



POLITECNICO DI TORINO

Corso di Laurea in Ingegneria Informatica indirizzo Software

Tesi di Laurea Magistrale

Industria 4.0: architettura IoT e manutenzione predittiva

Relatore

prof. Tania Cerquitelli

Candidato

Stefano MONTI

Supervisore Aziendale

ing. Andrea Venditti

ANNO ACCADEMICO 2018-2019

Sommario

L'inizio dell'industria 4.0 avvenne nel 2011 con la presentazione del piano politico tedesco **Industrie 4.0** per rinnovare l'industria manifatturiera. Solo poche nazioni, come USA e Giappone, già praticavano politiche simili senza riscuotere grande interesse nella ricerca scientifica ma, con il tempo, sempre più paesi hanno iniziato ad adottare piani per innovare il proprio settore industriale. Al giorno d'oggi, tutti gli stati del primo mondo hanno delle politiche economiche a riguardo. Questa trasformazione è stata definita dalla comunità scientifica come una rivoluzione industriale vera e propria, la quarta per esattezza, per via del profondo cambiamento sulla concezione dell'impianto industriale rispetto al passato.

I principi della trasformazione sono dovuti, in maggior sostanza, alla digitalizzazione e connessione dei dispositivi in modo da ottenere molti benefici: monitoraggio, autonomia, realtà aumentata, analisi, predizioni, simulazioni, gestione degli eventi in tempo reale, collaborazione tra macchine e uomo-macchina ottenendo, in maniera più generica, un'ottimizzazione del prodotto finale. Ciò comporta migliori prestazioni dal punto di vista della velocità, qualità, sicurezza e lato economico. Anche il lavoratore ottiene vari benefici legati all'ausilio di diversi strumenti per la gestione e monitoraggio dell'impianto semplificando il proprio operato e, allo stesso tempo, ottimizzando il risultato finale.

In particolare, si vuole mostrare l'implementazione del lato analisi nell'industria 4.0 tramite l'applicazione della prognostica e della manutenzione predittiva. L'apprendimento automatico applicato alle macchine può ricavare una serie di predizioni che aiutano i lavoratori nel prendere decisioni per la gestione dell'impianto sia rendono *intelligenti* gli apparecchi stessi in maniera tale da ottimizzare il proprio lavoro e diventare il più autonomi possibili. Lo scenario ideale per l'industria 4.0 è la totale autonomia delle macchine; quest'ultime riescono a capire il proprio andamento e, in base a ciò, prendono decisioni per migliorare la produzione, il tutto senza nessun intervento umano.

La seguente tesi ha quindi lo scopo di mostrare il perché questi argomenti stanno diventando sempre più importanti, ricercati e nominati. Nella prima fase si presentano le definizioni, benefici e struttura dei temi accennati sopra. Infine si sperimentano due applicazioni riguardo l'industria 4.0 e la manutenzione predittiva per mostrare una proiezione pratica dei concetti trattati in teoria.

Indice

1	Prefazione	1
1.1	Industria 4.0	1
1.2	Prognostics and Health Management (PHM)	1
1.3	Tesi	2
1.3.1	Obiettivi della tesi	2
1.3.2	Organizzazione della tesi	2
1.4	Specifiche	3
1.5	Strumenti utilizzati	3
1.5.1	IBM Cloud	5
1.5.2	IBM Watson IoT Platform	5
1.5.3	Node-RED	5
1.5.4	Cloudant NoSQL Database	5
1.5.5	Angular 5 o AngularJS	6
1.5.6	Node.js	7
1.5.7	IBM Watson Studio	7
1.5.8	Jupyter Notebook	8
2	Stato dell'arte	9
2.1	Industria 4.0	9
2.1.1	Rivoluzioni industriali: cause e conseguenze	9
2.1.2	Piano Industrie 4.0	13
2.1.3	Principi	14
2.1.4	Componenti	15
2.1.5	Sistemi ciberfisici	16
2.1.6	Internet delle cose (IoT)	19
2.1.7	Big data	19
2.1.8	Benefici	21
2.1.9	Sfide da affrontare e orientamento per il futuro	22
2.2	Il mercato odierno dell'industria 4.0	23

2.2.1	Iniziative nell'Unione europea	25
2.2.2	Il caso italiano	27
2.3	Soluzione architetture	31
2.4	Prognostic and health management	35
2.4.1	Manutenzioni	36
2.4.2	Diagnostica e prognostica	38
2.4.3	Architettura PHM	41
2.4.4	PHM e big data	45
2.4.5	PHM applicata ai sistemi ciberfisici	46
2.4.6	Benefici della PHM	47
2.4.7	Tipico piano di lavoro	49
2.4.8	Sfide ed orientamento per il futuro per la PHM	50
2.5	Stato odierno della PHM	54
2.5.1	Applicazioni e casi d'uso	58
3	Manutenzione predittiva	61
3.1	Descrizione del problema	61
3.1.1	Classificazione e regressione	62
3.1.2	Macchina ad iniezione: IntElect 160/520-680 Smart	63
3.1.3	Raccolta e struttura dati	64
3.2	Elaborazione e analisi dati	67
3.3	Quesito 1 - La macchina avrà un fallimento nei prossimi N secondi/ore/giorni/cicli macchina?	73
3.3.1	Quesito 1 - Problema dei dati sbilanciati	73
3.3.2	Quesito 1 - Creazione della variabile target <i>Target_N_Cycles</i>	75
3.3.3	Quesito 1 - Normalizzazione	76
3.3.4	Quesito 1 - Filtraggio	77
3.3.5	Quesito 1 - Principal Component Analysis (PCA)	77
3.3.6	Quesito 1 - Sottocampionamento e sovracampionamento	78
3.3.7	Quesito 1 - Creazione nuove variabili	81
3.3.8	Quesito 1 - Selezione misure	83
3.3.9	Quesito 1 - Modelli di apprendimento	84
3.3.10	Quesito 1 - Soluzione finale	85
3.4	Quesito 2 - La macchina si sta comportando in modo anomalo?	86
3.4.1	Quesito 2 - Creazione della variabile target <i>Target_multi_class</i>	87
3.4.2	Quesito 2 - Normalizzazione	87
3.4.3	Quesito 2 - Filtraggio	88
3.4.4	Quesito 2 - Principal Component Analysis (PCA)	88

3.4.5	Quesito 2 - Sottocampionamento e sovracampionamento	89
3.4.6	Quesito 2 - Creazione nuove variabili	91
3.4.7	Quesito 2 - Selezione misure	91
3.4.8	Quesito 2 - Modelli di apprendimento	91
3.4.9	Quesito 2 - Soluzione finale	92
3.5	Quesito 3 - Quanto tempo o cicli macchina mancano prima che si abbia il prossimo fallimento?	94
3.5.1	Quesito 3 - Creazione della variabile target <i>rul</i>	95
3.5.2	Quesito 3 - Normalizzazione	96
3.5.3	Quesito 3 - Filtraggio	96
3.5.4	Quesito 3 - PCA	97
3.5.5	Quesito 3 - Creazione nuove variabili	98
3.5.6	Quesito 3 - Modelli di apprendimento e selezione misure	98
3.5.7	Quesito 3 - Soluzione finale	98
3.6	Conclusioni e sviluppi futuri	100
4	Implementazione soluzione	101
4.1	IBM Cloud	102
4.2	Declinazione architettura	102
4.3	Connessione impianto	103
4.4	Comunicazione e registrazione macchina - server IBM Watson IoT Platform	104
4.5	Elaborazione dei dati ricevuti - server Node-RED	105
4.6	Storage - Database Cloudant	106
4.7	Lato server - server Plant Control Room	107
4.8	Lato client - client Plant Control Room	109
4.8.1	Dashboard	109
4.8.2	Pagina di dettaglio dell'apparecchiatura	111
4.8.3	Pagina di registrazione della macchina	112
4.9	Monitoraggio e analisi dei pattern anomali real-time	114
5	Conclusioni	116
5.1	Risultati ottenuti	116
5.2	Sviluppi futuri	116
5.3	Conoscenze acquisite	118
	Bibliografia	120

Capitolo 1

Prefazione

1.1 Industria 4.0

L'industria 4.0 deriva da un'iniziativa politica tedesca iniziata nel 2011 chiamata **Industrie 4.0** [1] [2]. Questa ha l'intento di cambiare il metodo di lavoro negli impianti industriali, in particolare manifatturieri, grazie all'applicazione di nuove tecnologie, alcune già diffuse in altri settori, in modo da ottenere migliori prestazioni. Il cambiamento non solo dei mezzi utilizzati ma del *modus operandi* ha portato ad identificare in esso la nascita della quarta rivoluzione industriale. Le trasformazioni che vengono apportate sono legate principalmente alla digitalizzazione e connessione degli attori in gioco. Dal primo fattore deriva principalmente il beneficio di monitorare l'intera catena di montaggio, è così possibile ottimizzare ogni passaggio dall'inizio al risultato finale consci del comportamento di qualsiasi dispositivo. La connessione fornisce a tutti gli strumenti una conoscenza globale del lavoro e una collaborazione che prima non erano possibili perciò, nel passato, questo elemento doveva essere gestito da un ente terzo ovvero un operaio. L'unione tra digitalizzazione e connessione permette di ottenere un sistema autogestito e ottimizzato, aperto verso big data, analisi ed apprendimento automatico, dove l'umano è rilegato a competenze meno pratiche ma più tecniche con il compito di prendere decisioni che per la macchina sono troppo *difficili*. Allo stesso tempo i lavoratori possono sfruttare diversi strumenti che lo aiutino a gestire il proprio lavoro al meglio, per esempio applicazioni software per la visualizzazione del comportamento dei macchinari. Naturalmente l'industria 4.0 è un sistema molto più complesso di quanto appena descritto e offre benefici e potenzialità in continua espansione, per una definizione completa e dettagliata si rimanda a Industria 4.0.

1.2 Prognostics and Health Management (PHM)

La **Prognostics and Health Management** è la disciplina che si occupa del comportamento e stato di un qualsiasi dispositivo [103]. Analizzando l'andamento di un apparecchio, è possibile migliorare, monitorare e gestire la sua affidabilità, disponibilità, sicurezza e manutenzione. Tutto ciò, applicato a livello industriale, comporta un miglioramento delle prestazioni totali dell'impianto ottenendo, di conseguenza, un maggior sfruttamento economico delle risorse disponibili. Grazie all'utilizzo di diagnosi e prognosi, è possibile studiare la storia passata di un determinato oggetto e, in base ad essa, effettuare delle predizioni nel futuro grazie alle quali possono essere prese decisioni molto migliori proprio per il fatto di avere una visione su ciò che accadrà. Un esempio è la manutenzione predittiva, attraverso la sua applicazione si può predire il prossimo fallimento di una macchina in modo da ottimizzare le manutenzioni e ottenere i benefici elencati precedentemente. Ciò compone, come si vedrà meglio in seguito, la parte più avanzata dell'industria 4.0 ovvero un impianto che è totalmente autogestito ed impara dai propri sbagli, immagazzina sempre più dati (e fallimenti) generando uno storico maggiore a cui attingere e quindi ottimizzarsi. Infatti, per il momento, è

solitamente l'uomo a dover interpretare e sfruttare l'analisi ottenuta dalla PHM ma si sta cercando di fare in modo che le macchine prendano autonomamente decisioni sempre più complesse in modo da non necessitare aiuti esterni. Maggiori dettagli sono trattati nella sezione Prognostic and health management.

1.3 Tesi

La seguente tesi rappresenta il lavoro e la ricerca svolti presso **Blue Reply** grazie al supporto di Fabio Raimondi e Vincenzo Stola. Essa è stata svolta con il relatore Tania Cerquitelli ed è la conclusione del percorso di laurea magistrale in **Ingegneria informatica indirizzo Software** presso il **Politecnico di Torino**. Gli argomenti trattati sono principalmente due: **industria 4.0** e **Prognostics and Health Management (PHM)**.

1.3.1 Obiettivi della tesi

La tesi si impone di istruire il lettore riguardo i temi **industria 4.0** e **PHM**, in particolare:

- **Definizione, evoluzione e stato odierno.** In questo modo si vuole offrire una panoramica generale sugli argomenti trattati, mostrare il perché questi elementi sono importanti e le loro grandi potenzialità giustificando così il lavoro di tesi.
- **Architettura e classico modo di operare.** Questa fase dimostra l'unione tra la teoria e la pratica. Vuole mostrare la struttura generale delle applicazioni basate sull'industria 4.0 e manutenzione predittiva. Da ciò si possono apprendere i punti chiave e gli elementi in gioco di un qualsiasi programma in tale ambito ottenendo una proiezione pratica degli argomenti prima trattati solo ad alto livello.
- **Applicazione software.** Ha l'obiettivo di presentare i risultati pratici dei due temi trattati tramite sviluppo software. Ciò comporta la presentazione del lavoro svolto, la sua organizzazione e il modus operandi usato per arrivare al prodotto finale. Il lettore può così avere un esempio concreto di tutti i benefici e potenzialità esibiti nel corso della tesi.

1.3.2 Organizzazione della tesi

Questo capitolo funge da introduzione. Presenta in modo molto sintetico la struttura e le motivazioni della seguente tesi. Vengono presentate le specifiche iniziali, gli strumenti utilizzati e perché questi sono stati scelti rispetto ad altri.

Si continua nel secondo capitolo parlando della definizione sull'industria 4.0 e PHM. Questa prima parte vuole descrivere gli argomenti che poi verranno trattati nella pratica, la loro evoluzione fino ad arrivare al loro stato odierno e gli elementi di cui sono composti in modo da avere una visione globale su essi. In particolare si vuole mostrare la loro importanza per il futuro. Si parte da una definizione sull'argomento, si continua parlando della situazione odierna per conoscere lo stato dell'arte a cui si è arrivati ed infine si delinea un'architettura ad alto livello sia per l'industria 4.0 sia per la PHM presentando l'applicazione della teoria nella pratica.

Il terzo capitolo descrive il lavoro svolto riguardante la manutenzione predittiva. Vengono descritti i procedimenti percorsi per arrivare alla soluzione finale descrivendo l'acquisizione dei dati, la loro analisi e i modelli predittivi utilizzati. Si analizzano i pro e i contro dei vari modus operandi intrapresi limando pian piano il risultato finale. Infine si presenta il modus operandi definitivo per il lavoro svolto.

Nel quarto capitolo viene presentata un'applicazione per l'acquisizione e la visualizzazione di dati provenienti da una macchina industriale. Viene presentata la sua architettura e tutti i settori

da cui è composta partendo dall'impianto fino ad arrivare all'interfaccia grafica. In seguito si propone una soluzione per integrare la parte di manutenzione predittiva descritta nel capitolo precedente con la struttura appena presentata.

Il quinto ed ultimo capitolo riassume i risultati ottenuti durante tutto il lavoro di tesi. Si avanzano commenti su come il prodotto ottenuto potrebbe venir ampliato e migliorato in futuro e si traggono le conclusioni.

1.4 Specifiche

Il punto principale per lo sviluppo della parte pratica, trattata in Manutenzione predittiva e Implementazione soluzione, è l'utilità effettiva da parte dei lavoratori. Si vuole quindi fornire, secondo i principi dell'industria 4.0, uno strumento che aiuti gli operai di un'azienda nel loro lavoro semplificando il loro operato e, allo stesso tempo, migliorandolo cosicché da generare benefici all'intera azienda. Non è richiesto lo sviluppo di un'applicazione per ambito commerciale ma si punta ad ottenere una demo, lo scopo è mostrare il funzionamento e le funzionalità di un prodotto in quest'ambito evidenziando i frutti dell'implementazione dell'industria 4.0. Il programma dovrà comunque funzionare correttamente ma non sono richiesti tanti dettagli e funzionalità che la renderebbero completa e pronta all'utilizzo effettivo, sono necessari solo i servizi principali. Siccome l'azienda con cui è stata svolta la tesi non dispone di hardware per implementare l'architettura necessaria, è necessario sfruttare un servizio **Platform as a service (PaaS)** che fornisca le macchine necessarie, bisogna così preparare solo il software necessario senza preoccuparsi del lato server.

Lato Implementazione soluzione, si vuole sviluppare un'architettura che permetta di monitorare il sistema industriale, in particolare è necessaria un'applicazione che visualizzi l'andamento dello stabilimento e delle singole macchine. Quest'ultima deve essere accessibile via web in maniera tale che ogni operaio possa accedervi quando e dove vuole e l'interfaccia grafica deve facilmente adattarsi a smartphone e tablet, così l'utilizzatore non deve portarsi costantemente dietro un computer il quale risulterebbe pesante ed ingombrante. L'applicazione deve anche gestire il fatto che le macchine sono di tipo e modello diversi perciò i dati che esse sfruttano ed inviano cambiano da dispositivo a dispositivo. Per questo motivo, il programma deve essere flessibile nell'accettare e gestire qualsiasi tipo e quantità di dati in modo da non avere problemi di compatibilità. Le macchine prima di poter essere visualizzate devono venir registrate tramite l'applicazione.

Per la sezione Manutenzione predittiva, bisogna ottenere delle predizioni riguardo il tempo che intercorre tra il momento in cui viene svolta l'analisi e il prossimo fallimento della macchina. Naturalmente è preferibile la maggiore precisione possibile. I dati disponibili provengono totalmente da un'unico apparecchio, una macchina ad iniezione, quindi il lavoro di analisi verrà svolto su di essa. Il caso ideale sarebbe sviluppare un software completo che lavori in tempo reale, funzioni per ogni macchina e migliori il modello con il passare del tempo (e dei fallimenti) però ciò comporterebbe una notevole sfida. Bisogna infine trovare un modo per collegare la parte appena descritta con l'architettura completa sull'industria 4.0 in modo da avere un'entità unica.

1.5 Strumenti utilizzati

Siccome la tesi è basata sull'industria 4.0 e, come descritto in Soluzione architetture, le applicazioni a riguardo hanno molti attori in scena (principalmente sensori, database, applicazione per l'apprendimento automatico, lato client, server per le richieste utente) la loro gestione può essere molto difficile e complicata. Infatti bisognerebbe gestire comunicazione, manutenzione, sicurezza, accessi, velocità di risposta, eccetera di tutti i server necessari. Se non si avessero delle macchine già configurate ed attive, sarebbe necessario spendere molto tempo per configurare tutte le funzioni necessarie solo per avere l'architettura generale. Per questo motivo si è deciso di adottare **IBM Cloud**

che fornisce già la struttura appena indicata lasciando agli sviluppatori solo l'implementazione software.

In realtà esistono anche altri framework simili a IBM Cloud, quest'ultimo è stato scelto a causa della posizione della sua azienda sviluppatrice, IBM, nel mondo dell'industria 4.0 e Internet delle cose (IoT) mostrandosi leader nel settore nell'anno 2015 come raffigurato in figura 1.1. Bisogna però specificare che le aziende trattate nel grafico non sono leader a 360 gradi dell'argomento IoT: Intel è la migliore nel campo dei semiconduttori, IBM e Microsoft nell'ambito piattaforma e analisi, Cisco per la connettività. Riguardo la manutenzione predittiva, IBM, nel 2017, si presenta al primo posto paragonata alle sue concorrenti, figura 1.2. Oltretutto l'azienda con cui è stata svolta la tesi, Blue Reply, possedeva già un account premium per IBM Cloud ed era già conosciuto ed utilizzato da vari suoi dipendenti.

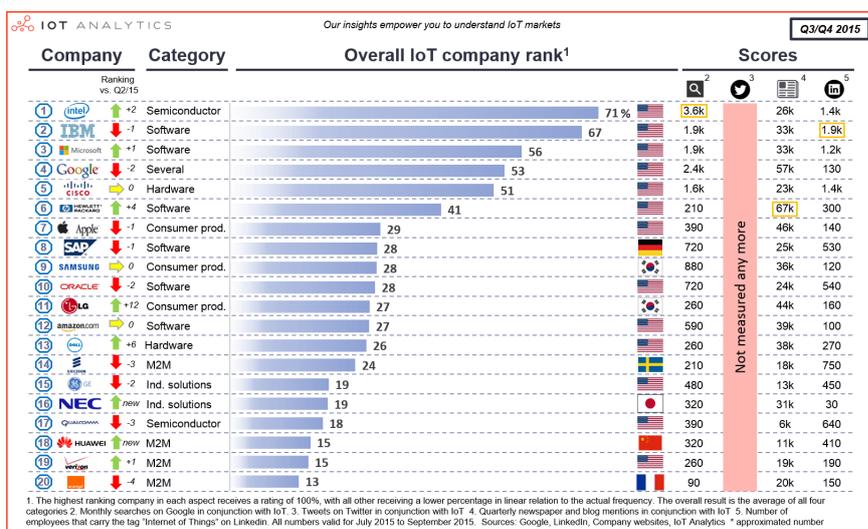


Figura 1.1. Classifica aziende in base allo sviluppo Internet delle cose (fine 2015) [4]

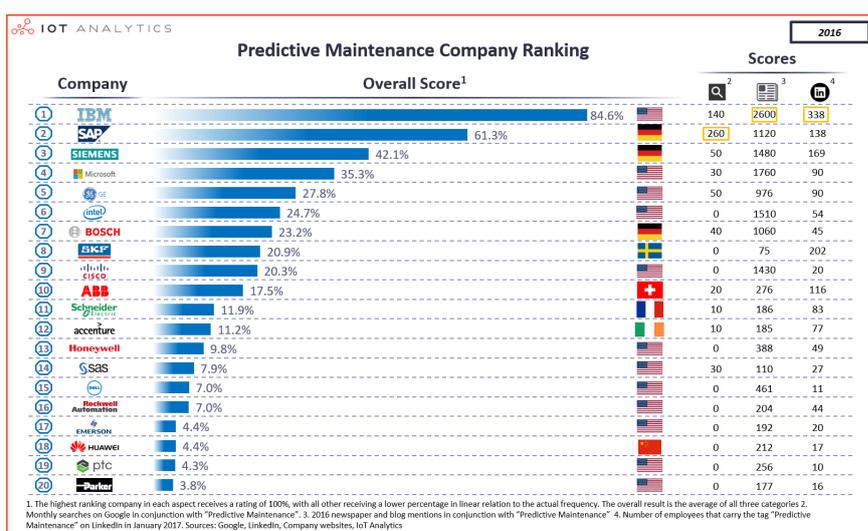


Figura 1.2. Classifica aziende in base alla manutenzione predittiva (2017) [5]

1.5.1 IBM Cloud

IBM Cloud (precedentemente chiamato IBM Bluemix) [6] è la combinazione tra **Platform as a service** (PaaS) e **Infrastructure as a Service** (IaaS). Il primo elemento indica una piattaforma di elaborazione che permette di sviluppare, testare ed implementare applicazioni senza la gestione e acquisto dell'hardware e software di base [7]. La seconda indica che le risorse a cui si accede sono virtualizzate e comporta benefici simili alla struttura precedente [8].

Il maggiore beneficio di IBM Cloud è quindi non dover gestire l'hardware per implementare una soluzione che necessita l'utilizzo di uno o più server. Oltretutto le macchine possono tranquillamente comunicare tra loro poiché configurate già in modo adeguato. La piattaforma fornisce molti servizi quali database, apprendimento automatico, immagazzinamento documenti, programmi per la sicurezza, macchine virtuali, **blockchain**, supporto per **Internet delle cose**, aiuto per applicazioni mobile eccetera [9]. Tutti gli elementi utilizzati per la parte pratica della tesi sono servizi sviluppati o compatibili con IBM Cloud.

1.5.2 IBM Watson IoT Platform

Watson IoT Platform è un servizio raggiungibile via web per la gestione e l'accesso a dispositivi **Internet delle cose** [12] [13]. Il suo punto di forza è la comunicazione sia con componenti IoT, tramite MQTT o HTTP, sia con applicazioni software, per mezzo di API. Può anche essere utilizzato per il salvataggio di dati e analisi legato ai dispositivi.

Per descrivere l'utilizzo che si è fatto bisogna prima spiegare che, la maggior parte dei sensori e dispositivi IoT, sfruttano il protocollo MQTT per comunicare con altre macchine salvando il messaggio in un server centrale detto broker il quale manipolerà e/o manderà questi dati ad un altro dispositivo [10]. Per l'applicazione descritta, il server **Watson IoT Platform** garantisce la ricezione di dati da parte delle macchine e, se la macchina è stata precedentemente registrata, inoltra il messaggio ricevuto al server **Node-RED**. I dispositivi, per poter attivare la comunicazione, devono essere registrati dagli operai tramite il lato client in modo da ottenere un identificativo univoco da associare ad ognuno di essi. Ciò comporta due benefici: solo i macchinari effettivamente registrati comunicano con l'architettura 4.0 realizzata, scartando tutti gli altri messaggi ricevuti, e i dati sono univocamente associati ad un apparecchio. Tutti gli identificativi ricevuti vengono salvati nel database **Cloudant**.

1.5.3 Node-RED

Node-RED è uno strumento per associare tra loro dispositivi **Internet delle cose**, API e servizi online [11] [10]. Si basa sul linguaggio **Javascript** tramite il framework **Node.js**. Fornisce un editor accessibile via web e permette di creare un flusso composto da blocchi dove ognuno di questi serve per manipolare i dati ricevuti come input, grazie ad un'interfaccia grafica intuitiva il lavoro è in buona parte semplificato.

L'applicazione **Node-RED** riceve i dati delle macchine da **Watson IoT Platform** e questi vengono manipolati ed inviati al database a seconda del loro contenuto. Per esempio è possibile generare degli allarmi se le misure ricevute sono sopra una certa soglia massima. Si gestiscono così gli eventi dei dispositivi (nel nostro caso, invio delle misurazioni effettuate in un ciclo macchina) in modo asincrono ovvero appena questi vengono ricevuti. Oltretutto sfrutta **Node.js** i quali benefici sono descritti nella sezione **Node.js**. **Node-RED** è compatibile con IBM Cloud tramite il servizio **Node-RED Starter**.

1.5.4 Cloudant NoSQL Database

Cloudant è un database NoSQL dove i dati vengono salvati come documenti **JSON** [14] [15]. È un prodotto IBM open-source basato sul progetto **Apache CouchDB**. Essendo sviluppato per IBM

Cloud, è di tipo **Database as a Service (DBaaS)** ovvero è un servizio cloud che permette di accedere ad una base di dati senza dover configurare e gestire hardware[16].

È stato utilizzato lato database per salvare i dati dei sensori, gli eventi di allarme e manutenzioni, informazioni delle macchine eccetera. È stato scelto proprio per il fatto di essere NoSQL, in particolare per tre benefici: la flessibilità, la mole di dati e perché i **big data** forniscono vari benefici per il lato analisi (per approfondire leggere PHM e big data). Il primo motivo è dovuto al fatto che non si conoscono a prescindere il tipo e la quantità dei dati da ricevere, una macchina può avere un numero di sensori variabile e questi possono appartenere a qualsiasi categoria. Perciò non è possibile utilizzare una base di dati relazionale dove le tabelle sono prestabilite. Il secondo motivo è dovuto al fatto che la base di dati dovrà contenere una grande quantità di dati visto che i sensori effettuano misurazioni in modo periodico ovvero ogni ciclo macchina (nelle macchine che il testista ha avuto modo di analizzare la durata di un ciclo macchina in stato normale è di circa 20 secondi).

1.5.5 Angular 5 o AngularJS

AngularJS è un framework open-source basato su **Javascript** per la scrittura di applicazioni web lato client [17] [18]. È supportato da **Google**. Fu rilasciato nel 2010, **Angular 5** era l'ultima versione durante lo svolgimento del lavoro ed è disponibile dal 2017. Questo ha avuto successo grazie al suo paradigma **model-view-controller (MVC)** e alla modularità causata dalla separazione dei lavori tra diversi componenti. Essendo un framework, predispone in modo intelligente l'applicazione secondo un determinato approccio, sfruttando al meglio **HTML** e **Javascript**.

Angular 5 è stato utilizzato per l'implementazione lato client dell'applicazione web dovuta alla visualizzazione dei dati e trattata in Implementazione soluzione. Principalmente, è stato scelto a causa della sua costante diffusione, nella figura 1.3 si può notare l'andamento delle ricerche **Google** tra i più famosi framework per sviluppo web. Tramite un sondaggio di **Stack overflow** del 2017 condotto su 64.000 sviluppatori, figura 1.4, è possibile notare che **Angular 5** e **Node.js** (trattato sotto) sono i framework più utilizzati. Quindi si è preferito seguire la tendenza in modo da rimanere aggiornati sui linguaggi di programmazione che sembrano avere potenzialità per il futuro.

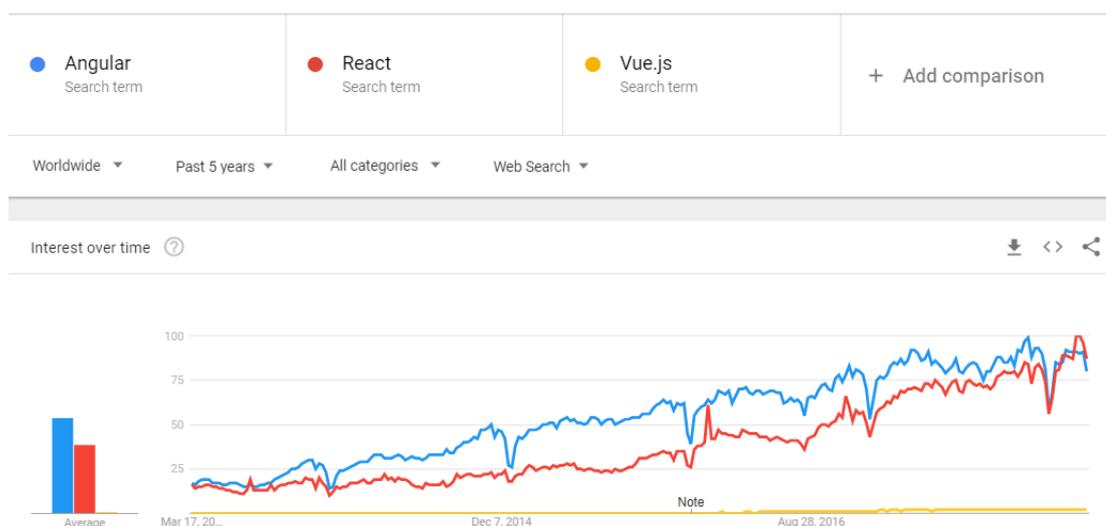


Figura 1.3. Ricerche **Google** sui principali framework web [19]

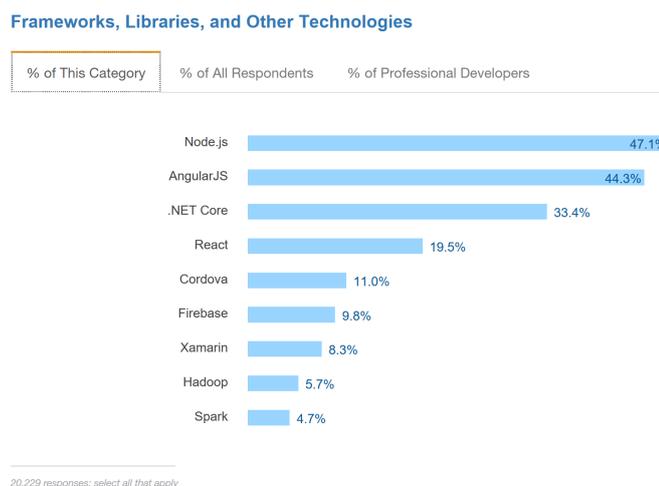


Figura 1.4. Percentuale dei framework, librerie ed altre tecnologie usate dagli sviluppatori nel 2017 [20]

1.5.6 Node.js

`Node.js` è un framework open-source basato su `Javascript` usato principalmente per la programmazione lato server [21] [22]. La sua prima versione fu rilasciata nel 2009. È diventato importante grazie alla caratteristica di avere un'architettura basata sugli eventi di tipo asincrono. Usando `PHP` o `ASP` il server, quando riceve una richiesta, non può iniziare una nuova operazione finché non ha finito la precedente poiché lavora con un comportamento sincrono; usando `Node.js` il server può ricevere e gestire più richieste in parallelo ottenendo così maggiori prestazioni [23].

Questo è stato usato per il lato server dell'applicazione web che verrà discussa in Implementazione soluzione. È stato scelto per due motivi: perché sta venendo sempre più utilizzato come dimostra la figura 1.4 e perché si integra bene con `Angular 5`. Infatti sia `Node.js` che `Angular JS` fanno parte dello stesso gruppo di framework `MEAN` [24], tutti i suoi componenti sono open-source, basati su `Javascript` e permettono la programmazione di applicazioni web.

1.5.7 IBM Watson Studio

`Watson Studio` fornisce una piattaforma con vari strumenti per gestire, visualizzare ed analizzare dati [25]. In particolare, permette di applicare algoritmi di apprendimento automatico. Fornisce anche vari tipi di interfaccia grafica per semplificare queste operazioni. Sono disponibili anche alcuni strumenti di apprendimento automatico già preimpostati quali riconoscimento visivo, trasformazione di voce in testo e viceversa. Gli unici servizi utilizzati sono stati `Machine learning` e `Jupyter Notebook` per l'applicazione sulla manutenzione predittiva.

Questo è stato scelto sia perché fornisce vari strumenti per l'apprendimento automatico compatibili con `IBM Cloud` sia per il supporto con `Jupyter Notebook` i quali benefici sono trattati in `Jupyter Notebook`. In realtà non è stato molto sfruttato perché è stato usato `Jupyter Notebook` direttamente in locale. Ciò si è reso necessario perché `IBM Watson`, non avendo un account premium, lo si può utilizzare solo per un certo periodo di tempo, troppo corto rispetto a quello richiesto per la fase di sperimentazione. Però essendo i due prodotti compatibili, il risultato ottenuto in locale può essere facilmente caricato su `Watson Studio` integrando il lavoro con la rimanente architettura sull'industria 4.0.

1.5.8 Jupyter Notebook

Jupyter Notebook (prima **IPython Notebooks**) è un ambiente di sviluppo open-source accessibile tramite web [26] [27]. Questo permette di scrivere e avviare parti di codice in modo dinamico e, cosa più importante, ha accesso a molte librerie per l'apprendimento automatico, visualizzazione e gestione dati soprattutto grazie al supporto **Python**.

È stato utilizzato per la sperimentazione riguardo la manutenzione predittiva quindi lato apprendimento automatico nell'architettura industria 4.0. In particolare è stato scelto perché ha molta documentazione, librerie e supporto da parte dell'utenza per questo ambito. Cercando sul web, **Jupyter Notebook** viene spesso consigliato tra i migliori ambienti di sviluppo (IDE) per **Data Science** ma le opinioni sono spesso discordanti, non sembra esserci ad oggi un IDE veramente superiore in qualità rispetto alla concorrenza. Oltretutto è compatibile con **IBM Watson Studio**.

Capitolo 2

Stato dell'arte

2.1 Industria 4.0

L'industria 4.0 (I40) [28] è la trasformazione digitale delle industrie, in particolare nell'ambito manifatturiero, che si basa sulla raccolta e utilizzo delle informazioni legate a persone, processi, servizi, sistemi e macchinari in modo da realizzare un ecosistema *intelligente* (*smart*) generando innovazione e collaborazione. In chiave più generica definisce l'evoluzione in ambito industriale avuta dal 2011 a oggi verso una maggiore informatizzazione e utilizzo di dispositivi collegati tra loro. In realtà non esiste una definizione canonica. Questa è basata sull'uso di tecnologie dette sistemi ciberfisici, o cyber-physical systems (CPS). Dei protagonisti principali della sua attuazione sono i dispositivi **Internet delle cose** (IoT) che permettono la collaborazione all'interno di un'unica grande struttura da parte di mezzi come macchine industriali, database, smartphone, tablet, sensori e attuatori sfruttando concetti come **big data**, rete Internet, intelligenza artificiale e analisi dati.

In senso pratico, l'industria 4.0 è il completo controllo del ciclo produttivo industriale, l'evento che tutti i dispositivi siano collegati e i loro comportamenti siano costantemente immagazzinati e monitorati porta ad una gestione che sarebbe impossibile per un lavoratore umano. Il fatto che le macchine siano costantemente connesse e il salvataggio minuzioso delle informazioni riguardo i loro comportamenti apre ad innumerevoli benefici quali controllo remoto, tracciabilità, manutenzione predittiva, analisi del funzionamento, applicazioni in tempo reale, aggiornamenti facili e veloci, monitoraggio. I punti cardini sono quindi la raccolta informazioni e la connessione tra i dispositivi le quali rendono il lavoro utile finale molto maggiore rispetto alla somma del risultato di tutti i singoli dispositivi. Altro aspetto importante è la personalizzazione, in questo modo si può uscire dal formato standard dell'industria e creare un'architettura e metodo di lavoro ottimizzato per i propri obiettivi.

Ciò ha portato all'applicazione dello stesso principio anche in molti diversi ambiti come settore sanitario, farmaceutica, urbanistica, statistica, automobilismo e altri. Possiamo quindi dedurre che l'industria 4.0 è un concetto molto vasto che ancora non ha fissato i suoi limiti ma, anzi, ha un grande potenziale e punta ad espandersi ulteriormente nel futuro. Per avere un'idea generale degli aspetti trattati, guardare la figura 2.1.

2.1.1 Rivoluzioni industriali: cause e conseguenze

Per capire le motivazioni e cause che hanno portato all'industria 4.0 bisogna ricapitolare le ragioni storiche delle precedenti rivoluzioni industriali [32]:

- **Prima rivoluzione industriale** [33] [34]. Avvenne nella seconda metà del 1700 in Inghilterra e interessò principalmente il settore tessile e metallurgico. Ciò trasformò un sistema

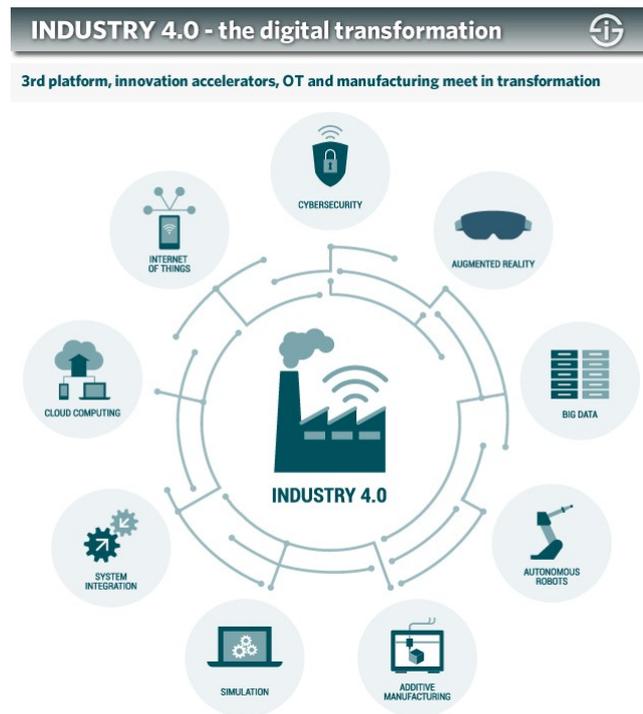


Figura 2.1. Sguardo generico all'industria 4.0 [28]

economico che era storicamente composto da agricoltura, artigianato e commercio a uno basato sull'industria caratterizzato dall'uso di macchine prima azionate tramite forza meccanica (telaio) e poi grazie all'invenzione del motore a vapore (macchina a vapore). Non esistono delle cause specifiche per questo avvenimento ma si deve guardare alla somma di diversi fattori. Alcuni tra i motivi sono: una classe agricola non molto forte come in altri paesi, una società borghese ormai consolidata arricchitasi tramite il commercio, la ricchezza e forza del paese dovute al fatto di essere una potenza commerciale e militare, l'aumento della popolazione e quindi della richiesta. Le conseguenze furono la creazione del capitalismo e della classe operaia.

- **Seconda rivoluzione industriale** [35]. Ebbe luogo nell'Europa occidentale durante la seconda metà del 1800. Le sue cause sono dovute principalmente al progresso scientifico avvenuto in quel periodo storico che portò a varie invenzioni applicabili all'industria e, nella vita comune, generò condizioni di vita migliori. In particolare, le scoperte che più influenzarono questa rivoluzione furono il petrolio e l'elettricità dalle quali si diramarono una serie di altre invenzioni. Tantissime discipline ebbero il loro splendore: la medicina intraprese moltissime scoperte e cure ad antiche malattie quali tubercolosi, peste, lebbra, malaria, furono migliorate le situazioni igienico-sanitarie, la scoperta dell'acciaio comportò un grande sviluppo ferroviario e navale, le telecomunicazioni ebbero la nascita del telegrafo e poi del telefono. L'approccio dei lavoratori all'industria e al lavoro fu radicalmente cambiato ed ottimizzato da Taylor e Ford, principi che valgono tutt'oggi. Si ha la creazione di vere e proprie grandi industrie.
- **Terza rivoluzione industriale** [36]. Si realizzò nella seconda metà del 1900 (secondo dopoguerra) nei paesi del primo mondo. Il suo sviluppo fu causato da: crescita delle conoscenze scientifiche e tecnologiche nate durante le guerre mondiali per scopi bellici e poi nel dopoguerra (per esempio, la rete ARPANET, divenuta poi Internet, da parte del dipartimento della difesa degli Stati Uniti), crescita economica grazie alla ricostruzione post-bellica, scoperta

e sfruttamento dell'energia atomica. Per Stati Uniti e Unione Sovietica parte la corsa allo spazio aprendo nuove conoscenze su un campo ancora poco conosciuto. Le invenzioni che più hanno determinato la rivoluzione sono state quelle in campo informatico, elettronico e telematico: transistor, microprocessori, radio, Internet e poi Web hanno portato un considerevole impatto con un'evoluzione esponenziale nel tempo. Si ha la creazione di multinazionali e il fenomeno della globalizzazione.

- **Quarta rivoluzione industriale o industria 4.0** [32]. Questa si sta sviluppando da circa il 2011 ed è ancora in corso, nata in Germania ma subito diffusa nei paesi del primo mondo. È strettamente legata alla rivoluzione precedente ma, al contrario degli altri casi, non è sostanzialmente basata su nuove invenzioni e scoperte del periodo ma il fulcro è un'evoluzione e perfezionamento di prodotti già esistenti. Per esempio, il lato **big data** costituisce un elemento importante dell'architettura 4.0 ma questa è solo una particolare applicazione di una base di dati, stesso discorso per Internet applicato a diversi dispositivi anziché unicamente a terminali appositi. Quindi più che invenzioni si ha un corretto sfruttamento, collegamento e miglioramento di mezzi già applicati nella terza rivoluzione industriale e la generazione di nuove idee e metodi di lavoro applicate ai lavoratori e al modo di gestire lo stabilimento. Le cause non sono quindi da ricercare in qualcosa di fisico ma, in modo generico, nel fenomeno di digitalizzazione. Conseguenze di quest'ultima, applicate ora all'industria, sono la continua connessione di tutti i dispositivi, l'acquisizione e salvataggio costante di dati e informazioni. La creazione di programmi per l'utilizzo e il monitoraggio delle macchine porta ad una più facile gestione dell'impianto, in questo modo può essere aumentata la sua complessità, cosa che prima non era possibile a causa della limitatezza umana. I lavoratori non devono più svolgere grandi impieghi manuali ma sono richieste conoscenze più tecniche riguardo la macchina e il funzionamento dei vari dispositivi forniti dall'azienda.

Possiamo quindi sintetizzare che l'industria 4.0 non è la conseguenza diretta di invenzioni e scoperte come avvenuto nel passato ma è dovuta al fenomeno digitale. La tabella 2.1 mette a confronto le varie rivoluzioni industriali.

Rivoluzione industriale	Industria	Caratteristiche	Conseguenze
<p>Prima rivoluzione industriale - Inghilterra (1750 - 1850)</p>	<ul style="list-style-type: none"> • Trasporti • Macchina a vapore di James Watt(1765) • Metallurgia • Industria tessile • Ingegneria 	<ul style="list-style-type: none"> • Abolizione schiavitù • Persone migrano nelle grandi città • Specializzazione • Nuove tecnologie • Capitalizzazione 	<ul style="list-style-type: none"> • Urbanizzazione • Aumento della popolazione • Miglioramento della vita • Rivoluzione sociale
<p>Seconda rivoluzione industriale - Gran Bretagna, Germania, USA (1870 - 1914)</p>	<ul style="list-style-type: none"> • Ingegneria • Telecomunicazioni • Industria chimica • Industria navale • Gestione aziendale 	<ul style="list-style-type: none"> • Carbone (basso costo) • Elettricità • Produzione di massa 	<ul style="list-style-type: none"> • Ferrovie • Produzione di ferro e acciaio • Produzione di massa • Ampio uso dei macchinari • Produzione di carta e gomma • Fertilizzanti
<p>Terza rivoluzione industriale - Paesi del primo mondo (circa 1960 - 2011)</p>	<ul style="list-style-type: none"> • Trasporti • Informatica • Elettronica • Ambito medico • Ingegneria genetica 	<ul style="list-style-type: none"> • Manifattura digitale • Nuovi processi • Software • Nuovi materiali • Robot con movimenti agili • Trasporti con minori emissioni 	<ul style="list-style-type: none"> • Produzione di massa • Produzione in paesi a basso costo di manodopera • Richiesta di lavoratori con specifiche competenze • Energia rinnovabile
<p>Quarta rivoluzione industriale - Germania (dal 2011)</p>	<ul style="list-style-type: none"> • Tutti i tipi di industria • Tutti i tipi di economia • Molte discipline 	<ul style="list-style-type: none"> • Digitalizzazione • Internet • Nuove tecnologie 	<ul style="list-style-type: none"> • Aziende digitali • Intelligenza artificiale • Problema della disoccupazione causata dalla sostituzione uomo-macchina

Tabella 2.1. Confronto tra le 4 rivoluzioni industriali [37]

2.1.2 Piano Industrie 4.0

Il termine industria 4.0 è la traduzione dal tedesco di **Industrie 4.0** [29]. Questo è il nome di un’iniziativa strategica nazionale nata in Germania grazie al **Ministero dell’educazione e ricerca (BMBF)** e il **Ministero degli affari economici ed energia (BMWI)**, il vocabolo è poi diventato famoso poiché utilizzato durante l’evento **Hannover Messe** del 2011 [30]. Il piano ha una durata prevista di 10-15 anni ed ha avuto inizio nell’anno 2011 con l’obiettivo di non adagiare l’industria tedesca all’immobilità ma renderla dinamica nonostante essa fosse già leader nel settore. Per esso è stato stanziato un fondo di 200 milioni di euro. **Industrie 4.0** era stato inizialmente pensato unicamente per l’ambito manifatturiero infatti il suo focus era, a grande livello, assicurare il futuro dell’industria manifatturiera tedesca [31] ma lo stesso concetto è stato trasposto a vari altri settori.

È necessario (...) gestire velocemente la fusione tra il mondo online e il mondo della produzione industriale. In Germania, noi la chiamiamo industria 4.0

(Angela Merkel, cancelliere federale della Germania 2005-in carica)

La Germania fu influenzata dalla concorrenza statunitense che iniziava, anche se timidamente, a proporre incentivi riguardo l’**Advanced manufacturing**, la quale aveva concetti simili all’industria 4.0, e portò quindi il paese tedesco a doversi innovare per non essere sorpassato in un campo in cui era sempre stato leader mondiale. Perciò bisognava attuare una doppia strategia: da una mano doveva mantenere l’egemonia già consolidata, dall’altra era necessario adottare nuovi sistemi ciberfisici in modo da prepararsi al cambiamento e creare un nuovo tipo di mercato diverso da quello, ormai classico e ben definito, dell’industria 3.0. Puntando sulla ricerca, la Germania riuscì a sintetizzare in otto punti chiave i principi di questa nuova gestione industriale:

- **Standardizzazione e architettura di riferimento.** Realizzazione di ambienti standard per determinati scenari, in questo modo è possibile partire da una solida base per la generazione di un impianto industriale di un qualsiasi tipo senza dover studiare ogni volta la soluzione base.
- **Gestione di sistemi complessi.** Le macchine utilizzate migliorano con il tempo la loro efficienza ma anche la loro complessità, per questo è necessario istruire ed aiutare i lavoratori in modo da gestire un aumento della difficoltà. Oltretutto è bene associare a queste macchine altri strumenti o programmi per semplificare la loro fruizione.
- **Connessione a banda larga.** Fornisce un efficiente accesso a qualsiasi informazione grazie Internet e migliora enormemente le comunicazioni.
- **Sicurezza.** Con ciò si intende sia sicurezza fisica per i lavoratori e l’ambiente in cui si lavora sia sicurezza informatica dei dati usati e generati dalle varie macchine. I sistemi devono quindi essere progettati già con i requisiti di protezione necessari (**security by design**). È importante informare i dipendenti riguardo le pratiche per la tutela della loro stessa persona e dei dati trattati.
- **Organizzazione del lavoro e progettazione.** Essendo che le macchine aumentano la loro autonomia, i lavoratori sono portati a compiere meno sforzi fisici però svolgono un ruolo orientato al controllo dello stato dell’impianto e quindi si fanno carico di responsabilità. Ciò comporta anche una maggiore possibilità di crescita.
- **Istruzione e sviluppo professionale.** Questo punto chiave è strettamente associato a quello precedente, i compiti svolti dal lavoratore mutano profondamente passando da un approccio nettamente manuale a uno prevalentemente mentale e conoscitivo. È necessario che il dipendente sia istruito e aggiornato costantemente sul suo campo di professione.

- **Piano legislativo.** L'azienda deve essere in ordine con le leggi e l'attuale legislazione. Ciò comporta un trattamento adeguato dei dati, del personale, degli accessi, delle condizioni di lavoro, eccetera. Tutto ciò deve essere applicato e monitorato sia nel lato pratico che burocratico.
- **Efficienza delle risorse.** Questo comporta da un lato la salvaguardia dell'ambiente dato dal minore inquinamento, dall'altro versante un risparmio economico dovuto alla riduzione delle materie prime necessarie per svolgere un determinato lavoro.

2.1.3 Principi

Come già detto, non esiste una definizione ben delineata dell'industria 4.0 e, di conseguenza, non esistono dei principi saldi su cui basare la sua definizione. Per questo motivo, Hermann, Pentek e Otto hanno analizzato le parole più utilizzate nelle pubblicazioni riguardanti l'industria 4.0 e, in base a queste, hanno stilato i suoi 4 punti chiave fondamentali [38]. Cercando nella letteratura, molti approvano questa ricerca, altri aggiungono o accomunano alcuni concetti non distaccandosi di molto dall'idea dei tre studiosi presentati, per questo motivo la loro architettura sembra tutt'oggi molto valida. Ecco i 4 criteri su cui si basa l'industria 4.0:

- **Interconnessione.** La connessione deve avvenire tra tutti gli attori in scena nell'azienda: macchine, dispositivi, sensori e soprattutto persone. In questo modo tutti condividono dati che possono essere sfruttati dagli altri partecipanti della rete in modo da ottenere una forte collaborazione per un fine comune. Il rapporto che si instaura deve essere di 3 tipi: umano-umano, umano-macchina e macchina-macchina. Ognuno tra questi è necessario per ottenere buone prestazioni. Ciò genera il problema del formato per la comunicazione tra macchine; i sistemi utilizzati potrebbero scambiare dati secondo un metodo diverso non riuscendo a capirsi tra loro. Per questo motivo è importante la standardizzazione in modo che l'industria diventi modulare, garantendo che due prodotti collaborino tra loro nonostante abbiano produttori diversi. Allo stesso tempo, la connessione ad una rete comune e l'immagazzinamento di una moltitudine di dati comporta un sostanziale rischio dal punto della sicurezza informatica, è bene che questa venga considerata sia a basso (singolo dispositivo) che alto livello (intero impianto). Nella tabella 2.2 è possibile analizzare le possibilità e i rischi dell'interconnessione.

Possibilità	Rischi
– Semplificazione nella collaborazione	– Sicurezza dei dati
– Flusso di informazioni comuni	– Grande mole di dati da gestire
– Riduzione degli sprechi per il rilevamento dei dati	– I dati possono essere mal interpretati
– Miglioramento qualità dei dati	– Mancanza di comunicazione standard

Tabella 2.2. Possibilità e rischi dell'interconnessione [39]

- **Trasparenza dell'informazione.** Ciò è resa necessaria dalla complessità, numero di macchine e persone interconnesse tra loro. I sensori digitalizzano il mondo reale ma i dati grezzi ricavati vanno elaborati in informazioni tenendo conto del contesto in maniera da prendere le giuste decisioni. Oltretutto i dati vanno visualizzati in modo consono da essere letti facilmente da un umano, per esempio attraverso dei grafici. Bisogna quindi contestualizzare i dati in modo da ottenere trasparenza per garantire la loro giusta interpretazione. Nella tabella 2.3 è possibile analizzare le possibilità e i rischi della trasparenza dell'informazione.

Possibilità	Rischi
<ul style="list-style-type: none"> – Miglioramento dell'analisi dedotta dai dati – I lavoratori hanno una visione contestualizzata del lavoro della macchina 	<ul style="list-style-type: none"> – Un contesto errato comporta decisioni errate

Tabella 2.3. Possibilità e rischi della trasparenza dell'informazione[39]

- **Decisioni decentralizzate.** La giusta combinazione tra i due punti precedenti deve portare alla presa di decisioni riguardo il comportamento delle macchine da parte dei dispositivi stessi. In questo modo abbiamo in particolare due vantaggi: gli umani non vengono scomodati per ogni minima alterazione dello stato del macchinario, ogni congegno può prendere decisioni migliori rispetto a quelle umane perché ha a disposizione una moltitudine di dati che una persona non riuscirebbe ad analizzare in un tempo breve. La presenza di un lavoratore è richiesta solo in caso la macchina non riesca a trarre una determinata scelta, per esempio due obiettivi vanno in contrasto e non si sa a quale dare maggior rilevanza. Nella tabella 2.4 è possibile analizzare le possibilità e i rischi delle decisioni decentralizzate.

Possibilità	Rischi
<ul style="list-style-type: none"> – Utilizzo efficiente delle risorse – L'impianto è flessibile in base al suo andamento – Reazione rapida agli imprevisti 	<ul style="list-style-type: none"> – Responsabilità da chiarire – La macchina può prendere decisioni errate senza consultazione umana – Dipendenza dalla tecnologia

Tabella 2.4. Possibilità e rischi delle decisioni decentralizzate [39]

- **Assistenza tecnica.** Essendo che le macchine sono quasi del tutto autonome, i lavoratori non devono più monitorare costantemente l'apparecchiatura ma il loro ruolo è costituito solo da prendere le giuste decisioni che sono troppo difficili per essere svolte da un robot. Però anche per gli umani potrebbero presentarsi situazioni difficili da gestire, perciò è necessario fornire vari strumenti di assistenza tecnica che aiutino la persona ad avere un quadro il più completo possibile sullo stato della macchina. Per esempio, potrebbe essere di aiuto avere un'applicazione che mostri grafici e andamenti di dispositivi e dell'impianto. Entrano così in gioco servizi mobili quali smartphone, tablet, smartwatch. Un esempio pratico di una tale applicazione è definito nel dettaglio in Implementazione soluzione. Nella tabella 2.5 è possibile analizzare le possibilità e i rischi della trasparenza dell'assistenza tecnica.

2.1.4 Componenti

I principi descritti nella sezione precedente vengono in pratica attuati da diversi dispositivi. Anche in questo caso la suddivisione non è ben delineata ma viene presentata in modo leggermente diverso da ogni autore. Nel seguente testo verrà presa in considerazione lo schema tracciato, come per Principi, da Hermann, Pentek e Otto [40]. I componenti dell'industria 4.0 sono:

Possibilità	Rischi
<ul style="list-style-type: none"> – Monitoraggio semplificato – Aumento delle prestazioni lavorative – Decisioni umane più accurate 	<ul style="list-style-type: none"> – Dipendenza dagli strumenti informatici utilizzati

Tabella 2.5. Possibilità e rischi dell'assistenza tecnica [39]

- **Sistemi ciberfisici (CPS).** Questi costituiscono l'attuazione del mondo digitale nella realtà. Possono essere programmati e comandati dai lavoratori in modo da attuare certi lavori, rispetto alle normali macchine queste sono molto più flessibili poiché programmabili e restituiscono un feedback in modo da poter monitorare il loro comportamento e tracciare un andamento nel tempo. La loro evoluzione è divisa in 3 fasi: la prima era costituita dall'aggiunta di sistemi di identificazione come RFID, nella seconda generazione sono stati aggiunti sensori e attuatori che però rimanevano ancora limitati, nell'ultimo stadio i sistemi ciberfisici sono collegati alla rete e possono salvare e analizzare i dati. In particolare sono diventati importanti dopo il loro collegamento alla rete Internet.
- **Internet delle cose o Internet of Things (IoT).** IoT è la connessione e collaborazione di vari elementi ad un'unica rete, solitamente Internet [41]. I dispositivi IoT sono qualsiasi oggetto che interagisce e coopera in una rete composta da altrettanti componenti in modo da raggiungere un obiettivo comune. Alcuni esempi sono: sensori, attuatori, smartphone, telecamere, robot, automobili intelligenti. Anche i sistemi ciberfisici possono essere considerati IoT, questi comunicano e cooperano per arrivare ad un prodotto finale.
- **Internet dei servizi o Internet of Services (IoS).** È la fornitura di un servizio tramite Internet e l'architettura necessaria per rendere possibile l'offerta. Ciò comporta molti benefici. Il lavoratore può accedere al servizio da un qualsiasi dispositivo connesso alla rete tramite un web browser. La macchina stessa può essere programmata per accedere e comunicare con l'applicazione grazie a delle API. Si ha maggiore flessibilità perché il programma fornito non è dipendente dal modello delle macchine, sistema operativo, terminale da cui si accede, eccetera. L'applicazione può essere venduta a più aziende proprio perché flessibile quindi è compatibile a qualsiasi ambiente di lavoro e si può accedere al servizio anche fuori dall'impianto industriale.
- **Industria intelligente (detta anche smart).** È basata sul fornire aiuto a macchine e lavoratori all'interno dell'infrastruttura industriale. In pratica è attuata da sistemi che lavorano in background e forniscono vari servizi utili. Per esempio, un magazzino potrebbe indicare la quantità degli oggetti rimasti e, se questa scende sotto una certa soglia, ordinare una determinata quantità di quel prodotto; ciò è svolto in maniera invisibile al lavoratore che così ha meno carico di lavoro poiché non deve controllare la quantità delle merci ed effettuare gli ordini.

Nella tabella 2.6 è possibile vedere come i componenti aderiscono ai principi dell'industria 4.0.

2.1.5 Sistemi ciberfisici

I sistemi ciberfisici [42] sono dei dispositivi che collaborano tra loro e che interagiscono sia con il mondo reale sia con il mondo digitale tramite servizi i quali accedono ai dati usando una rete, spesso Internet, come mezzo. È proprio l'intersezione tra il mondo fisico e digitale che genera il loro valore. Automobili autonome, macchine utilizzate per operazioni chirurgiche, segnaletica urbana

	Sistemi ciberfisici	Internet delle cose	Internet dei servizi	Industria intelligente
Interconnessione	X	X	X	X
Trasparenza dell'informazione	X	-	-	X
Decisioni decentralizzate	X	-	-	X
Assistenza tecnica	-	-	X	X

Tabella 2.6. Come i componenti concretizzano i principi dell'industria 4.0 [40]

intelligente, servizi intelligenti per la casa sono solo alcuni esempi, da essi si intuisce che i sistemi ciberfisici non sono utilizzati unicamente in ambito industriale ma trovano luogo in una moltitudine di applicazioni. La loro connessione applica il concetto di **Internet delle cose**, fattore che li rende ancora più performanti grazie alla condivisione di dati da parte di molti prodotti. Per una panoramica sull'argomento, osservare la figura 2.2.

