

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



TESI DI LAUREA

Sviluppo di un modulo di supervisione data-driven per sistemi di protezione catodica nelle reti di distribuzione gas

Development of a data-driven supervision module for cathodic protection systems in gas distribution networks

Relatore

Prof. Alessandro Freddi

Correlatore

Emanuele Leoni

Candidata

Beatrice Moliterno

ANNO ACCADEMICO 2023-2024

*Il passato ed il futuro
non sono realtà ma solo effimere illusioni.
Devo liberarmi del tempo
e vivere il presente giacché non esiste altro tempo
che questo meraviglioso istante.*

Alda Merini, "Il mio passato"

Introduzione	1
1 La protezione catodica	3
1.1 La corrosione dei metalli	3
1.1.1 Gli aspetti elettrochimici	4
1.1.2 Gli aspetti termodinamici	5
1.1.3 Gli aspetti cinetici	6
1.2 La protezione catodica - generalità	8
1.2.1 Le condizioni di protezione	8
1.2.2 I sistemi di protezione catodica	9
1.2.3 Il monitoraggio	10
1.3 Le soluzioni proposte dall'azienda Automa s.r.l.	11
1.3.1 Il dispositivo <i>G4C-PRO</i>	11
1.3.2 Gli alimentatori di corrente per la protezione catodica	12
1.3.3 La gestione di un sistema di protezione catodica con alimentatore di corrente a potenziale costante	12
2 Il dataset ed i sistemi scelti	15
2.1 L'origine dei dati	15
2.1.1 Il database	15
2.1.2 Il dataset	16
2.2 La scelta delle ddp.dc da predire	17
2.2.1 Il sistema <i>MV11</i>	19
2.2.2 Il sistema <i>S04ML</i>	21
3 Le strategie di predizione adottate	26
3.1 Il preprocessing dei dati	26
3.2 La <i>Rolling window</i>	26
3.2.1 Il modello <i>Linear regression (OLS)</i>	27
3.2.2 Il modello <i>SVR (ϵ-insensitive loss)</i>	29
3.2.3 Il modello <i>Voting regressor</i>	31
3.2.4 La gestione dei dati e dei valori <i>NaN</i>	31
3.3 La rete neurale <i>LSTM</i>	32
3.3.1 Il funzionamento della rete <i>LSTM</i>	32
3.3.2 La gestione dei dati e dei valori <i>NaN</i>	34

3.4	La valutazione dei modelli	34
4	L'implementazione ed i risultati della <i>rolling window</i>	38
4.1	L'implementazione della <i>rolling window</i>	38
4.2	I risultati della predizione	39
4.2.1	La predizione della ddp.dc - ID 17271	39
4.2.2	La predizione della ddp.dc - ID 8984	47
4.3	Confronto dei risultati delle ddp.dc	53
5	L'implementazione ed i risultati della <i>LSTM</i>	55
5.1	L'implementazione delle reti <i>LSTM</i>	55
5.2	I risultati della predizione	56
5.2.1	La predizione della ddp.dc - ID 17271	57
5.2.2	La predizione della ddp.dc - ID 8984	62
5.3	Confronto dei risultati delle ddp.dc	67
6	Conclusioni	69
	Bibliografia	72

Elenco delle figure

1.1	Il ciclo di trasformazione dei materiali metallici [Pedferri e Lazzari, 2006] . . .	3
1.2	Le fasi del processo di corrosione [Pedferri, 2010]	4
1.3	Diagramma di Evans - riduzione di ossigeno [Pedferri, 2010]	7
1.4	Diagramma di Evans - sviluppo di idrogeno [Pedferri, 2010]	7
1.5	Le condizioni elettrochimiche in presenza di una corrente catodica esterna [Pedferri, 2010]	9
1.6	I potenziali di protezione adottati nei terreni [Lazzarri <i>et al.</i> , 2006]	9
1.7	a) Il sistema di protezione catodica con anodo galvanico; b) Il sistema di protezione catodica a corrente impressa [Lazzarri <i>et al.</i> , 2006]	10
1.8	La misurazione del potenziale tramite elettrodo di riferimento [Lazzarri <i>et al.</i> , 2006]	11
1.9	Il dispositivo G4C-Pro ed il suo collocamento	11
1.10	Una generica cartografia di WebProcat raffigurante un sistema di protezione catodica dotato di un solo alimentatore di corrente impressa, 8 punti di misura telecomandati e 13 punti di misura gestiti tramite operatore	13
1.11	Il funzionamento di un sistema di protezione catodica a corrente impressa con alimentatore a potenziale costante	14
2.1	Logo di CENTRIA s.r.l.	15
2.2	Le prime 5 righe del dataframe contenente le informazioni utili per le analisi	17
2.3	Lista degli eventi prevedibili per ciascun punto di misura del sistema con ID S03ML	17
2.4	Le prime 30 righe del foglio excel che riassume le principali caratteristiche di interesse di ciascun sistema, utili per la scelta delle ddp.dc da prevedere . . .	18
2.5	La cartografia corrispondente al sistema con ID MV11	19
2.6	Le caratteristiche riassuntive del sistema con ID MV11	19
2.7	L'andamento nel tempo della corrente, della ddp.dc in corrispondenza dell'alimentatore di corrente e della ddp.dc in corrispondenza del punto con ID 17271, nonché il punto di misura da prevedere	20
2.8	Boxplot relativo alla distribuzione di corrente dell'alimentatore del sistema con ID MV11.	21
2.9	Boxplot relativi alle distribuzioni di differenza di potenziale in corrispondenza dell'alimentatore e del punto di misura telecomandato del sistema con ID MV11.	21
2.10	La cartografia corrispondente al sistema con ID S04ML	22
2.11	Le caratteristiche riassuntive del sistema con ID S04ML	23

2.12	L'andamento nel tempo della corrente, della ddp.dc in corrispondenza dell'alimentatore di corrente e delle ddp.dc in corrispondenza dei punti di misura telecontrollati, in ordine di distanza dall'alimentatore di corrente; il punto di misura con ID 8984 è il punto di misura da prevedere	24
2.13	Boxplot relativo alla distribuzione di corrente dell'alimentatore del sistema con ID S04ML.	25
2.14	Boxplot relativi alle distribuzioni di differenza di potenziale in corrispondenza dell'alimentatore e dei punti di misura telecontrollati del sistema con ID S04ML.	25
2.15	L'andamento nel tempo della ddp.dc con ID 8984, il punto di misura da prevedere, per visualizzare i valori mancanti	25
3.1	Uno schema semplificato del funzionamento della <i>rolling window</i> utilizzata per la previsione di serie temporali	27
3.2	Un esempio semplificato della <i>regressione lineare</i> con metodo OLS	28
3.3	(a) L'impostazione della loss per una <i>linear SVR</i> - (b) la <i>ϵ-insensitive loss function</i> [Bi <i>et al.</i> , 2011]	30
3.4	L'architettura della rete LSTM [Islam <i>et al.</i> , 2019]	32
3.5	Schema illustrativo del passaggio dai primi $K+N$ elementi del dataset iniziale nella prima sequenza del dataset X e nella prima sequenza del dataset Y	35
3.6	Schema illustrativo dello shift della finestra temporale di lunghezza fissa K sul dataset iniziale ed il passaggio dai $K+N$ elementi del suddetto dataset nella seconda sequenza del dataset X e nella seconda sequenza del dataset Y	35
3.7	Schema semplificato per la valutazione delle metriche	36
4.1	L'istogramma che illustra il MAE calcolato su ciascun giorno predetto per la ddp.dc con ID 17271, rispetto a ciascun modello utilizzato, su di una finestra fissa scorrevole di 45 campioni	40
4.2	L'istogramma che illustra l' R^2 calcolato su ciascun giorno predetto per la ddp.dc con ID 17271, rispetto a ciascun modello utilizzato, su di una finestra fissa scorrevole di 45 campioni	40
4.3	L'istogramma che illustra il MAE calcolato su ciascun giorno predetto per la ddp.dc con ID 17271, rispetto a ciascun modello utilizzato, su di una finestra fissa scorrevole di 30 campioni	41
4.4	L'istogramma che illustra l' R^2 calcolato su ciascun giorno predetto per la ddp.dc con ID 17271, rispetto a ciascun modello utilizzato, su di una finestra fissa scorrevole di 30 campioni	41
4.5	L'istogramma che illustra il MAE calcolato su ciascun giorno predetto per la ddp.dc con ID 17271, rispetto a ciascun modello utilizzato, su di una finestra fissa scorrevole di 15 campioni	42
4.6	L'istogramma che illustra l' R^2 calcolato su ciascun giorno predetto per la ddp.dc con ID 17271, rispetto a ciascun modello utilizzato, su di una finestra fissa scorrevole di 15 campioni	42
4.7	Il boxplot che illustra l'errore di previsione commesso su ciascun giorno predetto per la ddp.dc con ID 17271, rispetto a ciascun modello utilizzato, su di una finestra fissa scorrevole di 45 campioni	43
4.8	Il boxplot che illustra l'errore di previsione commesso su ciascun giorno predetto per la ddp.dc con ID 17271, rispetto a ciascun modello utilizzato, su di una finestra fissa scorrevole di 30 campioni	43
4.9	Il boxplot che illustra l'errore di previsione commesso su ciascun giorno predetto per la ddp.dc con ID 17271, rispetto a ciascun modello utilizzato, su di una finestra fissa scorrevole di 15 campioni	44

4.10	La predizione delle ddp.dc con <i>ID 17271</i> effettuata il giorno 2020-03-04 in cui si evidenzia un comportamento simile tra i modelli <i>SVR</i> e <i>Voting Regressor</i> . . .	45
4.11	La predizione delle ddp.dc con <i>ID 17271</i> effettuata il giorno 2020-03-05 in cui si evidenzia un comportamento più reattivo da parte dell' <i>SVR</i> , ma entrambi buoni predittori dei campioni fuori soglia	45
4.12	La predizione delle ddp.dc con <i>ID 17271</i> effettuata il giorno 2020-03-10 in cui si evidenzia come il <i>Voting Regressor</i> riesce a catturare meglio la tendenza dei dati rispetto all' <i>SVR</i> , grazie alla sua componente lineare	46
4.13	La predizione delle ddp.dc con <i>ID 17271</i> effettuata il giorno 2020-05-06 in cui si evidenzia come l' <i>SVR</i> riesca ad effettuare una previsione migliore rispetto al <i>Voting Regressor</i>	46
4.14	Le metriche e la confusion matrix relativa al quinto giorno di previsione della ddp.dc con <i>ID 17271</i> relativo al <i>Voting Regressor</i> considerando una finestra di 15 giorni	46
4.15	L'istogramma che illustra il <i>MAE</i> calcolato su ciascun giorno predetto per la ddp.dc con <i>ID 8984</i> , rispetto a ciascun modello utilizzato, su di una finestra fissa scorrevole di 45 campioni.	47
4.16	L'istogramma che illustra l' R^2 calcolato su ciascun giorno predetto per la ddp.dc con <i>ID 8984</i> , rispetto a ciascun modello utilizzato, su di una finestra fissa scorrevole di 45 campioni.	48
4.17	L'istogramma che illustra il <i>MAE</i> calcolato su ciascun giorno predetto per la ddp.dc con <i>ID 8984</i> , rispetto a ciascun modello utilizzato, su di una finestra fissa scorrevole di 30 campioni.	48
4.18	L'istogramma che illustra l' R^2 calcolato su ciascun giorno predetto per la ddp.dc con <i>ID 8984</i> , rispetto a ciascun modello utilizzato, su di una finestra fissa scorrevole di 30 campioni.	49
4.19	L'istogramma che illustra il <i>MAE</i> calcolato su ciascun giorno predetto per la ddp.dc con <i>ID 8984</i> , rispetto a ciascun modello utilizzato, su di una finestra fissa scorrevole di 15 campioni.	49
4.20	L'istogramma che illustra l' R^2 calcolato su ciascun giorno predetto per la ddp.dc con <i>ID 8984</i> , rispetto a ciascun modello utilizzato, su di una finestra fissa scorrevole di 15 campioni.	50
4.21	Il boxplot che illustra l'errore di previsione commesso su ciascun giorno predetto per la ddp.dc con <i>ID 8984</i> , rispetto a ciascun modello utilizzato, su di una finestra fissa scorrevole di 45 campioni	50
4.22	Il boxplot che illustra l'errore di previsione commesso su ciascun giorno predetto per la ddp.dc con <i>ID 8984</i> , rispetto a ciascun modello utilizzato, su di una finestra fissa scorrevole di 30 campioni	51
4.23	Il boxplot che illustra l'errore di previsione commesso su ciascun giorno predetto per la ddp.dc con <i>ID 8984</i> , rispetto a ciascun modello utilizzato, su di una finestra fissa scorrevole di 15 campioni	51
4.24	La predizione delle ddp.dc con <i>ID 8984</i> effettuata il giorno 2016-09-19 in cui si evidenzia come l' <i>SVR</i> ed il <i>emphVoting Regressor</i> abbiano un andamento molto simile	52
4.25	La predizione delle ddp.dc con <i>ID 8984</i> effettuata il giorno 2026-12-02 in cui si evidenzia come l' <i>SVR</i> riesca ad effettuare una previsione migliore rispetto al <i>Voting Regressor</i>	52
4.26	La predizione delle ddp.dc con <i>ID 8984</i> effettuata il giorno 2026-12-09 in cui si evidenzia come <i>Voting Regressor</i> riesce a catturare maggiormente la tendenza che hanno i dati rispetto all' <i>SVR</i>	53

4.27	Le metriche e la confusion matrix relativa al quinto giorno di previsione della ddp.dc con <i>ID 8984</i> relativo al <i>Voting Regressor</i> considerando una finestra di 15 giorni	53
5.1	L'istogramma che illustra il <i>MAE</i> calcolato su ciascun giorno predetto, addestrandolo la rete per la ddp.dc con <i>ID 17271</i> rispetto al modello univariabile e multivariabile, considerando, sul dataset iniziale, una finestra scorrevole di 15 e 45 giorni	57
5.2	L'istogramma che illustra l' R^2 calcolato su ciascun giorno predetto, addestrandolo la rete per la ddp.dc con <i>ID 17271</i> rispetto al modello univariabile e multivariabile, considerando, sul dataset iniziale, una finestra scorrevole di 15 e 45 giorni	58
5.3	L'andamento dell' <i>MSELoss</i> durante il train ed il test effettuato su 150 epoche – modello multivariabile con finestra di 45 campioni	59
5.4	L'andamento nel tempo del primo giorno di previsione - modello multivariabile e finestra 45 giorni	59
5.5	L'andamento nel tempo del secondo giorno di previsione - modello multivariabile e finestra 45 giorni	60
5.6	L'andamento nel tempo del terzo giorno di previsione - modello multivariabile e finestra 45 giorni	60
5.7	L'andamento nel tempo del quarto giorno di previsione - modello multivariabile e finestra 45 giorni	61
5.8	L'andamento nel tempo del quinto giorno di previsione - modello multivariabile e finestra 45 giorni	61
5.9	Le metriche e la confusion matrix relativa al quinto giorno di previsione per il modello multivariabile con finestra di 45 giorni	62
5.10	L'istogramma che illustra il <i>MAE</i> calcolato su ciascun giorno predetto, addestrandolo la rete per la ddp.dc con <i>ID 8984</i> rispetto al modello univariabile e multivariabile, considerando, sul dataset iniziale, una finestra scorrevole di 15 e 45 giorni	63
5.11	L'istogramma che illustra l' R^2 calcolato su ciascun giorno predetto, addestrandolo la rete per la ddp.dc con <i>ID 8984</i> rispetto al modello univariabile e multivariabile, considerando, sul dataset iniziale, una finestra scorrevole di 15 e 45 giorni	63
5.12	L'andamento dell' <i>MSELoss</i> durante il train ed il test effettuato su 250 epoche – modello multivariabile con finestra di 45 campioni	64
5.13	L'andamento nel tempo del primo giorno di previsione - modello multivariabile e finestra 45 giorni	64
5.14	L'andamento nel tempo del secondo giorno di previsione - modello multivariabile e finestra 45 giorni	65
5.15	L'andamento nel tempo del terzo giorno di previsione - modello multivariabile e finestra 45 giorni	65
5.16	L'andamento nel tempo del quarto giorno di previsione - modello multivariabile e finestra 45 giorni	66
5.17	L'andamento nel tempo del quinto giorno di previsione - modello multivariabile e finestra 45 giorni	66
5.18	Le metriche e la confusion matrix relativa al quinto giorno di previsione per il modello multivariabile con finestra di 45 giorni	67

Elenco delle tabelle

4.1	Valori di <i>MAE</i> calcolati eseguendo il <i>fit</i> del modello <i>Voting Regressor</i> su tutta la lunghezza della serie temporale corrispondente alla <i>ddp.dc</i> con <i>ID 17271</i> e alla <i>ddp.dc</i> con <i>ID 8984</i> , considerando una finestra scorrevole di 15 giorni . . .	54
4.2	I valori di R^2 calcolati eseguendo il <i>fit</i> del modello <i>Voting Regressor</i> su tutta la lunghezza della serie temporale corrispondente alla <i>ddp.dc</i> con <i>ID 17271</i> e alla <i>ddp.dc</i> con <i>ID 8984</i> , considerando una finestra scorrevole di 15 giorni	54
5.1	La struttura e gli iperparametri scelti per ciascun modello di rete neurale <i>LSTM</i> addestrata sui dati della <i>ddp.dc</i> con <i>ID 17271</i> e della <i>ddp.dc</i> con <i>ID 8984</i> . . .	56
5.2	Valori di <i>MAE</i> ottenuti dai modelli migliori monovariati della rete <i>LSTM</i> per la <i>ddp.dc</i> con <i>ID 17271</i> e la <i>ddp.dc</i> con <i>ID 8984</i>	67
5.3	I valori di R^2 ottenuti dai modelli migliori monovariati della rete <i>LSTM</i> per la <i>ddp.dc</i> con <i>ID 17271</i> e la <i>ddp.dc</i> con <i>ID 8984</i>	68
5.4	Valori di <i>MAE</i> ottenuti dai modelli migliori della rete <i>LSTM</i> per la <i>ddp.dc</i> con <i>ID 17271</i> e la <i>ddp.dc</i> con <i>ID 8984</i> , considerando una finestra scorrevole di 45 giorni	68
5.5	I valori di R^2 ottenuti dai modelli migliori della rete <i>LSTM</i> per la <i>ddp.dc</i> con <i>ID 17271</i> e la <i>ddp.dc</i> con <i>ID 8984</i> , considerando una finestra scorrevole di 45 giorni	68

Il concetto di *Industria 4.0* non si limita ad essere un neologismo per identificare la quarta rivoluzione industriale ma un simbolo di cambiamento profondo, tecnologico e visionario. L'integrazione di tecnologie all'avanguardia ha portato a cambiamenti trasformativi in vari settori. La *protezione catodica* è stata da tempo riconosciuta come un metodo efficace per combattere la corrosione nelle strutture metalliche, garantendone la longevità e la sicurezza. Tuttavia, l'integrazione delle tecnologie dell'*Industria 4.0* introduce nuove sfide nella gestione convenzionale dei sistemi di protezione catodica, con l'emergente necessità di abbracciare la digitalizzazione e l'automazione per ottimizzare la gestione della corrosione. Grazie ai nuovi dispositivi interconnessi, l'analisi dei big-data e gli algoritmi di intelligenza artificiale, gli approcci convenzionali di protezione catodica possono essere rivalutati, migliorandone l'efficacia.

La presente tesi si colloca proprio in tale contesto e si propone di eseguire un'*analisi predittiva*, una tecnica ormai fondamentale nel settore industriale, per la quale, tramite l'osservazione di dati passati è possibile ottenere dati futuri, con un certo margine di errore. Le analisi di cui si tratterà, sono svolte per conto dell'azienda *Automa s.r.l.*, la quale si occupa di monitoraggio remoto in ambito Oil e Gas. Come già anticipato, i sistemi di protezione catodica permettono di rallentare il processo elettrochimico legato alla corrosione delle pipeline interrate, abbassando la differenza di potenziale, tra la tubazione ed il terreno, al di sotto di -0.85 V, condizione tale per cui la velocità di corrosione si riduce a valori inferiori a 0.01 mm/anno. In questo elaborato, si farà riferimento ai cosiddetti *sistemi a corrente impressa*, i quali, tramite una corrente di alimentazione esterna, permettono di abbassare il potenziale al di sotto della soglia di protezione. A causa di correnti interferenti, condizioni meteorologiche avverse o guasti all'alimentatore, è di fondamentale importanza monitorare il comportamento del sistema di protezione catodica. L'azienda, appunto, si occupa del controllo remoto di tali sistemi, in particolare, tramite il loro dispositivo, chiamato *G4C-PRO*, installato in corrispondenza di ciascun punto di misura di ciascun sistema di interesse, ricevono report giornalieri comprendenti tutte le principali informazioni legate allo stato del sistema. In particolare, in questi report sono riportate le medie giornaliere di *ddp.dc*, le differenze di potenziale delle pipeline in un determinato istante temporale e la corrispondente *I*, la corrente di alimentazione per la protezione costante della rete metallica. I dati a cui si farà riferimento nel presente elaborato appartengono all'azienda *Centria s.r.l.*, una società che gestisce il vettoriamento di gas naturale, nonchè cliente di *Automa s.r.l.*, responsabile, dunque, della salvaguardia delle sue reti di distribuzione, grazie al monitoraggio remoto dei sistemi di protezione. L'alimentatore ha una modalità di funzionamento automatica, ma la sua regolazione è tale per cui, quando si rilevano una serie di campioni fuori soglia in corrispondenza di

uno o più punti di misura, si valuta a posteriori un'azione *correttiva* da compiere. L'obiettivo è, dunque, quello di trasformare l'azione *correttiva* in azione *predittiva*, mantenendo la stessa tecnologia di regolazione di corrente; dunque, sfruttando i dati storici raccolti dall'azienda, si vogliono andare a prevedere le differenze di potenziale dei punti di misura dislocati lungo la pipeline.

Nella fase preliminare di questo elaborato si spiegherà dettagliatamente il processo di corrosione, le tecniche generali per poterle gestire e le soluzioni proposte dall'azienda *Automa s.r.l.*. Successivamente, si analizzerà il dataset in questione e verranno prese in considerazione due ddp.dc da prevedere, che registrano molti campioni consecutivi fuori soglia. Per valutare la migliore strategia di previsione, saranno implementate due tecniche per poi confrontarle: la *rolling window* e la rete neurale *LSTM*. La prima si baserà sull'individuazione di un modello locale, basando la sua previsione su dati recenti, la seconda si baserà sull'individuazione di un modello generale, basando la sua previsione su dati storici. Infine, si otterranno statistiche basate sull'errore di previsione per valutare le performance di ciascun modello ottenuto e saranno confrontate per valutare la migliore strategia da adottare.

La presente tesi è strutturata come di seguito specificato:

- Nel *capitolo 1* sarà descritto il processo di corrosione dei metalli, con particolare attenzione agli aspetti elettrochimici, termodinamici e cinetici legati a tale fenomeno. Saranno analizzate le condizioni affinché ci sia protezione catodica, i relativi sistemi ed il suo monitoraggio. Infine, saranno descritte le soluzioni proposte da *Automa s.r.l.* per la supervisione delle pipeline interrate: il dispositivo *G4C-Pro* e gli alimentatori di corrente a potenziale costante.
- Nel *capitolo 2* saranno analizzati i dati utilizzati per l'attività di previsione e sarà illustrata la modalità di ricerca delle due ddp.dc più interessanti da predire. Infine, saranno descritti i sistemi relativi alle due ddp.dc scelte: la ddp.dc con *ID 17271* e della ddp.dc con *ID 8984*, analizzando gli andamenti temporali e le distribuzioni della corrente e delle differenze di potenziale di tutti i suoi punti di misura.
- Nel *capitolo 3* sarà illustrato il preprocessing dei dati effettuato per la gestione dei valori *NaN* e le tecniche associate alla *rolling window* e alla rete neurale *LSTM*. Si approfondiranno i modelli di *Linear Regression*, *SVR* e *Voting regressor*, utilizzati per l'implementazione della *rolling window* ed il funzionamento della rete neurale. Per entrambe le metodologie sarà spiegato dettagliatamente la relativa gestione dei dati e dei valori *NaN*.
- Nel *capitolo 4* saranno descritti e confrontati i risultati ottenuti dalla previsione della ddp.dc con *ID 17271* e della ddp.dc con *ID 8984*. In particolare, saranno illustrate le metriche di valutazione delle performance su diversi modelli ottenuti considerando finestre temporali di lunghezze diverse e diverse funzioni di regressione: *Linear Regression*, *SVR* e *Voting regressor*.
- Nel *capitolo 5* saranno descritti e confrontati i risultati ottenuti dalla previsione della ddp.dc con *ID 17271* e della ddp.dc con *ID 8984*. In particolare, saranno illustrate le metriche di valutazione delle performance su diversi modelli: univariabile, considerando solo la differenza di potenziale scelta e multivariabile, considerando la differenza di potenziale scelta e la relativa corrente di alimentazione.
- Nel *capitolo 6* saranno riportate le conclusioni del presente elaborato, decretando la metodologia che sarà adottata in futuro dall'azienda *Automa s.r.l.*

La protezione catodica

Nel presente capitolo sono riportati i principali aspetti chimico-fisici legati alla corrosione, il processo di deterioramento che agisce sulle condutture metalliche interrate ed, in particolare, è illustrato come è possibile rallentare tale fenomeno grazie alla protezione catodica. Ci si è concentrati sui sistemi a corrente impressa, adottati dall'azienda Automa s.r.l, nonchè oggetto di studio; infine, sono state illustrate le principali soluzioni proposte dall'azienda per il monitoraggio remoto della protezione catodica ed il telecontrollo degli alimentatori di corrente con regolazione a potenziale costante.

1.1 La corrosione dei metalli

I metalli, fatta esclusione dei cosiddetti "metalli nobili" come l'oro ed il platino, sono esposti al rischio di *corrosione*, il processo di deterioramento o degradazione del materiale a causa dell'interazione con l'ambiente. Tale fenomeno comporta la trasformazione del metallo in un composto più stabile, che corrisponde generalmente alla forma chimica in cui questo si trova in natura come minerale, e dalla quale è stato estratto tramite un processo metallurgico. Per questa ragione, il fenomeno della corrosione è talvolta indicato con il termine *antimetallurgia*. In Figura 1.1 è riportata la schematizzazione del processo appena descritto.

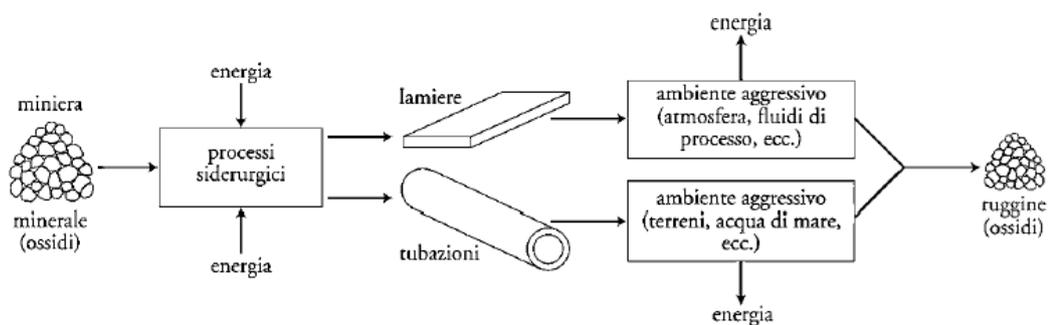


Figura 1.1: Il ciclo di trasformazione dei materiali metallici [Pedefferri e Lazzari, 2006]

Esistono due diverse tipologie di fenomeni corrosivi:

- **corrosione a umido**, legata all'instaurazione di processi elettrochimici favoriti dal contatto tra il materiale ed un ambiente elettrolitico, come acqua o soluzioni contenenti elettroliti, i quali sono capaci di condurre elettricità;

- **corrosione a secco**, legata a processi chimici cinetici e termodinamici favoriti da un ambiente non elettrolitico, solitamente costituito da gas a temperature elevate.

Le reti metalliche interrate, come le reti cittadine di distribuzione del gas, sono soggette al primo tipo di corrosione in quanto il suolo, in presenza di acqua e sali, crea un ambiente simile ad una soluzione elettrolitica; al contrario, un terreno privo di umidità non presenta caratteristiche corrosive. Tuttavia, il terreno stesso introduce variabili specifiche e condizioni che possono influenzare e, quindi, accelerare il processo corrosivo, come la composizione chimica (la concentrazione di cloruri e solfati), la presenza di microrganismi (alcuni batteri riducono i solfati a solfuri), la porosità (i terreni argillosi sono poco permeabili e limitano la penetrazione dell'ossigeno, creando condizioni anerobiche).

1.1.1 Gli aspetti elettrochimici

La corrosione è un fenomeno di natura elettrochimica [Roberge, 2008] e può essere schematizzato attraverso quattro processi riportati in Figura 1.2.

