



UNIVERSITÀ DEGLI STUDI DI MODENA E REGGIO EMILIA

DIPARTIMENTO DI SCIENZE FISICHE, INFORMATICHE E
MATEMATICHE

Corso di Laurea Magistrale in Informatica

Manutenzione Predittiva Log-Based: Un Caso di Studio

Relatore:

Ch.mo Prof.

Riccardo MARTOGLIA

Candidato:

Mat. 160048

Sebastiano MANFREDINI

ANNO ACCADEMICO 2021/2022

*Dedicata a coloro che
ogni giorno mi hanno supportato
in questo cammino.*

INDICE

1	Introduzione	4
2	L'Azienda	5
2.1	Storia	5
2.2	Mercato	6
2.3	Prodotti	6
2.3.1	Dispositivi per il deposito	6
2.3.2	Dispositivi per il ricircolo	7
2.3.3	Dispositivi da front-office	9
2.3.4	Dispositivi per il self-service assistito	9
2.4	Software	10
2.4.1	C-Link	11
2.4.2	C-Service	12
3	Manutenzione predittiva	13
3.1	Approcci alla manutenzione	14
3.1.1	Manutenzione correttiva	14
3.1.2	Manutenzione preventiva	14
3.1.3	Manutenzione predittiva	16
3.1.3.1	Manutenzione predittiva log-based	17

INDICE

4	Implementazione	20
4.1	Analisi dei log	20
4.2	Modello	20
4.2.1	EDA	20
4.2.2	Generazione dei learning set	20
4.2.3	Estrazione delle feature	23
4.2.3.1	Feature statistiche	23
4.2.3.2	Feature pattern-based	24
4.2.4	Selezione delle feature	24
4.2.5	Algoritmi di labeling	25
4.2.5.1	Instance-MIL	25
4.2.5.2	Aggregated-MIL	25
4.2.5.3	Hybrid-MIL	26
4.2.6	Ottimizzazione del modello	26
4.2.7	Gestione di dati sbilanciati	26
5	Conclusioni	29
	Ringraziamenti	30
	Appendici	31
	Bibliografia	35
	Elenco delle figure	46
	Elenco delle tabelle	47
	Elenco degli acronimi	48

CAPITOLO 1

INTRODUZIONE

CAPITOLO 2

L'AZIENDA

2.1 Storia

CIMA S.p.A nasce nel 1955 e da oltre 55 anni si occupa di gestire la custodia e la movimentazione del denaro per terzi.

In questi anni la gamma di prodotti offerti è notevolmente aumentata ed evoluta. Inizialmente, il business principale dell'azienda erano le bussole antirapina bancarie. Una bussola antirapina è una porta rinforzata e ad ante tipicamente rotanti in grado di offrire un alto grado di sicurezza contro le rapine. In questa prima fase quindi, i clienti principali dell'azienda erano le banche.

Agli inizi degli anni '90 si è deciso di estendere la gamma di prodotti offerti iniziando a progettare, sviluppare e produrre dispositivi per il *cash handling* automatico. Per automated cash handling device si intendono dispositivi in grado di gestire la movimentazione di denaro, in deposito o anche in ricircolo, in maniera completamente automatica. Inizialmente, anche questi dispositivi erano destinati al mondo bancario ma, nel corso degli anni 2000, CIMA è entrata nel mercato del retail vendendo le proprie soluzioni a supermercati, autostrade ed in generale a negozi di piccole, medie e grandi

dimensioni, offrendo diversi dispositivi per le diverse esigenze.

La sede principale dell'azienda è a Mirandola (MO), ma CIMA opera in tutto il mondo sfruttando una fitta rete di distributori che si occupano della rivendita e dell'assistenza dei prodotti in America, Africa, Asia e Oceania.

2.2 Mercato

Il mercato principale di CIMA è quindi la gestione del denaro contante in ambito retail e bancario. Nel 2019, a livello mondiale, il 30.2% dei pagamenti in ambito retail veniva effettuato in contanti[1] in un mercato da 1.02 trilioni di dollari di ricavi nel 2018[2]. In Europa, il 73% delle transazioni POS e P2P veniva effettuata in contanti nel 2019[3]. Tra il 2020 e il 2021, il COVID-19 ha forzato la transizione verso pagamenti digitali e il contante ha perso circa il 16% delle quote di mercato[4]. Nonostante ciò, il contante rimane ad oggi la modalità di pagamento maggiormente utilizzata in molti paesi di Europa, Centro e Sud America, Asia e Africa[5]. Inoltre, tale decrescita in realtà ha rafforzato il mercato di cui CIMA fa parte poiché ha reso ancora più necessario l'abbattimento dei costi di gestione del denaro contante[4]. Tali costi possono essere fortemente diminuiti tramite l'utilizzo di sistemi automatici.

2.3 Prodotti

I prodotti attualmente offerti dall'azienda si possono suddividere in quattro categorie: a) dispositivi per il deposito e b) ricircolo del denaro; c) soluzioni di front office per la gestione dei pagamenti; d) dispositivi per il self-service assistito.

2.3.1 Dispositivi per il deposito

Un'unità da deposito è un dispositivo tipicamente utilizzato nel backoffice in ambito retail. Tale unità è suddivisibile in due parti. Nella parte inferiore vi è una cassaforte per la conservazione sicura del denaro depositato; nella parte

superiore vi è invece una bocchetta per l'inserimento del denaro, un'unità di trasporto, una bocchetta di uscita e un dispositivo per il riconoscimento e la verifica di autenticità del contante depositato. Tali dispositivi permettono quindi la conservazione sicura e il ritiro agevole e immediato del contante depositato. Un'altra funzionalità molto importante di tali dispositivi è la valuta in tempo reale. Le macchine da deposito CIMA infatti, permettono l'accredito istantaneo del depositato su un conto corrente tramite un istituto bancario che supporta tale funzionalità. Per garantire questa funzionalità è fondamentale che tale tipologia di macchine offrano un'operatività continua; solo un numero esiguo di downtime possono essere tollerati durante il ciclo di vita di un dispositivo di questo tipo.



(a) SDM 500, unità per il deposito delle banconote



(b) CDS 801, unità per il deposito delle monete

Figura 1: Esempi di dispositivi per il deposito di CIMA

2.3.2 Dispositivi per il ricircolo

Un'unità per il ricircolo del denaro è un dispositivo tipicamente utilizzato in ambito retail che consente sia il deposito che il prelievo del contante. Anche

Tutte le immagini di dispositivi CIMA sono protette da copyright ©2022 CIMA S.p.A

2. L'AZIENDA

in questo caso, le unità si possono dividere in una parte superiore, costituita da una bocchetta di ingresso e di uscita, un trasporto, un riconoscitore e una parte inferiore costituita da una cassaforte. In questo caso, nella cassaforte sono presenti più magazzini che permettono il ricircolo del denaro oltre che ad una sacca per il deposito del denaro.

L'ambito principale di utilizzo di questa tipologia di dispositivi è la grande distribuzione. In tale ambito, lo scopo principale di una macchina da ricircolo è quello di ottimizzare il processo di gestione del contante; in una giornata tipo, nelle prime ore del mattino il contante viene prelevato per rifornire le casse, mentre nelle ultime ore della sera il contenuto delle casse viene depositato per garantire sicurezza e anche un riconteggio immediato del contante movimentato durante la giornata.



(a) SDM 505, unità per il deposito delle banconote



(b) CDS 802, unità per il deposito delle monete

Figura 2: Esempi di dispositivi per il ricircolo di CIMA

2.3.3 Dispositivi da front-office

InLane è la soluzione di CIMA per le casse automatiche di front end. Tale unità consente l'automatizzazione della gestione del denaro nel punto cassa, il cosiddetto self-checkout. Per self-checkout si intende che è il cliente a provvedere alla scelta del prodotto e al relativo pagamento, inserendo il contante e percependo l'eventuale resto. Oltre al self-checkout, automatizzare la gestione del denaro contante rimuove l'onere di gestione del contante all'operatore del punto cassa; questo, oltre a permettere una maggior igiene, permette anche una maggiore sicurezza riducendo frodi e furti.

Una macchina da front-office combina quindi le capacità di una macchina da ricircolo ad uno schermo di diverse possibili dimensioni per l'interazione con il cliente finale.



(a) InLane 300



(b) InLane Kiosk

Figura 3: Esempi di dispositivi da front-office di CIMA

2.3.4 Dispositivi per il self-service assistito

Le unità per il self-service assistito sono dispositivi utilizzati in ambito bancario. Tali dispositivi combinano le funzionalità di un ATM e un TCR ma vengono inseriti all'interno della banca per consentire maggiore contatto

degli agenti bancari con i clienti della filiale. Alcune operazioni infatti possono essere eseguite solo essendo assistiti da un agente della filiale. CIMA offre la propria soluzione anche in questo ambito.

Tali dispositivi quindi sono unità per il ricircolo del denaro che però permettono anche la lettura e la gestione di documenti bancari per eseguire le operazioni in modalità assistita. Questo è reso possibile anche grazie ad uno schermo al di sopra della macchina che permette ai clienti di vedere l'andamento e i risultati delle operazioni che stanno eseguendo.