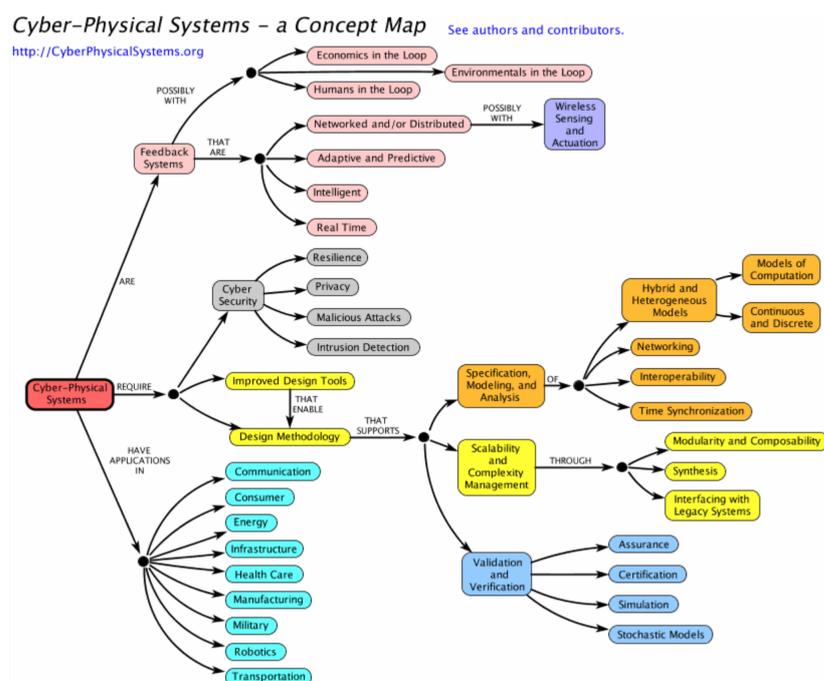


Figura 2.2. Mappa concettuale sui sistemi ciberfisici [43]

Grazie ad essi è cambiato il concetto di macchina industriale [44]. Le caratteristiche fondamentali ricercate in un dispositivo non sono più la precisione e la velocità di produzione ma l'abilità nel prendere decisioni in modo autonomo. Ciò è mostrato meglio nella tabella 2.7.

L'unione e la collaborazione in modo autonomo di vari sistemi ciberfisici genera un sistema di produzione ciberfisico(CPPS). Questo deve presentare 3 caratteristiche principali:

- **Intelligenza.** Gli elementi che compongono il sistema devono lavorare in modo autonomo, saper acquisire dati ed informazioni e prendere decisioni.
- **Connettività.** Saper sfruttare le connessioni con gli altri attori del sistema (anche umani) in modo che la cooperazione e collaborazione diano veramente un valore aggiunto al lavoro.
- **Responsività.** Bisogna saper gestire al meglio cambiamenti ed imprevisti.

	Sorgente	Industria moderna		Industria 4.0.	
		Attributi	Tecnologie	Attributi	Tecnologie
Componente	Sensore	Precisione	Sensori intelligenti e rivelazione guasti	Predizione automatica	Monitoraggio degradazione della macchina e predizione tempo di vita rimanente della macchina
Macchina	Controllore	Qualità e velocità di produzione	Monitoraggio basato sulle condizioni e diagnostica	Predizione e confronto con andamento ideale automatici	Predizione e monitoraggio automatico della salute della macchina
Sistema di produzione	Sistema manifatturiero connesso	Produttività	Riduzione degli sprechi	Configurazione, manutenzioni e organizzazione automatiche	Produttività assicurata con un intervento umano minore possibile

Tabella 2.7. Differenza delle specifiche dei macchinari nell'industria moderna rispetto all'industria 4.0 [44]

La struttura di un CPPS può essere schematizzata nell'architettura 5C [45] come in figura 2.3. Questa è una piramide formata da:

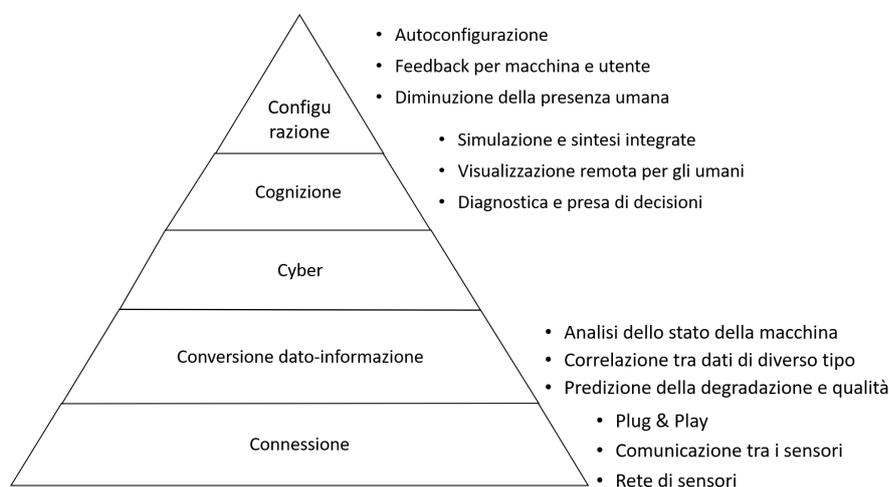


Figura 2.3. Architettura 5C, implementazione di un sistema di produzione ciberfisico [45]

- **Connessione.** Il cardine su cui si basa tutto il sistema è l'acquisizione dei dati. Riguardo ciò sono importanti alcuni principi: considerare i vari tipi di dati, la procedura di misurazione e il trasferimento al server centrale. È quindi importante scegliere dei sensori adeguati.
- **Conversione dato-informazione.** Una volta ottenuti i dati, bisogna trarre informazioni utili da essi. Per questo si utilizzano strumenti basati sulla *Prognostic and health*

management (di cui si parlerà meglio nella sezione Prognostic and health management e PHM applicata ai sistemi ciberfisici) per fare predizioni di vario tipo tra cui il tempo stimato di vita di una macchina.

- **Cyber.** Le informazioni vengono mandate a tutti i dispositivi in modo da formare una rete interconnessa. Le macchine possono così conoscere l’andamento degli altri membri della rete e comparare il proprio lavoro con quello altrui. È possibile anche analizzare l’operato attuale rispetto al passato traendo maggiori informazioni da ciò.
- **Cognizione.** Questo livello ha l’obiettivo di visualizzare i dati per il lavoratore aiutandolo a prendere una decisione grazie a grafici e comparazioni.
- **Configurazione.** Ha l’obiettivo di restituire un feedback all’utente e in base ad esso vengono fatte delle scelte di autoconfigurazione in modo che la macchina riesca ad attuare correzioni e azioni preventive autonomamente.

2.1.6 Internet delle cose (IoT)

L’**internet delle cose** è un paradigma secondo il quale ogni dispositivo come sensori, attuatori, smartphone, eccetera sono connessi tra loro e interagiscono per un obiettivo comune [46]. Il punto cardine è che, grazie alla collaborazione tra gli elementi nella rete, si ottiene un risultato migliore della somma dei risultati ottenuti singolarmente. Pian piano lo stesso concetto si sta diffondendo anche nell’ambiente domestico con oggetti come frigoriferi, televisioni, condizionatori, lampade, telecamere per la sicurezza, i quali ora possono essere regolati tramite un qualsiasi smartphone. Lo stesso concetto si applica attraverso l’industria 4.0, tutte le macchine sono connesse all’intero sistema, scambiano informazioni tra loro ottimizzando la collaborazione, possono essere monitorate e gestite facilmente dai lavoratori tramite cellulari o tablet e, grazie a tutti i benefici precedenti, ricavano una produzione migliore sotto vari aspetti. La vera rivoluzione non è dovuta alla potenza di calcolo e alla connessione Internet in sé ma al fatto che ciò sia possibile in qualsiasi oggetto anche di ridotte dimensioni [47]. Ciò apre i benefici di **big data**, analisi, diagnostica, collaborazione, monitoraggio, comunicazione che prima non erano possibili a questo livello. La connessione Internet di questi dispositivi comporta però un rilevante problema: la sicurezza.

Nella figura 2.4 è raffigurata l’architettura IoT [48]. A livello inferiore si trova il *dispositivo* vero e proprio composto da software e hardware, entrambi servono per fornire i servizi principali dell’oggetto e la scheda di rete per la connessione. Nel mezzo invece troviamo la *connettività* attraverso la rete, essa unisce il mezzo IoT con il *cloud* rendendo possibili i benefici del lato superiore. Al vertice troviamo il *cloud* che fornisce vari vantaggi: gestione del dispositivo, comunicazione macchina-macchina e uomo-macchina, analisi e apprendimento automatico. I campi di *sicurezza*, *integrazione con i sistemi aziendali* e *sorgenti esterne* percorrono completamente l’altezza dell’architettura, infatti tutti e tre i settori orizzontali descritti precedentemente devono implementarli.

2.1.7 Big data

I **big data** non sono caratterizzati unicamente, come suggerirebbe il nome, da una mole di dati ma anche da una struttura non prefissata ed eterogenea [49]. Per questo motivo sono difficili da trattare con i classici database relazionali ma hanno bisogno di tecniche più flessibili, solitamente NoSQL. Oltretutto sono frequentemente associati ad analisi in tempo reale quindi la velocità di elaborazione è rilevante, a maggior ragione è bene adottare nuove metodologie che garantiscano un’attesa minima; il compito non è di semplice soluzione vista la loro quantità e disomogeneità. I **big data** vengono spesso definiti tramite le tre “V” che, con il passare del tempo e a seconda del testo in analisi, sono diventate quattro o cinque. Il primo che diede questa designazione fu l’azienda **Gartner** enunciando le dimensioni *volume*, *velocità* e *varietà*. La quarta, *veracità*, fu aggiunta da IBM. La quinta, *valore*, può cambiare a seconda del testo in esame (a volte questa è definita come

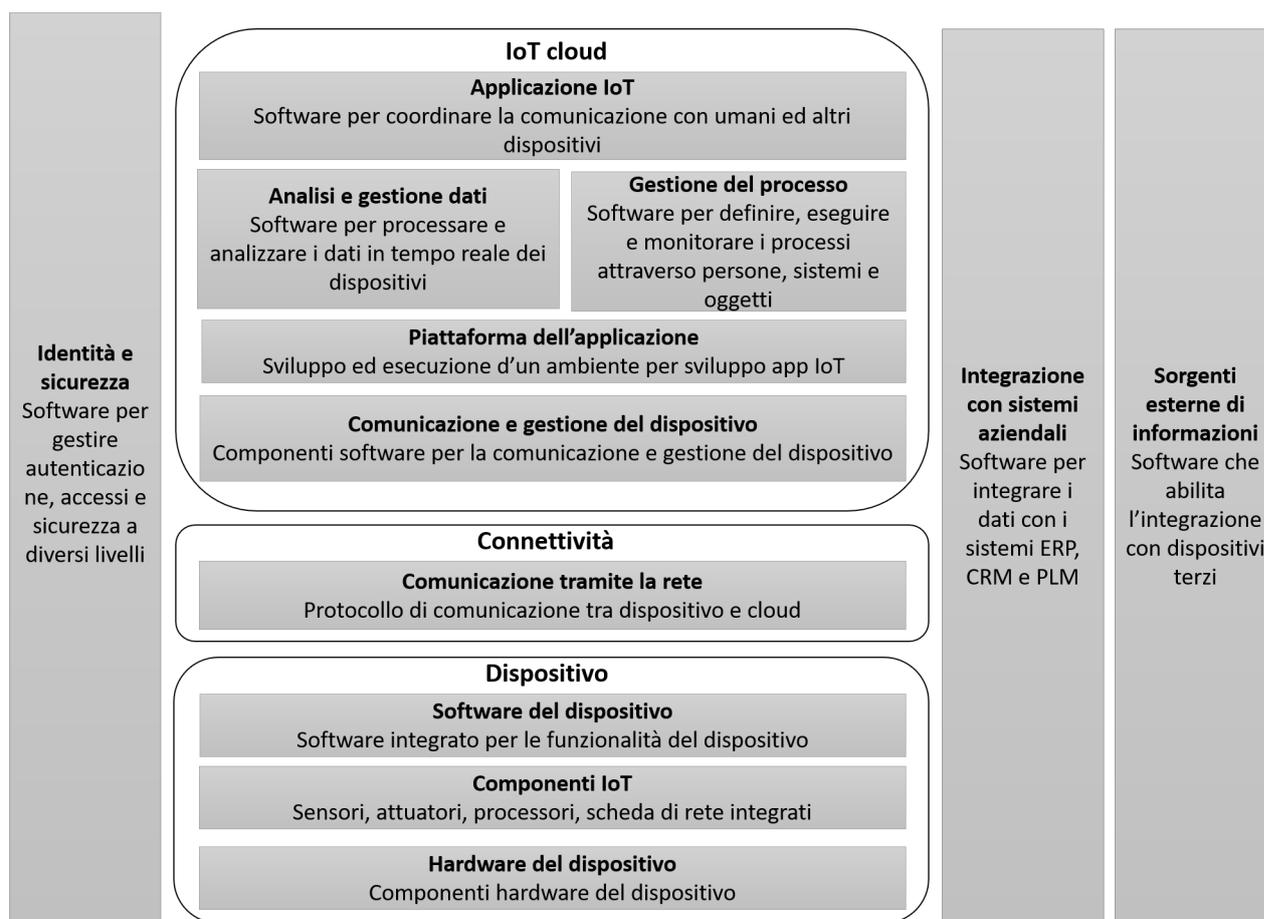


Figura 2.4. Architettura Internet delle cose [48]

la somma di tutti i punti chiave precedenti [50]). Ecco le cinque “V” a cui si è arrivati oggi [49] [51] [52]:

- Volume.** La quantità di dati prodotti ed immagazzinati cresce con un rateo sempre maggiore. Alcuni indicano i dati come *“big”* se oltrepassano in dimensione totale un terabyte o petabyte ma non si ha una soglia convenzionale per passare a dati *“normali”* a *“grandi”*. Ciò comporta problemi sia a livello hardware (costo della memoria) sia software (algoritmi lenti per input di certe dimensioni).
- Varietà.** I dati sono di diverso tipo e provengono da sorgenti differenti. Questi possono essere strutturati, semi-strutturati o senza struttura quindi bisogna riuscire ad integrare tutte e tre le classificazioni per trarne informazioni. Per esempio i sensori possono produrre: testo, dati web, audio, video, file di log, tweet, eccetera.
- Velocità.** La frequenza con cui i dati devono essere creati, processati ed analizzati aumenta sempre di più. In alcuni casi è richiesta un'analisi in tempo reale, è quindi logico diminuire la latenza al minimo.
- Veracità.** Si intende l'affidabilità dei dati. Questa è importante perché da essa deriva la qualità delle previsioni ricavate dalla fase di analisi. È quindi importante pulire i dati, usare sensori adatti ed eliminare il rumore.

- **Valore.** Una volta ottenuti tutti i dati secondo i principi appena enunciati, è necessario estrarre informazioni utili. Il concetto è rispondere alla domanda “Quali valori e risultati posso ricavare?”. La risposta va ricercata tramite un lavoro di analisi. Un esempio è l’autoapprendimento e autonomia delle macchine che imparano dal proprio passato e dalla comunicazione con altri dispositivi.

Sfruttare i **big data** nell’industria 4.0 è quindi necessario se si vuole adottare una politica di digitalizzazione visto che l’eterogeneità e la grande quantità di dati contraddistinguono l’ambiente industriale. Ciò può essere un problema ma una risorsa allo stesso tempo. TDWI ha intervistato 325 persone (in maggioranza informatici) ottenendo, come rappresentato in figura 2.5, che la maggior parte vede i **big data** come un’opportunità anziché un grattacapo.

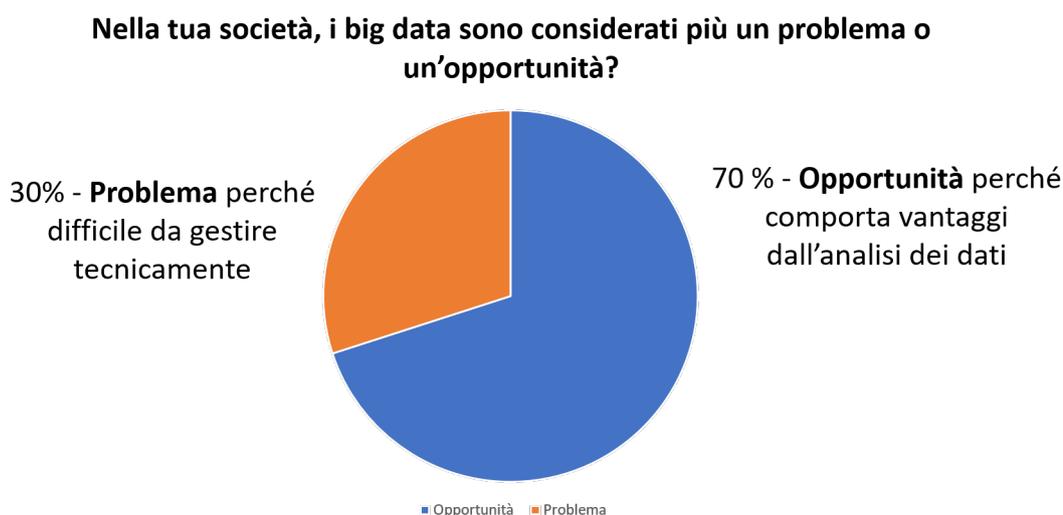


Figura 2.5. Big data problema o opportunità? Basato su 325 risposte nel quarto trimestre 2011 [53]

Siccome i benefici principali legati ai **big data** derivano dal lato analisi, questi verranno presentati nella sezione PHM e big data, dopo aver definito la **Prognostic and health management**.

2.1.8 Benefici

Siccome l’industria 4.0 è principalmente dovuta alla digitalizzazione, i suoi benefici sono una conseguenza diretta del mondo digitale [28]. Ecco un elenco dei principali vantaggi:

- **Aumento della produttività dovuto all’ottimizzazione e automazione.** È il beneficio che più spunta all’occhio. Ciò è la conseguenza di diversi aspetti: risparmio di tempo e materiale, aumento del profitto, riduzione degli sprechi, prevenzione di errori e ritardi, ottimizzazione del macchinario in base alle condizioni del momento eccetera.
- **Eventi in tempo reale.** Le macchine, in base al contesto in cui si trovano al momento, modificano il loro modo di operare ottimizzando varie caratteristiche quali velocità di produzione, risparmio energetico, qualità del risultato. Ogni singolo evento migliorato non comporta una grande utilità ma quest’ultima è guadagnata dalla somma di tutti i vantaggi ottenuti.
- **Continuità della macchina grazie alla manutenzione predittiva e monitoraggio.** Parte del tempo perso (e quindi guadagno economico perso) nella produzione è causato dai fallimenti delle macchine che devono così rimanere ferme fino all’attività di manutenzione. Predire la rovina di un dispositivo comporta una miglora sul tempo di lavoro dell’impianto grazie a manutenzioni mirate.

- **Migliore qualità del prodotto finale.** Il monitoraggio del comportamento di un macchinario garantisce controllo e predizione della qualità. Questo beneficio è uno dei più ricercati, in particolare secondo la mentalità della terza rivoluzione industriale.
- **Migliori condizioni di lavoro e sostenibilità.** I lavoratori sono aiutati da diversi strumenti durante i loro compiti e hanno in carico meno attività fisiche. Il monitoraggio delle macchine aumenta la sicurezza dell'impianto e, di conseguenza, dei suoi frequentatori. L'utilizzo di materiali naturali o meno pericolosi diminuisce il rischio di malattie nel lungo periodo.
- **Aumento della flessibilità.** La stessa capacità dei servizi informatici riguardo scalabilità e flessibilità viene trasposta nell'ambito industriale. In questo modo è possibile avere più livelli indipendenti tra loro ma interconnessi allo stesso tempo e lavorare in fasce più o meno basse. Allo stesso tempo ci si può adattare alle diverse situazioni senza particolari problemi.
- **Personalizzazione.** Ciò va a coprire una richiesta che tende ad aumentare nell'ultimo periodo: sia persone che aziende preferiscono un oggetto il più adatto possibile ai propri gusti ed esigenze. Però l'industria, nella terza rivoluzione industriale, era molto standardizzata poiché si puntava a migliorare la produzione, per questo motivo elaborare uno stesso prodotto con leggere modifiche comportava costi notevoli. L'industria 4.0 è invece molto più flessibile rispetto al passato rendendo le modifiche nella catena di montaggio molto più facili da applicare.
- **Sviluppo di modelli di ricavo innovativi.** L'industria 4.0 apre a nuovi tipi di economia e quindi di guadagno. In questo modo si riesce ad essere competitivi con le altre aziende e proporre il proprio prodotto in maniera più accattivante per investitori e clienti.

2.1.9 Sfide da affrontare e orientamento per il futuro

Come già detto più volte, l'industria 4.0 è ancora nella sua fase iniziale quindi esistono vari problemi riguardo la sua implementazione e sfide che dovrebbero venir risolte negli anni grazie alla ricerca e lo studio. Ecco una lista degli impedimenti più importanti [54]:

- **Molte industrie non sono ancora pronte a supportare un cambiamento nel loro classico modo di operare.** Queste non posseggono né la tecnologia né la cultura riguardante l'industria 4.0 per adottare l'evoluzione. Bisogna quindi sviluppare un modello standard in modo da rendere più facile l'adozione sia pratica che concettuale.
- **Scalabilità.** Siccome l'industria 4.0 necessita molti dispositivi sempre più complessi, diventa necessario poter lavorare allo stesso modo a prescindere dal numero di oggetti connessi.
- **Analisi dei dati.** Ogni sistema produce una grande mole di dati in tempo reale e molti di questi non hanno un grande significato se non analizzati. È quindi necessaria un'intelligenza che ricavi informazioni dai dati grezzi. Altro fattore che rende il tutto più difficile è che i macchinari sono eterogenei e quindi la stessa applicazione potrebbe funzionare bene solo con certe attrezzature e non con altre.
- **Sfide legate all'Internet delle cose.** Il concetto IoT è abbastanza difficile da applicare nella sua idealità, connettere molti dispositivi eterogenei e permettere la loro corretta comunicazione non è facile. Oltretutto non esiste tutt'ora un metodo standard da applicare. Tra l'altro non sempre gli oggetti collegati hanno grandi velocità e questo comporta rallentamenti e ritardi. Diviene quindi necessaria una loro ottimizzazione.
- **Standardizzazione.** Questa comporterebbe una grande semplicità nell'adozione dell'industria 4.0, infatti uno maggiori problemi odierni è riuscire a strutturare in modo efficiente e funzionale tutti i diversi tipi di macchinari quindi avere un modello già preimpostato ed ottimizzato renderebbe il lavoro molto più semplice con risultati migliori. Ciò aiuterebbe la

connessione dei dispositivi IoT, la scalabilità dell'impianto e la modularità degli apparecchi utilizzati.

- **Sicurezza e protezione della privacy.** Siccome tutte le tecnologie utilizzate hanno, o avranno, un base digitale, è bene puntare sulla sicurezza informatica sia per incolumità e privacy dei lavoratori sia per il corretto trattamento dei dati utilizzati. Per esempio in ambito medico, è normale trattare i dati di alcuni pazienti ma è dovere mantenerli anonimi e il loro utilizzo deve essere limitato solo al campo di ricerca, non bisogna quindi diffonderli o usarli per scopi terzi. Altro esempio è l'intuizione del modello di lavoro adottato dall'impianto grazie ai suoi dati causando spionaggio industriale. Ultimo scenario, tramite un attacco informatico è possibile spegnere tutti i macchinari di una certo impianto causando una sostanziale perdita di tempo e denaro.

La ricerca punta quindi a risolvere i problemi appena elencati. Ci sono però alcuni campi che presentano forti potenzialità e per questo ci si sta indirizzando con forza su di essi. Come già detto varie volte, ognuno adotta delle politiche diverse per l'industria 4.0 quindi non esiste un elenco ufficiale su cui tutti convergono ma si è cercato di raccogliere gli aspetti più citati. Ecco l'elenco dei principali campi di ricerca su cui ci si sta dirigendo [54] [55] [56] [57] [58]:

- **Simulazione e gemello digitale.** Questa permette di virtualizzare una macchina e testarla anche senza averne una fisica. Di conseguenza si può prevedere il suo andamento, la sua efficienza e, cosa molto importante, è possibile istruire i lavoratori sul suo utilizzo senza il rischio di guastare un vero e proprio apparecchio, cosa che comporterebbe una perdita economica. È anche possibile vedere il comportamento della macchina nella simulazione e, solo se il risultato positivo, applicarlo nella realtà. Il gemello digitale è una copia digitale di ogni dispositivo, come già detto è possibile svolgere delle azioni nella simulazione e prevedere cosa accadrebbe.
- **Blockchain.** Il World Economic Forum ha predetto che entro il 2027 il 10% del PIL globale sarà immagazzinato tramite una tecnica **blockchain** [59]. La sua applicazione ha potenzialità riguardo scalabilità e sicurezza.
- **Realtà aumentata e virtuale.** Queste possono essere usate sia per la simulazione sia per aiutare i lavoratori. Con la realtà aumentata è possibile mostrare i passi da svolgere per arrivare ad un obiettivo, mostrare informazioni, aiutare l'umano a trovare un certo oggetto nel magazzino, eccetera.
- **Manutenzione predittiva, apprendimento automatico e monitoraggio.** La manutenzione predittiva permette di conoscere in anticipo quando la macchina subirà un guasto perciò è possibile schedulare al meglio le manutenzioni e gli orari di lavoro comportando efficienza e nessuna perdita economica dovuta a ritardi. L'apprendimento automatico, su cui si basa la manutenzione predittiva, permette quindi di prevedere l'andamento della macchina e, di conseguenza, ottenere un livello sempre più alto di automazione; la macchina capisce da sola quando sta per avere un fallimento, aumentare o rallentare il suo operato, avvisare un umano riguardo un'anomalia, eccetera. In generale, il monitoraggio comporta l'ottimizzazione dell'impianto migliorando la produzione nel suo complesso poiché si conosce in modo approfondito lo stato attuale di tutte le attrezzature, informazioni non raggiungibili in modo così approfondito da un umano.

2.2 Il mercato odierno dell'industria 4.0

Pian piano ogni paese sta cambiando le proprie strategie industriali in modo da renderle compatibili con l'industria 4.0. Partendo dalla Germania con **Industrie 4.0**, molti altri paesi hanno iniziato piani simili, per esempio: Giappone tramite **Society 5.0** che ha come obiettivo migliorare l'intera società dal 2016, Regno Unito con **Industrial strategy** dal 2017, Singapore con **Smart Industry**

Readiness Index attuato da Novembre 2017 [60]. Però ancora nessuno è riuscito a portare a termine quest'evoluzione, la maggior parte dei paesi è ancora all'inizio e nemmeno gli Stati più avanzati sono nel pieno del cambiamento.

Per capire lo stato dell'industria 4.0 è meglio prima presentare l'economia generale mondiale. **World economic forum** ha stilato un report [61] nel 2018 prendendo in considerazione le 100 nazioni con l'economia più forte, le quali rappresentano circa il 96% sia del **Valore aggiunto manifatturiero (MVA)** sia del PIL di tutto il mondo. L'analisi è stata condotta in base alla struttura della produzione (structure of production) e i principali criteri che più influenzano la sua crescita (drivers of production), ciò è mostrato in figura 2.6.



Figura 2.6. Criteri dell'analisi sulla produzione mondiale [61]

Siccome ogni nazione ha un proprio tipo di economia, esse sono state divise in 4 tipi:

- **Leader (leading)**. Sono i paesi già affermati nel settore e solitamente responsivi ai cambiamenti, nella storia hanno dimostrato di ottenere buone prestazioni. Sono quindi i paesi con una maggiore forza economica.
- **Obsoleti (legacy)**. Si collocano in questa sezione gli Stati con una forte produzione al giorno d'oggi ma che non sembrano adattarsi all'innovazione.
- **Alto potenziale (high potential)**. Questi non hanno una grande economia nel tempo odierno ma presentano vari criteri che fanno credere in un loro sviluppo verso l'innovazione.
- **Nascente (nascent)**. Non hanno una forte produzione e allo stesso tempo non posseggono ancora i mezzi per arrivare all'industria 4.0.

Nella figura 2.7 è possibile notare la disposizione nel grafico delle fazioni appena elencate e, nella figura 2.8, come si collocano le nazioni rispetto il modo di operare descritto precedentemente.

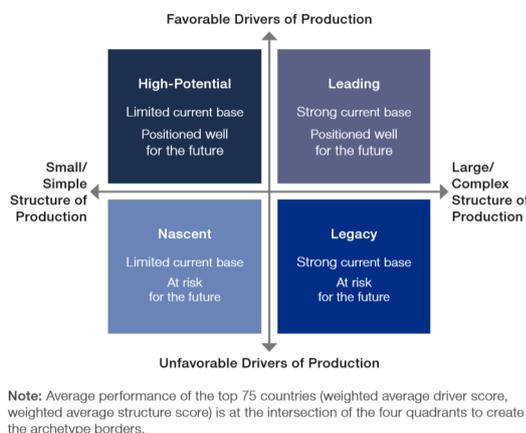


Figura 2.7. Tipi di paesi mondiali in base al loro approccio sull'innovazione [61]

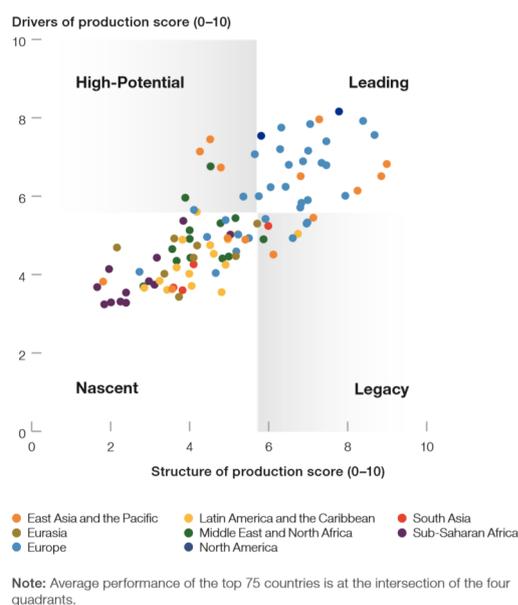


Figura 2.8. Collocazione dei paesi riguardo l'innovazione a livello mondiale [61]

Si deduce dal grafico in figura 2.8 che i 25 Stati leader sono coloro che già hanno l'economia più sviluppata infatti solo questi paesi generano più del 75% del valore aggiunto manifatturiero, ciò conferma il fatto che anche in futuro avranno un ruolo importante nell'economia mondiale. Al contempo si può notare che si ha una maggiore densità di nazioni non pronte ad affrontare l'industria 4.0 nel breve periodo perciò dovranno effettuare un grosso e tempestivo sforzo se non vogliono essere sommerse dalle potenze leader. Gli attori obsoleti sono i paesi che hanno beneficiato per un certo periodo della bassa manodopera ma quest'ultima sta venendo pian piano sostituita dalla tecnologia. Gli elementi ad alto potenziale sono composti dai paesi in ascesa da alcuni anni. Col tempo si avrà una minore redistribuzione della produzione mondiale e, in particolare, la manodopera a basso prezzo tenderà a scomparire. Le nazioni che punteranno più sul cambiamento dovrebbero riuscire ad emergere e mostrare la loro superiorità anche se, per ora, lo stadio dell'industria 4.0 è ancora troppo acerbo per determinare quali saranno i reali vincitori di questa rivoluzione.

2.2.1 Iniziative nell'Unione europea

Per analizzare la situazione dell'Unione europea (UE), è stato utilizzato un report della Commissione europea [62] stilato a Maggio 2017 (in questo documento l'Italia viene presentata in modo diverso rispetto alla sezione Il caso italiano, ciò è dovuto al fatto che l'Italia ha iniziato ad adottare un vero e proprio piano 4.0 nel 2017 e quindi non è stato considerato). Come già detto precedentemente a livello globale, anche nell'UE la rivoluzione 4.0 è iniziata da poco ed ancora in pochi paesi, solo il 41% delle aziende europee ha iniziato ad adottare nuove tecnologie. In compenso il 75% pensa che l'industria 4.0 sia una grande opportunità e il 64% di chi si è spinto verso questa evoluzione ha ricevuto risultati positivi. Nella figura 2.9, è possibile vedere una panoramica dei principali piani 4.0 nell'Unione europea.

I piani adottati sono abbastanza diversi tra loro ma hanno tutti come obiettivo l'aumento della produttività e competitività. Quasi la totalità di essi sono stati il risultato di strategie nazionali anziché di iniziative da parte di privati. I punti cardine su cui si è basata la moltitudine sono: acquisizione di nuove tecnologie (Italia e Regno Unito), sviluppo e ricerca di nuovi prodotti per il miglioramento dei processi industriali (Italia e Germania) e supporto per innovazione e commercio (Regno Unito, Francia e Spagna). Solo Germania e Francia specificano lo sviluppo

	Launch date	Target audience	Budget	Funding approach
	2015	Industry & production base, SMEs & mid-caps	Approx. €10 billion	Mixed
	2011	Manufacturers / producers, SMEs & policy-makers	€200 million	Mixed
	2012	Large companies, SMEs, universities, research centres	€45 million	Public
	2014	General business community	€25 million	Mixed
	2016	Industry, above all SMEs & micro-enterprises	€97.5 million	Public
	2013	Research, academia & industrial & service SMEs	€50 million	Mixed
	2012	Business, industry & research organisations	€164 million	Mixed
	2016	Industry & service sector companies, trade unions	Not yet defined	Public

Figura 2.9. Piani sull'industria 4.0 delle nazioni nell'Unione europea [62]

su tecnologie **Internet delle cose** e sistemi ciberfisici, gli altri finanziano, in modo generico, tecnologie innovative senza imporre vincoli. Quasi nessuna nazione ha puntato unicamente su particolari settori industriali ma ha proposto il proprio piano a qualsiasi ambito. In tabella 2.8 è possibile vedere una panoramica delle azioni e risultati conseguiti da alcuni paesi europei.

Gli autori del report hanno infine osservato dei principi comuni per una politica efficace sull'industria 4.0:

- I piani 4.0 hanno beneficiato del fatto di avere obiettivi chiari e misurabili supportati da indicatori qualitativi e quantitativi.
- Il finanziamento pubblico è importante ma lo è anche quello privato. Bisogna quindi fare in modo di incentivare i privati affinché puntino sull'industria 4.0. Allo stesso modo bisogna puntare sulla ricerca sia pubblica che privata.
- Un cambiamento che parte dall'interno dell'industria (dal basso) rende meglio rispetto ad uno globale della nazione (dall'alto) e garantisce la continuazione dello sviluppo anche dopo il termine dei finanziamenti statali.
- Prestiti e detrazione delle tasse rendono buoni risultati.
- Piccole e medie imprese hanno spesso bisogno di soluzioni più personalizzate.
- Quasi tutti puntano alla digitalizzazione però si concentrano unicamente sull'installazione di nuove tecnologie, bisognerebbe quindi allinearsi meglio agli obiettivi stabiliti.

Nazione	Titolo	Focus	Obiettivi	Destinazione	Tipo di finanziamento	Bilancio	Risultati ottenuti
Germania	Industrie 4.0	Innovazione tecnologica basata sull'integrazione orizzontale e verticale che includa aspetti di sicurezza e nuovi metodi di lavoro	Guidare la manifattura verso la digitalizzazione, interconnessione dei sistemi industriali, supportare la ricerca	Produttori e piccole e medie aziende(SME)	Unione tra fondi pubblici e privati con un rateo tra 2:1 e 5:1.	200 milioni di euro	Riduzione della segregazione industriale, applicazione della ricerca nella pratica, creazione di un'architettura standard da applicare ai nuovi arrivati
Francia	Alliance pour l'Industrie du Futur	Diagnosi SME, incentivi per la modernizzazione, sviluppo offerta nazionale e tecniche avanzate per la manifattura	Modernizzare la produzione e implementare l'uso di tecnologie digitali per creare nuove sorgenti di crescita e lavoro	Industria generica, in particolare SME	In maggioranza pubblico tramite prestiti e detrazioni tasse	Circa 10 miliardi di euro	Forniti prestiti a più di 800 aziende, modernizzata la produzione di 3400 imprese
Regno Unito	HVM Catalyst (HVMC)	Supportare la manifattura, in particolare puntare a ricerca e sviluppo per una forte crescita	Migliorare la crescita della manifattura	Industrie e organizzazioni di ricerca	Finanziato in egual modo da pubblico e privato, si ha una condivisione delle entrate commerciali	164 milioni di euro investiti dal governo tra il 2012 e 2018, nel periodo 2015/2016 hanno avuto un'entrata di 79,7 milioni di € con una spesa di 61,3 milioni.	Per ogni euro speso dai fondi pubblici sono stati generati 17€

Tabella 2.8. Panoramica delle politiche 4.0 adottate da alcuni paesi europei [62]

2.2.2 Il caso italiano

Nel Settembre 2016, fu presentato a Milano il **Piano nazionale Industria 4.0** con l'approvazione dell'allora Presidente del Consiglio Matteo Renzi e Ministro dello Sviluppo Economico Carlo Calenda [63]. Questo è stato il primo piano (per ora anche l'unico) per la digitalizzazione dell'industria italiana verso il tipo di azienda, ormai delineato e già avviato da vari altri Stati, chiamato Industria 4.0. La durata dell'operazione fu prevista per l'arco 2017-2020. Durante il 2016, la situazione dell'industria italiana era già da tempo in declino, negli ultimi vent'anni ha perso il 21.6% degli impiegati nell'industria manifatturiera e il 20.1% della sua influenza sul PIL [64]. Le linee guida prefissate durante la presentazione erano:

- Agire in modo neutrale.

- Intervenire in modo orizzontale e non settoriale.
- Favorire l'evoluzione e l'acquisizione degli strumenti in modo da ottenere un salto tecnologico e garantire i benefici di un'adeguata strumentazione.
- Riuscire ad indirizzare le industrie italiane verso il progresso lasciando però il timone all'azienda stessa.

Possiamo quindi intuire che un punto cardine era la neutralità dello Stato nei confronti delle aziende, a prescindere dalla loro grandezza e settore. Si cerca di fare in modo che le aziende vengano aiutate dal governo in questa prima fase di sviluppo ma devono essere loro stesse a manovrare quest'evoluzione, ciò dovrebbe comportare il fatto che la società sarà in grado di migliorarsi anche dopo la fine del piano in modo che la crescita economica continui successivamente all'ausilio statale anziché fermarsi subito. Alla presentazione furono mostrate anche le strategie di manovra da attuare:

- **Investimenti.** Incentivare investimenti sia per i privati riguardo le tecnologie IoT sia per la ricerca, sviluppo e innovazione. Si decide un incremento da 80 a 90 miliardi di euro di incentivi privati nell'anno 2017 e, in particolare, altri 11.3 miliardi in più solo per le tecnologie 4.0 nel periodo 2017-2020.
- **Competenze.** Cercare di portare la cultura 4.0 a vari livelli d'istruzione: istituti tecnici superiori, università, dottorati. Vengono previsti 200.000 universitari, 3000 manager e 1400 dottorati di ricerca sul tema Industria 4.0.
- **Informazione.** Sensibilizzare sull'importanza dell'industria 4.0.
- **Infrastrutture.** Portare a termine il Piano Banda Ultra Larga in modo da garantire alta velocità della rete Internet e collaborare per definire gli standard e le linee guida per il mondo IoT. Il piano prevede il 50% delle aziende italiane coperte con una connessione a 100 Mbps entro il 2020.
- **Strumenti pubblici di supporto.** Supportare investimenti innovativi e rafforzare la posizione italiana nei mercati internazionali. Si prevede un incremento di 100 milioni di euro per le catene digitali di vendita secondo il piano **Made in Italy**.

Il piano entra in vigore nel 2017 e viene accolto molto bene dalle aziende, in particolare per l'iperammortamento al 250% (trattato più avanti). Mai lo Stato italiano aveva stanziato tanti finanziamenti privati per lo sviluppo tecnologico, fondi che riguardano tra il 90 e 99% delle aziende italiane [65]. Ecco l'elenco degli aiuti attuati dall'Italia per il piano [66], da *Iper e superammortamento a Startup e piccole e medie imprese (PMI) innovative* hanno come obiettivo l'innovazione, da *Fondo centrale di garanzia a Salario di produttività* la parola chiave è competitività:

- **Iper e superammortamento.** Questi servono per supportare economicamente le industrie italiane negli acquisti delle nuove strumentazioni. L'iperammortamento consiste in una supervalutazione del 250% sugli investimenti in chiave 4.0, il superammortamento equivale alla supervalutazione del 130% su qualsiasi strumento nuovo acquistato. In questo modo le aziende pagano una quantità minore di imposte, meno di quelle che avrebbero pagato senza aver compiuto investimenti [65].
- **Nuova Sabatini.** Serve per facilitare l'accesso al credito delle imprese e, come per l'elemento precedente, punta all'investimento di beni per la produzione. Il finanziamento deve essere di durata non superiore a 5 anni, di importo compreso tra 20000 e 2000000€ e deve essere interamente utilizzato per investimenti. Questo utilizza un tasso d'interesse del 2.75% o, nel caso di tecnologie 4.0, del 3.575% [67].

- **Credito d'imposta per Ricerca e Sviluppo.** Anche in questo caso l'aiuto è costituito da finanziamenti però, anziché colpire direttamente le aziende, sono indirizzati alla **Ricerca e Sviluppo** in modo da puntare su nuovi processi e prodotti che potranno poi essere applicati alle imprese. Consiste nel credito d'imposta del 50%, fino a 20 milioni €/anno, per la ricerca fondamentale, industriale e sviluppo sperimentale che possono concretizzarsi in: assunzione di personale altamente qualificato, contratti di ricerca con università, enti di ricerca, startup, strumenti di laboratorio.
- **Patent box.** L'agevolazione consiste in una tassazione agevolata sui beni immateriali quali brevetti industriali, marchi registrati, know how e software protetto da copyright in modo da rendere il mercato italiano più competitivo rispetto all'estero ed incentivare la collocazione in Italia dei beni immateriali anziché in altri paesi. Nella pratica è la riduzione delle aliquote IRES e IRAP del 50% dal 2017 in poi sulla licenza d'uso di beni immateriali.
- **Startup e piccole e medie imprese (PMI) innovative.** Questi finanziamenti sono indirizzati unicamente alle nuove imprese (startup) che puntano sull'innovazione in modo da favorire la ricerca e lo sviluppo delle aziende di domani. In generale si vuole difendere la cultura dell'innovazione. L'Italia si impegna a finanziare le startup aiutandole a diventare PMI senza detrarre questi benefici dopo la trasformazione e senza far pesare i debiti a lavoratori e manager in caso di fallimento dell'azienda.
- **Fondo centrale di garanzia.** Serve per sostenere le imprese che non dispongono di sufficienti garanzie per accedere al credito bancario. Lo Stato garantisce fino all'80% del finanziamento sia per utilizzo di liquidità sia per investimenti.
- **Aiuto alla crescita economica (ACE).** Rendimento nozionale del nuovo capitale proprio uguale a 2.3% nel 2017 e 2.7% dal 2018 in modo da rafforzare le strutture finanziarie delle aziende rendendole più competitive.
- **IRES, IRI e contabilità per cassa.** Ha l'obiettivo di ridurre la pressione fiscale per le imprese lasciando gli utili in azienda. Ciò si traduce nella diminuzione dell'IRES da 27.5% a 24% in modo da renderla più vicina alla media dell'Unione europea. Allo stesso modo l'IRI viene diminuito al 24%.
- **Salario di produttività.** L'idea è l'aumento dell'efficienza dei lavoratori e, allo stesso tempo, un incremento del loro salario. Oltretutto si cerca in essi maggiore responsabilità e dinamicità. Per questo motivo si propone una tassazione di vantaggio flat al 10% sui premi salariali dovuti ad un aumento della produttività aziendale per un massimo di 3000€.

Nel Settembre 2017, il ministro Calenda presenta i primi risultati del piano attuato confermando la sua continuazione per il 2018 e proponendo nuovi finanziamenti e aiuti; il progetto cambia nome diventando **Piano nazionale Impresa 4.0**. Durante l'evento sono anche stati mostrati i primi risultati del progetto [68], in figura 2.10 è possibile vedere l'andamento economico italiano nel 2017:

- Grazie alla *Nuova Sabatini*, l'iper e superammortamento, gli ordinativi delle aziende sono aumentati del 9% nel primo semestre rispetto lo stesso periodo del 2016.
- Con i contributi per la ricerca e sviluppo, molte aziende hanno iniziato a spendere in questo ambito. Su 68000 imprese intervistate, 24000 hanno dichiarato di voler usufruire dei benefici nel 2017. Tra queste, 11300 hanno aumentato la spesa per la ricerca del 10-15%, altri 4500 non avevano proprio un campo di ricerca nel 2016.
- Nel primo semestre gli investimenti in early stage sono aumentati del 2% rispetto al 2016 che è un risultato minore alle attese, per questo motivo con il nuovo piano si cercherà di eliminare le misure che non hanno avuto successo e proporre di nuove per cercare di farle decollare.



Figura 2.10. Andamento macroeconomico italiano nel 2017 [69]

Visti i buoni successi del 2017, Calenda decide di portare avanti il piano potenziandolo con maggiori incentivi. Gli obiettivi sono ora due: consolidare gli investimenti nell'innovazione in modo da proseguire l'andamento del 2017 e potenziare gli investimenti nel capitale umano e nelle competenze [69]. Il secondo intento comprende sia la formazione in fatto di scuola, università e ricerca sia in ambito lavorativo; per l'Italia è importante formare già da ora i nuovi lavoratori in modo che abbiano le adeguate competenze quando l'industria 4.0 sarà la normalità e non il futuro. In particolare si punta a potenziare gli istituti tecnici superiori, in Italia molto meno frequentati rispetto ad altri paesi dell'Unione europea (UE), in modo da formare 20000 studenti entro il 2020. D'altronde, come da figura 2.11, l'Italia è storicamente indietro nella digitalizzazione e specializzazione rispetto agli altri paesi dell'UE.

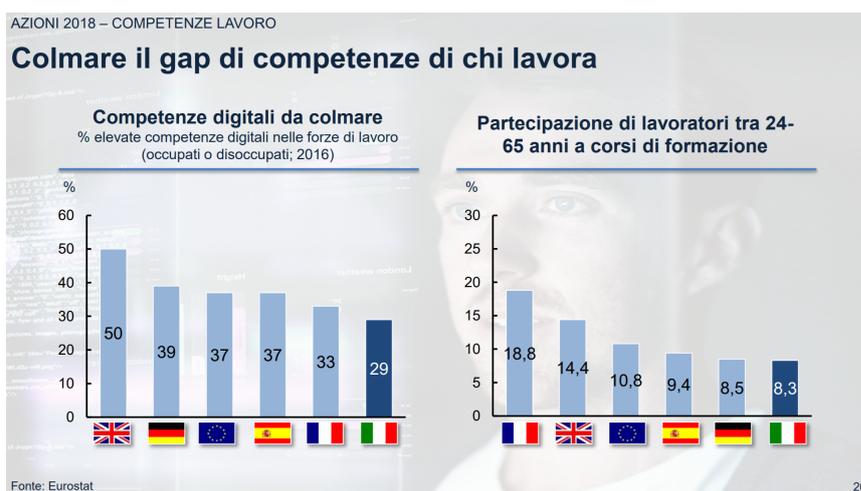


Figura 2.11. Formazione lavorativa nell'Unione europea nell'ambito 4.0 [69]

Nella pratica, vengono aggiunti 3 nuovi punti chiave per garantire una crescita nella formazione lavorativa [70]:

- **Accordi per l'innovazione.** Riguarda progetti per attività di ricerca industriale e sviluppo

sperimentale sempre aventi come fulcro l'innovazione. Ogni progetto deve avere una spesa da 5 a 40 milioni di euro, con una durata minore di 36 mesi. L'incentivo corrisponde a un contributo di almeno il 20% sui costi totali.

- **Credito d'imposta per formazione 4.0.** Serve per supportare i corsi di formazione dei lavoratori in modo da ridurre la mancanza di competenze digitali. L'incentivo copre il 40% dei costi fino ad un massimo di 300000€. Le attività dovranno riguardare ambiti quali marketing, informatica e tecnologie di produzione.
- **Fondo per il capitale immateriale, competitività e produttività.** Si cerca di accrescere la competitività e produttività del sistema economico nazionale in modo da rendere più agevole la crescita delle aziende nel mercato europeo e mondiale. Ciò si attua grazie a progetti portati avanti dall'Italia e concretizzati da soggetti sia pubblici che privati.

Naturalmente, essendo questa tesi scritta a metà 2018, non ci sono ancora dati ufficiali sull'impatto del nuovo piano. Nel maggio 2018, l'**Università degli studi di Padova** e la società **Considi** hanno pubblicato una panoramica a riguardo [71]. Le aziende possono essere divise in 4 fazioni in modo quasi omogeneo: i non interessati al 4.0 che sono ancora un numero sostanziale (27.5%), chi adotterà le tecnologie al più presto (22.3%), chi ha iniziato da poco con l'adozione del 4.0 (25.4%) e chi invece ha già apportato un'evoluzione importante al proprio impianto (24.5%). Circa il 74% punta a migliorare la produzione e l'efficienza, il 61% vuole affinare il servizio clienti, infine il 25% punta ad un nuovo modello di business. Allo stesso tempo in molti hanno iniziato la formazione dei propri dipendenti: il 51% di chi ha adottato tecnologie 4.0 sta cercando di rafforzare le competenze per l'innovazione, il 46% punta sul digitale e quasi il 70% pensa che una formazione continua sia necessaria per il continuo miglioramento dell'azienda.

2.3 Soluzione architeturale

Come descritto nel capitolo Piano Industrie 4.0, uno degli obiettivi del progetto **Industrie 4.0** è la definizione di un'architettura standard in modo da facilitare lo sviluppo senza dover studiare da zero ogni caso specifico d'impianto. Tutt'oggi non esiste una struttura ideale a cui tutti si riferiscono. Per ora il modello più autorevole è il **Reference Architectural Model for Industrie 4.0 (RAMI 4.0)** [72], ideato in Germania proprio per il piano **Industrie 4.0** e presentato alla fiera di Hannover del 2015 [73]. Il suo obiettivo principale è lo sviluppo di un linguaggio e cultura comuni in modo che tutti i diversi partecipanti possano capirsi facilmente tra loro e abbiano una visione d'insieme sul tema. La sua rappresentazione è visibile in figura 2.12.

I principali benefici di **RAMI 4.0** sono:

- Architettura basata sui servizi. Ogni azienda può implementare le proprie attività come meglio crede.
- Combina gli elementi fisici con il ciclo di vita di tutto il processo in modo da trovare i giusti punti di contatto tra teoria e pratica.
- I processi complessi sono divisi in sviluppi più piccoli e semplici in modo da poter essere gestiti al meglio.

RAMI 4.0 è sviluppato su 3 dimensioni, eccole nel dettaglio [72] [74] [73]:

- **Livelli di gerarchia.** Costituisce l'implementazione fisica dell'industria 4.0. Partendo dal basso si trovano gli oggetti più semplici e basilari e, andando verso l'alto, aumenta la complessità fino a coprire l'impianto e l'azienda stessa. La struttura deriva dall'**IEC 62264** (figura 2.13) che è un modello standard per l'industria 3.0. Quasi tutti i campi presenti nel vecchio metodo sono ripresi anche ora ma sfruttati sotto un punto di vista diverso. Secondo la

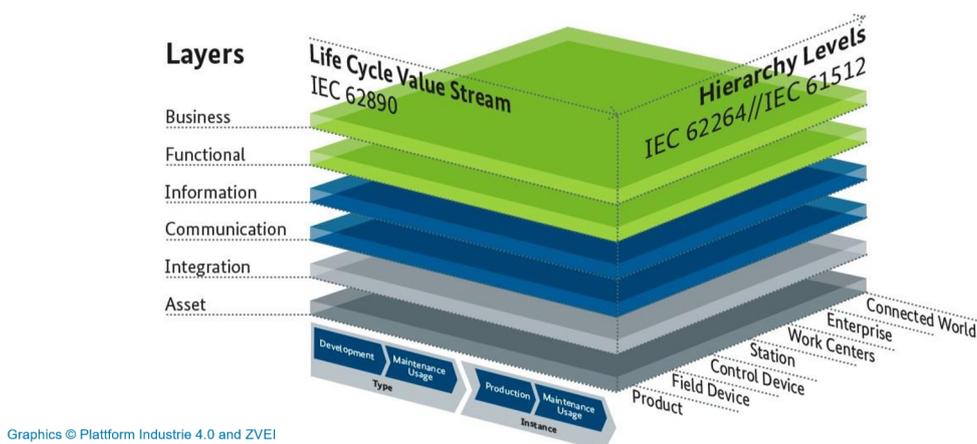


Figura 2.12. Architettura RAMI 4.0 [72]

concezione passata tutti i livelli erano isolati, la comunicazione era minima, l'efficienza era basata unicamente sulla potenza hardware e il prodotto finale non influenzava il processo da cui derivava. Con RAMI 4.0 tutti i dispositivi comunicano e collaborano tra loro attraverso una rete, quasi sempre Internet, senza vincoli di gerarchia. In questo modo si crea un sistema altamente flessibile e ricco di potenzialità. Oltretutto il prodotto finale continua a collaborare con il sistema. L'intero complesso può essere rappresentato più ad alto livello in soli 3 strati (figura 2.14): *Mondo connesso* che costituisce la rete e l'impianto di comunicazione, *Prodotti intelligenti* formato dal risultato finale della produzione e *Industria intelligente* che costituisce tutti i macchinari per svolgere l'elaborazione dalle materie prime alla soluzione finale.

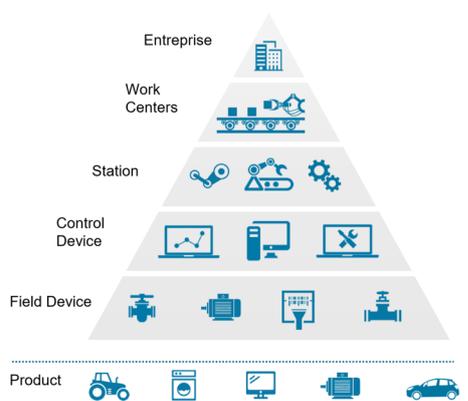


Figura 2.13. Gerarchia IEC 62264 [72]

- Ciclo di vita.** Basato su IEC 62890, rappresenta il ciclo di vita dei vari componenti e, in particolare, del prodotto finale. Quest'asse è composta da *Tipo* e *Istanza*. Con la prima si decide la tipologia del prodotto finale e, nella seconda, si producono le istanze dell'oggetto secondo le specifiche decise precedentemente. All'interno di *Tipo* si trova *Sviluppo* che è il lato teorico ed astratto, esso presenta le fasi di sviluppo, costruzione, simulazione, creazione di prototipi, test; in poche parole si percorrono tutti i procedimenti per definire il risultato finale. In *Istanza* è presente *Produzione* la quale è la scena più pratica, è legata alla creazione vera e propria del prodotto, acquisizione e salvataggio dati, associazione di ogni elemento con il suo numero seriale. Per entrambi gli elementi appena presentati, si aggiunge uno stato di

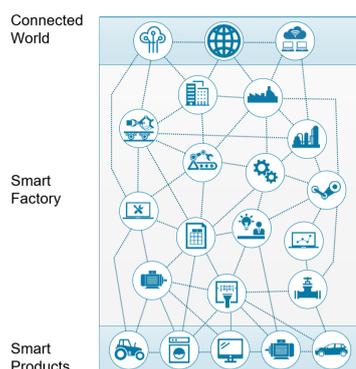


Figura 2.14. Gerarchia RAMI 4.0 [72]

Manutenzione e utilizzo che, nel primo caso, si compone in aggiornamenti software, manuali d'istruzioni, cicli di manutenzione e, per il secondo termine, rottamazione, riciclo, utilizzo, manutenzioni. Per una rappresentazione grafica, vedere figura 2.15.

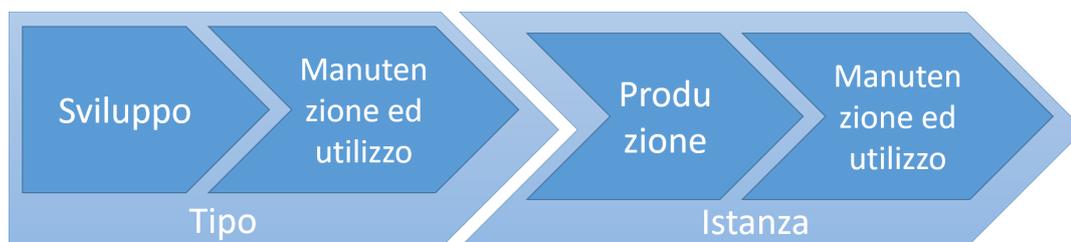


Figura 2.15. Ciclo di vita RAMI 4.0 [72]

- **Livelli** [75]. Essi rappresentano i processi aziendali, le funzioni, le informazioni e i benefici ricavati dai dati. Queste caratteristiche sono fortemente presenti per entrambi gli assi descritti sopra e presentano la forza dell'industria 4.0. Nello specifico sono (figura 2.16):
 - **Asset.** Sono i componenti fisici semplici che, assieme, compongono l'intero sistema. Per esempio: sensori, attuatori, parti meccaniche, documenti, eccetera.
 - **Integrazione.** È la collaborazione tra gli asset e il mondo informatico, la cooperazione tra hardware e software.
 - **Comunicazione.** Costituisce il dialogo tra tutti i vari dispositivi (fisici, digitali, meccanici, umani) a prescindere del loro livello.
 - **Informazione.** Accesso, visualizzazione, analisi e sfruttamento dei dati ottenuti dai vari dispositivi.
 - **Funzionalità.** Sfruttamento dei benefici di ogni asset e servizi offerti da ogni dispositivo.
 - **Azienda.** Realizzazione e concretizzazione dei processi aziendali.

È possibile notare che lo schema RAMI 4.0 è molto teorico, la motivazione va ricercata nel fatto che la Germania non vuole imporre un proprio schema fisso e determinato ai cittadini ma fornire solo una struttura su cui basarsi e discutere. Scenari più concreti vanno quindi ricercati in società private, in questo caso specifico verrà mostrata un'implementazione IBM [76]; i motivi per cui è stata scelta IBM sono già stati spiegati in Strumenti utilizzati. La struttura dell'applicazione è divisa in:



Figura 2.16. Livelli RAMI 4.0 [72]

- **Dispositivi al bordo.** I sistemi ciberfisici, i terminali e le applicazioni tramite i quali i lavoratori interagiscono con l'impianto appartengono a questa categoria. I gateway sono usati per integrare i dispositivi appena elencati e fare in modo che essi possano comunicare con l'impianto, questi gestiscono anche l'immagazzinamento locale e l'analisi dei dati. La sezione *Dispositivi al bordo* può autonomamente coordinare la comunicazione tra i suoi elementi senza l'ausilio degli componenti presentati sotto.
- **Impianto.** Questo governa i dati ed informazioni a livello di sistema industriale. Il suo scopo principale è il salvataggio, la gestione e l'analisi dei dati; ciò viene già in parte fatto nella sezione precedente ma qui viene svolto meglio poiché si possono sfruttare i dati derivanti da tutto l'impianto. È anche possibile accedere ad applicazioni terze d'ausilio. Solitamente vengono prese decisioni e feedback da rispedire ai dispositivi semplici in modo da ottimizzare il lavoro e la collaborazione. Avendo una visione d'insieme più alta è possibile gestire meglio le macchine e la loro collaborazione.
- **Azienda.** È composto dall'unione di tutti gli impianti di una stessa impresa. Questo si compone quasi degli stessi elementi dell'unità precedente. L'unica cosa che lo differenzia è il fatto che le decisioni possono essere prese ad un livello superiore e quindi, avendo un contesto più ampio, dovrebbero essere migliori di quelle stabilite inferiormente. A ciò si può aggiungere l'aiuto di sistemi aziendali e altre aziende partner.