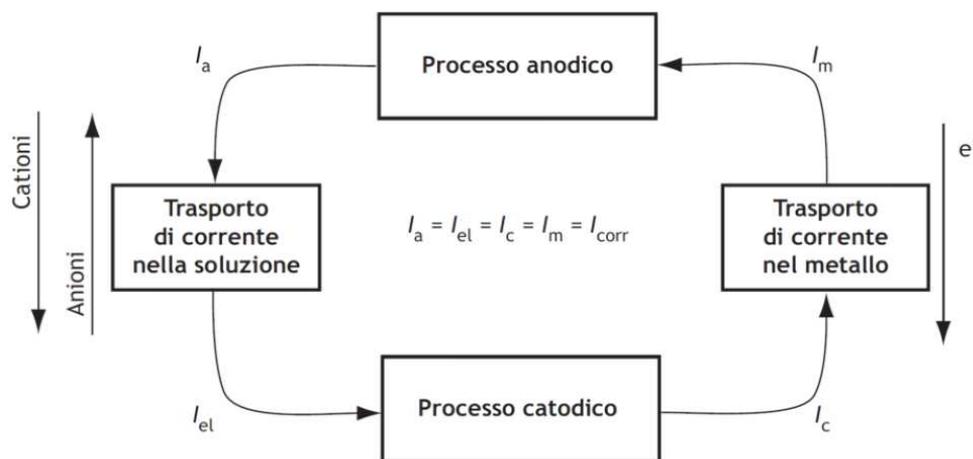


Figura 1.2: Le fasi del processo di corrosione [Pedefferri, 2010]

- **Processo anodico**, dato dalla reazione di *ossidazione* del metallo che comporta la perdita di elettroni e può essere descritta dalla seguente equazione chimica:



dove M rappresenta un qualsiasi metallo, M^{z+} è lo ione metallico e ze^{-} è il numero di elettroni prodotti. Nel caso specifico del ferro la reazione diventa:



- **Processo catodico**, dato dalla reazione di *riduzione* che comporta l'acquisto di elettroni dall'area anodica e può essere descritta dalla seguente equazione chimica:



dove A è la specie ossidata, ze^{-} è il numero di elettroni, A^{a-} è la specie ridotta e a è il numero di ioni rilasciati. Negli ambienti naturali, le possibili due reazioni catodiche sono *riduzione di ossigeno* e *sviluppo di idrogeno*, secondo le seguenti reazioni:

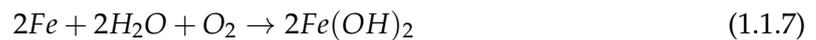


- **Trasporto di elettroni**, per il quale si genera una corrente tra la zona catodica e la zona anodica; in particolare, in Figura 1.2, si evidenzia la I_c , corrente associata alla reazione catodica e la I_m , corrente che scorre all'interno del metallo.
- **Trasporto di ioni**, per il quale si genera una corrente tra la zona anodica e la zona catodica; in particolare, in Figura 1.2, si evidenzia la I_a , corrente associata alla reazione anodica e la I_{el} , corrente che scorre all'interno dell'elettrolita.

La reazione di corrosione complessiva, valida per un metallo qualsiasi in ambiente acquoso, in generale, è data dalla somma delle due semi-reazioni anodica e catodica:



Considerando il ferro in ambiente acquoso come il terreno, la reazione complessiva diventa:



Dalla reazione complessiva si osserva che l'ossigeno molecolare, essendo tra i reagenti, è il principale responsabile della corrosione e questo vale per tutti i reagenti catodici.

Inoltre, è possibile notare che le correnti sono tutte uguali tra loro e pari alla corrente di corrosione I_{corr} , di conseguenza vale che:

$$I_a = I_{el} = I_c = I_m = I_{corr} \quad (1.1.8)$$

1.1.2 Gli aspetti termodinamici

Nei processi di corrosione è di fondamentale importanza considerare il problema da un punto di vista termodinamico. La corrosione è una reazione spontanea che prevede una diminuzione di energia libera del sistema, la quale viene ceduta sotto forma di calore. La spontaneità del processo di corrosione può essere spiegato dal punto di vista termodinamico attraverso la variazione di *energia libera di Gibbs* (G); una reazione è spontanea se vale:

$$\Delta G < 0 \quad (1.1.9)$$

La variazione di energia libera può essere espressa come:

$$\Delta G = -zF\Delta E \quad (1.1.10)$$

dove z è il numero di elettroni coinvolti, F la costante di Faraday e ΔE la differenza tra il potenziale di equilibrio della reazione catodica $E_{eq,c}$ ed il potenziale di equilibrio della reazione anodica $E_{eq,a}$. Tale differenza di potenziale è chiamata *lavoro motore* ed è riportata nella seguente equazione:

$$\Delta E = E_{eq,c} - E_{eq,a} \quad (1.1.11)$$

Dunque, è possibile affermare che la condizione necessaria affinché si abbia corrosione è:

$$\Delta E > 0 \implies E_{eq,c} > E_{eq,a} \quad (1.1.12)$$

I potenziali di equilibrio sono definiti dalla *legge di Nernst* per cui vale la seguente relazione (in forma semplificata):

$$E_{eq} = E^\circ + K \cdot \log C \quad (1.1.13)$$

dove E° è il potenziale standard della reazione, K è la costante dei gas (funzione della temperatura), mentre C dipende dall'attività delle sostanze coinvolte nella reazione elettrochimica. La concentrazione degli ioni è solitamente assunta pari a 10^{-6} mol/L, come proposto da *Pourbaix*.

L'ossidazione di un metallo avviene se il metallo stesso si trova ad un potenziale (E) più nobile di quello di equilibrio, infatti se:

$$E > E_{eq} \quad (1.1.14)$$

il metallo passa in soluzione ed assume un comportamento anodico. Al contrario, se vale che:

$$E < E_{eq} \quad (1.1.15)$$

gli ioni si depositano ed il metallo assume un comportamento catodico.

1.1.3 Gli aspetti cinetici

La termodinamica fornisce informazioni riguardo agli stati di equilibrio, ma non è sufficiente per comprendere appieno il fenomeno, infatti, non fornisce alcuna indicazione sulla velocità con cui si sviluppa la corrosione. Molte reazioni, pure avendo un G negativo, si portano allo stato stabile in un tempo molto lungo e di conseguenza, a livello pratico, non sono preoccupanti dal punto di vista della corrosione. Dunque, dal punto di vista ingegneristico è essenziale determinare la velocità di corrosione, in modo tale da poter mettere in atto le misure di prevenzione più adeguate.

A seguito della circolazione di corrente in presenza di un processo catodico più nobile dell'ossidazione del metallo o in presenza di una corrente esterna di polarizzazione, il potenziale del metallo si discosta da quello di equilibrio. La relazione che lega il potenziale alla densità di corrente scambiata tra metallo-elettrolita ha la seguente forma generale [Pedferri, 2018]:

$$E = E_{eq} \pm f(i) \quad (1.1.16)$$

dove $f(i)$ è una generica funzione, in genere logaritmica, positiva per il processo anodico di ossidazione e negativa per il processo catodico di riduzione; tale funzione esprime la dipendenza del potenziale dalla densità di corrente, detta *sovratensione* (η), per cui vale:

$$\eta = E - E_{eq} \quad (1.1.17)$$

Nel caso del ferro in ambiente neutro e acido, il comportamento elettrochimico in senso anodico è definito dalla *legge di Tafel* descritta dalla seguente relazione:

$$E = E_{eq} + b \cdot \log(i) \quad (1.1.18)$$

dove b è la pendenza della retta di Tafel (tipicamente compresa tra 60 e 100 mV/decade) e i è la densità di corrente anodica.

Quando si innesca un processo di corrosione, si verificano contemporaneamente sia la reazione anodica che la reazione catodica purché $E_{eq,c} > E_{eq,a}$. Gli elettroni prodotti dal processo anodico sono gli stessi che vengono consumati dal processo catodico, perciò

entrambe le reazioni devono avere la stessa velocità. La velocità di reazione è correlata alla velocità di corrente dei due processi e dal momento che deve essere coincidente si verifica che:

$$i_a = i_c = i_{corr} \quad (1.1.19)$$

dove i_a è la densità di corrente del processo anodico, i_c è la densità di corrente del processo catodico e i_{corr} è la densità di corrente scabiata tra i due processi, la quale determina la velocità di corrosione.

I diagrammi che correlano il potenziale del processo anodico e catodico ed il logaritmo della densità di corrente sono denominati *diagrammi di Evans* e sono di fondamentale importanza per lo studio della corrosione. In Figura 1.3 e 1.4 sono riportati i diagrammi di Evans considerando come reazione catodica, rispettivamente, la riduzione dell'ossigeno e lo sviluppo dell'idrogeno.

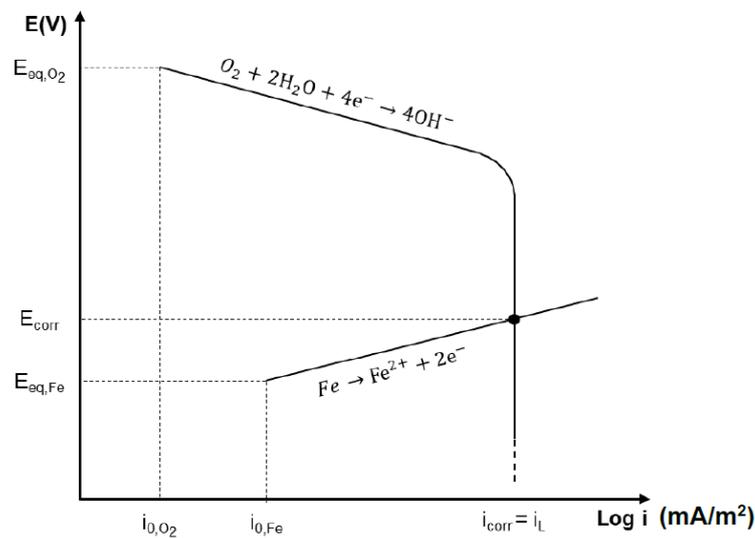


Figura 1.3: Diagramma di Evans - riduzione di ossigeno [Pedefferri, 2010]

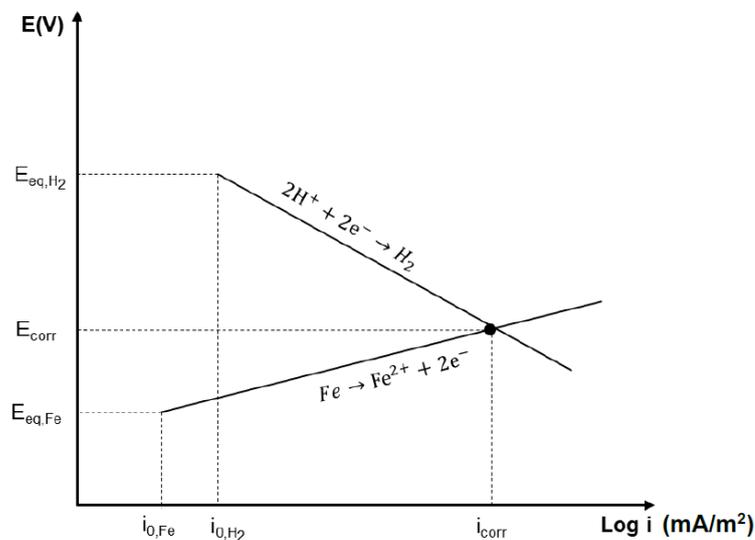


Figura 1.4: Diagramma di Evans - sviluppo di idrogeno [Pedefferri, 2010]

Il punto di funzionamento del sistema elettrochimico è dato dal punto (E_{corr}, i_{corr}) , l'intersezione tra la curva anodica e la curva catodica, dove (E_{corr}) è il *potenziale di libera corrosione* e (i_{corr}) è la *corrente di corrosione*. In particolare, E_{corr} cade nell'intervallo compreso tra i potenziali di equilibrio $E_{eq,a}$ e $E_{eq,c}$, mentre i_{corr} coincide con la densità limite di diffusione di ossigeno (i_L) in ambienti a controllo diffuso e determina la velocità di corrosione in $\mu m/anno$, definibile tramite la seguente equazione:

$$v_{corr} \approx 1.17 \cdot i_L \approx 11.7 \cdot [O_2] \cdot 2^{\frac{T-25}{25}} \quad (1.1.20)$$

dove $[O_2]$ è la concentrazione di ossigeno presente nell'ambiente e T è la temperatura in °C.

1.2 La protezione catodica - generalità

La protezione catodica [Revie, 2016] è una tecnica elettrochimica di controllo della corrosione cui si fa ricorso per prevenire o ridurre il deterioramento di metalli esposti ad ambienti aggressivi. Questa tecnica si basa sull'applicazione di una corrente continua tra un elettrodo (anodo) posto nell'ambiente e la superficie della struttura da proteggere (catodo). La circolazione di corrente, facendo riferimento alle reazioni descritte dalla (1.1.2) e (1.1.4), permette di fornire all'ossigeno un numero di elettroni pari a quello richiesto, prevenendo la dissoluzione del ferro e riducendo la sua velocità di corrosione. Di conseguenza, il potenziale del metallo si abbassa e, se portato al di sotto di quello di equilibrio ($E < E_{eq}$), la reazione anodica non avviene ed il metallo non si ossida, bensì, se sono già presenti dei composti ossidati sulla struttura da proteggere, essi tenderebbero a ridursi, tornando alla forma metallica pura. Queste condizioni sono dette *immunità termodinamica*. Se il potenziale del metallo (E) diminuisce rispetto al potenziale di corrosione (E_{corr}), ma non al punto tale da annullare il lavoro motore (1.1.11), la velocità di corrosione è ridotta ma non annullata e vale la seguente relazione:

$$E_{eq} < E < E_{corr} \quad (1.2.1)$$

Quindi, il potenziale del metallo è molto vicino al potenziale di equilibrio ($E \approx E_{eq}$), ma leggermente più alto, di conseguenza la struttura è in uno stato di *quasi immunità*. Dal punto di vista ingegneristico le condizioni di quasi immunità possono essere sufficienti per proteggere la struttura in quanto, sebbene E non sia inferiore ad E_{eq} , ne è talmente vicino che la reazione di ossidazione è molto lenta e la corrosione avviene ad una velocità trascurabile.

1.2.1 Le condizioni di protezione

Le condizioni di protezione catodica sono ottenute imponendo una corrente esterna tra un anodo e la struttura da proteggere.

La densità di corrente tale per cui la rete metallica è in protezione è chiamata *corrente di protezione* (I_{prot}), la quale dipende dalla reazione catodica. In presenza di una corrente esterna (I_e), secondo l'equazione (1.1.8), vale la seguente relazione:

$$I_a = I_c - I_e \quad (1.2.2)$$

dove I_a è la corrente associata alla reazione anodica e I_c la corrente associata alla reazione catodica. La corrosione non avviene se la I_a è nulla e affinché valga $I_a = 0$, la corrente catodica esterna I_e deve essere uguale alla corrente del processo catodico I_c , pari agli elettroni richiesti dalla reazione chimica. Dunque la condizione affinché non ci sia corrosione è:

$$I_c = I_e - I_{prot} \quad (1.2.3)$$

In Figura 1.5 sono schematizzate le condizioni elettrochimiche appena descritte tramite il diagramma di Evans.

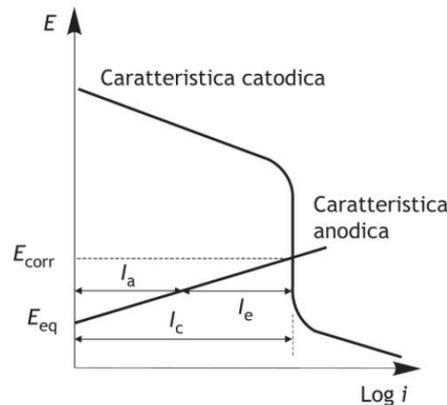


Figura 1.5: Le condizioni elettrochimiche in presenza di una corrente catodica esterna [Pedferri, 2010]

Il potenziale al di sotto del quale la rete metallica è in condizioni di immunità (o di quasi immunità) è chiamato *potenziale di protezione* (E_{prot}). Nella Figura 1.6 è riportata una tabella con i valori di potenziale di alcuni metalli nel terreno. Per l'acciaio al carbonio nel terreno aerato, il potenziale di protezione è -0.85 V CSE (come definito dallo standard internazionale ISO 15589-1); in condizioni anaerobiche ed in presenza di batteri solfato-riduttori, il potenziale di protezione è -0.95 V CSE. Tali valori sono considerati di quasi-immunità e corrispondono a velocità di corrosione inferiori a 0.01 mm/anno, considerata ingegneristicamente trascurabile.

Metalli	Terreno V vs CSE
Acciaio al carbonio:	
- condizioni normali	-0,85
- condizioni anaerobiche	-0,95
- nel calcestruzzo	-0,75
Rame e sue leghe	-0,45/-0,60
Piombo	-0,50/-0,65
Zinco	-1,00
Alluminio	-0,80
Acciaio inossidabile	-0,40

Figura 1.6: I potenziali di protezione adottati nei terreni [Lazzarri *et al.*, 2006]

1.2.2 I sistemi di protezione catodica

Esistono due principali sistemi di protezione catodica: ad *anodi galvanici (o sacrificali)* ed a *corrente impressa*.

- **Anodi galvanici** (Figura 1.7 - a): la struttura metallica da proteggere è accoppiata galvanicamente con un metallo meno nobile che funge da anodo. Questo collegamento permette il passaggio di corrente elettrica tra i due materiali, innescando un processo elettrochimico che riduce la corrosione della struttura da proteggere. Infatti, in una coppia galvanica, il metallo con potenziale elettrochimico più basso (l'anodo sacrificale, come zinco o magnesio) si ossida più facilmente ed inizia a corrodersi, mentre il metallo con potenziale più alto (la struttura da proteggere) agisce come il catodo e viene preservato dalla corrosione. Il vantaggio di questo sistema è la sua semplicità ed

autonomia, non richiede una manutenzione frequente o apparecchiature complesse, ma lo svantaggio è che gli anodi si consumano nel tempo e quindi devono essere periodicamente sostituiti, oltre che il sistema stesso non è molto efficace in ambienti in cui le resistenze sono elevate come i suoli secchi.

- **Corrente impressa** (Figura 1.7 - b): la struttura metallica è protetta tramite una corrente esterna, fornita da una fonte di alimentazione come un trasformatore o un generatore di corrente continua. Il polo positivo del generatore è collegato ad un opportuno dispersore anodico di corrente costituito da materiali duresvoli come grafite, titanio o platino, mentre il polo negativo è collegato alla rete metallica. Il vantaggio di questo sistema è la sua versatilità ed adattabilità a strutture di grandi dimensioni ed a condizioni ambientali molto sfavorevoli, ovvero condizioni in cui il livello di corrosione è elevato (ambienti ad alta resistività). Risulta particolarmente utile ed efficace per lunghe tubazioni interrate, ma lo svantaggio è che richiede ispezioni periodiche e manutenzione, oltre ad essere costoso richiedendo una fonte di alimentazione costante e potrebbe essere soggetto ad interferenze.

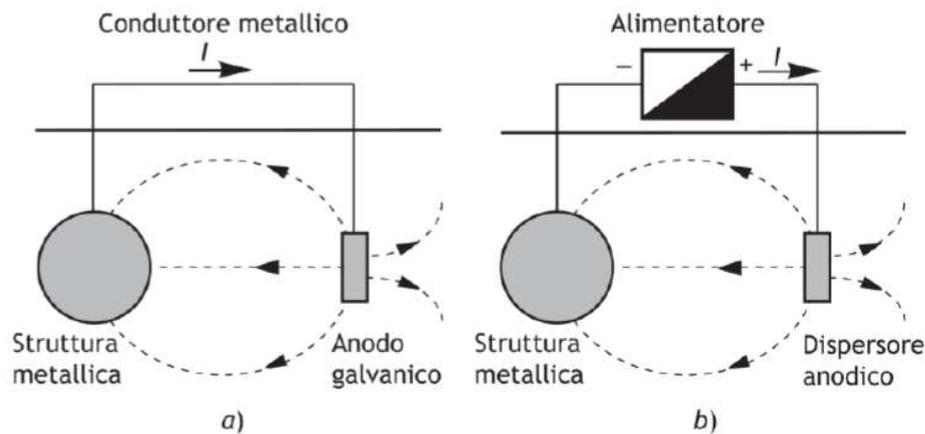


Figura 1.7: a) Il sistema di protezione catodica con anodo galvanico; b) Il sistema di protezione catodica a corrente impressa [Lazzarri *et al.*, 2006]

1.2.3 Il monitoraggio

Il monitoraggio della protezione catodica include tutte le operazioni mirate a controllare, sia in maniera diretta che indiretta, il livello di protezione delle strutture. Il metodo più comunemente adottato per verificare l'efficacia della protezione catodica si basa sulla misurazione del potenziale. Questa misurazione (Figura 1.8) viene effettuata tramite un elettrodo di riferimento, posizionato a contatto con l'elettrolita in cui la rete metallica è immersa. Generalmente l'elettrodo utilizzato nei terreni è di rame (Cu) o solfato di rame ($CuSO_4$) in quanto molto affidabile, stabile e fornisce un riferimento elettrochimico costante. La misura del potenziale è eseguita tramite un Voltmetro ad elevata impedenza: il polo negativo è collegato all'elettrodo di riferimento, mentre il polo positivo è collegato alla struttura da monitorare. Il potenziale misurato viene, successivamente, confrontato con il potenziale di protezione richiesto che, come spiegato nel Sottoparagrafo 1.2.1, risulta essere di -0.85 V CSE per l'acciaio in terreni aerati [International Organization for Standardization, 2015].

In genere, il potenziale misurato ad impianto acceso, chiamato E_{ON} , è dato dalla seguente relazione:

$$E_{on} = E_{IR-free} + I \cdot R = E_{IR-free} + \rho \cdot i \cdot d \quad (1.2.4)$$

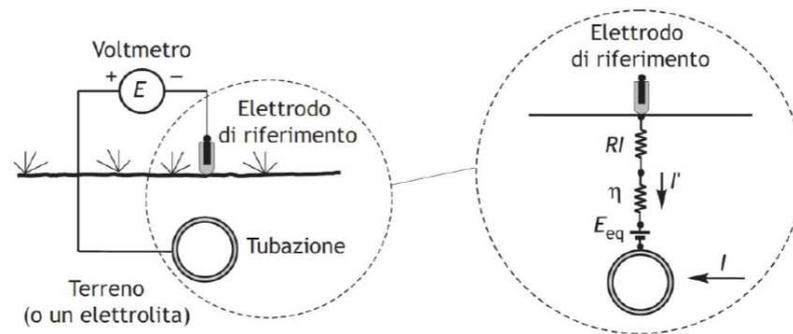


Figura 1.8: La misurazione del potenziale tramite elettrodo di riferimento [Lazzari *et al.*, 2006]

dove $I \cdot R$ è il termine di caduta ohmica dovuto alla corrente circolante (I) nel terreno, ρ la resistività del terreno e d la distanza tra l'elettrodo e la struttura. Il potenziale misurato al netto delle cadute ohmiche è detto $E_{IR-free}$ e corrisponde al vero livello di polarizzazione della struttura.

1.3 Le soluzioni proposte dall'azienda Automa s.r.l.

Automa s.r.l. è un'azienda di Casine di Paterno (AN) che si occupa della progettazione, ingegnerizzazione e produzione di tecnologie Made in Italy per il monitoraggio remoto in ambito Oil, Gas e Water. In particolare, ha sviluppato un dispositivo chiamato *G4C-PRO* per il monitoraggio remoto della protezione catodica, fondamentale, quindi per il telecontrollo degli alimentatori di corrente esterna.

1.3.1 Il dispositivo *G4C-PRO*

Il dispositivo *G4C-PRO* (Figura 1.9) è collocato all'interno di una cassetta metallica posta in corrispondenza di ciascun punto in cui viene effettuato il monitoraggio della protezione catodica. Questo dispositivo effettua una misurazione al secondo di potenziale ON, interferenza AC, corrente di giunto, erogata dall'alimentatore o di drenaggio, corrente di polarizzazione. Tali misurazioni sono, successivamente, elaborate in un report giornaliero in cui è riportato il valore medio, minimo e massimo con data ed ora di rilevamento, lo scarto quadratico medio, la moda, la variabilità ed il numero totale di valori fuori soglia. Questi campioni sono, poi, registrati sulla piattaforma locale di gestione e raccolta dati chiamata *WebProcat*, in cui sono riportate informazioni aggiuntive come la cartografia e tramite cui è possibile configurare i dispositivi o telecontrollare gli alimentatori di corrente.

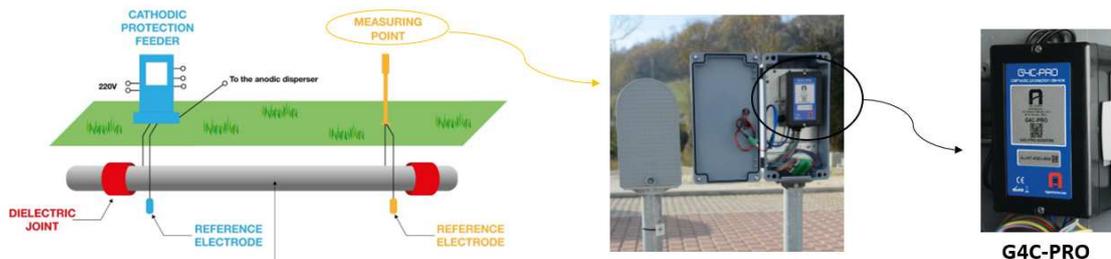


Figura 1.9: Il dispositivo *G4C-Pro* ed il suo collocamento

Tramite i dati giornalieri forniti dal *G4C-PRO* è possibile valutare quando un sistema è fuori soglia, ovvero quando un singolo punto di misura ha almeno un campione giornaliero di differenza di potenziale misurata (E_{ON}) superiore a $-0.85V$. Il rilevamento del dato fuori soglia può avvenire sia in modalità manuale che in modalità automatica. In modalità manuale, un operatore verifica da *WebProcat*, tramite specifici indicatori di allarme, se sono presenti punti fuori protezione; in modalità automatica, la piattaforma di raccolta e gestione dati è configurata in modo da inviare uno status giornaliero dei dispositivi via e-mail all'operatore. In seguito a tale rilevamento, l'operatore valuta come agire in seguito ai dati analizzati e le verifiche effettuate lungo la rete.

1.3.2 Gli alimentatori di corrente per la protezione catodica

L'alimentatore è l'elemento del sistema di protezione catodica addetto ad erogare la corrente necessaria a rendere la struttura metallica protetta, nonchè il catodo del processo elettrochimico. Gli attuali alimentatori prevedono due principali modalità di funzionamento: manuale o automatica.

- **manuale:** l'alimentatore prevede una *regolazione a potenziale costante* tale per cui la tensione in uscita è regolata in modo tale da erogare la corrente (I) ottenuta, secondo la legge di Ohm, dalla relazione $I = V/R$, dove R è la resistenza totale del circuito elettrico ai capi dell'alimentatore, le cui componenti principali sono la resistenza della struttura da proteggere e la resistenza del terreno in cui è interrata. In questa regolazione, la corrente erogata dipenderà fortemente dalla stagionalità (terreno umido/secco) e da variazioni puntuali della resistenza del terreno in conseguenza delle condizioni meteorologiche (pioggia, caldo, etc...). Questa modalità è ormai piuttosto superata dalle modalità di funzionamento automatico. In questa regolazione, la presenza di un elettrodo di riferimento per effettuare una misura di potenziale On (E_{ON}) non è necessaria.
- **automatica:** l'alimentatore prevede una *regolazione a potenziale costante* o una *regolazione a corrente costante*:
 - *regolazione a corrente costante:* all'alimentatore è indicato un valore di corrente di uscita (I) da mantenere costante, l'alimentatore ha un canale di misura che legge il valore della corrente effettivamente erogata e varia nel tempo la sua tensione di uscita (V) per mantenere in ogni momento costante la corrente al valore di I impostato. In questa regolazione, la presenza di un elettrodo di riferimento per effettuare una misura di potenziale On (E_{ON}) non è necessaria.
 - *regolazione a potenziale costante:* all'alimentatore viene indicato un valore chiamato *potenziale On locale* da mantenere costante, l'alimentatore ha un canale di misura che legge la misura di potenziale On e varia nel tempo la corrente erogata (I) per mantenere in ogni momento costante il potenziale On al valore impostato. In questa regolazione, la presenza di un elettrodo di riferimento per effettuare la misura di potenziale On (E_{ON}) è necessaria.

Nella presente tesi sono stati presi in esame tutti i sistemi di protezione catodica a corrente impressa con alimentatori gestiti in modalità automatica e regolazione a potenziale costante.

1.3.3 La gestione di un sistema di protezione catodica con alimentatore di corrente a potenziale costante

Si definisce *sistema di protezione catodica* una pipeline interrata di una determinata area geografica in cui è presente uno o più alimentatori che erogano una corrente tale da mantene-

re la differenza di potenziale, tra la conduttura metallica ed il terreno, al di sotto di $-0.85V$. Lungo la conduttura sono presenti dei punti di misura, distanti l'uno dall'altro, in cui è controllata la differenza di potenziale del metallo rispetto ad un elettrodo di riferimento. In Figura 1.10 è riportato un esempio di cartografia, reperita dalla piattaforma *WebProcat*, in cui è rappresentato un sistema di protezione catodica. Il simbolo in rosso rappresenta l'alimentatore di corrente, i simboli in arancione i punti di misura telecontrollati ed i simboli in verde verde i punti di misura gestiti tramite operatore. In ciascun punto di misura telecontrollato è installato un dispositivo *G4C-PRO* che effettua 86.400 misurazioni giornaliere.

