bianco gaussiano alle colonne numeriche.

(escludendo il tempo). Il livello di rumore è stato determinato come il 2% della deviazione standard della feature originale.

Questa strategia consente di simulare variabilità realistiche nei dati, migliorando la capacità del modello di adattarsi a situazioni non identiche ai casi osservati durante l'addestramento. Alla fine del processo, il dataset aumentato è stato mescolato casualmente per evitare pattern ripetitivi.

5.2.3 Generazione e selezione delle feature

Il dataset aumentato è stato caricato nel *Diagnostic Feature Designer* per la generazione delle feature, utilizzando una segmentazione temporale basata su finestre di dati (*frame policy*). I parametri scelti per questa operazione sono stati:

- *Frame Size (FS)*: 0.256s;
- *Frame Rate (FR)*: 0.256s.

Successivamente, è stata effettuata una selezione delle feature utilizzando il test di *Kruskal-Wallis*, un metodo non parametrico che permette di identificare le feature più rilevanti per il problema di regressione. Il numero finale di feature selezionate è pari a 52. La Figura 5.1 mostra una rappresentazione delle feature più significative.

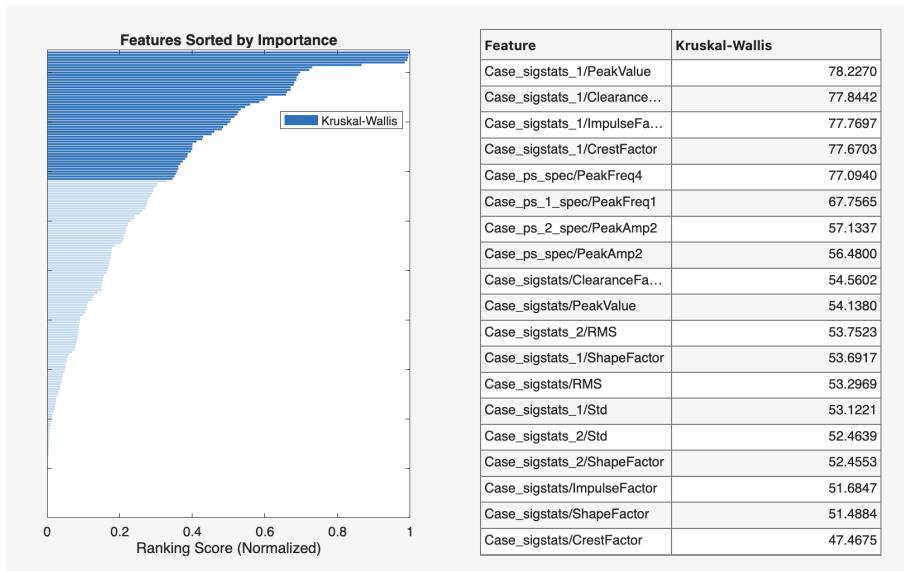


Figura 5.1: Selezione delle feature per il Task 5.

5.3 Scelta della regressione come tecnica di apprendimento

A differenza del Task 4, in cui l'obiettivo era classificare il guasto in una delle quattro valvole, nel Task 5 il valore da predire è continuo, poiché rappresenta la percentuale di apertura della valvola. In questi casi, le tecniche di regressione sono preferibili rispetto a quelle di classificazione, poiché permettono di stimare un valore numerico anziché assegnare un'etichetta discreta.

Dopo un'analisi comparativa tra diversi modelli di regressione, il modello più efficace si è rivelato essere **Bagged Trees**. Questo approccio combina più alberi decisionali, riducendo la varianza e migliorando la robustezza delle previsioni. Il modello è stato validato utilizzando *K-fold cross-validation* con $K = 10$, garantendo una valutazione affidabile delle prestazioni e minimizzando il rischio di overfitting.

5.4 Valutazione delle prestazioni del modello

Per valutare l'accuratezza del modello, le previsioni sono state confrontate con i valori reali del test set, forniti dalla competizione. La Figura 5.2 mostra il confronto tra i valori reali e quelli predetti dal modello.

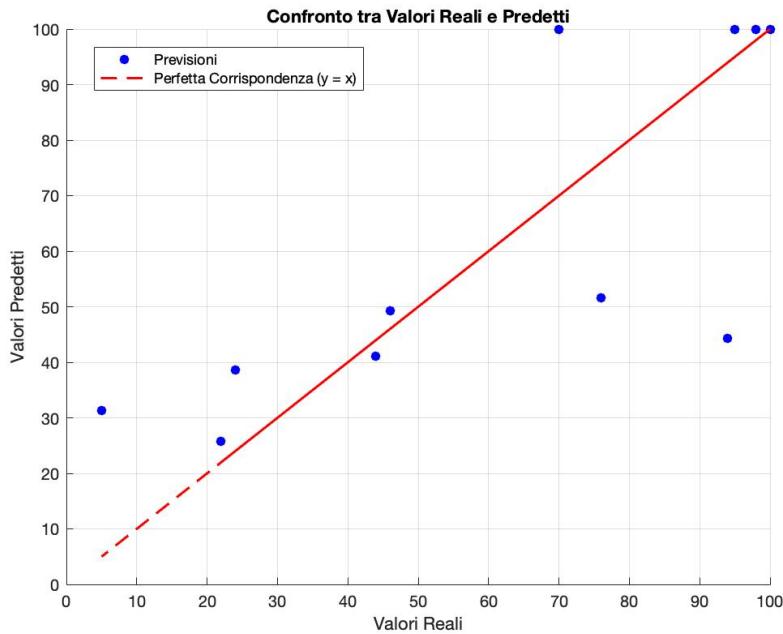


Figura 5.2: Confronto tra valori reali e predetti. La linea rossa rappresenta la corrispondenza perfetta ($y = x$).

Dall'analisi del grafico, si osserva che il modello segue abbastanza bene la distribuzione dei valori reali e cattura con precisione la maggior parte dei dati dimostra la sua solidità e affidabilità.

Le metriche utilizzate per valutare il modello sono:

- *RMSE (Root Mean Square Error)*: misura l'errore quadratico medio tra le previsioni e i valori reali. Penalizza maggiormente gli errori più grandi, garantendo che il modello sia accurato anche per i valori estremi.
- *MAE (Mean Absolute Error)*: misura l'errore medio assoluto tra le previsioni e i valori reali. Fornisce un'indicazione chiara dell'errore medio in termini di percentuale di apertura.

I risultati ottenuti sono stati i seguenti: **RMSE = 10.3506, MAE = 3.5204**.

Questi valori indicano che il modello ha raggiunto ottime prestazioni nella predizione della percentuale di apertura delle valvole guaste. Il RMSE relativamente basso conferma che il modello è in grado di effettuare previsioni precise, mentre il MAE suggerisce un errore medio ridotto, pari a circa 3.52 unità sulla scala da 0 a 100. Questo dimostra la robustezza del modello e la sua utilità nel supportare la manutenzione predittiva del sistema.