Figura 4: CasHere, dispositivo CIMA per il self-service assistito

2.4 Software

In CIMA viene seguito lo sviluppo completo di tutti i prodotti, dalla progettazione fino alla messa in produzione. Tutti gli aspetti progettuali, dal punto di vista meccanico, elettrico e informatico vengono gestiti internamente; il reparto di ricerca e sviluppo, che conta circa il 30% dei dipendenti totali, è quindi la parte fondamentale di un'azienda come CIMA che richiede il know-how completo dei dispositivi che commercializza.

Il software fa parte degli ambiti di cui CIMA ha il controllo completo. CIMA, infatti, progetta e sviluppa il firmware, CAOS, e il software dell'unità di riconoscimento delle banconote per tutte le macchine in produzione.

Oltre a ciò, CIMA sviluppa due applicativi per il controllo remoto e la diagnosi: C-Link e C-Service

2.4.1 C-Link

C-Link è la piattaforma di gestione remota delle macchina CIMA. Tale applicativo offre un'interfaccia web che permette di visualizzare lo stato della macchina e di effettuare diversi tipi di operazioni sui dispositivi collegati. C-Link è una piattaforma modulare composta da 4 moduli:

- service and updates management: modulo che permette l'aggiornamento remoto e lo scaricamento dei log delle macchine collegate
- cash flow management: modulo che permette il monitoraggio remoto del contenuto delle macchine
- security and configuration management: modulo per la gestione remota della configurazione e dei parametri di sicurezza delle macchine
- fitness statistics: modulo che permette la generazione di report per la banca centrale europea del conteggio di banconote valide e invalide rilevate

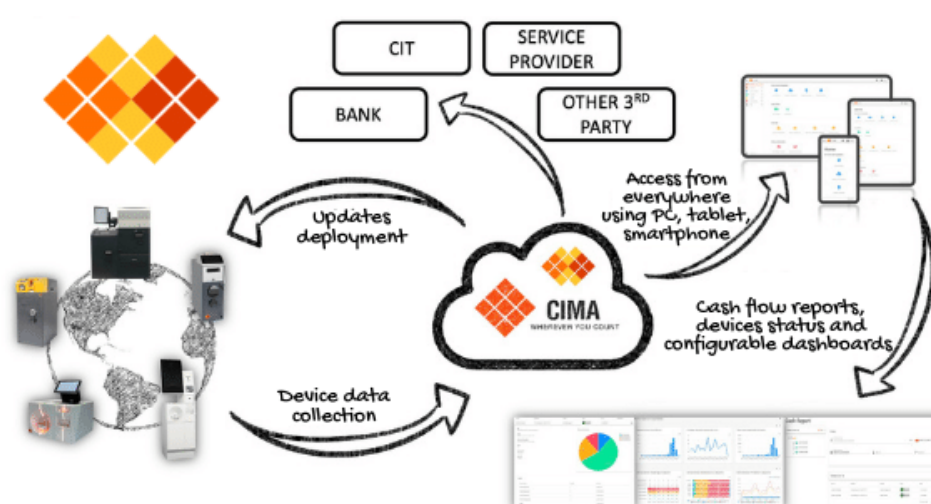


Figura 5: Schema delle funzionalità di C-Link

C-Link viene offerto in due modalità: on-premise o in cloud. Nella modalità on-premise, C-Link viene installato direttamente sui server del cliente; nella modalità cloud, C-Link viene installato su server AWS e viene offerto ai clienti in modalità SaaS.

2.4.2 C-Service

C-Service è un tool ad uso interno di CIMA. La funzionalità principale dell'applicativo è l'analisi dei log prodotti dalle macchine. Tale software, quindi, viene principalmente utilizzato dai tecnici in campo durante le installazioni o gli interventi di manutenzione per capire lo stato della macchina. C-Service non permette solamente di visualizzare i log *raw* della macchina, ma è in grado di analizzare e interpretare i dati scaricati per fornire delle analisi. Tali analisi contengono grafici e tabelle che riassumono i dati di log e permettono ai tecnici di comprendere meglio e più velocemente lo stato del dispositivo analizzato.

C-Service, oltre a permettere lo scaricamento e l'analisi dei log passati, permette anche di collegarsi tramite porta USB o tramite rete alla macchina per controllare in diretta lo stato del dispositivo monitorato.

Tale strumento viene quindi utilizzato sia dai tecnici in campo che dagli ingegneri del centro ricerca e sviluppo per comprendere cosa non funziona e/o cosa è migliorabile di un modello di macchina in produzione o che sta ancora subendo il processo di test e validazione.

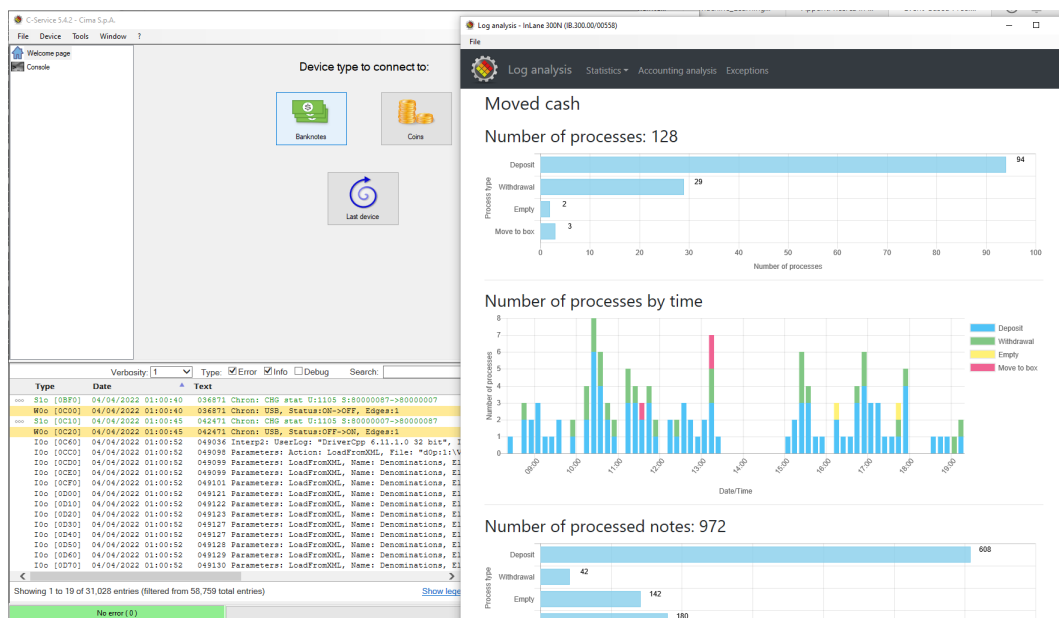


Figura 6: Esempio di schermata di C-Service con un'analisi prodotta

CAPITOLO 3

MANUTENZIONE PREDITTIVA

Il processo di manutenzione è definito come un insieme di attività effettuate per riportare il sistema in uno stato in cui può svolgere le funzioni a lui designate[6].

La manutenzione è una parte fondamentale nel ciclo di vita di un prodotto. Il successo di un'azienda infatti è strettamente legato all'affidabilità dei prodotti che mette in commercio e alle strategie di manutenzione applicate[7]. Applicare strategie di manutenzione inefficaci, oltre a generare costi aggiuntivi enormi[8], impatta in maniera sostanziale anche nella reputazione con i clienti[9].

Esiste quindi una forte motivazione economica dietro l'investimento in nuove tecniche di manutenzione. Nel 2021 si stima che i costi e le perdite associate alla manutenzione in ambito manifatturiero ammontino a 222 miliardi di dollari nei soli USA[10]. Si stima inoltre che, nella moderna industria 4.0, i costi connessi alla manutenzione rappresentino dal 15% al 45% dei costi di produzione totali dei beni[11]. Minimizzare tali costi può quindi condurre anche ad un vantaggio competitivo non trascurabile.

3.1 Approcci alla manutenzione

Esistono tre approcci fondamentali alla manutenzione in ambito industriale[12]: a) manutenzione correttiva o *Run-to-failure*, b) manutenzione preventiva e c) manutenzione predittiva.

3.1.1 Manutenzione correttiva

Nell'approccio R2F gli interventi manutentivi di riparazione o sostituzione dei componenti vengono effettuati in conseguenza a un guasto[13]. Non si fa quindi nessuno sforzo per cercare di anticipare il downtime ma si agisce esclusivamente per correggere un failure. Questo è l'approccio più semplice e intuitivo ma è anche il meno utile. Infatti, questo approccio è il più costoso fra i tre[14]. I costi derivano principalmente dalle spese dovute all'intervento correttivo e dalla lunga durata del downtime della macchina.

Bisogna inoltre considerare che se si decide di adottare la R2F, è assolutamente necessario avere disponibilità immediata di tutti i pezzi di ricambio necessari durante gli interventi. I costi di questo approccio, quindi, derivano anche dall'alto costo di conservazione in magazzino di una grande quantità e varietà di pezzi di ricambio o, se si decide di non aver un magazzino completamente fornito, dal costo di acquisto e spedizione immediata presso i rivenditori di componenti sostituivi.

Nonostante questo sia un approccio manutentivo largamente adottato in ambito manifatturiero[10], è immediatamente chiaro che, se utilizzato in maniera esclusiva, non è un approccio sostenibile sia dal punto di vista dei costi che di reputazione con il cliente.