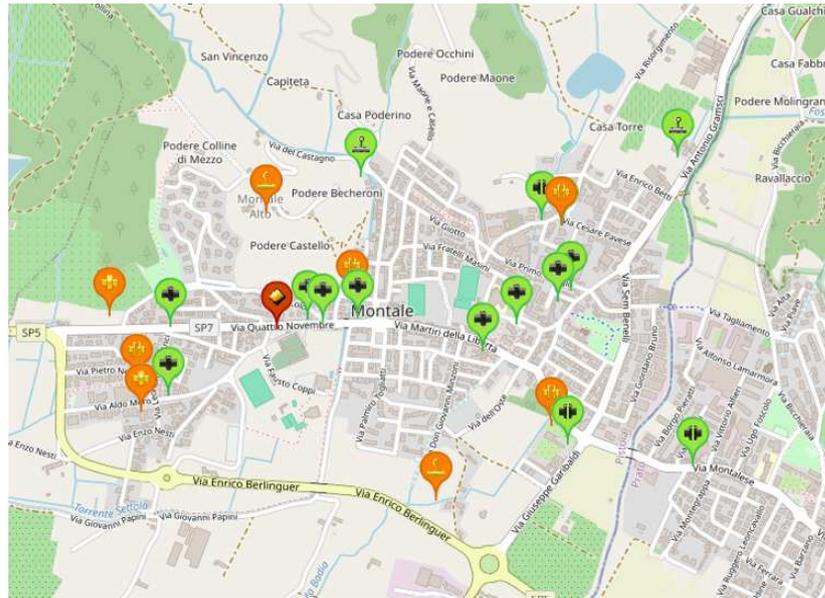


Figura 1.10: Una generica cartografia di *WebProcat* raffigurante un sistema di protezione catodica dotato di un solo alimentatore di corrente impressa, 8 punti di misura telecontrollati e 13 punti di misura gestiti tramite operatore

Per quanto concerne l'alimentatore a potenziale costante, questo dispone di un PID che ha il compito di regolare la corrente di alimentazione; tale PID riceve in ingresso la differenza tra un set-point inserito manualmente, scelto da tecnici esperti, ed il potenziale *On* misurato nel punto in corrispondenza dell'alimentatore, chiamato *potenziale On locale*. Dunque, l'alimentatore eroga una quantità di corrente tale da portare la differenza di potenziale in corrispondenza dell'alimentatore ad un valore pari al set-point. In Figura 1.11 è riportato lo schema a blocchi riassuntivo del funzionamento di un sistema di protezione catodica a corrente impressa con alimentatore a potenziale costante. I colori scelti richiamano la legenda riportata nella cartografia, infatti, in rosso è riportato l'alimentatore, il PID ed il punto di misura in corrispondenza dell'alimentatore, mentre in arancione la pipeline su cui sono distribuiti tutti i punti di misura telecontrollati.

Un aspetto importante da notare è che la corrente influisce su tutta la conduttura, ma la sua regolazione è effettuata solo in corrispondenza dell'alimentatore; infatti, quest'ultimo non ha informazioni sull'effetto che la corrente erogata ha effettivamente sul resto della struttura: questa analisi è di solito gestita a posteriori da tecnici esperti, che dalle misure e verifiche fatte lungo la rete, possono decidere di cambiare il valore del setpoint indicato come target di potenziale locale all'alimentatore. Per tale ragione, se si registrassero una serie di campioni fuori soglia dovuti, ad esempio, ad un guasto, l'azione da compiere sarebbe *correttiva*. L'obiettivo della presente tesi, infatti, è trasformare l'azione *correttiva* in azione *predittiva*, andando a prevedere le differenze di potenziale dei punti di misura dislocati lungo la pipeline.

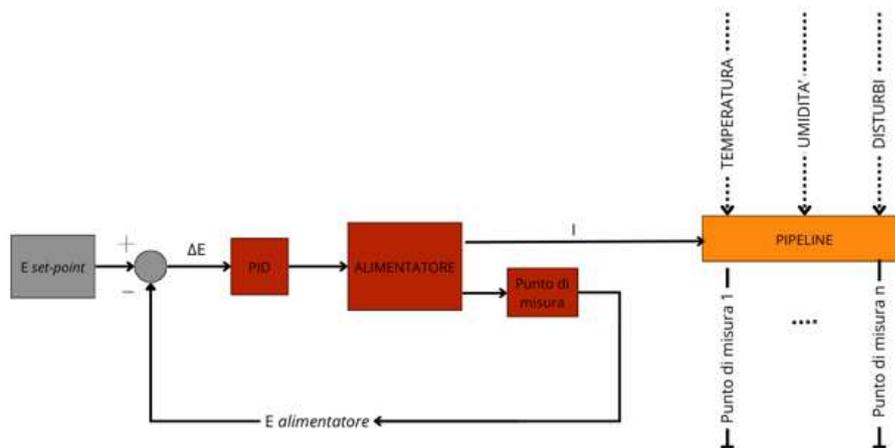


Figura 1.11: Il funzionamento di un sistema di protezione catodica a corrente impressa con alimentatore a potenziale costante

Il dataset ed i sistemi scelti

In questo capitolo sono riportate tutte le informazioni relative ai dati utilizzati per lo sviluppo del presente elaborato; in particolare, sono descritti i sistemi scelti di cui sarà effettuata la previsione della differenza di potenziale E_{on} , misurata dal dispositivo G4C-PRO, di uno specifico punto di misura. Le analisi principali sono state svolte in Python, sfruttando la piattaforma Jupyter Notebook.

2.1 L'origine dei dati

L'azienda AUTOMA s.r.l. predispone di un'applicazione web chiamata *WebProcat* per la gestione, raccolta ed organizzazione dei dati storici relativi ai sistemi di protezione catodica di ciascuno dei loro clienti. I campioni su cui è stata sviluppata la seguente tesi fanno riferimento all'azienda CENTRA s.r.l. (2.1), la società di ESTRA SpA che si occupa del vettoriamento di gas naturale, della distribuzione e della vendita di GPL.



Figura 2.1: Logo di CENTRIA s.r.l.

2.1.1 Il database

Dalla piattaforma web sono state estratte tutte le informazioni di interesse e trasferite su *Microsoft SQL Server*, un sistema software progettato per la gestione delle basi di dati. Le tabelle di interesse sono state:

- **tblPoint**, contenente tutte le informazioni relative ai punti di misura di ciascun sistema di protezione catodica;
- **tblSystem**, contenente tutte le informazioni relative ai sistemi di protezione catodica;
- **tblMeasure**, contenente le caratteristiche delle misure effettuate sui punti di misura di ciascun sistema di protezione catodica;

- **tblDualChMeasure**, contenente i valori numerici delle misure effettuate sui punti di misura di ciascun sistema di protezione catodica.

2.1.2 Il dataset

Il database è stato connesso alla piattaforma *Jupyter Notebook*, tramite la libreria *pyodbc* di Python che si interfaccia con *ODBC Driver 17 for SQL Server*, il quale permette la connessione a *Microsoft SQL Server*. Tramite Query sono stati estratti i dati utili per l'analisi, in un periodo temporale compreso tra il 2012-01-01 e il 2024-04-18. In particolare, le tabelle precedentemente descritte sono state unite tramite la funzione "inner join" e sono state connesse come segue:

- **tblMeasure e tblPoint**, tramite l'attributo *PointId*, per connettere le caratteristiche delle misure ai rispettivi punti di misura;
- **tblPoint e tblSystem**, tramite l'attributo *SystemId*, per connettere i punti ai rispettivi sistemi;
- **tblMeasure e tblDualChMeasure**, tramite l'attributo *measureID*, per connettere le caratteristiche delle misure ai valori delle misure stesse.

Una volta unite le tabelle sono stati eseguiti alcuni filtri e sono state prese in considerazione, per ciascun sistema di protezione catodica, le misure di differenza di potenziale relative ai punti di misura telecontrollati e la corrente e differenza di potenziale relativi all'alimentatore di corrente. I campioni sono stati salvati in un dataframe ed i campi scelti per l'analisi sono i seguenti:

- *SystemName*: il codice identificativo del sistema di protezione catodica, composto da un alimentatore ed "n" punti di misura.
- *PointId*: il codice identificativo del punto corrispondente all'alimentatore o al punto di misura telecontrollato di un determinato sistema.
- *FromDateTime*: la data di inizio relativa alla registrazione dei campioni.
- *Ch1MeasureUnit*: la misura del campione giornaliero corrispondente all'alimentatore o al punto di misura telecontrollato. In particolare:
 - Punto di alimentazione:
 - * *I*: corrente di alimentazione;
 - * *ddp.dc*: la differenza di potenziale continua del punto in corrispondenza dell'alimentatore;
 - Punto di misura:
 - * *ddp.dc*: la differenza di potenziale continua del punto i-esimo.
- *Ch1Med*: il valore medio della misura giornaliera registrata.
- *GpsX*: la coordinata geografica del punto di misura corrispondente alla longitudine.
- *GpsY*: la coordinata geografica del punto corrispondente alla latitudine.

In Figura 2.2 sono riportate le prime cinque righe del dataframe considerato.

	SystemName	PointId	FromDateTime	Ch1MeasureUnit	Ch1Med	GpsX	GpsY
1	S04VE	9231	2012-01-01	ddp.dc	-1.919	44.0676664042303	11.1500086303478
2	S01SE	9283	2012-01-01	ddp.dc	-1.863	43.823836786675	11.2176265133854
3	S04VA	9275	2012-01-01	ddp.dc	-1.794	43.931262718749	11.127180476381
4	S05ML	8980	2012-01-01	ddp.dc	-1.945	43.9136341951042	11.0088220643801
5	S02LS	9277	2012-01-01	ddp.dc	-1.209	43.7703650308824	11.1021590799331

Figura 2.2: Le prime 5 righe del dataframe contenente le informazioni utili per le analisi

2.2 La scelta delle ddp.dc da predire

Ai fini della tesi, la predizione è stata effettuata su una singola differenza di potenziale di un determinato sistema. Dunque, si è rivolta maggiore attenzione verso due sistemi con le misure di ddp.dc più interessanti da predire. Per effettuare la scelta sono stati raccolti in un unico dataframe tutti i sistemi con un solo alimentatore e nessun drenaggio o attraversamento ferroviario. In seguito, per ciascuno dei punti di misura di ogni sistema, sono stati individuati tutti gli eventi "prevedibili", ovvero tutti quegli eventi tali per cui i campioni di differenza di potenziale risultino al di sopra della soglia di protezione per 15 giorni consecutivi. Tutti gli eventi tali per cui i campioni di ddp.dc risultano al di sopra di -0.85 V per meno di 15 giorni consecutivi, sono considerati "imprevedibili", quindi oggetto di disturbi sconosciuti alle analisi. La distinzione in eventi "prevedibili" ed "imprevedibili" è stata eseguita per una pura semplificazione. È stata stilata una lista degli eventi "prevedibili" di ciascun punto di ogni sistema ed in Figura 2.3 è riportato l'esempio del sistema *S03ML*, composto da 5 punti di misura; il primo registra 3 eventi prevedibili ciascuno di 24, 24 e 16 giorni in cui la differenza di potenziale è stata al di sopra della soglia di protezione, l'ultimo, invece, non ha eventi prevedibili.

```
S03ML: 5
[[24, 24, 16], [24, 15], [25, 16], [24, 16], []]
```

Figura 2.3: Lista degli eventi prevedibili per ciascun punto di misura del sistema con ID *S03ML*

Per ciascun sistema è stato individuato il punto di misura con maggiori eventi prevedibili e, di questo, è stata effettuata la somma aritmetica dei campioni fuori soglia. Un ulteriore aspetto da prendere in considerazione per effettuare una previsione attendibile della ddp.dc è la gestione appropriata dei dati mancanti, ovvero quelli corrispondenti, sia ai valori *NaN* registrati nel dataset, sia alle righe completamente assenti. Dunque, per ciascun punto di misura con il maggior numero di eventi prevedibili nel proprio sistema, sono stati calcolati i giorni mancanti. Una volta terminate tali analisi, sono stati registrati in un unico dataframe tutte le ddp più interessanti per ciascun sistema, con le relative informazioni. I campi delineati per tale dataframe sono:

- *SystemName*: codice identificativo del sistema;
- *num_point*: il numero di punti di misura telecontrollati di ciascun sistema (tra i punti è compreso anche l'alimentatore);
- *system_missing_days*: il numero totale di dati mancanti individuate per l'intero sistema;
- *max_FS*: la somma aritmetica dei campioni fuori soglia corrispondenti al punto di misura con maggiori eventi prevedibili;

- *point_max_FS*: il punto di misura con maggiori eventi prevedibili;
- *pointId_missig_days*: il numero totale di date mancanti individuate per il punto con maggiori eventi prevedibili;
- *pointId_%_missing_days*: la percentuale di date mancanti del punto con maggiori eventi prevedibili, calcolata rispetto all'intero periodo temporale in cui sono stati registrati i campioni del punto stesso;
- *pointId_start_date*: la data di inizio registrazione dei campioni corrispondenti al punto con maggiori eventi prevedibili;
- *pointId_end_date*: la data di fine registrazione dei campioni corrispondenti al punto con maggiori eventi prevedibili.

La scelta è stata effettuata sulla base del campo *max_FS* e, come illustrato in Figura 2.4, è stata evidenziata la corrispondente colonna tramite una scala di colori, dove il rosso rappresenta il valore più alto ed il verde il valore più basso. Questo campo è stato ordinato in modo decrescente. Inoltre, è stata evidenziata anche la colonna corrispondente all'attributo *pointId_%_missing_days*, un importante aspetto da prendere in considerazione per la previsione.

SystemName	num_pointId	system_missing_days	max_FS	pointId_max_FS	pointId_missing_days	pointId_%_missing_days	pointId_start_date	pointId_end_date
AG31	5	-3278	888	17164	142	6%	2018-02-13	2024-04-18
MV11	2	-27	643	17271	24	1%	2018-06-14	2024-04-18
SA24	4	-2788	636	17394	434	22%	2018-06-07	2023-12-08
PS38	3	-1574	517	17336	310	14%	2018-02-13	2024-04-18
CA26	3	-1657	466	17201	99	7%	2020-06-30	2024-04-18
S04ML	6	-7349	445	8984	609	15%	2013-05-23	2024-04-18
G60226SPC07	4	-117	430	17485	4	1%	2022-05-13	2024-04-16
CA28	3	-2044	394	17206	439	21%	2018-06-19	2024-04-18
S03SC	7	-16539	355	9090	232	5%	2012-01-01	2024-04-18
FR50	4	-2504	343	17230	1	0%	2020-12-03	2024-04-18
RI18	2	-48	292	17363	3	0%	2018-06-13	2024-04-18
MF35	3	-1897	245	17289	8	0%	2018-06-14	2024-04-18
S01CL	4	-7521	241	5892	21	0%	2012-10-17	2024-04-18
G04S11	3	-3537	235	12486	780	27%	2016-04-12	2024-04-18
S07SE	4	-4662	233	5766	1	0%	2012-12-18	2024-04-18
S01MM	9	-16009	233	9227	513	24%	2018-06-02	2024-04-18
S04LS	4	-5374	231	9540	141	3%	2012-08-02	2024-04-18
SP37	3	-2489	220	17406	1	0%	2018-06-15	2024-04-18
PB16	4	-3190	215	17331	191	14%	2020-07-07	2024-04-18
S02CB	7	-9191	186	6058	3	0%	2012-06-06	2024-04-18
CA19	4	-2204	186	17187	238	11%	2018-06-19	2024-04-18
SIG003	2	-3	184	18568	1	0%	2017-07-13	2024-04-18
S01CB	7	-14923	180	6031	52	1%	2012-09-18	2024-04-18
S01LS	5	-5822	169	9298	30	1%	2012-01-20	2024-04-18
S02CR	5	-5144	156	6092	121	3%	2012-07-06	2024-04-18
SIG002	1	-122	156	18552	122	5%	2017-03-14	2024-01-06
C007	5	-5600	155	11018	1515	56%	2016-11-28	2024-04-18
G23S04	4	-4127	154	9452	1981	47%	2012-08-28	2024-04-18
S05VE	6	-10206	149	10141	52	1%	2012-08-29	2024-04-18
G04S07	4	-2869	148	9714	881	21%	2012-09-18	2024-04-18

Figura 2.4: Le prime 30 righe del foglio excel che riassume le principali caratteristiche di interesse di ciascun sistema, utili per la scelta delle ddp.dc da prevedere

La scelta è ricaduta sui sistemi *MV11* e *S04ML*. Il sistema *MV11* dispone del punto di misura con *ID 17271* che ha molti punti fuori soglia e l'1% dei dati risultano essere mancanti, ma i campioni sono registrati in un arco di tempo limitato (poco meno di 6 anni). Il sistema *S04ML* dispone del punto di misura con *ID 8984* che ha molti punti fuori soglia ed il 15% dei dati risultano essere mancanti, ma i campioni sono registrati in un arco di tempo più ampio (poco meno di 11 anni).

2.2.1 Il sistema MV11

Il sistema *MV11* è un sistema situato nei pressi del comune di Macchia Valfortone, in provincia di Campobasso. È costituita da un solo alimentatore e 6 punti di misura, di cui, escludendo il punto corrispondente all'alimentatore, solo uno è telecontrollato. In Figura 2.5 è riportata la cartografia della zona in questione che è possibile visualizzare da *WebProcat*; il punto cerchiato in rosso è quello su cui è stata effettuata la previsione.

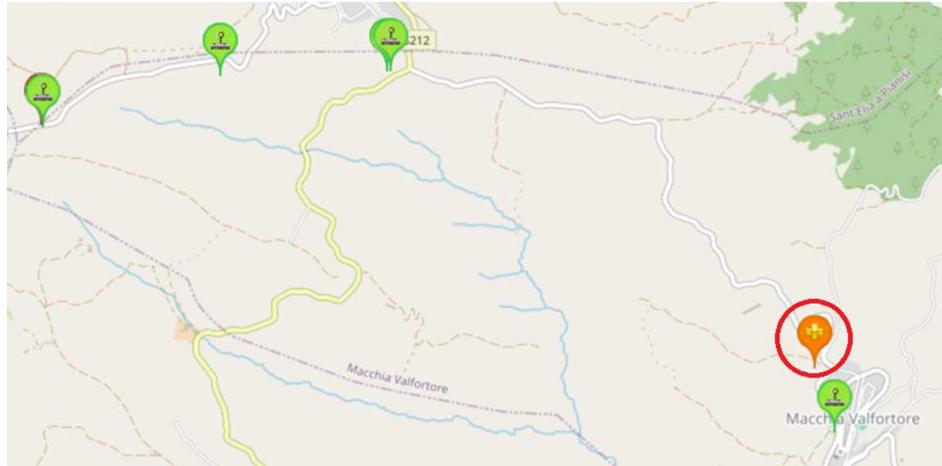


Figura 2.5: La cartografia corrispondente al sistema con *ID MV11*

In Figura 2.6 sono illustrate le caratteristiche principali dei punti di cui è composto il sistema di protezione catodica considerato. Alcuni dei punti descritti non sono visualizzabili sulla cartografia in quanto risultano molto vicini e, per la dimensione dell'immagine, sembrano sovrapposti.

	TIPOLOGIA DI PUNTO	NOME PUNTO	ID PUNTO	MISURE PUNTO
	ALIMENTATORE	ALM00058PC01	17266	l - ddp.dc
	PUNTO DI MISURA (Telecontrollato – giunto chiuso)	GD000384G1_F	17271	ddp.dc
	PUNTO DI MISURA (Controllo tramite operatore)	MV11PC2 MV11PC3 MV11PC5 MV11PC7	-	-
	PUNTO DI MISURA (Controllo tramite operatore – giunto chiuso)	GD000386G1_F	-	-

Figura 2.6: Le caratteristiche riassuntive del sistema con *ID MV11*

I dati relativi alla corrente sono registrati nel database a partire dal 2020-03-17, mentre la ddp.dc in corrispondenza dell'alimentatore (PointId 17266) è registrata a partire dal 2018-06-15 e la ddp.dc del punto telecontrollato identificato con l' *ID 17271* è registrata a partire dal 2018-06-14. Dunque, il dataset considerato per tale sistema, riporta 1.493 campioni di corrente e 2.113 campioni di ddp.dc da prevedere, ovvero la ddp.dc con *ID 17271*.

In Figura 2.7 sono riportati gli andamenti temporali della corrente e delle differenze di potenziale in corrispondenza dei punti di misura, ordinati in base alla distanza dall'alimentatore; in particolare, sono stati evidenziati nei grafici delle ddp.dc, tramite degli "scatter" di colore rosso, tutti i punti che superano la soglia di protezione, ovvero il valore di -0.85V . È evidente come il punto con ID 17271 ha molti punti di misura fuori soglia, concentrati tra la metà dell'anno 2018 e la metà dell'anno 2021.

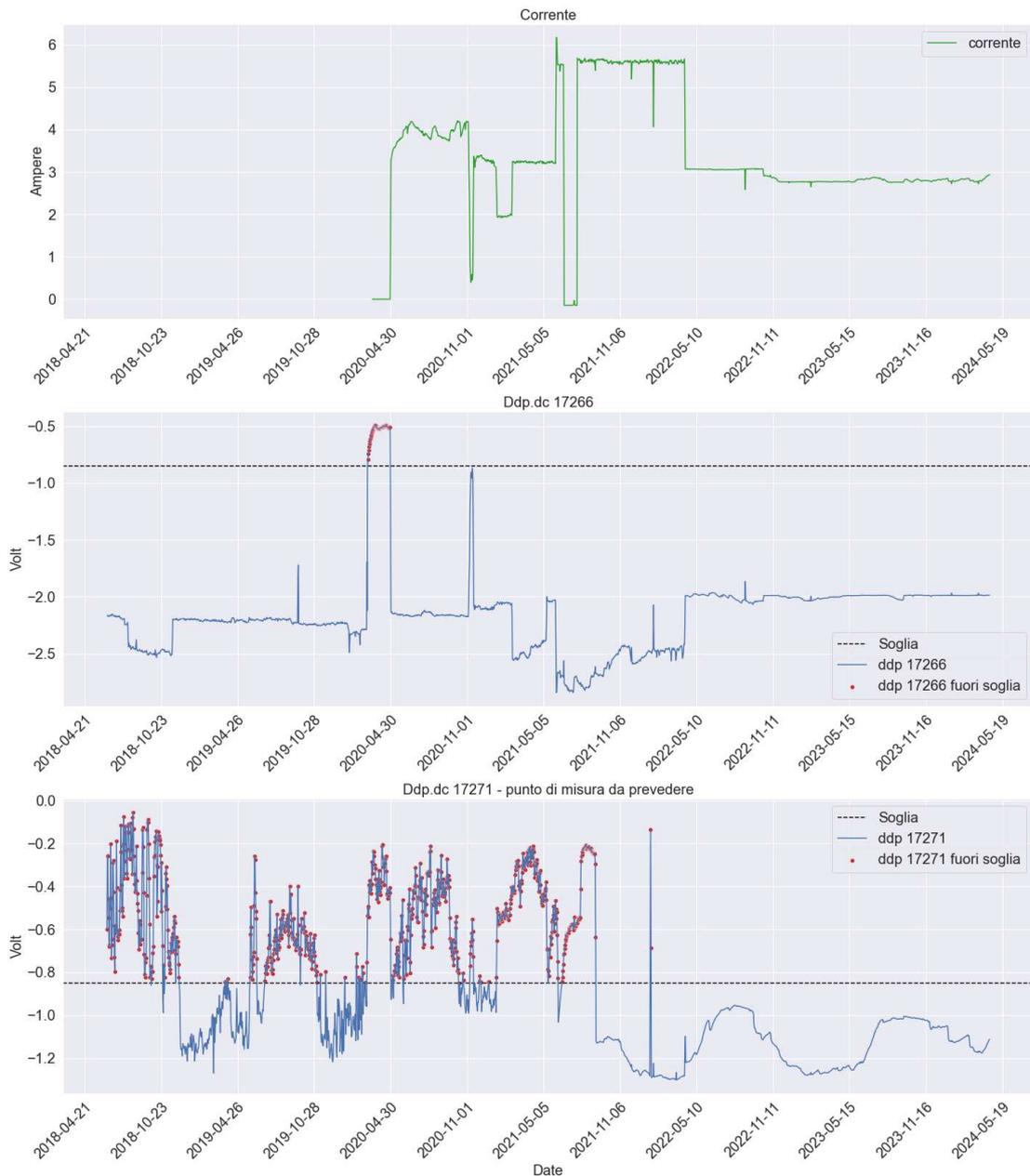


Figura 2.7: L'andamento nel tempo della corrente, della ddp.dc in corrispondenza dell'alimentatore di corrente e della ddp.dc in corrispondenza del punto con ID 17271, nonché il punto di misura da prevedere

Per visualizzare la distribuzione della corrente e della differenza di potenziale dei punti di misura, sono riportati in Figura 2.8 e 2.9 i rispettivi boxplot. Il primo mostra che la mediana si aggira intorno ai 3A ed il valore dei campioni si estende tra un minimo di circa 1.8A ed un massimo di circa 5.8A; i valori al di sotto dello zero sono degli outlier ed evidenziano

che in quei determinati istanti l'alimentatore satura. Gli altri due boxplot evidenziano che, la distribuzione dei dati della ddp con ID 17266 sono prevalentemente sotto la soglia, a differenza della ddp ID 17271.

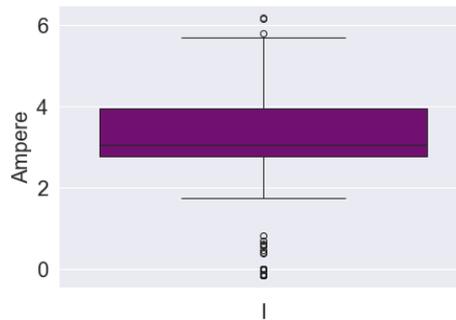


Figura 2.8: Boxplot relativo alla distribuzione di corrente dell'alimentatore del sistema con ID MV11.

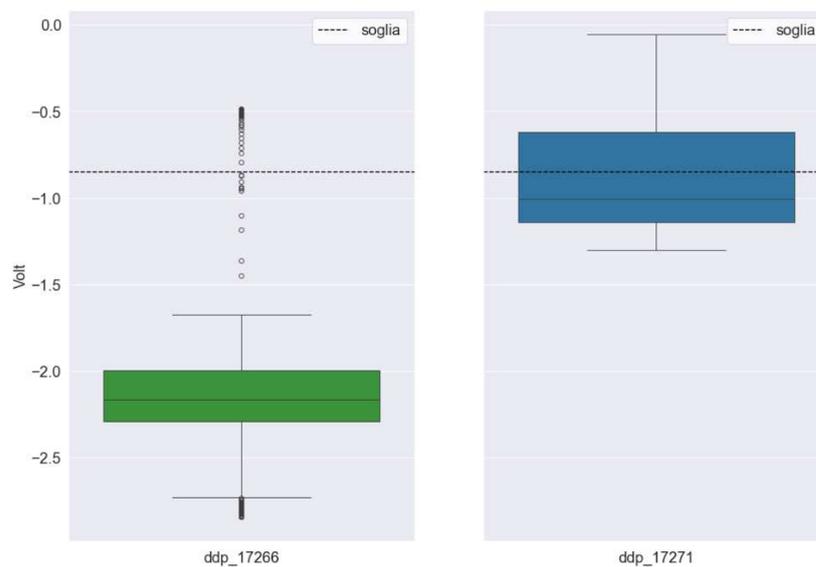


Figura 2.9: Boxplot relativi alle distribuzioni di differenza di potenziale in corrispondenza dell'alimentatore e del punto di misura telecontrollato del sistema con ID MV11.

2.2.2 Il sistema S04ML

Il sistema S04ML è un sistema situato nei pressi del comune di Montale, in provincia di Pistoia. È costituita da un solo alimentatore e 10 punti di misura, di cui, escludendo il punto corrispondente all'alimentatore, 4 sono telecontrollati. In Figura 2.10 è riportata la cartografia della zona in questione ed il punto cerchiato in rosso è quello su cui è stata effettuata la previsione.

In Figura 2.11 sono illustrate le caratteristiche principali dei punti di cui è composto il sistema di protezione catodica considerato. Alcuni dei punti descritti non sono visualizzabili sulla cartografia in quanto risultano molto vicini e, per la dimensione dell'immagine, sembrano sovrapposti.

I dati relativi alla corrente sono registrati nel database a partire dal 2018-10-23, mentre la ddp.dc in corrispondenza dell'alimentatore (PointId 9224) è registrata a partire dal 2018-06-15; le ddp.dc dei relativi punti telecontrollati sono registrati:

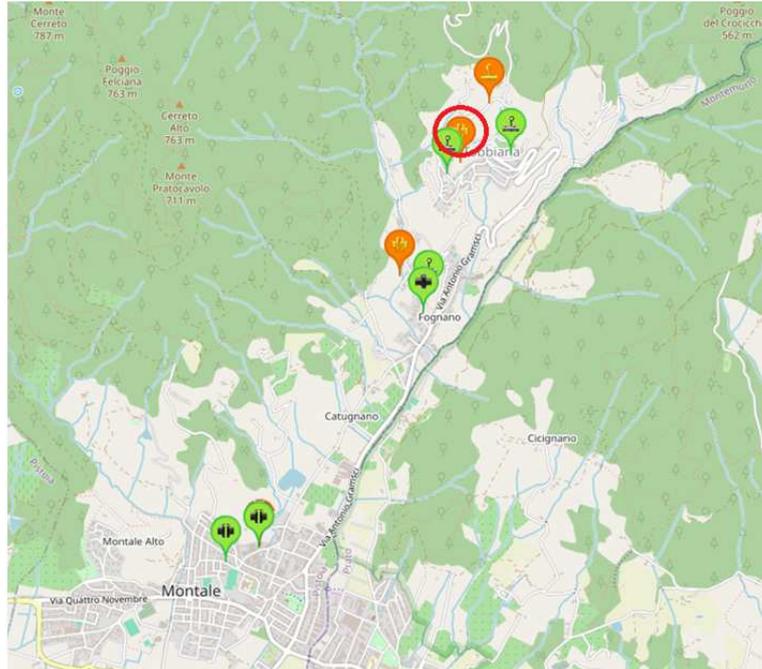


Figura 2.10: La cartografia corrispondente al sistema con ID S04ML

- ID 9225, a partire dal 2012-12-10,
- ID 9167, a partire dal 2015-10-24,
- ID 8984, a partire dal 2013-05-23,
- ID 9086, a partire dal 2012-12-10.

Dunque, il dataset considerato per tale sistema, riporta 1.982 campioni di corrente e 3.377 campioni di ddp.dc da prevedere, ovvero la ddp.dc con ID 8984.