In figura 2.17, è possibile visualizzare l'architettura proposta da IBM; si possono notare nell'immagine alcuni numeri in dei cerchi grigi, essi rappresentano i punti fondamentali del sistema, eccoli presentati più nel dettaglio:

1. Nell'ambito della produzione i dispositivi comunicano con i server per l'analisi dei dati tramite protocolli OPC, OPC UA, MQTT o TCP/IP. Lo stesso avviene per la comunicazione inversa.
2. I dati vengono analizzati e filtrati restituendo i risultati ottenuti, assieme ad un certo feedback, ai sistemi ciberfisici. In questo modo vengono ottimizzati i processi lavorativi.
3. I dati vengono inviati al settore *Impianto* tramite protocollo OPC UA o MQTT. Solitamente ciò avviene in formato JSON o XML.
4. I dati vengono salvati, normalizzati e, più in generale, "puliti". Anche qui avviene un lavoro di analisi però guardando tutto ciò ad un contesto più alto quindi i risultati dovrebbero essere migliori.

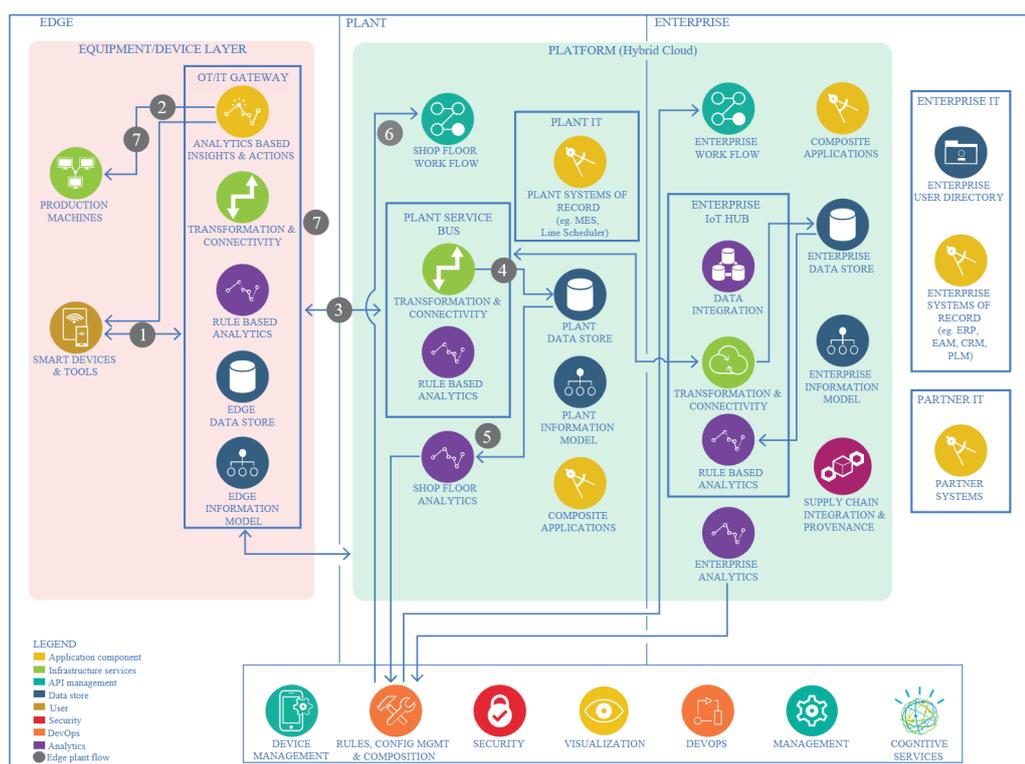


Figura 2.17. Architettura per industria 4.0 proposta da IBM [76]

5. In base ai dati ricevuti, vengono applicate certe regole e prese decisioni sulle azioni da effettuare. Per esempio si può inviare un avvertimento se la macchina sta lavorando in modo anomalo.
6. Se necessario, sono disponibili servizi a livello di impianto o azienda
7. Tutte le deduzioni e risultati ottenuti in *Impianto* e *Azienda* vengono inviati alle singole macchine. I dispositivi ricevono soprattutto comandi dall'alto in modo da ottenere ottimizzazioni migliori poiché decise ai livelli superiori.

2.4 Prognostic and health management

Il lato analisi nell'ambito industria 4.0 si basa sulla disciplina di **Gestione della prognostica e salute**, meglio conosciuta come **Prognostic and health management (PHM)** [77] [78]. Il suo obiettivo principale è effettuare previsioni su un certo evento futuro. In particolare si cerca di presagire anomalie, fallimenti, qualità del lavoro e comportamenti non ordinari di un generico dispositivo. Nel caso in cui si cerchi di prevedere i prossimi fallimenti in modo da schedulare al meglio le manutenzioni, si entra nel campo della manutenzione predittiva. Sempre più settori come industriale, aerospaziale, automobilistico stanno adottando questo approccio per diversi motivi, per esempio si può indovinare il guasto di un prodotto, prevedere la durata di vita per una batteria oppure riconfigurare una sonda spaziale in base alle previsioni sulla sua salute. La conoscenza anticipata di un particolare evento porta molti benefici quali sicurezza del sistema applicato, affidabilità, corretto funzionamento, qualità del prodotto finale, ottimizzazione del tempo e delle risorse, risparmio economico. Un elemento su cui si pone spesso grande interesse è il **Tempo di vita rimanente utile**, solitamente conosciuto come **Remaining useful time (RUL)**, che identifica quante unità di tempo o cicli macchina rimangono ad un dispositivo prima che questo subisca

un fallimento; di conseguenza le manutenzioni vengono schedate al meglio grazie alla conoscenza del futuro. Le previsioni sono dedotte attraverso lo studio degli errori passati (diagnostica) in modo da trovare un'impronta ridondante nel tempo per trarre deduzioni su ciò che accadrà (prognostica). Conseguenza diretta della conoscenza sugli eventi futuri di un certo strumento, l'uomo prende provvedimenti per migliorare la sua efficienza ottimizzando le prestazioni, ciò può essere applicato ad ogni macchinario all'interno di un impianto industriale. La PHM sta venendo sempre più utilizzata grazie alla continua digitalizzazione dei prodotti, quest'ultimi sono sempre più elettronici e ricchi di sensori in modo da poter essere analizzati secondo determinati parametri rendendo così possibile un loro monitoraggio. Questa disciplina può essere applicata in tempo reale o offline, il primo caso avviene solitamente per sistemi dove si vuole garantire sicurezza e quindi in caso di fallimento previsto si deve subito avvisare un umano incaricato, il secondo scenario si implementa se si vogliono ottenere predizioni più accurate ma esse non vengono subito riportate poiché si aspetta di raccogliere una maggiore quantità di dati per ottenere di conseguenza un'affidabilità migliore dal proprio output.

2.4.1 Manutenzioni

La PHM è legata a doppio filo con l'attività di manutenzione [79] poiché è proprio quest'ultima a garantire la buona salute di una macchina. La manutenzione è l'unione delle azioni che vengono effettuate su un dispositivo riguardo il suo ciclo di vita. Può anche essere definita come gli atti condotti in risposta ad un fallimento o riparazione precedentemente pianificata. Tradizionalmente esistono due tipi di manutenzione:

- **Correttiva.** Viene attuata dopo un fallimento; un oggetto viene utilizzato fino al suo guasto e dopodiché si interviene. L'approccio è detto di tipo reattivo poiché è conseguenza di un evento. Ciò comporta il massimo sfruttamento dell'oggetto in questione ma ne consegue inaffidabilità (perché non è conosciuto il tempo esatto del fallimento dell'oggetto) e problemi di sicurezza (una ruota di un'automobile scoppia durante l'utilizzo potendo causare un incidente stradale) portando un danno temporale ed economico causato dal fallimento inatteso. È la tipologia di manutenzione solitamente più costosa.
- **Preventiva.** Viene attuata precedentemente al guasto, può venir svolta ciclicamente nel tempo proprio perché non si conosce il momento del prossimo dissesto. Questa è migliore perché si basa su fatti reali; conoscendo la vita media di una macchina, si applica una manutenzione prima del fallimento stimato. Vengono così coperti i lati negativi della manutenzione correttiva ma si generano invece danni economici dovuti al fatto che vengono rinnovati componenti che potrebbero essere ancora validi ed utilizzabili. Oltretutto il fallimento può avvenire prima del giorno in cui era stata decisa la manutenzione ciclica incorrendo nel caso precedente. Lo scenario migliore si presenta quando la manutenzione preventiva viene effettuata appena prima del fallimento ma ciò sarebbe possibile solo conoscendo l'esatto momento dell'evento di guasto. Altri possibili problemi sono: ogni macchina lavora in modo diverso in base all'ambiente in cui si trova quindi la frequenza delle manutenzioni può cambiare, i produttori potrebbero non avere la stessa esperienza dei manutentori e voler massimizzare la vendita dei pezzi di ricambio.

Grazie alla PHM la manutenzione preventiva viene divisa in due rami come in figura 2.18:

- **Predeterminata.** Questa, rispetto alla definizione di tipo preventivo, aggiunge il fatto che le manutenzioni vengono schedate in modo ciclico a seconda del tempo medio tra i fallimenti del prodotto in esame. Come suggerisce il nome, le azioni di manutenzione vengono stabilite in modo predeterminato.
- **Predittiva o basata sulla condizione o predictive maintenance.** Le decisioni vengono prese in base al passato e presente della macchina, a seconda del suo andamento vengono



Figura 2.18. Tipi di manutenzione [79]

fatte previsioni sul futuro del dispositivo e si schedulano le manutenzioni di conseguenza. È basata sulla prognosi e diagnosi, descritte nella sezione Diagnostica e prognostica. La sua procedura è divisibile in 4 stadi come in figura 2.19:

- **Acquisizione dati.** Le macchine sono equipaggiate con vari sensori che effettuano costantemente misurazioni. Questi dati dovrebbero sintetizzare in valori numerici l'andamento della macchina, essi possono essere suddivisi a seconda che descrivano la condizione della macchina (temperatura, vibrazione, pressione) oppure rappresentino gli eventi accaduti (fallimento, installazione, diverso modo di operare, accensione, spegnimento). La loro qualità influenza, anche pesantemente, l'efficacia delle fasi successive. Essi vengono solitamente inviati e immagazzinati in un database in modo da poter essere consultati da una certa logica. Questo processo è descritto più tecnicamente nelle sezioni Internet delle cose (IoT) e Connessione impianto.
- **Analisi.** I dati acquisiti vengono analizzati per trarre delle informazioni utili. I dati dovranno venir filtrati ed elaborati in modo da ottenere informazioni attendibili. Il suo scopo è trarre deduzioni dall'input ricevuto, per esempio calcolare il RUL della macchina. Alcuni obiettivi sono: eliminare il rumore nelle misure dei sensori, filtrare i dati che non contengono informazioni utili, rappresentare graficamente i valori per gli utilizzatori, applicare un modello in modo da trovare una certa degradazione dei componenti, calcolare la vita rimanente della macchina. I dati possono essere acceduti come streaming oppure offline a seconda della reattività desiderata.
- **Decisione.** In base ai risultati dell'analisi si decide un determinato modo di operare per reagire al comportamento della macchina. Per esempio, se si sta rilevando un malfunzionamento è possibile spegnere l'attrezzatura o mandare un avviso ad una persona competente in modo da evitare o avvertire riguardo un determinato rischio, al contrario se il macchinario ha buone prestazioni potrebbe essere una buona scelta aumentare la sua velocità di produzione.
- **Implementazione.** Si attua la decisione presa precedentemente. Quest'ultimo processo è tutt'oggi applicato solo se si ha una buona affidabilità di tutti i passi precedenti, se infatti viene presa una decisione sbagliata si avrebbe un risultato opposto al desiderato. Bisogna quindi decidere quanta libertà di scelta deve essere affidata alla macchina e quali lavori devono venir svolti unicamente da esseri umani poiché troppo delicati. Perciò quest'ultima fase non è ancora molto sviluppata ma la maggior parte dei dispositivi si limitano ad inviare avvisi e allarmi, saranno poi esseri umani ad interpretare questi segnali. Solitamente vengono anche salvati dei report in modo che gli utilizzatori possano vedere le decisioni intraprese dalla macchina.

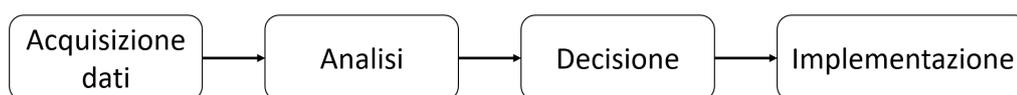


Figura 2.19. Fasi della manutenzione predittiva [79]

La prossima evoluzione sembra essere l'automanutenzione o manutenzione proattiva dove la macchina stessa riesce ad intuire il momento migliore per la riparazione e lo esegue in modo autonomo [80]. In figura 2.20 è possibile avere una panoramica delle manutenzioni nel tempo.

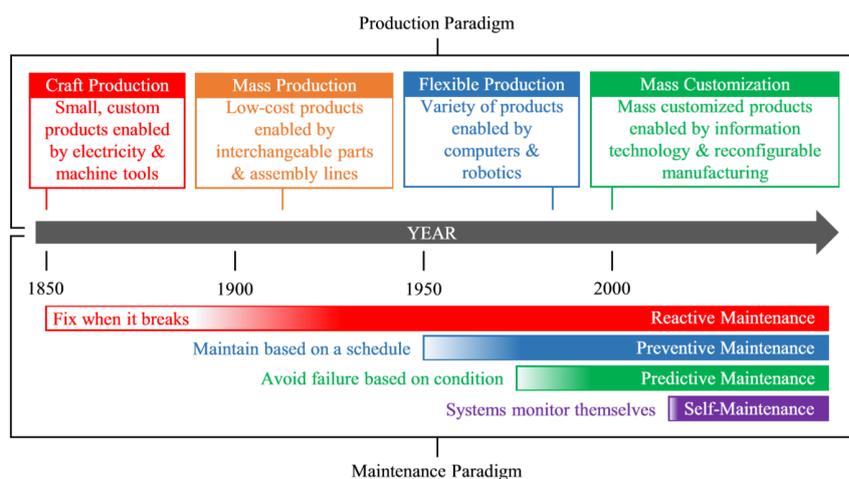


Figura 2.20. Tipi di manutenzione nel tempo [80]

2.4.2 Diagnostica e prognostica

Lo studio della PHM si compone in diagnostica e prognostica [81] [106]. Entrambe queste dottrine derivano dall'ambito sanitario e molti dizionari presentano le loro definizioni unicamente applicate alla medicina, però lo stesso significato può essere trasposto in ambito industriale, sostituendo il paziente con la macchina e il medico con l'operaio [78]. Secondo l'enciclopedia **Treccani**, la diagnostica è definita come

“Complesso di dottrine e tecniche, manuali, strumentali e di laboratorio, tese a formulare la diagnosi” [83]

quindi

“Nel linguaggio tecnico, la diagnosi è frutto dell'analisi di malfunzionamenti, guasti ed errori in apparati, sistemi o programmi di calcolo, che porta a individuare il punto in cui questi si sono verificati e a identificarne il tipo” [84]

Nell'ambito dell'industria 4.0, è il processo per determinare le cause, la dimensione, il tempo e, in generale, più informazioni possibili riguardo un fallimento dopo che questo è accaduto in modo da categorizzarlo. Si cerca di trovare dei parametri comuni che causano un determinato tipo di fallimento in modo da classificarli il meglio possibile. La categorizzazione è il fine ultimo della diagnostica in modo da poter accomunare tutti i possibili comportamenti di una macchina. Questa fase sembra in contraddizione con la manutenzione predittiva che cerca di predire l'insuccesso prima

che accada ma in realtà è un passo necessario per arrivare alla prognostica; un approfondito studio di tutte le possibili casistiche e loro cause permette di predire ciò che accadrà in futuro proprio poiché si ha la conoscenza del passato.

L'enciclopedia **Treccani** definisce la prognostica solo in ambito medico e riporta

“Relativa alla prognosi” [85]

di conseguenza la definizione di prognosi è

“Previsione sull'ulteriore decorso e soprattutto sull'esito di un determinato quadro morboso in esame” [86]

ma lo stesso concetto è possibile proiettarlo nel quadro di una macchina anziché di un paziente umano. È quindi la disciplina che si pone l'obiettivo di dedurre predizioni in base ad uno stato passato o presente. I suoi risultati permettono di arrivare al tempo di vita rimanente della macchina così da schedulare le manutenzioni. La prognostica può essere svolta tramite vari approcci secondo la figura 2.21.

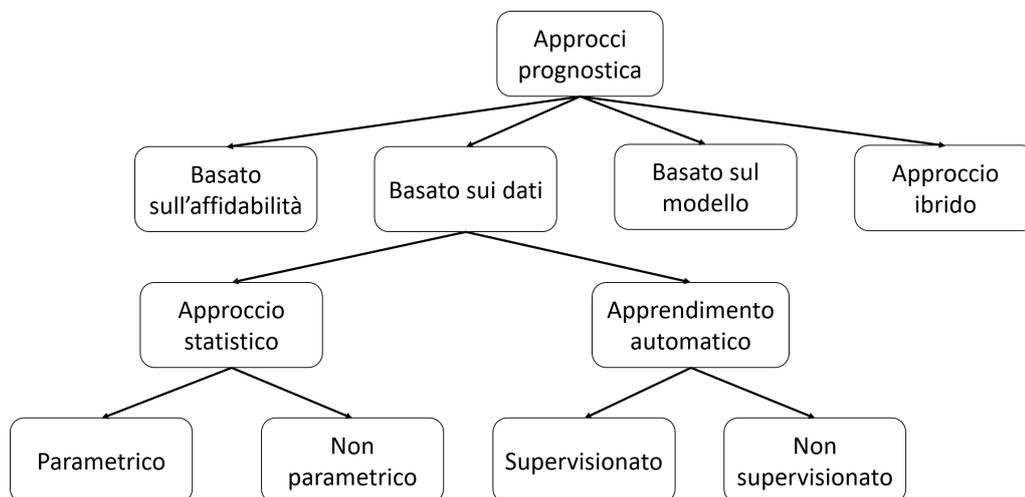


Figura 2.21. Approcci della prognostica

- **Basato sull'affidabilità.** Questo è strettamente collegato alla manutenzione predeterminata descritta nella sezione Manutenzioni. Si basa sulla statistica applicata ad una grande mole di dati ed è solitamente fornita dal produttore della macchina. Molte macchine vengono utilizzate e si registrano le loro condizioni, infine si elaborano delle statistiche che vengono riportate nel manuale del macchinario. Per esempio, si calcola il tempo di vita medio di un dispositivo e, in base a ciò, il produttore consiglia all'utilizzatore di effettuare delle manutenzioni cicliche. Il metodo è semplice ma, allo stesso tempo, non particolarmente efficace poiché basato su un approccio solamente statistico.
- **Basato sui dati.** Questi vengono considerati come delle scatole nere perché non necessitano una conoscenza e dei modelli specifici solo per il campo in esame. Ciò è un pro e un contro, ne consegue che un modello può venir utilizzato a prescindere dal tipo di dati analizzati ma allo stesso tempo questo non sarà ottimizzato per la specifica funzione trattata e quindi potrebbe avere prestazioni minori rispetto una soluzione ad hoc; allo stesso tempo è possibile analizzare un sistema molto complesso senza doverlo conoscere in modo approfondito. L'efficacia di questo approccio consiste nella quantità e qualità dei dati utilizzati ma non sempre è facile averli e garantire una certa bontà, oltretutto è necessario ottenere un certo

numero di fallimenti visto che sono proprio questi l'obiettivo dell'analisi. Di conseguenza ciò potrebbe diventare un grosso svantaggio se non si riesce ad ottenere un giusto insieme di dati da utilizzare, è anche possibile che la macchina non sia proprio provvista di sensori dai quali trarre misurazioni. È diviso in approccio statistico e apprendimento automatico.

- **Approccio statistico.** È basato sull'analisi delle proprietà statistiche dei dati. Vengono fatte alcune osservazioni iniziali che verranno poi ricalcolate con i nuovi dati in modo da essere comparate. Questo modo operandi è semplice e veloce però necessita delle statistiche iniziali molto accurate perché da esse verranno dedotte le anomalie, se queste sono sbagliate anche tutte le predizioni saranno alterate. Può essere ulteriormente diviso in parametrico e non parametrico.
 - * **Parametrico.** I valori calcolati sono proprietà tipicamente statistiche come media, varianza, minimo, massimo. Vengono calcolati alcuni parametri statistici quando la macchina è nel suo stato di corretto funzionamento. Una volta ottenuti nuovi dati, si ricalcolano i parametri e li si comparano con quelli della macchina normalmente funzionante, se questi differiscono troppo allora sta avvenendo un comportamento anomalo. Solitamente seguono alcune distribuzioni tipiche della statistica come la normale, esponenziale, Weibull. Questi numeri rappresentano l'andamento corretto della macchina e il suo stato ordinario. In base alla distanza tra i valori precalcolati e nuovi si può dare una stima del RUL.
 - * **Non parametrico.** Questo metodo è più flessibile perché non si basa su una distribuzione statistica specifica. In molti casi l'approccio non parametrico è necessario perché spesso la realtà non segue modelli matematici perfetti. Un esempio è il **Nearest neighbour (NN)** che viene usato per stimare la funzione di densità dei dati. Questo associa i valori in base alla loro distribuzione, infatti i dati derivanti da un comportamento corretto saranno la moltitudine e molto vicini poiché la macchina passa la maggior parte del tempo comportandosi come previsto e assumendo le stesse caratteristiche. Invece le misurazioni rilevate durante uno stato anomalo saranno lontani dalla massa venendo così rilevati dall'algoritmo.
- **Apprendimento automatico o machine learning.** Questo termine non ha una definizione precisa ma viene spesso definito con le parole usate da Arthur Samuel

“Apprendimento automatico: campo di studio che dona ai computer l'abilità di imparare senza essere stati esplicitamente programmati” [87]

(Arthur Samuel)

Può venir usato per effettuare delle predizioni grazie a modelli predittivi oppure classificare dei dati in comportamento normale o anomalo. Può essere suddiviso in supervisionato e non supervisionato.

- * **Supervisionato.** Se si conoscono tutte le possibili classificazioni o etichette del risultato finale allora l'approccio di apprendimento automatico è detto supervisionato. Il suo scopo è, dato un input, predire il corretto output. Per esempio, viene utilizzato quando si vuole predire un comportamento corretto oppure anomalo quindi conosciamo già tutti i possibili output (stato normale, stato alterato). Alcuni dei più famosi algoritmi sono: alcune reti neurali, macchine a vettori di supporto (dette anche support vector machine), reti bayesiane.
- * **Non supervisionato.** Se non si conoscono tutti i possibili allora si tratta di apprendimento automatico non supervisionato, l'obiettivo sarà proprio definire le classificazioni. Questo tipo di processi viene utilizzato proprio per trovare un'associazione tra i vari dati e raggrupparli secondo parametri comuni. Nel caso della PHM, gli algoritmi non supervisionati sono usati per catalogare i dati in normali o anomali. Una volta avvenuta questa suddivisione, è possibile applicare ulteriori tecniche per effettuare predizioni. Alcuni algoritmi di questo genere sono: alcune reti neurali, analisi delle componenti principali (più conosciuto come principal component analysis), filtro di Kalman, algoritmi Monte Carlo sequenziali (detto anche particle filter).

- **Basato sul modello (detto anche basato sulla fisica).** Questo ha come cardine la conoscenza in profondità del sistema in esame, è importante conoscere approfonditamente non solo il funzionamento della macchina ma anche le sue condizioni in stato normale e anomalo, come vengono alterati i cicli macchina a seconda del suo stato, svolgimento e cause dei fallimenti. È quindi necessario studiare i modelli matematici e fisici con le quali una macchina svolge il proprio lavoro. Grazie a ciò si ottengono dei risultati migliori poiché ottimizzati per uno specifico ambito ma, allo stesso tempo, il medesimo lavoro non può venir riutilizzato per più scenari. Il sistema da studiare potrebbe essere molto complicato quindi la procedura di analisi può comportare molto tempo e fatica. La sua efficienza è dovuta al fatto che i componenti di una macchina si degradano proprio seguendo le leggi fisiche quindi la predizione sarà molto accurata. La precisione è strettamente dipendente al modello di apprendimento applicato, se la funzione matematica usata descrive fedelmente il lavoro del macchinario allora si avrà un'elevata accuratezza, al contrario si potrebbe ottenere un risultato scarso. Come già detto, molti sistemi sono complessi di conseguenza non è facile trovare il modello fisico adeguato, questo approccio è usato solo se si vuole un'alta meticolosità del risultato nonostante il tempo, costo e sforzo che sarà necessario per raggiungerlo.
- **Approccio ibrido.** Come suggerisce il nome, è la fusione di due o più tecniche descritte sopra. Ogni processo presentato ha delle limitazioni e punti di forza, per questo motivo è utile unire più metodi di lavoro in modo da sfruttare al meglio i loro benefici a seconda dei casi. L'esempio più semplice è unire un approccio basato sui dati e uno sul modello in modo da avere un'ottima precisione riguardo i componenti semplici da analizzare e una buona predizione nelle parti più complesse che richiederebbero troppo lavoro e costo.

Ecco una comparazione degli approcci della prognostica nella tabella 2.9 e in figura 2.22.

	Basato sui dati	Basato sul modello
Vantaggi	<ul style="list-style-type: none"> • Semplicità • Lo stesso algoritmo è applicabile in diversi scenari • Basso costo 	<ul style="list-style-type: none"> • Alta precisione • Approccio deterministico • Possibilità di poter predire diversi tipi di degradazione
Svantaggi	<ul style="list-style-type: none"> • Necessita molti dati riguardo i fallimenti • Dato uno stesso input potrebbe restituire diversi output • Minore precisione 	<ul style="list-style-type: none"> • Necessita il calcolo di un modello matematico • Alto costo • Difficile da applicare per sistemi complessi

Tabella 2.9. Confronto tra gli approcci della prognostica [88]

2.4.3 Architettura PHM

La PHM non presenta dei componenti specifici come nel caso dell'industria 4.0 ma fornisce solo certi elementi ad alto livello da implementare concretamente senza grandi vincoli. La sua composizione è costituita quasi totalmente da elaborazioni software, l'unica parte hardware è dovuta all'acquisizione, salvataggio e comunicazione dati che necessitano sensori e database. L'esecuzione delle applicazioni software avviene su uno o più server. Siccome la tesi si pone l'obiettivo di presentare

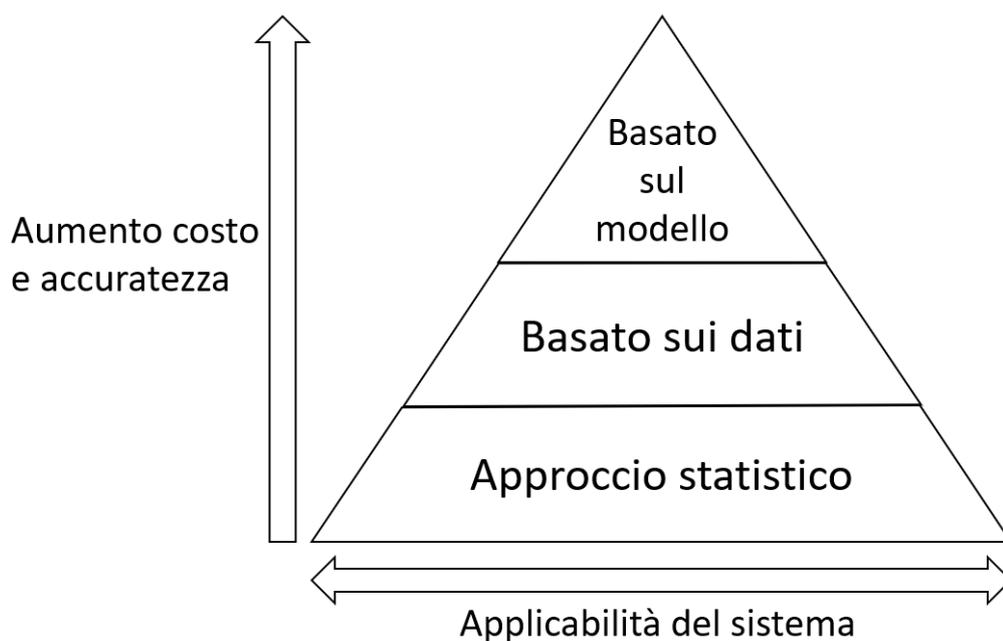


Figura 2.22. Comparazione approcci della prognostica [89]

la PHM nell'ambito industria 4.0, alcuni argomenti vengono trattati nel dettaglio in altre sezioni; i sensori utilizzati sono dispositivi IoT, già presentati in Internet delle cose (IoT), le macchine sono collegate al database secondo certi protocolli, un esempio è spiegato in Connessione impianto, il lato base di dati è strettamente legato al fenomeno **big data** che verrà introdotto in PHM e big data, i sistemi ciberfisici sfruttano la PHM per autoconfigurarsi come riportato nella sezione PHM applicata ai sistemi ciberfisici.

Si presenta ora in figura 2.23 la struttura funzionale della PHM [90] [91] [92] [93]. Essa mostra ad alto livello i blocchi essenziali che costituiscono la PHM, ognuno di essi rappresenta una funzionalità necessaria per arrivare ad un'architettura completa. La maggior parte di essi vengono implementati tramite software.

I moduli dell'architettura sono:

- **Sorgente dati.** Questa può essere un qualsiasi dispositivo. Può venir identificato come una qualunque macchina con associati vari sensori che calcolano varie misurazioni in modo da ottenere lo stato della macchina in un preciso momento. I valori ottenuti vengono poi salvati sul database ed inviati al modulo acquisizione dati in modo che possa iniziare il processo da parte della PHM. Alcuni esempi di sorgente dati sono: macchina o impianto industriale, aereo, automobile, drone, server.
- **Database.** Serve per immagazzinare i vari dati necessari agli altri moduli. Oltre a salvare le misurazioni provenienti dal dispositivo sorgente, registra anche le manutenzioni, caratteristiche e configurazione della macchina, eventuali report e, in generale, tutte le informazioni che possono risultare utili per la PHM.
- **Modulo acquisizione dati.** Ha la funzione di ottenere e caricare i dati necessari per i moduli successivi. I valori ricevuti vengono mandati al *modulo processamento dati*.
- **Modulo processamento dati.** I dati vengono elaborati secondo certi criteri. È possibile che i dati ottenuti siano grezzi e debbano venir modificati, filtrati, aggregati e, in modo più generico, manipolati in modo da poter essere utilizzati senza problemi dai successivi moduli.

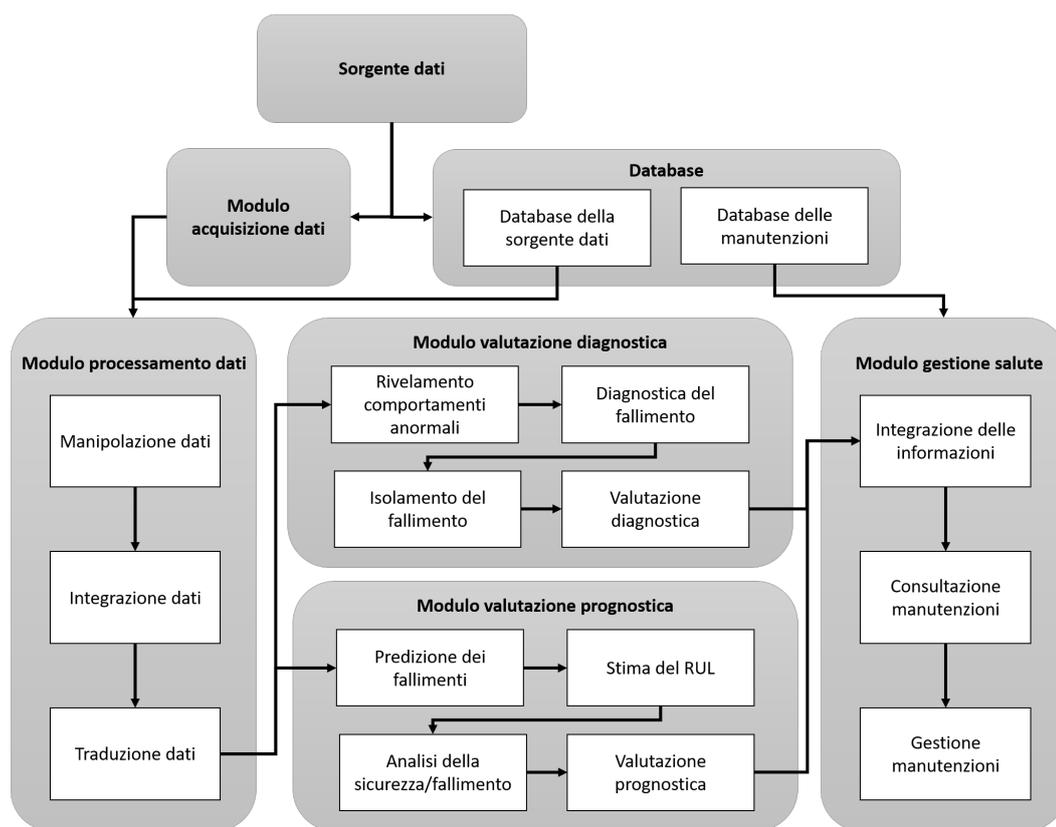


Figura 2.23. Architettura PHM [90]

In questa fase è possibile che vengano estratte o dedotte delle informazioni dai valori ottenuti in modo da generare ulteriori criteri per l'applicazione della PHM. Una volta che i dati sono stati trattati, vengono inviati ai *moduli di valutazione diagnostica e prognostica*.

- **Modulo valutazione diagnostica.** Ha lo scopo di applicare il metodo diagnostico ai valori ricevuti. Il modulo rileva i comportamenti anomali e ricerca la loro motivazione in modo da trovare la connessione causa-effetto. Una volta fatto ciò cerca di classificare il problema e trovare una sua *“impronta”* in modo da poterlo riconoscere in futuro e classificarlo. Il frutto del suo lavoro viene inviato al *modulo gestione salute*.
- **Modulo valutazione prognostica.** Questa è la parte che effettivamente produce le previsioni per conoscere lo stato futuro della macchina e gestire quindi al meglio il sistema. Il modulo applica dei modelli di apprendimento per stimare il RUL, i prossimi fallimenti e, in maniera più generale, delle previsioni sul futuro del dispositivo. Il risultato ottenuto viene inviato al *modulo gestione salute*.
- **Modulo gestione salute.** Esso riceve varie informazioni dal *modulo valutazione diagnostica* e *modulo valutazione prognostica*, in base ad esse decide come configurare il sistema. Per esempio può decidere di ignorare il tutto, schedulare delle manutenzioni, aumentare o diminuire la produzione. Le decisioni vengono prese studiando sia gli output dei due moduli precedenti sia i dati conservati nel database quali report e informazioni sulle manutenzioni passate.

Dall'elenco puntato appena presentato e dalla figura 2.23 possiamo notare che la PHM si concretizza con l'applicazione della diagnostica e della prognostica, gli altri elementi servono per supportare o limare queste due discipline. Come si vedrà nella sezione Tipico piano di lavoro, il lavoro da svolgere per uno sviluppatore è molto simile alle funzioni dei moduli appena presentati.

Si vuole ora presentare un'implementazione pratica dell'architettura appena presentata. L'applicazione [94] [95] è fornita da Microsoft Azure e la sua struttura è rappresentata in figura 2.24. Il programma vuole mostrare una soluzione per la manutenzione predittiva nel settore aerospaziale usando come sorgente un'ipotetico motore di un aereo il quale viene virtualizzato.

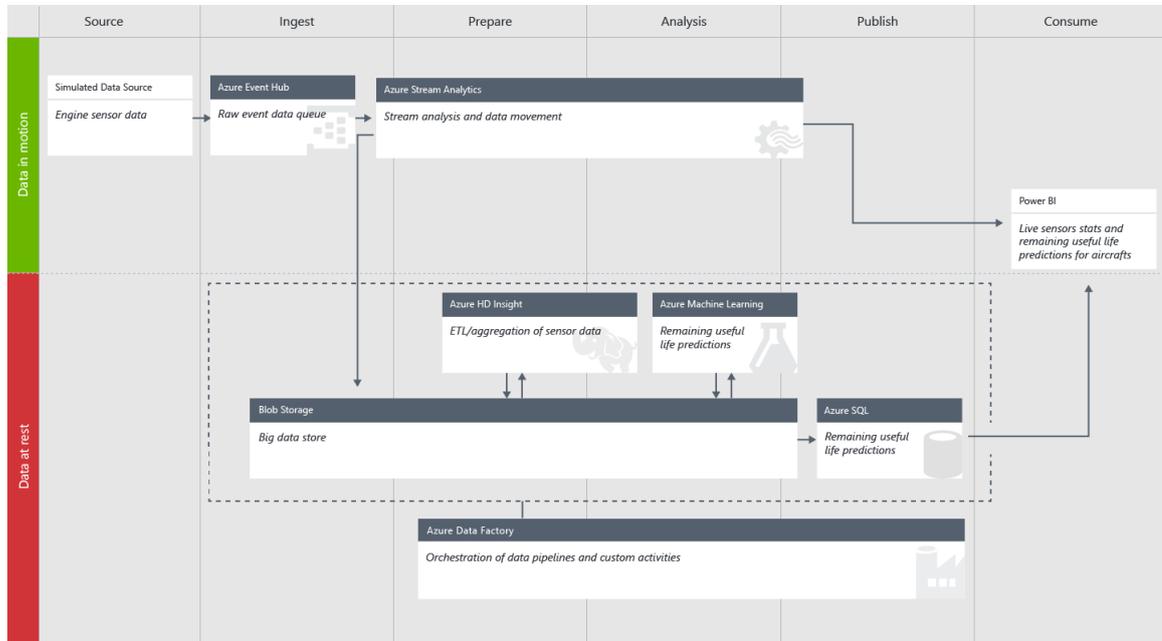


Figura 2.24. Soluzione Cortana Intelligence per la manutenzione predittiva nel settore aerospaziale proposta da Microsoft Azure [95]

I suoi componenti sono:

- **Simulated Data Source.** La sorgente dei dati è simulata ma si avrebbe lo stesso comportamento con un dispositivo fisico. Essa rappresenta virtualmente un motore di un aereo.
- **Azure Event Hub.** È un blocco per gestire gli eventi. In questo caso accetta i dati ricevuti dalla sorgente e li invia a **Azure Stream Analytics** e all'elemento **Azure Data Factory**. Questo blocco corrisponde al *modulo acquisizione dati* spiegato precedentemente.
- **Azure Stream Analytics.** Effettua delle analisi in tempo reale sui dati che riceve periodicamente. Essendo che vengono effettuate in poco tempo e ci si basa solo sui valori ricevuti in quel momento, le predizioni non sono altamente accurate ma sono tempestive. I risultati sono inviati e visualizzati nel **Power BI**.
- **Azure Data Factory.** Serve per gestire più dati in sequenza e determinate attività. Coordina gli elementi **Blob Storage**, **Azure HD Insight**, **Azure Machine Learning** e **Azure SQL**.
- **Blob Storage.** Serve per immagazzinare i dati ricevuti in modo da avere un certo insieme da poter analizzare.
- **Azure HD Insight.** Questo esegue alcuni script per la gestione dei dati come aggregazione, filtraggio, modifiche per far in modo che l'input sia più facilmente analizzabile dai blocchi successivi. Svolge lo stesso lavoro del *modulo processamento dati* dell'architettura precedente.
- **Azure Machine Learning.** Produce predizioni, tra cui il RUL, sullo stato futuro della sorgente in base ai dati ricevuti. Lavora come il *modulo valutazione prognostica*.

- **Azure SQL.** È usato per salvare in un database i risultati ottenuti da *Azure Machine Learning*. In questo modo l’output può essere acceduto e mostrato da *Power BI*. Ciò dimostra l’utilizzo di un database come nell’architettura presentata sopra anche se questo non viene introdotto esplicitamente.
- **Power BI.** Tramite un’interfaccia grafica mostra i risultati prodotti in modo che l’applicazione possa facilmente essere utilizzata dagli utenti. Grazie alle informazioni ottenute i tecnici possono schedulare al meglio le manutenzioni ed ottimizzare il loro lavoro.

Si può quindi notare una correlazione tra i blocchi dell’architettura precedente e la soluzione appena presentata anche se quest’ultima non presenta i moduli legati alla diagnostica e alla salute. A volte l’elemento sulla salute non viene riportato perché non tutti i dispositivi sorgente possono essere autoconfigurati sul momento e il suo sviluppo dipende direttamente dalla macchina a cui verrà applicato; probabilmente il blocco non è stato inserito per semplicità. Se ne deduce quindi che il primo schema può essere implementato nella pratica.

2.4.4 PHM e big data

Nella sezione precedente è stato detto che serve un dispositivo per il salvataggio dei dati ricevuti ovvero un database. Non si è però entrati nel dettaglio nella sua descrizione nonostante i valori che esso contiene siano la benzina della PHM. I database utilizzati in quest’ambito e per l’industria 4.0 sono di tipo **big data**, già presentati nella sezione Big data dove erano stati tralasciati i benefici poiché più adatti ad essere presentati ora. Ecco le motivazioni che rendono la collaborazione tra PHM e **big data** migliore rispetto ad un database di tipo relazionale [53]:

- **Big data offrono un grande campione statistico comportando risultati migliori dall’analisi.** La maggior parte degli strumenti ed algoritmi per l’analisi hanno bisogno di molti dati e, in generale, le previsioni diventano più accurate maggiore è la quantità di input ricevuto. Ciò rende i **big data** adatti per la situazione, al contrario dei database basati su SQL.
- **La tecnologia odierna può ora gestire i big data.** Inizialmente il problema era particolarmente legato ad hardware poco performante, rendendo le varie operazioni troppo lente a causa della mole di istruzioni da eseguire.
- **L’economia odierna permette di trattare molti dati senza grandi spese.** L’abbassamento del prezzo di dispositivi di memoria e l’aumento della velocità della rete Internet rende accessibile l’analisi di **big data** anche da parte di piccole/medie aziende.
- **Si può imparare molto dai dati disordinati, tanto quanto la loro dimensione.** Le nuove analisi e scoperte avvengono proprio analizzando dati che, precedentemente, erano considerati semplicemente confusionari ma che, tramite nuovi metodi, nascondono informazioni importanti.
- **Analisi su grandi campioni di dati rivela e genera cambiamenti economici.** Le scoperte e previsioni ottenute tramite i **big data** permettono di predire l’andamento economico della propria azienda migliorando le decisioni intraprese.

Ora osserviamo il lavoro condotto da TDWI [53] tramite varie interviste per dedurre i benefici prodotti, nel mondo reale, dall’utilizzo dei **big data**. È stato chiesto a degli utenti “*Quali dei seguenti benefici conseguirebbe la tua azienda implementando un’analisi sui big data?*”, i punti fondamentali ricavati dalle risposte sono (figura 2.25):

- **Analisi basate sui clienti.** Sono possibili indagini sugli utenti di una certa azienda migliorando i propri servizi agli occhi del pubblico, i principali campi sono: miglioramento

della targetizzazione nei social network, segmentazione in gruppi della propria utenza in modo da presentare servizi più adatti ad un determinato gruppo, riconoscimento di sconti e opportunità di mercato.

- **Gestione aziendale.** Si possono ricavare previsioni e pianificazioni riguardo il mercato e i suoi cambiamenti.
- **Applicazioni specifiche di analitica.** Queste trattano argomenti strettamente legati alle previsioni come rilevamento di frodi, quantificazione dei rischi, andamento del mercato eccetera.



Figura 2.25. Quali dei seguenti benefici conseguirebbe la tua azienda implementando un'analisi sui *big data*? (Max 5 risposte) Basato su 1623 risposte di 325 persone, 5 risposte per utente, in media [53]

Sempre TDWI [53], ha anche domandato ai propri intervistati “Nella tua organizzazione, quali sono le principali barriere per implementare l'analisi dei *big data*?”. Ecco i risultati (figura 2.26):

- **Staff e capacità inadeguate.** I lavoratori non hanno abbastanza conoscenze per poter gestire questo nuovo tipo di analisi, soprattutto per il fatto che è un campo abbastanza recente. Oltretutto l'azienda non cerca di istruire i propri dipendenti in tal ambito.
- **Mancanza di supporto a riguardo da parte dell'azienda.** L'argomento non viene proprio trattato o pubblicizzato a livello aziendale.
- **Problemi con il software usato per i database.** Quest'ultimo spesso non sopporta il lato analisi, ha problemi di scalabilità oppure è troppo lento per gestire una mole di dati.

2.4.5 PHM applicata ai sistemi ciberfisici

Come già detto nella sezione Sistemi ciberfisici, i sistemi ciberfisici sono migliori rispetto ai mezzi sfruttati dall'industria 3.0 poiché possono essere riprogrammati, comunicano tra loro, raccolgono misurazioni e possono prendere decisioni in base al contesto attuale. La PHM si occupa proprio di creare informazioni utili dai dati grezzi ed effettuare delle predizioni in modo da configurare le macchine di conseguenza. Si ottiene così il beneficio di una macchina autonoma che cambia le proprie impostazioni sul momento senza la necessità di un intervento umano, oltretutto alcune decisioni dovrebbero essere intraprese con maggior coscienza poiché il dispositivo stesso conosce una mole di dati non analizzabili facilmente da un umano.

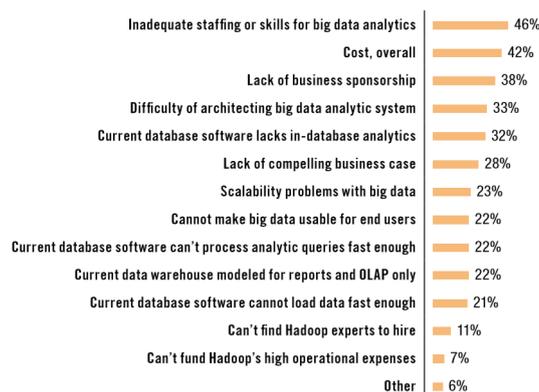


Figura 2.26. *Nella tua organizzazione, quali sono le principali barriere per implementare l'analisi dei big data?* (Max 5 risposte) Basato su 1153 risposte di 325 persone, 3,5 risposte per utente, in media [53]

La PHM applicata ai sistemi ciberfisici può essere schematizzata in 4 livelli [96] [97] (questi seguono la struttura presentata in Architettura PHM e l'architettura 5C dei sistemi ciberfisici descritta nella sezione Sistemi ciberfisici):

- **Acquisizione dati.** I dati vengono principalmente ottenuti dalle misurazioni dei sensori montati sulla macchina. Sono anche di notevole importanza i valori ricevuti dagli altri macchinari nell'impianto, in questo modo è possibile conoscere lo stato dell'impianto e trarre vantaggi dalla collaborazione dei dispositivi.
- **Processamento dati.** I dati grezzi vengono manipolati e modificati in modo da generare informazioni utili, vengono considerate sia le misurazioni dei sensori sia i dati ricevuti dalle altre macchine e dall'impianto totale.
- **Prognostica.** In base agli input ricevuti vengono effettuate varie predizioni sul futuro dell'attrezzatura e dell'intero sistema. Viene quindi applicata la prognostica sui dati ricevuti per cercare di ricavare informazioni sull'andamento dell'impianto in esame. Le decisioni verranno prese in base al suo output. Le predizioni vengono anche mostrate ai lavoratori in modo che possano osservare lo stato dell'impianto.
- **Decisioni post-prognostica e configurazione.** La macchina stabilisce, in base alle predizioni effettuate nel blocco precedente, come comportarsi e si autoconfigura a seconda delle sue scelte personali. Siccome anche gli utenti possono accedere alle predizioni, è possibile che alcune decisioni vengano prese dai lavoratori o avvengano lavori di manutenzione non eseguibili dalla macchina stessa. È largamente possibile che un dispositivo non riesca sempre a scegliere tra varie opzioni e sia allora indispensabile un intervento umano. Naturalmente si punta ad avere autoconfigurazioni sempre più ottimali fino ad arrivare al caso ideale in cui l'uomo è superfluo e la macchina è completamente autonoma.

In figura 2.27 è presentata la struttura appena descritta dei sistemi ciberfisici orientata in ambito PHM.

2.4.6 Benefici della PHM

Grazie alla predizione dello stato futuro di una macchina, gli utilizzatori possono ottenere molti vantaggi di diverso tipo [98]:

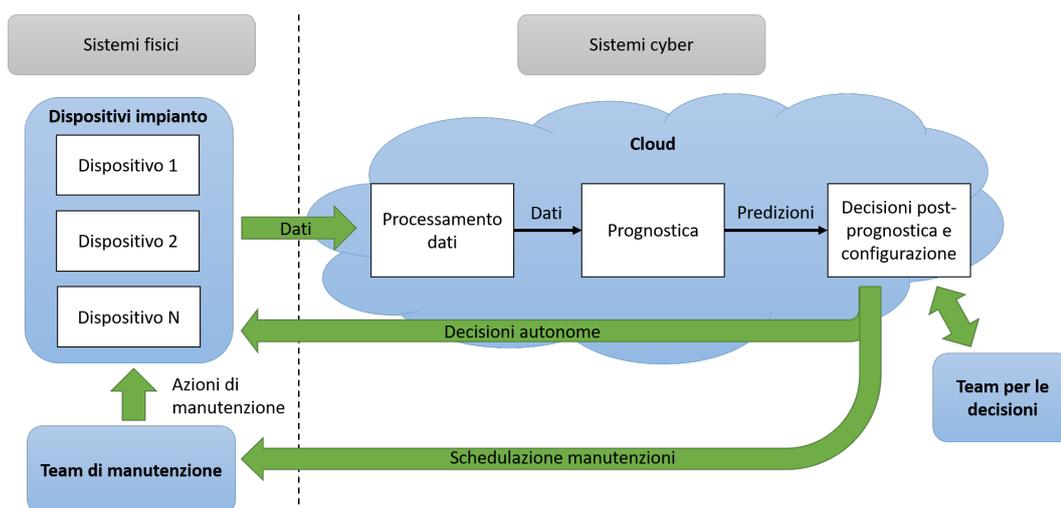


Figura 2.27. Struttura della PHM nei sistemi ciberfisici [96]

- **Affidabilità.** Essendo che il comportamento della macchina viene monitorato continuamente e può essere predetto il suo stato futuro, si garantisce con una certa accuratezza il normale funzionamento per un prossimo determinato intervallo di tempo. Si è quindi più sicuri del fatto che il macchinario porterà a termine il lavoro affidatogli.
- **Sicurezza.** Se una macchina lavora in modo non consono, questa può generare un pericolo per gli utilizzatori e l'ambiente. Un esempio sono gli aerei di linea, non riuscire a predire i malfunzionamenti di un loro qualsiasi componente comporterebbe un serio rischio all'incolumità di guidatori e passeggeri. Prevedere i fallimenti dà la possibilità di evitarli.
- **Migliore gestione delle manutenzioni e disponibilità.** Grazie allo sfruttamento nella manutenzione predittiva presentata in Manutenzioni, è possibile predire il prossimo errore o guasto quindi eseguire una manutenzione nel momento più opportuno. Di conseguenza, la macchina rimane il minor tempo possibile in uno stato non produttivo. Ciò è strettamente legato al lato economico.
- **Monitoraggio.** È possibile osservare l'andamento delle prestazioni di un dispositivo in modo da garantire la sua affidabilità, invecchiamento, qualità e agire di conseguenza, per esempio sostituendo le vecchie macchine con delle nuove o acquistare un nuovo modello se quello utilizzato precedentemente non rispetta le aspettative. Un elemento molto importante è la visualizzazione dell'andamento dell'impianto da parte del lavoratore che può così prendere decisioni migliori poiché ha una conoscenza più completa della situazione che lo circonda. Con l'utilizzo dei sistemi ciberfisici, essi stessi possono controllare il loro andamento e autoconfigurarsi in modo da ottimizzare il proprio operato e guadagnare autonomia da parte dell'uomo.
- **Qualità.** I dati ottenuti possono venire utilizzati per predire il pregio del prodotto finale elaborato da una macchina. In questo modo si possono effettuare dei lavori di riparazione prima che il risultato finale diventi inferiore a certe soglie. Si riesce così ad evitare gli sprechi dovuti ad oggetti prodotti che non rispettano certe caratteristiche imposte dall'azienda.
- **Economico.** I risparmi derivano da vari fattori: minore costo delle ispezioni, evitare perdite dovute al fatto che la macchina non può lavorare poiché aspetta una manutenzione, non sostituire pezzi ancora funzionanti, conoscere esattamente quale parte della macchina si sta guastando in modo da agire nel dettaglio, migliorare le azioni di manutenzione. Quest'ultime vengono schedulate al meglio in modo da non essere né correttive né preventive ma effettuate

esattamente quando servono. Naturalmente il guadagno economico dipende da vari fattori: installazione dei sensori nella macchina se non già provvista, accuratezza effettiva delle predizioni conseguite, casualità dei fallimenti (se questi sono puramente casuali non possono venir predetti), frequenza dei fallimenti nel tempo, ulteriore carburante o corrente elettrica dovuta al processo di manutenzione predittiva[99]. Nella figura 2.28 è raffigurato un generico andamento dei costi dei vari tipi di manutenzione descritti in Manutenzioni. Un esempio è dato da [100] che analizza il motore e la pompa idraulica di un elicottero AH-64 Apache. I costi della PHM considerati sono stati divisi in 4 tipi: sviluppo, prezzo del mezzo fisico, spese di manutenzione ed effetti collaterali. Si ottiene che, per il motore, si ha un risparmio di 77.500\$ e, riguardo la pompa idraulica, 7.300\$ per ogni manutenzione. Tutta l'analisi è trattata in [100].

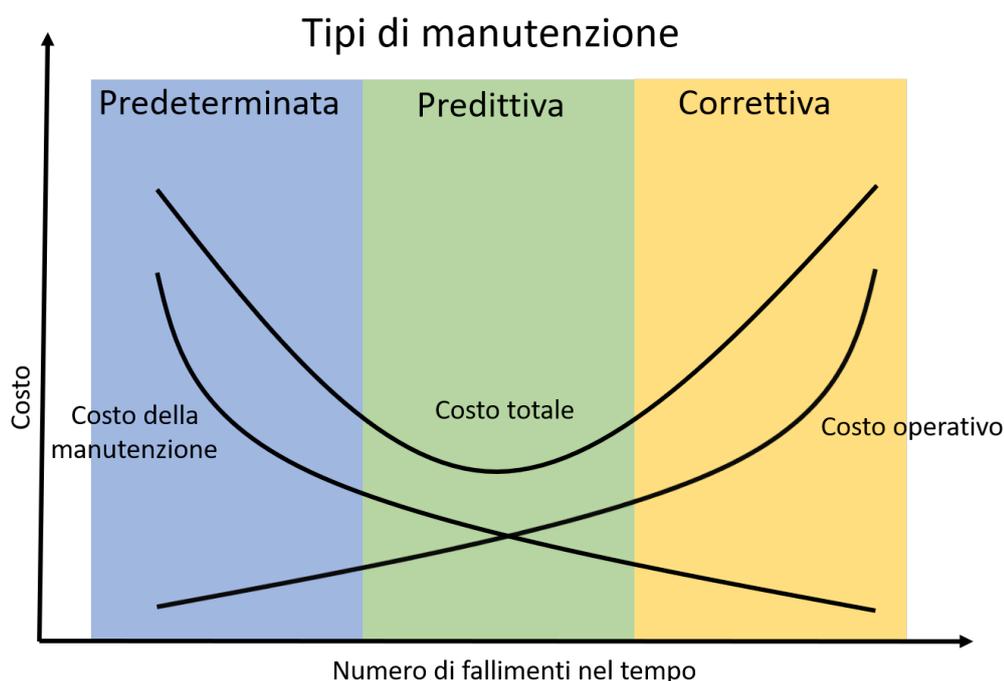


Figura 2.28. Costi derivanti dai vari tipi di manutenzione [77]

2.4.7 Tipico piano di lavoro

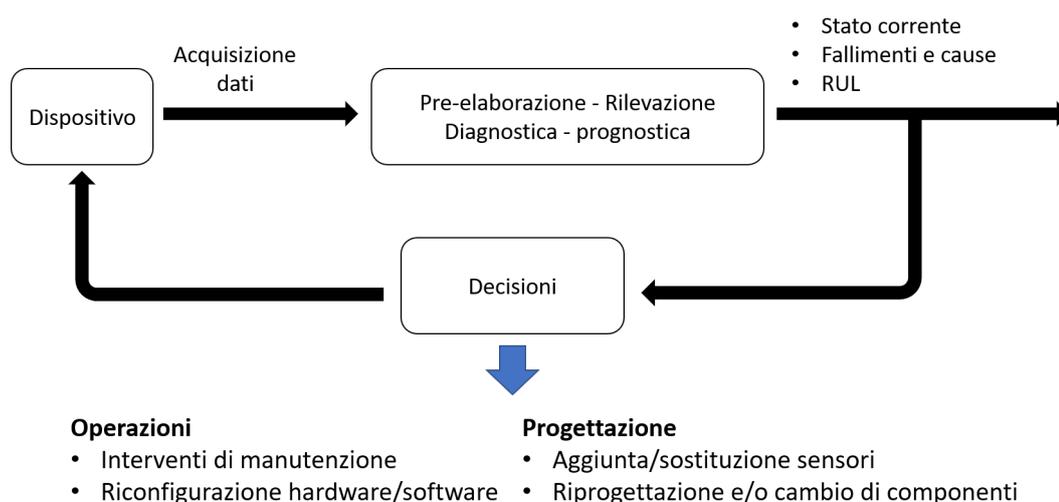
Solitamente, nella programmazione, si conoscono dallo stato dell'arte uno o più algoritmi standard per certi specifici problemi [101]. Anche l'apprendimento automatico presenta diverse procedure ma nessuna di esse è la migliore in assoluto per una certa casistica. Ciò viene presentato dal teorema No **free lunch**, traducibile come **Nessun piatto pronto**, il quale stabilisce che, in una matrice di tutti i problemi e algoritmi, la performance media di tutti gli algoritmi è equivalente [102]. Diventa quindi necessario provare diversi approcci prima di trovare il migliore per il proprio scenario specifico. Esiste comunque un orientamento ad alto livello, l'implementazione di ogni suo blocco dipenderà dal caso in esame come detto prima. Ecco i passi principali della PHM [103] [104] [105]:

- **Studio del sistema in esame.** Si analizza il dispositivo in esame, il suo modo di operare e, in particolare, il suo comportamento prima e durante il fallimento. Questo studio preliminare sarà utile per avere analisi più approfondite in futuro.