3.1.2 Manutenzione preventiva

L'approccio preventivo alla manutenzione ha diverse definizioni. In [15] viene definita come una strategia di pianificazione a lungo termine per eseguire azioni manutentive all'interno di intervalli di tempo predeterminati. Durante la fase di pianificazione, tipicamente si tiene conto del tempo passato dall'ultimo

3. MANUTENZIONE PREDITTIVA

intervento manutentivo o delle ore di operatività della macchina[14]. Per questo motivo, spesso la manutenzione preventiva viene indicata anche come *Time-based maintenance*[15, 16].

L'obiettivo di questa strategia è quindi di cercare di prevenire i failure intervenendo prima che questi avvengano. Questo approccio permette di ridurre i costi legati ai failure e al downtime collegato aumentando quindi la qualità e l'affidabilità generale del prodotto[17].

In ambito industriale, tipicamente la pianificazione degli intervalli di tempo per la manutenzione preventiva viene effettuata tramite l'esperienza di esperti di dominio[16]. Spesso infatti, l'idea di base è quella di applicare gli interventi di manutenzione a intervalli regolari di lunghezza fissa[18]. La lunghezza di tali intervalli viene stimata da ingegneri e tecnici basandosi sull'osservazione dell'operatività della macchina e sulla loro esperienza[16]. Ovviamente, un approccio di questo tipo è soggettivo, non riproducibile e non efficiente. I tecnici deputati all'osservazione delle macchine infatti devono svolgere questo compito per settimane se non mesi e nel momento in cui viene sviluppata una nuova macchina l'intero processo deve essere ripetuto da capo.

La manutenzione preventiva è quindi un approccio che risulta efficace nel ridurre il downtime della macchine ma non è un approccio scalabile. Inoltre, tale approccio alla manutenzione spesso porta a costi aggiuntivi inutili.

Si stima infatti che un terzo dei costi di manutenzione siano riconducibili a interventi inutili o errati[14] ed in particolare che più del 50% degli interventi di manutenzione preventiva siano inutili[19]. Per inutili, si intende che i componenti soggetti a manutenzione preventiva vengono sostituiti prima che vi sia un bisogno reale di sostituzione. Questo porta a costi aggiuntivi non necessari dato che il ciclo di vita di tali componenti viene terminato prima che essi diventino effettivamente improduttivi.

La scalabilità e i costi aggiuntivi sono quindi i principali limiti dell'approccio preventivo alla manutenzione.

3.1.3 Manutenzione predittiva

Come nel caso della manutenzione preventiva anche la manutenzione predittiva viene definita in vari modi. In [14] viene definita come un programma completo di gestione della manutenzione che utilizza gli strumenti più efficaci e convenienti a disposizione per ottenere le condizioni operative reali dei macchinari e, basandosi su tali dati, programma gli interventi manutentivi in base all'effettiva necessità. In [20] viene chiamata *manutenzione prognostica* e viene definita come il processo di predizione dello stato futuro o del RUL della macchina basandosi sulle condizioni storiche o attuali di essa. In [16] viene anche chiamata *manutenzione condition-based* e viene definita come un approccio alla manutenzione in cui vengono pianificate le azioni di manutenzione sulla base delle informazioni raccolte attraverso il processo di monitoraggio delle condizioni dei macchinari.

In generale quindi, l'obiettivo della manutenzione predittiva è quello di cercare di intervenire preventivamente ma solo quando è effettivamente necessario, azzerando quindi gli interventi manutentivi inutili.

Una delle parti fondamentali della manutenzione predittiva è quindi il sistema di monitoraggio. Tipicamente, il sistema di monitoraggio viene realizzato aggiungendo sensori ai macchinari per registrare e monitorare i segnali di interesse. In questo senso, la tecnica più utilizzata si basa sul monitoraggio delle vibrazioni[21, 22, 23]. In questo approccio lo stato di salute della macchina viene monitorato con l'ausilio di sensori di vibrazione, per rilevare anomalie che potrebbero indicare danni o degrado[16]. Un'altra tecnica molto utilizzata prevede il monitoraggio di segnali acustici[24, 25]. Spesso queste due tecniche di monitoraggio e processing dei segnali vengono utilizzate in congiunzione[26]. Altre tecniche prevedono il monitoraggio di condizioni chimiche[27], elettriche[28] o fisiche[29].

Nella PdM, per eseguire le predizioni, si può utilizzare un approccio *data-driven* o *model-based*[30, 31]. Nell'approccio data-driven vengono sfruttate diverse tecniche di data mining e machine learning per apprendere il comportamento del sistema basandosi sui dati storici registrati[30, 32, 33,

34]. Tale approccio ha dimostrato di ottenere grandi successi in diversi settori industriali, diventando ad oggi la principale tecnica di PdM utilizzata[34, 35]. Nel secondo approccio invece, si sfrutta l'esperienza di esperti di dominio per definire un modello analitico o un set di regole che rappresentano il comportamento del sistema[36]. Come nel caso della manutenzione preventiva, sfruttare le competenze di esperti di dominio non è un approccio scalabile. L'approccio *Rule-based*, infatti, può essere utilizzato solo in sistema in piccola scala; mantenere aggiornati manualmente i set di regole che rappresentano il sistema richiede molto tempo e sforzo da parte degli esperti, sforzo che non è sostenibile su sistemi in larga scala[37].

La manutenzione predittiva basata sul monitoraggio di dati da sensori di varia natura ha dimostrato di essere efficace in molti settori. In [38] si evidenzia che tale approccio ha permesso una riduzione dal 27% al 40% dei costi di manutenzione in diversi ambiti industriali. In [9] si evidenzia che con la PdM, i downtime pianificati e non pianificati, i costi di manutenzione, la dimensione delle giacenze in magazzino e gli interventi manutentivi inutili possono essere sensibilmente diminuiti.

Nonostante ciò, bisogna considerare che, in confronto alla PvM e alla CM, i costi della PdM sono tipicamente più alti[9]. Ciò è dovuto al fatto che spesso può risultare molto costoso o impraticabile per motivi logistici o tecnici, aggiungere sensori per il monitoraggio dei macchinari. Altre volte, i dati su cui basare la stima dello stato della macchina, non sono rilevabili tramite sensori[39]. In questi casi esiste un'alternativa: utilizzare i dati di log[40].

3.1.3.1 Manutenzione predittiva log-based

Nella manutenzione predittiva *log-based* i dati utilizzati per la stima dello stato della macchina vengono estratti dagli event-log del software di gestione della macchina o del sistema IT a cui il dispositivo è connesso[41]. Ad esempio, nel caso di dispositivi per l'automated cash handling, tutte le operazioni, come i depositi, prelievi, ritiri così come gli errori e le eccezioni, vengono salvati nei log della macchina[37, 42, 43].

3. MANUTENZIONE PREDITTIVA

La PdM log-based è quindi un approccio *data-driven* in cui vengono processati i dati estratti dai log per tentare di derivare pattern di failure[30]. Tali pattern vengono poi sfruttati nella fase di pianificazione degli interventi di manutenzione per intervenire sulle macchine nel prima che avvengano downtime.

Oltre ai costi minori, un altro vantaggio nell'utilizzare i log invece che i dati raw provenienti dai sensori è che i log sono tipicamente di dimensione minore e sono facilmente accessibili e semplici da trasmettere ad un unità centrale che si occupa delle predizioni[44].

Come già spiegato nella sezione precedente, la tecnica decisionale maggiormente utilizzato in ambito PdM si basa sull'utilizzo di diverse tecniche di machine learning. Il machine learning si può suddividere in due macroaree: learning supervisionato e non supervisionato[45]. Esistono due filoni di ricerca anche in ambito PdM log-based data-driven, ognuno dei quali afferenti ad una delle due tecniche di machine learning.

Learning non supervisionato In [42] viene presentato un caso di studio di applicazione della manutenzione predittiva basata sui log di macchine ATM. Nell'articolo gli autori decidono di utilizzare un metodo non supervisionato di change-point detection. Change-point detection è una tecnica di data mining che consiste nel trovare cambiamenti bruschi nei dati quando una o più proprietà di una serie temporale cambiano[46]. Nell'ambito della manutenzione predittiva, questo cambio di comportamento nei dati corrisponde a una degradazione dello stato delle macchine.

Per fare ciò, nell'articolo viene presentata una tecnica che considera finestre di tempo la cui durata viene progressivamente aumentata di 7 giorni fino a che non avviene un failure. Ad ogni iterazione, che corrisponde ad una estensione della finestra temporale, si tenta di trovare il punto di cambiamento.

Le tecniche di CPD richiedono ovviamente dati numerici su cui lavorare; nell'articolo, per trasformare gli eventi estratti dai log in dati numerici, semplicemente ne viene contata la frequenza.

Nessuna tecnica presentata nell'articolo porta a risultati soddisfacenti.

Learning supervisionato Nell'utilizzo del learning supervisionato in ambito di manutenzione preventiva log-based, il principale problema da affrontare è la trasformazione del problema di predizione in un problema di classificazione. In particolare, la maggior parte degli articoli in letteratura trasforma tale problema in un problema di classificazione binaria[33, 35, 37, 40, 43, 47, 48, 49]. Tale problema viene risolto tramite il cosiddetto metodo della finestra mobile[37]. La tecnica della finestra mobile verrà spiegata nella sottosezione 4.2.2.