In Figura 2.12 sono riportati gli andamenti temporali della corrente e delle differenze di potenziale in corrispondenza dei punti di misura, ordinati in base alla distanza crescente dall'alimentatore di corrente; in particolare, sono stati evidenziati nei grafici delle ddp, tramite degli "scatter" di colore rosso, tutti i punti che superano la soglia di protezione, ovvero il valore di $-0.85V$. È evidente come il punto con ID 8984 ha molti campioni fuori soglia rispetto agli altri.

Per visualizzare la distribuzione della corrente e della differenza di potenziale dei punti di misura, sono riportati in Figura 2.13 e Figura 2.14 i rispettivi boxplot. Il primo mostra che la mediana si aggira intorno a 8A ed il valore dei campioni si estende tra un minimo di circa 5.9A ed un massimo di 10A. I valori al di sopra del massimo ed al di sotto del minimo risultano degli outlier ed è proprio in corrispondenza di questi valori di corrente che, tendenzialmente, le rispettive differenze di potenziale dei punti di misura sono fuori soglia. Gli altri boxplot evidenziano che, man mano che la distanza tra il punto di misura e l'alimentatore aumenta, la distribuzione dei campioni si avvicina sempre di più alla soglia; in particolare, il grafico corrispondente alla ddp.dc con ID 8984 ha una mediana poco al di sotto di $-0.85V$, ma il cui valore risulta essere il più alto rispetto a tutte le mediane dei rispettivi boxplot di ddp.dc presi in esame.

Come anticipato nel Paragrafo 2.2, il punto di misura con ID 8984, nonché uno dei punti di misura scelti per la previsione, riporta che il 15% dei dati è mancante. A tal proposito, per visualizzare i suddetti campioni, Figura 2.15 è riportato l'andamento nel tempo della

	TIPOLOGIA DI PUNTO	NOME PUNTO	ID PUNTO	MISURE PUNTO
	ALIMENTATORE	MLA04M4_F	9224	1 - ddp.dc
	PUNTO DI MISURA (Telecontrollato – giunto chiuso)	ML25G2VF	9225	ddp.dc
	PUNTO DI MISURA (Telecontrollato – giunto aperto)	ML20G2VB ML22G2VB	9167 8984	ddp.dc ddp.dc
	PUNTO DI MISURA (Telecontrollato)	ML27P1_B	9086	ddp.dc
	PUNTO DI MISURA (Controllo tramite operatore)	-	-	-
	PUNTO DI MISURA (Controllo tramite operatore – giunto chiuso)	-	-	-
	PUNTO DI MISURA (Controllo tramite operatore – giunto aperto)	-	-	-

Figura 2.11: Le caratteristiche riassuntive del sistema con *ID S04ML*

ddp.dc corrispondente, ponendo una particolare attenzione alle ddp.dc mancanti, evidenziate tramite uno “scatter” di colore arancione, volutamente in corrispondenza del valore 0V. In particolare, è evidente la presenza di due gruppi di ddp.dc mancanti, il primo comprende 41 campioni ed il secondo 563 campioni; inoltre sono presenti 5 dati mancanti isolati tra loro. Questi dati saranno opportunamente gestiti come descritto nel prossimo paragrafo.

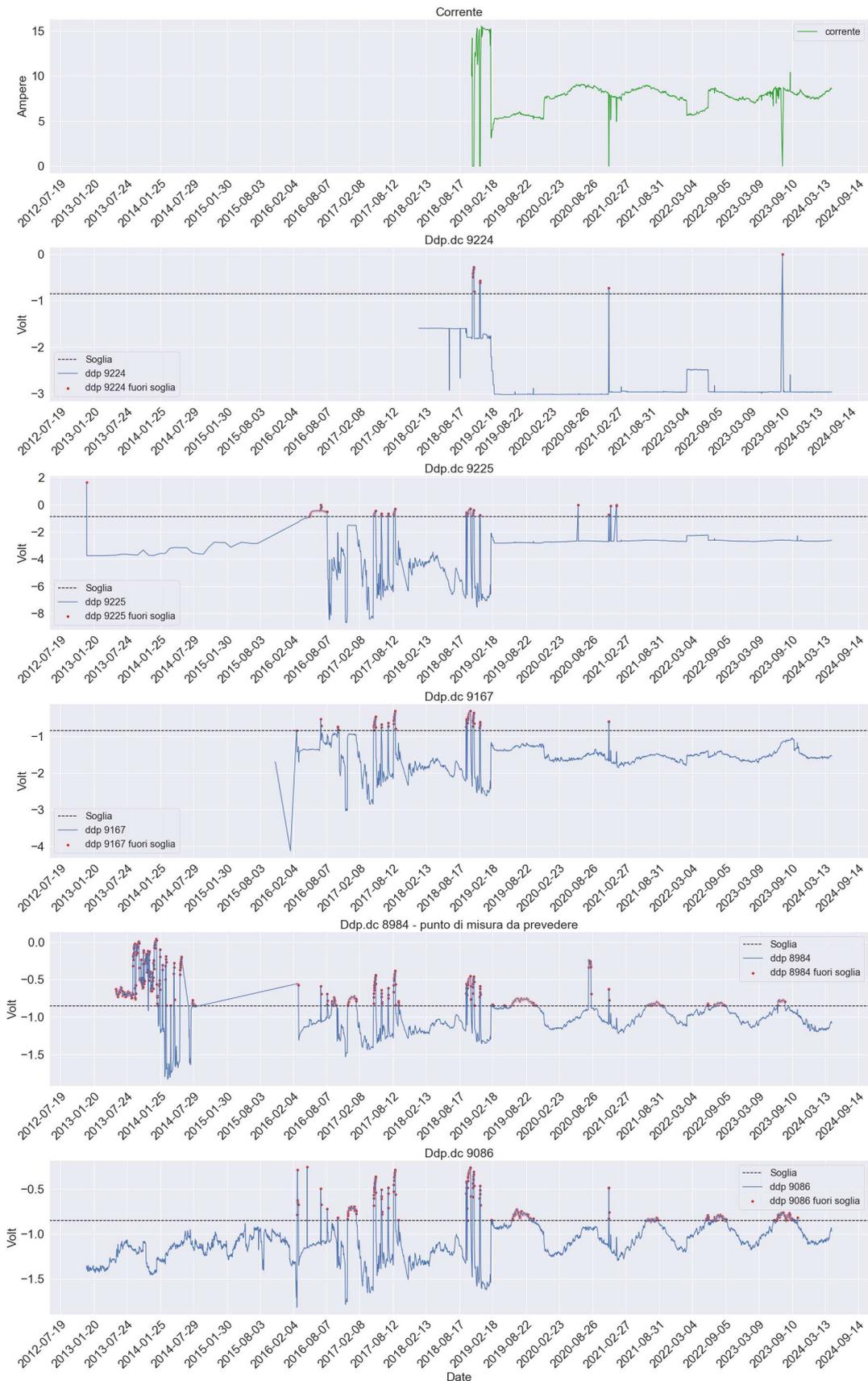


Figura 2.12: L'andamento nel tempo della corrente, della ddp.dc in corrispondenza dell'alimentatore di corrente e delle ddp.dc in corrispondenza dei punti di misura telecontrollati, in ordine di distanza dall'alimentatore di corrente; il punto di misura con ID 8984 è il punto di misura da prevedere



Figura 2.13: Boxplot relativo alla distribuzione di corrente dell'alimentatore del sistema con ID S04ML.

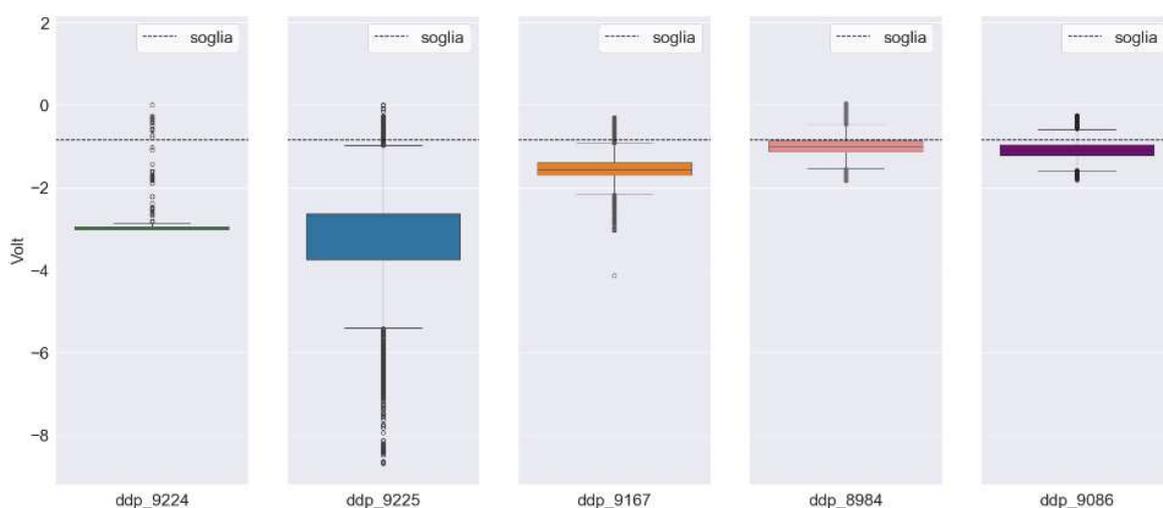


Figura 2.14: Boxplot relativi alle distribuzioni di differenza di potenziale in corrispondenza dell'alimentatore e dei punti di misura telecontrollati del sistema con ID S04ML.

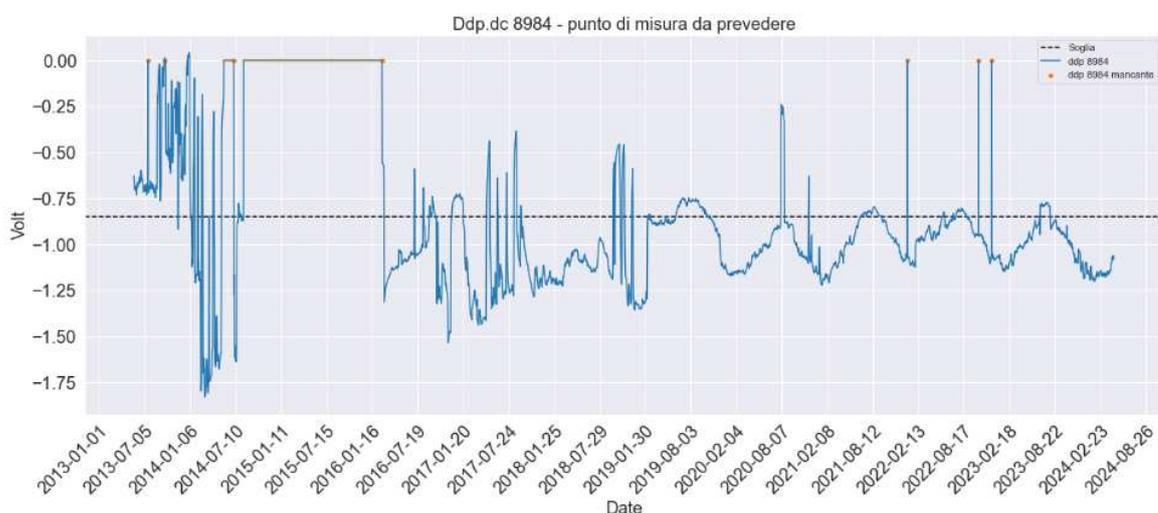


Figura 2.15: L'andamento nel tempo della ddp.dc con ID 8984, il punto di misura da prevedere, per visualizzare i valori mancanti

Le strategie di previsione adottate

In questo capitolo sono illustrate nel dettaglio le metodologie adottate per la previsione della differenza di potenziale dei punti di misura scelti e per la loro valutazione. Gli approcci scelti sono la rolling window e la rete neurale LSTM; il primo permette di ottenere la previsione basata su dati recenti, ricavando un modello ad ogni iterazione, mentre il secondo, sulla base dello storico dati, crea un modello generale della ddp.dc da cui si ottiene la previsione. Tutti algoritmi sono stati implementati in Python, sfruttando la piattaforma Jupyter notebook.

3.1 Il preprocessing dei dati

Un importante aspetto da valutare prima di effettuare la previsione della differenza di potenziale scelta per i sistemi presi in considerazione riguarda la gestione dei dati mancanti, ovvero quelli corrispondenti, sia ai valori *NaN* registrati nel dataset, sia alle righe completamente assenti. I modelli sono stati pensati per lavorare con finestre temporali di giorni consecutivi, pertanto, è stato necessario inserire le righe corrispondenti alle date mancanti, valorizzando a *NaN* i campi di queste. La gestione di tali valori sarà spiegata più dettagliatamente per ciascun approccio nelle prossime sezioni.

3.2 La Rolling window

La *rolling window* (o finestra mobile) [Zivot e Wang, 2006] è una tecnica utilizzata nell'analisi e previsione di serie temporali che sfrutta, principalmente, una semplice operazione chiamata *fit*. Per definire la procedura di *fit*, si considerino m coppie di misure

$$\{(x_1, y_1), (x_2, y_2), \dots, (x_m, y_m)\} \subset \mathbb{R}^p \times \mathbb{R} \quad (3.2.1)$$

che possono essere descritte dalla seguente relazione [Strutz, 2016]:

$$y_i = f(x_i, a) + e_i, \quad i = 1, 2, \dots, m \quad (3.2.2)$$

dove $f(x)$ è la funzione che descrive i dati privi di rumore, a è l'insieme degli p parametri ignoti ed e_1, e_2, \dots, e_m è l'insieme degli errori dovuti a variazioni non controllabili ed imprevedibili della variabile dipendente. L'obiettivo del *fit* è trovare la stima dei parametri a del modello che meglio descrivono la relazione tra le variabili dipendenti (y_i) e le variabili

indipendenti (x_i). Dunque, il fine ultimo è la minimizzazione di una funzione obiettivo (o funzione costo), cioè un criterio di ottimalità che determina quanto $\Gamma(x_i, a)$ si adatta bene ai dati.

La tecnica della *rolling window* consiste nell'esecuzione iterativa di un *fit* su una finestra temporale fissa che scorre progressivamente sui dati e di una previsione di un numero di campioni successivi alla finestra considerata, effettuata a partire dalla funzione trovata all'*i*-esima iterazione. Per avere una visione più chiara di questa metodologie, in Figura 3.1 è riportata la schematizzazione generale in cui si evidenzia come è effettuata la previsione per tre iterazioni. In particolare, dato un vettore di lunghezza T , al passo 0 si considera una finestra temporale di lunghezza fissa k e su di essa si esegue un *fit*, la quale permette di ottenere la funzione matematica che meglio approssima i dati della finestra scelta. Tale funzione è utilizzata per prevedere gli n valori futuri, indicati in figura con il vettore arancione. Al passo successivo, ovvero al passo 1, la finestra temporale scorre di un campione in avanti, l'operazione di *fit* viene rieseguita sulla nuova finestra temporale e, tramite la nuova funzione ottenuta, si esegue la previsione dai campioni $k+2$ ai campioni $k+n+1$.

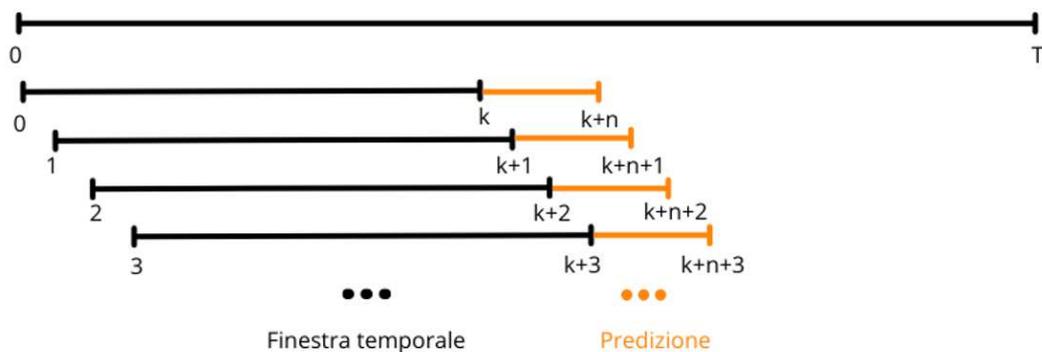


Figura 3.1: Uno schema semplificato del funzionamento della *rolling window* utilizzata per la previsione di serie temporali

In questo elaborato le funzioni utilizzate per effettuare il *fit* su ciascuna finestra temporale sono: *Linear Regression*, *SVR* e *Voting Regressor*. Nelle prossime sezioni saranno descritti più dettagliatamente i suddetti modelli ed il loro specifico utilizzo.

3.2.1 Il modello *Linear regression* (OLS)

La regressione lineare [Rao e Toutenburg, 1995] è una funzione matematica basata sull'equazione di una retta, infatti, date le misure descritte in (3.2.1), la funzione $f(x_i)$, relativa alla (3.2.20), assume la seguente forma:

$$f(x_i) = a_0 + a_1 x_i, \quad i = 1, 2, \dots, m \quad (3.2.3)$$

dove a_0 rappresenta l'intercetta della retta e a_1 il coefficiente che pesa le variabili indipendenti x_i , nonché il coefficiente angolare della retta. L'equazione (3.2.20), diventa:

$$y_i = a_0 + a_1 x_i + e_i, \quad i = 1, 2, \dots, m \quad (3.2.4)$$

L'obiettivo della regressione lineare è trovare la stima dei parametri a_0 e a_1 del modello ed, in questo elaborato, il metodo utilizzato per la loro stima è chiamato *ordinary least squares* (OLS) [Wooditch *et al.*, 2021]. Secondo tale metodo, definendo il *residuo* come la differenza tra i valori osservati y_i e quelli predetti dal modello $f(x_i)$, si vogliono stimare i coefficienti della

funzione in modo tale che la somma dei quadrati dei residui sia minima. In altre parole, il metodo *OLS* minimizza la seguente funzione obiettivo:

$$S(a_0, a_1) = \sum_{i=1}^m (y_i - a_0 - a_1 x_i)^2 = \sum_{i=1}^m (y_i - \hat{y}_i)^2 = \sum_{i=1}^m (e_i)^2 \quad (3.2.5)$$

In particolare, la stima dei coefficienti \hat{a}_1 e \hat{a}_0 è data dalle seguenti relazioni:

$$\hat{a}_1 = \frac{\sum_{i=1}^m (x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^m (x_i - \bar{x})^2} \quad (3.2.6)$$

$$\hat{a}_0 = \bar{y} - \hat{a}_1 \bar{x} \quad (3.2.7)$$

dove:

- $\bar{x} = \frac{1}{m} \sum_{i=1}^m x_i$ è la media dei valori di x_i ,
- $\bar{y} = \frac{1}{m} \sum_{i=1}^m y_i$ è la media dei valori di y_i .

In questo modo è possibile ottenere la stima dei valori predetti dal modello \hat{y}_i , descritta dalla seguente forma:

$$\hat{y}_i = \hat{a}_0 + \hat{a}_1 x_i, \quad i = 1, 2, \dots, m \quad (3.2.8)$$

In Figura 3.2 è riportato un esempio semplificato di regressione lineare che sfrutta il metodo di ottimizzazione *OLS*. I punti blu corrispondono ai campioni y_i osservati e la retta rossa corrisponde alla retta stimata; nell'immagine di sinistra si osservano linee tratteggiate verticali che rappresentano i residui, mentre nell'immagine di destra si osservano quadrati rosso chiaro che rappresentano i quadrati dei residui.

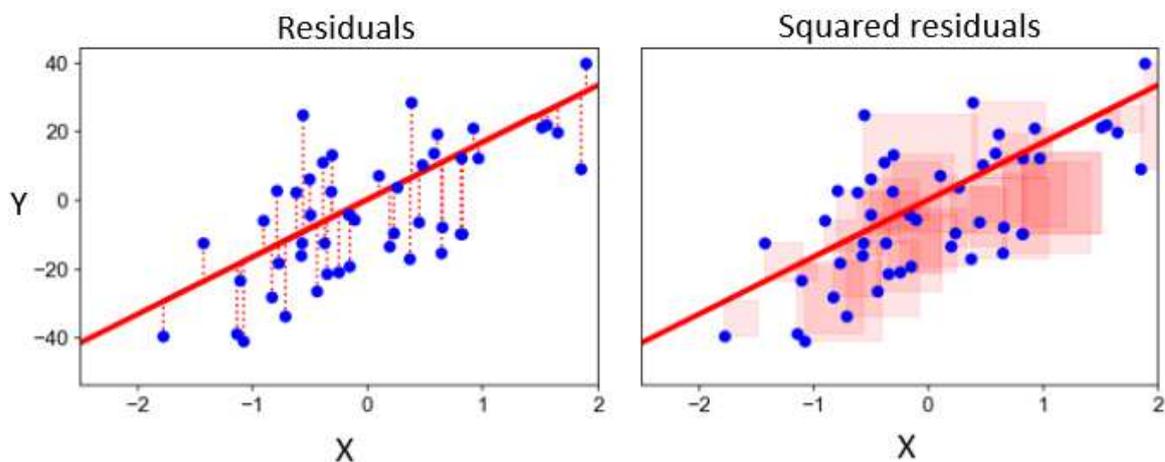


Figura 3.2: Un esempio semplificato della *regressione lineare* con metodo *OLS*

3.2.2 Il modello SVR (ε -insensitive loss)

Il modello di regressione SVR è più complesso rispetto al modello descritto nel sottoparagrafo precedente; per risolvere questo problema, date le misure descritte dalla 3.2.1 e facendo riferimento all'equazione (3.2.20), si vuole determinare una funzione $f(x_i)$ che definisce un iperpiano con la seguente forma:

$$f(x_i) = \langle w, x_i \rangle + b \quad (3.2.9)$$

dove $\langle \cdot, \cdot \rangle$ denota il prodotto interno tra i pesi $w \in \mathbb{R}^p$ ed i campioni $x \in \mathbb{R}^p$, mentre $b \in \mathbb{R}$ rappresenta il bias della funzione.

L'obiettivo della regressione SVR [Drucker *et al.*, 1997] è trovare una $f(x)$ in grado di predire accuratamente i valori delle variabili y_i sulla base delle variabili x_i . Questa funzione è chiamata ε -SVR e deve, sia essere il più piatta possibile, sia avere al massimo una deviazione ε dai valori osservati y_i . In altre parole si vuole minimizzare una funzione obiettivo che riesca a soddisfare entrambe le specifiche, trovandone un compromesso. Il problema di ottimizzazione può essere sintetizzato come segue [Smola e Schölkopf, 2004]:

$$\begin{aligned} \text{Minimize} \quad & \frac{1}{2} \|w\|^2 + C \sum_{i=1}^m (\zeta_i + \zeta_i^*) \\ \text{Subject to} \quad & \begin{cases} y_i - f(x_i) \leq \varepsilon + \zeta_i, \\ f(x_i) - y_i \leq \varepsilon + \zeta_i^*, \\ \zeta_i \geq 0, \zeta_i^* \geq 0. \end{cases} \end{aligned} \quad (3.2.10)$$

Nella funzione da minimizzare, il primo termine vuole soddisfare la specifica di planarità della $f(x)$ andando a minimizzare la norma euclidea dei vettori peso w . Mentre, il secondo termine introduce le cosiddette *variabili di slack* ζ_i e ζ_i^* che rappresentano gli errori, tra la y_i osservata e quella predetta dal modello, maggiori di ε ; in questo caso si vuole minimizzare la somma di tali errori. La costante $C > 0$ determina il compromesso tra la planarità di $f(x)$ ed la penalizzazione degli errori. A questo metodo è possibile associare la cosiddetta ε -insensitive loss function $|\zeta|_\varepsilon$ definita come:

$$|\zeta|_\varepsilon := \begin{cases} 0, & \text{se } |\zeta| \leq \varepsilon, \\ |\zeta| - \varepsilon, & \text{altrimenti.} \end{cases} \quad (3.2.11)$$

In figura 3.3 (a) è riportato un esempio semplificato di *linear SVR* in cui è evidenziata in rosso la funzione che rappresenta l'iperpiano e le linee tratteggiate che determinano il margine di tolleranza oltre il quale l'errore è penalizzato come ζ , quindi sarà da minimizzare. Invece, in figura 3.3 (b) è riportata la ε -insensitive loss function, tale per cui, per i campioni che risultano all'interno del margine creato da ε , la loss risulta pari a 0, mentre, per i campioni che risultano all'esterno del margine creato da ε , la loss risulta pari alla differenza tra il modulo di ζ ed ε .

Il problema di ottimizzazione riportato in (3.2.10) può essere risolto più facilmente nella sua formulazione duale, per cui il problema assume la seguente forma:

$$\begin{aligned} \text{Maximize} \quad & \begin{cases} -\frac{1}{2} \sum_{i,j=1}^m (\alpha_i - \alpha_i^*)(\alpha_j - \alpha_j^*) \langle \mathbf{x}_i, \mathbf{x}_j \rangle \\ -\varepsilon \sum_{i=1}^m (\alpha_i + \alpha_i^*) + \sum_{i=1}^m y_i (\alpha_i - \alpha_i^*) \end{cases} \\ \text{Subject to} \quad & \sum_{i=1}^m (\alpha_i - \alpha_i^*) = 0, \quad 0 \leq \alpha_i, \alpha_i^* \leq C. \end{aligned} \quad (3.2.12)$$

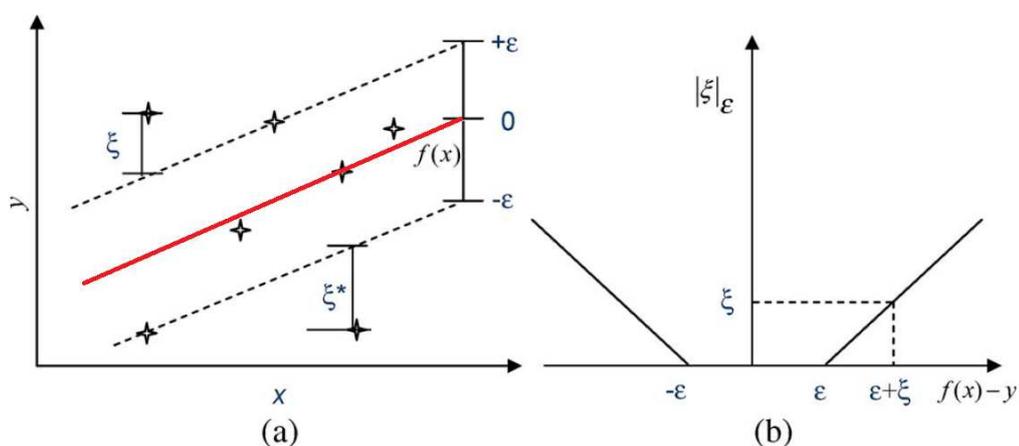


Figura 3.3: (a) L'impostazione della loss per una *linear SVR* - (b) la ϵ -insensitive loss function [Bi et al., 2011]

dove α_i e α_i^* sono i moltiplicatori di Lagrange dell' i -esimo campione osservato; inoltre, $(\alpha_i - \alpha_i^*)$ sono diversi da zero in corrispondenza delle variabili x_i , chiamate *Support Vectors*, e la loro combinazione determina i pesi w :

$$w = \sum_{i=1}^m (\alpha_i - \alpha_i^*) x_i. \tag{3.2.13}$$

Sfruttando l'equazione 3.2.9 e 3.2.13 è possibile riscrivere la funzione dell'iperpiano:

$$f(x_i) = \sum_{i=1}^m (\alpha_i - \alpha_i^*) \langle x, x_i \rangle + b \tag{3.2.14}$$

Tuttavia, nella maggior parte dei problemi reali, la relazione tra le variabili x_i e y_i è non lineare e questo rende necessario mappare le variabili x_i in uno spazio più complesso. Si consideri la funzione:

$$\phi(\cdot) = \mathbb{R}^p \rightarrow \mathbb{R}^Q \tag{3.2.15}$$

che denota una trasformazione non lineare, la quale permette di mappare il vettore degli $x_i \in \mathbb{R}^p$ in uno spazio Q -dimensionale, lo spazio delle features. Tramite il cosiddetto *kernel trick* è possibile trasformare le variabili x_i nello spazio \mathbb{R}^Q senza utilizzare la funzione $\phi(\cdot)$, in modo da semplificare l'elaborazione. Dunque, si definisce la *funzione kernel* $K(\cdot, \cdot)$ come segue [Cortes e Vapnik, 1995]:

$$K(x, x_i) = \langle \phi(x), \phi(x_i) \rangle \tag{3.2.16}$$

In questo elaborato, la funzione kernel applicata è la *gaussian radial basis function (RBF)* che assume la seguente forma:

$$K(x, x_i) = \exp\left(-\frac{\|x - x_i\|^2}{2\sigma^2}\right) \tag{3.2.17}$$

dove $\|x - x_i\|^2$ rappresenta la distanza euclidea al quadrato tra le coppie di vettori appartenenti ad \mathbb{R}^p e σ rappresenta un parametro libero positivo, utilizzato per l'ottimizzazione della funzione.

Una volta definita la funzione kernel, è possibile riscrivere l'equazione (3.2.9) secondo la relazione:

$$f(x_i) = \sum_{i=1}^m (a_i - a_i^*) K(x, x_k) + b \quad (3.2.18)$$

ed il problema di regressione si riduce all'individuazione di $(\alpha_i - \alpha_i^*)$.