- **Acquisizione dati.** È una parte essenziale poiché tutto il lavoro successivo si baserà su ciò. I dati sono legati o ai sensori o agli eventi. Nel primo caso si trasforma un valore analogico in digitale in modo da poterlo quantificare e salvare; alcuni esempi sono temperatura, umidità, voltaggio, vibrazioni. Con la seconda tipologia vengono registrati eventi come riparazioni, fallimenti, cambio di parametri. La qualità di questo blocco è decisa dall'accuratezza dei sensori utilizzati e da che tipo di misurazioni vengono effettuate; per esempio, se il fallimento è dovuto all'aumento di temperatura, l'acquisizione di quest'ultima sarà necessaria.
- **Pre-elaborazione dati.** I valori grezzi vengono puliti. Infatti dei dati contenenti errori o rumore possono portare a risultati errati o alterati. È buona pratica effettuare delle trasformazioni in essi in modo da poterli studiare al meglio come ridurre il rumore, eliminare le anomalie, gestire gli elementi mancanti, normalizzarli e trovare correlazioni tra i vari tipi di dati acquisiti. Si possono anche effettuare proiezioni per presentare i valori su altri domini, per esempio da tempo a frequenza tramite le trasformate di Fourier. Un altro passo importante è trovare quali misurazioni portano informazioni utili e quali, invece, sviano l'analisi; si rileva ora utile la parte di studio del sistema poiché già dalla fisica del modello in esame è possibile dedurre quali entità saranno determinanti.
- **Rilevamento.** Consiste nel capire lo stato attuale del dispositivo in esame e prevedere fallimenti o anomalie in un certo intervallo di tempo prossimo. È quindi necessario conoscere i fattori e l'andamento che portano alla degradazione del sistema. Solitamente si rileva un futuro guasto in base alla differenza tra il comportamento attuale rispetto a quello normale.
- **Diagnostica.** È l'analisi da attuare dopo il fallimento. Bisogna isolare i componenti che hanno causato il problema, identificare la tipologia di guasto ed analizzare i valori precedenti all'evento. In tale maniera si studia il comportamento anomalo della macchina e si istruisce l'applicazione per riconoscere lo stesso modus operandi in futuro.
- **Prognostica.** Dopo aver raccolto abbastanza informazioni sul sistema, è possibile effettuare previsioni sempre più precise riguardo il futuro. Così è possibile ottimizzare il sistema proprio a seconda delle previsioni. Un esempio è il RUL di un macchinario per schedare al meglio le manutenzioni.
- **Decisioni.** Questo blocco è strettamente legato al precedente, è proprio grazie alle previsioni effettuate che vengono prese le decisioni per ottimizzare l'intero complesso. Ciò a cui si punta è l'autoconfigurazione delle macchine stesse, queste capiscono da sole il loro andamento e cambiano il loro modo di operare di conseguenza. Però non tutti i macchinari sono abbastanza avanzati da poter supportare tali operazioni. Per il motivo appena presentato spesso sono i lavoratori a prendere decisioni ed applicarle grazie all'aiuto di vari strumenti. In figura 2.29, è possibile vedere la posizione del lato *Decisioni* nell'intero piano di lavoro.
- **Interfaccia uomo-macchina.** Si concretizza in un'interfaccia grafica tramite la quale gli operai possono accedere a tutte le informazioni ricavate dai passi precedenti in un modo facile da visualizzare per un umano. Tramite esso possono prendere varie decisioni e gestire l'intero impianto. Questo passaggio non è necessario ma spesso utilizzato come aiuto per i lavoratori.

2.4.8 Sfide ed orientamento per il futuro per la PHM

La PHM comporta diverse sfide. Una sua scorretta applicazione comporterebbe maggiori danni rispetto ad una sua assenza. In alcuni settori, come si vede nella sezione Applicazioni e casi d'uso, la prognostica viene sfruttata per fornire una maggiore sicurezza, si immagini l'ambiente aerospaziale, energetico, nucleare, chimico, trasporti, veicoli, eccetera. Di conseguenza delle scorrette previsioni possono comportare maggiori danni rispetto i benefici e causare disastri a discapito sia della azienda che implementa l'applicazione sia degli utilizzatori umani. Oltretutto previsioni poco accurate non portano benefici ma solo uno spreco economico per la realizzazione ed implementazione hardware

Figura 2.29. *Decisioni* in PHM [103]

e software. Se ne deduce quindi che la prognostica ha una potenzialità sia positiva che negativa e deve quindi essere installata al meglio per ottenere dei reali benefici. Ecco le sfide da affrontare per raggiungere una corretta applicazione della PHM [103] [106] [107] [108] :

- **Quantificazione delle incertezze.** Naturalmente qualsiasi predizione sul futuro comporta incertezze poiché la realtà non è ideale ma imperfetta intrinsecamente. Esistono molti fattori di incertezza: i sistemi fisici cambiano in base all'ambiente esterno (esempio temperatura, umidità, pressione), modelli matematici (la realtà non segue perfettamente il modello matematico associato), il mondo reale è soggetto a degradazione (la macchina lavora perfettamente all'inizio ma perde la sua precisione con il tempo), incertezza nelle predizioni (effettuare delle predizioni comporta sempre degli errori in una certa misura), incertezza sulle soglie del fallimento (un guasto non avviene sempre superati determinati limiti ma le condizioni che causano il dissesto possono cambiare). Tutti questi problemi sono intrinseci nella natura e non possono essere eliminati, si può solo cercare di ridurli il più possibile. Essendo un argomento abbastanza lungo, le incertezze verranno presentate meglio più avanti sempre in questa sezione.
- **Validazione e verifica.** Questa fase indica l'abilità di quantificare la qualità di un determinato algoritmo. È indispensabile per: poter comparare più modus operandi riconoscendo il migliore, quantificare l'errore tra la predizione e la realtà quindi l'accuratezza di un'applicazione, poter indicare delle soglie di precisione tra i requisiti, identificare punti deboli e forti di un algoritmo. Le metriche utilizzate possono essere di tipo funzionale o basate sull'utente finale. Nel primo caso, il più utilizzato, si calcola quanto ci si avvicina o allontana ad un determinato obiettivo. Il secondo caso mostra i benefici rispetto un certo punto di vista proprio del cliente, ciò è strettamente legato ai requisiti del committente.
- **Standardizzazione.** Si intende la generazione di elementi e termini standard per la prognostica. In questo modo si semplificano i discorsi attorno alle problematiche, la comunità scientifica ha una visione comune sulla disciplina, si definiscono e si identificano in modo oggettivo gli elementi in gioco e quindi diventa più facile l'individuazione di problemi e soluzioni nell'architettura applicata. La standardizzazione deve avvenire in tre campi: termini e definizioni, sviluppo del sistema e metriche. Il primo blocco riguarda il dialogo attorno alla PHM, in questo modo è possibile una migliore comunicazione e comprensione tra la comunità scientifica. Per sviluppo del sistema si intende la definizione di un procedimento standard e generico per l'applicazione della prognostica da cui derivare i casi specifici. Tramite ciò

si semplifica l'implementazione della PHM, ogni sviluppatore non deve ricreare l'architettura base da zero ma può partire da uno schema standard. Le metriche sono necessarie per valutare quantitativamente la qualità di un certo algoritmo prognostico. Avere degli standard porterebbe la comparazione tra più metodi di diverso tipo senza problemi di incompatibilità. Ciò è strettamente collegato al punto *Validazione e verifica* riportato sopra.

- **Post prognostica.** Essa si occupa dell'autoconfigurazione dei dispositivi in base alle predizioni effettuate su di essi. Ciò presenta una certa complessità poiché unisce prognostica, macchina in esame e sua configurazione, decisioni e manutenzioni; la sua corretta realizzazione può avvenire solo se anche gli elementi elencati sono applicati propriamente. Oltretutto la configurazione cambia da macchina a macchina quindi bisogna studiare ogni caso singolarmente. Questo ambito ha una forte potenzialità ma allo stesso tempo una sua cattiva applicazione porterebbe una performance peggiore rispetto alla sua assenza.
- **Conoscenza della tecnologia e complessità del sistema.** Spesso le macchine e dispositivi a cui applicare la PHM sono complessi e costituiti da diversi dispositivi differenti tra loro. A volte è necessario conoscere nel dettaglio il loro funzionamento e spesso tutte le informazioni necessarie sono conosciute solo in parte da più persone a diversi livelli (un programmatore non ha le stesse conoscenze di un manutentore e viceversa). Non è quindi facile per un'unica persona comprendere appieno l'intero sistema che sta trattando e, di conseguenza, lavorare al meglio. Oltretutto la tecnologia diventa col tempo sempre più performante ma contemporaneamente più complessa a discapito della comprensione umana. Ciò genera altre sfide specifiche al caso o settore in esame.
- **Analisi dei benefici ottenuti.** Questa parte riguarda più il lato aziendale. Ogni cliente, prima di applicare o comprare qualsiasi metodo o mezzo, vuole la sicurezza che ciò porti dei benefici a sé stesso e che questi siano ben definiti e quantificabili. Naturalmente si desidera che il risparmio futuro dovuto all'acquisto sia maggiore delle spese impiegate per l'installazione, mantenimento, ridefinizione e cambiamento del lavoro rispetto a prima. Come visto nel capitolo Benefici della PHM, la PHM comporta vari risparmi e benefici però nel caso reale ciò non è scontato, una sua scorretta applicazione può non portare i ricavi previsti, oltretutto la sua utilità dipende anche dal caso e dall'ambiente in esame. Prendendo per ipotesi che non si avranno perdite ma solo guadagni, rimane il problema di quantificare quest'ultimi. Come appena presentato, la prognostica comporta varie sfide e incertezze quindi è difficile assicurare una certa performance nel futuro.

In tabella 2.10 sono presentate alcune sfide che si riscontrano durante la realizzazione di un'applicazione PHM, gli stadi riportati sono gli stessi già elencati in Tipico piano di lavoro. Esse sono legate al lato pratico e all'implementazione della soluzione, quelle presentate nel precedente elenco puntato sono associate più alla disciplina PHM vera e propria. I problemi della tabella sono quindi più facilmente risolvibili e dipendono in parte dalla bravura degli sviluppatori.

Siccome la prognostica si pone l'obiettivo di predire l'andamento di sistemi ingegneristici, ciò provoca l'imbattersi con diverse sorgenti di incertezze le quali procurano diversi tipi di problemi. Perciò il suo output non sarà mai perfetto ma sempre legato alla probabilità. Per esempio se si vuole conoscere il comportamento futuro di un veicolo elettrico si dovrebbe sapere a prescindere il terreno in cui ci si troverà, le condizioni ambientali, i parametri legati ad ogni suo componente (per esempio la pressione e l'usura delle ruote), lo stato e la carica della sua batteria, la sua velocità e accelerazione; ciò naturalmente non è possibile saperlo poiché sono eventi futuri, di conseguenza anche la predizione finale sarà imperfetta. Esistono alcuni modi per correggere in parte gli errori a cui si va incontro ma sono dei lavori abbastanza articolati, oltretutto aumentano la loro difficoltà con l'incremento della complessità del sistema in esame. Per prima cosa è sicuramente necessario individuare ed isolare le sorgenti dell'incertezza in modo da studiarle in modo autonomo all'intero ambiente. Per far in modo che esse siano più facilmente analizzate, si è decisa la seguente classificazione [109] [111]:

Stadio	Sfide
Acquisizione dati	<ul style="list-style-type: none"> • Corretta selezione dei sensori • Fallimenti nei sensori • Selezione dell'acquisizione dei dati del sistema • Salvataggio dati
Pre-elaborazione dati	<ul style="list-style-type: none"> • Integrazione o fusione dati • Dimensionalità eccessiva dei dati • Estrazione, selezione e riduzione delle variabili • Costruzione di un indice della salute
Rilevamento, diagnostica, prognostica	<ul style="list-style-type: none"> • Soglie dei fallimenti • Rilevamento e isolamento del guasto • Strumenti di rilevamento • Massimo RUL da prevedere • Gestione dell'incertezza • Metriche sulla qualità del risultato finale
Decisioni	<ul style="list-style-type: none"> • Riconfigurazione del sistema • Cambiamento degli obiettivi

Tabella 2.10. Sfide della PHM durante la sua applicazione [103]

- **Incerteza presente.** È legata ai dispositivi fisici. Solitamente per le previsioni si sfruttano le misurazioni su alcuni parametri dei dispositivi costituenti l'intero sistema. Ogni misurazione è però soggetta ad incerteza causata dall'ambiente esterno, malfunzionamenti, sensibilità, accuratezza e varie specifiche associate agli strumenti applicati per il rilevamento. Può essere ridotta utilizzando sensori e filtri migliori in modo da ottenere valori più accurati però ciò comporta un aumento del prezzo dei prodotti usati e una maggiore frequenza di manutenzioni. L'incerteza è detta presente poiché è legata allo stato attuale del sistema.
- **Incerteza futura.** Per essa si intendono i contesti durante i quali avvengono la diagnosi e la prognosi. Non è possibile conoscere l'ambiente futuro a cui si riferiscono le predizioni, di conseguenza quest'ultime non potranno mai essere accurate poiché si riferiscono a parametri e criteri che non si possono conoscere con certezza. A riguardo è stato presentato sopra l'esempio del veicolo elettrico del quale non si conoscono i vari parametri futuri legati a esso stesso e all'ambiente esterno.
- **Incerteza del modello.** Essa è l'errore tra la predizione effettuata dall'algoritmo applicato e l'evento reale. È estremamente difficile costruire un modello matematico che delinea l'andamento reale influenzato da moltissimi fattori quali degradazione, condizioni ambientali, eccetera, poiché la realtà non è semplicemente lineare. La matematica su cui l'uomo si basa per definire gli eventi naturali è propria di un mondo ideale ed è unicamente un'approssimazione del mondo effettivo. La realtà è composta da una moltitudine così vasta di elementi da rendere estremamente difficile la loro considerazione.
- **Incerteza nel metodo di predizione.** Una volta trovate tutte le fonti di incerteza, rimane difficile quantificare il loro effetto combinato sulle predizioni. Il problema diventa spinoso all'aumentare della complessità del caso da studiare.

La prognostica deve quindi identificare tutte le sorgenti, quantificare la loro singola incertezza ed infine calcolare il totale di tutti questi fattori in modo da avere un intervallo di valori entro il quale si trova sicuramente il risultato reale. Questo non è certamente un lavoro facile. Per schematizzare meglio la gestione delle incertezze, sono state classificate 4 attività [109] [111] [110]:

- **Rappresentazione e interpretazione dell’incertezza.** Durante questa sezione si decide come interpretare l’incertezza e come rappresentarla quantitativamente. Si può applicare una teoria probabilistica, dell’insieme sfocato (fuzzy set), basata sulle evidenze oppure un’analisi degli intervalli. Solitamente si applica un approccio probabilistico quindi i parametri vengono visti come distribuzioni probabilistiche.
- **Quantificazione dell’incertezza.** Si vuole trasformare l’incertezza in un valore numerico in modo da poter avere un criterio su cui basare la sua intensità. Bisogna stimare i parametri della distribuzione probabilistica prima della predizione. Le incertezze vanno considerate singolarmente.
- **Propagazione dell’incertezza.** Si stabiliscono in base alle informazioni ottenute dai passi precedenti gli stati futuri e le incertezze a loro associate. Ciò è calcolato propagando le varie sorgenti delle incertezze attraverso un modello di degradazione-predizione. Dopodiché si utilizzano queste informazioni per ricavare il RUL e la sua incertezza.
- **Gestione dell’incertezza.** Questa è una fase generica dove si cerca di migliorare il lavoro effettuato con l’obiettivo sia di diminuire l’incertezza sia di calcolarla in modo più preciso.

Guardando lo stato dell’arte attuale, non esistono obiettivi definiti o un particolare orientamento per il futuro della PHM, attualmente si punta unicamente ad un’accuratezza migliore nelle predizioni cercando di risolvere le sfide che sono appena state presentate. Oltre ciò si tenta di applicare la prognostica sempre in più campi, evento realizzabile grazie al continuo andamento verso la digitalizzazione in qualsiasi ambito.

2.5 Stato odierno della PHM

Il settore PHM è in costante sviluppo, in particolare il ramo di manutenzione predittiva. I campi *apprendimento automatico* e *piattaforma IoT* sono stati posti nel 2016 [112] e 2017 [113] nella cima *Picco delle aspettative gonfiate* del grafico **Gartner Hype Cycle for Emerging Technologies** il quale presenta ogni anno le tecnologie a cui si dà maggiore potenziale per lo sviluppo futuro. Nel 2018 invece troviamo solo la *piattaforma IoT* [114]. **Google Trend** mostra una forte impennata riguardo le ricerche dell’argomento apprendimento automatico e un aumento costante ma non tanto ripido sull’argomento manutenzione predittiva.

Si vuole ora mostrare il mercato della manutenzione predittiva per cui si farà fede al lavoro svolto da **Pwc** nel Giugno 2017 intitolato *Predictive Maintenance 4.0 - Predict the unpredictable* [115]. Per mostrare al meglio la situazione attuale, classifichiamo in 4 livelli le aziende a seconda delle funzionalità implementate in questo settore:

- **Livello 1 - Ispezioni visive.** Ispezioni periodiche fisiche, ci si basa solamente sull’ispezione di un esperto che osserva l’interfaccia grafica proposta. Un operaio deve periodicamente andare a controllare le macchine.
- **Livello 2 - Ispezioni degli strumenti.** Ispezioni periodiche basate sulla combinazione delle osservazioni di un esperto e l’output di un dispositivo.
- **Livello 3 - Monitoraggio in tempo reale.** Continue ispezioni in tempo reale con la creazione di allarmi secondo certe regole a cui vengono associati un determinato livello di criticità.

Current predictive maintenance maturity level

Figura 2.30. Corrente livello di maturità della manutenzione predittiva [115]

- **Livello 4 - Manutenzione predittiva con big data.** Continue ispezioni in tempo reale con la creazione di allarmi in base a delle tecniche predittive.

Dalla figura 2.30 si può vedere che la maggior parte delle aziende non arriva al livello 3, solo l'11% raggiunge la categoria massima. Ciò significa che il lavoro di manutenzione predittiva non è ancora abbastanza diffuso ma deve essere ampliato col tempo.

Il settore che è più avanzato in questo ambito è quello ferroviario, nel 42% dei casi si ha un livello 4 rispetto all'11% di media.

Si vuole allora analizzare l'interesse delle aziende per migliorare la loro applicazione dell'industria 4.0. Dalla figura 2.31, circa la metà degli intervistati dichiara che la propria compagnia ha un piano per migliorarsi, un quinto di essi lo sta già portando avanti. Rimane importante il fatto che invece la seconda metà non abbia programmato nulla a riguardo (bisogna però considerare la possibilità che alcuni intervistati non fossero a conoscenza di eventuali piani nonostante questi siano stati concepiti).

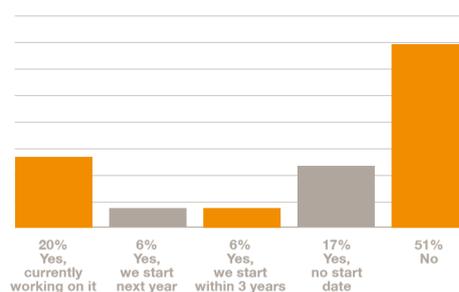
Future plans for PdM 4.0

Figura 2.31. Piani futuri per la manutenzione predittiva [115]

Studiando quantitativamente il mercato, la manutenzione predittiva ha fruttato circa 1.5 miliardi di dollari nel 2016, con una crescita stimata, secondo IoT Analytics nel 2016 [116], fino a quasi 11 miliardi per il 2022. In figura 2.32 è mostrata la sua crescita negli anni in miliardi di dollari. Consultancy.uk propone dati aggiornati al 2018 [117], in quest'anno il mercato è di 3.1 miliardi di dollari (come previsto da IoT Analytics nel 2016) e stima 11 miliardi nel caso ottimistico per il 2022 (anche questo caso molto simile alle previsioni di IoT Analytics del 2016).

La figura 2.33 mostra l'andamento del mercato appena citato. Il tasso annuo di crescita composto (CAGR) è quindi del 39% nel caso ottimistico e del 27% nella predizione base.

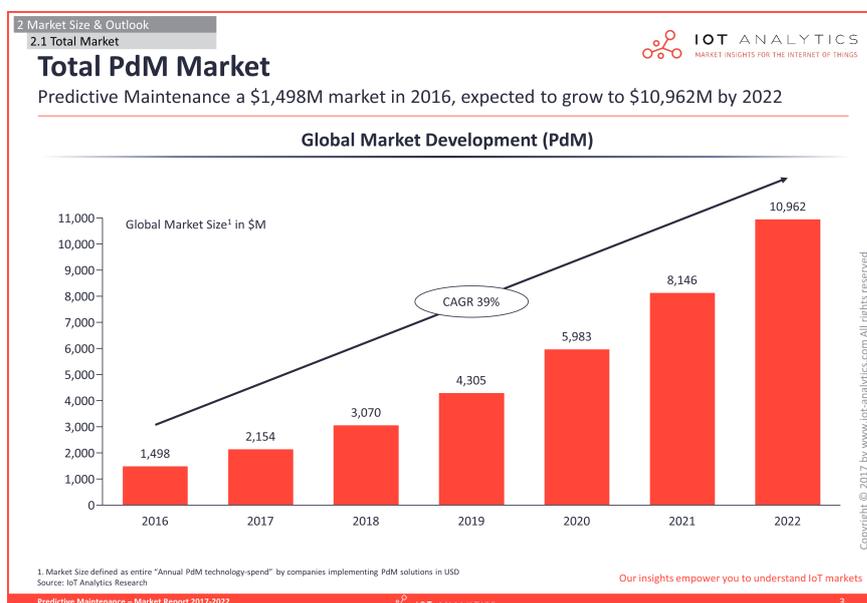


Figura 2.32. Andamento mercato sulla manutenzione predittiva dal 2016 al 2022 da parte di IoT Analytics, svolto nel 2016 [116]

Predictive maintenance.

Forecast global market development to 2022 [USD billion].

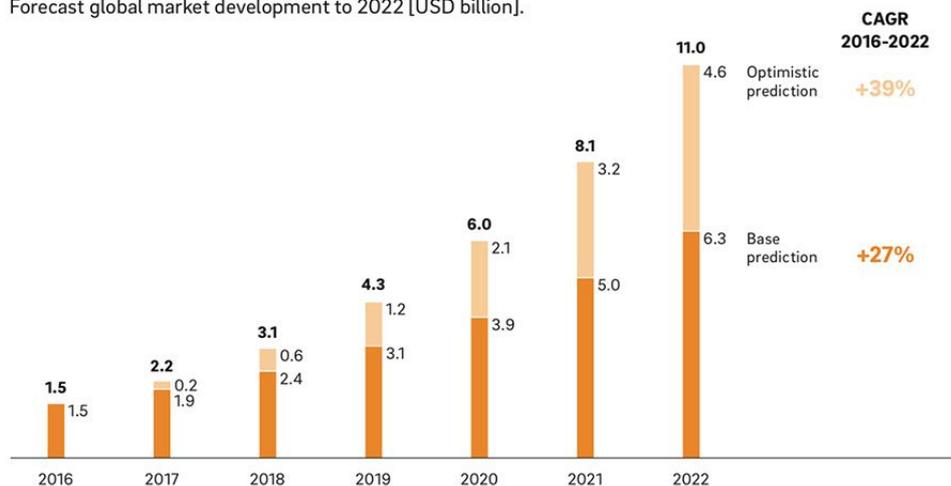


Figura 2.33. Andamento mercato sulla manutenzione predittiva dal 2016 al 2022 da parte di Consultancy.uk, svolto nel 2018 [117]

IoT Analytics ha anche stilato nel 2017 una classifica delle aziende più avanti nell'ambito della manutenzione predittiva [5], essa è riportata in figura 2.34. In questo modo si ha una visione delle aziende leader ovvero coloro che più gestiscono il mercato e l'andamento del settore. Il podio è composto da:

1. **IBM.** Il sistema cognitivo IBM Watson permette di creare varie implementazioni della manutenzione predittiva. Un caso è Predictive Maintenance and Quality (PMQ) che verrà

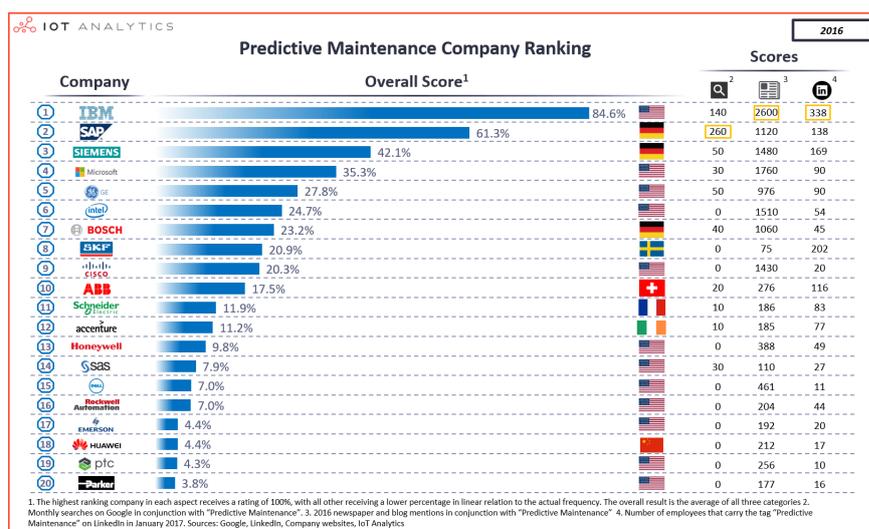


Figura 2.34. Classifica aziende in base alla manutenzione predittiva secondo IoT Analytics (2017) [5]

presentato meglio sotto, per esempio esso è stato applicato agli ascensori prodotti dall'azienda Kone [119]. Serve per monitorare, analizzare e generare report sulla salute di un qualsiasi sistema.

2. **SAP.** Essa ha saputo integrare al suo software gestionale già leader nel mercato la manutenzione predittiva in modo da migliorarsi e fornire nuovi servizi. La sua soluzione più importante è chiamata **Predictive Maintenance and Service**, un esempio della sua realizzazione concreta è avvenuto nell'azienda **Kaeser Kompressoren**.
3. **Siemens.** Questa è più indirizzata verso l'automazione, macchine e impianti industriali. Lavorando su macchinari che si prestano al monitoraggio, possiedono già in partenza una mole di dati su cui lavorare. Un esempio pratico si ha nel **NASA Armstrong Flight Center** sui sistemi di raffreddamento.

Un'applicazione software leader del settore è **Predictive Maintenance and Quality (PMQ)** [118] prodotta da **IBM**, questa verrà ora presentata per avere un'idea dell'uso e dei servizi di un tale prodotto. Nell'apposito sito web viene presentata come un programma focalizzato a identificare e gestire i rischi e la salute di un qualsiasi impianto o operazione aziendale. Utilizza l'apprendimento automatico per migliorare e rendere autonomi i propri processi, identifica i fattori che influiscono positivamente o negativamente, effettuano comparazioni e riassumono il loro lavoro in dei report. I benefici elencati da **IBM** sono: riduzione dei periodi di inattività, ottimizzazione delle manutenzioni e delle risorse, estensione della vita dei prodotti, visualizzazione delle performance in esame, semplificazione del lavoro dei manutentori e riduzione dei costi legati all'implementazione e all'informatica. I servizi più importanti sono:

- **Visibilità dello stato di manutenzione di ogni dispositivo.** I dispositivi vengono rappresentati come delle carte da gioco, ad ognuno di questi è associato uno status che indica se la macchina è mantenuta bene. Allo stesso tempo l'applicazione consiglia e ottimizza le risorse e le manutenzioni.
- **Fattori conduttori e di rischio.** Si analizza lo stato della macchina e si cerca sia di predire i prossimi fallimenti sia di individuare gli attori che generano i guasti. Si cerca di classificare il contesto durante il quale la macchina lavora bene o male in modo da effettuare predizioni più accurate.

- **Comparazione delle performance dei vari dispositivi.** Avvengono varie comparazioni tra i diversi prodotti in modo da identificare i fattori che migliorano o peggiorano la salute del sistema.
- **Apprendimento automatico e analisi avanzata.** Grazie all'apprendimento automatico si cerca di trovare gli effetti nel comportamento della macchina causati da un evento di manutenzione. In questo modo potranno essere schedate al meglio le azioni, vengono subito richieste se si predice un fallimento imminente o sono posticipate se superflue.

Si vogliono ora presentare i benefici percepiti dai lavoratori stessi. **IoT Analytics** ha compiuto dei sondaggi [120] su 151 dirigenti d'azienda in ambito industriale ed ha chiesto loro di selezionare i tre principali benefici derivanti dal lato analisi. La più importante è considerata come 3 punti, la seconda 2 punti e l'ultima 1. Il risultato in figura 2.35 mostra che i vantaggi più sentiti sono l'aumento delle entrate economiche e della soddisfazione del cliente.



Figura 2.35. Migliori benefici ottenuti dalla PHM in ambito industriale secondo un sondaggio di **IoT Analytics** svolto a fine 2016 [120]

2.5.1 Applicazioni e casi d'uso

La PHM ha trovato una moltitudine di applicazioni in diversi settori. La sua flessibilità abbraccia un largo insieme di settori risolvendo vari tipi di problemi. Fino ad ora si è presentata la disciplina indirizzandola al settore industriale in modo da collegarla meglio all'ambiente dell'industria 4.0 però le predizioni derivanti dalla prognostica possono essere orientate verso i più svariati obiettivi. Per prima cosa, il lavoro sulla manutenzione predittiva può essere impiegato su qualsiasi dispositivo come ascensori, semafori, veicoli, bancomat in modo da prevedere i guasti e schedare al meglio le manutenzioni; il concetto è il medesimo delle macchine manifatturiere. Scoprire un determinato andamento futuro può essere particolarmente utile se applicato a divisioni quali mercati finanziari, vendite di un certo prodotto per l'azienda produttrice, meteo, stime su certi eventi (ad esempio come cambierà la ricchezza/povertà di un paese nei prossimi anni, quantità di nuove nascite nel prossimo anno), eccetera. L'applicazione dell'apprendimento automatico porta un passo avanti sia l'industria sia la società in cui viviamo verso il 4.0 e la digitalizzazione fornendo servizi e ottimizzazioni prima non possibili. Ecco alcune sue applicazioni pratiche e casi di studio [106] [121] [122] [123]:

- **Ambito industriale.** Questo settore è già stato ampiamente presentato e verrà trattato in maggiore dettaglio nella sezione Manutenzione predittiva dove è descritta un'applicazione della manutenzione predittiva. In particolare viene sfruttato il settore manifatturiero. La

PHM è implementata in maggioranza per prevedere i fallimenti e l'andamento della salute delle macchine di un impianto ottimizzando gli strumenti utilizzati. L'applicazione e benefici della prognostica sui sistemi ciberfisici sono già stati introdotti nella sezione PHM applicata ai sistemi ciberfisici.

- **Ambito aerospaziale, ferroviario, navale, trasporti.** Ciò comprende qualsiasi tipo di trasporti, sia pubblici sia privati. Il corretto funzionamento dei veicoli è importante sia per il trasporto merci sia lo spostamento umano. Un guasto inatteso comporta ritardi e disagi di diverso tipo, è di notevole importanza la continuità e l'affidabilità dei mezzi impiegati. In caso di persone a bordo, un incidente può ledere la sicurezza dei passeggeri. È di notevole importanza la puntualità la quale è possibile solo tramite un corretto funzionamento del sistema, allo stesso tempo è necessario predire i ritardi in modo da prendere provvedimenti consci di ciò. Oltre ai mezzi di trasporto veri e propri, è indispensabile anche la gestione degli ambienti che orbitano attorno ad essi quali aeroporti, porti, stazioni ferroviarie, fermate dei bus. Anche quest'ultimi elementi devono presentare sicurezza e corretto funzionamento. Altro fattore è la previsione degli eventi quali meteo e traffico in modo da scegliere le strade e momenti più adatti per il trasporto. I benefici da raggiungere grazie alla PHM sono sicurezza e corretto funzionamento dai quali derivano puntualità, affidabilità, qualità del servizio e reputazione per l'azienda proprietaria.
- **Difesa.** In ambito militare si vuole da un lato ottimizzare le proprie armi dall'altro proteggere i militari durante l'utilizzo di esse. In particolare per il secondo motivo, la PHM fa in modo che gli strumenti utilizzati lavorino sempre alla perfezione predicendo in anticipo eventuali fallimenti. Ciò è strettamente legato a tutti i trasporti impiegati in ambito militare: aerei, navi, carri armati, furgoni, automobili. Le armi non devono per alcun motivo andare fuori controllo e danneggiare oggetti e persone fuori dai loro obiettivi. Il punto fondamentale di questo settore è quindi la sicurezza.
- **Ambito energetico.** L'ambito energetico di avvicina molto a quello industriale, per esempio le turbine eoliche possono essere considerate come una qualsiasi macchina manifatturiera. L'unico dettaglio è che in questo caso le macchine utilizzate sono spesso più grandi, costose e potenzialmente pericolose. Si possono facilmente immaginare i danni dovuti ad un incidente in un impianto nucleare. Nel settore energetico è quindi maggiormente importante la sicurezza e, di conseguenza, la previsione di guasti e fallimenti.
- **Batterie.** Si è notato che la PHM può essere applicata alle batterie in modo da conoscere il momento in cui si scaricheranno, ciò è molto simile al calcolo del RUL.
- **Ambito elettronico.** Siccome la maggior parte dei dispositivi si muove sempre più verso un orientamento elettronico e digitale, l'applicazione della PHM arriva a frontiere più ampie. Un esempio dell'importanza della prognostica in questo settore è la continuità nell'erogazione di un servizio online di qualsiasi tipo: sito web, accesso database, acquisti tramite carta di credito, messaggistica, utilizzo di un'applicazione mobile.
- **Automobili.** Il fatto che le automobili sono sempre più elettroniche ha permesso l'applicazione della PHM. In questo caso sono importanti le analisi in tempo reale come la previsione di scontri con altre macchine o ostacoli di qualsiasi tipo, la macchina può autonomamente decidere di frenare in modo da evitare o attenuare il danno. Il primo passo per una maggiore sicurezza è evitare guasti di qualsiasi pezzo montato nell'auto in modo da prevenire gli incidenti. Un altro fattore è la predizione del traffico e quindi del tracciato più veloce per arrivare in un certo luogo in modo da migliorare l'esperienza di guida.
- **Ambito medico.** Ormai il settore medico è sempre più basato su macchine e strumenti elettronici. Ciò permette di ottenere diagnosi sempre più precise e meno invasive però un malfunzionamento potrebbe portare a errori sull'output. Oltretutto alcuni macchinari vengono utilizzati per interventi chirurgici quindi la precisione è un criterio fondamentale. Anche in questo caso la sicurezza e il corretto funzionamento sono gli obiettivi della PHM.

Dall'elenco appena stilato, si può quindi notare che la maggior parte delle applicazioni è legata ai dispositivi elettronici e meccanici e agli ambienti vicini ad essi. Quasi tutti ottengono una maggior sicurezza e affidabilità dai mezzi utilizzati e ciò comporta una lunga serie di benefici a seconda del settore in esame, a livello generale essi sono riportati nella sezione Benefici della PHM. La PHM comporta quindi un'ottimizzazione prima non raggiungibile semplicemente tramite l'intervento umano.

Capitolo 3

Manutenzione predittiva

3.1 Descrizione del problema

Nell'industria 4.0 si introduce l'utilizzo di tecniche di apprendimento automatico per effettuare previsioni sullo stato e lavoro dell'impianto industriale. Lo scopo dell'applicazione da sviluppare sarà effettuare delle previsioni sulla salute di una certa macchina e, più nel dettaglio, prevedere i suoi prossimi fallimenti in modo che gli operai possano effettuare le adeguate manutenzioni predittive e organizzare al meglio il loro lavoro. Si è utilizzato Python come linguaggio di programmazione sfruttando la piattaforma Jupyter Notebook, la loro scelta è spiegata nella sezione Jupyter Notebook. L'applicazione dovrà rispondere alle seguenti domande [124] [125]:

- **La macchina avrà un fallimento nei prossimi N secondi/ore/giorni/cicli macchina?** Questo è un problema di classificazione binaria. Nel nostro caso si considereranno i prossimi cicli macchina come unità di tempo. L'output è booleano ovvero sì/no, vero/falso oppure 1/0. Le predizioni ottenute sono utili perché il lavoratore può, in questo modo, conoscere il comportamento futuro della macchina e quindi agire di conseguenza. In figura 3.1 è mostrato un tipico lavoro di etichettatura per un problema di classificazione binaria.

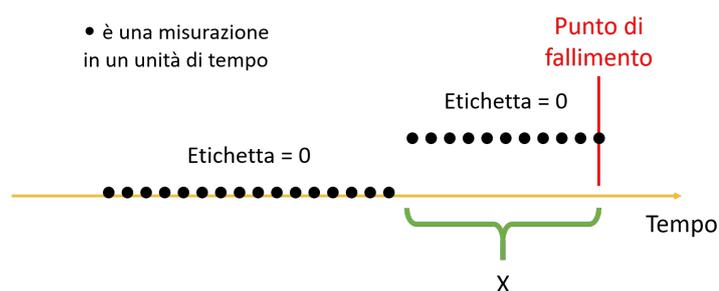


Figura 3.1. Etichette per la classificazione binaria [124]

- **La macchina si sta comportando in modo anomalo?** La risposta alla domanda è un problema di classificazione multi classe. L'output è un'etichetta che rappresenta lo stato della macchina, le etichette totali possono essere più o meno numerose a seconda della precisione che si vuole ottenere, usandone unicamente due si ritorna al caso precedente. Per esempio, applicando una classificazione in 3 sezioni queste potrebbero essere lo stato *Normale* (comportamento come previsto), *Anomalo* (stato leggermente diverso da quello programmato) e *Critico* (macchina vicina alla rottura); tra *Normale* e *Critico* possono essere posti vari stati

intermedi in modo da sapere quanto l'andamento del macchinario si scosta da quello che dovrebbe seguire. Un esempio della classificazione è mostrato in figura 3.2.

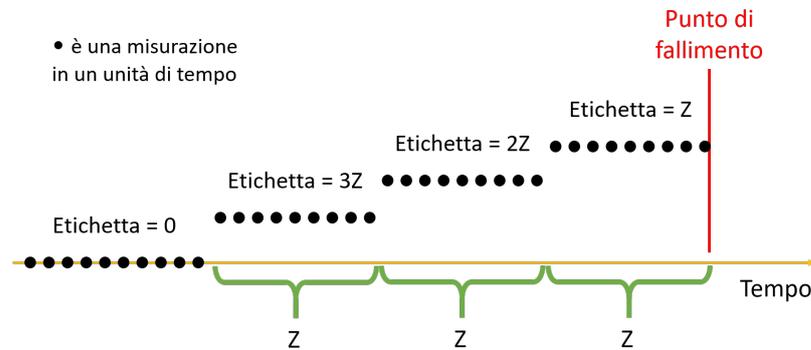


Figura 3.2. Etichette per la classificazione multi classe [124]

- **Quanto tempo o cicli macchina mancano prima che si abbia il prossimo fallimento?** Ciò corrisponde alla ricerca del Tempo di vita rimanente utile o Remaining Useful Life (RUL) ovvero l'ammontare di tempo tra il momento attuale e l'attimo in cui avverrà il prossimo guasto sulla macchina considerata. Dal punto di vista dell'apprendimento automatico, si applica un lavoro di regressione. In questo caso il tempo verrà considerato in cicli macchina poiché i macchinari su cui si basa questo studio lavorano in modo periodico in base ad essi. L'output sarà un numero positivo continuo (in particolar modo intero poiché considereremo i cicli macchina). Per la sua predizione sono necessari vari dati sulla storia della macchina e un processo di degradazione graduale. Il lavoro effettuato per questo tipo di problema è mostrato in figura 3.3.

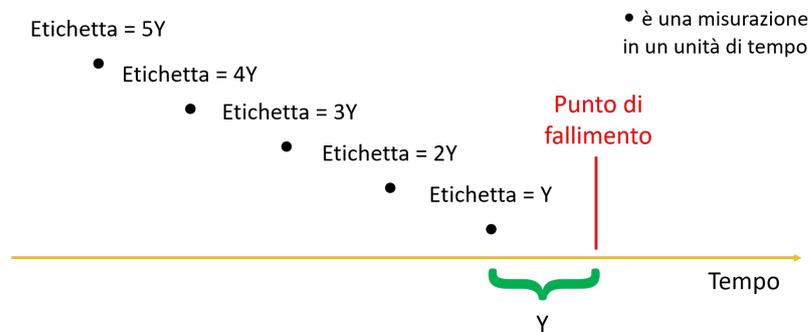


Figura 3.3. Etichette per la regressione [124]

3.1.1 Classificazione e regressione

Per capire meglio il lavoro che verrà effettuato per l'applicazione bisogna presentare i classici modus operandi per questo tipo di problemi. Questi sono due: classificazione e regressione [126] [127] [128].

Per classificazione si intende una funzione (f) che mappa delle variabili in input (X) su delle variabili discrete in output (y). In parole più semplici, si associano a dei dati una certa classe discreta; alcuni esempi sono: decidere se un'email è spam oppure no, se un'immagine rappresenta un cane o un gatto, individuare lo stato di una macchina industriale come normale, intermedio

oppure pericolo. Si cade in questa tipologia quando l'output è discreto, a ciò corrispondono la prima e la seconda domanda dell'applicazione presentata. Un esempio grafico è dato dalle figure 3.1 e 3.2. In caso le variabili in output siano solo due, si ha una classificazione binaria, altrimenti è multi classe.

La regressione consiste in una funzione (f) che mappa delle variabili in input (X) su delle variabili continue in output (y) quindi si associano a dei dati un valore continuo quasi sempre numerico. Alcuni esempi sono: predire l'andamento in borsa, il tempo che manca prima di un certo evento, la temperatura di un oggetto dopo un certo utilizzo, quantità di traffico (o automobili) in una certa zona e tempo. La terza domanda presentata sopra e la figura 3.3 rientrano in questo caso. Quindi, se l'output è una variabile continua allora si tratta un caso di regressione.

Si avranno metodi diversi per valutare la qualità del risultato a seconda della tipologia di problema in questione. Anche gli algoritmi utilizzati e l'approccio ai dati sarà diverso. In entrambi i casi si utilizzano algoritmi supervisionati [129] ovvero sono già conosciuti tutti i possibili output, essi sono quelli descritti nelle domande dell'applicazione su Descrizione del problema.

3.1.2 Macchina ad iniezione: IntElect 160/520-680 Smart

La sorgente dei dati è una macchina ad iniezione (o pressa ad iniezione), in particolare essa è di tipo IntElect 160/520-680 Smart del produttore Sumitomo (SHI) Demag Plastics Machinery GmbH [130] [131] [132]; è mostrata in figura 3.4.

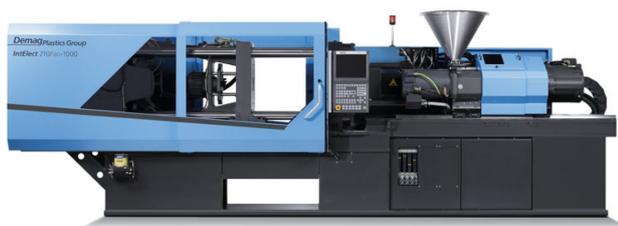


Figura 3.4. Macchina ad iniezione IntElect 160/520-680 Smart prodotta da Sumitomo (SHI) Demag Plastics Machinery GmbH [132]

Una classica macchina ad iniezione lavora prendendo del materiale plastico che, venendo fuso, viene iniettato in uno stampo in modo da modellarlo e produrre un oggetto plastico con una determinata forma [133]. Una macchina ad iniezione è divisa in due sezioni fondamentali: unità di iniezione e di chiusura [134]. Il primo gruppo è composto principalmente da vite e motore per l'iniezione, tramoggia, miscelatore ed essiccatore per i granuli di plastica; il suo scopo è prendere, fondere ed iniettare nell'altra unità i polimeri. La seconda struttura è definita da due stampi (uno mobile e l'altro fisso), gli estrattori e un sistema di raffreddamento; il suo compito è contenere il materiale iniettato secondo una certa forma, raffreddarlo ed espellerlo. Lo schema appena presentato è visibile in figura 3.5.

Il dispositivo lavora in modo periodico, la macchina esegue periodicamente le stesse operazioni per la creazione di un nuovo pezzo plastico, ogni esecuzione è detta ciclo macchina. I sensori del macchinario effettuano le misurazioni ogni ciclo. Un ciclo macchina è definito come [135]:

- **Preparazione granuli.** Vengono inseriti nell'apposita tramoggia i granuli di polimeri. Questi sono miscelati dalla rotazione della vite e riscaldati dalle resistenze.
- **Prima fase d'iniezione.** La vite spinge il materiale attraverso il canale d'iniezione in modo da mandarlo nello stampo. Quest'ultimo è chiuso da un pistone idraulico per evitare l'apertura dovuta all'elevata spinta.

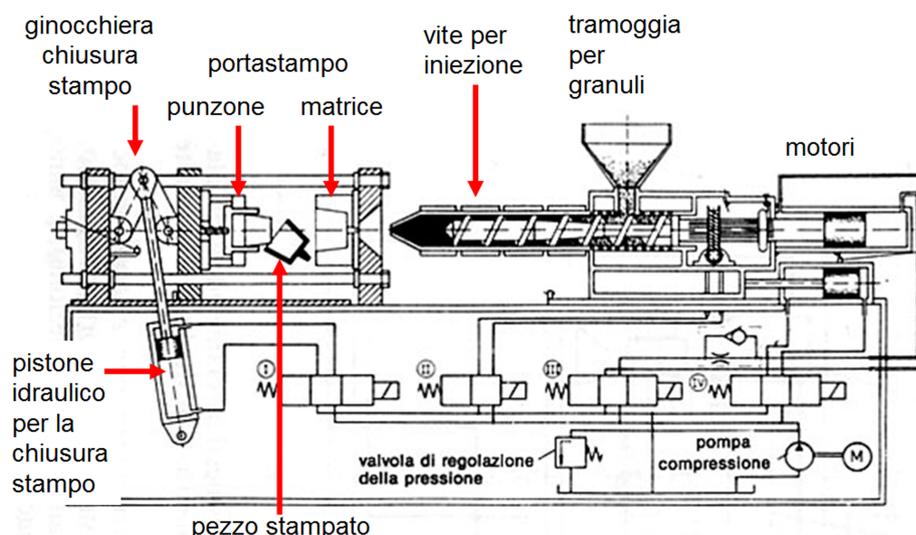


Figura 3.5. Schema di una tipica macchina ad iniezione [135]

- **Seconda fase d'iniezione.** Lo stampo è completamente pieno ma, siccome la plastica fusa diminuisce il suo volume una volta solidificata, la vite riesegue un'iniezione a pressione minore.
- **Solidificazione.** L'oggetto viene raffreddato ed estratto dallo stampo.

Nel manuale della macchina [132], viene suggerito dallo stesso produttore un servizio periodico di manutenzione in modo da garantire sicurezza ed evitare inutili spese. Dalla figura 3.6 si possono leggere le azioni da effettuare e la loro periodicità. Le manutenzioni suggerite, come descritto nella sezione Manutenzioni, sono quindi di tipo predeterminato. Il sottoscritto non ha avuto modo di vedere la macchina fisica da cui sono stati presi i dati e parlare con i manutentori responsabili quindi non conosce le attività realmente svolte sul macchinario, di conseguenza ci si baserà sul piano di manutenzione proposto dal produttore. Ci si aspetta che vengano eseguiti tutti i controlli indicati nel manuale più eventuali azioni di manutenzione correttiva per gli eventi non previsti. La maggior parte dei lavori da effettuare sono basati sulla lubrificazione dei componenti e la sostituzione di alcuni pezzi usuranti.

La macchina dispone di un terminale dal quale è possibile configurarla, monitorarla e trovare la sua documentazione in modo che il lavoratore possa gestirla facilmente tramite un'interfaccia grafica. Grazie ad esso è anche possibile scaricare i dati salvati sul macchinario.

3.1.3 Raccolta e struttura dati

I dati raccolti sono contenuti in due file con estensione *txt*. Il primo presenta le misurazioni effettuate dal 07/03/2018 ore 17:51:03 fino al 12/03/2018 ore 11:31:50 e contiene 8929 cicli macchina, il secondo raccoglie 20000 cicli ottenuti dal 21/05/2018 ore 17:08:31 al 29/05/2018 ore 13:17:10. Entrambi i documenti sono ottenuti scaricandoli dalla macchina stessa tramite il suo terminale. Questi sono formati da un'intestazione e dalle misurazioni effettuate come mostrato in figura 3.7. All'inizio del documento vengono inseriti, in ordine: il nome del produttore (DEMAG Plastics Group) e della macchina (NC5 IntElect 160/520 72800071), il nome delle misure e le unità di misura associate. Poi, per ogni riga, sono riportate le misurazioni effettuate per ogni ciclo macchina. A volte sono presenti delle righe che iniziano con "Tolleranze" oppure "Valore impostato", esse

6.3 Piano di manutenzione

Intervalli	Lavori da eseguire	vedi Capitole
in caso di apparizione della segnalazione di guasto sul video	Eseguire i lavori di manutenzione seguendo le relative istruzioni	
mensilmente	Lubrificazione centrale – verificare livello di riempimento della cartuccia del grasso	6.6
annualmente	Sostituire le Backup-batterie	6.14
	Lubrificare cuscinetto motore del dosaggio (UI 45 - UI 680)	6.5.6
ogni 500 ore di servizio	Lubrificazione centrale – verificare livello di riempimento della cartuccia del grasso	6.6
	filtro acqua 2FT20 - pulire il setaccio	6.15.4
	Pulizia e decalcificazione del raffreddamento acqua (zona di riempimento plastificazione)	6.15
	verificare e, se necessario, sostituire le condutture / i tubi	6.7
ogni 1000 ore di servizio	Lubrificare cuscinetto espulsore	6.5.2
	Lubrificare corona dentata regolazione altezza stampo	-
	Lubrificare guida lineare unità d'iniezione	6.5.5
	Verifica tensione cinghia dentata	6.10
Ogni 3 mesi	Lubrificare cuscinetto unità d'iniezione (UI 45 - UI 680)	6.5.7
	Contenitore dell'olio idraulico dispositivo ad ugelli – controllare il livello dell'olio	6.5.8
ogni 2000 ore di servizio	Lubrificare filetto della colonna regolazione altezza dell'attrezzo	6.5.3
	Lubrificare le guide della piastra di reazione/ il telaio macchina	6.5.4
	Lubrificare le guide della piastra portastampi mobile	6.5.1
	Verifica cinghia dentata e dischi dentati unità di chiusura, unità di iniezione e espulsore, sostituzione cinghia dentata in caso di usura dei denti e/o rotture	6.9 6.10
	Verifica viti di tensione set di tensione dell'espulsore	6.11
ogni 10000 ore di servizio	Cambio d'olio contenitore dell'olio idraulico dispositivo ad ugelli	6.5.8
ogni 3 anni	Sostituire le cinghie dentate di unità di chiusura, unità di iniezione ed estrattore	6.10
ogni 20000 ore di servizio	Controllo visivo, filtri nelle porte dell'armadio elettrico	6.13
	Sostituire ventilatore cabina elettrica	6.13
ogni 20000 ore di servizio, al più tardi dopo 5 anni	Sostituire condotte flessibili	6.7



ZE 1332, ZE 1333, ZE 5482, ZE 5483

Tutti i tubi flessibili ad alta pressione devono essere sostituiti con tubi nuovi ogni 5 anni, in modo da evitare danni causati dall'eccessivo invecchiamento.

Utilizzare solo tubi flessibili originali (catalogo pezzi di ricambio).

Figura 3.6. Manutenzioni consigliate dal produttore per la macchina IntElect 160/520-680 Smart [132]

sono dovute al fatto che è stata cambiata una certa configurazione della macchina e, nel nostro caso, sono inutili.

Le misurazioni vengono effettuate dai sensori montati sulla macchina. Nel manuale non sono elencati i sensori e il significato dei loro dati quindi le seguenti informazioni sono dedotte dall'installazione dei file ottenuti e dallo studio di una tipica macchina ad iniezione [133] [136]; perciò le definizioni riportate per le misure potrebbero essere errate. Non conoscendo nello specifico il modello dei sensori utilizzati non è possibile conoscere la qualità dei dati ricavati da essi. Tutti i valori, tranne quelli di *Data* e *Tempo*, sono riportati come numeri reali finiti positivi, in particolare per *Cicli* essi sono interi. Le grandezze registrate sono:

Reset valori
NCS IntElect 160/520 72800071

DEMAG Plastics Group

Data	Tempo	Cicli	Tempo ciclo s	Tempo iniezione s	Tempo dosaggio s	Tempo s	Cuscino ccm	Stop vite ccm	PM Pressi bar_spec	p-ini mas bar_spec	chius.st. s	apert.sta s	Stampo Pos.apert mm	Pos.dosag stop ccm	PM Pos ccm	Z 2 Temp °C	ZA Temp °C	UG1 Temp °C	Z 1 Temp °C
07.01.18	13:51:03	000145159	28.819	1.547	3.008	14.48	81.23	1519.3	1538.0	1.850	1.920	330.00	96.51	15.44	275	80	285	275	
07.01.18	13:51:26	000145160	28.821	1.547	3.042	14.42	81.23	1520.3	1535.9	1.850	1.920	330.00	96.50	15.44	275	80	285	275	
07.01.18	13:51:50	000145161	28.830	1.547	3.024	14.23	81.23	1517.4	1537.9	1.850	1.920	330.00	96.50	15.44	275	80	285	275	
07.01.18	13:52:14	000145162	28.830	1.547	3.027	14.47	81.23	1517.2	1536.7	1.850	1.920	330.00	96.50	15.44	275	80	285	275	
07.01.18	13:52:38	000145163	28.820	1.547	3.019	14.52	81.23	1518.9	1535.1	1.850	1.920	330.00	96.51	15.44	275	80	285	275	
07.01.18	13:53:02	000145164	28.820	1.547	3.014	14.55	81.23	1519.2	1537.4	1.850	1.920	330.00	96.49	15.45	275	80	285	275	
07.01.18	13:53:26	000145165	28.820	1.547	3.025	14.59	81.23	1519.1	1537.8	1.850	1.920	330.00	96.51	15.44	275	80	285	275	
07.01.18	13:53:49	000145166	28.829	1.547	3.032	14.17	81.23	1516.7	1537.8	1.850	1.920	330.00	96.50	15.44	275	80	285	275	
07.01.18	13:54:13	000145167	28.831	1.547	3.012	14.28	81.23	1514.9	1535.9	1.850	1.920	330.00	96.52	15.44	275	80	285	275	
07.01.18	13:54:37	000145168	28.819	1.547	3.046	14.52	81.23	1523.2	1537.5	1.850	1.920	330.00	96.51	15.44	275	80	285	275	
07.01.18	13:55:01	000145169	28.831	1.547	3.057	14.20	81.23	1515.5	1536.2	1.850	1.920	330.00	96.50	15.44	275	80	285	275	
07.01.18	13:55:25	000145170	28.820	1.547	3.049	14.35	81.23	1512.1	1543.4	1.850	1.920	330.00	96.51	15.44	275	80	285	275	
07.01.18	13:55:48	000145171	28.830	1.547	3.028	14.24	81.23	1517.5	1537.2	1.850	1.920	330.00	96.51	15.44	275	80	285	275	
07.01.18	13:56:12	000145172	28.820	1.547	3.016	14.26	81.23	1518.4	1533.8	1.850	1.920	330.00	96.50	15.44	275	80	285	275	
07.01.18	13:56:36	000145173	28.820	1.547	3.030	14.58	81.22	1520.7	1538.0	1.850	1.920	330.00	96.50	15.44	275	80	285	275	
07.01.18	13:57:00	000145174	28.820	1.547	3.015	14.40	81.23	1518.5	1538.3	1.850	1.920	330.00	96.50	15.44	275	80	285	275	
07.01.18	13:57:24	000145175	28.830	1.547	3.044	14.51	81.23	1514.5	1537.5	1.850	1.920	330.00	96.51	15.44	275	80	285	275	
07.01.18	13:57:48	000145176	28.829	1.547	3.037	14.15	81.23	1517.8	1539.7	1.850	1.920	330.00	96.49	15.45	275	80	285	275	

Figura 3.7. Esempio del file contenente i dati della macchina

- **Data.** È la data in cui è stata effettuata la misurazione in formato *dd.MM.yy* (giorno.mese.anno).
- **Tempo.** È l'ora in cui è iniziato il ciclo macchina in formato *hh:mm:ss* (ore:minuti:secondi).
- **Cicli.** È un identificativo univoco per indicare un ciclo macchina.
- **Tempo ciclo (s).** È il tempo per eseguire un completo ciclo macchina.
- **Tempo iniezione (s).** È il tempo per effettuare l'iniezione per la creazione di un oggetto.
- **Tempo dosaggio (s).** È il tempo per prendere i granuli di plastica dalla tramoggia e miscelarli.
- **Cuscino (cm³).** È lo spazio tra la punta della vite in fase di iniezione e la posizione che la vite otterrebbe con la sua massima corsa.
- **Stop vite (cm³).** È l'arretramento della vite dalla posizione di iniezione.
- **Pressione di mantenimento (PM) specifica (bar).** La pressione di mantenimento è la pressione effettuata per mantenere nella sua posizione lo stampo contenente la plastica fusa.
- **Pressione specifica d'iniezione massima (bar).** È la massima pressione esercitata dalla vite durante la fase d'iniezione.
- **Tempo chiusura stampo (s).** È il tempo durante il quale lo stampo rimane chiuso per la solidificazione.
- **Tempo apertura stampo (s).** È il tempo durante il quale lo stampo rimane aperto per espellere il pezzo.
- **Stampo posizione aperta (mm).** Forse è la distanza tra i due elementi dello stampo quando questi si separano per espellere il nuovo oggetto prodotto.
- **Posizione dosaggio stop (cm³).** È la pressione esercitata per mantenere lo stampo nella sua posizione finché il materiale al suo interno si solidifica.
- **Pressione di mantenimento (PM) posizione (cm³).** La pressione di mantenimento è la pressione esercitata per mantenere lo stampo chiuso, il sottoscritto non è riuscito a capire cosa s'intende però con la sua posizione.
- **Z2 temperatura (°C).** È la temperatura di un certo punto della macchina. Non è specificato cosa si intenda per Z2.
- **ZA temperatura (°C).** È la temperatura di un certo punto della macchina. Non è specificato cosa si intenda per ZA.

- **UG1 temperatura (°C)**. È la temperatura di un certo punto della macchina. Non è specificato cosa si intenda per *UG1*.
- **Z1 temperatura (°C)**. È la temperatura di un certo punto della macchina. Non è specificato cosa si intenda per *Z1*.

Il primo file contiene 3 fallimenti, il secondo 4. Gli eventi di guasto vengono dedotti dal fatto che una o più misurazioni non sono riportate nel file (spazio bianco), ciò può anche essere dedotto dal fatto che, sommando all'orario di un ciclo macchina il tempo di ciclo, questo non coincide con l'orario del prossimo ciclo. Oltretutto, dopo il dissesto, viene riportata un'altra riga, con lo stesso valore di ciclo macchina del precedente, contenente le misurazioni non presenti prima. Anche l'ultima riga non presenta tutti i dati, ciò è probabilmente dovuto al fatto che quei valori non sono ancora stati scritti durante la fase di scaricamento (poiché essi sono estratti quando la macchina sta lavorando); per questo motivo non bisogna considerare la fine del file come un fallimento. Il guasto non viene registrato da nessuna parte ma può essere dedotto solamente dallo studio dei dati, non è quindi possibile conoscere la sua ragione o tipologia. Per ipotesi si decide che la macchina non viene mai spenta dagli operai, ciò verrebbe trattato come un fallimento poiché non è possibile distinguere i due casi. Proprio perché non si conosce la causa dei guasti e non ci si è potuti confrontare con i manutentori, non si conoscono le misure a cui prestare maggiore attenzione per determinare la salute della macchina.

3.2 Elaborazione e analisi dati

Prima di iniziare ad addentrarsi nell'applicazione vera e propria per ottenere dei risultati, verranno studiati i dati ottenuti in modo da avere una maggiore coscienza di essi ed effettuare il lavoro successivo in base a ciò. Quindi questa parte sarà utile per le prossime sezioni dove si andrà nello specifico cercando di rispondere alle domande proposte in Descrizione del problema.

Per prima cosa, in entrambi i file vengono eliminate alcune righe inutili ovvero una certa parte finale del documento. Innanzitutto l'ultima riga non contiene tutte le misurazioni perché i dati vengono scaricati quando la macchina sta lavorando, essa è incompleta poiché il documento viene scaricato quando i sensori non hanno ancora finito il loro lavoro per quel ciclo macchina oppure non hanno ancora scritto il valore ottenuto in memoria. Essendo che la macchina inizia un ciclo appena finito quello precedente, è difficile avere l'ultima riga completa perciò va eliminata. Oltretutto la sua cancellazione è importante perché la riga potrebbe venir confusa con un fallimento visto che non contiene alcune misurazioni, il metodo usato per riconoscere i guasti è spiegato in Raccolta e struttura dati. Vengono anche eliminate tutte le righe dall'ultimo fallimento contenuto nel file in poi, ciò perché non è presente il guasto legato a questi dati, quindi non si conosce il loro RUL o stato di salute esatto e, di conseguenza, sono inutili.