Esiste poi un filone di ricerca alternativo in cui si decide di trasformare il problema di predizione in un problema di regressione. In machine learning, i metodi di regressione hanno l'obiettivo di prevedere il valore di variabili numeriche continue in cui è definita una relazione di ordine[50]. In [44] e [51] l'idea è di partire dai valori continui di sensori dell'emissione acustica e da tali valori vengono generati artificialmente eventi discreti. Gli eventi vengono partizionati in range temporali che a loro volta sono suddivisi in sottosegmenti di una durata pari alla granularità temporale dell'analisi. L'obiettivo in questo caso è di far imparare al modello una funzione che quantifichi il rischio che avvenga un failure nel prossimo futuro basandosi sugli eventi del recente passato. Per quantificare il rischio viene utilizzata una sigmoide che mappa valori più alti ai segmenti che sono più vicini al failure della macchina. Questo viene fatto poiché si reputa che gli eventi più vicini al failure sono più indicativi ed importanti rispetto a quelli più lontani.

Tra i due approcci, la classificazione binaria ottiene risultati migliori ed è un metodo più robusto rispetto alla regressione[30].

CAPITOLO 4

IMPLEMENTAZIONE

4.1 Analisi dei log

4.2 Modello

4.2.1 EDA

4.2.2 Generazione dei learning set

Visti i risultati in letteratura evidenziati nella sottosottosezione 3.1.3.1, si è deciso di trasformare il problema di predizione in un problema di classificazione tramite la tecnica della finestra mobile. Tramite tale tecnica è possibile generare un numero configurabile di istanze per il training e il test set.

In questa tecnica vengono incrociati i dati provenienti dai log con i dati provenienti dai sistemi di ticketing utilizzati per registrare gli interventi tecnici condotti sulle macchine.

La Figura 7 illustra la tecnica. Per generare le istanze per il training e il test set viene considerato un punto nel tempo in cui il modello esegue la predizione, il *prediction point*. Per ogni prediction point viene generata un

4. IMPLEMENTAZIONE

istanza costituita da un insieme di feature e una label. Tale istanza viene suddivisa in tre finestre temporali:

- *observation window*, finestra di tempo in cui si accumulano i dati da utilizzare per la predizione. Tali dati includono il contante processato (PC), gli scarti (R), le eccezioni (ME) e i dati di calibrazione dei sensori (SC). Ogni observation window è ulteriormente suddivisa in sotto-finestre di una durata pari alla granularità dell'analisi. Nel caso di CIMA, i dati di log vengono analizzati e aggregati ogni 24 ore, quindi una observation window di lunghezza X avrà una durata complessiva di X giorni.
- *buffer window* o *response window*, finestra che tipicamente viene inserita per motivi tecnici legati al tempo necessario per intervenire; rappresenta il tempo minimo che deve trascorrere tra la predizione e il primo failure possibile. Inserendo una finestra tra la buffer window e la prediction window si dà tempo ai tecnici in campo di raggiungere e riparare la macchina per cui è stata effettuata una predizione positiva prima che avvenga il downtime. La lunghezza della buffer window è un altro parametro Y .
- *prediction window*, finestra di tempo all'interno del quale possono avvenire i failure predetti. Durante la fase di generazione delle istanze per il training e test set, se è presente un intervento tecnico (CSL) nella PW la label associata all'istanza sarà positiva, altrimenti sarà negativa. La lunghezza della prediction window è un altro parametro Z .

La durata di queste finestre non è costante in letteratura poiché è strettamente legata al dominio studiato.

Spostando nel tempo il prediction point si riescono a generare diverse istanze di training o di test. In particolare, durante la fase di generazione degli insiemi si va a spostare il prediction point di N giorni per ottenere le diverse istanze. Nel caso di CIMA, N è stato fissato a 1 giorno.

Generando in questo modo il training set si ottiene un problema MIL[52]. Nel learning supervisionato classico, durante la fase di training il modello riceve

4. IMPLEMENTAZIONE

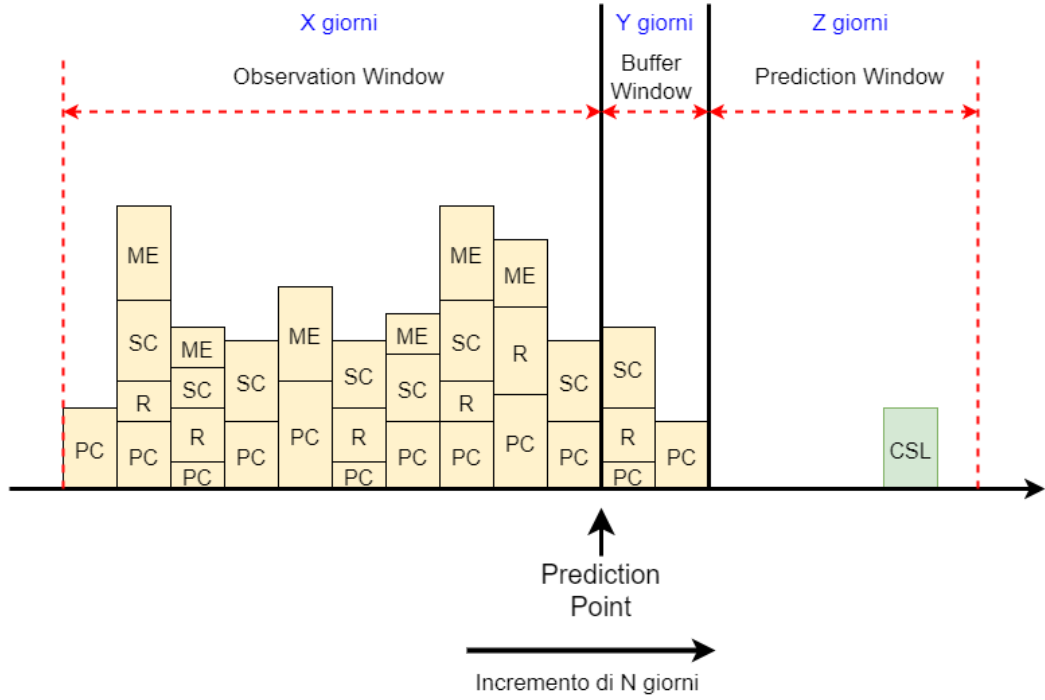


Figura 7: Schema di funzionamento della tecnica di finestra mobile

un insieme di esempi labellati in maniera indipendente; nel *Multiple Instance Learning* invece, durante la fase di allenamento, il modello riceve un insieme di *bag* a cui è associata una label[53]. Una *bag* è un insieme di esempi in una qualche relazione fra di loro. Nell'assunzione standard di MIL, ad una bag viene associata una label negativa se tutti gli esempi nella bag sono negativi, mentre ad una bag viene associata una label positiva se almeno un esempio risulta essere positivo[53].

Come spiegato in [40], trattare il problema della manutenzione predittiva log-based come un problema MIL ha diversi vantaggi:

- a) è più realistico assumere che almeno un log poco prima di un failure mostri i segnali di un imminente failure piuttosto che assumere che tutti i log poco prima del failure mostrino i segnali di un failure. Tale tipologia di macchine infatti, tendenzialmente non falliscono da un giorno all'altro, ma gradualmente il loro stato degrada passando da stati di normale attività ad anomali fino all'evento di failure.

- b) All'interno di una finestra di tempo in cui la macchina si trova in uno

stato di normale operatività, tutti i log non evidenziano segnali di failure imminenti

- c) L'obiettivo in questo tipo di problema è quello di predire in maniera corretta una bag e non la singola istanza.

4.2.3 Estrazione delle feature

Prima della fase di addestramento del modello viene effettuata una fase di feature extraction. La feature extraction è la fase durante la preparazione dei dati in cui vengono applicate delle trasformazioni alle feature presenti nel dataset per ottenere feature aggiuntive più significative e discriminanti per la classificazione[54]. Durante la fase di preparazione dei dati vengono estratte due tipologie di feature: statistiche e pattern based.

4.2.3.1 Feature statistiche

Durante la fase di feature extraction vengono generate diverse tipologie di feature statistiche. L'obiettivo principale è quello di far apprendere al modello correlazione fra la distribuzione statistica di eccezioni e scarti e gli eventi di failure. Esempi di feature statistiche base sono il numero di scarti calcolato per ogni causa di scarto, la media e la deviazione standard dei valori di calibrazione dei sensori per ogni sensore, il conteggio delle eccezioni per ogni tipologia.

Vengono poi estratte feature statistiche più avanzate calcolate a livello di bag come la distanza temporale minima, massima e media delle eccezioni dal punto di predizione e il *MTBE*, il tempo medio fra due eccezioni. Il *MTBE* per una bag i viene calcolato come:

$$MTBE_i = \frac{\sum_{j=0}^{N_i-2} (sdif(e_j, e_{j+1}))}{N_i}$$

dove N_i è il numero di eccezioni nella bag i , e_j rappresenta l'istante di avvenimento dell'eccezione j e *sdif* è la funzione che calcola la differenza in secondi fra due timestamp.

4.2.3.2 Feature pattern-based

Durante la fase di feature extraction vengono inoltre generate feature *pattern-based*. L'obiettivo di tale tipologia di feature è far apprendere al modello le relazioni fra sequenze di eccezioni e failure.