3.2.3 Il modello *Voting regressor*

Il modello *Voting regressor* [Zhou, 2012] permette di combinare modelli di regressione differenti; dato un insieme T di funzioni f_1, f_2, \dots, f_T , la previsione di ciascun modello è descritta dalla suddetta relazione:

$$y_i(x) = f_i(x) + e_i, \quad i = 1, 2, \dots, T \quad (3.2.19)$$

Combinando ciascuna previsione si ottiene l'*ensemble model*. In questo elaborato è stato sfruttato il metodo della media pesata delle previsioni ottenute dai due modelli descritti nei paragrafi precedenti: *Linear regression* e *SVR*. La media pesata fornisce la seguente previsione:

$$Y(x) = \sum_{i=1}^T w_i f_i(x) \quad (3.2.20)$$

dove T è pari a 2, f_i rappresenta il risultato dei singoli *fit* ottenuti con i suddetti regressori e w_i descrivono i pesi che determinano l'importanza di un modello rispetto all'altro. Inoltre, si specifica che la *Linear regression* e l'*SVR* sfruttano separatamente le proprie funzioni di loss (rispettivamente *OLS* e ϵ -insensitive loss) per ottenere la funzione che meglio descrive la relazione tra x_i e y_i , variabili definite in (3.2.1). Utilizzando un *Voting Regressor*, in questo caso specifico, è possibile ottenere una previsione basata sulla combinazione di un modello lineare ed uno non lineare.

3.2.4 La gestione dei dati e dei valori *NaN*

In questo elaborato la *rolling window* è stata eseguita su tutta la lunghezza della serie temporale presa in esame ed essendo un approccio locale, quindi senza memoria, ad ogni iterazione, viene costruita una nuova funzione tramite cui si effettua la previsione. Dopo aver scelto la finestra temporale su cui eseguire il *fit*, è stato effettuato un controllo sulla presenza o meno dei valori *NaN* della finestra stessa per, eventualmente, gestirli. Se tali valori sono presenti e la loro somma non supera una percentuale di tolleranza scelta in base alla lunghezza della serie temporale, si effettua una interpolazione lineare. In particolare, l'interpolazione viene eseguita all'indietro con un limite di 2 campioni e questo comporta che:

- se sono presenti più di 2 valori *NaN* consecutivi, l'interpolazione avviene solo per i 2 campioni più recenti;
- se il valore finale della finestra temporale è *NaN*, nonché il valore precedente al giorno in cui si effettua la predizione, non è soggetto all'interpolazione;
- se sono presenti più di 2 valori *NaN* consecutivi e uno di questi è l'ultimo valore della finestra temporale, ovvero il giorno precedente al giorno in cui si effettua la predizione, allora nessuno di essi è soggetto all'interpolazione.

Per confrontare il valore predetto con il valore reale è stato fondamentale gestire i valori *NaN* del set di dati predetto. Se tra i dati reali e predetti dal modello sono presenti dei valori *NaN*, si effettua una interpolazione in avanti con un limite di 1 campione e questo comporta che:

- se sono presenti più valori *NaN* consecutivi, l'interpolazione avviene solo per il campione più vicino al giorno in cui avviene la predizione;
- se il primo valore reale da predire è *NaN*, questo non è soggetto all'interpolazione.

Successivamente, si effettua un check per verificare che i valori da predire non abbiano valori *NaN*, se questi sono presenti, il confronto tra il valore reale e il valore predetto non avviene.

3.3 La rete neurale LSTM

Una rete neurale è un modello matematico che vuole emulare la struttura ed il funzionamento di una rete neurale biologica. Questa si compone dei cosiddetti *neuroni* o *nodi formali* che, interconnessi, formano un grafo costituito da tre strati: *input layer*, *hidden layer* ed *output layer*. Le *recurrent neural network* (RNN) sono di fatto delle reti multistrato, in cui i segnali che fuoriescono dai nodi di uno strato di livello superiore diventano input per gli strati di livello inferiore.

La LSTM (*Long Short-Term Memory*) [Kumar *et al.*, 2022] è un tipo di rete (RNN), infatti, a differenza delle reti *feed-forward*, i suoi neuroni sono in grado di fornire dei feedback. La sua architettura è in grado di elaborare dati sequenziali e desta particolare interesse per la capacità di modellare dipendenze temporali complesse e di gestire la memoria a lungo termine; infatti, risulta particolarmente utile per la previsione di serie temporali. Per tale ragione, per il raggiungimento dello scopo di tesi, si è fatto ricorso ad un modello di questo tipo per la previsione di più valori in avanti

3.3.1 Il funzionamento della rete LSTM

La struttura interna di LSTM è costituita da **gates** moltiplicativi che regolano il flusso di informazioni, ovvero il modo in cui vengono immesse nella rete, archiviate ed infine rilasciate. Inoltre, è costituita da una **memory cell** tramite cui i gates decidono quale informazione memorizzare e quale dimenticare. In Figura 3.4 è riportata l'architettura della rete neurale LSTM in cui si evidenziano il *forget gate* (f_t), l'*input gate* (i_t) e l'*output gate* (o_t).

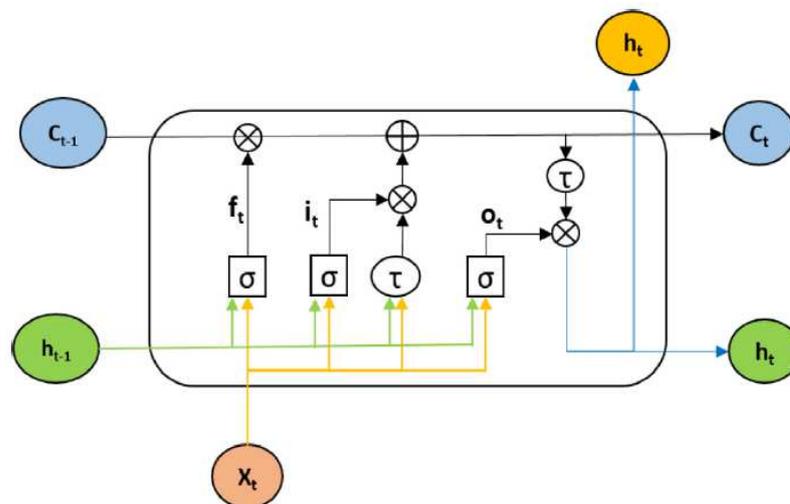


Figura 3.4: L'architettura della rete LSTM [Islam *et al.*, 2019]

Il **forget gate** controlla quali elementi dello stato della cella (memoria a lungo termine) sono rilevanti in base allo stato nascosto precedente h_{t-1} ed ai nuovi dati di input (x_t). Il suo output può essere descritto tramite la seguente equazione:

$$f_t = \sigma (X_t U_f + h_{t-1} W_f + b_f) \quad (3.3.1)$$

dove W_f e U_f sono le matrici dei pesi del *forget gate* f_t , b_f è il valore del bias del *forget gate* e σ la funzione di attivazione sigmoide. L'uscita dal *forget gate* corrisponde ad un vettore per cui ogni elemento è un valore compreso tra 0 e 1; se tale valore prodotto è vicino a 0, le informazioni sono considerate irrilevanti, mentre se è vicino ad 1, le informazioni sono considerata rilevanti. Questi valori di output sono, successivamente, moltiplicati, elemento per elemento, con lo stato della cella precedente (C_{t-1}) e questo comporta che le parti irrilevanti dello stato della cella sono sotto-pesate di un fattore vicino a 0, riducendo la loro influenza sui passaggi successivi.

L'**input gate** decide quali informazioni saranno aggiunte alla cella di memoria ed è composto da un strato σ e da uno strato \tanh . Il primo strato filtra le informazioni del nuovo vettore di memoria ed è descritta dall'equazione:

$$i_t = \sigma (X_t U_i + h_{t-1} W_i + b_i) \quad (3.3.2)$$

dove W_i e U_i rappresentano le matrici dei pesi dell'*input gate* i_t , \tanh la funzione di attivazione e b_i il valore del bias dell'*input gate*

Mentre, il secondo genera un nuovo vettore di valori (\tilde{C}_t) da aggiungere in memoria ed è descritta dall'equazione:

$$\tilde{C}_t = \tanh (X_t U_c + h_{t-1} W_c + b_c) \quad (3.3.3)$$

dove W_c e U_c rappresentano le matrici dei pesi della nuova cella di memoria \tilde{C}_t , \tanh la funzione di attivazione e b_c è il valore del bias della nuova cella di memoria \tilde{C}_t . In particolare la \tanh è utilizzata perché i suoi valori sono compresi nell'intervallo $[-1,1]$ e la capacità di produrre valori negativi è essenziale per ridurre l'influenza di un componente nello stato della cella. L'output dell'*input gate* è dato dalla seguente combinazione:

$$C_t = f_t \otimes C_{t-1} \oplus i_t \otimes \tilde{C}_t \quad (3.3.4)$$

L'**output gate** stabilisce quale parte della memoria contribuisce all'output della rete, in particolare, quali elementi dello stato della cella aggiornato sono rilevanti e dovrebbero essere emessi come nuovo stato nascosto. L'input dell'*output gate* è descritto dalla seguente relazione:

$$o_t = \sigma (X_t U_o + h_{t-1} W_o + b_o) \quad (3.3.5)$$

Mentre, il nuovo stato nascosto finale è dato dall'equazione:

$$h_t = o_t \otimes \tanh (C_t) \quad (3.3.6)$$

Per addestrare la rete su di un set di dati (*training set*), il modello calcola il valore della *loss* confrontando il risultato ottenuto con quello desiderato e sfrutta, successivamente, la *backpropagation* per l'aggiornamento dei pesi della rete tramite la discesa del gradiente. Tutti questi passaggi sono iterati per un numero finito di epoche ed una volta terminato l'addestramento, il modello viene testato su di un dataset che non ha visto durante il train (*test set*), in modo da valutare la sua capacità di generalizzazione.

3.3.2 La gestione dei dati e dei valori *NaN*

Definendo come X i predittori e come Y il target, affinché la rete venga addestrata e testata, è necessario che questi dati siano predisposti in *sequenze*. In generale, per poter creare un dataset X e Y si sceglie, innanzitutto, una finestra temporale di lunghezza fissa K ed un numero N di giorni corrispondenti alla variabile target che si vuole prevedere. Si crea, poi, simultaneamente, la prima sequenza di X e la prima sequenza di Y . Il primo elemento di X è una matrice che avrà tante righe quante sono i predittori in input al modello (se il modello è univariabile la feature è una sola, altrimenti il modello è multivariabile) e tante colonne quanti sono i giorni della finestra temporale che la rete vede per effettuare la predizione. Mentre, il primo elemento di Y ha tanti campioni della variabile target quanti sono i giorni che il modello deve essere in grado di prevedere. Successivamente, la finestra temporale scorre di un elemento sul dataset iniziale e si costruiscono le rispettive seconde sequenze di X e Y , che saranno concatenate nei rispettivi dataset. Questo processo viene iterato per tutta la lunghezza del *train set* e *test set*.

In Figura 3.5 e 3.6 è schematizzato il processo di creazione delle prime due sequenze dei dataset X e Y per un modello multivariabile che vuole prevedere N campioni in avanti. In ciascuna immagine, sulla sinistra, è raffigurato il dataset iniziale composto da 2 features, $F1$ e $F2$; in questo esempio, la variabile che si vuole predire è $F1$. Nella prima immagine è illustrato il passaggio dal dataset iniziale alle rispettive prime sequenze dei dataset X e Y . Nella seconda immagine si mostra il passaggio dal dataset iniziale alle rispettive seconde sequenze dei dataset di X e Y .

Dunque, i dataset X e Y vengono creati simultaneamente, tramite la concatenazione degli elementi opportuni, ma, prima che gli elementi siano inseriti nei dataset rispettivi, in ciascuno di essi, si è voluto verificare la presenza o meno di campioni *NaN*. Se tali valori non sono presenti, gli elementi x e y vengono aggiunti ai dataset rispettivi e si prosegue analizzando la nuova finestra temporale, altrimenti, devono sottoporsi ad alcuni check. Se l'elemento x presenta valori *NaN* e la loro somma non supera una percentuale di tolleranza scelta in base alla lunghezza della serie temporale, si effettua una interpolazione lineare. In particolare, l'interpolazione viene eseguita all'indietro con un limite di 2 campioni. Invece, se l'elemento y presenta valori *NaN*, si effettua una interpolazione in avanti con un limite di 1 campione. Successivamente, si verifica che, sia per x che per y , la finestra non abbia più valori *NaN*; se sono presenti, X e Y non saranno aggiunti rispettivamente ai dataset X e Y e, di conseguenza, non saranno considerati per la creazione del modello.

3.4 La valutazione dei modelli

Per valutare la bontà di previsione, sia per l'approccio tramite *rolling window* che per l'approccio basato su rete neurale *LSTM*, sono state utilizzate delle metriche. In particolare, si è scelto di valutare queste misure per ciascun giorno predetto, in modo da poter avere un'idea sull'incertezza della predizione di ciascun campione. Il calcolo della metrica generale è stato escluso in quanto questa avrebbe comportato l'analisi di dati ridondanti dato che ciascuno degli n giorni da predire risulta predetto n volte. Dunque, per il calcolo delle metriche è stato effettuato un raggruppamento degli n valori predetti dai modelli secondo la distanza dal giorno in cui è effettuata la predizione: il primo giorno di predizione avrà distanza 1 e l'ultimo distanza n . In Figura 3.7 è illustrato uno schema semplificato per la valutazione delle metriche; data una serie temporale di lunghezza T , $p+1$ finestre temporali che scorrono lungo tutta la serie, ciascuna di lunghezza fissa k ed n giorni di previsione, si evidenzia, ad ogni iterazione, in rosso il primo giorno di previsione ed in blu l'ultimo.

Le metriche principalmente adottate sono *MAE* ed R^2 [Johnson e Kuhn, 2013].

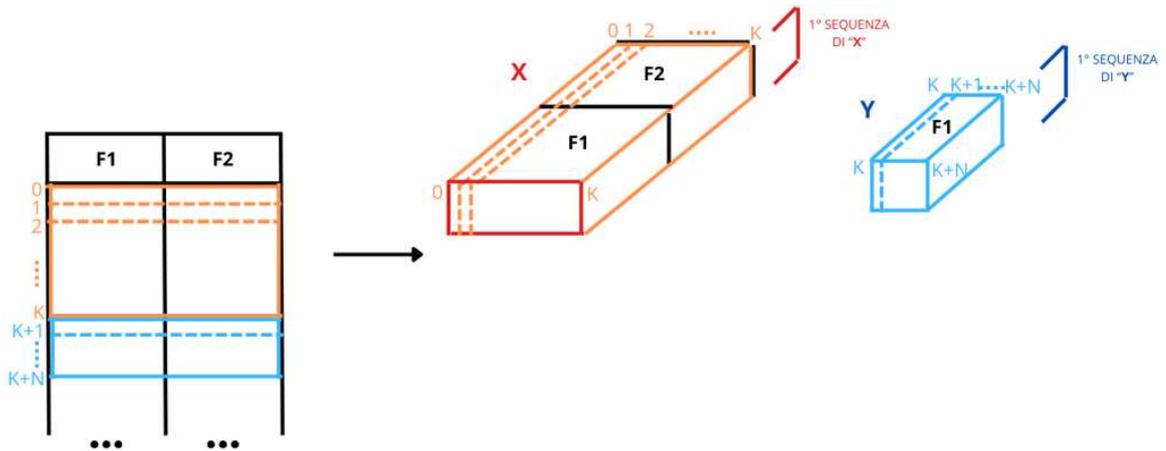


Figura 3.5: Schema illustrativo del passaggio dai primi $K+N$ elementi del dataset iniziale nella prima sequenza del dataset X e nella prima sequenza del dataset Y .

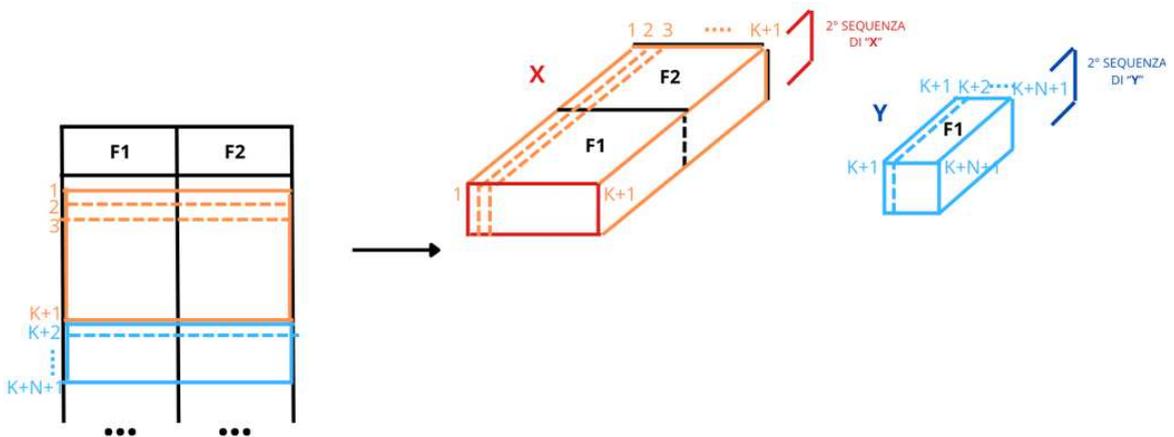


Figura 3.6: Schema illustrativo dello shift della finestra temporale di lunghezza fissa K sul dataset iniziale ed il passaggio dai $K+N$ elementi del suddetto dataset nella seconda sequenza del dataset X e nella seconda sequenza del dataset Y .

- **Mean absolute error (MAE):** è la media di quanto le previsioni differiscono dai valori reali, senza considerare se l'errore è positivo o negativo. La ricerca del modello perfetto conduce inevitabilmente a trovare il valore di MAE più basso possibile. Il grande vantaggio di questa metrica è la sua facile interpretazione, nonché la sua resistenza agli outliers. Facendo riferimento all'esempio generale in Figura 3.7, l'errore medio per il primo e l'ultimo giorno di previsione si può descrivere tramite le seguenti equazioni:

$$MAE_1 = \frac{\sum_{i=k+1}^{k+p+1} |y_i - \hat{y}_i|}{k + p + 1} \quad (3.4.1)$$

$$MAE_n = \frac{\sum_{i=k+n}^T |y_i - \hat{y}_i|}{T} \quad (3.4.2)$$

dove y_i è il valore i -esimo effettivo e \hat{y}_i è il valore i -esimo previsto.

- **Coefficiente di determinazione (R^2 o R - squared):** può essere interpretato come la proporzione della varianza della variabile dipendente che è prevedibile dalle variabili

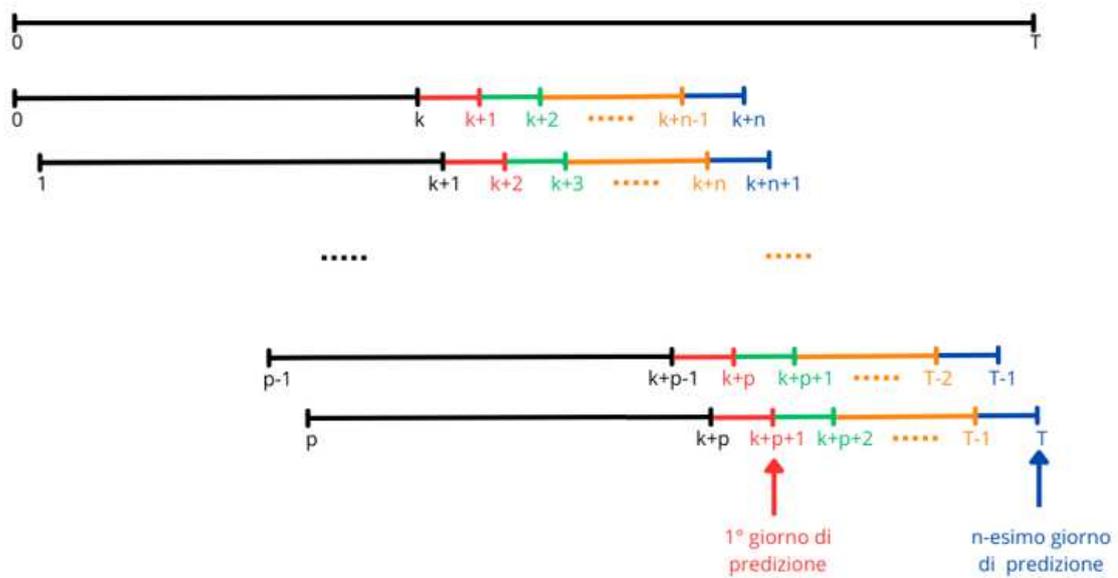


Figura 3.7: Schema semplificato per la valutazione delle metriche

indipendenti. Questa metrica indica quanto bene il modello riesce a spiegare i dati; un valore di R^2 vicino a 1 significa che il modello spiega bene la variabilità dei dati, mentre un valore vicino a 0 significa che il modello non spiega bene i dati. Facendo riferimento all'esempio generale in Figura 3.7, il coefficiente di determinazione per il primo e l'ultimo giorno di previsione si può descrivere tramite le seguenti equazioni:

$$R_1^2 = 1 - \frac{\sum_{i=k+1}^{k+p+1} (y_i - \hat{y}_i)^2}{\sum_{i=k+1}^{k+p+1} (y_i - \bar{y})^2} \quad (3.4.3)$$

$$R_n^2 = 1 - \frac{\sum_{i=k+n}^T (y_i - \hat{y}_i)^2}{\sum_{i=k+n}^T (y_i - \bar{y})^2} \quad (3.4.4)$$

dove y_i è il valore i -esimo effettivo, \hat{y}_i è il valore i -esimo previsto e \bar{y} è il valor medio dei valori effettivi.

Una volta individuato il modello migliore tramite le suddette valutazioni, per avere un'ulteriore verifica si è voluto analizzare la sua capacità di riconoscere, in questo caso specifico, i campioni fuori soglia. Per tale ragione è stata effettuata una classificazione del dataset reale e predetto, assegnando il valore 0 ai dati sottosoglia e il valore 1 ai dati fuori soglia. Sono state calcolate, su ciascun giorno di predizione, le seguenti metriche :

- **Accuracy:** misura la proporzione di campioni classificati correttamente sul totale dei campioni.

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}}$$

- **Precision:** misura la proporzione di campioni correttamente classificati come positivi sul totale dei campioni classificati come positivi.

$$\text{Precision} = \frac{TP}{TP + FP}$$

- **Recall (o Sensitivity):** misura la proporzione di campioni positivi correttamente identificati.

$$\text{Recall} = \frac{TP}{TP + FN}$$

- **F1-Score:** rappresenta la media armonica di Precision e Recall.

$$F1 = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}$$

Dove:

- TP (True Positives): campioni positivi correttamente classificati.
- TN (True Negatives): campioni negativi correttamente classificati.
- FP (False Positives): campioni negativi classificati erroneamente come positivi.
- FN (False Negatives): campioni positivi classificati erroneamente come negativi.

Riguardo queste ultime metriche, in tesi sono stati presentati soltanto i risultati ottenuti relativi all'ultimo giorno di previsione ed è stata visualizzata la relativa confusion matrix, per valutare quanto bene il modello riesce a distinguere tra le diverse classi da prevedere.

L'implementazione ed i risultati della *rolling window*

In questo capitolo è illustrata, innanzitutto, l'implementazione della rolling window che è stata eseguita in Python, sfruttando la piattaforma Jupyter Notebook; nei paragrafi successivi sono mostrati i principali risultati ottenuti dalla predizione della ddp.dc con ID 17271 del sistema MV11 e della ddp.dc con ID 8984 del sistema S04ML. Infine, è stato effettuato un confronto per la valutazione complessiva della metodologia utilizzata.

4.1 L'implementazione della *rolling window*

Per implementare la *rolling window*, innanzitutto, è stata scelta la lunghezza fissa della finestra temporale scorrevole su cui effettuare il *fit*, la percentuale accettata di valori *NaN* dei suddetti campioni per la loro gestione (spiegata nel sottoparagrafo 3.2.4) ed il numero di giorni da prevedere. Questo approccio è stato sperimentato su tre finestre di lunghezza differente: 45, 30 e 15 campioni; per ciascuna di esse, è stata scelta una percentuale di tolleranza di valori *NaN* che può essere presente tra i campioni stessi: 12% per la finestra da 45, 15% per la finestra da 30 e 20% per la finestra da 15. Il numero di campioni che si è deciso di prevedere è pari a 5.

Come già anticipato nel capitolo precedente, per ciascun esperimento, sono stati testati tre modelli per l'esecuzione del *fit* implementati mediante la libreria di *scikit-learn* di Python: *Linear regression*, *SVR* e *Voting Regressor*. In particolare, l'*SVR* è stato scelto con kernel di tipo radiale, per gestire efficacemente la non linearità dei dati ed un ϵ pari a 0.03, più piccolo rispetto al suo valore di default (0.1), in modo da avere una penalizzazione più alta dell'errore, come spiegato nel sottoparagrafo 3.2.2. Per bilanciare maggiormente la linearità e la non linearità dei dati, è stato implementato il *VotingRegressor*, combinando la *linear regression* e l'*SVR*, dando loro, rispettivamente, un peso del 30% e 70%. Le *loss function* utilizzate per l'ottimizzazione sono: l'*OLS* per la *Linear regression*, l' ϵ -insensitive loss per l'*SVR* e la combinazione di queste per il *Voting Regressor*.

Dunque, fissata una finestra temporale ed un modello, è stato implementato un ciclo for tale per cui, ad ogni iterazione, in seguito all'eventuale gestione dei valori *NaN*, è eseguito un *fit* per l'individuazione della funzione che meglio approssima i valori di differenza di potenziale nella finestra temporale scelta. Tale modello è stato poi utilizzato per la previsione dei 5 giorni successivi della stessa ddp.dc, effettuando, quindi, un "prolungamento" della funzione trovata; in questo approccio, quindi, il predittore è rappresentato dalla finestra temporale di ddp.dc corrente ed il target è rappresentato dai 5 giorni successivi alla finestra

stessa. La previsione è stata effettuata lungo tutta la serie temporale, sfruttando, appunto, il concetto di rolling window; tale approccio, infatti, è locale in quanto, ad ogni iterazione, è eseguito un nuovo *fit* ed una nuova predizione.

4.2 I risultati della predizione

Le differenze di potenziale scelte per effettuare la previsione sono la ddp.dc corrispondente al punto con *ID 17271* del sistema *MV11* e la ddp.dc corrispondente al punto con *ID 8984* del sistema *S04ML*. La prima differenza di potenziale registra 2.113 campioni, tra il 2018-06-14 e il 2024-04-18, mentre, la seconda registra 3.377 campioni, 2013-05-23 e il 2024-04-18. Come già anticipato, per il raggiungimento dell'obiettivo, sono stati sfruttati, separatamente, tutti i dati disponibili per ciascuna ddp.dc scelta. Nei prossimi sotto-pragrafi sono riportati i risultati ottenuti dalle metriche su ciascuna finestra temporale e funzione di regressione scelta.

4.2.1 La predizione della ddp.dc - *ID 17271*

Di seguito sono riportati sei istogrammi che descrivono, a coppie, i risultati dei 5 giorni predetti, rispetto alle tre finestre temporali scelte e a ciascun modello di regressione utilizzato. In Figura 5.1 e 5.2 sono riportati i risultati delle metriche *MAE* ed R^2 considerando la finestra di 45 giorni, in Figura 4.3 e 4.4 sono riportati i risultati delle metriche *MAE* ed R^2 considerando la finestra di 30 giorni ed, infine, in Figura 4.5 e 4.6 sono riportati i risultati delle metriche *MAE* ed R^2 considerando la finestra di 15 giorni.

Come ci si aspetterebbe, fissata una qualsiasi finestra temporale ed un qualsiasi modello utilizzato per il *fit*, si evidenzia come più ci si allontana dall'istante di predizione e più l'errore di stima aumenta, infatti, la previsione del campione nel giorno 1 ha un *MAE* più basso ed un R^2 più alto rispetto al giorno 5. Fissata una qualsiasi finestra temporale si osserva che il *Linear regression* utilizzato per il *fit* è quello che ha le prestazioni peggiori, sebbene l'errore di previsione è nell'ordine del centesimo. Il *VotingRegressor* rispetto all'*SVR* ha tendenzialmente un *MAE* inferiore ed un R^2 maggiore per ciascun giorno di previsione e questa differenza si nota prevalentemente nel terzo, quarto e quinto giorno di previsione. Confrontando le metriche ottenute dai rispettivi modelli su una finestra temporale di 45 e 15 giorni, la previsione acquisita con la finestra più grande è peggiore rispetto a quella ottenuta con la finestra più piccola. Inoltre, più la lunghezza della finestra diminuisce e maggiore è la similarità tra l'*SVR* ed il *VotingRegressor*, ma se i dati su cui effettuare il *fit* sono troppo pochi, le performance risultano leggermente peggiori.

Come ulteriore analisi si è voluto studiare la distribuzione data dalla differenza tra il valore reale e quello predetto ed a tal proposito è stato graficato un boxplot dell'errore. In Figura 4.7 è riportato un boxplot relativo alla finestra 45, in Figura 4.8 un boxplot relativo alla finestra 30 ed in Figura 4.9 un boxplot relativo alla finestra 15; per semplicità, in ognuno di questi sono stati eliminati gli outlier. Il valore centrale della distribuzione (la mediana) si mantiene intorno allo zero in tutti i giorni predetti su ciascuna finestra e modello utilizzata; questo conferma il perché i valori della metrica *MAE* risultano così bassi e coerentemente con i grafici precedenti, più ci si allontana dal giorno in cui si effettua la predizione e più la distribuzione dell'errore aumenta. Infatti, tra il primo ed il quinto giorno la variabilità centrale dell'errore aumenta (la differenza tra il primo ed il terzo quartile, nonché la cosiddetta *scatola* del boxplot), come anche i minimi ed i massimi assunti. Anche osservando l'errore risulta evidente come il *Voting Regressor* su una finestra di 15 campioni risulta il modello migliore tra tutti.