Una volta aver escluso i dati come appena riportato, i file, per essere accettati dall'applicazione, non devono contenere l'intestazione. Possono venir utilizzati lasciando unicamente le righe contenenti le misurazioni oppure ponendo come prima riga il nome dei campi separato da una virgola. Per ottenere un unico flusso dati i due file devono essere concatenati. Per effettuare l'operazione, bisognerà eliminare l'intestazione dal documento che verrà posto in coda. Siccome i file presentano il comportamento della macchina a distanza di poco più di 2 mesi, i loro valori hanno medie diverse, segno del fatto che la macchina sta lavorando in modo leggermente diverso. Come si vedrà nella fase di sperimentazione, l'unione dei dati non ha comportato problemi legati a questo motivo.

Le righe con valori mancanti verranno eliminate. Ciò avverrà dopo come spiegato nelle successive sezioni, ciò non viene effettuato prima perché esse rappresentano i fallimenti e quindi sono indispensabili per la creazione delle variabili target. Normalmente non si dovrebbero avere righe duplicate perché la macchina riporta i nuovi valori ottenuti per ogni ciclo, in ogni caso si può stabilire che due righe sono duplicate se presentano gli stessi valori di *Data* e *Tempo*. Quest'ultimi sono gli unici identificativi di ogni ciclo visto che i dati di *Cicli* possono ripartire da 1 ottenendo

due righe con lo stesso valore. Per i fallimenti avviene un caso particolare, la macchina riporta una seconda volta lo stesso valore di *Cicli* per fornire le misurazioni non scritte nella riga precedente a causa del guasto; entrambe le righe vengono comunque eliminate perché non contengono tutti i valori necessari. Si avrà quindi che il primo file contiene 8140 righe (più una di intestazione), il secondo da 15237 (più intestazione) e la loro concatenazione 23377 (più intestazione).

Tutte le colonne dei file sono mantenute senza modifiche tranne per *Data* e *Tempo* le quali vengono aggregate in *Timestamp*. In questo modo l'identificativo del tempo è contenuto in un'unica colonna utilizzando il formato standard *yyyy-MM-dd hh:mm:ss* (anno-mese-giorno ore:minuti:secondi) anziché *dd.MM.yy* e *hh:mm:ss*. In questo modo è possibile stampare facilmente dei grafici usando come asse X il tempo e come Y una certa misura. Tutti i file hanno 18 colonne. Verrà poi aggiunta la variabile *failure number* dall'applicazione stessa durante la creazione del target. Questa rappresenta un valore intero positivo che identifica univocamente il fallimento, ciò è utile nei grafici per stampare, con diversi colori, i vari comportamenti a seconda del guasto in modo tale da distinguerli facilmente.

Le misurazioni sono quasi tutte valori continui quindi non si hanno problemi dovuti ai dati categorici per l'implementazione dei modelli di apprendimento. *Data*, *Tempo* e *Timestamp* non sono continui ma non verranno utilizzati per effettuare le previsioni quindi non creano noia. È bene controllare che tutti i valori presi come input abbiano il formato che ci si aspetta. *Data* deve essere del tipo *dd.MM.yy*, *Tempo* deve seguire il pattern *hh:mm:ss*, i valori appartenenti a *Cicli* devono essere interi positivi, tutti gli altri numeri reali positivi oppure spazio vuoto (in caso di fallimento). Nei file ottenuti, tutti i cicli rispettano il formato appena specificato. Oltretutto si è notato che, nel secondo file, dopo il ciclo 277323 il conteggio di *Cicli* riparte da 1 anziché continuare con 277324. È quindi possibile che, ad un certo punto o secondo certe condizioni, la macchina riprenda l'enumerazione dei cicli da 1 perciò il valore di *Cicli* non identifica univocamente il ciclo. Di conseguenza, l'unico identificativo valido è la combinazione di *Data* e *Tempo* ovvero *Timestamp*. Siccome non è conosciuta l'origine dei fallimenti e non si è potuto comunicare con i manutentori effettivi della macchina, non si conoscono quali misure sono teoricamente più importanti delle altre perciò verranno inizialmente trattate tutte con la stessa rilevanza.

Ecco in tabella 3.1 una panoramica dei dati ottenuti come input. Da essa è possibile vedere la differenza tra i due file.

In figura 3.8 sono riportate le misurazioni in dei grafici di dispersione, l'asse Y presenta il valore della misura, l'asse X il numero di cicli macchina prima del fallimento (RUL). A sinistra si avrà RUL nullo ovvero ci si avvicina al fallimento, a destra si mostra il valore massimo quindi la macchina si è appena ripresa dal guasto precedente e si è rimessa a lavoro. Ogni curva corrisponde ad un fallimento diverso, quelli identificati come 1, 2 e 3 derivano dal primo file, gli altri dal secondo. L'asse X è troncato a massimo 4500 cicli, questo perché solo il fallimento numero 6 va oltre questa soglia (arriva fino a 8000) e quindi si è preferito effettuare un taglio per mostrare in modo più chiaro tutte le curve.

Osservando i grafici 3.8 si nota che la maggior parte delle curve sono linee rette e strette, questo perché la macchina lavora nel suo stato normale e quindi i parametri rimangono fissi e stabili fino al fallimento. Le misure che sembrano più interessanti per conoscere la salute del dispositivo sono *PM pressione* e *p-ini mas*, le loro curve hanno un andamento meno stabile e una deviazione più accentuata rispetto alle altre. In particolare presentano dei casi d'interesse nella zona vicino il guasto (a sinistra) e quando la macchina si riprende dal fallimento precedente (a destra). Anche *Cuscino* presenta una rilevante deviazione però mantiene la sua media costante, non ha andamenti verso l'alto o il basso. Si può dedurre che i dati non seguono una forte regressione ma la macchina mantiene uno stato di salute normale per quasi tutto il tempo.

Per poter osservare meglio l'andamento della macchina vicina ai fallimenti, in figura 3.9 sono riportati i grafici di dispersione come quelli in 3.8 ma mostrando solo la parte più vicina ai guasti. L'asse Y rappresenta il valore delle misurazioni, l'asse X mostra il numero di cicli macchina prima del fallimento (RUL). Come già detto, le curve 1, 2 e 3 derivano dal primo file, le rimanenti dal secondo. L'asse X è tagliato a massimo 400 cicli in modo da focalizzare la visualizzazione sul momento appena prima del dissesto.

Misura	File	Media	Minimo	Massimo	Intervallo	Varianza	Deviazione standard
Tempo ciclo	Primo	23.838	23.818	55.245	31.427	0.272	0.521
	Secondo	23.522	23.388	55.264	31.876	0.257	0.507
Tempo iniezione	Primo	1.149	1.146	1.158	0.012	0.00002	0.005
	Secondo	0.794	0.663	1.147	0.484	0.459	0.214
Tempo dosaggio	Primo	3.454	3.086	33.934	30.848	0.116	0.340
	Secondo	3.377	2.954	3.85	0.896	0.013	0.112
Cuscino	Primo	12.467	2.66	16.13	13.47	0.045	0.212
	Secondo	11.772	8.93	15.46	6.53	0.076	0.275
Stop vite	Primo	86.374	86.22	86.9	0.68	0.075	0.273
	Secondo	74.605	70.32	86.23	15.91	49.814	7.058
PM pressione	Primo	1317.089	1240.0	1521.3	281.3	63.084	7.943
	Secondo	1163.130	1125.7	1394.1	268.4	71.537	8.458
p-ini mas	Primo	1332.918	1237.2	1521.3	284.1	82.053	9.058
	Secondo	1190.877	1125.7	1403.6	277.9	162.065	12.730
Tempo chiusura sta	Primo	1.85	1.85	1.85	0.0	0.0	0.0
	Secondo	1.843	1.84	1.85	0.01	0.00002	0.004
Tempo apertura sta	Primo	1.99	1.99	2.0	0.01	0.00000001	0.0001
	Secondo	2.268	1.99	2.37	0.38	0.028	0.169
Stampo posizione aperta	Primo	300.0	300.0	300.0	0.0	0.0	0.0
	Secondo	285.387	280.0	300.0	20	78.724	8.872
Posizione dosaggio stop	Primo	76.641	29.51	77.19	47.68	0.349	0.590
	Secondo	62.525	57.36	76.52	19.16	71.969	8.483
PM pos	Primo	17.444	17.43	17.49	0.06	0.00004	0.007
	Secondo	17.436	17.41	17.49	0.08	0.00006	0.008
Z2 temp	Primo	254.999	254	255	1	0.001	0.033
	Secondo	258.655	252	261	9	4.930	2.220
ZA temp	Primo	60.0	60	60	0	0.0	0.0
	Secondo	60.0	60	60	0	0.0	0.0
UG1 temp	Primo	265.017	264	267	3	0.030	0.172
	Secondo	265.349	261	271	10	1.634	1.278
Z1 temp	Primo	254.994	250	259	9	0.046	0.214
	Secondo	258.644	250	261	11	4.979	2.231

Tabella 3.1. Analisi dei dati ottenuti

Come già notato con i grafici 3.8, anche in figura 3.9 si hanno, in maggioranza, delle curve rette e fine, segno del fatto che la macchina lavora in uno stato normale per la maggior parte del tempo. *Cuscino* ha una certa dispersione ma non si allontana mai dalla sua media. *PM pressione* e *p-ini mas* sono le uniche misure che presentano andamenti differenti vicini al guasto. La curva 3 ha una tendenza verso il basso, la numero 7 ha un comportamento particolare (decresce, ha un picco verso l'alto e decresce ancora), i fallimenti 1, 2, 4, 5, 6 non hanno un andamento ben definito e rimangono abbastanza stabili. In ogni caso nessuna misura si mostra con una regressione fissa e ben definita, ciò comporterà un lavoro difficile di predizione vista la mancanza di un pattern netto e ben stabilito.

Per fare in modo che i dati siano nello stesso intervallo di valori così da essere più facilmente capibili dai modelli di apprendimento, bisogna applicare delle tecniche di normalizzazione. Solitamente per questo lavoro le due formule più utilizzate sono quelle di ridimensionamento (scaling) e standardizzazione (standardization):

$$X_{ridim} = \frac{X - X_{min}}{X_{max} - X_{min}} \quad X_{standard} = \frac{X - \mu}{\sigma}$$

dove X sono tutti i dati di una misura, X_{ridim} e $X_{standard}$ sono il risultato ridimensionato e standardizzato, X_{min} e X_{max} sono rispettivamente il minimo e il massimo valore contenuto in X , μ e σ sono la media e la deviazione standard di X . Il ridimensionamento scala tutti i dati in modo che essi siano compresi tra 0 e 1 mantenendo inalterate le loro proporzioni. La standardizzazione invece serve per portare la media a 0 e la varianza a 1. I grafici ottenuti dai dati ridimensionati e standardizzati sono estremamente simili a quelli ricavati dalle misurazioni non alterate perciò non verranno riportati per evitare ridondanza, l'unica differenza è l'intervallo in cui sono comprese le

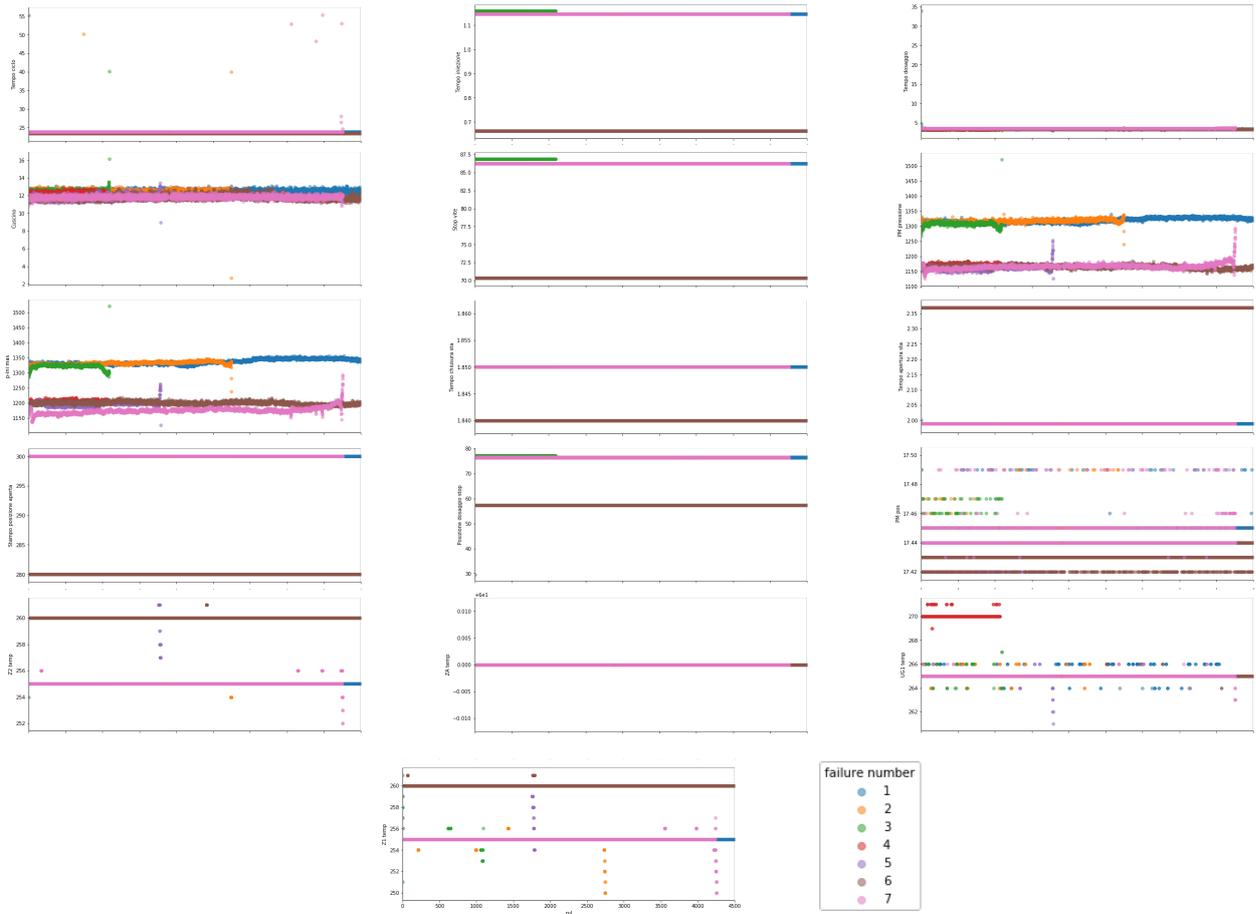


Figura 3.8. Misurazioni rappresentate in grafici di dispersione, ogni colore corrisponde ad un fallimento. Nell’asse Y si ha il valore delle misurazioni, sull’asse X il numero di cicli macchina prima del fallimento (RUL). Troncamento a 4500 cicli macchina.

curve. Un’evoluzione è l’utilizzo delle stesse tecniche ma applicate in modo locale su ogni fallimento ottenendo le seguenti espressioni matematiche:

$$X_{ridim}^{\#fail} = \frac{X^{\#fail} - X_{min}^{\#fail}}{X_{max}^{\#fail} - X_{min}^{\#fail}} \quad X_{standard}^{\#fail} = \frac{X^{\#fail} - \mu^{\#fail}}{\sigma^{\#fail}}$$

il concetto è lo stesso delle versioni precedenti solo che $X^{\#fail}$ non è l’insieme di tutti i valori di una misura ma raccoglie solo i dati legati ad un fallimento. Di conseguenza il massimo, minimo, media e deviazione standard sono calcolati localmente per ogni guasto. La variabile *failure number* presentata qualche capoverso fa risulta quindi utile per il caso in questione. Né il ridimensionamento locale né la standardizzazione locale cambiano in modo rilevante l’andamento delle curve ma solamente le traslano in direzione verticale. Con la prima trasformazione tutti i dati sono compresi tra 0 e 1, con la seconda la media di ogni curva è uguale zero. Unico cambiamento con una certa importanza è il grafico di *Tempo dosaggio* a seguito del ridimensionamento locale, questo ha una dispersione maggiore rispetto a prima nei fallimenti del secondo file, simile a quella delle curve di *Cuscino* in figura 3.8.

Vediamo la distribuzione dei valori ottenuti. Come già detto, il primo e il secondo file hanno dati con una media diversa di conseguenza è meglio consultare la distribuzione dei due file separati e poi assieme. In figura 3.10 sono mostrati gli istogrammi ottenuti dal secondo documento. Il primo file presenta distribuzioni molto simili tranne che per la misura *Tempo dosaggio* che viene

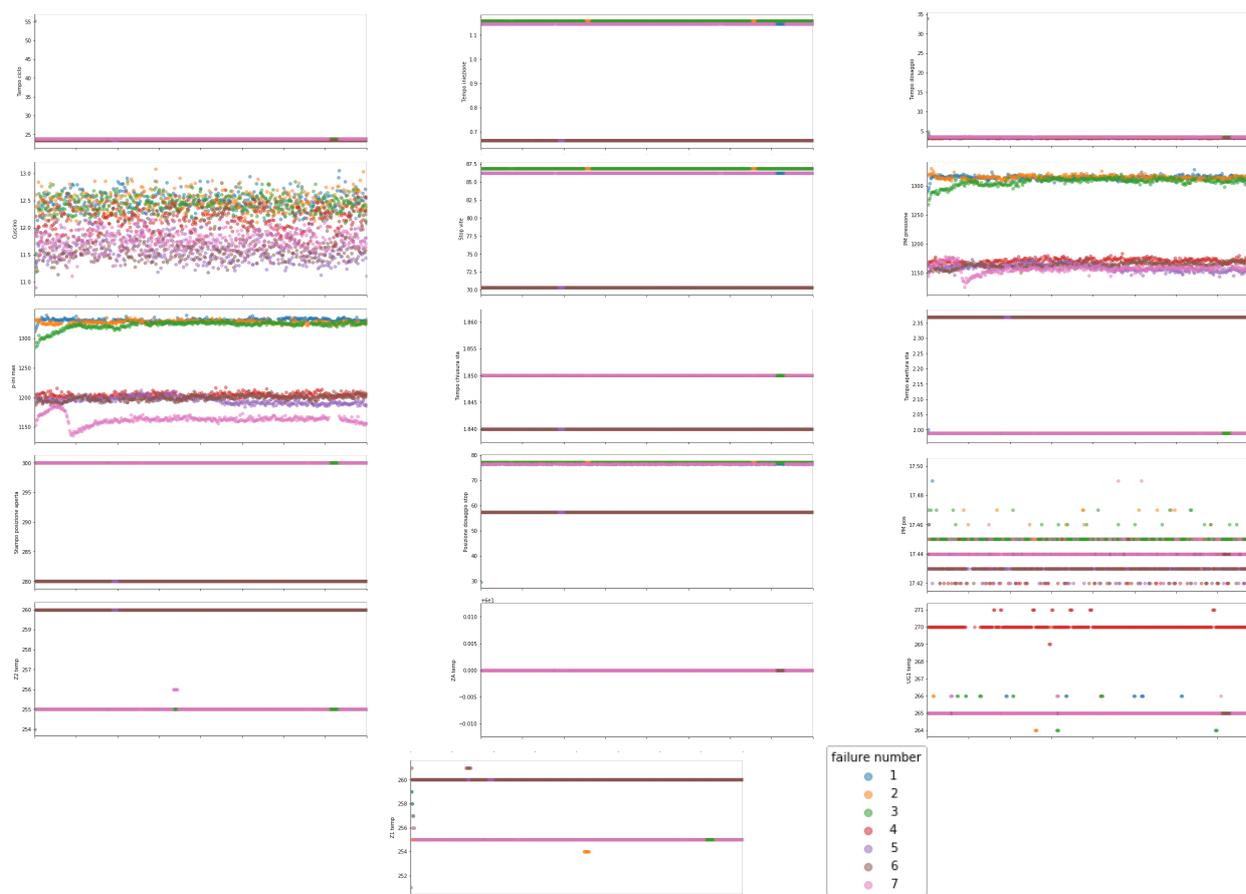


Figura 3.9. Misurazioni rappresentate in grafici di dispersione, ogni colore corrisponde ad un fallimento. Nell'asse Y si ha il valore delle misurazioni, sull'asse X il numero di cicli macchina prima del fallimento (RUL). Troncamento a 400 cicli macchina.

visualizzata come una singola barra, inoltre *Cuscino* ha una deviazione standard minore. Rispetto alla concatenazione di tutti i dati, *Tempo dosaggio* è presentata sempre con un'unica barra, *PM pressione* e *p-ini mas* presentano due gaussiane, quella a destra dovuta al primo file, l'altra al secondo.

Come già si poteva dedurre dai grafici di dispersione riportati precedentemente, la maggior parte delle misure ha una dispersione poco accentuata formata da una o due barre invece *Cuscino*, *PM pressione* e *p-ini mas* hanno una distribuzione normale. Unico caso particolare è *Tempo dosaggio* che nel grafico presentato è composto da due gaussiane invece nel primo file costituisce un'unica barra. Si evidenzia di nuovo il fatto che la macchina rimane per la maggior parte del tempo stabile mantenendo le sue caratteristiche il meno possibile alterate nel tempo.

In figura 3.11 sono mostrati i diagrammi a scatola e baffi (o box plot) del secondo file. Questi sono abbastanza simili a quelli del primo file, solo il *Tempo dosaggio* presenta vari valori fuori soglia (outliers) a differenza del grafico riportato in figura. Nella concatenazione dei due documenti, vengono presentati molti meno outliers poiché le scatole sono più grandi.

Come già deducibile dai grafici di dispersione e dalle distribuzioni riportate sopra, ci sono pochi valori fuori soglia. Solo *Cuscino*, *PM pressione* e *p-ini mas* presentano molti outliers, le altre misurazioni sono stabili con una bassa dispersione dei dati.

L'unica correlazione di una certa rilevanza tra le misure è il rapporto di *p-ini mas* con *PM pressione*, esse sono direttamente proporzionali.

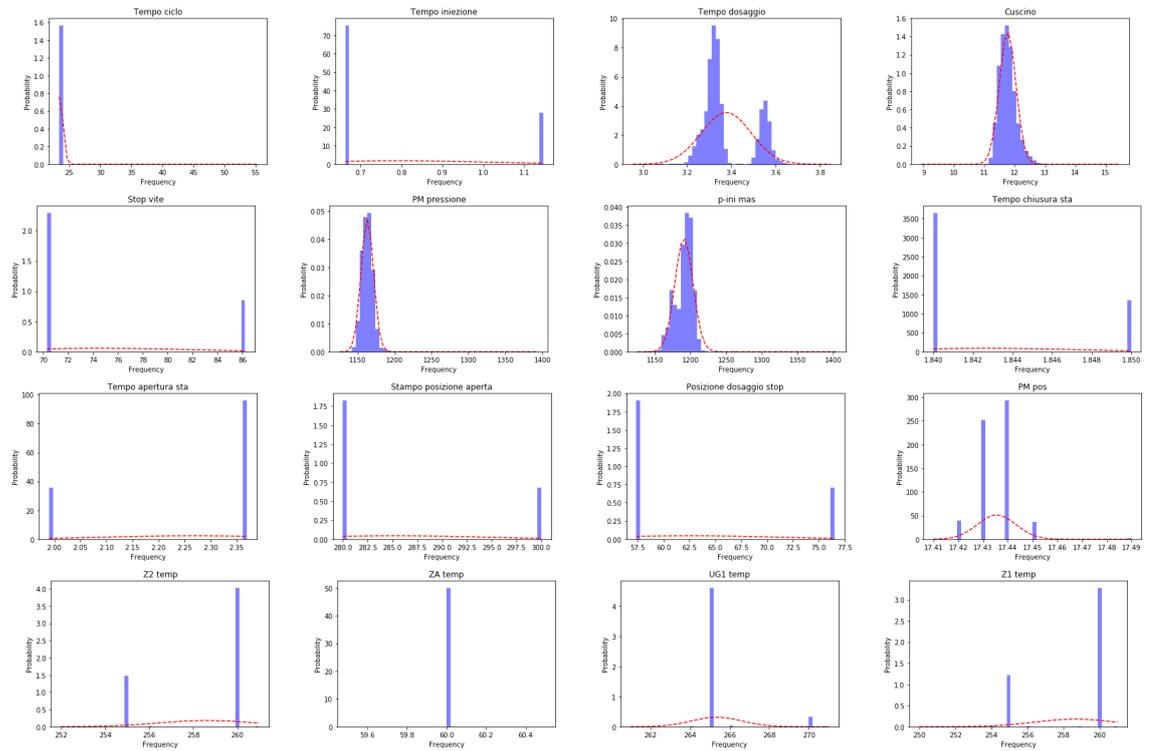


Figura 3.10. Distribuzione delle misurazioni contenute nel secondo file

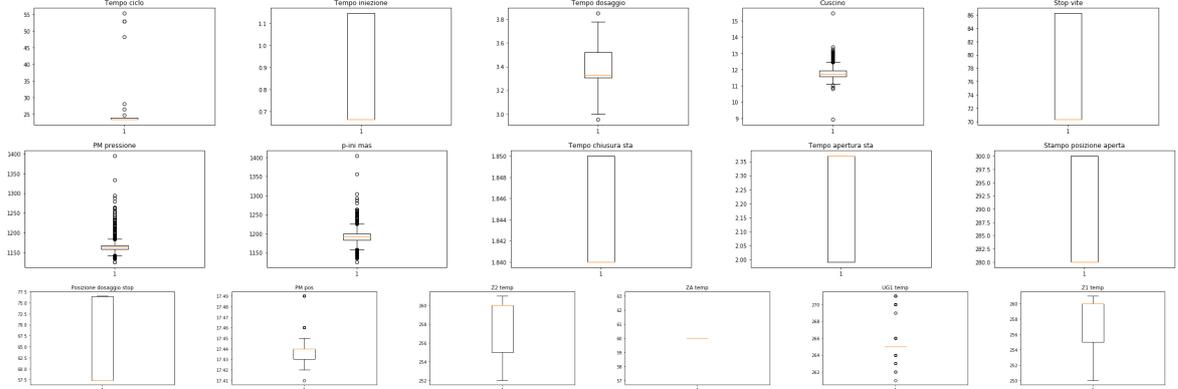


Figura 3.11. Diagrammi a scatola e baffi delle misurazioni contenute nel secondo file

Per l'allenamento dei modelli verranno usati inizialmente tutti i campi contenuti nei file tranne *Timestamp* e *Cicli*. Quest'ultimo viene inizialmente scartato perché la maggior parte dei modelli non tengono conto della posizione (e quindi degli identificativi) dei dati preferendo una loro particolare aggregazione; verrà studiata in seguito la sua introduzione. *Timestamp* non viene considerato poiché ogni ciclo non deve dipendere dal tempo in cui è stato eseguito. Vengono anche eliminate le nuove colonne create per una visualizzazione migliore dei dati (come *failure number* e *rul* nei casi in cui non è la variabile target). Si analizzerà poi se l'aggiunta di nuovi campi o l'eliminazione di alcune misure comporta un miglioramento nel risultato finale.

Per imitare un caso reale, i modelli verranno allenati usando i primi 6 fallimenti in ordine di tempo e la fase di test verrà applicata sul settimo e ultimo guasto. In questo modo si ipotizza di avere un certo storico di dati e dover prevedere il prossimo dissesto mentre il macchinario sta

lavorando. In particolare i test non sono stati eseguiti prendendo dei dati casuali tra quelli usati per l'allenamento poiché ciò causava un forte overfitting e, oltretutto, non si rispecchiava il caso reale. Se ne deduce che i fallimenti sfruttati per l'allenamento dei modelli devono essere necessariamente diversi da quelli del test e, possibilmente, anche anteriori nel tempo.

3.3 Quesito 1 - La macchina avrà un fallimento nei prossimi N secondi/ore/giorni/cicli macchina?

Il primo obiettivo dell'applicazione sviluppata è riconoscere se la macchina avrà un fallimento in un certo intervallo di tempo a partire da quando viene interrogata. In questo caso il tempo è identificato come N cicli macchina dove N è un numero intero positivo. Siccome la media del *Tempo ciclo* è, nel primo file, 23.8 secondi e, nel secondo documento, 23.5, si è ipotizzato di avere dei cicli di minimo 20 secondi; perciò i test sono stati effettuati (se non specificato diversamente) su 360 cicli ovvero un intervallo di almeno 2 ore. Naturalmente verranno provati anche altri valori di N in modo da vedere le prestazioni della macchina in base alle sue variazioni. L'applicazione finale dovrà fare in modo che l'utente possa selezionare lui stesso il valore di tempo (in cicli macchina o minuti/ore) per risultare maggiormente flessibile; ciò non è stato implementato perché essa non è stata ancora integrata con il resto del sistema descritto in Implementazione soluzione. L'output è un valore finito binario (0/1, sì/no, vero/falso) quindi abbiamo un problema di classificazione binaria come presentato in Classificazione e regressione. I due risultati finali verranno chiamati, per essere più facilmente riconoscibili, *Stato normale* (la macchina ha un comportamento senza anomalie quindi non fallirà entro N cicli macchina) e *Vicino fallimento* (si è previsto un guasto entro l'intervallo di tempo definito).

La figura 3.12 mostra i vari processi da sperimentare per arrivare al risultato finale. Essa rappresenta l'idea iniziale, è possibile che alcuni blocchi vengano eliminati o modificati durante il lavoro di sperimentazione. In figura 3.17, nella sezione Quesito 1 - Soluzione finale, è mostrata la sequenza finale.

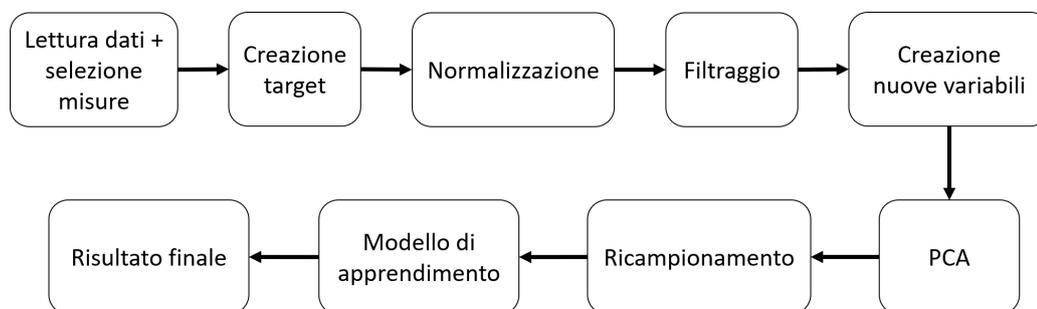


Figura 3.12. Quesito 1 - Sequenza di processi da effettuare per arrivare al risultato (stadio sperimentale)

3.3.1 Quesito 1 - Problema dei dati sbilanciati

Per il quesito posto in questa sezione, il maggior problema è dovuto allo sbilanciamento dei dati [137]. Questo avviene quando si vuole applicare una classificazione ma i valori in input hanno un rapporto sbilanciato tra le classi perciò i modelli non lavoreranno nel modo appropriato dando la precedenza alle etichette con una maggiore presenza. Non esiste una quantificazione predefinita da cui si può parlare di dati sbilanciati, solitamente la si attesta con uno sbilanciamento uguale o peggiore di 80%/20% (o rapporto 4:1) ma in realtà dipende dai dati in esame. Per esempio, si vuole applicare una classificazione binaria a dei dati i quali per il 90% sono associati alla prima etichetta, si ha quindi un rapporto 9:1 tra le due classi, la seconda etichetta verrà così discriminata.

Ciò perché i modelli di apprendimento trovano comodo predire sempre la prima etichetta visto che il 90% delle volte la valutazione sarà corretta. Lo stesso problema si può avere utilizzando una quantità maggiore di classi ma esso è più facilmente riconoscibile con due.

Nel nostro caso, usando 180 cicli come N sui dati di allenamento del modello, si hanno 1033 (circa 5.36%) valori di tipo *Vicino fallimento* e 18240 (circa 94.64%) con etichetta *Stato normale* quindi un rapporto di circa 18:1. Applicando una standardizzazione locale, allenando il modello **logistic regression** e usando il fallimento 7 come test, si ottiene una matrice di confusione come in tabella 3.2. Da essa si otterrebbe un'accuratezza del 95.6% tramite la formula [138] [139]

$$Accuratezza = \frac{VP + VN}{VP + FN + FP + VN} = \frac{3925 + 0}{3925 + 0 + 0 + 179} = 95.6\%$$

dove: VP = vero positivo, FP = falso positivo, VN = vero negativo, FN = falso negativo. Nell'esempio in esame, P è *Stato normale* (positivo), N è *Vicino fallimento* (negativo), V sta per "vero" ovvero se la predizione è corretta, F significa "falso" e indica che la predizione è errata. Quindi si otterrebbe un'alta accuratezza nonostante i fallimenti non vengano mai predetti. Per questo motivo il modello di apprendimento pensa di aver lavorato bene. Questo problema è detto paradosso dell'accuratezza [138].

		Classi reali	
		Stato normale (P)	Vicino fallimento (N)
Classi predette	Stato normale (P)	3925 (VP)	0 (FN)
	Vicino fallimento (N)	179 (FP)	0 (VN)

Tabella 3.2. Quesito 1 - Matrice di confusione ottenuta da un **logistic regression** usando per l'allenamento dei dati sbilanciati

L'accuratezza è quindi il criterio peggiore per stabilire la qualità delle previsioni ottenute in output quando si utilizzano dati sbilanciati; il primo passo per superare il paradosso dell'accuratezza è utilizzare le giuste misure per capire se il modello di apprendimento stia lavorando come ci si aspetta. Ecco i principali criteri su cui fare affidamento [139] [140]:

- **Precisione.** È calcolata tramite la formula $Precisione_{pos} = \frac{VP}{VP+FP} = \frac{3925}{3925+179} = 95.6\%$ per la classe positiva, $Precisione_{neg} = \frac{VN}{VN+FN} = \frac{0}{0+0} = 0.0\%$ per la negativa. In questo caso la precisione è alta per la classe maggioritaria ma nulla per quella minoritaria quindi possiamo già capire che il modello non presenta un buon risultato. Questa è una misura di correttezza poiché mostra la percentuale di dati a cui è stata assegnata una certa etichetta che appartengono realmente a quella classe.
- **Recupero o Richiamo (recall).** Dato dalla formula $Recupero_{pos} = \frac{VP}{VP+FN} = \frac{3925}{3925+0} = 100.0\%$ per la classe positiva e $Recupero_{neg} = \frac{VN}{VN+FP} = \frac{0}{0+179} = 0.0\%$ per la negativa. Anche in questo caso si può osservare che si ha un buon risultato solo per la classe maggioritaria a discapito di quella minoritaria. Questa è una misura di completezza o accuratezza poiché mostra la percentuale di dati di una certa classe classificati correttamente alla classe corrispondente.
- **Punteggio F1 (F1 score o F score o F Measure).** Si calcola come $F1 = 2 \cdot \frac{Precisione \cdot Recupero}{Precisione + Recupero}$. Vale 97.8% per *Stato normale* e 0.0% per *Vicino fallimento*. È una via di mezzo tra precisione e recupero.
- **Curva Receiver Operating Characteristic (ROC).** Rappresenta graficamente il rapporto tra i dati etichettati correttamente e quelli errati quindi mostra la capacità di dividere le misurazioni in due classi. Tecnicamente, stampa il rateo dei veri positivi rispetto al rateo dei falsi positivi per tutte le possibili soglie creando così una curva; maggiore è l'inclinazione di quest'ultima, migliore è il risultato ottenuto. Può essere applicata solo ad una classificazione binaria. Per cogliere immediatamente la qualità di una curva è comodo avere un unico

valore numerico ovvero l'**Area Under Curve (AUC)**. Essa è ottenuta calcolando l'area sotto la curva, in questo modo si può ottenere numericamente un criterio per valutare il risultato da cui risulta facile la comparazione tra due ROC.

I criteri a cui si farà più affidamento per la scelta del risultato migliore sono la curva ROC (accompagnata naturalmente dall'AUC) e il punteggio F1. Dalla sperimentazione si è notato che quest'ultimo rappresenta meglio la qualità del modello ottenuto quindi gli verrà data maggiore rilevanza.

Ora che si conosce come valutare il lavoro di un modello di apprendimento e si sa riconoscere il problema in questione, bisogna cercare di correggere lo sbilanciamento dei dati. Non esiste un unico metodo valido per qualsiasi caso quindi eccone alcuni [137] [142] [143] [144] :

- **Sovracampionamento della classe minoritaria (oversampling)**. Questa tecnica consiste nel duplicare, secondo un certo criterio, i dati della classe minoritaria in modo da ottenere la stessa quantità di misurazioni per le due etichette. Il metodo più semplice è la duplicazione casuale ma esistono metodi più complessi. Genera un numero maggiore di valori quindi è possibile avere rallentamenti a causa della loro mole. Un altro suo problema è la possibile generazione di overfitting. Verrà spiegata meglio e applicata in Quesito 1 - Sottocampionamento e sovracampionamento.
- **Sottocampionamento della classe maggioritaria (undersampling)**. Essa consiste nell'eliminazione di alcuni valori della classe maggioritaria in modo da equalizzare la loro quantità con quella minoritaria. Come nel caso precedente, l'approccio più semplice è selezionare in modo casuale i dati da scartare ma esistono algoritmi più efficienti. Se si ha già un problema di scarsità di dati, questi diminuiranno ancor più il loro numero. Un possibile problema è la perdita di informazioni dovuta allo scarto di alcuni dati. Nella sezione Quesito 1 - Sottocampionamento e sovracampionamento verrà presentata al meglio.
- **Modelli di apprendimento penalizzati**. Questi sono algoritmi che aumentano il costo degli errori di classificazione per la classe minoritaria, in questo modo si dà maggiore importanza ai dati minoritari rispetto a quelli maggioritari. Un esempio è l'**SVM** penalizzato il quale verrà trattato e testato in Quesito 1 - Modelli di apprendimento.
- **Algoritmi basati sugli alberi**. Questi sono certi modelli di apprendimento i quali solitamente lavorano bene anche con i dati sbilanciati. Alcuni tra i più famosi sono: **decision tree**, **random forest**, **C4.5**, **C5.0**, **CART**. Essi verranno testati in Quesito 1 - Modelli di apprendimento.

Solamente imponendo N uguale a circa 1100 o maggiore si ottiene un miglior bilanciamento delle classi e i modelli danno una maggiore importanza allo stato *Vicino fallimento* anche se si continua a non produrre importanti risultati. Utilizzando ora una **logistic regression** si ha la matrice di confusione $\begin{bmatrix} 2582 & 455 \\ 536 & 531 \end{bmatrix}$.

3.3.2 Quesito 1 - Creazione della variabile target *Target_N_Cycles*

La variabile da predire (variabile target) è chiamata *Target_N_Cycles*. Questa deve contenere la risposta alla domanda della sezione quindi essa conterrà un valore binario il quale indica o *Stato normale* oppure *Vicino fallimento*. *Target_N_Cycles* è derivato dalla variabile *rul*, questa è spiegata meglio in Quesito 3 - Creazione della variabile target *rul*. Per prima cosa è necessario trovare le righe che indicano il fallimento, queste vengono riconosciute dal fatto che non presentano tutti i valori delle misurazioni ma uno o più campi sono vuoti. Ogni volta che viene trovato un fallimento, si salva il suo valore di *Cicli* come *ciclo_fallimento* e si ottiene, per ogni riga *ciclo_riga_corrente* prima del guasto fino al fallimento precedente o inizio del file, $rul = ciclo_fallimento - ciclo_riga_corrente$. Le

righe dei fallimenti sono ora inutili e vengono quindi scartate, in questo modo si evita il problema di dati nulli o mancanti. Viene allora creata la variabile target come

$$Target_N_Cycles = \begin{cases} 0 & \text{se } rul > N \\ 1 & \text{se } rul \leq N \end{cases}$$

ovvero essa avrà valore 1 se mancano N o meno cicli macchina al fallimento, 0 altrimenti. Anche se la variabile rul non è più realmente utile ai fini dell'applicazione, essa non viene subito eliminata poiché viene sfruttata come asse X per alcuni grafici. Oltretutto verrà creata anche la variabile *failure number* la quale dà ad ogni fallimento un valore numerico univoco in modo da stampare i grafici con curve di colori diversi riconoscendo più facilmente l'andamento per ogni guasto. Siccome si conoscono tutti i possibili output della variabile target, il problema in esame è detto supervisionato [129].

3.3.3 Quesito 1 - Normalizzazione

Le tecniche di normalizzazione utilizzate sono ridimensionamento, standardizzazione, ridimensionamento locale e standardizzazione locale; esse sono state spiegate in dettaglio in Elaborazione e analisi dati. In tabella 3.3 sono mostrati i risultati finali, i vari criteri riportati sono riferiti alla classe minoritaria *Vicino fallimento*. Si è applicato $N = 360$, un filtro Savitzky-Golay di ordine 15, un sovracampionamento casuale e infine i modelli **logistic regression** e **ridge classifier**. Il filtro e il ricampionamento vengono applicati poiché senza essi si otterrebbero prestazioni abbastanza scarse a prescindere della normalizzazione, non si noterebbero quindi grandi differenze tra i vari metodi implementati.

Normalizzazione	Modello	AUC	F1	Recupero	Precisione
Nessuna normalizzazione	Logistic regression	0.99	18%	10%	100%
	Ridge classifier	0.99	0%	0%	0%
Ridimensionamento	Logistic regression	0.50	16%	100%	9%
	Ridge classifier	0.91	16%	100%	9%
Standardizzazione	Logistic regression	0.98	84%	99%	73%
	Ridge classifier	0.97	36%	29%	49%
Ridimensionamento locale	Logistic regression	0.95	16%	100%	9%
	Ridge classifier	0.93	16%	100%	9%
Standardizzazione locale	Logistic regression	0.94	65%	95%	50%
	Ridge classifier	0.97	62%	100%	45%

Tabella 3.3. Quesito 1 - Risultati ottenuti a seconda della normalizzazione effettuata

Dalla tabella 3.3 si possono effettuare varie considerazioni. Innanzitutto la mancanza di normalizzazione comporta risultati abbastanza sbilanciati verso la classe maggioritaria, il valore AUC molto alto può trarre in inganno ma tutti i restanti criteri rimangono troppo bassi. Il ridimensionamento, sia globale che locale, non comporta un miglioramento; esso aumenta il valore di

recupero a discapito della precisione. La standardizzazione invece aumenta in maniera importante la qualità del risultato. Per **logistic regression** abbiamo un maggiore miglioramento grazie alla standardizzazione globale, con **ridge classifier** si preferisce quello locale. Si è però notato che il metodo globale non sempre funziona bene come presentato nella tabella in esame ma ci sono alcuni casi in cui non si hanno risultati così buoni, invece la tecnica locale mantiene molto meglio le sue prestazioni. Per questo motivo si è scelto di applicare, per i prossimi test e per il programma finale, la standardizzazione locale.

3.3.4 Quesito 1 - Filtraggio

Per filtraggio, nell'ambito dell'apprendimento automatico, si intende l'eliminazione (il filtraggio appunto) del rumore dai dati di input [153]. Essendo che questi ultimi sono ricavati da dei sensori, la presenza del rumore è inevitabile quindi è buona pratica applicare un filtro per la pulizia dei valori ottenuti.

Il primo filtro applicato è detto Savitzky-Golay dai suoi creatori [154] [155] [156] [157]. Questo, per ogni valore, considera una certa finestra formata dai dati successivi e precedenti all'elemento in esame e approssima una curva di un grado più basso rispetto all'originale. Il vecchio valore sarà sostituito da quello appartenente alla nuova curva tracciata nella stessa ascissa.

Il secondo filtro che è chiamato Kalman, il suo ideatore [158] [159] [153]. È un filtro ricorsivo pensato per minimizzare l'errore quadratico medio. In particolare lavora molto bene sui sistemi gaussiani. Può lavorare sui dati in tempo reale.

In tabella 3.4 sono mostrati i risultati ottenuti applicando i filtri Savitzky-Golay (usando approssimazioni di grado 2, 5, 10 e 15) o Kalman. Il fallimento 7 è stato usato per il test, i rimanenti per l'allenamento dei modelli. Si è scelto $N = 360$, standardizzazione locale e sovracampionamento casuale. Quest'ultimo è stato applicato poiché, senza di esso, si ottengono sempre scarsi risultati dovuti allo sbilanciamento dei dati.

Analizzando la tabella 3.4, il filtro Kalman non migliora di tanto le prestazioni rispetto a quelle ottenute senza la sua applicazione. Oltretutto esso è lento, impiega 1032 secondi per essere eseguito sui dati di allenamento, il suo concorrente ne impiega solamente 3. Il filtro Savitzky-Golay incrementa tutti i criteri notevolmente, in particolare essi aumentano con il grado delle curve utilizzate dall'algoritmo fino al quindicesimo, poi i risultati si stabilizzano ottenendo all'incirca gli stessi valori finali. Oltretutto questo riesce a correggere parzialmente il problema dei dati sbilanciati. Se ne può dedurre che una semplificazione dei dati comporta un miglior output ma, se essa è eccessiva, si ha una perdita d'informazioni. Perciò, da ora in poi, l'applicazione utilizzerà il filtro Savitzky-Golay di grado 15. Un altro beneficio ottenuto è la visualizzazione dei dati, essi hanno una dispersione molto minore rispetto a prima rendendo la loro analisi visiva molto più semplice per l'uomo, i valori troncati a 400 cicli macchina sono presentati in dei grafici di dispersione in figura 3.13. Ad ogni modo non si riescono ad intuire ad occhio rilevanti andamenti delle curve.

3.3.5 Quesito 1 - Principal Component Analysis (PCA)

La **Principal Component Analysis (PCA)** [145] [146] [147] è una tecnica per ridurre la dimensionalità dei dati. Essa consiste nel rappresentare un insieme di elementi di dimensione M in uno spazio minore di M , ciò avviene grazie ad una proiezione dei valori dallo spazio originale allo spazio delle componenti principali. Queste ultime sono le dimensioni che raccolgono al meglio la correlazione e la varianza (quindi l'informazione) delle variabili. Essendo che le varie misure sono accorpate, si ha una perdita d'informazioni, bisogna quindi controllare se vale la pena ridurre la qualità del risultato finale per guadagnare in termini di tempo risparmiato poiché viene processata una quantità inferiore di elementi. In particolare, si ha una minore perdita d'informazioni maggiore è la correlazione delle variabili, due misure fortemente dipendenti possono venire aggregate senza alcun problema.

Filtro	Modello	AUC	F1	Recupero	Precisione
Nessun filtro	Logistic regression	0.65	7.48%	4.25%	31.25%
	Ridge classifier	0.71	20.76%	79.60%	11.94%
Savitzky-Golay grado 2	Logistic regression	0.82	22.38%	35.12%	16.42%
	Ridge classifier	0.93	39.31%	100.00%	24.46%
Savitzky-Golay grado 5	Logistic regression	0.86	35.91%	71.67%	23.96%
	Ridge classifier	0.95	50.86%	100.00%	34.11%
Savitzky-Golay grado 10	Logistic regression	0.96	56.05%	64.31%	49.67%
	Ridge classifier	0.98	63.36%	97.73%	46.88%
Savitzky-Golay grado 15	Logistic regression	0.94	65.37%	64.31%	49.67%
	Ridge classifier	0.97	61.77%	100.00%	44.68%
Kalman	Logistic regression	0.75	7.61%	4.25%	36.59%
	Ridge classifier	0.75	32.38%	46.74%	24.77%

Tabella 3.4. Quesito 1 - Risultati ottenuti a seconda del filtro applicato

I test che vengono ora presentati sono ricavati applicando $N = 360$, standardizzazione locale, filtro Savitzky-Golay di grado 15, sovracampionamento casuale e, infine, i modelli **logistic regression** e **ridge classifier**. In figura 3.14 sono mostrati i dati, dopo essere stati normalizzati, filtrati e ricampionati, in un grafico 2D; a sinistra senza PCA, a destra sfruttando 8 componenti principali; cambiando la quantità di componenti il grafico rimane lo stesso. La tabella 3.5 mostra i risultati ottenuti a seconda del numero di componenti principali applicate per la PCA; il campo *#componenti principali* è il numero di componenti principali selezionato, *Varianza mantenuta* è la frazione di varianza conservata dopo la PCA (e quindi quantità di informazione mantenuta), *Tempo (s)* è l'ammontare in secondi impiegato dai modelli per effettuare l'allenamento.

Dalla tabella 3.5 si può notare che le prestazioni sono sempre minori rispetto al risultato ottenuto senza ricampionamento. Con 8 componenti principali si raggiungono risultati simili di punteggio F1 ma minori di AUC. In particolare si ottiene una qualità pessima con 11 e 14 componenti. Il tempo impiegato con la PCA sfruttando 2 componenti rispetto a quello senza la sua applicazione è minore del 25% nel caso del **logistic regression** e del 30% per il **ridge classifier**. Si ha in ogni caso un'attesa minore grazie alla PCA ma questa non giustifica la perdita di performance, si è quindi deciso di non sfruttare la PCA per l'applicazione finale.

3.3.6 Quesito 1 - Sottocampionamento e sovracampionamento

Il ricampionamento è un metodo per combattere il problema dei dati sbilanciati. Esso consiste nella modifica dei dati in input in modo da distribuire in modo più uniforme le classi che si vogliono studiare. In questo caso verranno utilizzate le tecniche di sottocampionamento (undersampling) e sovracampionamento (oversampling), il primo consiste nell'eliminare alcuni valori della classe maggioritaria, il secondo nell'aggiungere nuove misurazioni nella classe minoritaria, in entrambi

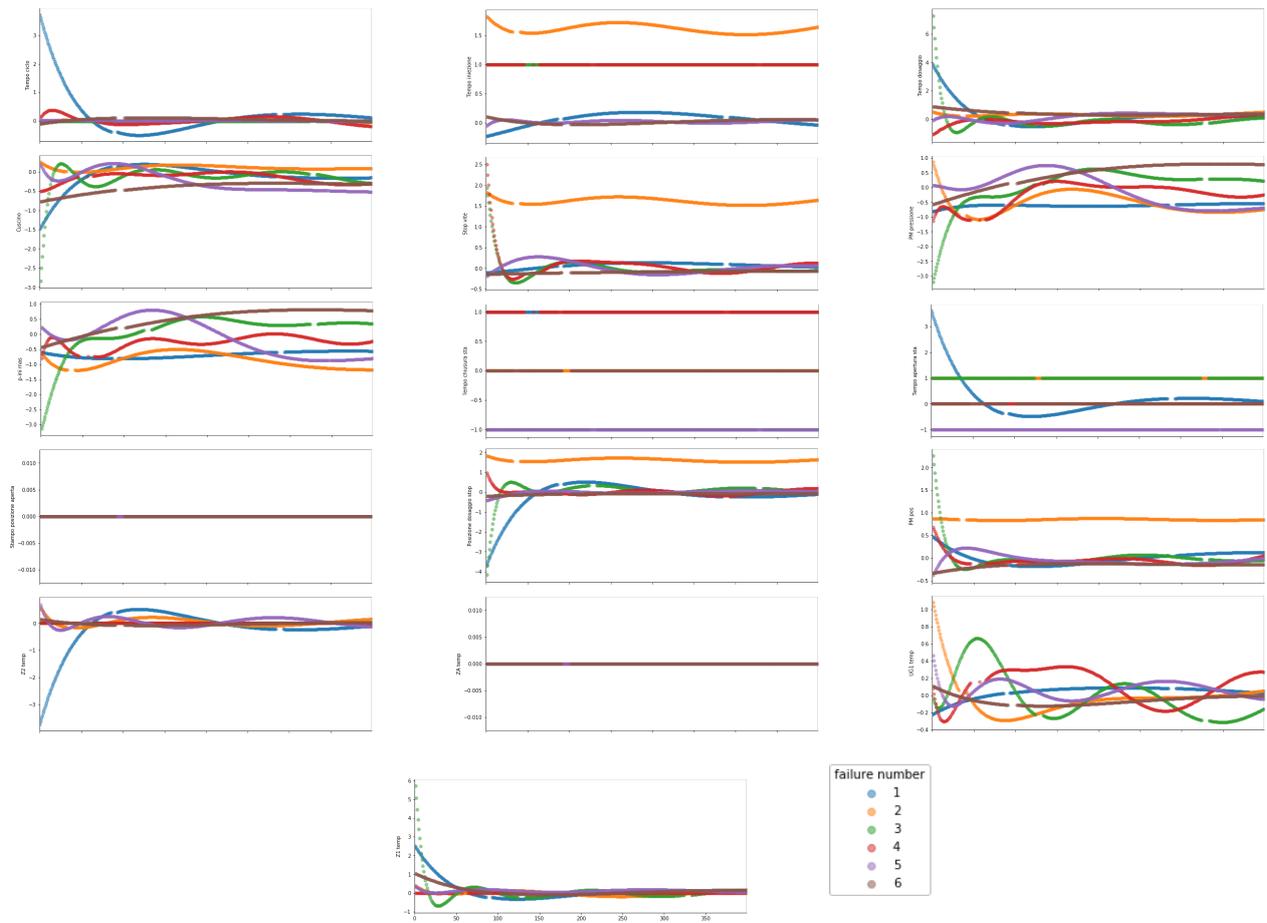


Figura 3.13. Quesito 1 - Primi 6 fallimenti dopo l'applicazione di standardizzazione locale e filtro Savitzky-Golay di grado 15. Troncamento a 400 cicli macchina.

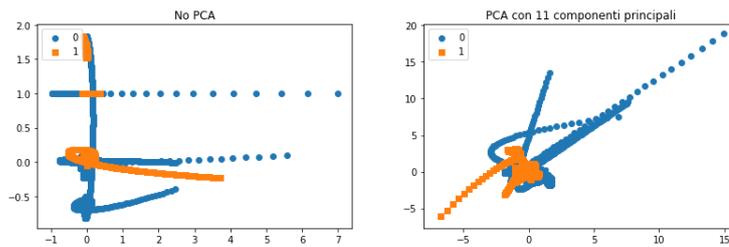


Figura 3.14. Quesito 1 - Dati rappresentati in due dimensioni con e senza l'applicazione della PCA (rispettivamente a destra e sinistra)

i casi si bilanciano le quantità delle due classi [137] [148] [149]. Un esempio grafico è fornito in figura 3.15. Il criterio secondo cui vengono scelti i dati da scartare o aggiungere varia a seconda dell'algoritmo utilizzato. Entrambi i metodi presentano dei difetti, il sottocampionamento può eliminare dati importanti procurando una perdita di informazioni, il sovracampionamento può generare overfitting poiché vengono ripetuti gli stessi valori [150] [151]. La libreria utilizzata nell'applicazione per effettuare il ricampionamento si chiama `imbalanced-learn` [152] [149].

Le tecniche di ricampionamento sono state testate utilizzando $N = 360$, standardizzazione locale e filtro Savitzky-Golay di grado 15. I risultati ottenuti sono presentati in tabella 3.6. Per la figura

#componenti principali	Varianza mantenuta	Modello	Tempo (s)	AUC	F1	Recupero	Precisione
No PCA	1.00	Logistic regression	9.39	0.93	65.05%	94.90%	49.48%
		Ridge classifier	4.25	0.97	61.34%	100.00%	44.24%
2	0.55	Logistic regression	6.99	0.92	36.24%	94.33%	22.42%
		Ridge classifier	3.00	0.92	37.42%	94.33%	23.19%
5	0.62	Logistic regression	7.75	0.80	28.60%	83.57%	17.25%
		Ridge classifier	3.45	0.83	30.03%	86.40%	18.18%
8	0.97	Logistic regression	7.38	0.87	63.50%	81.30%	52.09%
		Ridge classifier	3.05	0.86	63.71%	79.32%	53.23%
11	0.99	Logistic regression	8.01	0.81	0.00%	0.00%	0.00%
		Ridge classifier	3.01	0.87	21.84%	18.41%	26.86%
14	1.00	Logistic regression	8.50	0.88	0.00%	0.00%	0.00%
		Ridge classifier	3.51	0.87	0.00%	0.00%	0.00%

Tabella 3.5. Quesito 1 - Risultati ottenuti a seconda del numero di componenti principali impiegati dalla PCA

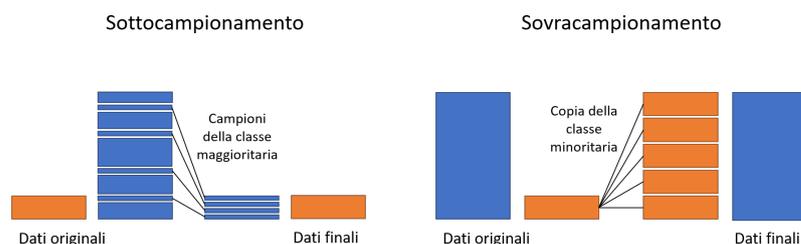


Figura 3.15. Quesito 1 - Schema del lavoro effettuato tramite sottocampionamento e sovracampionamento [151]

3.16 è stata applicata anche la PCA con 8 componenti, questo perché il lavoro di ricampionamento veniva visualizzato meglio; il grafico senza ricampionamento ha titolo *PCA con 8 componenti principali*. Ecco le varie implementazioni utilizzate [151] [149]:

- **Sottocampionamento casuale.** Vengono eliminati alcuni dati della classe maggioritaria in modo casuale per bilanciare le due classi. In figura 3.16 si può notare che molti cerchi blu sono scomparsi, essi sono i dati eliminati.
- **Sovracampionamento casuale.** Alcuni dati, in modo casuale, vengono duplicati finché non si avranno le stesse quantità della classe maggioritaria. In figura 3.16 non si nota alcuna differenza rispetto al grafico senza ricampionamento, ciò è dovuto al fatto che sono stati aggiunti nuovi valori ma, essendo questi uguali a quelli già esistenti, sono sovrapposti.
- **Sottocampionamento Tomek links.** I Tomek links sono delle coppie di dati con valori molto vicini ma appartenenti a classi opposte, essi vengono sostituiti con un unico campione della classe minoritaria in modo da apportare i seguenti benefici: si hanno più elementi della classe minoritaria, meno della maggioritaria e non si hanno elementi simili di classe opposta in modo da facilitare la classificazione [148]. In figura 3.16 non si distinguono grandi differenze rispetto al grafico senza ricampionamento.

- **Sovracampionamento SMOTE.** Vengono generati dei nuovi dati per la classe minoritaria utilizzando la tecnica SMOTE; si prendono alcuni elementi minoritari e si considerano i suoi k nearest neighbors, il nuovo punto verrà aggiunto nel vettore tra il valore in questione e uno dei suoi nearest neighbor. Si può notare in figura 3.16 che esistono alcuni quadratini arancioni in più rispetto al grafico senza ricampionamento.
- **Sovracampionamento SMOTE + sottocampionamento Tomek links.** Viene applicata prima la tecnica SMOTE e, in seguito, il sottocampionamento sfruttando i Tomek links. In figura 3.16 si hanno più quadratini arancioni rispetto al grafico con i dati sbilanciati.