In [37] i pattern vengono definiti come combinazioni di eccezioni che si ripetono in diverse OW. L'idea è quindi innanzitutto quella di calcolare le combinazioni non ordinate senza ripetizioni dei codici delle eccezioni che avvengono in una observation window. Ad es se in una OW avvengono le eccezioni e_1, e_2, e_3 , i pattern generati saranno $\langle e_1 \rangle, \langle e_2 \rangle, \langle e_3 \rangle, \langle e_1, e_2 \rangle, \langle e_1, e_3 \rangle, \langle e_2, e_3 \rangle, \langle e_1, e_2, e_3 \rangle$. Al termine della generazione di tutti i pattern possibili, vengono mantenuti solamente i pattern la cui confidenza supera un valore di soglia pari a 0.90. La confidenza di un pattern viene calcolata come il rapporto del numero di volte in cui un pattern appare in una bag positiva diviso il numero di volte in cui il pattern appare in una qualsiasi bag. In questo modo si utilizzano come feature solo quei pattern effettivamente discriminanti per le bag positive.

4.2.4 Selezione delle feature

Durante la fase di preparazione dei dati, si è deciso di testare diverse tecniche per la *feature subset selection*. L'idea di base della FS è di rimuovere le feature ridondanti o irrilevanti dal dataset in quanto tali feature potrebbero portare a una riduzione dell'accuratezza della classificazione e a un inutile aumento del costo computazionale[55].

In questo senso, sono state testati diversi approcci sia basati su metodi di tipo *Filter* che su metodi tipo *Wrapper*[56]. In ogni caso, in tutti i test effettuati durante il progetto, la FS ha condotto a risultati peggiori in termini di accuratezza della classificazione rispetto ai risultati ottenuti sottoponendo al learner tutte le feature. Inoltre, il vantaggio in termini di riduzione del costo computazionale non era tale da giustificare il degrado prestazionale.

Per questo motivo si è deciso di abbandonare l'utilizzo della tecnica.

4.2.5 Algoritmi di labeling

Durante lo sviluppo del modello sono stati testati tre algoritmi di labeling. Per algoritmi di labeling si intendono gli algoritmi per la trasformazione del problema MIL in un problema di learning supervisionato classico.

4.2.5.1 Instance-MIL

L'algoritmo di *Instance-MIL*[57], anche noto come *AllInstances*[40] è un algoritmo di labeling *instance-to-example* puro in cui ogni istanze di una bag diventa un esempio la cui label corrisponde a quella della bag. Quindi, in questo algoritmo di labeling, tutti le istanze in una bag positiva diventano esempi positivi e tutti le istanze in una bag negativa diventano esempi negativi. Tutte le istanze di tutte le bag vengono concatenate in unico dataset D_{all} definito come

$$D_{all} = \{(X_{ij}, y_i), i = 1 \dots B, j = 1 \dots b\}$$

dove B rappresenta il numero di bag e b rappresenta il numero di istanze di ogni bag. Nella fase di testing vengono sottoposte al classificatore le singole istanze di ogni bag per la predizione.

Nonostante la semplicità dell'approccio, tale algoritmo ha dimostrato di ottenere buoni risultati in diversi dataset[40, 58]

4.2.5.2 Aggregated-MIL

L'algoritmo di *Aggregated-MIL* è un algoritmo di labeling *meta-example* puro in cui per ogni bag viene generato un metaesempio come la media delle istanze della bag[57]. Il generico esempio (X_i, y_i) viene calcolato come

$$X_i = \text{mean}(\{X_{ij}, j = 1 \dots b_i\})$$

dove b_i è la lunghezza della bag i e *mean* è la funzione di media.

Nella fase di testing, anche le bag di test vengono mediate e viene sottoposto il meta-esempio al classificatore per la predizione.

4.2.5.3 Hybrid-MIL

L'algoritmo di *Hybrid-MIL* è un algoritmo di labeling ibrido *instance-to-example* e *meta-example*. In questo algoritmo ogni esempio in una bag negativa diventa un esempio negativo; per le bag positive viene calcolato un meta-esempio positivo come la media degli esempi nella bag[40]. Nella fase di testing vengono sottoposte al classificatore le singole istanze di ogni bag per la predizione. Il dataset D_{hybrid} sarà definito come

$$D_{hybrid} = \begin{cases} \{(X_{ij}, y_i), i = 1...B, j = 1...b\}, & \text{se } y_i = 1 \\ (mean(\{X_{ij}, j = 1...b\}), y_i), & \text{altrimenti} \end{cases}$$

dove B rappresenta il numero di bag e b rappresenta il numero di istanze di ogni bag.

4.2.6 Ottimizzazione del modello

Durante la fase di addestramento, è stata eseguita anche una fase di ottimizzazione del modello. In particolare, è stata utilizzata la tecnica della cross-validazione per stimare i valori ottimali degli iperparametri del modello.

Per l'ottimizzazione del modello è stata utilizzata la cosiddetta *grid search*, una tecnica di tuning in cui si performa una ricerca completa in un sottospazio dello spazio degli iperparametri dell'algoritmo utilizzato[59]. I vincoli del sottospazio e i valori ottimali trovati sono indicati nella Tabella A1, nella Tabella A2 e nella Tabella A3.

4.2.7 Gestione di dati sbilanciati

La caratteristica chiave che distingue la PdM dalla classificazione tradizionale è che nel caso della manutenzione predittiva gli eventi di failure sono molto rari[51]. Questo, banalmente, deriva dal fatto che una macchina in campo è molte più volte in stato operativo rispetto ad in stato di failure. In caso contrario si avrebbe un macchinario non funzionante.

Questo conduce ad un dataset fortemente sbilanciato. Un dataset per la classificazione binaria si dice sbilanciato quando il rapporto del numero di esempi delle due classi è fortemente a favore di una delle due. Utilizzare un dataset sbilanciato può causare diversi problemi soprattutto nel caso della classificazione binaria. Quando infatti una classe rappresenta un concetto e l'altra classe rappresenta la controparte di tale concetto, se una delle due classi è in un numero fortemente superiore rispetto all'altra, il bias induttivo degli algoritmi di learning tende a concentrarsi esclusivamente sulla classe sovrarappresentata[60, 61]. Sostanzialmente, nella fase di predizione, gli algoritmi allenati in maniera sbilanciata tendono a classificare tutti gli esempi con la label della classe in forte maggioranza. Questo problema ovviamente non è accettabile dato che l'obiettivo di un algoritmo di PdM è proprio quello di scaturire allarmi in maniera preventiva quando lo stato di una macchina degrada.

Per gestire un dataset fortemente sbilanciato esistono tre principali approcci[62]: a) a livello di dati, b) a livello di algoritmo o c) approcci ibridi. In questo progetto sono stati utilizzati approcci a livello di dati. Sono quindi state testate diverse tecniche di sampling. Esistono due principali tecniche per risolvere tale problema a livello di dati: *over-sampling* e *under-sampling*. Per ottenere un dataset bilanciato, nel primo caso gli esempi della classe minoritaria vengono replicati, nel secondo caso invece gli esempi della classe maggioritaria vengono rimossi[63]. Sono stati testati 4 metodi:

- SMOTE: Synthetic Minority Over-sampling Technique è una tecnica di over-sampling in cui vengono generati degli esempi cosiddetti sintetici a partire dagli esempi reali della classe minoritaria. Per generare gli esempi artificiali viene utilizzato l'algoritmo di K-Nearest neighbor[64]. In questo caso, la classe minoritaria nei dataset delle macchine CIMA, viene portata da un rapporto di circa 1:100 ad un rapporto 1:2.
- SMOTE + RUS: tecnica che combina sia oversampling con tecnica SMOTE che under-sampling con tecnica randomica. La tecnica consiste

nell'eseguire un over-sampling iniziale tramite SMOTE per portare le classi del dataset da un rapporto di circa 1:100 tra label positive a label negative, ad un rapporto 1:10 aumentando il numero di esempi positivi. Dopodiché vengono rimossi randomicamente dei sample negativi(RUS, Random Under-Sampling[63]) per portare le classi ad un rapporto di circa 1:2.

- SMOTE + Tomek: tecnica che combina sia l'over-sampling con tecnica SMOTE che under-sampling con tecnica Tomek links[65]. L'idea dietro tale tecnica è di evitare l'overfitting dovuto all'over-sampling tramite SMOTE[66]. Tomek infatti, non viene applicato solo alla classe maggioritaria, ma viene utilizzato come metodo di data cleaning, eliminando esempi di entrambe le classi. Al termine del procedimento il rapporto label positive e negative è circa 1:1.
- SMOTE + ENN: tecnica che combina sia l'oversampling con tecnica SMOTE che under-sampling con tecnica Edited Nearest Neighbour[67]. Come in SMOTE + Tomek, l'idea è quella di evitare l'overfitting come effetto collaterale dell'over-sampling, utilizzando in questo caso ENN come metodo di data cleaning. ENN tende a rimuovere più esempi rispetto a Tomek; tendenzialmente quindi viene eseguita una pulizia più approfondita[66]. Al termine del procedimento il rapporto label positive e negative è circa 1:1.

In questo progetto sono stati utilizzati approcci anche a livello algoritmico. Infatti, utilizzare algoritmi di tipo *Ensemble* è uno dei metodi più adottati in letteratura per gestire dataset sbilanciati[68, 69].