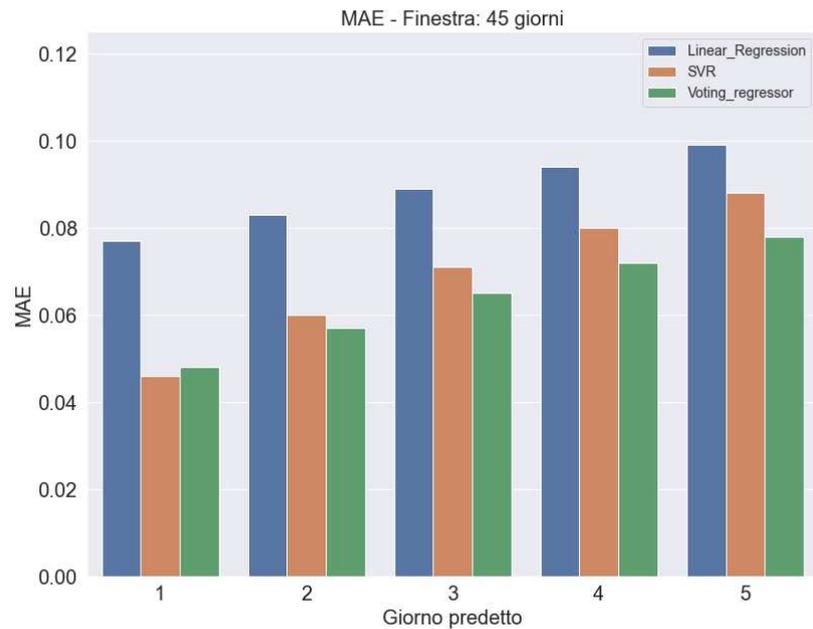


Figura 4.1: L'istogramma che illustra il MAE calcolato su ciascun giorno predetto per la ddp.dc con ID 17271, rispetto a ciascun modello utilizzato, su di una finestra fissa scorrevole di 45 campioni

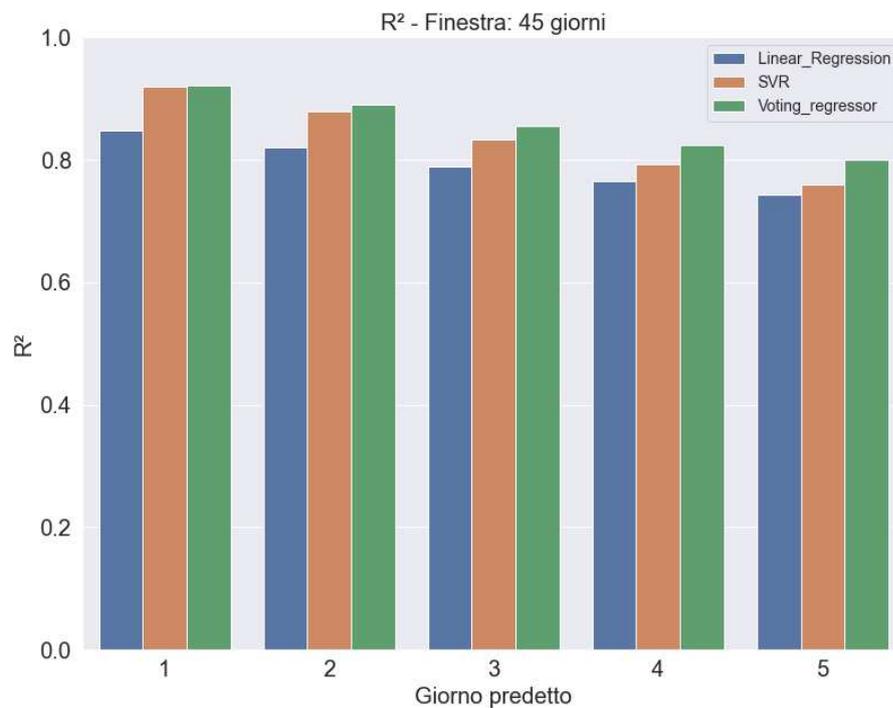


Figura 4.2: L'istogramma che illustra l' R^2 calcolato su ciascun giorno predetto per la ddp.dc con ID 17271, rispetto a ciascun modello utilizzato, su di una finestra fissa scorrevole di 45 campioni

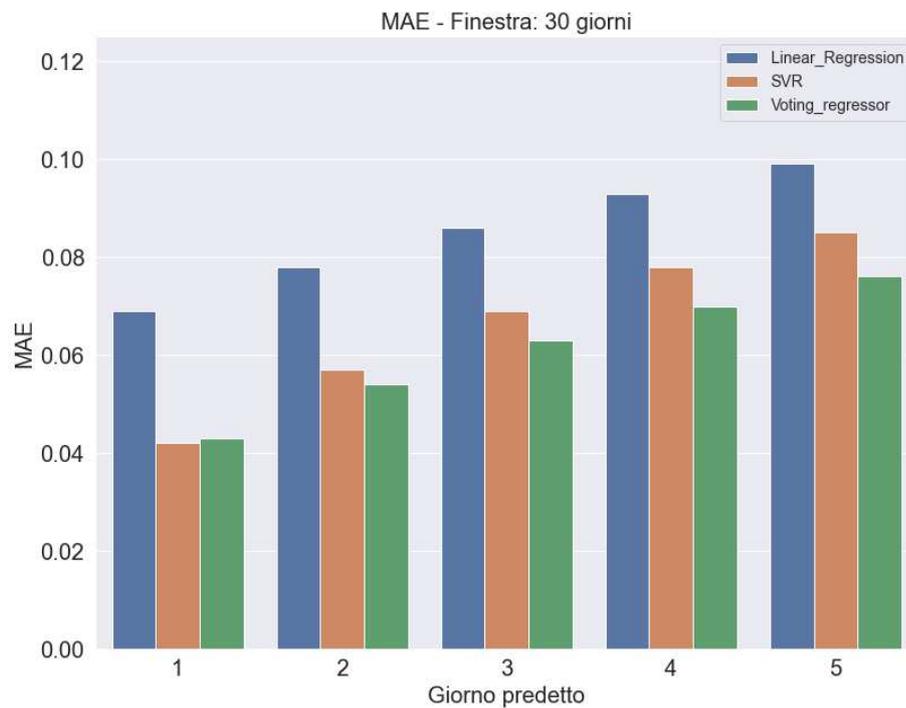


Figura 4.3: L'istogramma che illustra il MAE calcolato su ciascun giorno predetto per la ddp.dc con ID 17271, rispetto a ciascun modello utilizzato, su di una finestra fissa scorrevole di 30 campioni

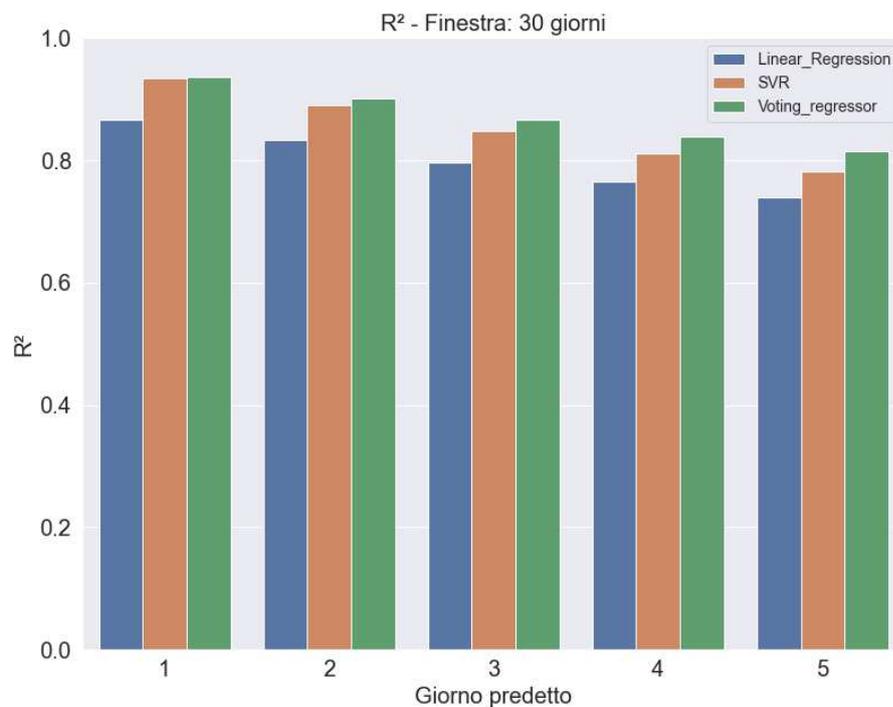


Figura 4.4: L'istogramma che illustra l'R² calcolato su ciascun giorno predetto per la ddp.dc con ID 17271, rispetto a ciascun modello utilizzato, su di una finestra fissa scorrevole di 30 campioni

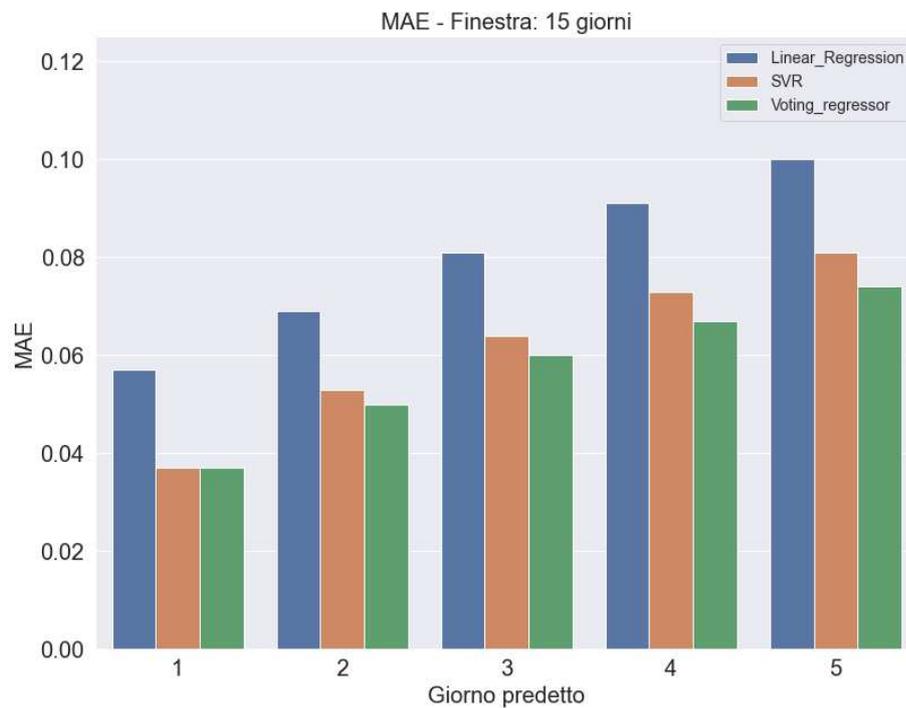


Figura 4.5: L'istogramma che illustra il MAE calcolato su ciascun giorno predetto per la ddp.dc con ID 17271, rispetto a ciascun modello utilizzato, su di una finestra fissa scorrevole di 15 campioni

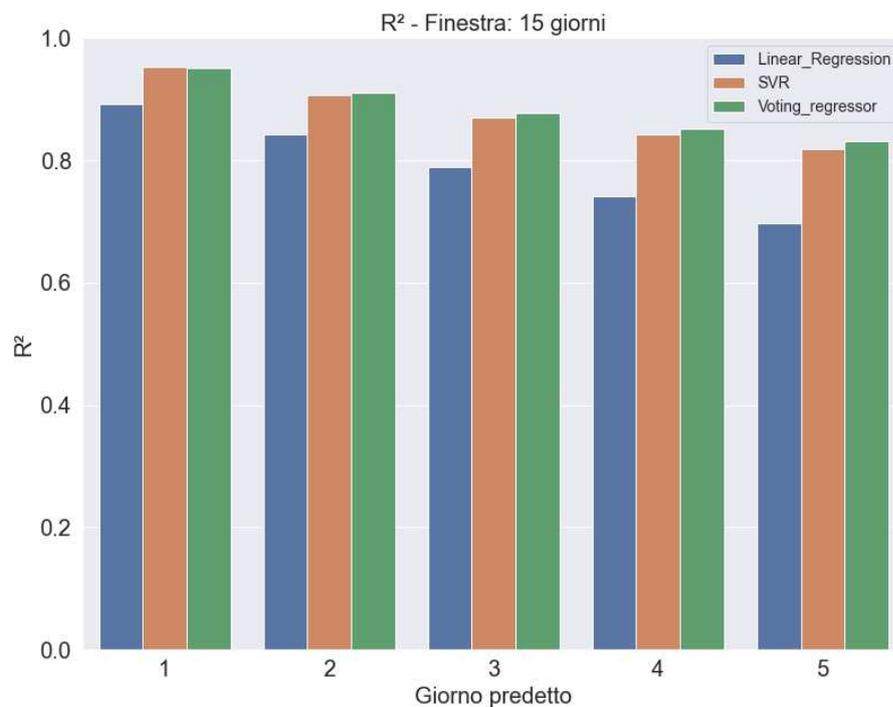


Figura 4.6: L'istogramma che illustra l' R^2 calcolato su ciascun giorno predetto per la ddp.dc con ID 17271, rispetto a ciascun modello utilizzato, su di una finestra fissa scorrevole di 15 campioni

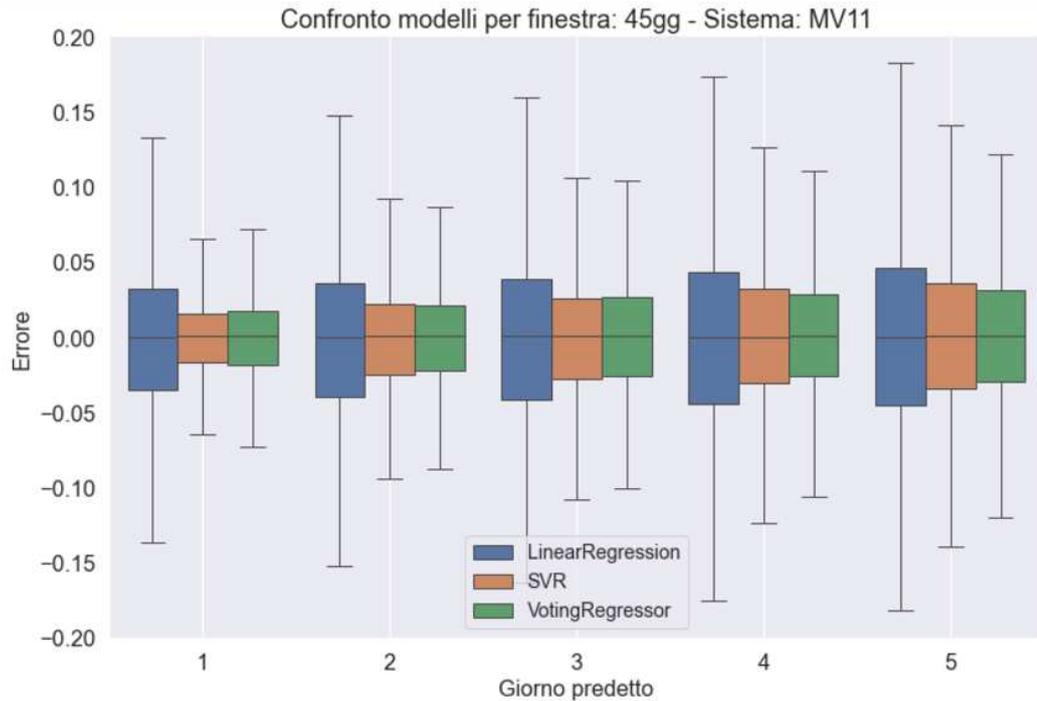


Figura 4.7: Il boxplot che illustra l'errore di previsione commesso su ciascun giorno predetto per la ddp.dc con ID 17271, rispetto a ciascun modello utilizzato, su di una finestra fissa scorrevole di 45 campioni

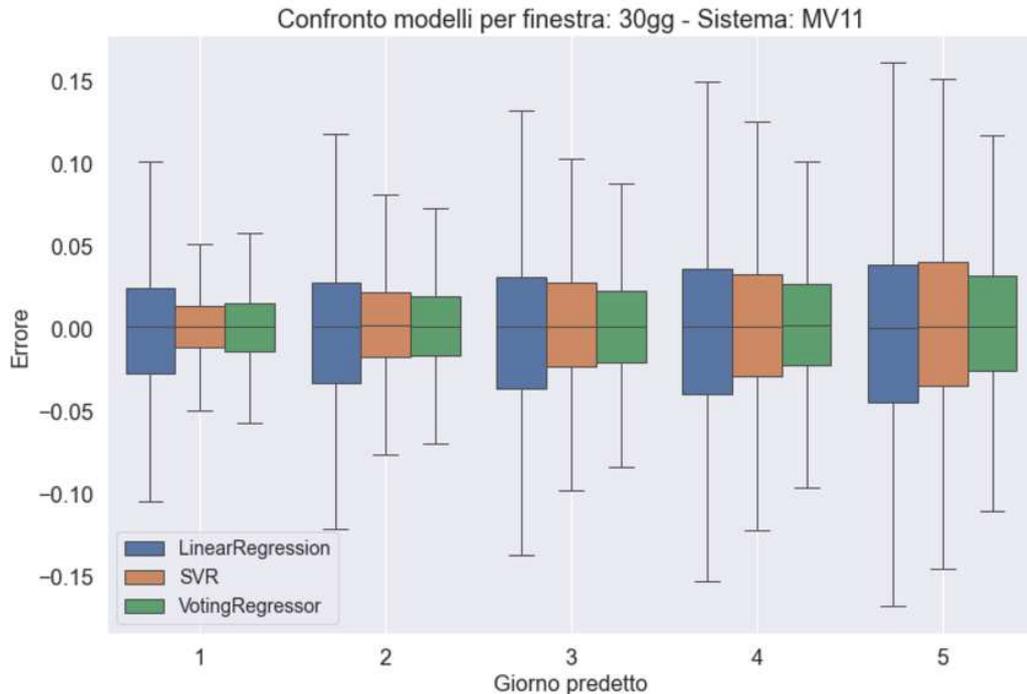


Figura 4.8: Il boxplot che illustra l'errore di previsione commesso su ciascun giorno predetto per la ddp.dc con ID 17271, rispetto a ciascun modello utilizzato, su di una finestra fissa scorrevole di 30 campioni

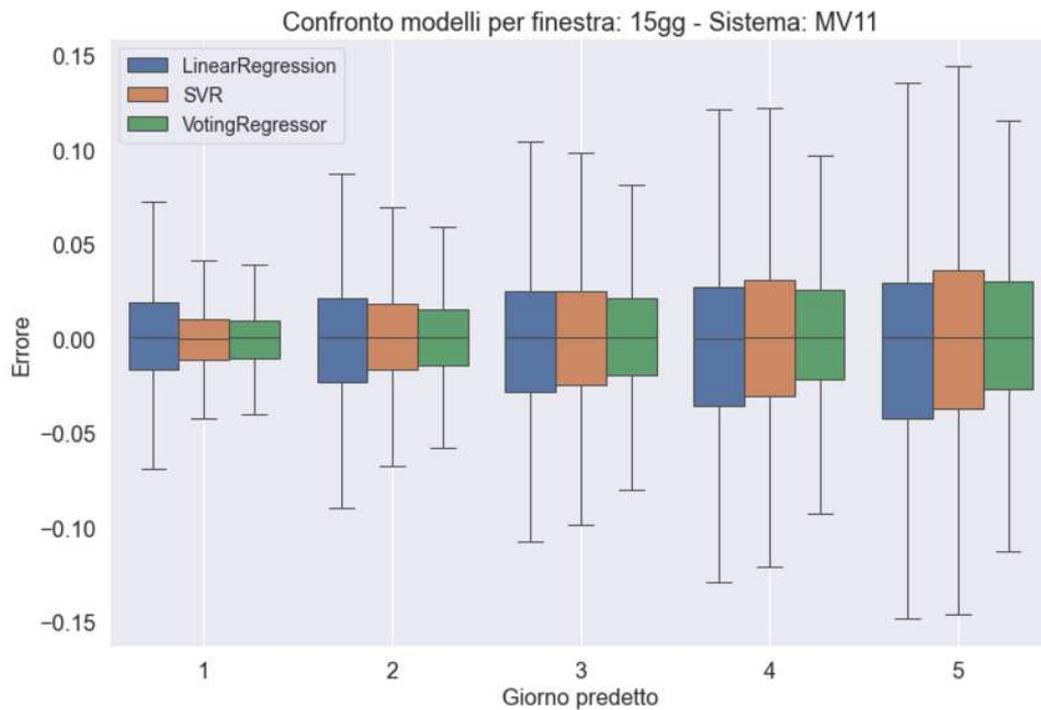


Figura 4.9: Il boxplot che illustra l'errore di previsione commesso su ciascun giorno predetto per la ddp.dc con ID 17271, rispetto a ciascun modello utilizzato, su di una finestra fissa scorrevole di 15 campioni

Dunque, i risultati migliori si ottengono su una finestra di 15 giorni ed il *Voting regressor* risulta il modello migliore che riesce a bilanciare la linearità e la non linearità nei dati e questo lo si dimostra nei frame dell'animazione riportati dalla Figura 4.10 alla Figura 4.13. È stato preso in considerazione un arco temporale più ristretto dove sono presenti diversi campioni fuori soglia, ovvero compreso tra il 2020-01-01 e il 2020-07-31. In questi frame:

- il punto blu sovrapposto alla retta tratteggiata in rosso rappresenta l'istante in cui viene effettuata la previsione, ovvero l'ultimo giorno della finestra temporale;
- i 15 punti blu antecedenti la retta rossa rappresentano i campioni che il modello utilizza per il *fit*;
- le curve sovrapposte ai punti blu prima della retta tratteggiata in rosso rappresentano le funzioni generate dal modello *SVR* e *VotingRegressor* su una finestra di 15 giorni;
- le curve sovrapposte ai punti blu dopo la retta tratteggiata in rosso rappresentano le 5 previsioni effettuate dai rispettivi modelli.

Nel primo frame in Figura 4.10 gli andamenti di entrambi i modelli risultano sovrapposti. Nel secondo frame in Figura 4.11 si evidenzia che, già dal secondo giorno fuori soglia, sia l'*SVR* che il *Voting Regressor* riescono a prevedere dei dati al di sopra di $-0.85V$, ma si osserva che l'*SVR* sembra essere più reattivo nel riconoscimento dei campioni fuori soglia, anche se entrambi i modelli riportano una previsione con andamento decrescente, differente rispetto all'andamento reale. Questo comportamento è dovuto al set di dati su cui si effettua il *fit*, infatti, prendendo in considerazione quest'ultimo frame, tali campioni hanno dei valori che si distribuiscono intorno a circa $-0.85V$, quindi la previsione difficilmente tenderà ad assumere valori più alti. Inoltre, da queste immagini si evince che, sebbene il *MAE* assuma valori alquanto piccoli nei cinque giorni di previsione, in questo contesto, l'errore riscontrato

sembra abbastanza elevato. Ad esempio, nel primo frame, l'errore più alto tra il valore reale e quello predetto di differenza di potenziale si aggira intorno a $-0.35V$, ma, osservando il boxplot in Figura 4.9, tale valore, sicuramente, risulta un outlier.

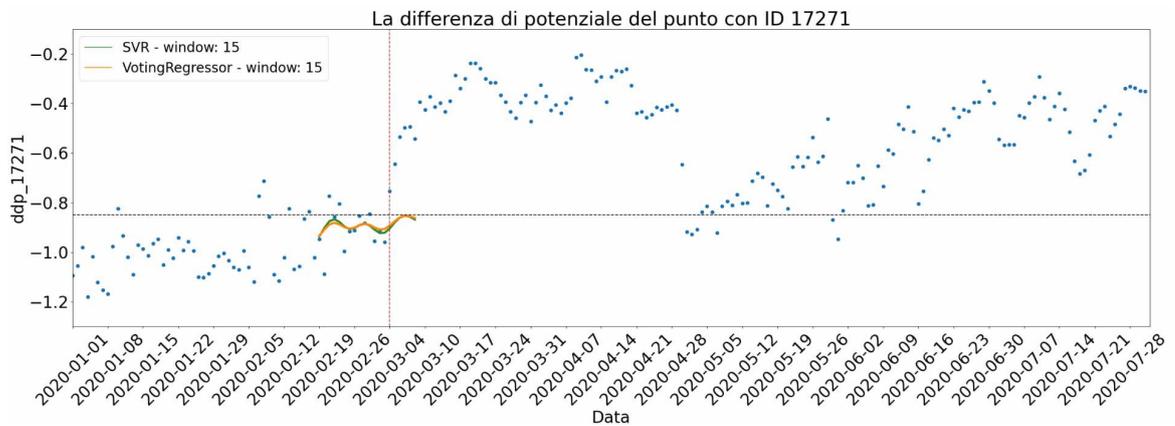


Figura 4.10: La predizione delle ddp.dc con ID 17271 effettuata il giorno 2020-03-04 in cui si evidenzia un comportamento simile tra i modelli SVR e Voting Regressor

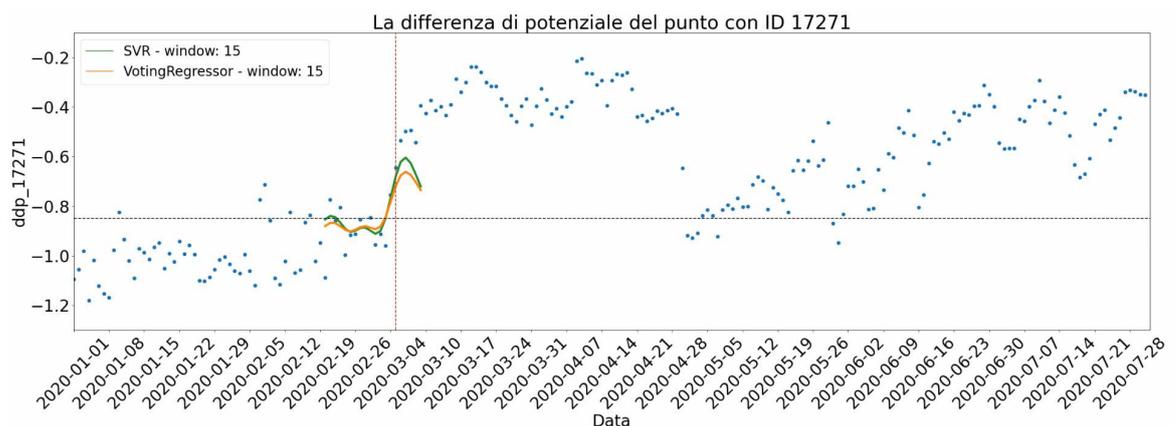


Figura 4.11: La predizione delle ddp.dc con ID 17271 effettuata il giorno 2020-03-05 in cui si evidenzia un comportamento più reattivo da parte dell'SVR, ma entrambi buoni predittori dei campioni fuori soglia

Nel frame in Figura 4.12 si evidenzia, invece, come il *Voting Regressor* riesce a catturare molto di più la tendenza dei dati in quanto, grazie alla sua componente lineare, la previsione sembra tendere maggiormente al comportamento dei dati effettivi.

È bene però specificare che, non sempre la previsione risulta essere migliore utilizzando il *Voting Regressor*. Infatti, se si effettua la previsione all'inizio di maggio, si evidenzia un decremento dei valori di ddp.dc e, dal frame riportato in Figura 4.13, si evidenzia che i valori stimati dall'SVR sono molto più vicini ai valori effettivi, rispetto alla stima effettuata dal *Voting Regressor*, il quale non riesce a catturare i valori fuori soglia. Dunque, sebbene quest'ultimo modello sia stato ritenuto il migliore, non sempre effettua la previsione migliore in assoluto.

Infine, dopo aver classificato i valori predetti ed i valori reali come spiegato nel paragrafo 3.4, sono state calcolate, per il modello *Voting Regressor* le metriche di Accuracy, Precision, Recall ed F1-score riportate in Figura 4.14 insieme alla relativa confusion matrix, in modo da verificare la sua capacità nel riconoscere le ddp.dc fuori soglia nel giorno più distante da

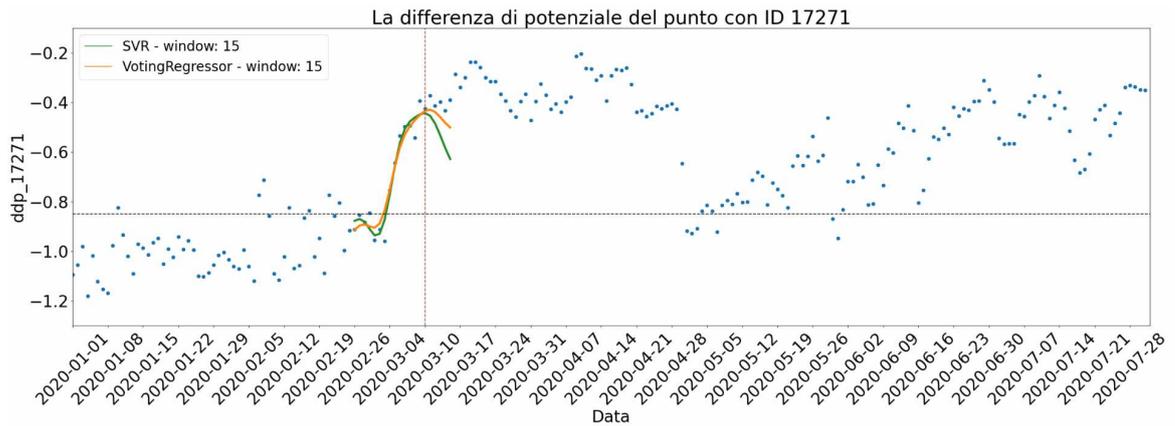


Figura 4.12: La predizione delle ddp.dc con ID 17271 effettuata il giorno 2020-03-10 in cui si evidenzia come il *Voting Regressor* riesce a catturare meglio la tendenza dei dati rispetto all'*SVR*, grazie alla sua componente lineare

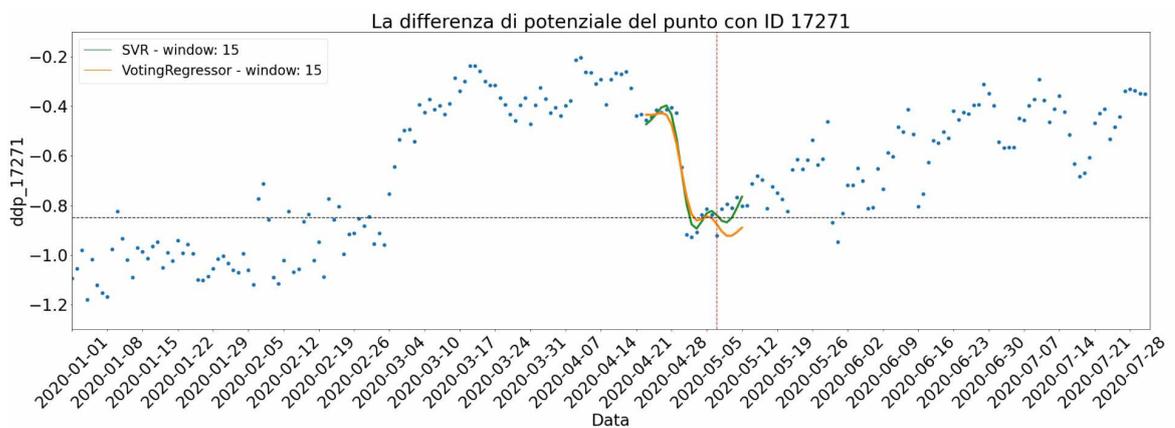


Figura 4.13: La predizione delle ddp.dc con ID 17271 effettuata il giorno 2020-05-06 in cui si evidenzia come l'*SVR* riesca ad effettuare una previsione migliore rispetto al *Voting Regressor*

quello in cui viene effettuata la previsione. Si evidenzia che l'accuratezza per l'ultimo giorno è del 93% e di tutti i dati fuori soglia il 3.6% dei campioni non è predetto correttamente.