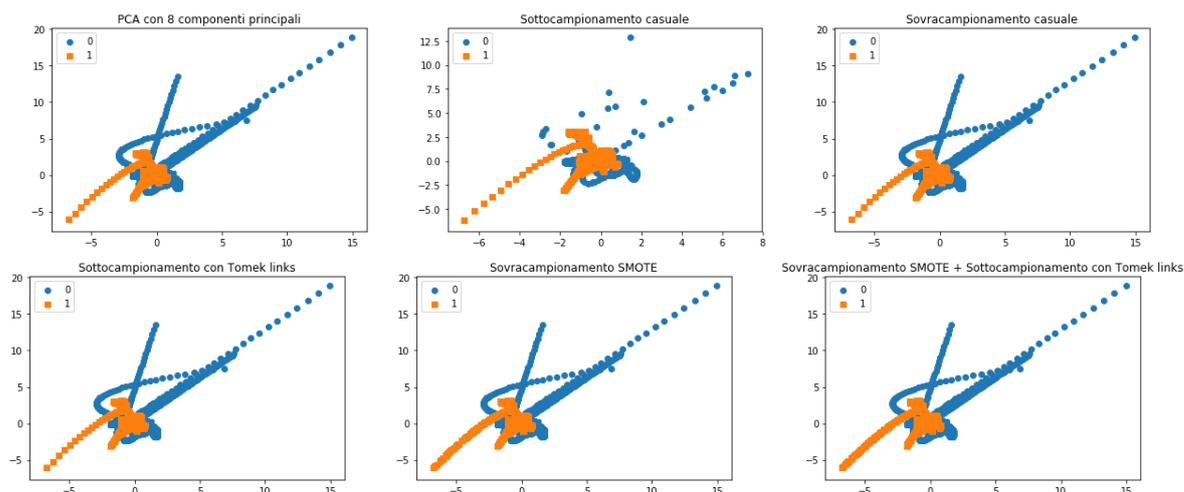


Figura 3.16. Quesito 1 - Risultato delle varie tecniche di ricampionamento

Dalla tabella 3.6 si enuncia che le tecniche migliori sono quelle di sovracampionamento, quest'ultime presentano prestazioni molto simili tra loro. Esse riescono a correggere in modo notevole il problema dei dati sbilanciati. Per l'applicazione finale verrà quindi utilizzato il sovracampionamento.

3.3.7 Quesito 1 - Creazione nuove variabili

Questa sezione viene presentata solo ora, nonostante nella figura 3.12 sia posta tra il filtraggio e la PCA, perché adesso si conoscono tutti i componenti da applicare per la soluzione finale (normalizzazione, filtraggio, ricampionamento) e quindi il risultato ottenuto dai modelli sarà più vicino a quello finale, di conseguenza si possono effettuare delle scelte migliori.

Per migliorare le prestazioni, a volte è utile l'aggiunta di nuove variabili in modo da donare maggiori informazioni al modello. Si è provato ad aggiungere la media e la varianza mobili ottenendo i risultati in tabella 3.7. Si è utilizzato $N = 360$, standardizzazione locale, filtro Savitzky-Golay di grado 15 e sovracampionamento casuale. La finestra utilizzata per il calcolo dei valori mobili è 5 ovvero, per ogni dato, si elabora la media e/o la varianza considerando il dato in questione e i 4 precedenti ad esso. Vengono quindi sfruttati 5 valori per ogni calcolo.

Dalla tabella 3.7 si può evincere che l'aggiunta di media e varianza mobili comporta solamente una perdita di prestazioni. Sono stati riportati i risultati ottenuti utilizzando una finestra di larghezza pari a 5 ma i test sono stati eseguiti anche con diversi valori. All'aumentare della grandezza della finestra si hanno prestazioni peggiori, con il suo valore minimo, 2, si ha una qualità leggermente maggiore. Detto ciò, per l'applicazione finale non verrà aggiunta nessuna nuova variabile.

	Modello	AUC	F1	Recupero	Precisione
Nessun ricampionamento	Logistic regression	0.90	7.81%	7.93%	7.69%
	Ridge classifier	0.94	10.89%	7.08%	23.58%
Sottocampionamento casuale	Logistic regression	0.95	60.34%	100.00%	43.21%
	Ridge classifier	0.96	62.14%	100.00%	45.08%
Sovracampionamento casuale	Logistic regression	0.94	65.51%	96.03%	49.71%
	Ridge classifier	0.97	60.54%	100.00%	43.42%
Sottocampionamento con Tomek links	Logistic regression	0.90	7.81%	7.93%	7.69%
	Ridge classifier	0.94	10.89%	7.08%	23.58%
Sovracampionamento SMOTE	Logistic regression	0.94	65.70%	96.03%	49.92%
	Ridge classifier	0.97	60.86%	100.00%	43.74%
SMOTE + Tomek links	Logistic regression	0.94	65.37%	94.90%	49.85%
	Ridge classifier	0.97	61.23%	100.00%	44.13%

Tabella 3.6. Quesito 1 - Risultati ottenuti a seconda del ricampionamento implementato

	Modello	AUC	F1	Recupero	Precisione
Nessuna nuova variabile	Logistic regression	0.95	66.34%	95.47%	50.83%
	Ridge classifier	0.97	62.92%	100.00%	45.90%
Media mobile	Logistic regression	0.92	58.98%	100.00%	41.82%
	Ridge classifier	0.92	58.54%	100.00%	41.38%
Varianza mobile	Logistic regression	0.92	60.50%	100.00%	43.37%
	Ridge classifier	0.91	58.98%	100.00%	41.82%
Media e varianza mobili	Logistic regression	0.92	61.74%	95.75%	45.55%
	Ridge classifier	0.90	56.43%	100.00%	39.31%

Tabella 3.7. Quesito 1 - Risultati ottenuti a seconda dell'aggiunta di media e varianza mobile con finestra di 5 valori come variabili

3.3.8 Quesito 1 - Selezione misure

Questa sezione viene presentata solo ora nonostante venga eseguita durante l'inizio dell'esecuzione dell'applicazione perché adesso si conoscono tutti i componenti da applicare per la soluzione finale (normalizzazione, filtraggio, ricampionamento, eccetera) e quindi il risultato ottenuto dai modelli sarà più vicino a quello finale, di conseguenza si possono effettuare delle scelte migliori.

Si vuole valutare l'aggiunta del campo *Cicli*. Se si usano come test dei dati presi casualmente da quelli utilizzati per l'allenamento si genera un forte overfitting. Però questo non è il caso in esame visto che l'allenamento e il test adoperano fallimenti diversi e quindi *Cicli* contiene valori differenti. Si è anche provato il caso in cui si usa il fallimento 6 come test (il quale presenta quasi 8400 cicli al contrario di tutti gli altri che arrivano a meno di 4500), i fallimenti da 1 a 5 come allenamento e ciò non comporta overfitting. Nessun problema anche usando i primi tre guasti come allenamento e il settimo come test nonostante essi abbiano valori di *Cicli* molto diversi. Si può quindi dedurre che l'inserimento della variabile in questione non comporta overfitting. Nella tabella 3.8 sono presentati i risultati aggiungendo o scartando l'attributo *Cicli* con $N = 360, 720$ oppure 1080, standardizzazione locale, filtro Savitzky-Golay di grado 15, sovracampionamento casuale e modello `logistic regression`.

	N	AUC	F1	Recupero	Precisione
Senza Cicli	360	0.94	65.89%	96.60%	50.00%
	720	0.89	42.96%	39.89%	46.53%
	1080	0.81	45.96%	45.37%	46.57%
Con Cicli	360	1.00	83.85%	100.00%	72.19%
	720	1.00	95.09%	100.00%	90.64%
	1080	0.98	68.93%	100.00%	52.59%

Tabella 3.8. Quesito 1 - Risultati ottenuti a seconda dell'assenza o presenza del campo *Cicli* per diversi valori di N

La presenza di *Cicli* contribuisce positivamente in modo notevole su tutti i risultati ottenendo un AUC molto alto. Ciò è dovuto al fatto che ora il modello ha un criterio per trattare i dati come una sequenza, infatti se abbiamo un ciclo X associato a *Vicino fallimento* ne consegue che tutti i cicli maggiori di X apparterranno alla medesima classe, lo stesso discorso per un ciclo appartenente a *Stato normale* rispetto ai cicli precedenti. La variabile *Cicli* verrà quindi inclusa assieme alle altre nell'applicazione finale.

Per ora sono sempre state prese in considerazione tutte le colonne dei file, si vuole analizzare se alcune di esse siano inutili o dannose ai fini delle predizioni. Si è quindi utilizzato il modello `logistic regression` per ottenere l'importanza data alle misure durante l'allenamento, questa è rappresentata come un numero reale. I test sono eseguiti con $N = 360$, il campo *Cicli* incluso, standardizzazione locale, filtro Savitzky-Golay di grado 15 e sovracampionamento casuale. Si riportano diversi fallimenti per allenamento e test visto che questi generano differenti risultati. La tabella 3.9 mostra il frutto del lavoro, la colonna *Variabili importanti* elenca le misure a cui viene associata un'importanza maggiore di zero.

Le variabili prese in maggiore considerazione cambiano a seconda dei dati utilizzati, la loro unione è: *Cicli*, *Tempo ciclo*, *Tempo iniezione*, *Cuscino*, *Stop vite*, *p-ini mas*, *UG1 temp* e *Z1 temp*. Il valore dell'importanza non è stato riportato per semplicità ma esso è particolarmente alto per i campi *Cicli* e *Tempo iniezione*. Inoltre si è notato che le prestazioni e il tempo impiegato per l'allenamento dei modelli rimangono entrambi inalterati sia mantenendo unicamente le misure importanti sia includendole tutte perciò l'eliminazione di alcune variabili è inutile. Per questo motivo si continuerà a considerare tutte le colonne disponibili.

Fallimenti per allenamento	Fallimento per test	Variabili importanti
1,2,3,4,5,6	7	Cicli, Tempo ciclo, Tempo iniezione, p-ini mas, UG1 temp
1,2,3,4,5	6	Cicli, Tempo ciclo, Tempo iniezione, Cuscino, Stop vite, p-ini mas, UG1 temp, Z1 temp
1,2,3,4	5	Cicli, Tempo ciclo, Tempo iniezione, Cuscino, Stop vite, p-ini mas, Z1 temp
1,2	3	Cicli, Tempo ciclo, Tempo iniezione, Cuscino, Stop vite, p-ini mas

Tabella 3.9. Quesito 1 - Variabili più importanti a seconda dei fallimenti usati per allenamento e test

3.3.9 Quesito 1 - Modelli di apprendimento

Si vuole ora scoprire quale modello di apprendimento propone le più accurate predizioni. La tabella 3.10 sono visualizzati i risultati con $N = 360$ e 1080 , il campo *Cicli* incluso, standardizzazione locale, filtro Savitzky-Golay di grado 15 e sovracampionamento casuale.

	N	AUC	F1	Recupero	Precisione
Logistic regression	360	1.00	84.15%	100.00%	72.63%
	1080	0.98	68.68%	100.00%	52.30%
Ridge classifier	360	1.00	80.69%	100.00%	67.62%
	1080	0.93	75.87%	100.00%	61.12%
Decision Tree	360	0.71	36.32%	59.21%	26.19%
	1080	0.88	74.63%	100.00%	59.52%
Random Forest	360	0.99	89.17%	80.45%	100.00%
	1080	0.99	64.67%	100.00%	47.79%
Support Vector Machine (SVC)	360	1.00	70.81%	100.00%	54.81%
	1080	1.00	68.41%	100.00%	51.99%
Linear SVC	360	1.00	82.67%	100.00%	70.46%
	1080	0.99	75.24%	100.00%	60.31%
K-nearest neighbors	360	0.97	75.27%	100.00%	60.34%
	1080	0.55	43.30%	100.00%	27.63%
Gaussian Naive Bayes	360	0.94	53.25%	94.05%	37.14%
	1080	0.97	58.46%	97.52%	41.74%

Tabella 3.10. Quesito 1 - Risultati ottenuti a seconda del modello utilizzato con $N = 360$ o 1080

I modelli testati si comportano tutti abbastanza bene, in particolare si ottengono ottimi risultati grazie a: **Logistic regression**, **Ridge classifier**, **Random forest** e **Linear SVC**. Da notare che 3 su 4 sono modelli lineari, ciò è dovuto al fatto che i dati sono dipendenti tra loro nel tempo e presentano quindi una certa linearità. Quest'ultima è accentuata dalla presenza del campo *Cicli* che permette di avere un attributo per identificare il tempo e dall'applicazione del filtro Savitzky-Golay che smussa la curva dei valori iniziali. Sfruttando quindi l'andamento nel tempo si può avere una maggiore accuratezza nelle predizioni. Per avere la più alta media tra il punteggio F1 e l'AUC, si è scelto di utilizzare il **Linear SVC** per l'applicazione finale. Ai fini di migliorare leggermente le

prestazioni si calcola 3 volte il **Linear SVC**, inizialmente con il parametro C di valore 0.01, poi 0.1 ed infine 1, dopodiché si tiene il modello con il risultato migliore.

3.3.10 Quesito 1 - Soluzione finale

Dopo tutte le fasi di sperimentazione effettuate, si è arrivati alla catena di metodi rappresentata in figura 3.17.

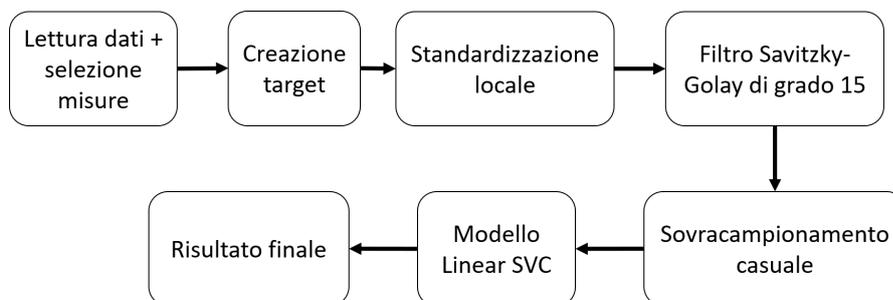


Figura 3.17. Quesito 1 - Sequenza di processi da effettuare per arrivare al risultato (stadio definitivo)

- **Letture dati + selezione misure.** Ciò serve per l'acquisizione dei dati tramite la lettura di un file. Tutte le misure tranne *Timestamp* vengono considerate per l'allenamento del modello di apprendimento.
- **Creazione target.** Viene generata la variabile target *Target_N_Cycles*. Vengono aggiunti anche altri attributi che però servono solo di aiuto per alcune stampe e grafici, verranno poi scartati prima dell'allenamento del modello.
- **Standardizzazione locale.** Questa è la normalizzazione migliore per i dati in esame. Viene applicata su ogni singolo fallimento ignorando i rimanenti in modo che essi non si influenzino tra loro.
- **Filtro Savitzky-Golay di grado 15.** Questo processo smussa la curva iniziale facilitando le predizioni ed eliminando il rumore dei sensori.
- **Sovracampionamento casuale.** Esso corregge il problema dei dati sbilanciati in caso di N non abbastanza grande.
- **Modello Linear SVC.** Avviene infine l'allenamento del modello e la fase di test. Essendo che i dati possiedono una certa linearità, i modelli di apprendimento lineari come **Linear SVC** sono particolarmente adatti per effettuare delle buone predizioni.

In tabella 3.11 sono mostrati i risultati ottenuti grazie al modus operandi appena descritto al variare di N . Si ricorda che siccome la media del tempo impiegato per ogni ciclo è circa 23.5 secondi o più, si ipotizza che 180 cicli equivalgano a un'ora di predizione nel futuro.

N	AUC	F1	Recupero	Precisione
360 (2 ore)	1.00	82.09%	100.00%	69.63%
720 (4 ore)	1.00	75.09%	60.11%	100.00%
1080 (6 ore)	0.99	75.60%	100.00%	60.77%
1440 (8 ore)	0.99	75.12%	100.00%	60.16%
1800 (10 ore)	0.90	78.50%	100.00%	64.61%

Tabella 3.11. Quesito 1 - Risultati ottenuti con l'applicazione finale al variare di N

Si può quindi concludere di aver raggiunto una buona qualità sulle predizioni necessarie rispetto a quelle attese, si conferma quindi la struttura dell'applicazione finale come appena descritta.

3.4 Quesito 2 - La macchina si sta comportando in modo anomalo?

Il quesito posto si occupa di predire lo stato attuale di salute della macchina in esame. Questo è un problema di classificazione multi classe; i 3 possibili output vengono tecnicamente presentati con i numeri 0, 1 e 2 e corrispondono rispettivamente alle stringhe, in modo da essere più facilmente riconoscibili da un umano, *Stato intermedio*, *Stato normale* e *Stato critico*. La seconda classe identifica una macchina che lavora secondo le modalità attese, l'ultima rappresenta un fallimento in arrivo, lo *Stato intermedio* si ottiene quando il comportamento del macchinario inizia ad allontanarsi da quello normale ma il distacco non è ancora eccessivo. Nella pratica, gli stati sono distinguibili in base ai cicli mancanti (RUL) dal momento attuale al prossimo fallimento. I valori M e N sono dei numeri interi positivi e definiscono le soglie per passare da una classe all'altra. Siccome la media del tempo impiegato dai cicli è, nel primo file, 23.8 secondi e, nel secondo documento, 23.5, si è ipotizzato di avere dei cicli di minimo 20 secondi perciò sono stati definiti $M = 1800$ e $N = 900$ ovvero rispettivamente 10 e 5 ore. Si avrà quindi lo *Stato normale* se il prossimo fallimento avverrà tra più di 10 ore ($RUL > M$), *Stato intermedio* se compreso tra 10 e 5 ($N < RUL \leq M$), *Stato critico* se meno di 5 ore ($RUL \leq N$). Non è consigliato porre M maggiore di 1800 cicli poiché alcuni fallimenti hanno una durata inferiore, di conseguenza alcuni casi non presenterebbero la classe *Stato normale* rendendola inutile e ricadendo nella situazione di classificazione binaria come nel primo quesito. Sarebbe meglio che, in futuro, venga implementato il fatto che l'utente stesso possa selezionare i valori soglia (in cicli macchina o minuti/ore) in modo da ottenere un'applicazione più flessibile e adatta all'utilizzatore.

Come già mostrato in Quesito 1 - Selezione misure, anche in questo caso si hanno prestazioni migliori con l'inclusione del campo *Cicli*. È stato già dimostrato che la sua aggiunta non genera overfitting per l'allenamento e test applicati per la seguente sperimentazione. Perciò la variabile *Cicli* verrà presa in considerazione da subito.

Come per la classificazione binaria, anche in questo caso abbiamo un problema di dati sbilanciati, essendo già stato trattato in Quesito 1 - Problema dei dati sbilanciati non verrà spiegato di nuovo per ridondanza. Si ha la situazione in cui *Stato normale* e *Stato critico* vengono predetti con un'elevata esattezza ma *Stato intermedio* viene sempre assegnato erroneamente alle altre due classi ottenendo così una precisione, recupero e F1 uguali a 0. Bisogna quindi fare in modo che quest'ultima classe non venga ignorata. Si utilizzeranno i criteri precisione, recupero, punteggio F1 e AUC per stabilire la qualità dei modelli ottenuti per non incorrere nel paradosso dell'accuratezza. In particolare verrà preso in maggiore considerazione il punteggio F1 poiché, dalla sperimentazione, si è notato che esso rispecchia al meglio la qualità del modello ricavato. Nel quesito precedente i criteri in considerazione sfruttavano la media binaria ma, nel caso di classificazione multi classe, ciò non è possibile ma è necessario usufruire della media micro o macro [160] [161]. Prendiamo per esempio il calcolo della precisione. Nel primo scenario vengono sommati i veri positivi, falsi positivi e falsi negativi di tutte le classi e vengono poi applicate le classiche formule, si avrà quindi $Prec_{micro} = \frac{VP_{somma}}{VP_{somma} + FP_{somma}} = \frac{VP_1 + \dots + VP_N}{VP_1 + \dots + VP_N + FP_1 + \dots + FP_N}$. Per la media macro si effettua una media aritmetica tra le precisioni delle varie classi $Prec_{macro} = \frac{Prec_1 + Prec_2 + \dots + Prec_N}{N}$. Le regole si applicano allo stesso modo per richiamo, F1 e AUC. In caso di dati sbilanciati è consigliato utilizzare la media micro quindi i criteri che verranno riportati saranno sempre riferiti a questa metodologia.

Siccome alcuni modelli permettono di lavorare unicamente su due classi alla volta, è necessario applicare dei metodi in modo da adattarli ad una classificazione multi classe. Per questa sperimentazione, la tecnica applicata è uno **contro tutti** (o **one vs all**) [162], essa genera un numero di modelli binari pari alla quantità dei possibili output. Ogni modello lavora secondo il classico comportamento binario sfruttando da una parte una singola classe e dall'altra tutte le rimanenti.

Siccome il modus operandi è molto simile a quello impiegato per il quesito precedente Quesito 1 - La macchina avrà un fallimento nei prossimi N secondi/ore/giorni/cicli macchina?, si cercherà di andare più velocemente per evitare eccessiva ridondanza. La catena iniziale dei processi da applicare è già stata presentata in figura 3.12, essa presenta l'idea iniziale da cui partire e

verrà probabilmente modificata in base ai risultati della sperimentazione. La scelta finale è invece mostrata in figura 3.20.

3.4.1 Quesito 2 - Creazione della variabile target *Target_multi_class*

La variabile target è chiamata *Target_multi_class* e rappresenta lo stato di salute della macchina considerata rispondendo al quesito di questa sezione. I suoi possibili valori sono 3: *Stato normale*, *Stato intermedio* e *Stato critico*. Il primo rappresenta un comportamento corretto, l'ultimo indica che un fallimento è vicino nel tempo e il rimanente è uno stadio intermedio tra i due precedenti ovvero la macchina inizia ad allontanarsi dalla sua condotta prevista ma è ancora lontana dal guasto. *Target_multi_class* è derivata dall'attributo *rul* il quale verrà trattato nel dettaglio in Quesito 3 - Creazione della variabile target *rul*. Entrambe le variabili dipendono dai fallimenti quindi bisogna localizzarli, le loro righe non presentano uno o più misurazioni e, di conseguenza, lasciano dei campi vuoti. Per ogni ciclo *ciclo_riga_corrente* partendo da un fallimento *ciclo_fallimento* fino al fallimento precedente o inizio del file si calcola $rul = ciclo_{fallimento} - ciclo_{riga_corrente}$. Ora le righe rappresentanti i guasti sono inutili e vengono quindi eliminate, in questo modo non si ha più il problema dei dati incompleti, nulli o mancanti. Infine viene creata la variabile target

$$Target_multi_class = \begin{cases} 1 & \text{se } rul > M \\ 0 & \text{se } N < rul \leq M \\ 2 & \text{se } rul \leq N \end{cases}$$

dove 0 rappresenta *Stato intermedio*, 1 è associato a *Stato normale* e 2 a *Stato critico*. Si è notato che questa associazione tra numero e classe produce un miglior risultato, per esattezza corregge meglio il problema dei dati sbilanciati. L'output del target è quindi un numero intero compreso tra 0 e 2. Più tecnicamente si ha lo *Stato intermedio* se mancano più di M cicli al fallimento, *Stato intermedio* se più di N ma meno o uguale a M , *Stato critico* se meno di N . Anche se non più utile, *rul* non viene subito eliminato poiché verrà impiegato come asse X per alcuni grafici, verrà comunque scartato prima dell'allenamento dei modelli. Viene anche creata la variabile *failure number* che associa un valore numerico univoco ad ogni fallimento, grazie a ciò è possibile stampare le curve dei guasti nei grafici con colori diversi in modo da distinguerle più facilmente. Anche questa non verrà utilizzata per l'allenamento venendo distrutta prima. Il problema è detto supervisionato [129] poiché si conoscono tutti i possibili output della variabile target.

3.4.2 Quesito 2 - Normalizzazione

Le tecniche di normalizzazione utilizzate sono ridimensionamento, standardizzazione, ridimensionamento locale e standardizzazione locale; esse sono state spiegate in dettaglio in Elaborazione e analisi dati. La tabella 3.12 mostra i risultati ottenuti al variare della normalizzazione implementata. Si è utilizzato $M = 1800$, $N = 900$, filtro Savitzky-Golay di ordine 15, un sovracampionamento casuale e modello *logistic regression*. Si è applicato il filtro e il ricampionamento poiché, senza di essi, si ottengono sempre scarsi risultati quindi non si riesce a capire bene l'efficacia della normalizzazione sul risultato finale.

Dalla sperimentazione ricavata dalla tabella 3.12 si nota che, senza la normalizzazione, il problema dei dati sbilanciati porta il modello a restituire sempre come output la classe *Stato intermedio* fornendo così un pessimo risultato. Ciò viene più o meno corretto dalle tecniche applicate, la standardizzazione globale è quasi inutile, i ridimensionamenti locale e globale forniscono un discreto risultato ma il metodo più efficace è la standardizzazione locale. Sarà quindi quest'ultima tecnica ad essere applicata in futuro per la soluzione finale.

Normalizzazione	AUC	F1	Recupero	Precisione
Nessuna normalizzazione	0.20	0.00%	0.00%	0.00%
Ridimensionamento	0.64	55.09%	67.89%	46.35%
Standardizzazione	0.52	1.59%	00.90%	6.73%
Ridimensionamento locale	0.75	50.94%	49.32%	52.67%
Standardizzazione locale	0.88	70.35%	64.64%	77.17%

Tabella 3.12. Quesito 2 - Risultati ottenuti a seconda della normalizzazione effettuata

3.4.3 Quesito 2 - Filtraggio

Le due tipologie di filtro sperimentate sono Savitzky-Golay e Kalman, la loro teoria è già stata presentata in Quesito 1 - Filtraggio. Il risultato del lavoro effettuato applicando i filtri Savitzky-Golay (con approssimazioni di grado 2, 5, 10 e 15) o Kalman è mostrato in tabella 3.13. Sono state applicate le seguenti configurazioni: $M = 1800$, $N = 900$, standardizzazione locale, sovracampionamento casuale e **logistic regression**. Il sovracampionamento casuale è stato implementato per correggere il problema dei dati sbilanciati, in caso contrario si ottengono sempre risultati scarsi e risulta difficile comparare i vari filtri tra loro.

Filtro	AUC	F1	Recupero	Precisione
Nessun filtro	0.72	39.34%	28.29%	64.57%
Savitzky-Golay grado 2	0.96	70.74%	54.73%	100.00%
Savitzky-Golay grado 5	0.77	64.32%	66.81%	62.02%
Savitzky-Golay grado 10	0.76	57.98%	54.53%	61.89%
Savitzky-Golay grado 15	0.89	70.33%	63.77%	78.40%
Savitzky-Golay grado 20	0.84	66.04%	63.04%	69.34%
Kalman	0.75	45.44%	34.16%	67.83%

Tabella 3.13. Quesito 2 - Risultati ottenuti a seconda del filtro applicato

Qualsiasi filtro implementato migliora i risultati ottenuti. Kalman è il peggiore e impiega circa 1140 secondi per la sua applicazione quindi verrà scartato a favore del suo concorrente che mostra un output migliore e impiega circa 3 secondi. Savitzky-Golay incrementa le prestazioni in particolare configurando il grado 2 e 15. Nonostante il grado 2 sia leggermente migliore, si è notato che, al variare di M ed N , il grado 15 mantiene meglio le sua efficienza quindi sarà quest'ultimo ad essere utilizzato per l'applicazione finale. L'applicazione di questa tecnica corregge in parte il problema dei dati sbilanciati. Oltretutto si ha ora una migliore visualizzazione dei dati, le curve ottenute sono le stesse già presentate in figura 3.13.

3.4.4 Quesito 2 - Principal Component Analysis (PCA)

La PCA serve per ridurre la dimensionalità dei dati, trasforma i dati da uno spazio di grandezza K a una dimensione minore di K in modo da ridurre complessità e tempo di calcolo per una mole

troppo grande di valori. Essa è stata spiegata meglio in Quesito 1 - Principal Component Analysis (PCA).

La sperimentazione è avvenuta con $M = 1800$, $N = 900$, standardizzazione locale, filtro Savitzky-Golay di grado 15, sovracampionamento casuale e modello di apprendimento **logistic regression**. In figura 3.18 sono mostrati i fallimenti da 1 a 6 con e senza l'applicazione della PCA con 8 componenti principali in un grafico di due dimensioni; la visualizzazione non cambia modificando il numero delle componenti. La tabella 3.14 mostra i risultati ottenuti a seconda del numero di componenti principali applicate per la PCA; il campo *#componenti principali* è il numero di componenti selezionato, *Varianza mantenuta* è la frazione di varianza conservata dopo la PCA (e quindi quantità di informazione mantenuta) e *Tempo (s)* è l'ammontare in secondi impiegato dal modello per effettuare l'allenamento.

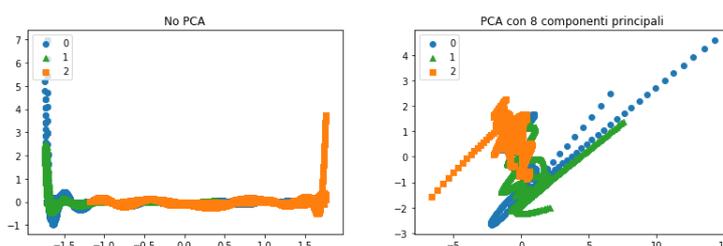


Figura 3.18. Quesito 2 - Dati rappresentati in due dimensioni con e senza l'applicazione della PCA (rispettivamente a destra e sinistra)

#componenti principali	Varianza mantenuta	Tempo (s)	AUC	F1	Recupero	Precisione
No PCA	1.00	13.31	0.88	70.39%	64.45%	77.54%
2	0.55	6.13	0.86	63.96%	54.46%	77.47%
5	0.84	6.31	0.84	60.93%	54.14%	69.66%
8	0.96	6.75	0.85	59.59%	52.46%	68.96%
11	0.99	7.91	0.88	64.90%	56.68%	75.91%
14	1.00	10.02	0.90	69.38%	61.70%	79.25%

Tabella 3.14. Quesito 2 - Risultati ottenuti a seconda del numero di componenti principali impiegati dalla PCA

Analizzando la tabella 3.14, la varianza viene mantenuta abbastanza bene, dimezzando le variabili da 17 a 8 si ha ancora una conservazione del 96%. Il tempo di esecuzione risulta diminuito rispetto all'assenza di PCA fino ad arrivare ad una riduzione di massimo 54%. Tutti i criteri, a differenza del tempo, sono minori implementando la PCA con 11 o meno componenti principali. Con 14 si ha un AUC appena maggiore e F1 leggermente minore però, al diminuire di M e N , le prestazioni calano e si ha un maggiore distacco. Per questo motivo si è deciso di non applicare la PCA poiché la maggior velocità ottenuta non giustifica il taglio di prestazioni, si è quindi voluto dare una maggiore importanza a quest'ultime.

3.4.5 Quesito 2 - Sottocampionamento e sovracampionamento

Il sottocampionamento e il sovracampionamento sono delle tecniche per aggiungere o scartare dei dati in modo da correggere il problema dei dati sbilanciati, questi metodi sono già stati descritti in Quesito 1 - Sottocampionamento e sovracampionamento. In questo caso i valori di M e N sono diminuiti in modo da mostrare meglio il problema dei dati sbilanciati. La sperimentazione in tabella 3.15 è avvenuta utilizzando $M = 720$, $N = 360$, standardizzazione locale, filtro Savitzky-Golay di grado 15 e modello **logistic regression**; i criteri F1, recupero e precisione sono riferiti a *Stato intermedio* visto che è questo ad essere affetto dal problema dei dati sbilanciati. Inoltre per la figura

3.19 è stata applicata anche una PCA con 8 componenti principali in modo che la visualizzazione dei dati fosse più intuitiva per il lettore. La legenda indica che i cerchi azzurri si associano allo stadio *Stato intermedio*, i triangoli verdi alla classe *Stato normale* e i quadrati arancioni allo *Stato critico*. Analizzando i grafici possiamo dedurre che il sottocampionamento casuale mostra un'eliminazione di alcuni dati, il sovracampionamento casuale non presenta differenza perché le nuove misurazioni sono sovrapposte a quelle vecchie visto che presentano gli stessi valori, il sottocampionamento con Tomek links rimane uguale a quello con solo la PCA poiché, come si vedrà tra poco, non scarta né genera nessun dato, le tecniche SMOTE e SMOTE + Tomek links mostrano più quadrati arancioni (dovrebbero avere anche più cerchi azzurri ma sono coperti dagli altri campioni).

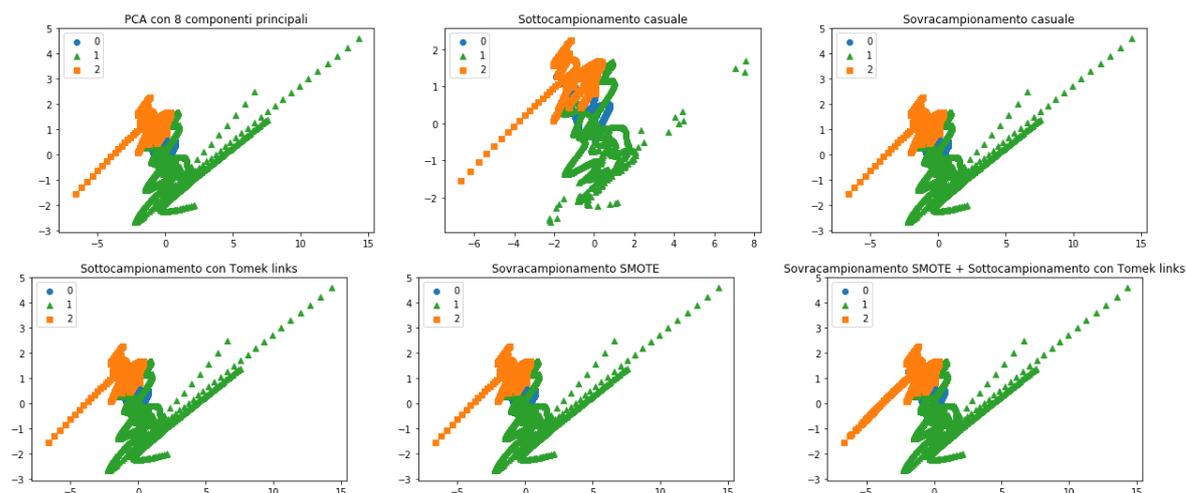


Figura 3.19. Quesito 2 - Risultato delle varie tecniche di ricampionamento

	F1	Recupero	Precisione
Nessun ricampionamento	0%	0%	0%
Sottocampionamento casuale	58%	44%	86%
Sovracampionamento casuale	71%	67%	75%
Sottocampionamento con Tomek links	0%	0%	0%
Sovracampionamento SMOTE	69%	100%	53%
SMOTE + Tomek links	72%	69%	74%

Tabella 3.15. Quesito 2 - Risultati ottenuti a seconda del ricampionamento implementato

Dalla tabella 3.15 si nota che tutti, tranne i Tomek links, correggono il problema dei dati sbilanciati. In particolare i sovracampionamenti hanno buone prestazioni con punteggi F1 simili, per l'applicazione finale verrà implementato il ricampionamento SMOTE + Tomek links visto che è leggermente migliore rispetto agli altri. Da notare che il sottocampionamento con Tomek links non porta nessuna variazione rispetto all'assenza di ricampionamento, ciò è dovuto al fatto che, in questo caso, l'algoritmo né scarta né aggiunge nessun dato quindi è come se non venisse attuato.

3.4.6 Quesito 2 - Creazione nuove variabili

Questa sezione è presentata ora nonostante nell'applicazione venga eseguita tra il filtraggio e l'eventuale PCA perché, conoscendo i componenti da applicare alla soluzione finale, si può scegliere più consciamente l'effetto della creazione di nuove variabili.

Per la tabella 3.16, si è provato a donare all'applicazione delle nuove variabili contenenti la media e la varianza mobile dei dati, la finestra di calcolo utilizzata è di 5 elementi. La rimanente configurazione è $M = 1800$, $N = 900$, standardizzazione locale, filtro Savitzky-Golay di grado 15, ricampionamento SMOTE + Tomek links e modello `logistic regression`.

	AUC	F1	Recupero	Precisione
Nessuna nuova variabile	0.88	70.33%	64.69%	77.05%
Media mobile	0.88	71.53%	67.85%	75.62%
Varianza mobile	0.89	72.00%	67.05%	77.74%
Media e varianza mobili	0.89	72.60%	68.80%	76.85%

Tabella 3.16. Quesito 2 - Risultati ottenuti a seconda dell'aggiunta di media e varianza mobile con finestra di 5 valori come variabili

Dall'analisi della tabella 3.16 si evince che l'aggiunta di nuove variabili aumenta leggermente il punteggio F1 lasciando quasi invariato il parametro AUC. Cambiando la finestra di calcolo delle variabili mobili a 2, 10, 20 e 50 si ottengono risultati molto simili ma leggermente inferiori. Si è scelto di includere sia le nuove variabili sulla media che sulla varianza mobile. Unica pecca è il tempo di esecuzione, questo passa da circa 13/14 secondi a 35/40, si ritiene che una durata simile sia ancora accettabile quindi non si ha alcun problema; in caso la velocità diventi un requisito necessario si consiglia di saltare la parte sull'aggiunta delle nuove variabili.

3.4.7 Quesito 2 - Selezione misure

Si vuole studiare se alcune misure risultano ininfluenti o dannose ai fini del risultato finale. Per far ciò si è usato il `logistic regression` per vedere l'associazione tra una misura e l'importanza data dal modello tramite un valore reale. In tabella 3.17 sono riportate nel campo *Variabili importanti* gli attributi con un'importanza maggiore di zero. Si è utilizzato: $M = 1800$, $N = 900$, standardizzazione locale, filtro Savitzky-Golay di grado 15, aggiunta delle variabili con media e varianza mobile, ricampionamento SMOTE + Tomek links. Rispetto a Quesito 1 - Selezione misure non sono presentati i casi composti dall'allenamento dei fallimenti da 1 a 4 con test 5 e allenamento 1, 2 con test 3 poiché sia il guasto 5 sia il terzo hanno meno di 1800 cicli e quindi la classe *Stato normale* non è presente.

Come si deduce dalla tabella 3.17, le misure più importanti non sono sempre le stesse quindi c'è la possibilità di eliminare alcune variabili che risultano importanti solo per alcuni casi. Oltretutto, scartando tutti gli attributi tranne quelli definiti come importanti, si ottengono dei risultati uguali o leggermente peggiori, il tempo diminuisce solamente di 1 o 2 secondi. Per questo motivo non conviene effettuare nessuna selezione ma includere tutte le misure senza distinzioni.

3.4.8 Quesito 2 - Modelli di apprendimento

Si vuole ora scegliere il modello di apprendimento che meglio predice l'output desiderato. La fase di sperimentazione è avvenuta con le due coppie $M = 1800 / N = 900$ e $M = 720 / N = 360$. I dati hanno successivamente subito standardizzazione locale, filtro Savitzky-Golay di grado 15, aggiunta delle variabili con media e varianza mobile, ricampionamento SMOTE + Tomek links. Il report del lavoro effettuato è in tabella 3.18.

Fallimenti per allenamento	Fallimento per test	Variabili importanti
1,2,3,4,5,6	7	Cicli, Tempo dosaggio, Cuscino, p-ini mas, Tempo chiusura sta, Posizione dosaggio stop, Cicli_std, Tempo ciclo_std, Cuscino_std, Stop vite_std, PM pressione_std, p-ini mas_std, PM pos_std, UG1 temp_std, Tempo iniezione_avg, Tempo dosaggio_avg, Cuscino_avg, p-ini mas_avg, Tempo apertura sta_avg, Posizione dosaggio stop_avg, PM pos_avg, UG1 temp_avg, Z1 temp_avg
1,2,3,4,5	6	Cicli, Tempo dosaggio, Cuscino, p-ini mas, Posizione dosaggio stop, Cicli_std, Tempo iniezione_std, Tempo dosaggio_std, Cuscino_std, Stop vite_std, PM pressione_std, Tempo apertura sta_std, Posizione dosaggio stop_std, PM pos_std, UG1 temp_std, Tempo dosaggio_avg, Cuscino_avg, p-ini mas_avg, Posizione dosaggio stop_avg, Z1 temp_avg

Tabella 3.17. Quesito 2 - Variabili più importanti a seconda dei fallimenti usati per allenamento e test

	N	M	AUC	F1	Recupero	Precisione
Logistic regression	360	720	0.99	93.97%	90.15%	98.14%
	900	1800	0.89	71.93%	65.56%	79.67%
Ridge classifier	360	720	0.97	91.50%	89.63%	93.44%
	900	1800	0.84	66.42%	63.29%	69.87%
Decision Tree	360	720	0.84	78.93%	78.93%	78.93%
	900	1800	0.72	48.00%	48.00%	48.00%
Random Forest	360	720	0.99	87.33%	78.59%	98.26%
	900	1800	0.78	37.02%	27.07%	58.51%
Support Vector Machine (SVC)	360	720	0.96	88.60%	90.71%	86.59%
	900	1800	0.79	65.76%	65.10%	66.43%
Linear SVC	360	720	0.99	93.67%	89.20%	98.62%
	900	1800	0.89	71.35%	66.07%	77.53%
K-nearest neighbors	360	720	0.64	51.20%	51.07%	51.32%
	900	1800	0.47	28.98%	28.98%	28.98%
Gaussian Naive Bayes	360	720	0.72	64.70%	96.07%	48.77%
	900	1800	0.77	62.54%	75.56%	53.34%

Tabella 3.18. Quesito 2 - Risultati ottenuti a seconda del modello utilizzato con le due coppie $N = 360$ e $M = 720$, $N = 900$ e $M = 1800$

Dalla tabella 3.18 si dimostra che, come nel quesito precedente, i modelli più performanti sono quelli di tipo lineare ovvero **logistic regression**, **ridge classifier** e **linear SVC**. Ciò è dovuto al fatto che le misurazioni sono legate tra loro nel tempo e quindi presentano una certa linearità, come abbiamo già visto nei grafici riportati nelle figure 3.8 , 3.9 e 3.13 esse compongono una certa curva. Si è scelto di implementare il **logistic regression** per la soluzione finale poiché produce i migliori risultati. Per cercare di ottenere un'efficienza ancor maggiore, il modello viene testato attraverso una combinazione dei parametri C (con valori 0.01, 0.1, 1 e 10) e $solvers$ (con parametri *liblinear* e *lbfgs*).

3.4.9 Quesito 2 - Soluzione finale

Ecco, in figura 3.20, i vari elementi implementati per arrivare all'applicazione finale.

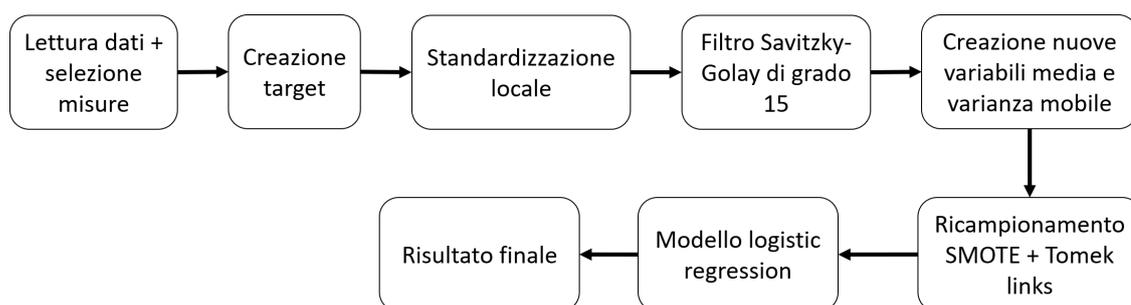


Figura 3.20. Quesito 2 - Sequenza di processi da effettuare per arrivare al risultato (stadio definitivo)

- **Letture dati + selezione misure.** I dati vengono acquisiti dai file, viene creata la variabile *Timestamp* da *Data* e *Tempo* le quali vengono poi scartate. Tutte le variabili iniziali tranne *Timestamp* verranno utilizzate in futuro per l'allenamento dei modelli.
- **Creazione target.** Viene creato l'attributo target *Target_multi_class* contenente le tre classi *Stato normale*, *Stato intermedio* e *Stato critico*.
- **Standardizzazione locale.** La standardizzazione porta la media dei dati a zero in modo che vengano analizzati più facilmente dai modelli. Essa è applicata ad ogni fallimento in modo indipendente rispetto gli altri.
- **Filtro Savitzky-Golay di grado 15.** Il filtro trasforma la curva iniziale delle misurazioni in un'approssimazione di grado 15 eliminando così il rumore dei sensori e rendendo più semplici le predizioni poiché si ha un andamento di grado definito.
- **Aggiunta di nuove variabili riportanti media e varianza mobile.** Si forniscono al modello maggiori informazioni sui dati creando nuove variabili. La finestra utilizzata per il calcolo della media e varianza mobile è di 5 elementi.
- **Sovracampionamento SMOTE + sottocampionamento con Tomek links.** Il ricampionamento bilancia le varie classi, in particolare *Stato intermedio*, risolvendo il problema dei dati sbilanciati.
- **Modello logistic regression.** Viene infine applicato il modello di apprendimento. Si è notato che i modelli di tipo lineare forniscono i migliori risultati, ciò è dovuto al fatto che i dati hanno una certa linearità dovuta al tempo.

Sono ora mostrate, in tabella 3.19, le prestazioni ottenute al variare di M e N grazie all'applicazione della soluzione finale. È utile ricordare che il tempo impiegato per ogni ciclo è in media circa 23.5 secondi quindi si ipotizza che 180 cicli equivalgano a un'ora di predizione.

N	M	AUC	F1	Recupero	Precisione
180 (1 ora)	360 (2 ore)	0.99	90.60%	97.59%	84.55%
360 (2 ore)	720 (4 ore)	0.99	93.96%	90.07%	98.19%
540 (3 ore)	1080 (6 ore)	0.86	61.43%	60.73%	62.14%
720 (4 ore)	1440 (8 ore)	0.90	72.44%	75.85%	69.31%
900 (5 ore)	1800 (10 ore)	0.89	71.93%	66.32%	78.58%

Tabella 3.19. Quesito 2 - Risultati ottenuti con l'applicazione finale al variare di N e M

Dalla tabella 3.19 possiamo vedere buone prestazioni per tutti i casi, in particolare con N minore o uguale 360. L'AUC è sempre mantenuto sopra l'85% però il punteggio F1 arriva anche a circa il 60% per $N = 540$ e $M = 1080$, fatto che sarebbe meglio perfezionare. I risultati ottenuti sono buoni anche se migliorabili quindi verranno mantenuti senza modifiche nella soluzione finale.

3.5 Quesito 3 - Quanto tempo o cicli macchina mancano prima che si abbia il prossimo fallimento?

Il quesito 3 vuole predire il tempo che intercorre tra il momento attuale e il prossimo fallimento ovvero il **Remaining Useful Life (RUL)**. Il tempo è calcolato in cicli macchina. L'output è un numero intero positivo perciò si ha un lavoro di regressione. Al contrario dei precedenti quesiti di classificazione, ora l'utente può conoscere la quantità esatta di cicli mancanti per il prossimo guasto; egli può approssimare il RUL da cicli macchina a unità di tempo moltiplicando l'output per il valore medio del tempo impiegato da ogni ciclo (dai dati ottenuti è 23.8 secondi per il primo file, 23.5 per il secondo).

Le metriche utilizzate per quantificare la qualità dei risultati prodotti dai modelli sono [163] [164]:

- **Radice dell'errore quadratico medio o Root Mean Square Error (RMSE)**. Consiste nella deviazione standard campionaria delle differenze tra i valori reali e le associate predizioni. È calcolata come

$$RMSE = \sqrt{\frac{1}{n} \sum_{j=1}^n (y_j - \hat{y}_j)^2}$$

dove y_j è la reale misurazione j , \hat{y}_j è il valore predetto j . Bisogna quindi fare in modo che il suo valore sia il più possibile vicino a zero.

- **Errore assoluto medio o Mean Absolute Error (MAE)**. È la media degli errori assoluti. Per errore si intende la differenza tra i valori reali e quelli predetti. Il MAE è lineare poiché a tutti gli errori viene dato lo stesso peso. La sua formula è

$$MAE = \frac{1}{n} \sum_{j=1}^n |y_j - \hat{y}_j|$$

dove y_j è la reale misurazione j , \hat{y}_j è il valore predetto j . Come per il caso precedente, si ha lo scenario ideale se il suo valore è nullo.

- **Coefficiente di determinazione R^2** . Non esiste una definizione canonica per questo criterio. Ecco la sua rappresentazione matematica

$$R^2 = 1 - \frac{\sum_{j=1}^n (y_j - \hat{y}_j)^2}{\sum_{j=1}^n (y_j - \bar{y})^2} = 1 - \frac{\frac{1}{n} \sum_{j=1}^n (y_j - \hat{y}_j)^2}{\frac{1}{n} \sum_{j=1}^n (y_j - \bar{y})^2}$$

dove y_j è la reale misurazione j , \hat{y}_j è il valore predetto j , \bar{y} è la media dei valori reali. Il numeratore equivale al MSE ovvero il RMSE senza la radice quadrata e il denominatore è la varianza dei valori reali. Si ottiene lo scenario ideale se il risultato è uguale a 1. Maggiore il MSE, peggiore sarà il risultato di R^2 .

Come già detto in Quesito 1 - Selezione misure, il campo *Cicli* costituisce un problema di overfitting se i dati di allenamento e test provengono dallo stesso fallimento e solitamente i modelli non utilizzano degli identificativi. In questo caso, però, la variabile *Cicli* dona un criterio al modello per fargli capire che le misurazioni trattate presentano una linearità. Ciò porta ad un aumento sostanziale delle predizioni, ecco i risultati dopo aver applicato troncamento RUL a 2500, standardizzazione locale, filtro Savitzky-Golay di ordine 15 e modello `linear regression`

$$conCicli = \begin{cases} RMSE = & 486 \\ MAE = & 399 \\ R^2 = & 0.66 \end{cases} \quad noCicli = \begin{cases} RMSE = & 793 \\ MAE = & 673 \\ R^2 = & 0.09 \end{cases}$$

Per questo motivo l'attributo in questione non verrà scartato ma incluso assieme agli altri per l'allenamento.

In figura 3.21 è mostrata la sequenza di elementi da implementare per la sperimentazione. Essa è solo la struttura iniziale, in base ai risultati che si otterranno verrà modificata. In figura 3.22 è mostrata la catena finale. La selezione delle misure è nello stesso blocco dell'applicazione dei modelli poiché viene applicata in modo dinamico. Il modello viene allenato la prima volta per conoscere le misure più importanti, quest'ultime vengono salvate e le altre scartate, allora avviene un ulteriore allenamento con successiva fase di test. In questo modo gli attributi vengono scelti al meglio. Si ottengono però alcuni casi in cui lo scarto di alcune misurazioni comporta solamente una perdita di qualità. In tutte le fasi di sperimentazione che verranno presentate i risultati si riferiscono al modello migliore tra i due casi.

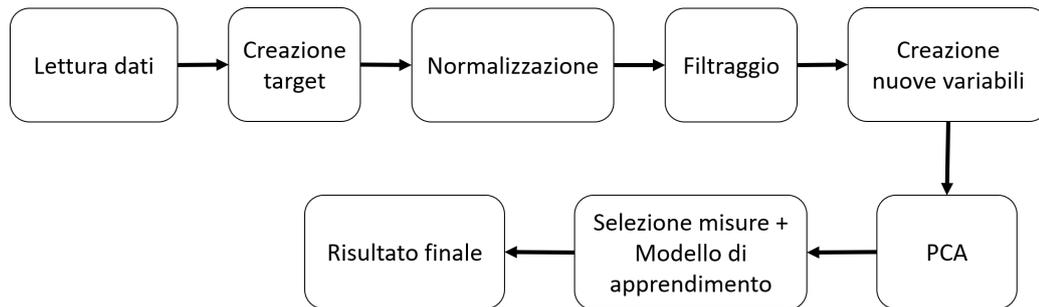


Figura 3.21. Quesito 3 - Sequenza di processi da effettuare per arrivare al risultato (stadio sperimentale)

3.5.1 Quesito 3 - Creazione della variabile target *rul*

Il target è la quantità di cicli macchina che intercorrono tra il momento attuale e il prossimo fallimento, la variabile che contiene questa informazione è detta *rul*. Per trovare quest'informazione è necessario calcolare la differenza tra il valore del ciclo del prossimo fallimento $ciclo_{fallimento}$ e il ciclo attuale $ciclo_{riga_corrente}$ calcolando

$$rul = ciclo_{fallimento} - ciclo_{riga_corrente}$$

Siccome si hanno fallimenti con lunghezze diverse, solitamente si ottengono, in valore assoluto, errori minori per i guasti più corti e maggiori per i più lunghi. Avendo un intervallo maggiore di possibili output, si ha una maggiore imprecisione. Per questo motivo si è pensato di porre un massimo $maxRUL$ al RUL predetto. È meglio avere una maggiore precisione con una soglia massima anziché non avere limiti ma prestazioni peggiori. Prima si calcola la variabile *rul* come descritto prima e poi, qualsiasi valore oltre $maxRUL$, viene imposto uguale a $maxRUL$. In questo modo, se il RUL è maggiore di $maxRUL$ verrà predetto lo stesso come $maxRUL$.

$$rul = \begin{cases} ciclo_{fallimento} - ciclo_{riga_corrente} & \text{se } ciclo_{fallimento} - ciclo_{riga_corrente} < maxRUL \\ maxRUL & \text{se } ciclo_{fallimento} - ciclo_{riga_corrente} \geq maxRUL \end{cases}$$

Si è deciso di imporre $maxRUL = 2500$. Visto che la media del tempo impiegato per ogni ciclo macchina è, rispettivamente nei due documenti, 23.8 e 23.5 secondi, si è ipotizzato di avere dei cicli di minimo 20 secondi. 2500 cicli corrispondono quindi ad almeno 13.89 ore, una quantità di tempo per le predizioni abbastanza elevata per il lavoro in questione. L'utente dovrà essere avvisato che l'output 2500 può rappresentare una quantità indefinita maggiore al suo effettivo valore. La lunghezza dei fallimenti estrapolati è, dal primo al settimo, 4418, 2670, 1052, 1048, 1712, 8373 e

4105 quindi il valore scelto come *maxRUL* coglie, tranne in un caso, almeno il 50% del massimo *RUL*. Oltretutto si è testato che il valore 2500 comporta buone prestazioni. A volte avviene che i modelli restituiscono predizioni sopra la soglia imposta perciò l'output viene infine troncato a *maxRUL*, grazie a ciò si migliorano leggermente le prestazioni.

Una volta che la variabile target è stata creata, le righe rappresentanti i fallimenti vengono scartate poiché contengono delle misurazioni vuote; in questo modo viene gestito il problema dei dati nulli. Oltretutto viene creato l'attributo *failure number* ovvero viene assegnato ad ogni fallimento un numero intero positivo identificativo in modo da poter distinguere i guasti, ciò è particolarmente utile nella rappresentazione dei dati nei grafici colorando ogni fallimento con un colore diverso. Siccome si conoscono tutti i possibili output, il problema è classificato come supervisionato [129].

3.5.2 Quesito 3 - Normalizzazione

I tipi di normalizzazione testati sono quelli già presentati nella sezione Elaborazione e analisi dati: ridimensionamento, standardizzazione, ridimensionamento locale e standardizzazione locale. I risultati ottenuti dalla loro applicazione sono presentati in tabella 3.20. Si è implementato *maxRUL* = 2500, filtro Savitzky-Golay di ordine 15 e modello **linear regression**.

Normalizzazione	RMSE	MAE	R ²
Nessuna normalizzazione	833	731	0.00
Ridimensionamento	1108	733	-0.78
Standardizzazione	800	709	0.08
Ridimensionamento locale	524	459	0.60
Standardizzazione locale	486	399	0.66

Tabella 3.20. Quesito 3 - Risultati ottenuti a seconda della normalizzazione effettuata

Grazie alla tabella 3.20 si può dedurre la migliore tecnica di normalizzazione per il caso in esame. Si nota che l'assenza di normalizzazione, il ridimensionamento e la standardizzazione globali portano pessimi risultati con un valore R² molto vicino allo 0, addirittura negativo in un caso. Con il ridimensionamento e standardizzazione locali si ottengono buoni risultati, ciò significa che i fallimenti devono essere trattati in modo indipendente dagli altri. Per l'applicazione verrà utilizzata la standardizzazione locale visto che fornisce il risultato più accurato.

3.5.3 Quesito 3 - Filtraggio

Come per i due quesiti precedenti, le tipologie testate dei filtri sono Savitzky-Golay e Kalman, esse sono state descritte in maggior dettaglio in Quesito 1 - Filtraggio. Siccome il primo filtro può sfruttare approssimazioni di grado diverso, è stato testato con grado 2, 5, 10, 15 e 20. Si è utilizzata la configurazione *maxRUL* = 2500, standardizzazione locale e modello **linear regression**. I risultati sono riportati in tabella 3.21.

Dalla tabella 3.21 si evince che il caso migliore si ottiene senza l'applicazione di alcun filtro. Il filtro Kalman non modifica le prestazioni e, oltretutto, è molto lento poiché impiega più di 1000 secondi per essere applicato al contrario del filtro Savitzky-Golay che ne richiede circa 3. La qualità del filtro Savitzky-Golay aumenta all'aumentare del grado di interpolazione. Si è però notato che il modello **gradient boosting** con Savitzky-Golay di grado 15 ottiene risultati migliori rispetto a quelli raggiunti dal **linear regression**. Per questo motivo si proverà ad utilizzare sia il metodo

Filtro	RMSE	MAE	R²
Nessun filtro	367	303	0.81
Savitzky-Golay grado 2	1029	922	-0.53
Savitzky-Golay grado 5	786	549	0.11
Savitzky-Golay grado 10	550	421	0.56
Savitzky-Golay grado 15	486	399	0.66
Savitzky-Golay grado 20	473	394	0.68
Kalman	367	303	0.81

Tabella 3.21. Quesito 3 - Risultati ottenuti a seconda del filtro applicato

senza filtri sia la tecnica Savitzky-Golay di grado 15 finché uno dei due si mostrerà definitivamente migliore.

3.5.4 Quesito 3 - PCA

La tecnica PCA ha lo scopo di ridurre la dimensionalità di un insieme di dati passando da uno spazio di dimensionalità K a una grandezza minore di K . In questo modo dovrebbe ridursi il tempo di calcolo poiché si ha una quantità minore di misurazioni da elaborare. È stata già mostrata in maggior dettaglio nella sezione Quesito 1 - Principal Component Analysis (PCA).

I risultati della sua applicazione, al variare del numero di componenti principali applicati, sono presentati in tabella 3.22. La colonna *#componenti principali* è il numero di componenti applicato, *Varianza mantenuta* è la frazione di varianza conservata dopo la PCA (e quindi quantità di informazione mantenuta) e *Tempo (s)* è l'ammontare in secondi impiegato dal modello per effettuare l'allenamento e la selezione delle misure più importanti. Per i test sono stati applicati *maxRUL* = 2500, standardizzazione locale, Savitzky-Golay di grado 15 e modello *linear regression*. Non vengono riportati i risultati ottenuti senza l'implementazione di un filtro poiché questi sono pessimi (circa RMSE = 1000, MAE = 800, R² negativo) quindi, in questo caso, la PCA è altamente sconsigliata.

#componenti principali	Varianza mantenuta	Tempo (s)	RMSE	MAE	R²
No PCA	1.00	2.92	486	399	0.66
2	0.55	2.37	358	289	0.81
5	0.84	2.41	378	303	0.79
8	0.96	2.39	358	289	0.81
11	0.99	2.48	390	312	0.78
14	1.00	2.66	390	312	0.78

Tabella 3.22. Quesito 3 - Risultati ottenuti a seconda del numero di componenti principali impiegati dalla PCA

Dalla tabella 3.22 si deduce che l'applicazione della PCA migliora le prestazioni. Il tempo impiegato rimane circa costante quindi il suo leggero risparmio verrà considerato ininfluenza. In particolare si ottengono le migliori prestazioni con 2 e 8 componenti principali. Oltretutto questa configurazione migliora leggermente i risultati ottenuti senza l'applicazione del filtro e PCA riportati in tabella 3.21. Si è scelto di utilizzare 8 componenti poiché si conserva così una maggiore varianza.

3.5.5 Quesito 3 - Creazione nuove variabili

Questa parte viene presentata solo adesso anche se nella catena di figura 3.21 è posta tra il filtraggio e la PCA poiché, conoscendo i blocchi da applicare o eliminare, possono venir effettuate scelte migliori avendo già una più concreta visione della soluzione.