CAPITOLO 5

CONCLUSIONI

RINGRAZIAMENTI

APPENDICI

Appendice A

Tabella A1: Range degli iperparametri e valori ottimali stimati per modello Inlane

Modello	Range iperparametri	Iperparametri ottimali(Aggregated)	Iperparametri ottimali (AllInstances)
Random Forest	Number of trees = [100, 200, 400, 800, 2000] Max depth = [10, 20, 80, 110, None] Min samples to split internal node = [2, 5, 10] Min sample to be leaf = [1, 2, 4] Bootstrap = [True, False]	Number of trees = Max depth = Min samples to split internal node = Min sample to be leaf = Bootstrap = True	Number of trees = Max depth = Min samples to split internal node = Min sample to be leaf = Bootstrap = True
Gradient Boosting	Learning rate = [0.05, 0.1, 0.2] Max depth = [5, 6, 7, 8, 9] Number of trees = [50, 100, 200] Sample rate= [0.7,0.8,0.9]	Learning rate = Max depth = Number of trees = Sample rate =	Learning rate = Max depth = Number of trees = Sample rate =
AdaBoost	Number of trees = [50, 100, 200, 400]	Number of trees = 400	Number of trees = 400

5. CONCLUSIONI

	Learning rate = [0.1, 0.5, 1] Algorithm = ["SAMME", "SAMME.R"]	Learning rate = 1 Algorithm = SAMME.R	Learning rate = 1 Algorithm = SAMME.R
Extreme Gradient Boosting	Learning rate = [0.05, 0.1, 0.2] Max depth = [4,6,8,10] Number of trees = [100, 200, 300] Sample rate = [0.7, 0.8, 0.9] Column sample rate per tree = [0.2, 0.5, 0.7, 0.9, 1.0]	Learning rate = Max depth = Number of trees = Sample rate = Column sample rate per tree =	Learning rate = Max depth = Number of trees = Sample rate = Column sample rate per tree =

Tabella A2: Range degli iperparametri e valori ottimali stimati per modello SDM100

Modello	Range iperparametri	Iperparametri ottimali(Aggregated)	Iperparametri ottimali (AllInstances)
Random Forest	Number of trees = [100, 200, 400, 800, 2000] Max depth = [10, 20, 80, 110, None] Min samples to split internal node = [2, 5, 10] Min sample to be leaf = [1, 2, 4] Bootstrap = [True, False]	Number of trees = Max depth = Min samples to split internal node = Min sample to be leaf = Bootstrap = True	Number of trees = Max depth = Min samples to split internal node = Min sample to be leaf = Bootstrap = True
Gradient Boosting	Learning rate = [0.05, 0.1, 0.2] Max depth = [5, 6, 7, 8, 9] Number of trees = [50, 100, 200] Sample rate= [0.7,0.8,0.9]	Learning rate = Max depth = Number of trees = Sample rate =	Learning rate = Max depth = Number of trees = Sample rate =
AdaBoost	Number of trees = [50, 100, 200, 400] Learning rate = [0.1, 0.5, 1] Algorithm = ["SAMME", "SAMME.R"]	Number of trees = 400 Learning rate = 0.5 Algorithm = SAMME.R	Number of trees = 400 Learning rate = 1 Algorithm = SAMME.R
Extreme	Learning rate = [0.05, 0.1, 0.2]	Learning rate =	Learning rate =

5. CONCLUSIONI

Gradient Boosting	Max depth = [4,6,8,10] Number of trees = [100, 200, 300] Sample rate = [0.7, 0.8, 0.9] Column sample rate per tree = [0.2, 0.5, 0.7, 0.9, 1.0]	Max depth = Number of trees = Sample rate = Column sample rate per tree =	Max depth = Number of trees = Sample rate = Column sample rate per tree =
-------------------	---	--	--

Tabella A3: Range degli iperparametri e valori ottimali stimati per modello CasHere

Modello	Range iperparametri	Iperparametri ottimali(Aggregated)	Iperparametri ottimali (AllInstances)
Random Forest	Number of trees = [100, 200, 400, 800, 2000] Max depth = [10, 20, 80, 110, None] Min samples to split internal node = [2, 5, 10] Min sample to be leaf = [1, 2, 4] Bootstrap = [True, False]	Number of trees = Max depth = Min samples to split internal node = Min sample to be leaf = Bootstrap = True	Number of trees = Max depth = Min samples to split internal node = Min sample to be leaf = Bootstrap = True
Gradient Boosting	Learning rate = [0.05, 0.1, 0.2] Max depth = [5, 6, 7, 8, 9] Number of trees = [50, 100, 200] Sample rate = [0.7, 0.8, 0.9]	Learning rate = Max depth = Number of trees = Sample rate =	Learning rate = Max depth = Number of trees = Sample rate =
AdaBoost	Number of trees = [50, 100, 200, 400] Learning rate = [0.1, 0.5, 1] Algorithm = ["SAMME", "SAMME.R"]	Number of trees = 400 Learning rate = 0.5 Algorithm = SAMME.R	Number of trees = 400 Learning rate = 0.5 Algorithm = SAMME.R
Extreme Gradient Boosting	Learning rate = [0.05, 0.1, 0.2] Max depth = [4,6,8,10] Number of trees = [100, 200, 300] Sample rate = [0.7, 0.8, 0.9]	Learning rate = Max depth = Number of trees = Sample rate =	Learning rate = Max depth = Number of trees = Sample rate =

5. CONCLUSIONI

	Column sample rate per tree = [0.2, 0.5, 0.7, 0.9, 1.0]	Column sample rate per tree =	Column sample rate per tree =
--	--	----------------------------------	----------------------------------

BIBLIOGRAFIA

- [1] *The Global Payments Report*. Worldpay from FIS, 2019, p. 9.
- [2] Philip Bruno, Olivier Denecker e Marc Niederkorn. *Global Payments Report 2019: Amid sustained growth, accelerating challenges demand bold actions*. McKinsey & Company, 2019.
- [3] *Study on the payment attitudes of consumers in the euro area*. European central bank, 2020. DOI: 10.2866/240571.
- [4] Alessio Botta, Philip Bruno e Jeff Galvin. *The 2021 McKinsey Global Payments Report*. McKinsey & Company, 2021.
- [5] Alessio Botta, Philip Bruno, Reet Chaudhuri, Marie-Claude Nadeau, Gustavo Tayar e Carlos Trascasa. *The 2020 McKinsey Global Payments Report*. McKinsey & Company, 2020.
- [6] Salih O. Duffuaa e A. Raouf. *Planning and Control of Maintenance Systems*. 2^a ed. Springer Cham, 2015. ISBN: 978-3-319-19803-3.
- [7] Srinivas Kumar Pinjala, Liliane Pintelon e Ann Vereecke. «An empirical investigation on the relationship between business and maintenance strategies». In: *International Journal of Production Economics* 104.1 (2006). Strategic Issues and Innovation in Production Economics, pp. 214–229. ISSN: 0925-5273. DOI: <https://doi.org/10.1016/j>.

- ijpe.2004.12.024. URL: <https://www.sciencedirect.com/science/article/pii/S0925527305000216>.
- [8] Douglas Thomas et al. «Advanced maintenance in manufacturing: costs and benefits». In: *Annual conference of the PHM society. September*. 2018, pp. 24–27.
- [9] Yongyi Ran, Xin Zhou, Pengfeng Lin, Yonggang Wen e Ruilong Deng. *A Survey of Predictive Maintenance: Systems, Purposes and Approaches*. 2019. DOI: 10.48550/ARXIV.1912.07383. URL: <https://arxiv.org/abs/1912.07383>.
- [10] Douglas S Thomas e Brian Weiss. «Maintenance Costs and Advanced Maintenance Techniques in Manufacturing Machinery: Survey and Analysis». In: *International Journal of Prognostics and Health Management* 12.1 (2021).
- [11] Erim Sezer, David Romero, Federico Guedea, Marco Macchi e Christos Emmanouilidis. «An industry 4.0-enabled low cost predictive maintenance approach for smes». In: *2018 IEEE International Conference on Engineering, Technology and Innovation (ICE/ITMC)*. IEEE. 2018, pp. 1–8.
- [12] Gian Antonio Susto, Andrea Schirru, Simone Pampuri, Seán McLoone e Alessandro Beghi. «Machine Learning for Predictive Maintenance: A Multiple Classifier Approach». In: *IEEE Transactions on Industrial Informatics* 11.3 (2015), pp. 812–820. DOI: 10.1109/TII.2014.2349359.
- [13] Benjamin S Blanchard, Dinesh C Verma e Elmer L Peterson. *Maintainability: a key to effective serviceability and maintenance management*. Vol. 13. John Wiley & Sons, 1995.
- [14] R. Keith Mobley. *An introduction to predictive maintenance*. 2^a ed. Butterworth-Heinemann, 2002. ISBN: 0-7506-7531-4.