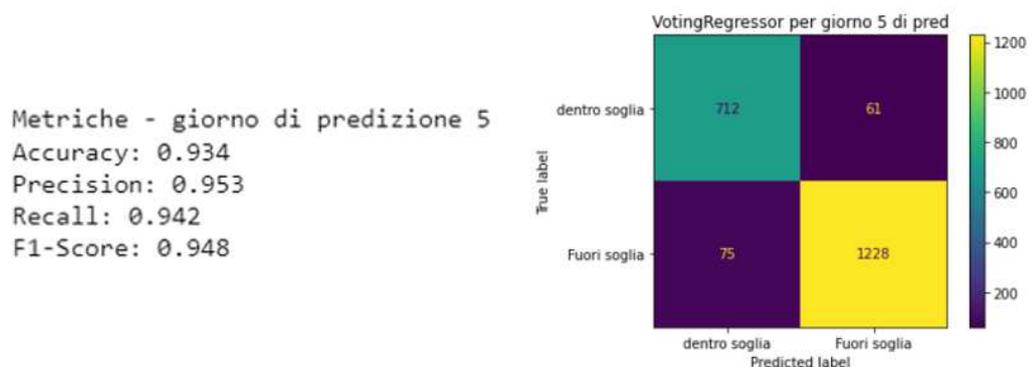


Figura 4.14: Le metriche e la confusion matrix relativa al quinto giorno di previsione della ddp.dc con ID 17271 relativo al *Voting Regressor* considerando una finestra di 15 giorni

4.2.2 La predizione della ddp.dc - ID 8984

Di seguito sono riportati sei istogrammi che descrivono, a coppie, i risultati dei 5 giorni predetti, rispetto alle tre finestre temporali scelte e a ciascun modello di regressione utilizzato. In Figura 5.10 e 5.11 sono riportati i risultati delle metriche MAE ed R^2 considerando la finestra di 45 giorni, in Figura 4.17 e 4.18 sono riportati i risultati delle metriche MAE ed R^2 considerando la finestra di 30 giorni ed, infine, in Figura 4.19 e 4.20 sono riportati i risultati delle metriche MAE ed R^2 considerando la finestra di 15 giorni.

I risultati ottenuti per la previsione della ddp.dc con *ID 8984* seguono le stesse caratteristiche descritte per la previsione della ddp.dc con *ID 17271*. Una differenza che è possibile notare in questo caso è che, considerando la finestra temporale di 15 giorni che riporta i risultati migliori, l'*SVR* ha un R^2 leggermente più alto rispetto al *Voting Regressor*, ma il MAE risulta leggermente peggiore, dunque, come per la ddp.dc precedentemente analizzata, si preferisce il *Voting Regressor* perché risulta essere un ottimo compromesso in termini di previsione. Per completezza, anche in questo caso sono stati visualizzati i boxplot relativi a ciascuna finestra e modello (Figura 4.21, Figura 4.22 e Figura 4.23) e si evidenzia, come nel caso precedente, che la mediana dell'errore si mantiene sempre intorno all'origine e che la sua distribuzione è piuttosto contenuta. In particolare, il *Voting Regressor* in corrispondenza della finestra di 15 giorni ha errori che si aggirano tra 0.025 V e -0.025 V.

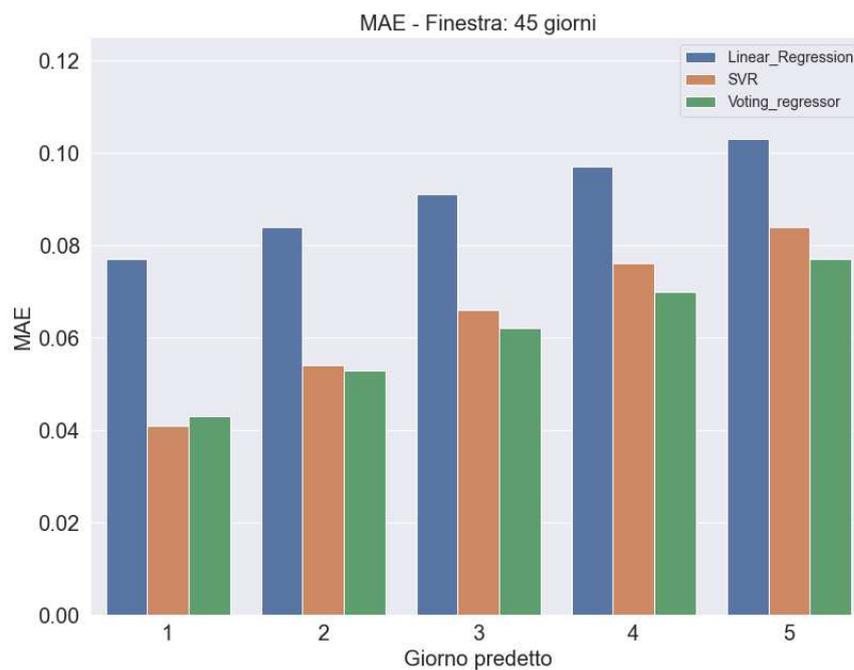


Figura 4.15: L'istogramma che illustra il MAE calcolato su ciascun giorno predetto per la ddp.dc con *ID 8984*, rispetto a ciascun modello utilizzato, su di una finestra fissa scorrevole di 45 campioni.

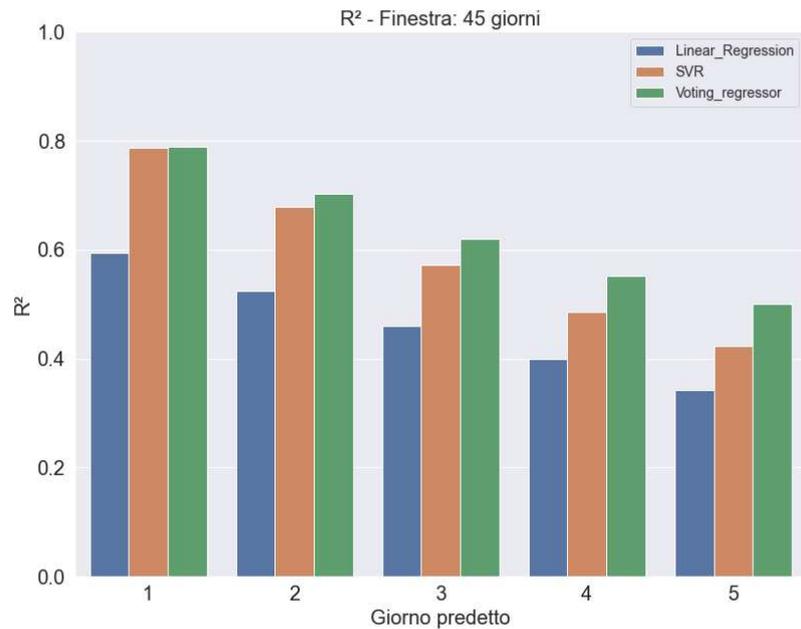


Figura 4.16: L'istogramma che illustra l' R^2 calcolato su ciascun giorno predetto per la ddp.dc con ID 8984, rispetto a ciascun modello utilizzato, su di una finestra fissa scorrevole di 45 campioni.

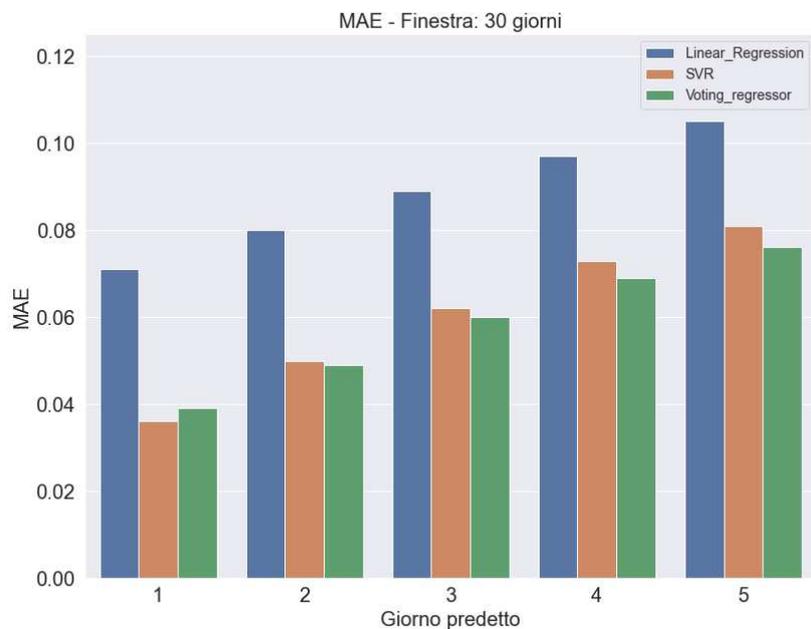


Figura 4.17: L'istogramma che illustra il MAE calcolato su ciascun giorno predetto per la ddp.dc con ID 8984, rispetto a ciascun modello utilizzato, su di una finestra fissa scorrevole di 30 campioni.

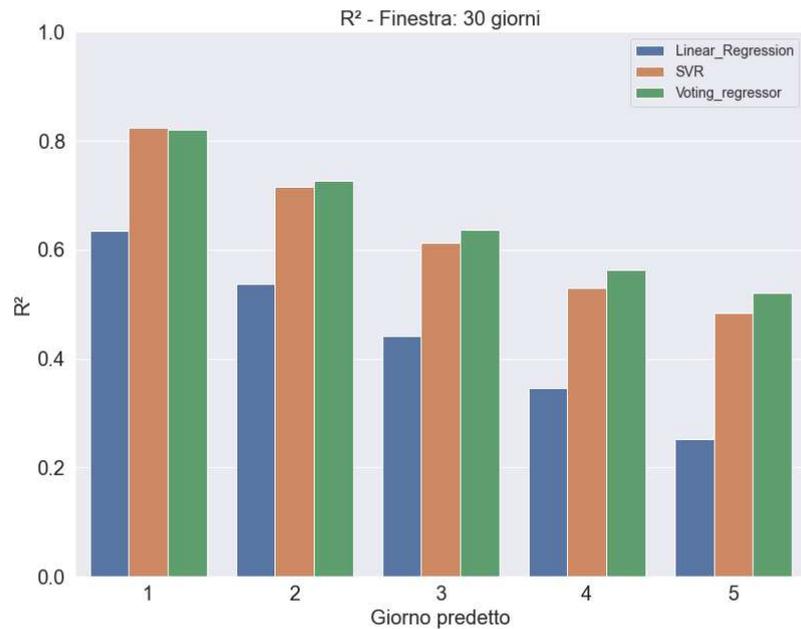


Figura 4.18: L'istogramma che illustra l' R^2 calcolato su ciascun giorno predetto per la ddp.dc con ID 8984, rispetto a ciascun modello utilizzato, su di una finestra fissa scorrevole di 30 campioni.

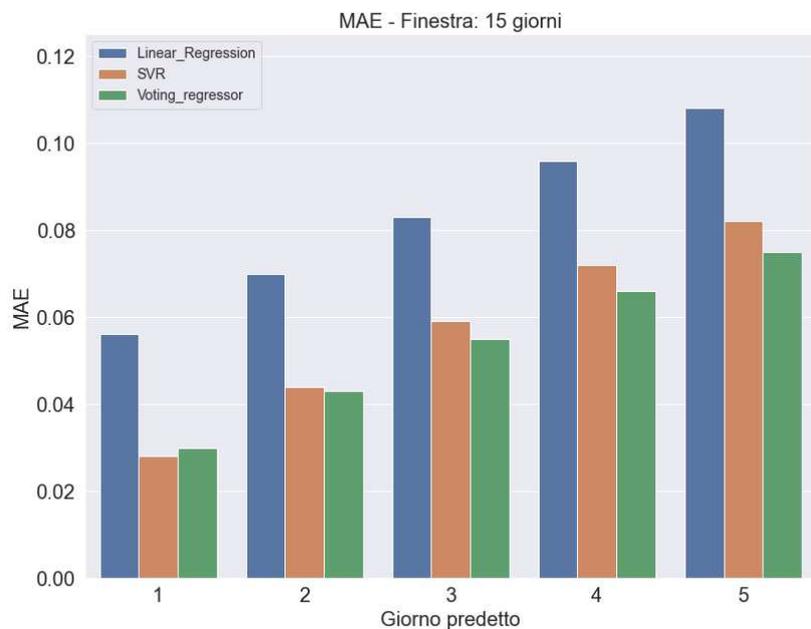


Figura 4.19: L'istogramma che illustra il MAE calcolato su ciascun giorno predetto per la ddp.dc con ID 8984, rispetto a ciascun modello utilizzato, su di una finestra fissa scorrevole di 15 campioni.

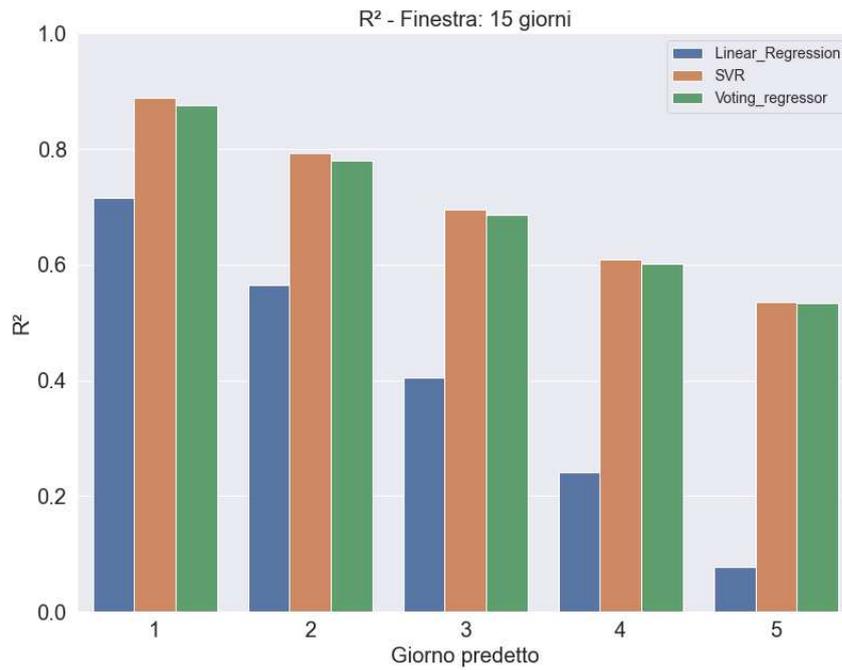


Figura 4.20: L'istogramma che illustra l' R^2 calcolato su ciascun giorno predetto per la ddp.dc con ID 8984, rispetto a ciascun modello utilizzato, su di una finestra fissa scorrevole di 15 campioni.

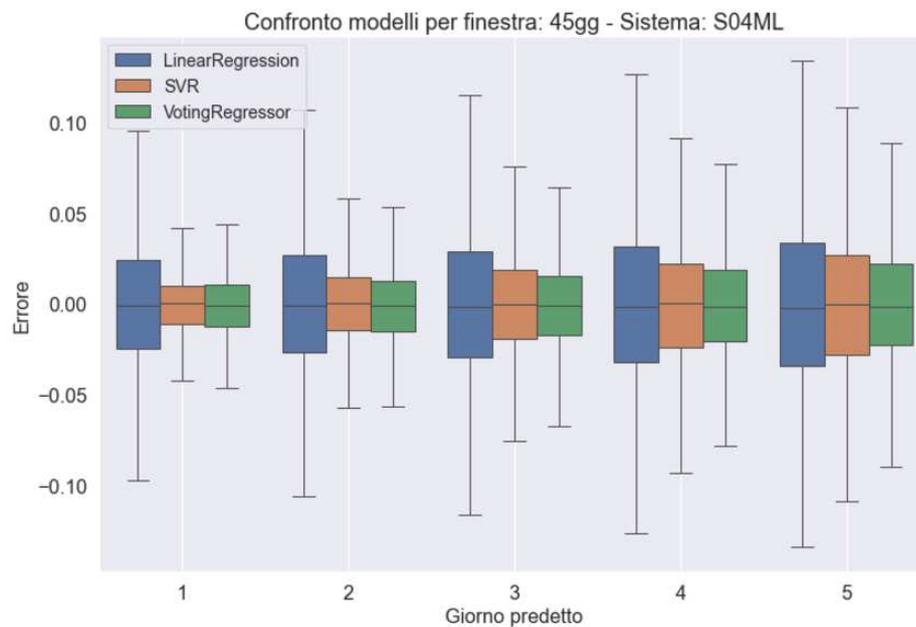


Figura 4.21: Il boxplot che illustra l'errore di previsione commesso su ciascun giorno predetto per la ddp.dc con ID 8984, rispetto a ciascun modello utilizzato, su di una finestra fissa scorrevole di 45 campioni

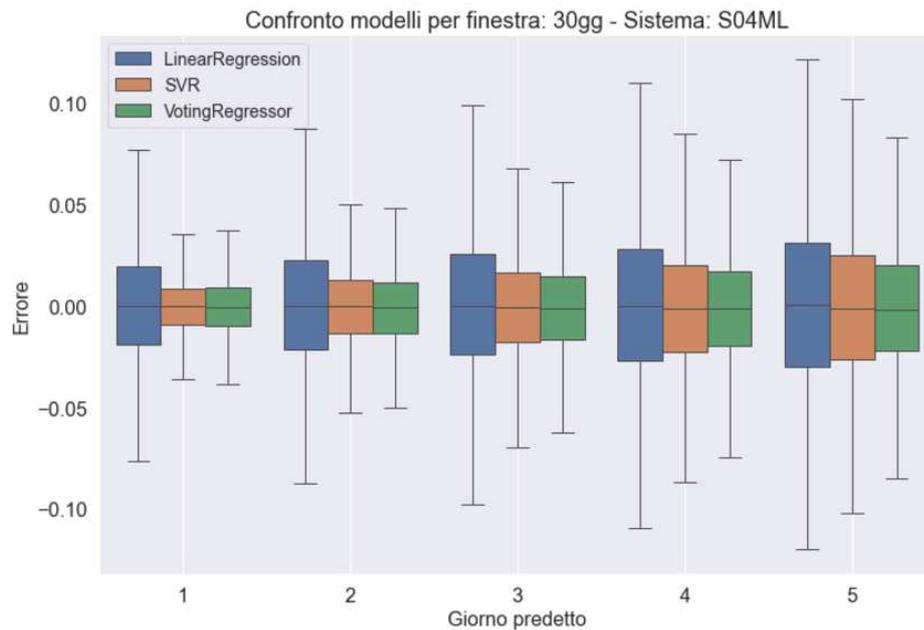


Figura 4.22: Il boxplot che illustra l'errore di previsione commesso su ciascun giorno predetto per la ddp.dc con ID 8984, rispetto a ciascun modello utilizzato, su di una finestra fissa scorrevole di 30 campioni

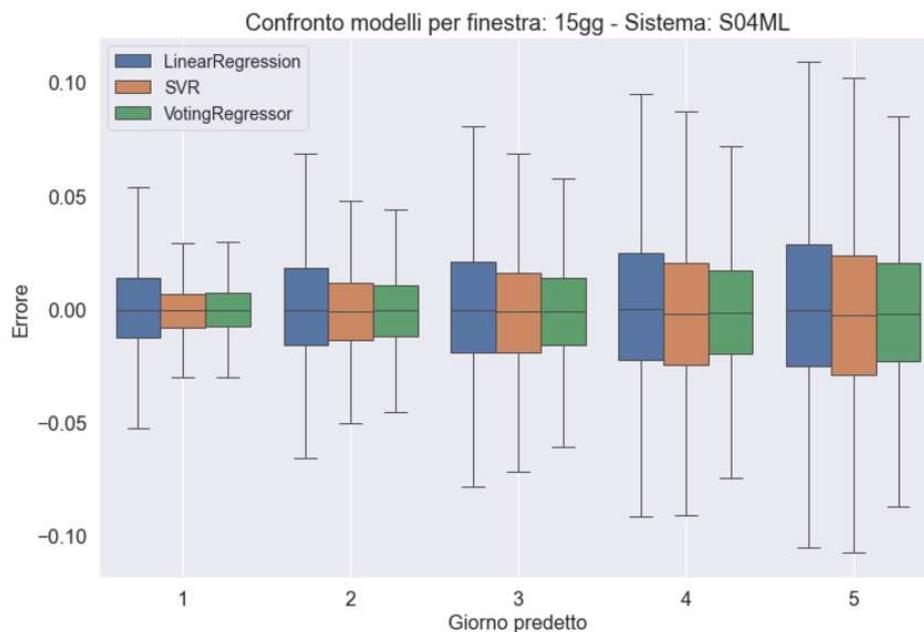


Figura 4.23: Il boxplot che illustra l'errore di previsione commesso su ciascun giorno predetto per la ddp.dc con ID 8984, rispetto a ciascun modello utilizzato, su di una finestra fissa scorrevole di 15 campioni

Tramite i frame delle animazioni riportate di seguito è possibile anche verificare la bontà del modello *Voting Regressor* con finestra di 15 giorni. E' stato preso in considerazione un arco temporale più ristretto dove sono presenti diversi campioni fuori soglia, ovvero compreso tra il 2016-07-01 e il 2017-02-28; ogni frame riporta le stesse caratteristiche definite precedentemente per la ddp.dc con id 17271. Nella prima immagine in Figura 4.24 si evidenzia

che l'SVR ed il *Voting Regressor* hanno andamenti molto simili, ma il primo è leggermente migliore, in particolare, la previsione del terzo, quarto e quinto giorno della curva arancione è leggermente più distante dai punti effettivi rispetto alle previsioni della curva verde. In Figura 4.25 entrambi i modelli riportano un andamento decrescente, differente rispetto all'andamento reale e dovuto al delta tra i campioni della finestra temporale su cui è stato eseguito il *fit*, comportamento osservato anche nella ddp.dc con ID 17271. Sebbene il primo risulta più reattivo nella previsione di campioni fuori soglia, alle iterazioni successive, come in Figura 4.26 si evidenzia come il *Voting Regressor* riesce a catturare più efficacemente la tendenza che hanno i dati rispetto all'SVR, ma, a causa della brusca salita di ddp.dc, solo al quinto giorno fuori soglia riesce ad individuare che tutti i 5 valori predetti sono al di sopra di -0.85V.

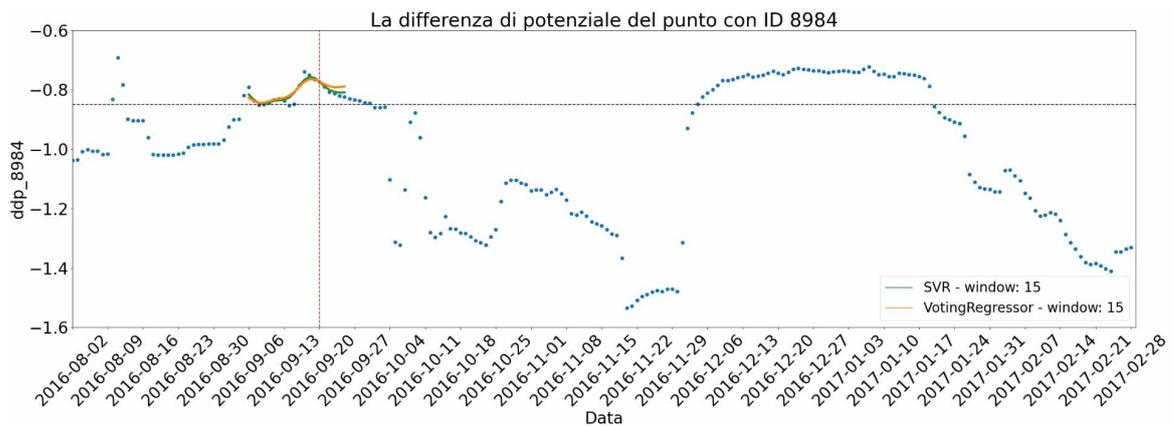


Figura 4.24: La predizione delle ddp.dc con ID 8984 effettuata il giorno 2016-09-19 in cui si evidenzia come l'SVR ed il emphVoting Regressor abbiano un andamento molto simile

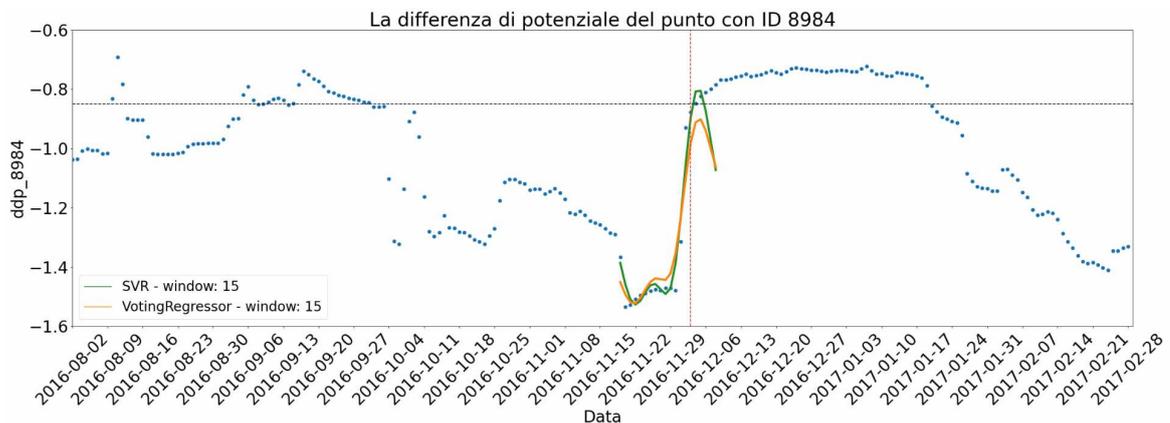


Figura 4.25: La predizione delle ddp.dc con ID 8984 effettuata il giorno 2026-12-02 in cui si evidenzia come l'SVR riesca ad effettuare una previsione migliore rispetto alVoting Regressor

Infine, dopo aver classificato i valori precetti ed i valori reali come spiegato nel paragrafo 3.4, sono state calcolate, per il modello *Voting Regressor* le metriche di Accuracy, Precision, Recall ed F1-score riportate in Figura 4.27, insieme alla relativa confusion matrix, in modo da verificare la sua capacità nel riconoscere le ddp.dc fuori soglia nel giorno più distante da quello in cui viene effettuata la previsione. Si evidenzia che l'accuratezza per l'ultimo giorno è del 91% e di tutti i dati fuori soglia il 8.6% dei campioni non è predetto correttamente.