Si vogliono creare delle nuove variabili in modo da dare maggiori informazioni ai modelli come la media e la varianza mobile. Si è applicato $maxRUL = 2500$, standardizzazione locale, Savitzky-Golay di grado 15 e modello **linear regression**. La finestra per le variabili mobili è di 10 elementi. La PCA non viene applicata poiché essa diminuisce drasticamente le prestazioni. Senza l'applicazione del filtro si ottengono risultati leggermente peggiori (con media e varianza mobili si ha $RMSE = 384$, $MAE = 300$, $R^2 = 0.79$). Variando la finestra di calcolo la qualità è simile o peggiore. Il frutto della sperimentazione è riportato in tabella 3.23.

	RMSE	MAE	R²
Nessuna nuova variabile	486	399	0.66
Media mobile	359	275	0.81
Varianza mobile	614	428	0.45
Media e varianza mobili	357	276	0.82

Tabella 3.23. Quesito 3 - Risultati ottenuti a seconda dell'aggiunta di media e varianza mobile con finestra di 10 valori come variabili

Con la creazione di nuove variabili si producono miglioramenti applicando la media mobile oppure sia media che varianza. In particolare quest'ultima opzione è leggermente più performante rispetto a quella presentata in Quesito 3 - PCA però non mantiene la stessa efficienza anche con gli altri modelli quindi è preferita la precedente tecnica con la PCA. Oltretutto questo nuovo metodo impiega 6.5 secondi al contrario di 2.5 del suo concorrente.

3.5.6 Quesito 3 - Modelli di apprendimento e selezione misure

Bisogna ora scegliere il modello più performante. Siccome si ottengono risultati migliori o peggiori a seconda della configurazione e modello utilizzati, si vogliono presentare due modus operandi: uno con la PCA e uno senza. La sperimentazione è avvenuta con $maxRUL = 2500$, standardizzazione locale, Savitzky-Golay di grado 15. Il frutto ottenuto è riportato in tabella 3.24. Si ricorda che l'applicazione esegue prima il modello sfruttando tutte le misure disponibili poi cerca quali tra quest'ultime sono le più importanti per l'allenamento e riapplica il modello prendendo unicamente le variabili più importanti, infine si sceglie il risultato migliore. Non sempre l'eliminazione di alcuni attributi porta predizioni più accurate. L'output ottenuto viene poi troncato a $maxRUL$ se ci sono valori maggiori di questa soglia.

Dalla tabella 3.24, considerando solo i casi con la PCA, si può dedurre che tutti i modelli applicati tranne **Elastic network** ottengono risultati simili, in particolare **linear regression**, **lasso** e **ridge** riportano criteri quasi uguali. Senza la PCA, tutti i modelli, in particolare quelli basati sugli alberi, peggiorano le loro performance a differenza del **gradient boosting** che migliora. I modelli con maggiore efficienza sono il **gradient boosting** e il **decision tree** con PCA, si è scelto il primo senza PCA poiché più accurato.

3.5.7 Quesito 3 - Soluzione finale

Si può ora definire la catena definitiva dei blocchi da applicare per ottenere le migliori predizioni, ciò è definito in figura 3.22.

Modello	Presenza PCA	RMSE	MAE	R ²
Linear regression	PCA	358	289	0.81
	No PCA	486	399	0.66
Lasso	PCA	359	290	0.81
	No PCA	369	291	0.80
Ridge	PCA	358	289	0.81
	No PCA	486	398	0.66
Elastic network	PCA	490	429	0.65
	No PCA	500	443	0.64
Gradient boosting	PCA	340	267	0.83
	No PCA	294	193	0.88
Decision tree	PCA	340	267	0.83
	No PCA	708	521	0.28
Random forest	PCA	368	244	0.80
	No PCA	711	513	0.27

Tabella 3.24. Quesito 3 - Risultati ottenuti a seconda del modello utilizzato



Figura 3.22. Quesito 3 - Sequenza di processi da effettuare per arrivare al risultato (stadio definitivo)

- **Letture dati.** Si ottengono le misurazioni da dei file di input. Le variabili *Tempo* e *Data* vengono accorpate in *Timestamp*. Tutte le colonne tranne *Timestamp* vengono considerate dall'applicazione.
- **Creazione target.** Viene creata la variabile *rul* che contiene la quantità di cicli macchina mancanti al prossimo guasto. Se un elemento è oltre la soglia *maxRUL*, il suo valore viene assegnato a *maxRUL*.
- **Standardizzazione locale.** Questa tecnica si è rivelata il miglior metodo di normalizzazione. Viene applicata ai fallimenti in modo indipendente rispetto agli altri. In questo modo la media dei dati è portata a zero.
- **Filtro Savitzky-Golay di grado 15.** Il filtro smussa l'andamento delle misurazioni ed elimina il rumore generato dai sensori. La curva finale sarà un'approssimazione di grado 15 rispetto all'originale.
- **Selezione misure + Modello gradient boosting.** Viene allenato il **gradient boosting** considerando tutte le misure, vengono scelte le variabili più determinanti per la qualità del risultato e si riapplica il modello con solo le misure più importanti. Se un suo valore è maggiore di *maxRUL* (2500 in questo caso), esso viene troncato a *maxRUL*. Si sceglie il risultato migliore tra i due.

Si hanno infine i seguenti risultati finali

$$RisultatoFinale = \begin{cases} RMSE = & 294 \\ MAE = & 193 \\ R^2 = & 0.88 \end{cases}$$

Sapendo che mediamente il tempo impiegato per svolgere ogni ciclo è di circa 23.5 secondi ed avendo un MAE di 193 cicli, ne consegue che ogni predizione ha un errore medio di circa ± 1.25 ore. In figura 3.23 sono illustrati, in un grafico di dispersione, gli output predetti dalla soluzione finale (pallini azzurri) rispetto a quelli reali (diagonale tratteggiata). L'asse X indica il RUL reale, l'asse Y il RUL predetto dalla soluzione implementata.

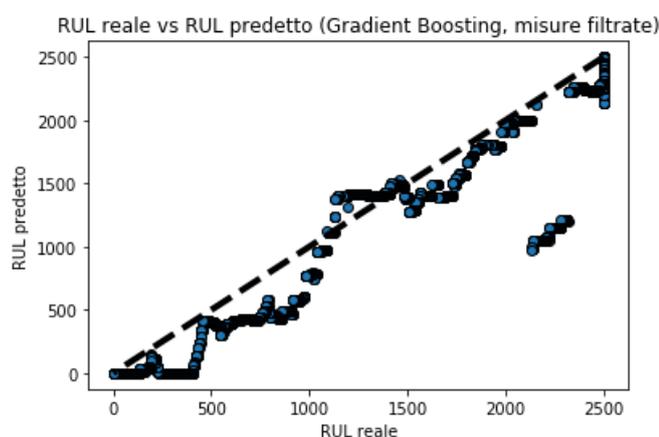


Figura 3.23. Quesito 3 - Output della soluzione finale

3.6 Conclusioni e sviluppi futuri

È stata appena presentata un'applicazione per predire i fallimenti di una macchina ad iniezione. Grazie ad essa un operaio può conoscere in anticipo lo stato futuro dell'impianto e quindi agire di conseguenza ottenendo i vari benefici presentati nella sezione Benefici della PHM. L'applicazione fornisce 3 tipi di predizioni, una per ogni quesito, in modo da rispondere al meglio ai requisiti dell'utente. Si offre così un panorama a 360° sulle potenzialità fornite dalla manutenzione predittiva.

Il lavoro svolto può venir ampliato ed esteso. Per prima cosa bisognerebbe testare il progetto per diversi tipi di macchina, non è stato possibile svolgere test con tipologie diverse dai macchinari ad iniezione poiché gli unici dati disponibili appartengono ad uno stesso modello. È possibile che il lavoro svolto funzioni egualmente a prescindere dalla provenienza delle misurazioni visto che tutte le variabili, tranne per *Data*, *Tempo* e *Timestamp* che vengono scartate, sono trattate senza conoscere il loro significato di conseguenza possono anche appartenere ad una macchina diversa con sensori differenti. Naturalmente, per non necessitare modifiche, il file di input dovrà avere lo stesso formato. Il programma potrebbe venir utilizzato in questo modo ma sarebbe molto più completo e comodo se venisse implementato con l'architettura IoT presentata in Implementazione soluzione, le modifiche e aggiustamenti da compiere per unire le due parti sono descritte in Monitoraggio e analisi dei pattern anomali real-time. Sarebbe anche bene ottimizzare i risultati ottenuti nei quesiti 2 e 3, in particolare il secondo non presenta ottime prestazioni per valori di N e M elevati.

Capitolo 4

Implementazione soluzione

L'applicazione è intitolata **Plant Control Room** la quale può essere tradotta con **Stanza di controllo dell'impianto**. Come suggerisce il nome, il programma ha l'obiettivo di monitorare e gestire un determinato impianto industriale. Già spiegato nel capitolo **Industria 4.0**, l'industria 4.0 ha, tra i suoi punti fondamentali, un approccio più facile ed efficiente da parte dei lavoratori con le macchine che lo circondano. **Plant Control Room** è ideato per essere utilizzato in ambito industriale, in particolare manifatturiero, da qualsiasi operaio. L'applicazione ha l'obiettivo di gestire i macchinari e controllare il loro andamento tramite un'interfaccia grafica semplice, essa è raggiungibile via web, in modo che si possa accedere da qualsiasi dispositivo, in particolare da apparecchi mobili. Oltre a mostrare i dati ricevuti dai sensori, sono stati aggiunti gli status delle macchine in modo da conoscere più facilmente la loro salute ed è stata implementata la gestione di allarmi e manutenzioni ai fini di poter coordinare meglio l'impianto. I servizi offerti sono:

- **Stato dell'impianto.** In base allo stato attuale dell'impianto vengono mostrati dei grafici con degli indicatori per avere una visione generale della salute dello stabilimento. A schermo appaiono anche il numero delle manutenzioni e allarmi generati nell'ultimo periodo. Affrontato nel dettaglio in **Dashboard**.
- **Dettaglio della macchina.** Si mostrano le caratteristiche, le immagini, i grafici sulle misure e le ultime 10 manutenzioni e allarmi associati al macchinario. In questo modo vengono mostrate nello specifico le informazioni di un unico dispositivo. Per una maggiore analisi, vedere **Pagina di dettaglio dell'apparecchiatura**.
- **Registrazione e modifica di un macchinario.** Prima di poter ricevere e mostrare i dati di un apparecchio, questo deve venir registrato. Durante questa fase è possibile definire alcune sue caratteristiche quali *Nome*, *Modello*, *Produttore*, *Descrizione*, eccetera e l'utente può crearne di nuove oltre a quelle standard. Si definiscono anche le misure che devono essere accettate e il loro limite inferiore e superiore sotto o sopra il quale deve essere generato un allarme. Tutte le informazioni sulla macchina possono essere modificate/aggiunte/rimosse in futuro. Questa funzionalità è spiegata in modo approfondito in **Pagina di registrazione della macchina**.
- **Gestione manutenzioni.** Serve per registrare e modificare un'azione di manutenzione. Quest'ultima può essere composta da diverse attività e associata ad una o più macchine. Anche in questo caso è possibile modificare/aggiungere/eliminare le caratteristiche di una manutenzione quando si vuole. Non verrà spiegata nel dettaglio perché non è stata sviluppata dal sottoscritto.
- **Gestione allarmi.** Rende disponibile la visualizzazione e gestione degli allarmi generati. In questo modo, tramite un evento di manutenzione, gli allarmi vengono risolti ed eliminati. Non verrà spiegata nel dettaglio perché non è stata sviluppata dal sottoscritto.

L'utente, grazie a tutti i servizi presentati, può monitorare l'impianto in modo migliore e più semplice rispetto a prima. Infatti, grazie all'aiuto dell'applicazione, l'operaio conosce gli allarmi e le manutenzioni di una macchina accedendo da un qualsiasi dispositivo (ciò era fatto precedentemente per via cartacea o tramite un file **Excel** da dover condividere con gli altri), viene avvisato in caso di comportamenti anomali tramite la generazione di allarmi (prima doveva essere il lavoratore ad accorgersi delle anomalie), visualizza l'andamento dell'impianto e delle singole macchine grazie a vari grafici (cosa non disponibile nel passato) e accede alle informazioni associate ad una macchina quali nome, caratteristiche, eccetera (prima raccolte in modo cartaceo o file **Excel** da cercare e condividere). In più tutto ciò viene acceduto da un qualsiasi dispositivo, anche mobile, (maggiore accessibilità) ed è condiviso con chi chiunque (migliore collaborazione e condivisione). L'unione di tutti questi benefici giustifica l'uso di **Plant Control Room**. Non tutte le funzionalità verranno viste nel dettaglio perché sono state sviluppate dopo la fine del lavoro della seguente tesi presso l'azienda **Blue Reply**, esse sono state presentate solo per mostrare le potenzialità dell'applicazione in questione. È altamente possibile che **Plant Control Room** verrà modificato ed ampliato ancora in futuro. Le specifiche sono state già presentate nella sezione Specifiche, la scelta dei vari strumenti è stata giustificata in Strumenti utilizzati.

4.1 IBM Cloud

L'architettura dell'applicazione è basata su **IBM Cloud** [165] (anche conosciuto con il suo vecchio nome **IBM Bluemix**). Le motivazioni della sua scelta sono state già trattate in **IBM Cloud**. Esso è la combinazione tra **Platform as a service (PaaS)** e **Infrastructure as a Service (IaaS)**. Ciò significa che viene fornita una piattaforma per sviluppare, testare ed eseguire applicazioni software senza l'acquisto e gestione di hardware apposito [7] e, allo stesso tempo, le risorse accedute sono virtualizzate comportando una serie di benefici [8]. In questo modo possono venir utilizzati i server messi a disposizione da **IBM** senza che i programmatori debbano preoccuparsi del lato hardware. **IBM Cloud** semplifica il lavoro totale fornendo un ambiente già pronto all'utilizzo, oltretutto viene fornita anche la sicurezza, affidabilità, funzionamento, potenza di calcolo e manutenzione necessaria. La piattaforma offre vari servizi per database, Internet delle cose, big data, sicurezza informatica, apprendimento automatico, gestione web e mobile su cui basare ed ampliare il proprio lavoro. I vari elementi che verranno descritti sotto sono appunto basati su **IBM Cloud**.

4.2 Declinazione architettura

L'applicazione è composta, a livello hardware, dai seguenti elementi (essi verranno poi descritti nel dettaglio nelle sezioni successive):

- **Sensori + macchina.** I sensori sono le sorgenti dei dati, uno o più di essi sono associati ad una macchina (o dispositivo) il quale, tramite protocollo **MQTT**, invia le misurazioni al server **Watson IoT Platform**. L'invio di dati dovrebbe essere periodico, nel caso di macchinari industriali la frequenza dovrebbe essere un ciclo macchina. Per poter effettuare la comunicazione, il dispositivo deve autenticarsi utilizzando il token rilasciato dal server **Watson IoT Platform**.
- **Server Watson IoT Platform.** Esso tiene conto di tutte le macchine registrate e rilascia il token per l'autenticazione. Riceve i dati dalle macchine e, se quest'ultime sono registrate, li inoltra al server **Node-RED**, altrimenti li scarta. Mette in comunicazione la sorgente dei dati e i server per gestire i dati.
- **Server Node-RED.** Riceve i dati delle macchine dal server **Watson IoT Platform** e li salva nel database **Cloudant**. In base alle misurazioni ricevute genera anche degli allarmi e modifica lo status delle macchine sul database. In questo modo vengono effettuate delle operazioni in tempo reale in base ai messaggi ricevuti.

- **Database Cloudant.** Il database immagazzina i dati necessari per le macchine e le loro misurazioni, oltretutto salva anche gli allarmi e le manutenzioni.
- **Server Plant Control Room.** Questo funge da proxy tra il client e i rimanenti server per ottenere i dati di cui il client necessita. Esso comunica con il database per chiedere, salvare o modificare i dati riguardanti macchine, allarmi o manutenzioni ed interagisce con il server **Watson IoT Platform** per registrare una macchina ed ottenere il token per configurarla.
- **Client Plant Control Room.** Il client è costituito da un'applicazione web in modo che essa possa essere acceduta facilmente e contemporaneamente dai lavoratori di un impianto industriale. In questo modo gli operai possono effettuare le loro operazioni semplicemente tramite un'interfaccia grafica. Tutti i dati mostrati vengono prima richiesti al server **Plant Control Room**.

In figura 4.1 è mostrata l'architettura di **Plant Control Room**.

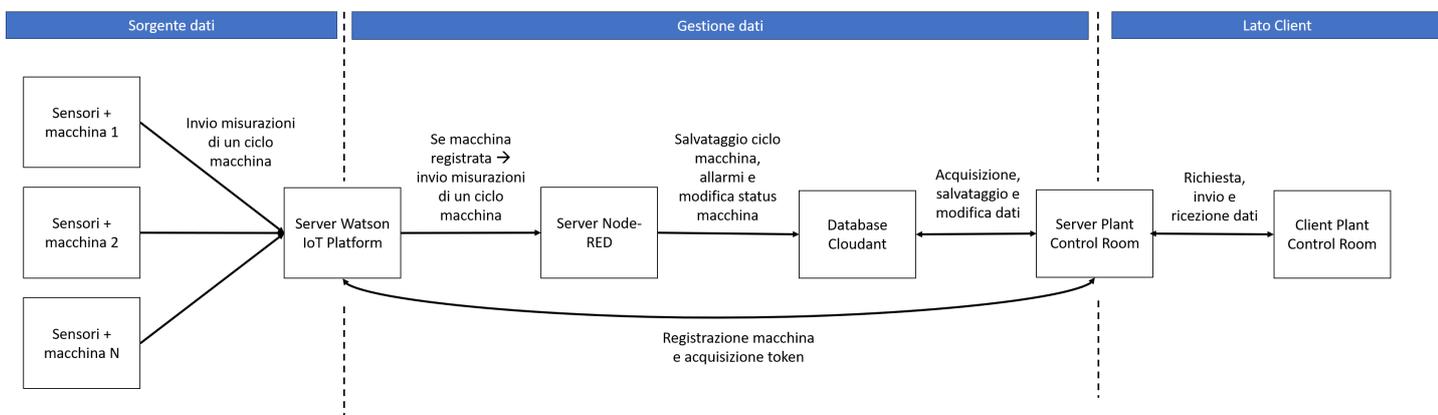


Figura 4.1. Architettura hardware di **Plant Control Room**

4.3 Connessione impianto

I dati vengono generati dai sensori connessi ad un certo dispositivo, quest'ultimo raccoglie le varie misurazioni effettuate e le invia al server **Watson IoT Platform**. La macchina deve essere prima registrata dall'applicazione affinché i suoi messaggi vengano accettati, questa fase è spiegata meglio in Comunicazione e registrazione macchina - server IBM Watson IoT Platform. I sensori possono essere di qualsiasi tipo, basta che generino dei valori numerici. Il macchinario può essere di qualunque tipologia; anche se l'applicazione è pensata per l'industria manifatturiera, qualsiasi dispositivo viene accettato. Ci si aspetta che i dati vengano inviati periodicamente, con un certo intervallo di tempo si spedisce un messaggio contenente i valori di tutte le misurazioni effettuate dai sensori in quel periodo.

Il protocollo con cui vengono trasferiti i dati è **MQTT** [166]. Questo si basa sul principio pubblicazione/sottoscrizione. Un elemento (nel nostro caso uno dei dispositivi con sensori) pubblica un evento (messaggio con all'interno delle misurazioni) riguardo un certo argomento su un server detto broker (server **Watson IoT Platform**) il quale ha il compito di gestire il salvataggio dei messaggi ricevuti e la comunicazione con i vari attori in campo. Un qualsiasi altro dispositivo può sottoscrivere un certo argomento e il broker, quando riceverà dei dati per quell'argomento, inoltrerà il messaggio a tutti gli iscritti. Per **Plant Control Room** si ha che il server **Watson IoT Platform** comunica con le macchine tramite **MQTT** come appena descritto però invia i dati agli iscritti (in questo caso l'unico è il server **Node-RED**) tramite delle API apposite. Lo schema **MQTT** è presentato in figura 4.2.

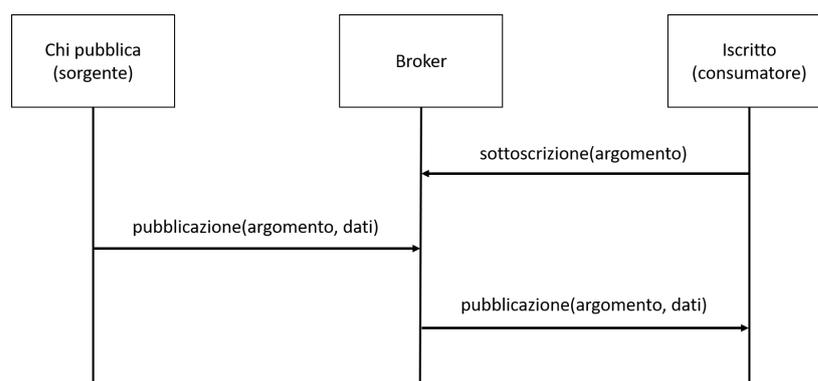


Figura 4.2. Schema del protocollo MQTT [166]

L'unica macchina fisica con cui sono stati eseguiti dei test è una scheda **Arduino** con installate una scheda di rete Wi-fi **ESP32** [167] [168] e un sensore di temperatura e umidità **DHT22** [169] [170], la prima serve per la comunicazione con il server **Watson IoT Platform**, il secondo per ottenere delle misurazioni riguardo la temperatura e umidità nell'ufficio di lavoro presso **Blue Reply**.

4.4 Comunicazione e registrazione macchina - server IBM Watson IoT Platform

IBM Watson IoT Platform permette la gestione dei dispositivi IoT e la comunicazione con essi. La motivazione della sua scelta è spiegata in **IBM Watson IoT Platform**. Come già spiegato in **Connessione impianto**, il server **Watson IoT Platform** comunica con le macchine tramite protocollo **MQTT** e svolge il ruolo di broker. I messaggi ricevuti dal server vengono accettati solo se il dispositivo mittente è stato registrato, in questo modo, per ragioni di sicurezza, vengono accettati solo i dati di cui si conosce il mandante. Il messaggio, una volta accettato, verrà poi inoltrato ad altre applicazioni, nel nostro caso **Node-RED**. Lo schema delle comunicazioni è mostrato in figura 4.3.

Per registrare una macchina bisogna seguire i seguenti passi:

- **Registrazione dispositivo.** L'utilizzatore deve accedere alla *pagina di registrazione della macchina* dell'applicazione web (essa è descritta in *Pagina di registrazione della macchina*), da qui può effettuare la registrazione della macchina. Quest'ultima può essere un oggetto qualunque che implementa il protocollo **MQTT**, non necessariamente un apparecchio industriale. L'evento avviene, da parte dell'utente, solamente attraverso l'interfaccia grafica quindi è semplice da effettuare.
- **Token di autenticazione.** Una volta effettuata la registrazione, viene salvato nel database il token di autenticazione restituito dal server **Watson IoT Platform**. Il token è una stringa lunga tra 8 e 36 caratteri i quali possono essere alfanumerici o i seguenti caratteri speciali: `-!&@?*\+.\[\]\{\}`. L'utilizzatore potrà vedere il token quando meglio desidera nella *pagina di dettaglio dell'apparecchiatura*, descritta in *Pagina di dettaglio dell'apparecchiatura*.
- **Configurazione della macchina per l'autenticazione.** Il dispositivo deve essere configurato in modo da utilizzare il token descritto nel passo precedente per l'autenticazione attraverso il protocollo **MQTT**.
- **Comunicazione con Watson IoT Platform.** Ora tutto è pronto per l'utilizzo e la macchina può inviare messaggi al server **Watson IoT Platform**. Quest'ultimo si prodigherà per inoltrare i dati al server **Node-RED**.

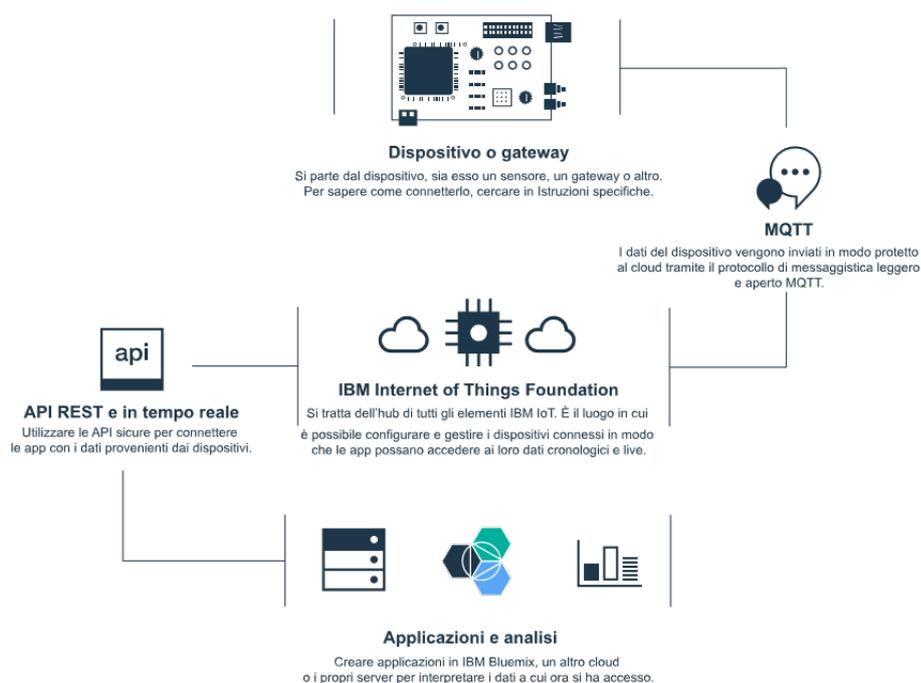


Figura 4.3. Schema delle comunicazioni intraprese da Watson IoT Platform [13]

La didascalia appena presentata è schematizzata nella figura 4.4.

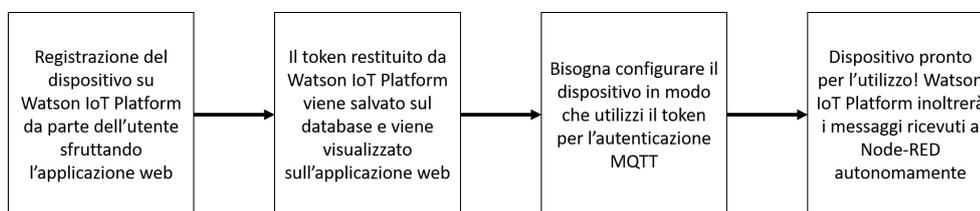


Figura 4.4. Passi per la registrazione di una macchina nell'applicazione

4.5 Elaborazione dei dati ricevuti - server Node-RED

Questa parte è stata sviluppata da Enrico Loparco. Node-RED permette di collegare vari elementi quali sensori, server, database, client per ottenere un'interazione ottimale [10]. Esso permette di svolgere diverse operazioni in tempo reale. Le motivazioni della sua scelta sono spiegate in Node-RED.

Il server Node-RED riceve i dati generati dalle macchine tramite il server Watson IoT Platform, svolge alcune azioni e salva i suoi risultati nel database. Il suo compito è:

- **Salvare sul database i dati ricevuti.** Bisogna prima controllare che la macchina sia effettivamente registrata nel database.
- **Controllare se le misure ricevute sono all'interno delle soglie massime e minime, in caso negativo generare degli allarmi.** Per ora gli allarmi sono sempre di tipo *Normal*, bisogna ancora definire secondo che criterio si assegna la gravità dell'allarme.

- Se sono stati generati allarmi, salvarli sul database.
- Se sono stati generati allarmi, modificare lo status della macchina associata. Ancora non implementato, è necessario prima decidere la logica secondo cui lo status viene alterato.

Lo schema del lavoro svolto dal server Node-RED è presentato in figura 4.5.

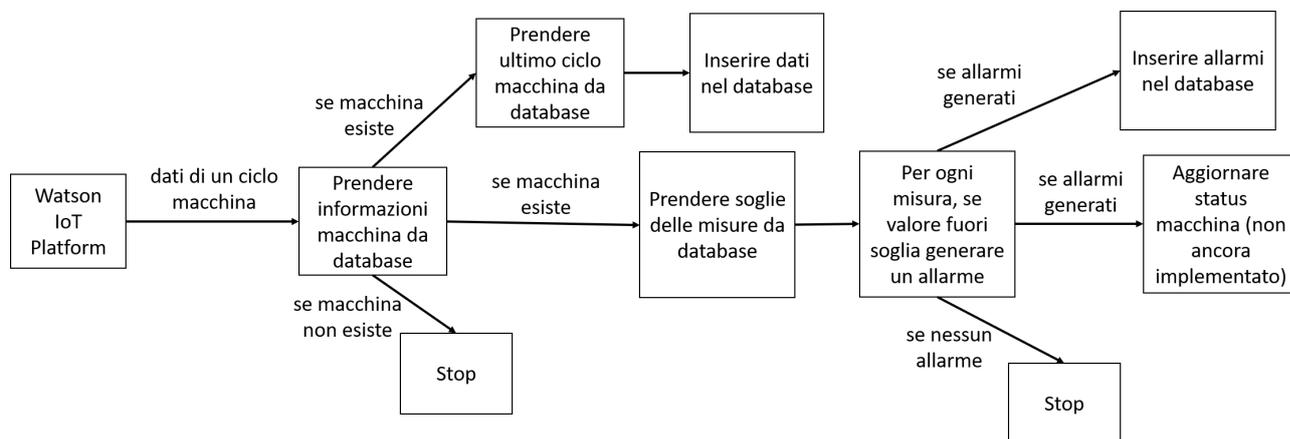


Figura 4.5. Schema del lavoro svolto dal server Node-RED

4.6 Storage - Database Cloudant

Come base di dati è stato utilizzato Cloudant, la sua scelta è motivata in Cloudant NoSQL Database. Esso immagazzina qualsiasi dato verrà poi visualizzato dal client. Essendo un database NoSQL non impone una struttura fissa ma ogni tupla (detta documento) può avere un numero e valore arbitrario di campi indipendentemente dalle altre. Qualsiasi informazione viene salvata come stringa secondo il formato JSON.

Per maggiore modularità e velocità, sono salvati diversi database separati tra loro. Il principale è chiamato *plant_control_room_db* e contiene gli eventi (allarme o manutenzione), le performance (di tipo *alert* o *plant*) e le informazioni sulle macchine registrate. La struttura dei suoi documenti e la descrizione dei suoi campi sono mostrate in figura 4.6. Gli eventi sono definiti dalla macchina a cui si riferiscono, il tempo in cui sono generati, una descrizione, una lettera che identifica se sono allarmi o manutenzioni e il loro livello. Gli allarmi possono essere di tipo *Normal*, *Warning* o *Critical*. Le manutenzioni possono avere un livello *In progress*, *Open* o *Expiring*. Le performance indicano lo stato generico dell'impianto. Il significato di *Alert performance* e *Plant performance* è spiegato in Dashboard. I documenti sulle macchine portano tutte le informazioni necessarie riguardo ad esse. Molto importanti sono i campi sullo status, le misure con le loro rispettive soglie e il token di autenticazione. L'utente può aggiungere o eliminare i campi facoltativi come meglio desidera.

Gli altri database hanno il nome $\{nome_macchina\}_db$ dove $\{nome_macchina\}$ è il nome univoco di ogni macchina registrata. Questi immagazzinano tutte le misurazioni effettuate dai sensori per il macchinario corrispondente. Siccome verranno registrati molti dati visto che essi sono inseriti periodicamente, si è pensato di creare un database per ogni dispositivo. In questo modo si ottengono risposte più veloci poiché avvengono su una quantità minore di tuple. Con un unico database si avrebbero tutte le misurazioni di ogni macchinario assieme dovendo così gestire una grossa mole di dati. La struttura dei documenti salvati è riportata in figura 4.7. Ogni documento rappresentante il ciclo macchina può contenere un numero indefinito di misure.

plant_control_room_db

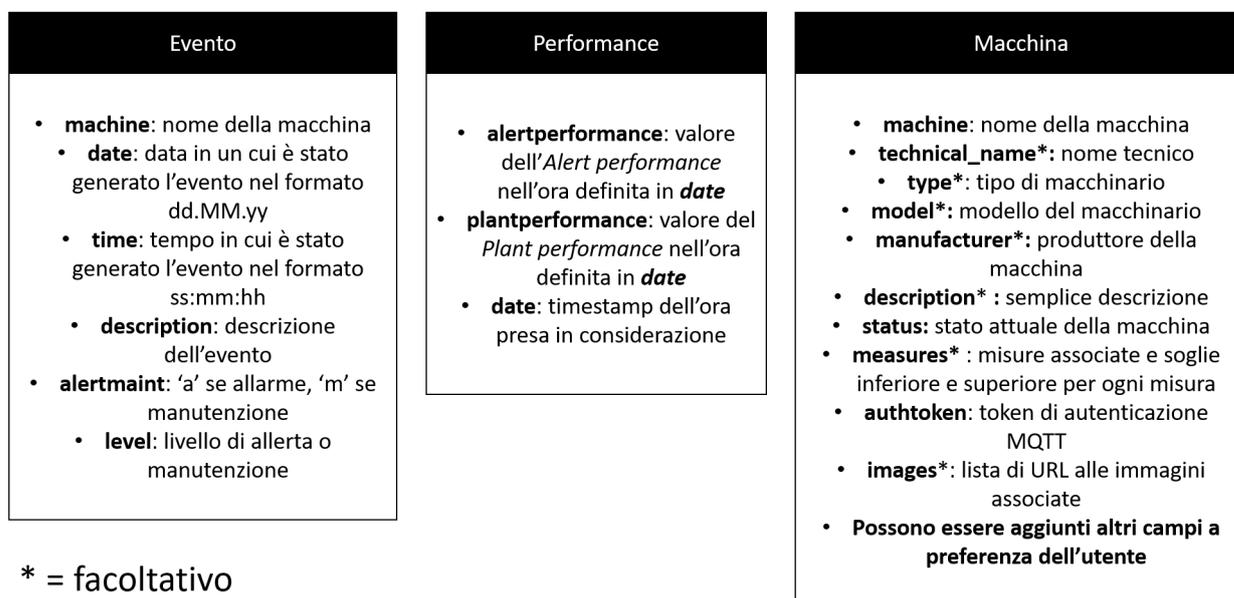


Figura 4.6. Struttura del database *plant_control_room_db*

{nome_macchina}_db



Figura 4.7. Struttura del database *nome_macchina_db*

4.7 Lato server - server Plant Control Room

Il lato server funge da proxy tra i client e la coppia database / server IBM Watson IoT Platform. Qualsiasi client, per ottenere i dati da visualizzare, chiede ciò che necessita al server Plant Control Room che si adopererà per scaricarli e inviarli in risposta. Allo stesso modo vengono effettuate inserzioni o modifiche, il client effettua una richiesta e il server la esegue. Questo è implementato in Node.js, la sua scelta è spiegata in Node.js. Sfrutta un'architettura di tipo REST, rendendo

facile per il client la comunicazione e il trasferimento dei dati. Oltretutto il protocollo utilizzato è privo di stato ovvero il server non deve salvare il contesto dei client rendendo il tutto più leggero, semplice e scalabile. Il server comunica con il database **Cloudant** grazie alle librerie **Cloudant Node.js Client** [171] e con il server **IBM Watson IoT Platform** tramite le librerie **Node.js Client Library** [172]. La comunicazione con la base di dati viene utilizzata per ottenere e salvare i dati presentati in Storage - Database **Cloudant**, il dialogo con il server **IBM Watson IoT Platform** è necessario per registrare una qualsiasi macchina ed ottenere il token associato ad essa. È stato scelto l'utilizzo di un proxy anziché la comunicazione diretta tra client e i vari server per fare in modo che l'utente non possa conoscere le richieste, le query e le URI sfruttate per la comunicazione; quindi la motivazione è dovuta a ragioni di sicurezza. Allo stesso tempo si rende il client più semplice e il server **Plant Control Room**, essendo una singola entità, può essere modificato e aggiornato facilmente senza dover installare una nuova versione in tutti i dispositivi client, evento molto più dispendioso. In figura 4.8 si mostrano le interazioni tra i vari dispositivi.

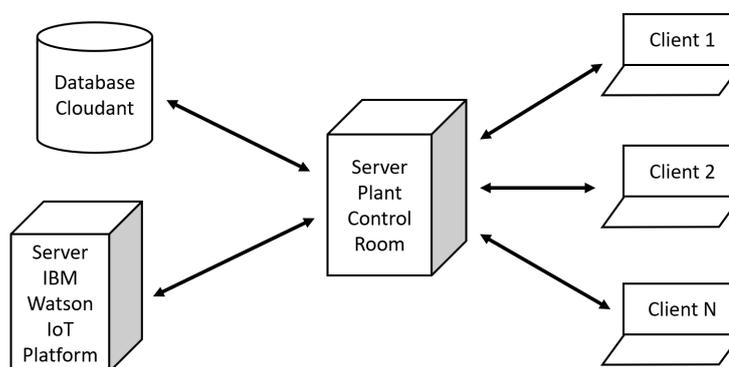


Figura 4.8. Interazioni intraprese dal server **Plant Control Room**

Ecco la lista delle API fornite dal server per il client:

- **POST /api/authenticate**. Controlla se lo username e password ricevuti sono corretti, per ora si può accedere solo con delle credenziali preimpostate.
- **GET /api/statuses**. Restituisce tutti gli status delle macchine dell'impianto.
- **GET /api/status/machine**. Restituisce lo status della macchina di nome *machine*.
- **GET /api/listmachines**. Restituisce una lista contenente i nomi di tutte le macchine.
- **PUT /api/machine**. Modifica le caratteristiche associate ad una certa macchina.
- **POST /api/machine**. Registra una macchina sul server **Watson IoT Platform** ottenendo il suo token di autenticazione e salva quest'ultimo sul database.
- **GET /api/events/kind**. Restituisce gli eventi di tipo *kind* in ordine cronologico. I tipi possono essere allarmi, manutenzioni oppure entrambi.
- **GET /api/events/kind/machine**. Restituisce gli eventi di tipo *kind* associati alla macchina *machine* in ordine cronologico. I tipi possono essere allarmi, manutenzioni oppure entrambi.
- **GET /api/detail/machine**. Restituisce le misurazioni degli ultimi 130 cicli della macchina *machine* in ordine decrescente rispetto al numero dei cicli.
- **GET /api/detail/fieldstoshow/machine**. Restituisce la lista delle misure da mostrare subito della macchina *machine* nella *pagina di dettaglio* descritta in Pagina di dettaglio dell'apparecchiatura.

- **GET /api/performances/date/time.** Restituisce i precedenti 49 valori del *Plant performance*, descritto in Dashboard. Essi dovrebbero rappresentare le performance dell'impianto industriale nelle ultime 48 ore.

4.8 Lato client - client Plant Control Room

Il lato client è la parte che gli operai utilizzeranno praticamente. Per lo sviluppo è stato impiegato **Angular 5**, la motivazione della sua scelta è spiegata in **Angular 5 o AngularJS**, e si è partiti dal template **SB Admin Angular5 BS4** [173]. Essendo un'applicazione web, può essere acceduto da un qualsiasi web browser, è stato testato con **Google Chrome 68.0.3440.75**, **Microsoft Edge 42.17134.1.0** e **Firefox 61.0.1**. Questo capitolo può essere sfruttato come manuale utente.

Lo scopo del lato client è mostrare i dati ricavati dalle macchine e gestire l'impianto in modo che i lavoratori possano effettuare tutte le operazioni semplicemente tramite un'interfaccia grafica. Come già spiegato in **Lato server - server Plant Control Room**, tutti i dati ricevuti ed inviati vengono richiesti o spediti al server **Plant Control Room** che lavora come proxy. Per ottimizzare la visualizzazione anche su smartphone e tablet, è stato sfruttato **Bootstrap** [174] in modo che, a seconda della grandezza dello schermo del dispositivo utilizzato, la disposizione degli elementi cambia da orizzontale a verticale e viceversa ottenendo una migliore interfaccia grafica, in questo modo nessun oggetto viene nascosto e si ottiene una maggiore semplicità e usabilità da parte dell'utente. Per i grafici sono state sfruttate le librerie **ng2-charts** [175] e **DevExtreme** [176]. In figura 4.9 è possibile vedere lo schema delle pagine web, le pagine per la gestione degli allarmi e manutenzioni sono state implementate dopo il mio lavoro, sono mostrate per avere una visione più completa dell'applicazione.

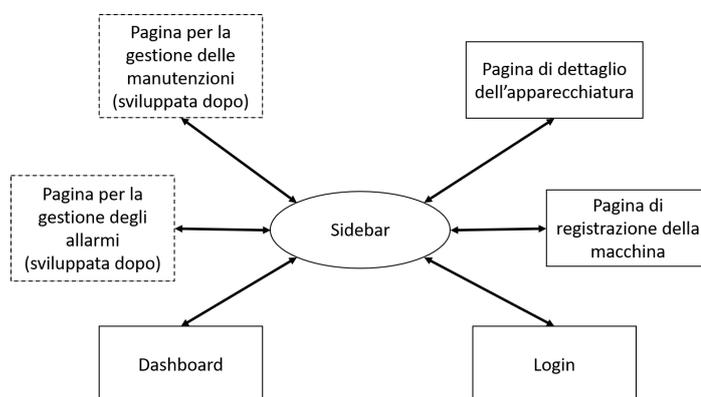


Figura 4.9. Plant Control Room - Schema pagine web

L'applicazione, al suo avvio, mostra una schermata di autenticazione ma, per ora, non è implementata poiché si accede con delle credenziali prestabilite; non è ancora possibile creare nuove utenze ed accedere con credenziali diverse. La pagina di login è presentata in figura 4.10.

4.8.1 Dashboard

Una volta effettuato il login si accede alla homepage o dashboard. Questa pagina ha lo scopo di mostrare lo stato attuale dell'impianto. Ecco la descrizione dei vari elementi mostrati:

- **Plant overview.** Mostra l'andamento dell'impianto nelle ultime 48 ore in un grafico di linea. Le due misure mostrate sono l'*Alert performance* e il *Plant performance* in una certa ora. Il *Plant performance* è un indicatore usato per avere un impatto immediato sul comportamento del sistema; siccome ogni macchina ha uno status che indica la sua salute, esso è

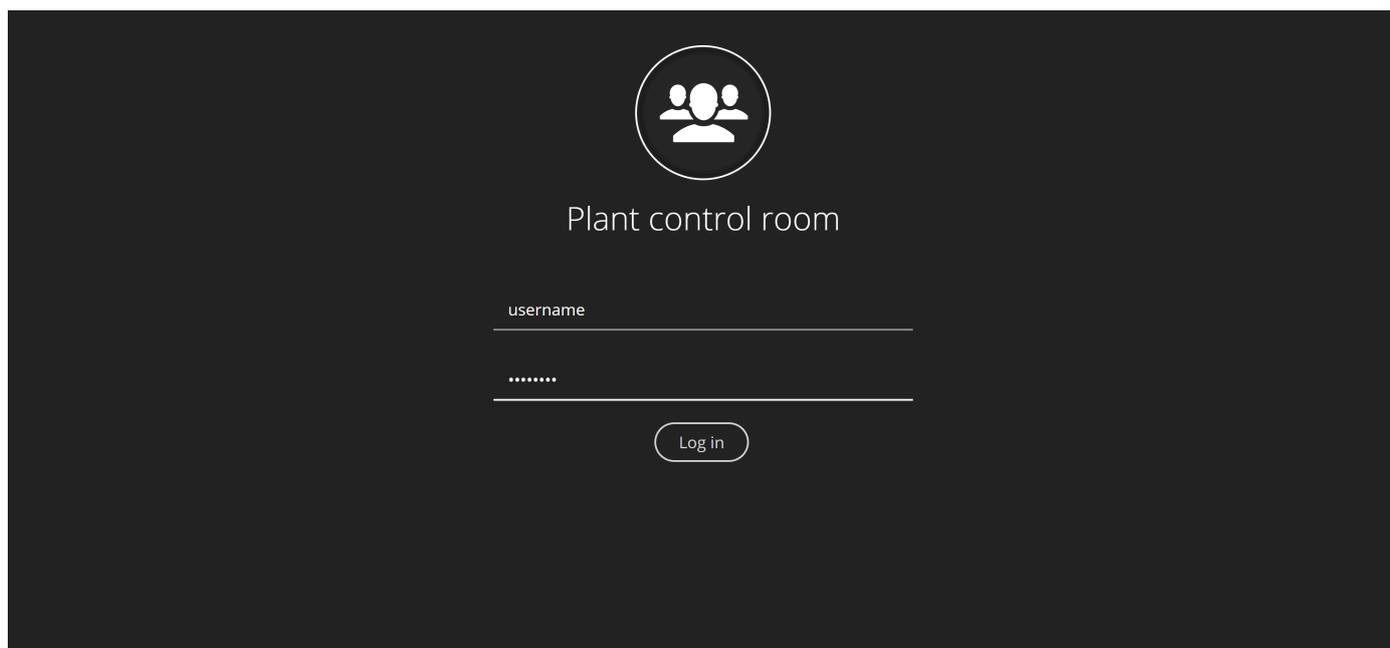


Figura 4.10. Plant Control Room - Pagina di login

calcolato come $\frac{100 \cdot pesoStatus}{numeroStatus}$ dove *numeroStatus* è la quantità degli status (quindi macchine) e *pesoStatus* è la loro somma dove gli status di tipo *Normal* valgono 1, i *Warning* hanno peso 0.5 e quelli *Critical* non vengono considerati. Si avrà quindi che, se tutte le macchine sono *Normal* allora *Plant performance* vale 100%, se tutti i macchinari sono *Critical* si avrà un risultato pari allo 0%. L'*Alert performance* è molto simile ma, anziché considerare gli status, tiene conto degli allarmi; il peso dei vari livelli è lo stesso presentato per il *Plant performance*.

- **Plant performance.** Mostra il *Plant performance* dello stabilimento nel momento attuale tramite un grafico a tachimetro. La spiegazione del *Plant performance* è già stata presentata nel punto precedente.
- **Equipment's status.** In un grafico a ciambella mostra la percentuale di macchine con status *Normal*, *Warning* e *Critical*.
- **Alerts.** Questa sezione mostra gli allarmi ancora non risolti, questi sono classificati, in base alla loro importanza, in *Critical*, *Warning* e *Normal*.
- **Maintenances.** Viene visualizzato il conteggio di tutte le manutenzioni registrate. Queste possono essere di tipo *In progress*, *Open* o *Expiring*.
- **Barra sottostante (Bottom bar).** Vengono visualizzate le quantità di allarmi generati nell'ultimo giorno, settimana, mese e anno. Per ora i valori mostrati sono numeri fissi.

In figura 4.11 è mostrata la dashboard appena presentata.

La navigazione del sito è fornita grazie ad una sidebar, essa presenta tutte le pagine disponibili che, al momento, sono la dashboard e le due sezioni presentate sotto. Grazie a ciò si può passare agilmente da una parte all'altra dell'applicazione rendendo il tutto più modulare, le prossime pagine da sviluppare potranno essere facilmente aggiunte poiché indipendenti dalle altre in modo da non modificare le parti già implementate. Da qui dovrebbe essere possibile effettuare il logout e accedere alle informazioni del proprio profilo, questa parte non è stata ancora sviluppata quindi non è disponibile per ora. La figura 4.12 mostra l'utilizzo della sidebar.

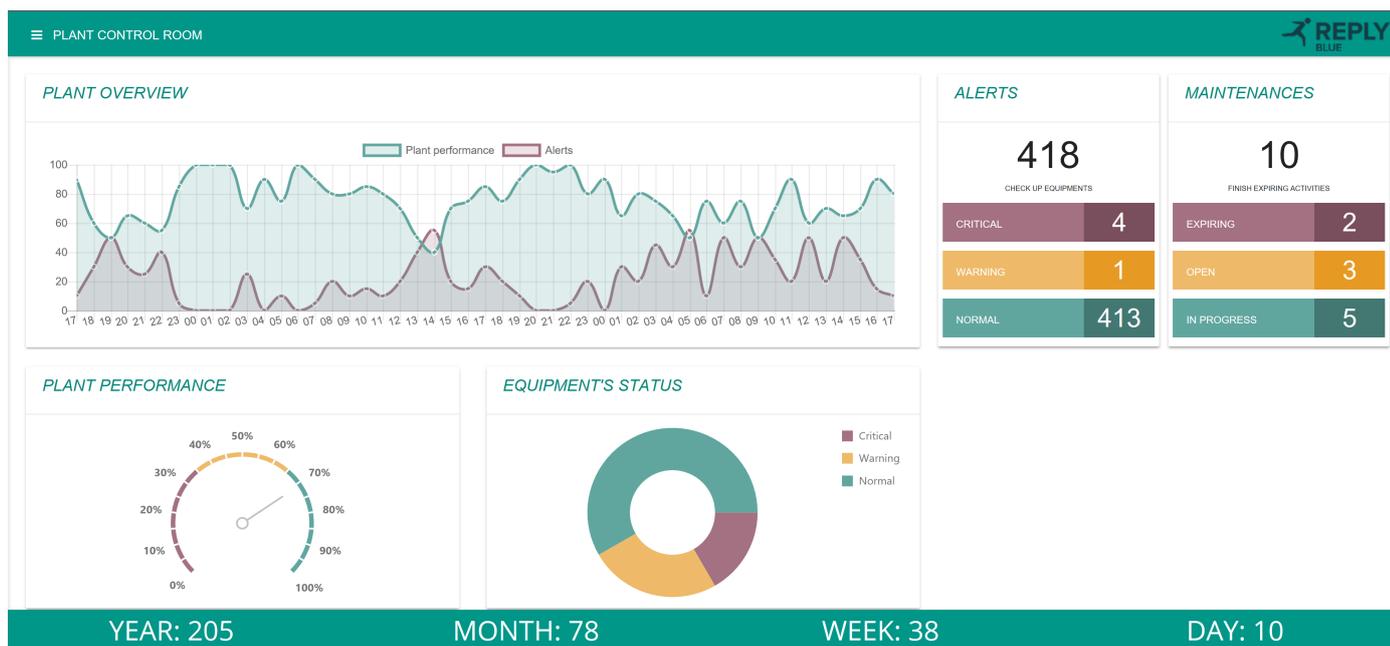


Figura 4.11. Plant Control Room - Dashboard

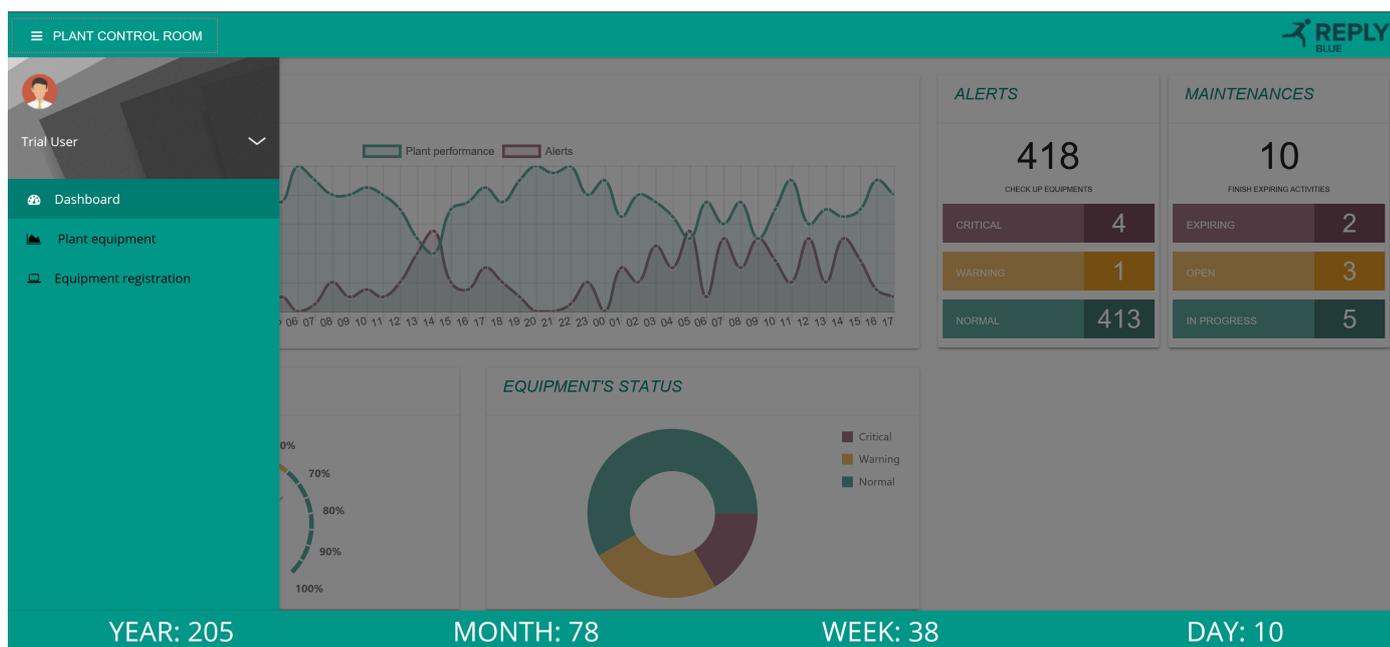


Figura 4.12. Plant Control Room - Dashboard, esempio di utilizzo sidebar

4.8.2 Pagina di dettaglio dell'apparecchiatura

Come presentato nella figura 4.13, nella pagina di dettaglio è possibile tramite un menu a tendina scegliere la macchina, la quale deve essere precedentemente registrata (spiegato sotto), e vedere le informazioni e dati legati ad essa. All'interno di una banda grigia vengono mostrate le caratteristiche associate al macchinario quali tipo, modello, status, eccetera; in particolare viene visualizzato il token associato. Nella versione aggiornata dopo il termine della mia esperienza presso Blue Reply

vengono anche mostrate le immagini associate all'attrezzatura selezionata.

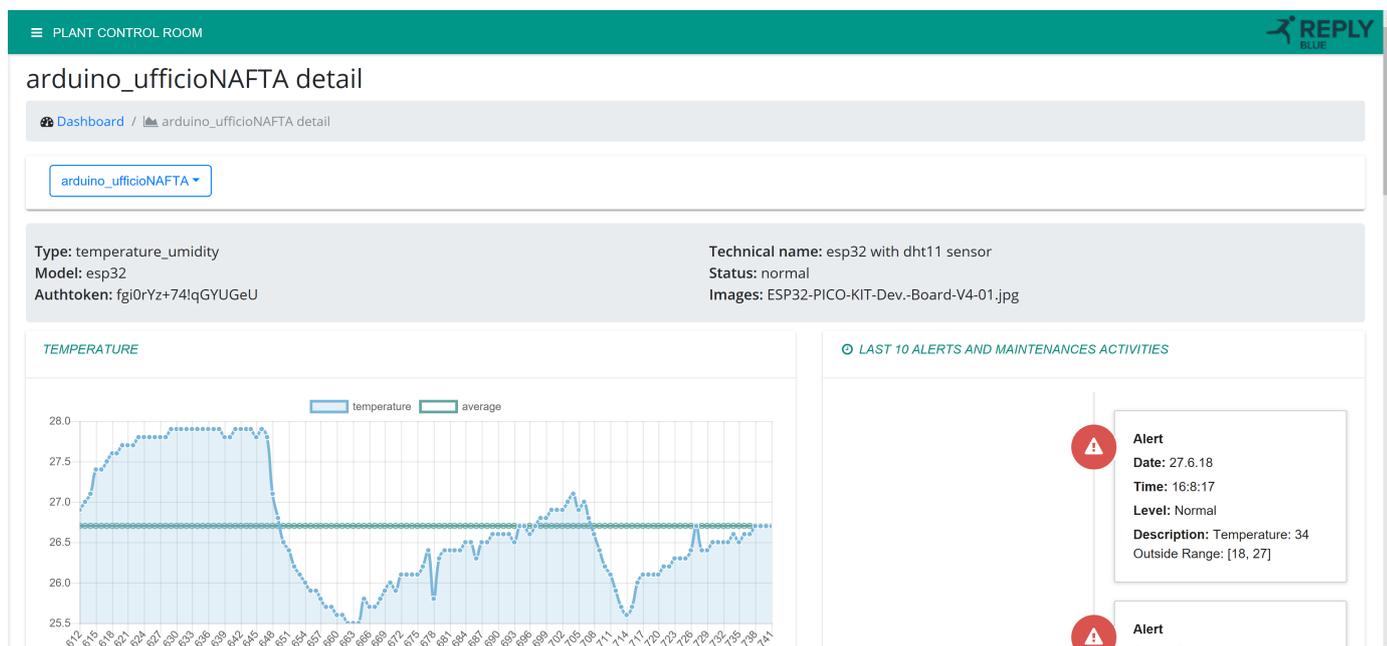


Figura 4.13. Plant Control Room - Pagina di dettaglio dell'apparecchiatura, parte 1

Nella figura 4.14 si possono vedere meglio i grafici e la timeline sviluppati. I grafici di linea mostrano i valori di una certa misura negli ultimi 130 cicli e la loro media in quell'intervallo. La pagina mostra automaticamente gli andamenti solo di una o due grandezze specificate nel database. Ciò è stato fatto perché alcune macchine hanno molti sensori (una/due decine) e quindi caricare tutti i dati sarebbe lento e pesante, allora vengono mostrati i grafici delle misure più importanti e gli altri possono essere visualizzati selezionandoli da un menu a tendina, la quale elenca tutte le grandezze registrate per quella macchina. La timeline sulla destra mostra gli ultimi 10 allarmi e manutenzioni legati al macchinario, per ognuno tra i due eventi si mostrano la data e ora della loro generazione, il livello e la descrizione associata.

4.8.3 Pagina di registrazione della macchina

Questa è divisa in due sezioni, la registrazione di una macchina e la modifica di una già registrata. Il primo caso è raffigurato in figura 4.15. L'applicazione non accetta dati da un dispositivo non registrato ovvero bisogna per forza associare un token univoco ad un oggetto prima di poterlo utilizzare. Durante la registrazione è necessario solamente inserire il nome dell'apparecchio. Le altre caratteristiche standard sono: nome tecnico, tipo, modello, produttore e descrizione però ne possono essere aggiunte altre personalizzate da un qualsiasi utente. L'applicazione non accetta misure non connesse ad una certa macchina quindi devono essere dichiarate durante la registrazione (o aggiunte successivamente nella sezione di modifica descritta sotto). Alla misura possono essere associate un nome "amichevole" (per intuire più facilmente il suo significato), un limite inferiore e uno superiore; se una misurazione è al di fuori dei limiti stabiliti viene generato un allarme tramite Node-RED. Nella versione successiva a quella sviluppata da me è anche possibile inserire dei documenti che potrebbero essere utili ai lavoratori come manuale utente, immagini, foto.