- [15] Ernnie Illyani Basri, Izatul Hamimi Abdul Razak, Hasnida Ab-Samat e Shahrul Kamaruddin. «Preventive maintenance (PM) planning: a review». In: *Journal of Quality in Maintenance Engineering* (2017).
- [16] Rosmaini Ahmad e Shahrul Kamaruddin. «An overview of time-based and condition-based maintenance in industrial application». In: *Computers & Industrial Engineering* 63.1 (2012), pp. 135–149. ISSN: 0360-8352. DOI: <https://doi.org/10.1016/j.cie.2012.02.002>. URL: <https://www.sciencedirect.com/science/article/pii/S0360835212000484>.
- [17] John S Usher, Ahmed H Kamal e Wasim Hashmi Syed. «Cost optimal preventive maintenance and replacement scheduling». In: *IIE transactions* 30.12 (1998), pp. 1121–1128.
- [18] Ronald V Canfield. «Cost optimization of periodic preventive maintenance». In: *IEEE Transactions on Reliability* 35.1 (1986), pp. 78–81.
- [19] Gregory W. Vogl, Brian A. Weiss e Moneer Helu. «A review of diagnostic and prognostic capabilities and best practices for manufacturing». In: *Journal of Intelligent Manufacturing* 30.1 (2019), pp. 79–95. DOI: 10.1007/s10845-016-1228-8.
- [20] Canh Ly, Kwok Tom, Carl S. Byington, Romano Patrick e George J. Vachtsevanos. «Fault diagnosis and failure prognosis for engineering systems: A global perspective». In: *2009 IEEE International Conference on Automation Science and Engineering*. 2009, pp. 108–115. DOI: 10.1109/COASE.2009.5234094.
- [21] Basim Al-Najjar e Imad Alsyof. «Enhancing a company's profitability and competitiveness using integrated vibration-based maintenance: A case study». In: *European journal of operational research* 157.3 (2004), pp. 643–657.

- [22] M Carmen Carnero. «Selection of diagnostic techniques and instrumentation in a predictive maintenance program. A case study». In: *Decision support systems* 38.4 (2005), pp. 539–555.
- [23] Vanraj, Deepam Goyal, Abhineet Saini, S. S. Dhami e B. S. Pabla. «Intelligent predictive maintenance of dynamic systems using condition monitoring and signal processing techniques — A review». In: *2016 International Conference on Advances in Computing, Communication, Automation (ICACCA) (Spring)*. 2016, pp. 1–6. DOI: 10.1109/ICACCA.2016.7578870.
- [24] Zeki Murat Çınar, Abubakar Abdussalam Nuhu, Qasim Zeeshan, Orhan Korhan, Mohammed Asmael e Babak Safaei. «Machine Learning in Predictive Maintenance towards Sustainable Smart manufacturing in Industry 4.0». In: *Sustainability* 12.19 (2020). ISSN: 2071-1050. DOI: 10.3390/su12198211. URL: <https://www.mdpi.com/2071-1050/12/19/8211>.
- [25] Eldho Babu, Jebin Francis, Esther Thomas, Rahul Cherian e Sudarsana S Sunandhan. «Review on Various Signal Processing Techniques for Predictive Maintenance». In: *2022 2nd International Conference on Power Electronics IoT Applications in Renewable Energy and its Control (PARC)*. 2022, pp. 1–8. DOI: 10.1109/PARC52418.2022.9726618.
- [26] H. M. Hashemian. «State-of-the-Art Predictive Maintenance Techniques». In: *IEEE Transactions on Instrumentation and Measurement* 60.1 (2011), pp. 226–236. DOI: 10.1109/TIM.2010.2047662.
- [27] Gary E Newell. «Oil analysis cost-effective machine condition monitoring technique». In: *Industrial Lubrication and tribology* (1999).
- [28] Vignesh V Shanbhag, Thomas JJ Meyer, Leo W Caspers e Rune Schlanbusch. «Failure Monitoring and Predictive Maintenance of Hydraulic Cylinder—State-of-the-Art Review». In: *IEEE/ASME Transactions on Mechatronics* 26.6 (2021), pp. 3087–3103.

- [29] P. Aivaliotis, K. Georgoulas e G. Chryssolouris. «A RUL calculation approach based on physical-based simulation models for predictive maintenance». In: *2017 International Conference on Engineering, Technology and Innovation (ICE/ITMC)*. 2017, pp. 1243–1246. DOI: 10.1109/ICE.2017.8280022.
- [30] Athanasios Naskos e Anastasios Gounaris. «Efficiency assessment of event-based predictive maintenance in Industry 4.0.» In: *ICDM*. 2019, pp. 103–117.
- [31] Marina Paolanti, Luca Romeo, Andrea Felicetti, Adriano Mancini, Emanuele Frontoni e Jelena Loncarski. «Machine learning approach for predictive maintenance in industry 4.0». In: *2018 14th IEEE/ASME International Conference on Mechatronic and Embedded Systems and Applications (MESA)*. IEEE. 2018, pp. 1–6.
- [32] Pedro Cesar Lopes Gerum, Ayca Altay e Melike Baykal-Gürsoy. «Data-driven predictive maintenance scheduling policies for railways». In: *Transportation Research Part C: Emerging Technologies* 107 (2019), pp. 137–154. ISSN: 0968-090X. DOI: <https://doi.org/10.1016/j.trc.2019.07.020>. URL: <https://www.sciencedirect.com/science/article/pii/S0968090X18314918>.
- [33] Matteo Calabrese, Martin Cimmino, Francesca Fiume, Martina Manfrin, Luca Romeo, Silvia Ceccacci, Marina Paolanti, Giuseppe Toscano, Giovanni Ciandrini, Alberto Carrotta et al. «SOPHIA: An event-based IoT and machine learning architecture for predictive maintenance in industry 4.0». In: *Information* 11.4 (2020), p. 202.
- [34] Weiting Zhang, Dong Yang e Hongchao Wang. «Data-Driven Methods for Predictive Maintenance of Industrial Equipment: A Survey». In: *IEEE Systems Journal* 13.3 (2019), pp. 2213–2227. DOI: 10.1109/JSYST.2019.2905565.
- [35] Parivash Pirasteh, Slawomir Nowaczyk, Sepideh Pashami, Magnus Löwenadler, Klas Thunberg, Henrik Ydreskog e Peter Berck. «Interactive

- Feature Extraction for Diagnostic Trouble Codes in Predictive Maintenance: A Case Study from Automotive Domain». In: *Proceedings of the Workshop on Interactive Data Mining*. WIDM'19. Melbourne, VIC, Australia: Association for Computing Machinery, 2019. ISBN: 9781450362962. DOI: 10.1145/3304079.3310288. URL: <https://doi.org/10.1145/3304079.3310288>.
- [36] K.L. Butler. «An expert system based framework for an incipient failure detection and predictive maintenance system». In: *Proceedings of International Conference on Intelligent System Application to Power Systems*. 1996, pp. 321–326. DOI: 10.1109/ISAP.1996.501092.
- [37] J Wang, C Li, S Han, S Sarkar e X Zhou. «Predictive maintenance based on event-log analysis: A case study». In: *IBM Journal of Research and Development* 61.1 (2017), pp. 11–121.
- [38] Alexandros Bousdekis, Dimitris Apostolou e Gregoris Mentzas. «Predictive Maintenance in the 4th Industrial Revolution: Benefits, Business Opportunities, and Managerial Implications». In: *IEEE Engineering Management Review* 48.1 (2020), pp. 57–62. DOI: 10.1109/EMR.2019.2958037.
- [39] Leticia Decker, Daniel Leite, Luca Giommi e Daniele Bonacorsi. «Real-Time Anomaly Detection in Data Centers for Log-based Predictive Maintenance using an Evolving Fuzzy-Rule-Based Approach». In: *2020 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*. 2020, pp. 1–8. DOI: 10.1109/FUZZ48607.2020.9177762.
- [40] Ruben Sipos, Dmitriy Fradkin, Fabian Moerchen e Zhuang Wang. «Log-Based Predictive Maintenance». In: *KDD '14*. New York, New York, USA: Association for Computing Machinery, 2014, pp. 1867–1876. ISBN: 9781450329569. DOI: 10.1145/2623330.2623340. URL: <https://doi.org/10.1145/2623330.2623340>.
- [41] Clemens Gutschi, Nikolaus Furian, Josef Suschnigg, Dietmar Neubacher e Siegfried Voessner. «Log-based predictive maintenance in discrete parts

- manufacturing». In: *Procedia CIRP* 79 (2019). 12th CIRP Conference on Intelligent Computation in Manufacturing Engineering, 18-20 July 2018, Gulf of Naples, Italy, pp. 528–533. ISSN: 2212-8271. DOI: <https://doi.org/10.1016/j.procir.2019.02.098>. URL: <https://www.sciencedirect.com/science/article/pii/S221282711930215X>.
- [42] Antoine Guillaume, Christel Vrain e Elloumi Wael. *Predictive maintenance on event logs: Application on an ATM fleet*. 2020. DOI: 10.48550/ARXIV.2011.10996. URL: <https://arxiv.org/abs/2011.10996>.
- [43] Amol B Mahamuni et al. «Predictive Maintenance of Cash Dispenser Using a Cognitive Prioritization Model». In: *Tehnički glasnik* 15.1 (2021), pp. 156–161.
- [44] Panagiotis Korvesis, Stephane Besseau e Michalis Vazirgiannis. «Predictive maintenance in aviation: failure prediction from post-flight reports». In: *2018 IEEE 34th International Conference on Data Engineering (ICDE)*. IEEE. 2018, pp. 1414–1422.
- [45] Jaime G. Carbonell, Ryszard S. Michalski e Tom M. Mitchell. «1 - An Overview of Machine Learning». In: *Machine Learning*. A cura di Ryszard S. Michalski, Jaime G. Carbonell e Tom M. Mitchell. San Francisco (CA): Morgan Kaufmann, 1983, pp. 3–23. ISBN: 978-0-08-051054-5. DOI: <https://doi.org/10.1016/B978-0-08-051054-5.50005-4>. URL: <https://www.sciencedirect.com/science/article/pii/B9780080510545500054>.
- [46] Samaneh Aminikhanghahi e Diane J Cook. «A survey of methods for time series change point detection». In: *Knowledge and information systems* 51.2 (2017), pp. 339–367.
- [47] Ke Zhang, Jianwu Xu, Martin Renqiang Min, Guofei Jiang, Konstantinos Pelechrinis e Hui Zhang. «Automated IT system failure prediction: A deep learning approach». In: *2016 IEEE International Conference on Big Data (Big Data)*. IEEE. 2016, pp. 1291–1300.
-

- [48] KamalaKanta Mishra e Sachin Kumar Manjhi. «Failure prediction model for predictive maintenance». In: *2018 IEEE International Conference on Cloud Computing in Emerging Markets (CCEM)*. IEEE. 2018, pp. 72–75.
- [49] Luca Giommi, D Bonacorsi, T Diotallevi, S Rossi Tisbeni, L Rinaldi, L Morganti, A Falabella, E Ronchieri, A Ceccanti e B Martelli. «Towards predictive maintenance with machine learning at the INFN-CNAF computing centre». In: *Int. Symp. on Grids & Clouds (ISGC). Taipei: Proceedings of Science*. 2019, pp. 1–17.
- [50] Manuel Fernández-Delgado, Manisha Sanjay Sirsat, Eva Cernadas, Sadi Alawadi, Senén Barro e Manuel Febrero-Bande. «An extensive experimental survey of regression methods». In: *Neural Networks* 111 (2019), pp. 11–34.
- [51] Athanasios Naskos, Georgia Kougka, Theodoros Toliopoulos, Anastasios Gounaris, Cosmas Vamvalis e Daniel Caljouw. «Event-based predictive maintenance on top of sensor data in a real Industry 4.0 case study». In: *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer. 2019, pp. 345–356.
- [52] Thomas G. Dietterich, Richard H. Lathrop e Tomás Lozano-Pérez. «Solving the multiple instance problem with axis-parallel rectangles». In: *Artificial Intelligence* 89.1 (1997), pp. 31–71. ISSN: 0004-3702. DOI: [https://doi.org/10.1016/S0004-3702\(96\)00034-3](https://doi.org/10.1016/S0004-3702(96)00034-3). URL: <https://www.sciencedirect.com/science/article/pii/S0004370296000343>.
- [53] Marc-André Carbonneau, Veronika Cheplygina, Eric Granger e Ghyslain Gagnon. «Multiple instance learning: A survey of problem characteristics and applications». In: *Pattern Recognition* 77 (2018), pp. 329–353. ISSN: 0031-3203. DOI: <https://doi.org/10.1016/j.patcog.2017.10.009>. URL: <https://www.sciencedirect.com/science/article/pii/S0031320317304065>.
- [54] Samina Khalid, Tehmina Khalil e Shamila Nasreen. «A survey of feature selection and feature extraction techniques in machine learning». In:

- 2014 *Science and Information Conference*. 2014, pp. 372–378. DOI: 10.1109/SAI.2014.6918213.
- [55] Avrim L. Blum e Pat Langley. «Selection of relevant features and examples in machine learning». In: *Artificial Intelligence* 97.1 (1997). Relevance, pp. 245–271. ISSN: 0004-3702. DOI: [https://doi.org/10.1016/S0004-3702\(97\)00063-5](https://doi.org/10.1016/S0004-3702(97)00063-5). URL: <https://www.sciencedirect.com/science/article/pii/S0004370297000635>.
- [56] Andreas Janecek, Wilfried Gansterer, Michael Demel e Gerhard Ecker. «On the Relationship Between Feature Selection and Classification Accuracy». In: *Proceedings of the Workshop on New Challenges for Feature Selection in Data Mining and Knowledge Discovery at ECML/PKDD 2008*. A cura di Yvan Saeys, Huan Liu, Iñaki Inza, Louis Wehenkel e Yves Van de Pee. Vol. 4. Proceedings of Machine Learning Research. Antwerp, Belgium: PMLR, set. 2008, pp. 90–105. URL: <https://proceedings.mlr.press/v4/janecek08a.html>.
- [57] Zhuang Wang, Liang Lan e Slobodan Vucetic. «Mixture Model for Multiple Instance Regression and Applications in Remote Sensing». In: *IEEE Transactions on Geoscience and Remote Sensing* 50.6 (2012), pp. 2226–2237. DOI: 10.1109/TGRS.2011.2171691.
- [58] Soumya Ray e Mark Craven. «Supervised versus multiple instance learning: An empirical comparison». In: *Proceedings of the 22nd international conference on Machine learning*. 2005, pp. 697–704.
- [59] Petro Liashchynskyi e Pavlo Liashchynskyi. *Grid Search, Random Search, Genetic Algorithm: A Big Comparison for NAS*. 2019. DOI: 10.48550/ARXIV.1912.06059. URL: <https://arxiv.org/abs/1912.06059>.
- [60] Ronaldo C Prati, Gustavo EAPA Batista e Maria Carolina Monard. «Data mining with imbalanced class distributions: concepts and methods.» In: *IICAI*. 2009, pp. 359–376.

- [61] Nitesh V. Chawla, Nathalie Japkowicz e Aleksander Kotcz. «Editorial: Special Issue on Learning from Imbalanced Data Sets». In: *SIGKDD Explor. Newsl.* 6.1 (giu. 2004), pp. 1–6. ISSN: 1931-0145. DOI: 10.1145/1007730.1007733. URL: <https://doi.org/10.1145/1007730.1007733>.
- [62] Sotiris Kotsiantis, Dimitris Kanellopoulos, Panayiotis Pintelas et al. «Handling imbalanced datasets: A review». In: *GESTS international transactions on computer science and engineering* 30.1 (2006), pp. 25–36.
- [63] Vaishali Ganganwar. «An overview of classification algorithms for imbalanced datasets». In: *International Journal of Emerging Technology and Advanced Engineering* 2.4 (2012), pp. 42–47.
- [64] Leif E Peterson. «K-nearest neighbor». In: *Scholarpedia* 4.2 (2009), p. 1883.
- [65] Ivan Tomek. «Two modifications of CNN». In: *IEEE Trans. Systems, Man and Cybernetics* 6 (1976), pp. 769–772.
- [66] Gustavo E. A. P. A. Batista, Ronaldo C. Prati e Maria Carolina Monard. «A Study of the Behavior of Several Methods for Balancing Machine Learning Training Data». In: *SIGKDD Explor. Newsl.* 6.1 (giu. 2004), pp. 20–29. ISSN: 1931-0145. DOI: 10.1145/1007730.1007735. URL: <https://doi.org/10.1145/1007730.1007735>.
- [67] Dennis L. Wilson. «Asymptotic Properties of Nearest Neighbor Rules Using Edited Data». In: *IEEE Transactions on Systems, Man, and Cybernetics* SMC-2.3 (1972), pp. 408–421. DOI: 10.1109/TSMC.1972.4309137.
- [68] Bartosz Krawczyk. «Learning from imbalanced data: open challenges and future directions». In: *Progress in Artificial Intelligence* 5.4 (2016), pp. 221–232. DOI: 10.1007/s13748-016-0094-0. URL: <https://doi.org/10.1007/s13748-016-0094-0>.

BIBLIOGRAFIA

- [69] Mikel Galar, Alberto Fernandez, Edurne Barrenechea, Humberto Bustince e Francisco Herrera. «A Review on Ensembles for the Class Imbalance Problem: Bagging-, Boosting-, and Hybrid-Based Approaches». In: *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)* 42.4 (2012), pp. 463–484. DOI: 10.1109/TSMCC.2011.2161285.

ELENCO DELLE FIGURE

1	Esempi di dispositivi per il deposito di CIMA	7
2	Esempi di dispositivi per il ricircolo di CIMA	8
3	Esempi di dispositivi da front-office di CIMA	9
4	CasHere, dispositivo CIMA per il self-service assistito	10
5	Schema delle funzionalità di C-Link	11
6	Esempio di schermata di C-Service con un'analisi prodotta . . .	12
7	Schema di funzionamento della tecnica di finestra mobile	22

ELENCO DELLE TABELLE

A1	Range degli iperparametri e valori ottimali stimati per modello Inlane	31
A2	Range degli iperparametri e valori ottimali stimati per modello SDM100	32
A3	Range degli iperparametri e valori ottimali stimati per modello CasHere	33

ELENCO DEGLI ACRONIMI

ATM Automated Teller Machine

TCR Teller Cash Recycler

MO Modena

AWS Amazon Web Services

SaaS Software as a Service

USB Universal Serial Bus

R2F Run to failure

PvM Manutenzione preventiva

PdM Manutenzione predittiva

USA United States of America

RUL Remaining Useful Lifetime

CM Manutenzione correttiva

POS Point of sale

P2P Person to person

ELENCO DELLE TABELLE

CPD	Change-point detection
MIL	Multiple instance learning
PW	Prediction window
OW	Observation window
BW	Buffer window
MTBE	Mean Time Between Exception
FS	Feature Selection
SMOTE	Synthetic Minority Over-sampling Technique
RUS	Random Under-Sampling
ENN	Edited Nearest Neighbour