La sezione di modifica (figura 4.16) visualizza i dati di una macchina già registrata, tramite un menu a tendina è possibile vedere e selezionare il macchinario che vogliamo analizzare. Qua vengono mostrate le informazioni già descritte per il segmento di registrazione descritto sopra; sia le caratteristiche del dispositivo sia le misure associate possono venire modificate, aggiunte o

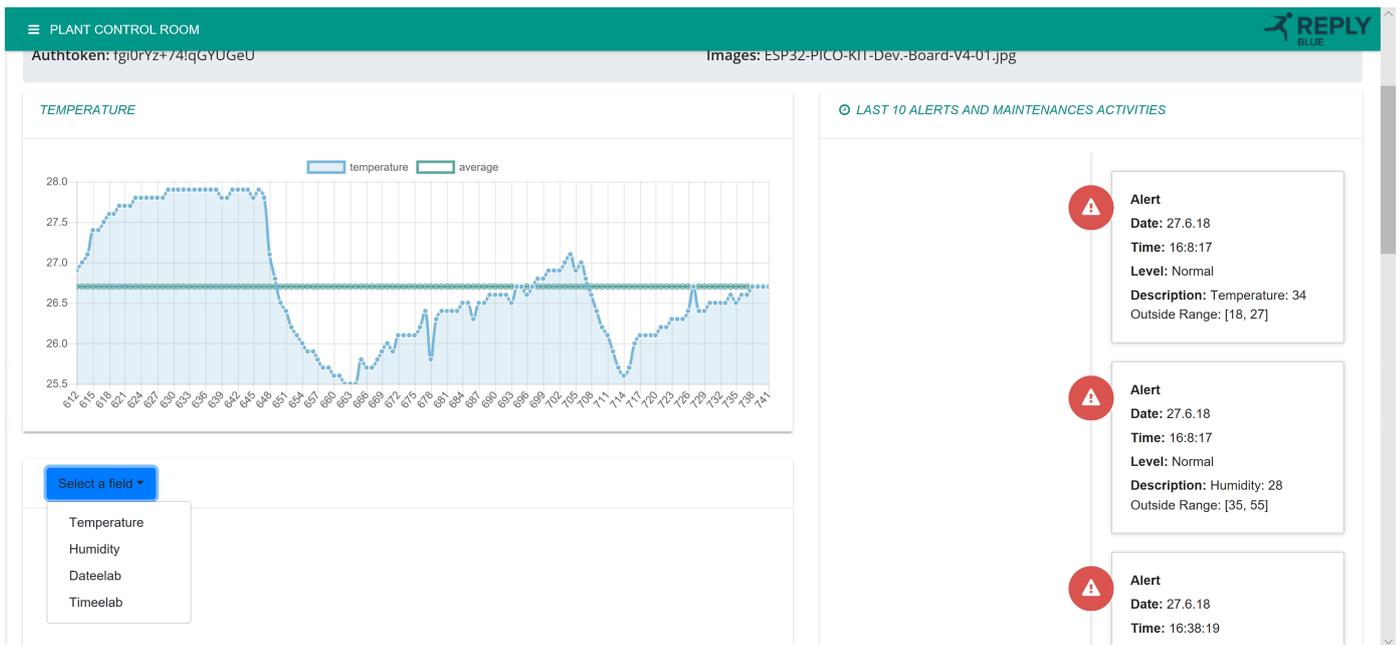


Figura 4.14. Plant Control Room - Pagina di dettaglio dell'apparecchiatura, parte 2

The screenshot displays the 'PLANT CONTROL ROOM' interface for machine registration. The top navigation bar includes 'Dashboard' and 'Machine registration'. Below the navigation bar are two buttons: 'Show and modify' and 'Add new'. The main content area is titled 'ADD NEW MACHINE' and contains a form with the following fields: 'Machine:', 'Technical Name:', 'Type:', 'Model:', 'Manufacturer:', and 'Description:'. Below these fields are two sections for adding new fields: 'Add New Machine Field:' and 'Add New Measure Field:', each with an 'Add' button. At the bottom of the form are 'Clean' and 'Save' buttons.

Figura 4.15. Plant Control Room - Pagina di registrazione, registrazione della macchina

rimosse. Nella versione successiva a quella sviluppata da me è possibile scaricare i file associati alla macchina o caricarne di nuovi.

The screenshot shows a web interface for 'PLANT CONTROL ROOM' with the 'REPLY BLUE' logo. A dropdown menu is set to 'Arduino ufficio01'. The form includes fields for 'Technical Name' (ESP32 with dht22 sensor), 'Type' (ESP), and 'Model' (32). There is an 'Images' field with a 'Delete' button. Below this is an 'Add New Machine Field' section with an 'Add' button. The 'Temperature' section has 'Friendly Name' (temperature), 'Lower Specification Limit' (16), and 'Upper Specification Limit' (27), each with a 'Delete' button. The 'Humidity' section has 'Friendly Name' (humidity), 'Lower Specification Limit' (10), and 'Upper Specification Limit' (100), each with a 'Delete' button. At the bottom is an 'Add New Measure Field' section with an 'Add' button.

Figura 4.16. Plant Control Room - Pagina di registrazione, modifica della macchina

4.9 Monitoraggio e analisi dei pattern anomali real-time

La parte per il monitoraggio e analisi legata alla manutenzione predittiva è stata sviluppata e presentata nella sezione Manutenzione predittiva. Essa non è stata realmente implementata su Plant Control Room quindi verrà descritto come ciò sarebbe dovuto essere in modo da avere una visione più completa dell'intero sistema. Per ampliare l'architettura con questo ulteriore elemento è necessario utilizzare il blocco di IBM Cloud chiamato IBM Watson Studio, la sua scelta è spiegata in IBM Watson Studio. Basta importare il progetto Jupyter nel server IBM Watson Studio in modo da renderlo disponibile alla struttura totale, il blocco IBM utilizzato è compatibile con Jupyter quindi non sono necessarie modifiche sul formato.

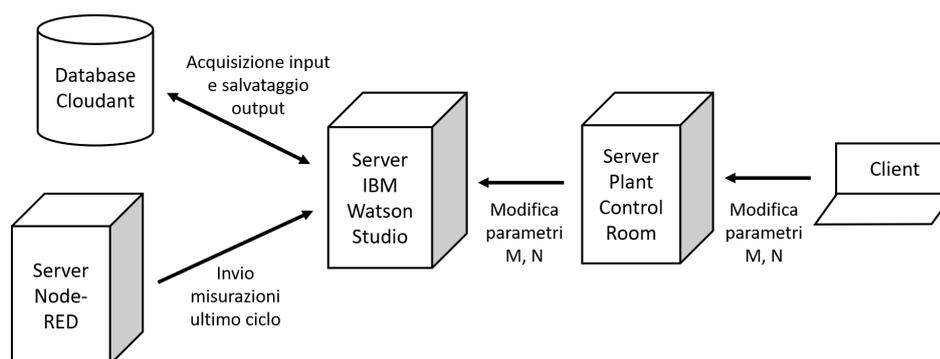


Figura 4.17. Interazioni del server IBM Watson Studio con gli altri componenti

Un evento importante da gestire è l'acquisizione e il salvataggio dei dati di input e output. L'applicazione è stata sviluppata per accedere alle misurazioni attraverso dei file ma è meglio che questi siano ottenuti e salvati tramite il database già implementato. L'applicazione dovrà richiedere i dati di input e salvare le predizioni in output per mezzo della base di dati Cloudant. Per avere un'analisi in tempo reale, bisogna fare in modo che il blocco Node-RED invii periodicamente al server

IBM Watson Studio i valori del ciclo macchina appena ricevuto, quest'ultime vengono processate dal modello che effettua la predizione e salva il risultato nel database. Si può fare in modo che, se si predice uno stato critico o un fallimento imminente, venga generato un allarme in modo da avvertire gli utenti. Si dovrebbe aggiungere, nell'applicazione web, una pagina per vedere tutti gli output generati in modo che gli operai possano vedere le previsioni sullo stato futuro della macchina e gestire manutenzioni e piani di lavoro di conseguenza. È bene implementare le modifiche tramite l'interfaccia grafica dei valori N e M per i quesiti 1 e 2 (spiegati nelle sezioni Quesito 1 - La macchina avrà un fallimento nei prossimi N secondi/ore/giorni/cicli macchina? e Quesito 2 - La macchina si sta comportando in modo anomalo?) in modo da rendere il tutto più flessibile. Bisognerebbe fare in modo che l'allenamento dei modelli venga compiuto periodicamente, per esempio ogni ora, in modo che essi siano istruiti secondo gli andamenti più recenti della macchina. Nella figura 4.17 è mostrata l'interazione del server IBM Watson Studio con gli altri elementi di Plant Control Room.

Capitolo 5

Conclusioni

5.1 Risultati ottenuti

Riguardo l'industria 4.0 e la PHM, sono state analizzate le loro discipline, pilastri e cultura in modo da conoscere i principi base sull'argomento. Si è anche vista l'architettura che li caratterizza e, più nel dettaglio, i loro componenti arrivando alle sfide e ai benefici che la loro applicazione comporta. Infine è avvenuta un'indagine sul loro stato attuale e previsioni per il futuro. Si è così avuta una visione a tutto tondo. È possibile concludere dicendo che entrambe sono in crescita anche se quest'intensità varia a seconda della nazione e azienda. Bisognerebbe quindi puntare su di esse per adattarsi al nuovo modello di business anziché rimanere ancorati a quello vecchio, tutte e due permettono di raggiungere risultati prima non possibili poiché precedentemente ci si basava in maggioranza sulle prestazioni individuali delle macchine e gli interventi umani. D'altro canto, in particolare la PHM rimane ancora una sfida e c'è molta ricerca da attuare prima che arrivi alla portata di tutti. La quasi totalità delle aziende industriali ha dei piani, alcuni in atto e altri per il futuro, per l'implementazione di entrambe, segno del fatto che le loro potenzialità e importanza sono riconosciute da molti.

L'applicazione sulla manutenzione predittiva mostra un esempio pratico di PHM. Essa dovrebbe essere completata collegandola ad una architettura come **Plant Control Room** in modo da ottenere dei dati in tempo reale da una macchina ed effettuare dei test sull'ambiente concreto. Questa non mostra grandi prestazioni però ha comunque lo scopo di introdurre il lavoro e il modus operandi in tale ambito. Si sono mostrate le potenzialità e i benefici che si otterrebbero da uno strumento simile. Oltretutto hanno avuto modo di essere applicate le conoscenze teoriche del settore imparando le principali tecniche di analisi, lo studio di un contesto reale e l'elaborazione dei dati.

Plant Control Room è quasi completa e viene continuata dai colleghi di **Blue Reply**. Una volta terminata verrà consegnata al cliente ed utilizzata. Rispecchia quindi uno strumento pratico e concreto. Il suo scopo è aiutare il manutentore nel suo lavoro rendendo il suo ruolo più incentrato sul controllo e monitoraggio rispetto alla manualità, in questo modo le conoscenze richieste sono più indirizzate sulla teoria rispetto che alla pratica. Oltretutto il lavoratore può farsi carico di maggiori responsabilità divenendo più intraprendente.

5.2 Sviluppi futuri

La tesi può venire ancora ampliata con alcuni elementi che momentaneamente sono stati tralasciati, sono quindi riportati ora in caso il lettore voglia approfondire.

Sarebbe interessante osservare un caso reale di una singola azienda prima e dopo l'implementazione dei componenti e principi dell'industria 4.0 e della PHM, ancora meglio considerare in modo

isolato il primo caso, poi il secondo ed infine la loro unione. L'analisi dovrebbe comprendere lo stato precedente della società, l'implementazione attuata e i cambiamenti finali raggiunti, di conseguenza si studiano le sfide, i benefici, i difetti, le mancate aspettative, l'impatto economico ottenuti a distanza di un certo periodo di tempo. In questo modo si tratterebbe tutto l'arco dell'evoluzione da industria 3.0 a 4.0 e dell'introduzione della prognostica.

Siccome sia l'industria 4.0 sia la PHM si stanno sviluppando nell'ultimo periodo, bisognerebbe seguire la loro crescita e andamento anno dopo anno.

Una proposta di approfondimento del lavoro già svolto è lo studio della situazione dell'industria 4.0 nei paesi oltre l'Italia (sezione Il caso italiano) o fuori dall'Unione europea (capitolo Iniziative nell'Unione europea), i più importanti sarebbero Stati Uniti e Giappone nel mondo e Germania, Francia, Regno Unito in Europa, in modo da vedere nel dettaglio l'applicazione di quest'evoluzione in diverse parti del mondo. Bisognerebbe anche seguire l'andamento del **Piano nazionale Impresa 4.0** in Italia, questo era stato previsto fino all'anno 2020 ma, visto il cambio di governo avvenuto nel Giugno 2018, è interessante sapere se verrà riconfermato, modificato o eliminato (durante la scrittura della tesi sembra che il piano stato stato riconfermato anche per il 2019 con alcune differenze).

L'analisi sulle incertezze (Sfide ed orientamento per il futuro per la PHM) può essere applicata ad un caso reale in modo da mostrare un tipico metodo di lavoro, ancora meglio se impiegato ad uno dei quesiti del programma presentato nel capitolo 3, Manutenzione predittiva. Quest'ultimo caso è un po' complicato perché, come già detto nella sezione apposita, non si conoscono le incertezze degli strumenti e dei modelli utilizzati. In compenso una sua attuazione porterebbe un fattore di completezza e affidabilità.

Per l'applicazione sulla manutenzione predittiva nel capitolo Manutenzione predittiva, possono essere apportate alcune modifiche, ampliamenti ed approfondimenti. Bisognerebbe testare il prodotto in caso reale allacciando una macchina ad iniezione, vedere le effettive prestazioni ed adoperare cambiamenti di conseguenza. Allo stesso tempo, dopo un certo periodo di utilizzo da parte del cliente, potrebbero sorgere problemi, servizi ed eventi non previsti e che necessitano ora una considerazione; in base al feedback restituito potrebbero rendersi indispensabili alcune azioni. Oltretutto si vedrebbero i risultati ottenuti tramite dati diversi dai file utilizzati e in quantità maggiore. Sarebbe meglio migliorare le prestazioni dei quesiti 2 e 3. Altro fatto da attuare è la visita reale all'impianto e alle macchine in modo da vedere il loro modo di operare, in particolare sarebbe rivelante l'incontro con i manutentori per sapere quali, come e con che frequenza i lavori di manutenzione venivano attuati nel concreto. Ciò non era stato possibile poiché il contratto per la tesi in azienda prevedeva la mia postazione unicamente nella sede di **Blue Reply**. Oltretutto è possibile aggiungere un ulteriore quesito: *qual è la probabilità che l'asset si guasti entro X unità di tempo per via della causa radice o del problema P_i ?* [122] Anziché restituire se la macchina fallirà entro X cicli, si restituisce la percentuale che l'attrezzatura ha di fallire. Il lavoro svolto sulla manutenzione predittiva potrebbe essere riutilizzato per ottenere un'analisi di qualità predittiva. Anziché voler predire i fallimenti, si vuole scoprire la qualità futura dei prodotti finali in modo da non sprecare tempo e denaro per creare oggetti non conformi alle regole aziendali. I dati da analizzare sarebbero gli stessi, bisognerebbe solo trovare un modo per associare l'input ad un indicatore di qualità. Momentaneamente i controlli avvengono tramite un'osservazione umana quindi bisogna trovare il modo di trasformare quest'analisi in parametri specifici e digitali.

Anche nell'applicazione **Plant Control Room** nel capitolo 4 (Implementazione soluzione) sono possibili alcune modifiche ed approfondimenti. Durante la scrittura della tesi e dopo la realizzazione dell'applicazione è uscita la nuova versione di **Angular**, la numero 6, si potrebbe aggiornare l'applicazione web in modo da rimanere al passo. La terminazione delle pagine mancanti sulla gestione di allarmi e manutenzioni è già stata svolta dai colleghi di **Blue Reply** ma dovrebbe ancora mancare il lato analisi poiché non previsto nel progetto. L'applicazione è stata terminata dall'azienda e il sottoscritto pensa che essa sia anche già stata consegnata al cliente per l'utilizzo. Non si conosce il feedback di quest'ultimo ma è possibile che ora siano necessari lavori di modifica causati da problemi, servizi mancanti, benefici constatati solo ora dal committente. Bisognerebbe implementare l'applicazione sulla manutenzione predittiva all'architettura già presentata in modo

da completarla e vedere la sua realizzazione pratica. Infine sviluppare il lato cognitivo in modo che le macchine associate si autoconfigurino ottimizzando il proprio operato.

5.3 Conoscenze acquisite

Grazie allo svolgimento della tesi si ha avuto l'opportunità di apprendere e applicare varie conoscenze, eccole presentate velocemente:

- **Punti fondamentali e cultura dell'industria 4.0.**
 - Si è partiti osservando la storia dell'evoluzione del settore industriale dalla sua nascita ad oggi considerando le motivazioni di questo sviluppo.
 - Nascita dell'industria 4.0 grazie al piano tedesco **Industrie 4.0** analizzando le sue motivazioni, principi e obiettivi.
 - Delineazione architettura e, in modo più approfondito, alcuni suoi componenti quali sistemi ciberfisici, **big data** e **Internet delle cose**.
 - Deduzione criteri fondamentali, benefici derivanti e obiettivi.
 - Infine si è studiato lo stato attuale dell'industria 4.0 nel mondo, Europa, Italia e il suo andamento futuro.
- **Basi della PHM.**
 - Per prima cosa sono state comprese la gestione e la classificazione delle manutenzioni.
 - Ricerche sulle discipline diagnostica e prognostica con conseguente loro definizione, lavoro, utilizzo, benefici e suddivisione.
 - Spiegazione dell'architettura completa e dei suoi componenti.
 - Si sono esaminati i principi e i benefici derivanti dalla PHM e sua applicazione ai sistemi ciberfisici e **big data**.
 - Per avere una conoscenza più pratica si è osservato un classico modo di operare.
 - Approfondimento sullo stato attuale e le sfide da affrontare per il futuro.
- **Sviluppo di un'applicazione riguardante la manutenzione predittiva.**
 - Innanzitutto si sono apprese le fondamentali tecniche di apprendimento automatico, linguaggio **Python**, principali librerie utili per la manutenzione predittiva e piattaforma **Jupyter**.
 - Metodologia per lo studio di un caso di analisi (in questo caso macchina ad iniezione).
 - Esaminazione e visualizzazione di certi dati in input grazie a grafici di linea, dispersione, distribuzioni, diagrammi a scatola e baffi, selezione e filtraggio variabili in modo da estrapolare più informazioni possibili.
 - Apprendimento del lavoro sull'elaborazione dati quali normalizzazione, filtraggio, **PCA**, ricampionamento e creazione di nuove variabili. Applicazione e comparazione dei modelli di apprendimento.
 - Selezione e valutazione dei criteri per stabilire la qualità del risultato ed esecuzione delle fasi di test per la scelta del metodo migliore da adottare.
- **Sviluppo di un'architettura e applicazione IoT.**
 - Grazie a questa sezione si sono appresi i linguaggi **Angular 5** e **Node.JS**, le piattaforme **IBM Cloud** e **Node-RED**, database **Cloudant** e si è fatta maggior pratica con **HTML** e **CSS**.
 - Studio e applicazione pratica di un'architettura **IoT** e dei suoi elementi associati.

- Gestione e approccio con una macchina industriale, nello specifico macchina ad iniezione.
- Acquisizione e manipolazione dati in tempo reale.
- Creazione, strutturazione ed utilizzo di un database NoSQL tramite **Cloudant**.
- Sviluppo di un server proxy in linguaggio **Node.JS** secondo le specifiche del protocollo **REST**.
- Programmazione di un'applicazione web tramite il framework **Angular 5** per la visualizzazione e gestione dati, allarmi, manutenzioni e macchine.

Bibliografia

- [1] Vasja Roblek, Maja Meško, Alož Kraprež, *A Complex View of Industry 4.0*, SAGE Open, 10.1177/2158244016653987, Aprile-Giugno 2016, <http://journals.sagepub.com/doi/abs/10.1177/2158244016653987>
- [2] Jan Smit, Stephan Kreutzer, Carolin Moeller, Malin Carlberg, *Policy department A: economic and scientific policy - Industry 4.0*, [http://www.europarl.europa.eu/RegData/etudes/STUD/2016/570007/IPOL_STU\(2016\)570007_EN.pdf](http://www.europarl.europa.eu/RegData/etudes/STUD/2016/570007/IPOL_STU(2016)570007_EN.pdf), document requested by the European Parliament's Committee on Industry, Research and Energy(ITRE), Febbraio 2016
- [3] Vepa Atamuradov, Kamal Medjaher, Pierre Dersin, Benjamin Lamoureux, Noureddine Zerhouni, *Prognostics and Health Management for Maintenance Practitioners - Review, Implementation and Tools Evaluation*, International Journal of Prognostics and Health Management, ISSN 2153-2648, 2017 060, https://www.phmsociety.org/sites/phmsociety.org/files/phm_submission/2016/ijphm_17_060.pdf
- [4] IoT analytics, Knud Lasse Lueth, *4 US companies classified as IoT leaders as Asian players catch up — IoT company ranking - Q3/Q4 2015 update*, <https://iot-analytics.com/4-us-companies-classified-iot-leaders-iot-revenue>, 7 Dicembre 2015
- [5] IoT analytics, Zana Diaz Williams, *The Top 20 Companies Enabling Predictive Maintenance*, <https://iot-analytics.com/top-20-companies-enabling-predictive-maintenance>, 6 Aprile 2017
- [6] IBM Cloud Docs, *Platform review*, <https://console.bluemix.net/docs/overview/ibm-cloud.html#overview>, ultimo aggiornamento il 22 Maggio 2018
- [7] Wikipedia, l'enciclopedia libera *Platform as a service* https://it.wikipedia.org/wiki/Platform_as_a_service
- [8] Interoute, GTT, *Cloud Articles* <https://www.interoute.it/what-iaas>
- [9] IBM Cloud, *Catalogo IBM*, <https://console.bluemix.net/catalog/>
- [10] HTML.it, Paolo Patierno, *Installare Node-RED* <http://www.html.it/pag/51924/installare-node-red-node-js-per-liot/>, 10 Febbraio 2015
- [11] Node-RED, *Node-RED home*, <https://nodered.org/>
- [12] IBM Cloud Docs, *About Watson IoT Platform*, https://console.bluemix.net/docs/services/IoT/iotplatform_overview.html#about_iotplatform, 22 Febbraio 2018
- [13] piattaforma IBM Watson IoT, *IBM Watson IoT Homepage*, <https://internetofthings.ibmcloud.com/#/>
- [14] IBM Cloud Docs, *Cloudant NoSQL DB - Overview* <https://console.bluemix.net/docs/services/Cloudant/cloudant.html#overview>
- [15] IBM, *Cloudant* <https://www.ibm.com/cloud/cloudant>
- [16] techopedia, *Database as a Service (DBaaS)*, <https://www.techopedia.com/definition/29431/database-as-a-service-dbaas>
- [17] Wikipedia, l'enciclopedia libera, *AngularJS*, <https://en.wikipedia.org/wiki/AngularJS>
- [18] HTML.it, Andrea Chiarelli, *AngularJS è un framework, non una libreria!*, <http://www.html.it/pag/52588/angularjs-e-un-framework-non-una-libreria>
- [19] Medium, TechMagic, *ReactJS vs Angular5 vs Vue.js - What to choose in 2018?*, <https://medium.com/@TechMagic/reactjs-vs-angular5-vs-vue-js-what-to-choose-in-2018-b91e028fa91d>

- [20] Stack overflow, *Developer Survey Results 2017*, <https://insights.stackoverflow.com/survey/2017#technology>
- [21] Wikipedia, l'enciclopedia libera, *Node.js*, <https://en.wikipedia.org/wiki/Node.js>
- [22] HTML.it, Alberto Bottarini, *Introduzione a Node.js*, <http://www.html.it/pag/32814/introduzione-a-node-js/>, 17 Aprile 2012
- [23] w3schools, *Node.js Introduction*, https://www.w3schools.com/nodejs/nodejs_intro.asp
- [24] Wikipedia, l'enciclopedia libera, *MEAN (software bundle)*, [https://en.wikipedia.org/wiki/MEAN_\(software_bundle\)](https://en.wikipedia.org/wiki/MEAN_(software_bundle))
- [25] IBM Watson, *Watson Studio overview*, <https://datascience.ibm.com/docs/content/getting-started/overview-ws.html?linkInPage=true>, ultimo aggiornamento 4 Maggio 2018
- [26] Wikipedia, l'enciclopedia libera, *Project Jupyter*, https://en.wikipedia.org/wiki/Project_Jupyter
- [27] Jupyter Notebook, *What is the Jupyter Notebook?*, <http://jupyter-notebook.readthedocs.io/en/stable/examples/Notebook/What%20is%20the%20Jupyter%20Notebook.html>
- [28] i-scoop, *Industry 4.0: the fourth industrial revolution - guide to Industrie 4.0*, https://www.i-scoop.eu/industry-4-0/#Industry_40_the_essence_explained_in_a_nutshell
- [29] Demetrius Klitou, Johannes Conrads, Morten Rasmussen, CARSA, Laurent Probst, Bertrand Pedersen, PwC *Germany: Industrie 4.0*, https://ec.europa.eu/growth/tools-databases/dem/monitor/sites/default/files/DTM_Industrie%204.0.pdf, report per la commissione europea, gennaio 2017
- [30] Wikipedia, l'enciclopedia libera, *Industry 4.0* https://en.wikipedia.org/wiki/Industry_4.0
- [31] Henning Kagermann, National Academy of Science and Engineering, Wolfgang Wahlster, German Research Center for Artificial Intelligence, Johannes Helbig, Deutsche Post AG, *Recommendations for implementing the strategic initiative INDUSTRIE 4.0*, report, Aprile 2013
- [32] Sentryo, *The 4 industrial revolutions*, <https://www.sentryo.net/the-4-industrial-revolutions/>
- [33] Wikipedia, l'enciclopedia libera, *Rivoluzione industriale*, https://it.wikipedia.org/wiki/Rivoluzione_industriale#Esplosione_demografica
- [34] Treccani, *Rivoluzione industriale*, http://www.treccani.it/enciclopedia/rivoluzione-industriale_%28Enciclopedia-delle-scienze-sociali%29/
- [35] Wikipedia, l'enciclopedia libera, *Seconda rivoluzione industriale*, https://it.wikipedia.org/wiki/Seconda_rivoluzione_industriale
- [36] Wikipedia, l'enciclopedia libera, *Terza rivoluzione industriale*, https://it.wikipedia.org/wiki/Terza_rivoluzione_industriale
- [37] Daniela Maňovčíková, *Industry 4.0 as the Culprit of Unemployment*, 12th IWKM 2017, 12 - 13 October 2017, Trenčín, Slovakia
- [38] Mario Hermann, Tobias Pentek, Boris Otto, *Design Principles for Industrie 4.0 Scenarios*, IEEE computer society, 10.1109/HICSS.2016.488, 2016
- [39] Walter Gasparetto, *Concetti Industria 4.0*, <https://www.confindustria.ud.it/upload/pagine/Industria%2040/Gasparetto%20-%20Fraunhofer.pdf>
- [40] Mario Hermann, Tobias Pentek, Boris Otto, *Design Principles for Industrie 4.0 Scenarios: A Literature Review*, ResearchGate, 10.13140/RG.2.2.29269.22248, 2015
- [41] Otto motors, *5 key industry 4.0 technologies*, <https://ottomotors.com/blog/5-industry-4-0-technologies>
- [42] L.Monostori, B.Kádár, T.Bauernhansl, S.Kondoh, S.Kumara, G.Reinhart, O.Sauer, G.Schuh, W.Sihn, K. Ueda, *Cyber-physical systems in manufacturing*, <https://www.sciencedirect.com/science/article/pii/S0007850616301974>, CIRP Annals, Volume 65, Issue 2, 2016, Pages 621-641
- [43] Ptolemy Project, Philip Asare, David Broman, Edward A. Lee, Gerro Prinsloo, Martin Torngren, S. Shyam Sunder, *Cyber-Physical Systems*, <https://ptolemy.berkeley.edu/projects/cps/>

- [44] Jay Lee, *Industry 4.0 in Big Data Environment*, https://www.researchgate.net/publication/285163697_Industry_40_in_Big_Data_Environment, German Harting Magazine Technology Newsletter, 2013
- [45] Jay Lee, Behrad Bagheri, Hung-An Kao, *A Cyber-Physical Systems architecture for Industry 4.0-based manufacturing systems*, <https://www.sciencedirect.com/science/article/pii/S221384631400025X>, Society of Manufacturing Engineers (SME), 2014
- [46] Luigi Atzori, Antonio Iera, Giacomo Morabito, *The Internet of Things: A survey*, *Computer Networks*, Volume 54, Issue 15, 28 Ottobre 2010, Pagine 2787-2805, <https://www.sciencedirect.com/science/article/pii/S1389128610001568>
- [47] Hermann Kopetz, *Internet of Things*, *Real-Time Systems* pagine 307-323, 26 Febbraio 2011, https://link.springer.com/chapter/10.1007/978-1-4419-8237-7_13
- [48] Felix Wortmann, Kristina Fluchter, *Internet of Things - Technology and Value Added*, *Bus Inf Syst Eng* 57(3):221?224 (2015), DOI: 10.1007/s12599-015-0383-3, <https://link.springer.com/content/pdf/10.1007/s12599-015-0383-3.pdf>
- [49] Shen Yin, Okyay Kaynak, *Big Data for Modern Industry: Challenges and Trends*, *Proceedings of the IEEE* (Volume: 103, Issue: 2, Feb. 2015), DOI: 10.1109/JPROC.2015.2388958, <https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=7067026&tag=1>
- [50] Deloitte, *Big data: Challenges and success factors*, https://www2.deloitte.com/content/dam/Deloitte/it/Documents/deloitte-analytics/bigdata_challenges_success_factors.pdf, 2013
- [51] Mark Beyer, *Gartner says solving big data challenge involves more than just managing volumes of data*, Gartner, 10 Luglio 2011, <https://www.gartner.com/newsroom/id/1731916>
- [52] IBM and Said Business School, *Analytics: The real-world use of big data: How innovative enterprises extract value from uncertain data*, IBM Institute for Business Value and Said Business School Executive Report, Ottobre 2012, <https://www-01.ibm.com/common/ssi/cgi-bin/ssialias?htmlfid=GBE03519USEN>
- [53] Philip Russom, *Big Data Analytics - TDWI best practices report*, <https://vivomente.com/wp-content/uploads/2016/04/big-data-analytics-white-paper.pdf>, quarto trimestre 2011
- [54] Li Da Xu, Eric L. Xu, Ling Li, *Industry 4.0: state of the art and future trends*, <https://www.tandfonline.com/doi/pdf/10.1080/00207543.2018.1444806?needAccess=true>, *International Journal of Production Research*, 56:8, 2941-2962, DOI: 10.1080/00207543.2018.1444806, 2018
- [55] GE Reports, François Barbier, *5 Trends For The Future Of Manufacturing*, <https://www.ge.com/reports/5-trends-future-manufacturing/>
- [56] Mayuri Patel, *Top Industry 4.0 Trends to Watch for in 2018*, <https://www.einfochips.com/blog/top-industry-4-0-trends-watch-2018/>, 24 Gennaio 2018
- [57] Industrial Automation Asia, *Top 5 Industry 4.0 Trends In Manufacturing*, <http://www.iaasiaonline.com/sector-spotlight/top-5-industry-4-0-trends-manufacturing-2/>, 21 Giugno 2017
- [58] M.Sc. Lukas Merkel, *Future trends of industry 4.0 cognitive assistance systems in manufacturing*, <http://s3platform.jrc.ec.europa.eu/documents/20182/221266/11.+Merkel+-+Fraunhofer+IGCV+-+keynote+-+Future+Trends.pdf/36d3df42-59df-4ed9-93ad-7a0505207adc>, 20 Settembre 2017
- [59] World Economic Forum, Davos-Klosters, *Deep Shift Technology Tipping Points and Societal Impact* 2015
- [60] World economic forum, Johan Aurik, Helena Leurent, *Which countries are best prepared for the future of production?*, <https://www.weforum.org/agenda/2018/01/time-for-action-who-is-really-ready-for-the-future-of-production/>, 11 Gennaio 2018
- [61] World economic forum, in collaboration with A.T. Kearney, *Readiness for the Future of Production Report 2018*, http://www3.weforum.org/docs/FOP_Readiness_Report_2018.pdf

- [62] European Commission, Demetrius Klitou, Johannes Conrads & Morten Rasmussen, CARSA and Laurent Probst & Bertrand Pedersen, PwC, *Key lessons from national industry 4.0 policy initiatives in Europe*, https://ec.europa.eu/growth/tools-databases/dem/monitor/sites/default/files/DTM_Policy%20initiative%20comparison%20v1.pdf, May 2017
- [63] Ministero dello Sviluppo Economico, *Presentato il Piano nazionale Industria 4.0*, <http://www.sviluppoeconomico.gov.it/index.php/it/194-comunicati-stampa/2035187-il-ministro-dello-sviluppo-economico-carlo-calenda-illustra-il-piano-nazionale-industria-4-0>
- [64] Il sole 24 ore, Stefano Denicolai, *Lavorare in fabbrica è una scelta di ripiego? Errore, è il futuro*, http://www.econopoly.ilsole24ore.com/2018/05/08/fabbrica-lavoro-futuro/#_ftn2
- [65] AGI, Arcangelo Rociola, *Come trasformare un'azienda con il digitale. Guida in 7 punti al Piano Industria 4.0*, https://www.agi.it/economia/piano_industria_40_come_funziona_superammortamento-1461853/news/2017-02-07/
- [66] Ministero dello Sviluppo Economico, *Piano nazionale Industria 4.0*, http://www.sviluppoeconomico.gov.it/images/stories/documenti/PIANO-NAZIONALE-INDUSTRIA-40_ITA.pdf
- [67] Ministero dello Sviluppo Economico, *Beni strumentali ("Nuova Sabatini")*, <http://www.sviluppoeconomico.gov.it/index.php/it/incentivi/impresa/beni-strumentali-nuova-sabatini>
- [68] AGI, Gaia Vendettuoli, *Ecco i risultati del piano Industria 4.0 (che cambia nome). Calenda: "Adesso fase due"*, https://www.agi.it/economia/industria_4_0_calenda_ministero_sviluppo-2168860/news/2017-09-20/
- [69] Ministero dello Sviluppo Economico, *Piano nazionale impresa 4.0 - Risultati 2017 - Azioni 2018*, http://www.sviluppoeconomico.gov.it/images/stories/documenti/impresa_40_risultati_2017_azioni_2018.pdf
- [70] Ministero dello Sviluppo Economico, *Piano nazionale impresa 4.0 - Guida agli investimenti*, http://www.sviluppoeconomico.gov.it/images/stories/documenti/investimenti_impresa_40_ita.pdf
- [71] Il sole 24 ore, Katy Mandurino, *Percorso a quattro tappe per arrivare all'industria 4.0*, <http://www.ilsole24ore.com/art/impresa-e-territori/2018-05-04/percorso-quattro-tappe-arrivare-all-industria-40-130920.shtml?uuid=AEDIp0iE,04/05/2018>
- [72] Karsten Schweichhart, *Reference Architectural Model Industrie 4.0 (RAMI 4.0)*, https://ec.europa.eu/futurium/en/system/files/ged/a2-schweichhart-reference-architectural_model_industrie_4.0_ami_4.0.pdf
- [73] Phoenix contact, *Il modello di riferimento per l'architettura tridimensionale RAMI 4.0*, https://www.phoenixcontact.com/online/portale/it?1dmy&urile=wcm:path:/itit/web/offcontext/insite_landing_pages/1323f37f-e566-4009-8645-661c715cea23/6ddf5dfb-dbc8-47c8-8f1a-dc915d263cd3/605016fb-ed97-4b22-a6fb-de1f93556226/605016fb-ed97-4b22-a6fb-de1f93556226
- [74] Marcos A. Pisching, Marcosiris A.O. Pessoa, Fabrício Junqueira, Diolino J. dos Santos Filho, Paulo E. Miyagi, *An architecture based on RAMI 4.0 to discover equipment to process operations required by products*, *Computers & Industrial Engineering*, 2 Gennaio 2018, <https://www.sciencedirect.com/science/article/pii/S0360835217306034>
- [75] Peter Adolphs, *RAMI 4.0 - An architectural Model for Industrie 4.0*, <https://www.omg.org/news/meetings/tc/berlin-15/special-events/mfg-presentations/adolphs.pdf>, 18 Giugno 2015
- [76] IBM, *Internet of Things for insights from connected devices - Industrie 4.0*, https://www.ibm.com/cloud/garage/architectures/iotArchitecture/industrie_40
- [77] Jamie Baalis Coble *Merging data sources to predict remaining useful life - An automated method to identify prognostic parameters*, PhD diss, University of Tennessee, 2010
- [78] Enrico Zio, *Prognostics and Health Management of Industrial Equipment*, Seifedine Kady, *Diagnostics and Prognostics of Engineering Systems: Methods and Techniques*, IGI

- Global, pp.333-356, 2012, <https://hal-supelec.archives-ouvertes.fr/hal-00778377/document>
- [79] Veronica Fornlöf *Improved remaining useful life estimations for on-condition parts in aircraft engines*, Skövde: University of Skövde, 2016, ISBN: 978-91-981474-9-0 <https://www.diva-portal.org/smash/get/diva2:945577/FULLTEXT01.pdf>
- [80] Gregory W. Vogl, Brian A. Weiss, Moneer Helu, *A review of diagnostic and prognostic capabilities and best practices for manufacturing*, Springer Science+Business Media New York, DOI: 10.1007/s10845-016-1228-8, 2015, https://ws680.nist.gov/publication/get_pdf.cfm?pub_id=917304
- [81] Thamo Sutharssan, Stoyan Stoyanox, Chris Bailey, Chunian Yin, *Prognostic and health management for engineering systems: a review of the data-driven approach and algorithms*, Journal of Engineering, 10.1049/joe.2014.0303, 2015
- [82] Hatem M. Elattar, Hamdy K. Elminir, A. M. Riad, *Prognostics: a literature review*, Springerlink.com, 10.1007/s40747-016-0019-3, 2016, <https://link.springer.com/article/10.1007/s40747-016-0019-3>
- [83] Treccani - la cultura italiana, *Diagnostica*, <http://www.treccani.it/enciclopedia/diagnostica/>
- [84] Treccani - la cultura italiana, *Diagnosi*, <http://www.treccani.it/vocabolario/diagnosi/>
- [85] Treccani - la cultura italiana, *Prognostico*, <http://www.treccani.it/vocabolario/prognostico/>
- [86] Treccani - la cultura italiana, *Prognosi*, <http://www.treccani.it/vocabolario/prognosi/>
- [87] Arthur Samuel, *Some Studies in Machine Learning Using the Game of Checkers*, IBM Journal of Research and Development. 3 (3): 210?229. doi:10.1147/rd.33.0210
- [88] Kamal Medjaher, Nouredine Zerhouni *Hybrid prognostic method applied to mechatronic systems*, International Journal of Advanced Manufacturing Technology, 10.1007/s00170-013-5064-0, 2013
- [89] Carl S. Byington, Michael J. Roemer, Thomas Galie, *Prognostic enhancements to diagnostic systems for improved condition-based maintenance*, IEEE aerospace conference proceedings, Big Sky, Montana, 2002
- [90] Rui Li, Wim J.C. Verhagen, Richard Curran, *A Functional Architecture of Prognostics and Health Management using a Systems Engineering Approach*, Proceedings of the European Conference of the PHM Society, Vol 4 No 1 (2018), 2 Luglio 2018, <https://www.phmpapers.org/index.php/phme/article/view/362>
- [91] Xinwei Li, Wenjin Zhang, *Design of Prognostic and Health Management Structure for UAV System*, 2011 21st International Conference on Systems Engineering, DOI: 10.1109/ICSEng.2011.11, 16-18 Agosto 2011, <https://ieeexplore.ieee.org/document/6041811/metrics#metrics>
- [92] Dennis Hoffman, *Prognostics and Health Management (PHM) / Condition Based Maintenance (CBM)*, l'articolo è parte del IEEE Reliability Society 2007 Annual Technology Report, <https://rs.ieee.org/tech-activities/38-2007>
- [93] Romano Patrick, Matthew J. Smith, Bin Zhang, Carl S. Byington, George J. Vachtsevanos, Romeo Del Rosario, *Diagnostic enhancements for air vehicle HUMS to increase prognostic system effectiveness*, 2009 IEEE Aerospace conference, DOI: 10.1109/AERO.2009.4839653, 24 Aprile 2009, <https://ieeexplore.ieee.org/document/4839653>
- [94] AzureML Team, *Predictive Maintenance*, <https://gallery.azure.ai/Solution/Predictive-Maintenance-10>, 23 Settembre 2016
- [95] , Fidan Boylu Uz, olprod, OpenLocalizationService, *Guida tecnica del modello di soluzione Cortana Intelligence per la manutenzione predittiva nel settore aerospaziale e in altri campi* <https://docs.microsoft.com/it-it/azure/machine-learning/team-data-science-process/cortana-analytics-technical-guide-predictive-maintenance>, 15 Marzo 2017
- [96] Safa Meraghni, Labib Sadek Terrissa, Soheyb Ayad, Nouredine Zerhouni, Christophe Varrier, *Post-prognostics decision in cyber-physical systems*, 2018 International Conference on Advanced Systems and Electric Technologies (IC_ASET), DOI: 10.1109/ASET.2018.8379859, 22-25 Marzo 2018, <https://ieeexplore.ieee.org/document/8379859>

- [97] Behrad Bagheri, Shanhu Yang, Hung-An Kao, Jay Lee, *Cyber-physical Systems Architecture for Self-Aware Machines in Industry 4.0 Environment*, IFAC (International Federation of Automatic Control) Hosting by Elsevier Ltd, DOI: 10.1016/j.ifacol.2015.06.318, 31 Agosto 2015, <https://www.sciencedirect.com/science/article/pii/S2405896315005571>
- [98] Bo Sun, Shengkui Zeng, Rui Kang, Michael Pecht, *Benefits Analysis of Prognostics in Systems*, Prognostics & System Health Management Conference (PHM2010 Macau), DOI: 978-1-4244-4758-9/10/\$26.00, 2010, <https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=5413503&tag=1>
- [99] R. Keith Mobley, *An introduction to predictive maintenance*, Butterworth-Heinemann, 2 edition, ISBN: 0-7506-7531-4
- [100] He Hou-bo, Zhao Jian-min, Xu Chang-an, *Cost-benefit Model for PHM*, Procedia Environmental Sciences, Volume 10, Parte A, Pagine 759-764, 2011, <https://www.sciencedirect.com/science/article/pii/S1878029611003185>
- [101] Jason Brownlee, *The Best Machine Learning Algorithm*, <https://machinelearningmastery.com/the-best-machine-learning-algorithm/>, 18 Giugno 2014
- [102] Wikipedia, l'enciclopedia libera, *No free lunch in search and optimization*, https://en.wikipedia.org/wiki/No_free_lunch_in_search_and_optimization
- [103] Vepa Atamuradov, Kamal Medjaher, Pierre Dersin, Benjamin Lamoureux, Nouredine Zerhouni, *Prognostics and Health Management for Maintenance Practitioners - Review, Implementation and Tools Evaluation*, International Journal of Prognostics and Health Management, ISSN 2153-2648, 2017 060, 11 Dicembre 2017, <https://www.phmsociety.org/node/2246>
- [104] Cesare Fantuzzi, Riccardo Rubini, Marco Cocconcelli, Cristian Secchi, Michele Cotogno, Giovanni Prata, *Prognostica e Manutenzione Preventiva*, Articolo pubblicato su Automazione Integrata, Giugno 2012, <https://www.crit-research.it/it/tecnologia/prognostica-e-manutenzione-preventiva/>
- [105] Sreerupa Das, Richard Hall, Stefan Herzog, Gregory Harrison, Michael Bodkin, *Essential Steps in Prognostic Health Management*, 2011 IEEE Conference on Prognostics and Health Management, DOI: 10.1109/ICPHM.2011.6024332, 20-23 Giugno 2011, <https://ieeexplore.ieee.org/document/6024332/>
- [106] Hatem M. Elattar, Hamdy K. Elminir, A. M. Riad, *Prognostics: a literature review*, Complex & Intelligent Systems, Giugno 2016, Volume 2, Issue 2, pagine 125-154, <https://link.springer.com/article/10.1007/s40747-016-0019-3>
- [107] Antonio J Guillén López, Adolfo Crespo Márquez, Juan Fco. Gómez Fernández, Alejandro Guerrero Bolaños, *Towards the Industrial Application of PHM: Challenges and Methodological Approach*, Second European Conference of the Prognostics and Health Management Society 2014, 13 Giugno 2014, <https://www.phmsociety.org/node/1295>
- [108] Bo Sun, Shengkui Zeng, Rui Kang, Michael G. Pecht, *Benefits and Challenges of System Prognostics*, IEEE Transactions on reliability, Volume 61, Numero 2, Giugno 2012, <https://ieeexplore.ieee.org/document/6191342/authors#authors>
- [109] Shankar Sankararaman, *Significance, interpretation, and quantification of uncertainty in prognostics and remaining useful life prediction*, Mechanical Systems and Signal Processing, Volumi 52-53, Febbraio 2015, pagine 228-247, <https://www.sciencedirect.com/science/article/pii/S0888327014002052>
- [110] Shankar Sankararaman, Kai Goebel, *A Novel Computational Methodology for Uncertainty Quantification in Prognostics Using The Most Probable Point Concept*, Annual Conference of the Prognostics and Health Management Society 2013, <http://www.phmsociety.org/node/1072/>
- [111] Shankar Sankararaman, Kai Goebel, *Uncertainty in Prognostics and Systems Health Management*, International Journal of Prognostics and Health Management, ISSN2153-2648, 2015 010, <https://www.phmsociety.org/node/1637>
- [112] Gartner, *Gartner's 2016 Hype Cycle for Emerging Technologies Identifies Three Key Trends That Organizations Must Track to Gain Competitive Advantage* <https://www.gartner.com/newsroom/id/3412017>, Agosto 2016

- [113] Gartner, Kasey Panetta, *Top Trends in the Gartner Hype Cycle for Emerging Technologies, 2017* <https://www.gartner.com/smarterwithgartner/top-trends-in-the-gartner-hype-cycle-for-emerging-technologies-2017/>, 15 Agosto 2017
- [114] Gartner, Kasey Panetta, *5 Trends Emerge in the Gartner Hype Cycle for Emerging Technologies, 2018* <https://www.gartner.com/smarterwithgartner/5-trends-emerge-in-gartner-hype-cycle-for-emerging-technologies-2018/>, 16 Agosto 2018
- [115] Mark Haarman, Michel Mulders, Costas Vassiliadis, *Predictive Maintenance 4.0 - Predict the unpredictable*, <https://www.pwc.nl/en/publicaties/predictive-maintenance-40-predict-the-unpredictable.html>, Giugno 2017
- [116] IoT Analytics, *Predictive Maintenance: Market Report 2017-22*, <https://iot-analytics.com/product/predictive-maintenance-market-report-2017-2022/>, Marzo 2017
- [117] Consultancy.uk, *Market for predictive maintenance to reach \$11 billion by 2022*, <https://www.consultancy.uk/news/18408/market-for-predictive-maintenance-to-reach-11-billion-by-2022>, 23 Agosto 2018
- [118] IBM, *IBM Predictive Maintenance*, <https://www.ibm.com/us-en/marketplace/prescriptive-maintenance-on-cloud>
- [119] KONE Corporation, *KONE revolutionizes elevator maintenance with new customizable KONE Care(TM) service offering and 24/7 Connected Services*, <https://globenewswire.com/news-release/2017/02/08/914900/0/en/KONE-revolutionizes-elevator-maintenance-with-new-customizable-KONE-Care-TM-service-offering-and-24-7-Connected-Services.html>, 8 Febbraio 2017
- [120] IoT Analytics, Digital Analytics Association Germany, Knud Lasse Lueth, *Industrial Analytics Report 2016/17*, <https://iot-analytics.com/product/industrial-analytics-report-201617/>, Dicembre 2016
- [121] Wikipedia, The Free Encyclopedia, *Prognostics*, https://en.wikipedia.org/wiki/Prognostics#Industrial_applications_and_case_studies
- [122] Fidan Boylu Uz, olprod, iamqinmu, OpenLocalizationService, *Guida di Azure AI per soluzioni di manutenzione predittiva*, <https://docs.microsoft.com/it-it/azure/machine-learning/team-data-science-process/cortana-analytics-playbook-predictive-maintenance#business-case-for-predictive-maintenance>, 11 Maggio 2018
- [123] Daeil Kwon, Melinda R. Hodkiewicz, Jiajie Fan, Tadahiro Shibutani, Michael G. Pecht, *IoT-Based Prognostics and Systems Health Management for Industrial Application*, IEEE Access (Volume: 4), DOI: 10.1109/ACCESS.2016.2587754, 2016, <https://ieeexplore.ieee.org/abstract/document/7520653/>
- [124] Microsoft Azure, Ramkumar Krishnan, Gary Ericson, C.J. Gronlund, William Anton Rohm, Craig Casey, Andrew Ivanov, Abdelkrim from Brussels, *Azure AI guide for predictive maintenance solutions*, <https://docs.microsoft.com/en-us/azure/machine-learning/team-data-science-process/cortana-analytics-playbook-predictive-maintenance>
- [125] BigData Republic, *Machine learning for predictive maintenance: where to start?* <https://medium.com/bigdatarepublic/machine-learning-for-predictive-maintenance-where-to-start-5f3b7586acfb>, 29 Agosto 2017
- [126] Machine Learning Mastery, Jason Brownlee, *Difference Between Classification and Regression in Machine Learning*, <https://machinelearningmastery.com/classification-versus-regression-in-machine-learning/>, 11 Dicembre 2017
- [127] Medium, David Fumo, *Classification Versus Regression - Intro To Machine Learning #5* <https://medium.com/simple-ai/classification-versus-regression-intro-to-machine-learning-5-5566efd4cb83>, 13 Febbraio 2017
- [128] Oracle + DataScience.com, Nikki Castle, *Regression vs. Classification Algorithms*, <https://www.datascience.com/blog/regression-and-classification-machine-learning-algorithms>, 8 Marzo 2018
- [129] Oracle + DataScience.com, Nikki Castle, *Supervised vs. Unsupervised Machine Learning*, <https://www.datascience.com/blog/supervised-and-unsupervised-machine-learning-algorithms>, 13 Luglio 2017
- [130] Sumitomo (SHI) Demag, *IntElect, Maximum Efficiency - Highest Precision.*, <https://www.sumitomo-shi-demag.eu/products/intelect.html>

- [131] Sumitomo (SHI) Demag, *IntElect - high precise, brochure*, https://www.sumitomo-shi-demag.eu/fileadmin/user_upload/downloads/download_center/Blaetterkataloge/Intelect_englisch/html5.html#/1
- [132] Sumitomo (SHI) Demag, *Documentazione, IntElect 160/520-680 Smart*
- [133] Wikipedia, L'enciclopedia libera, *Stampaggio a iniezione*, https://it.wikipedia.org/wiki/Stampaggio_a_iniezione
- [134] Giulia Faggionato, *Stampaggio ad iniezione con sistema idraulico ed elettrico. Analisi delle diverse caratteristiche*, Tesi di laurea in Ingegneria Gestionale, Università degli studi di Padova, Dipartimento di tecnica e gestione dei sistemi industriali, relatore prof. Guido Berti, anno accademico 2013/2014, http://tesi.cab.unipd.it/46614/1/Tesi_triennale_IG_Faggionato_Giulia.pdf
- [135] Tecnologie e materiali per il Design, Marco Bocellato, *Stampaggio ad iniezione, esempio di una macchina per stampaggio ad iniezione*, <https://tecmatied.wordpress.com/2013/02/08/stampaggio-ad-iniezione/>
- [136] Gianni Spagnoli, *Ottimizzazione del processo di stampaggio ad iniezione per compositi termoplastici caricati con fibre vegetali e minerali: effetto delle fibre sul comportamento meccanico* Tesi di laurea in Ingegneria dei Materiali, Università di Roma, La Sapienza, Facoltà di Ingegneria Civile e Industriale, relatore prof. ing. Marco Valente, anno accademico 2013/2014, <https://www.elettroplast srl.it/allegati/documenti/Tesi%20di%20Gianni%20Spagnoli%20ufficiale.pdf>
- [137] Machine learning mastery, Jason Brownlee, *8 Tactics to Combat Imbalanced Classes in Your Machine Learning Dataset*, <https://machinelearningmastery.com/tactics-to-combat-imbalanced-classes-in-your-machine-learning-dataset/>, 19 Agosto 2015
- [138] Wikipedia, The Free Encyclopedia *Accuracy paradox* https://en.wikipedia.org/wiki/Accuracy_paradox
- [139] Mohamed Bekkar, Hassiba Khelouane Djemaa, Taklit Akrouf Alitouche, *Evaluation Measures for Models Assessment over Imbalanced Data Sets*, Journal of Information Engineering and Applications, ISSN 2224-5782 (print) ISSN 2225-0506 (online) Vol.3, No.10, 2013, <https://www.iiste.org/Journals/index.php/JIEA/article/view/7633>
- [140] Machine learning mastery, Jason Brownlee, *Classification Accuracy is Not Enough: More Performance Measures You Can Use* <https://machinelearningmastery.com/classification-accuracy-is-not-enough-more-performance-measures-you-can-use/>, 21 Marzo 2014
- [141] Wikipedia, The Free Encyclopedia, *Cohen's kappa*, https://en.wikipedia.org/wiki/Cohen%27s_kappa
- [142] KDnuggets, Ye Wu, Rick Radewagen, *7 Techniques to Handle Imbalanced Data* <https://www.kdnuggets.com/2017/06/7-techniques-handle-imbalanced-data.html>
- [143] EliteDataScience, *How to Handle Imbalanced Classes in Machine Learning*, <https://elitedatascience.com/imbalanced-classes>, 5 Luglio 2017
- [144] Quora, Ian Vala, *In classification, how do you handle an unbalanced training set?*, <https://www.quora.com/In-classification-how-do-you-handle-an-unbalanced-training-set>, 4 Ottobre 2016
- [145] Machine Learning Mastery, Jason Brownlee, *How to Calculate the Principal Component Analysis from Scratch in Python*, <https://machinelearningmastery.com/calculate-principal-component-analysis-scratch-python/>, 2 Marzo 2018
- [146] DeZyre, *Principal Component Analysis Tutorial*, <https://www.dezyre.com/data-science-in-python-tutorial/principal-component-analysis-tutorial>
- [147] E. Martinelli, C. Di Natale, *L'Analisi delle Componenti Principali (PCA)*, http://www.uniroma2.it/didattica/Sensori_ed_applic_2/deposito/lez5.pdf, Università degli Studi di Roma "Tor Vergata", Introduzione al trattamento Statistico dei Dati
- [148] Wikipedia, The Free Encyclopedia, *Oversampling and undersampling in data analysis*, https://en.wikipedia.org/wiki/Oversampling_and_undersampling_in_data_analysis

- [149] G. Lemaitre, F. Nogueira, D. Oliveira, C. Aridas, *imbalanced-learn - User Guide* http://contrib.scikit-learn.org/imbalanced-learn/stable/user_guide.html, 2016
- [150] Cross Validated, Franck Dernoncourt, *Opinions about Oversampling in general, and the SMOTE algorithm in particular*, <https://stats.stackexchange.com/questions/234016/opinions-about-oversampling-in-general-and-the-smote-algorithm-in-particular>, 8 Settembre 2016
- [151] Kaggle, Rafael Alencar, *Resampling strategies for imbalanced datasets*, <https://www.kaggle.com/rafjaa/resampling-strategies-for-imbalanced-datasets>
- [152] GitHub, Guillaume Lemaitre, *imbalanced-learn*, <https://github.com/scikit-learn-contrib/imbalanced-learn>
- [153] Lindsay Kleeman, *Understanding and Applying Kalman Filtering*, http://biorobotics.ri.cmu.edu/papers/sbp_papers/integrated3/kleeman_kalman_basics.pdf, Department of Electrical and Computer Systems Engineering, Monash University, Clayton
- [154] Ronald W. Schafer, *What Is a Savitzky-Golay Filter?*, IEEE Signal Processing Magazine, vol. 28, no. 4, pp. 111-117, Luglio 2011, DOI: 10.1109/MSP.2011.941097, <https://ieeexplore.ieee.org.ezproxy.biblio.polito.it/document/5888646/>
- [155] Wikipedia, The Free Encyclopedia, *Savitzky-Golay filter*, [https://en.wikipedia.org/wiki/Savitzky?Golay_filter](https://en.wikipedia.org/wiki/Savitzky%20Golay_filter)
- [156] statistics4u, *Savitzky-Golay Filter*, http://www.statistics4u.com/fundstat_eng/cc_filter_savgolay.html
- [157] John Scott Bucknam, *Data analysis and processing techniques for remaining useful life estimations (2017)*, Theses and Dissertations, 2430, <https://rdw.rowan.edu/etd/2430/>
- [158] Wikipedia, The Free Encyclopedia, *Kalman filter*, https://en.wikipedia.org/wiki/Kalman_filter
- [159] Tony Lacey, *Tutorial: The Kalman Filter*, <http://web.mit.edu/kirtley/kirtley/binlustuff/literature/control/Kalman%20filter.pdf>
- [160] Alvira Swalin, *Choosing the Right Metric for Evaluating Machine Learning Models???* Part 2, <https://medium.com/usf-msds/choosing-the-right-metric-for-evaluating-machine-learning-models-part-2-86d5649a5428>, 2 Maggio 2018
- [161] Rushdi Shams, *Micro- and Macro-average of Precision, Recall and F-Score*, <http://rushdishams.blogspot.com/2011/08/micro-and-macro-average-of-precision.html>, 21 Agosto 2011
- [162] Roope Astala, Gary Ericson, Josée Martens, Jeannine Takaki, *One-vs-All Multi-class*, <https://docs.microsoft.com/en-us/azure/machine-learning/studio-module-reference/one-vs-all-multiclass>, 24 Gennaio 2018
- [163] Alvira Swalin, *Choosing the Right Metric for Evaluating Machine Learning Models???* Part 1, <https://medium.com/usf-msds/choosing-the-right-metric-for-machine-learning-models-part-1-a99d7d7414e4>, 7 Aprile 2018
- [164] Jason Brownlee, *Metrics To Evaluate Machine Learning Algorithms in Python*, <https://machinelearningmastery.com/metrics-evaluate-machine-learning-algorithms-python/>, 25 Maggio 2016
- [165] Raffaele Stifani, *IBM Bluemix The Cloud Platform for Creating and Delivering Applications*, IBM RedBooks, ISBN: 9780738454429, 01 settembre 2015 <http://www.redbooks.ibm.com/abstracts/redp5242.html?Open>
- [166] Urs Hunkeler, Hong Linh Truong, Andy Stanford-Clark, *MQTT-S - A publish/subscribe protocol for Wireless Sensor Networks*, 2008 3rd International Conference on Communication Systems Software and Middleware and Workshops (COMSWARE '08), Bangalore, 2008, pp. 791-798, doi: 10.1109/COMSWA.2008.4554519, <https://ieeexplore.ieee.org/abstract/document/4554519/>
- [167] Wikipedia, The Free Encyclopedia, *ESP32*, <https://en.wikipedia.org/wiki/ESP32>
- [168] Espressif Systems, *ESP32 Series Datasheet*, https://www.espressif.com/sites/default/files/documentation/esp32_datasheet_en.pdf
- [169] Waveshare Wiki, *DHT22 Temperature-Humidity Sensor*, https://www.waveshare.com/wiki/DHT22_Temperature-Humidity_Sensor

- [170] Thomas Liu, *Digital-output relative humidity & temperature sensor/module DHT22 (DHT22 also named as AM2302)*, <https://www.sparkfun.com/datasheets/Sensors/Temperature/DHT22.pdf>
- [171] Glynn Bird, *Cloudant Node.js Client*, <https://github.com/cloudant/nodejs-cloudant>
- [172] Jeffrey Dare, *Node.js Client Library*, <https://github.com/ibm-watson-iot/iot-nodejs>
- [173] Nitin Kumar Mishra, *SB Admin Angular5 BS4*, <https://github.com/start-angular/SB-Admin-BS4-Angular-6>
- [174] Andres Galante, Bardi Harborow, Connor Sears, Chris Rebert, fat, Heinrich Fenkart, Justin Dorfman, Johann-S, Julian Thilo, Mark Otto, Patrick H. Lauke, Thomas McDonald, XhmikosR, Zac Echola, *Bootstrap*, <https://getbootstrap.com/>
- [175] Dmitriy Shekhovtsov, *ng2-charts*, <https://github.com/valor-software/ng2-charts>
- [176] Developer Express, *DevExtreme - HTML5 JavaScript Component Suite for Responsive Web Development*, <https://js.devexpress.com/>