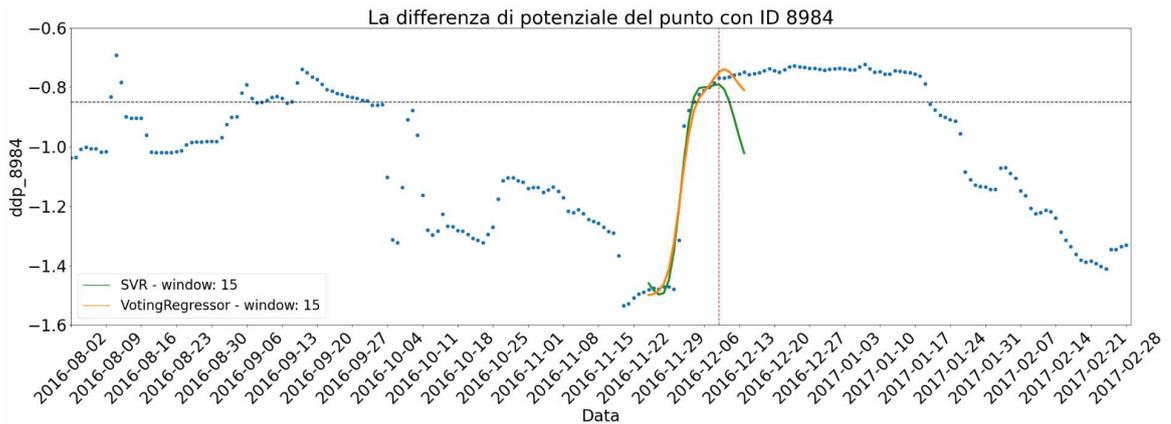


Figura 4.26: La predizione delle ddp.dc con ID 8984 effettuata il giorno 2016-12-09 in cui si evidenzia come *Voting Regressor* riesce a catturare maggiormente la tendenza che hanno i dati rispetto all'*SVR*

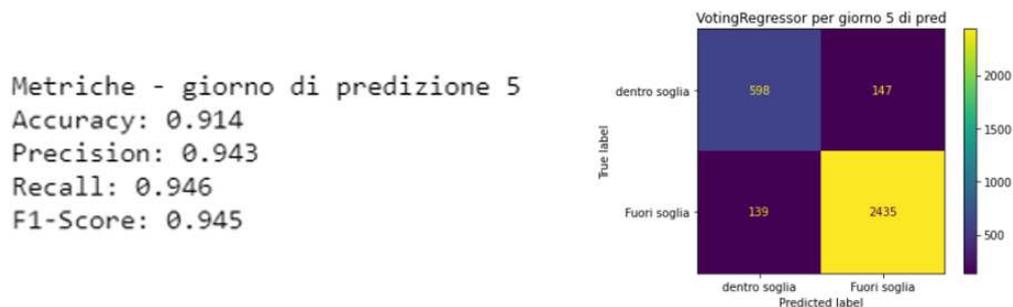


Figura 4.27: Le metriche e la confusion matrix relativa al quinto giorno di previsione della ddp.dc con ID 8984 relativo al *Voting Regressor* considerando una finestra di 15 giorni

4.3 Confronto dei risultati delle ddp.dc

In entrambi i casi analizzati il *VotingRegressor* risulta essere il predittore migliore su una finestra di 15 giorni, sia per i risultati ottenuti dalle metriche, in particolare per i valori riportati dal MAE, sia per i riscontri ottenuti a livello visivo con le animazioni. Inoltre, i risultati legati all'accuratezza nel riconoscimento delle ddp.dc fuori soglia dell'ultimo giorno di predizione sono abbastanza buoni. Il punto di debolezza del metodo e modello utilizzato è che, se si registrano, tra un giorno e l'altro, escursioni di ddp.dc moderatamente alte dovute ad un malfunzionamento, come si è potuto già analizzare, la previsione in quel caso sarebbe sempre soggetta ad un ritardo più o meno grande in base ai valori dei campioni nella finestra temporale. Tra i risultati ottenuti per la ddp.dc con ID 8984 si evidenzia che l' R^2 registrato per ciascuna finestra e giorno di predizione, se confrontato con i rispettivi risultati della ddp.dc con ID 17271 è moderatamente inferiore. Nella Tabella 4.1 e Tabella 4.2 sono riportati rispettivamente i MAE e gli R^2 ottenuti una volta eseguito il fitting, sfruttando il *VotingRegressor* per una finestra di 15 giorni.

Dunque, il modello creato ad ogni iterazione è in grado di spiegare meglio i dati della prima ddp.dc analizzata rispetto alla seconda ddp.dc. Questo probabilmente è dovuto al fatto che la ddp.dc con ID 8984 è descritta da una serie temporale composta da brusche variazioni tra campioni successivi.

Giorno	ddp.dc – ID 17271	ddp.dc – ID 8984
Finestra	15gg	15gg
MAE - Giorno 1	0.037	0.03
MAE - Giorno 2	0.05	0.043
MAE - Giorno 3	0.06	0.055
MAE - Giorno 4	0.067	0.066
MAE - Giorno 5	0.074	0.075

Tabella 4.1: Valori di MAE calcolati eseguendo il *fit* del modello *Voting Regressor* su tutta la lunghezza della serie temporale corrispondente alla ddp.dc con ID 17271 e alla ddp.dc con ID 8984, considerando una finestra scorrevole di 15 giorni

Giorno	ddp.dc – ID 17271	ddp.dc – ID 8984
Finestra	15gg	15gg
R^2 - Giorno 1	0.951	0.876
R^2 - Giorno 2	0.911	0.780
R^2 - Giorno 3	0.877	0.686
R^2 - Giorno 4	0.852	0.602
R^2 - Giorno 5	0.832	0.533

Tabella 4.2: I valori di R^2 calcolati eseguendo il *fit* del modello *Voting Regressor* su tutta la lunghezza della serie temporale corrispondente alla ddp.dc con ID 17271 e alla ddp.dc con ID 8984, considerando una finestra scorrevole di 15 giorni

L'implementazione ed i risultati della LSTM

In questo capitolo è illustrata, innanzitutto, l'implementazione della rete neurale LSTM che è stata eseguita in Python, sfruttando la piattaforma Jupyter notebook. Per ciascuna ddp.dc da predire, sono state addestrate due differenti reti neurali e nei paragrafi successivi sono mostrati i principali risultati ottenuti dalla predizione della ddp.dc con ID 17271 del sistema MV11 e della ddp.dc con ID 8984 del sistema S04ML. Infine, è stato effettuato un confronto per la valutazione complessiva della metodologia utilizzata.

5.1 L'implementazione delle reti LSTM

Per implementare una rete LSTM, innanzitutto, sono stati scelti i predittori ed il target su cui addestrare e, successivamente, testare la LSTM; sono stati implementati, per ciascun sistema, due tipologie di modelli;

- **univariabile**, considerando come variabile target le sequenze di ddp.dc da predire e come unico predittore le sequenze di ddp.dc su cui si effettua la predizione;
- **multivariabile**, considerando come variabile target le sequenze di ddp.dc da predire e come predittori le sequenze di ddp.dc e di corrente di alimentazione su cui si effettua la predizione.

Per ciascun modello è stata effettuata la divisione in train e test con la classica divisione 80-20, considerando come test, in entrambe le ddp.dc, il primo 20% della serie temporale; questa scelta è stata fatta, in quanto, in corrispondenza di tali valori, si evince un'andamento più variabile ed, in questo modo, si riuscirebbe a valutare meglio la capacità di generalizzazione della rete analizzando i risultati di una previsione non troppo "scontata". Successivamente, è stata effettuata una scalatura delle variabili con la tecnica del *MinMaxScaler* [Brownlee, 2020], implementato tramite la libreria di *scikit-learn* di Python. In generale, per scalare un set di dati, a ciascun campione X si applica la seguente formula:

$$X_{scaled} = \frac{X - X_{min}}{X_{max} - X_{min}} \quad (5.1.1)$$

dove X_{min} è il valore di minimo nel set di dati, mentre X_{max} è il valore di massimo nel set di dati. Sfruttando tale tecnica è stato scalato il train mediante la funzione `fit_transform()` e lo scaler ottenuto dal suddetto set di dati è stato, successivamente, applicato al test tramite la funzione `transform()`. Il passo successivo è stato implementare una funzione che creasse

le sequenze per i predittori ed i target come spiegato nel sottoparagrafo 3.3.2.; la gestione dei dati *NaN* è stata effettuata all'interno di questa funzione, durante, appunto, la creazione delle sequenze. Facendo riferimento all'immagine in Figura 3.5, se si considera un modello multivariabile, la *F1* corrisponde alla *ddp.dc* di un sistema ed *F2* corrisponde alla corrente dell'alimentatore dello stesso sistema; se si considera un modello univariabile, si fa riferimento solo alla *F1*. Per creare *X* è stato considerato sul dataset iniziale (composto dai campioni di *ddp.dc* ed, eventualmente, *I*) una finestra scorrevole di lunghezza di 45 campioni per un test ed una finestra di 15 per un altro test; mentre, per creare *Y* sono stati sempre considerati sul dataset iniziale i 5 campioni successivi alla finestra temporale corrente. Tale funzione è stata, successivamente, applicata sia sul dataset di train che sul dataset di test creando 4 set di dati chiamati *X_train_seq*, *Y_train_seq*, *X_test_seq* e *Y_test_seq*. In seguito, è stato implementato il modello, tramite la libreria *PyTorch* di Python, addestrato sul dataset di train e testato sul dataset di test. Per valutare le performance della rete sono state calcolate le metriche *MAE* ed R^2 dopo aver eseguito la funzione *inverse_scaler()* sulle previsioni effettuate sul test.

Nel prossimo paragrafo sono riportati i risultati ottenuti sia per la previsione della *ddp.dc* con *ID 17271* che la previsione della *ddp.dc* con *ID 8984* ottenuti dall'addestramento di due reti *LSTM* differenti; infatti, tali risultati fanno riferimento ai modelli migliori per la previsione di ciascuna *ddp.dc*, addestrati, quindi, su di una struttura ed iperparametri differenti, riportati nella Tabella 5.1. La struttura e gli iperparametri sono stati scelti in seguito ad un *fine tuning* tramite cui si è riscontrato, appunto, che la previsione migliore di ciascuna *ddp.dc* si ottiene sulla base di modelli diversi.

	ddp.dc – ID 17271		ddp.dc – ID 8984	
	Univariabile	Multivariabile	Univariabile	Multivariabile
Hidden size	256	256	512	512
N° layer	5	5	4	4
Learning rate	$5 \cdot 10^{-4}$	$5 \cdot 10^{-4}$	$5 \cdot 10^{-5}$	$5 \cdot 10^{-5}$

Tabella 5.1: La struttura e gli iperparametri scelti per ciascun modello di rete neurale *LSTM* addestrata sui dati della *ddp.dc* con *ID 17271* e della *ddp.dc* con *ID 8984*

Il primo sistema è stato addestrato su 150 epoche, mentre il secondo su 250; entrambi i sistemi sono stati allenati con *batch size* pari a 32 ed hanno sfruttato, come *loss function*, l'*MSELoss* e, come *ottimizzatore*, l'*Adam*. Al termine del training, è stato considerato il modello con il minimo valore di *MSE* per il calcolo delle metriche *MAE* ed R^2 , riportate nel prossimo paragrafo.

5.2 I risultati della predizione

Le differenze di potenziale scelte per effettuare la previsione sono la *ddp.dc* corrispondente al punto con *ID 17271* del sistema *MV11* e la *ddp.dc* corrispondente al punto con *ID 8984* del sistema *S04ML*. Per i relativi modelli univariabili, le variabili prese in considerazione sono la differenza di potenziale più critica del sistema *MV11*, che registra 2.113 campioni, tra il 2018-06-14 e il 2024-04-18 e la differenza di potenziale più critica del sistema *S04ML* che registra 3.377 campioni, tra il 2013-05-23 e il 2024-04-18. Mentre, per i relativi modelli multivariabili, le variabili prese in considerazione sono, oltre a quelle appena descritte, an-

che le relative correnti di alimentazione del sistem *MV11*, che registra 1.493 campioni, dal 2020-03-17 e il 2024-04-18 e la corrente di alimentazione del sistema *S04ML*, che registra 1.982 campioni, tra il 2018-10-23 e il 2024-04-18. Di seguito sono riportati i risultati ottenuti per ciascun modello. Un importante aspetto da prendere per la multivariabile è che, la corrente registra meno dati rispetto alla relativa *ddp.dc*, di conseguenza i campioni utilizzati per effettuare l'addestramento risultano in numero inferiore rispetto ai campioni utilizzati per il modello univariato. Infatti, per il dataset considerato per la multivariata è stato eseguito un filtraggio iniziale tale per cui i primi campioni corrispondano alle misurazioni effettuate il primo giorno in cui è disponibile la corrente, altrimenti il dataset riporterebbe *NaN* che verrebbero considerati nei set di train e test, alterandone l'effettiva dimensione utile.

5.2.1 La predizione della *ddp.dc* - ID 17271

Di seguito sono riportati sei istogrammi che descrivono, a coppie, i risultati ottenuti in seguito all'addestramento della rete. In particolare, in Figura 5.1 e Figura 5.2 sono riportati gli istogrammi che illustrano, rispettivamente, il *MAE* e l' R^2 calcolato su ciascun giorno predetto, rispetto al modello univariabile e multivariabile e considerando, sul dataset iniziale, una finestra scorrevole di 15 e 45 giorni. Analizzando la prima immagine si evidenzia che, come ci si aspetterebbe, all'aumentare del giorno di predizione, il *MAE* tende a crescere. Il modello monovariabile migliore si ha considerando una finestra di 45 campioni, ma tale metrica risulta superiore rispetto a quelle riportate dalla multivariabile, la quale, nel complesso, risulta migliore considerando sempre una finestra di 45 giorni. Osservando la seconda immagine, si osserva che, sebbene i valori più alti di R^2 siano riportati dal modello monovariato su una finestra di 45 campioni, non è sufficiente per decretarlo il più performante in quanto, a parità di giorno predetto, riporta, per ciascun giorno, un errore maggiore rispetto alla multivariata, la quale restituisce un R^2 abbastanza soddisfacente

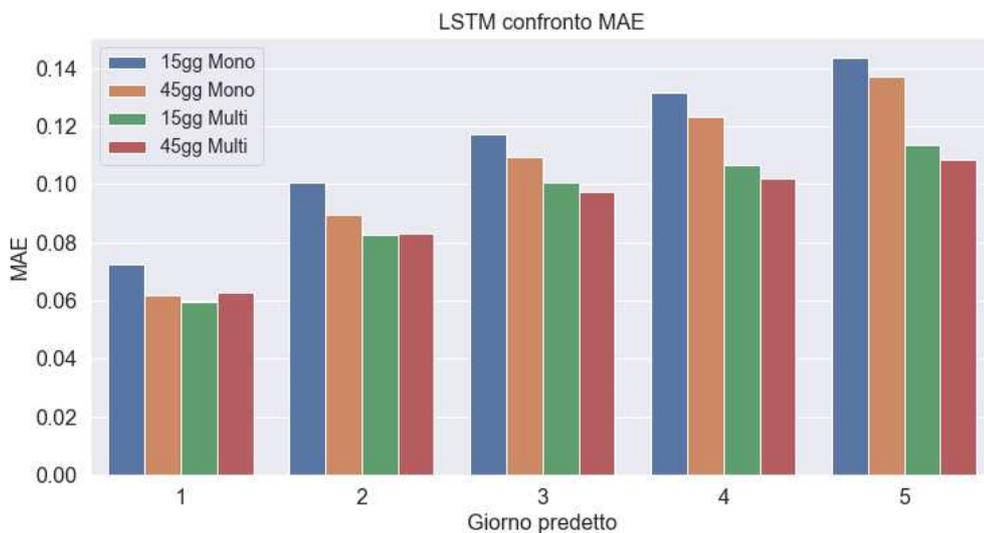


Figura 5.1: L'istogramma che illustra il *MAE* calcolato su ciascun giorno predetto, addestrando la rete per la *ddp.dc* con ID 17271 rispetto al modello univariabile e multivariabile, considerando, sul dataset iniziale, una finestra scorrevole di 15 e 45 giorni

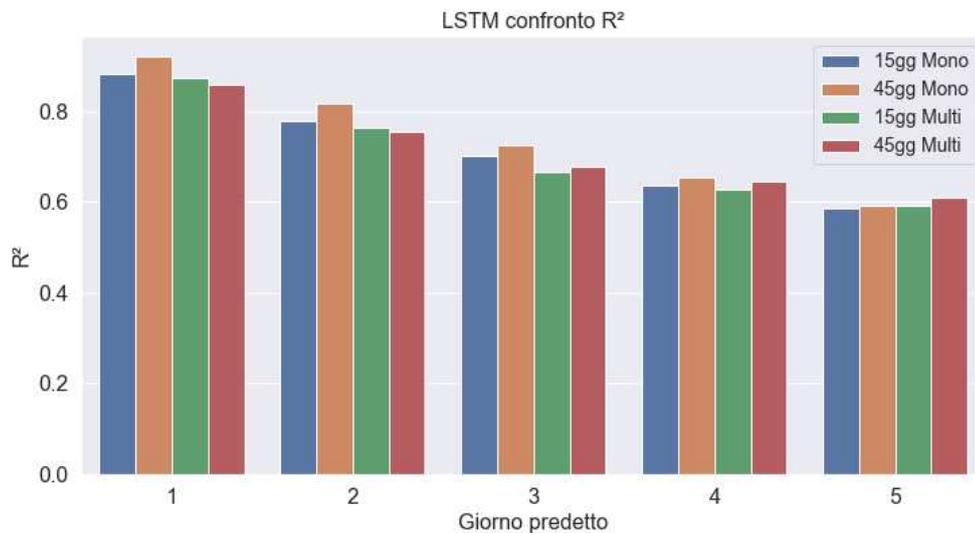


Figura 5.2: L'istogramma che illustra l' R^2 calcolato su ciascun giorno predetto, addestrando la rete per la ddp.dc con ID 17271 rispetto al modello univariabile e multivariabile, considerando, sul dataset iniziale, una finestra scorrevole di 15 e 45 giorni

In Figura 5.3 è riportato, per il modello multivariabile con finestra di 45 campioni, l'andamento dell'MSELoss durante il train ed il test effettuato su 150 epoche. La loss del test risulta superiore a quella del train e l'epoca migliore, in corrispondenza della quale è stato testato il modello, è la 93. Dalla 94-esima epoca il modello inizia ad overfittare.

Di seguito sono illustrati gli andamenti nel tempo della previsione sui dati di test dal primo (Figura 5.5) al quinto giorno (Figura 5.8) di previsione ed i rispettivi errori di predizione; tramite questi grafici è possibile visualizzare l'evoluzione della stima man mano che ci si allontana dal giorno in cui si esegue la previsione. Nella prima immagine si deduce che la previsione, evidenziata in rosso, segue molto bene l'andamento reale della previsione, soprattutto nella prima metà della serie temporale; mentre, l'errore è ben distribuito intorno allo zero ed il valore più alto raggiunto è di circa 0.35V. Nella seconda immagine, invece, è evidente come la previsione sia soggetta ad un delay, in quanto, sebbene l'andamento sembra seguirlo, la previsione è traslata verso destra ed infatti l'errore commesso ha una distribuzione molto più ampia.

Infine, dopo aver classificato i valori precetti ed i valori reali come spiegato nel paragrafo 3.4, sono state calcolate, per il modello multivariabile (finestra 45 giorni) le metriche di Accuracy, Precision, Recall ed F1-score riportate in Figura 5.9, insieme alla relativa confusion matrix, in modo da verificare la sua capacità nel riconoscere le ddp.dc fuori soglia nel giorno più distante da quello in cui viene effettuata la previsione. Si evidenzia che l'accuratezza per l'ultimo giorno è di circa l'88% e di tutti i dati fuori soglia il 4.8% dei campioni non è predetto correttamente.

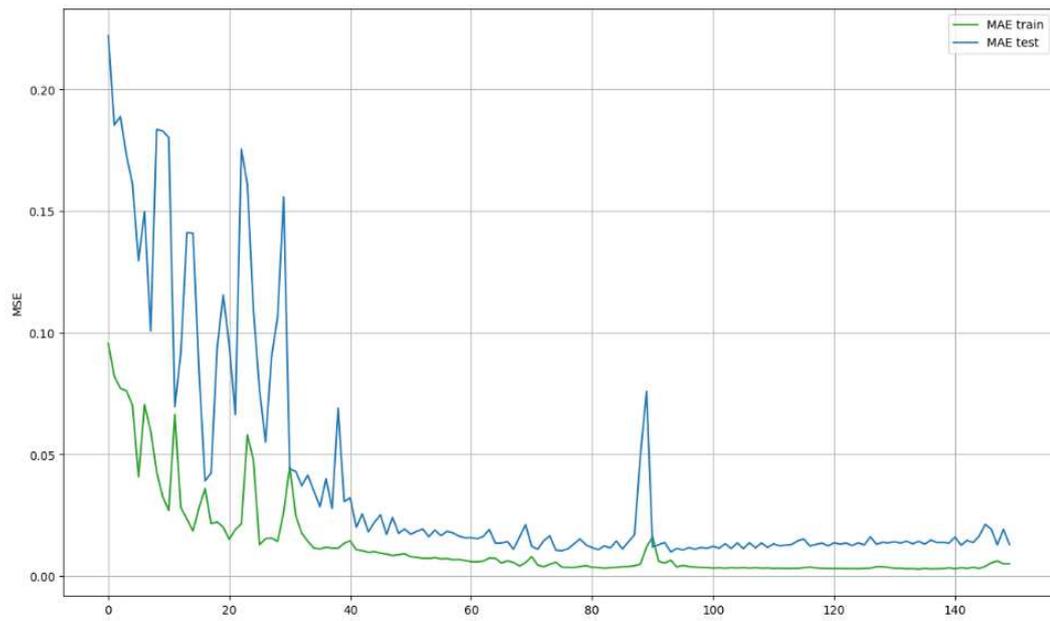


Figura 5.3: L'andamento dell'MSELoss durante il train ed il test effettuato su 150 epoche – modello multivariabile con finestra di 45 campioni



Figura 5.4: L'andamento nel tempo del primo giorno di previsione - modello multivariabile e finestra 45 giorni



Figura 5.5: L'andamento nel tempo del secondo giorno di previsione - modello multivariabile e finestra 45 giorni



Figura 5.6: L'andamento nel tempo del terzo giorno di previsione - modello multivariabile e finestra 45 giorni



Figura 5.7: L'andamento nel tempo del quarto giorno di previsione - modello multivariabile e finestra 45 giorni

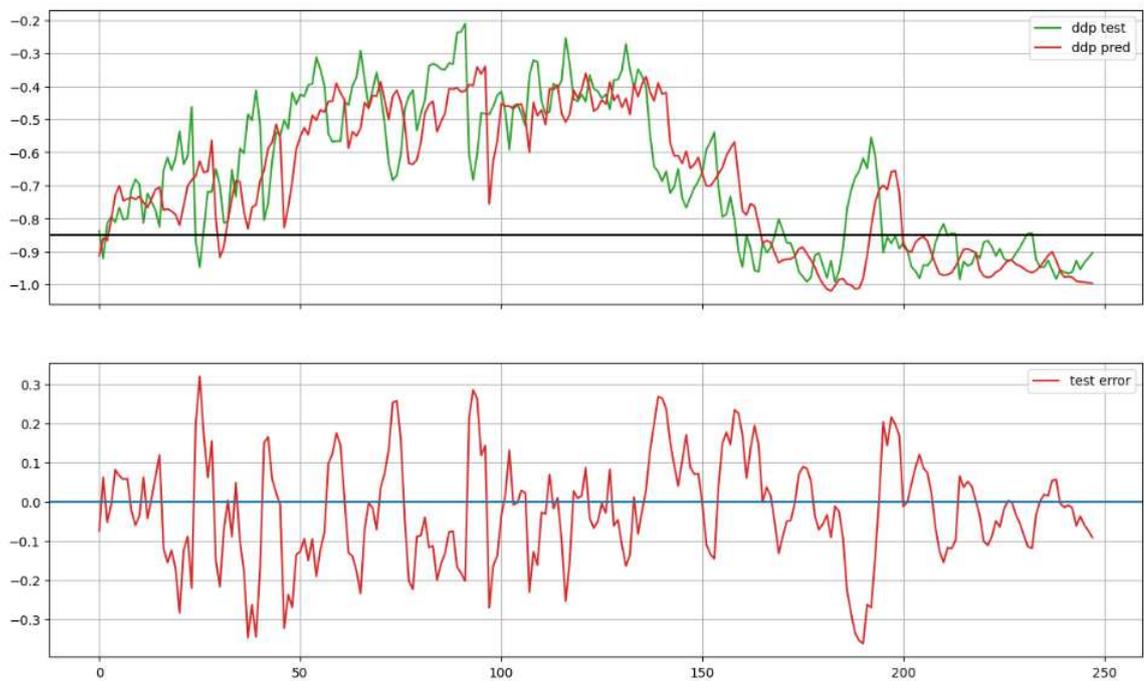


Figura 5.8: L'andamento nel tempo del quinto giorno di previsione - modello multivariabile e finestra 45 giorni

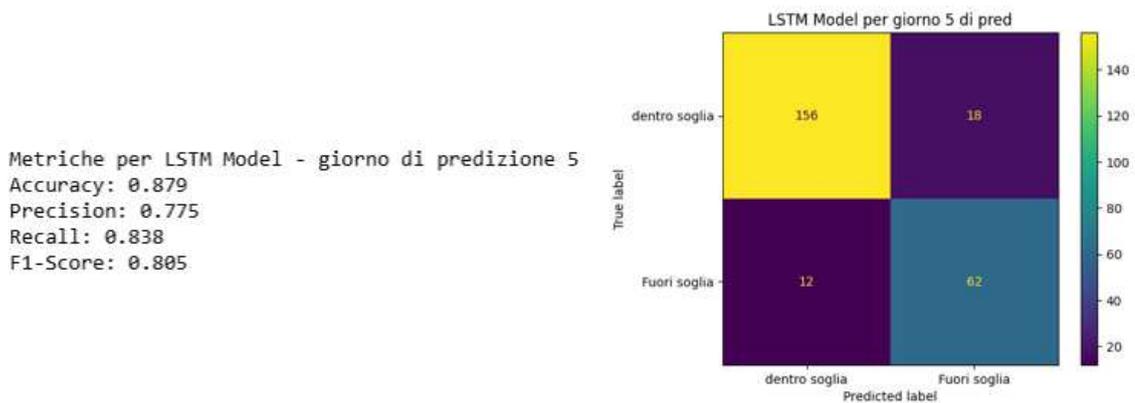


Figura 5.9: Le metriche e la confusion matrix relativa al quinto giorno di previsione per il modello multivariabile con finestra di 45 giorni

5.2.2 La previsione della ddp.dc - ID 8984

Di seguito sono riportati sei istogrammi che descrivono, a coppie, i risultati ottenuti in seguito all'addestramento della rete. In particolare, in Figura 5.10 e Figura 5.11 sono riportati gli istogrammi che illustrano, rispettivamente, il MAE e l' R^2 calcolato su ciascun giorno predetto, rispetto al modello univariabile e multivariabile e considerando, sul dataset iniziale, una finestra scorrevole di 15 e 45 giorni. In entrambi i grafici spicca subito all'occhio un grande divario tra i risultati ottenuti per la monovariata ed in risultati ottenuti per la multivariata; infatti, quest'ultimo risulta avere delle ottime performance sia a livello di MAE che a livello di R^2 . Sebbene la scelta della finestra, in questo caso, risulta quasi marginale dati i risultati di entrambe, si pone maggiormente attenzione al modello con finestra pari a 45 in quanto sembra prevedere leggermente meglio il quinto giorno.

In Figura 5.12 è riportato, per il modello multivariabile con finestra di 45 campioni, l'andamento dell'MSE Loss durante il train ed il test effettuato su 250 epoche. La loss del test risulta superiore a quella del train e l'epoca migliore, in corrispondenza della quale è stato testato il modello, è la 250.

Di seguito sono illustrati gli andamenti nel tempo della previsione sui dati di test dal primo (Figura 5.13) e al quinto giorno (Figura 5.17) di previsione ed i rispettivi errori di previsione; tramite questi grafici è possibile visualizzare l'evoluzione della stima man mano che ci allontaniamo dal giorno in cui si esegue la previsione. Nella prima immagine si deduce che la previsione, evidenziata in rosso, segue molto bene l'andamento reale della previsione, soprattutto nel tratto in cui la ddp.dc va oltre la soglia per diversi campioni consecutivi; infatti l'errore, in corrispondenza di tali valori, è prossimo allo zero. Nella seconda immagine, invece, è evidente come la previsione sia soggetta ad un leggero delay, ma l'errore di previsione commesso risulta sempre prossimo allo zero. Analizzando entrambi i grafici, si può osservare che il modello sembra non essere in grado di prevedere tratti in cui la differenza di potenziale ha brusche variazioni, a differenza dei tratti della serie temporale in cui la differenza tra un campione e l'altro è piuttosto contenuta.

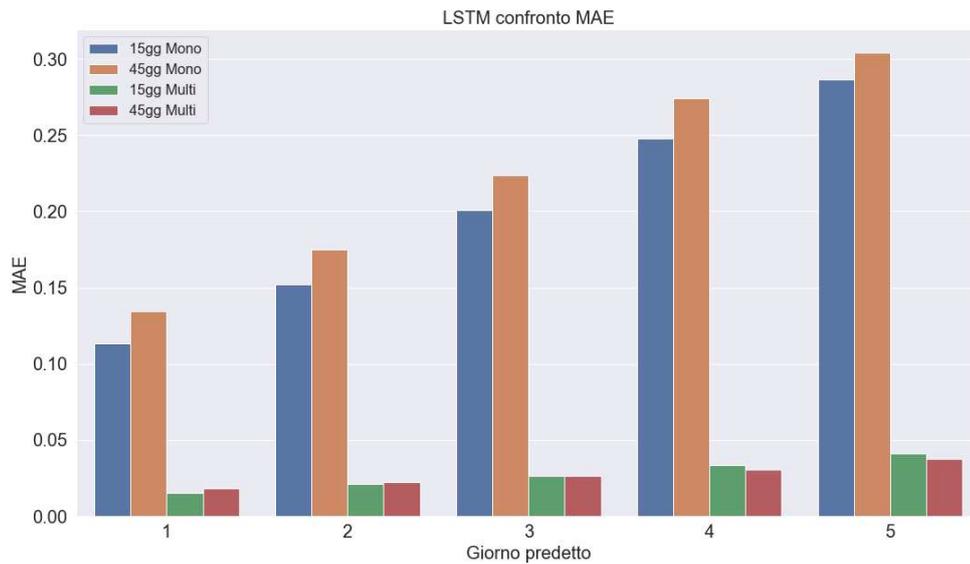


Figura 5.10: L'istogramma che illustra il MAE calcolato su ciascun giorno predetto, addestrando la rete per la ddp.dc con ID 8984 rispetto al modello univariabile e multivariabile, considerando, sul dataset iniziale, una finestra scorrevole di 15 e 45 giorni

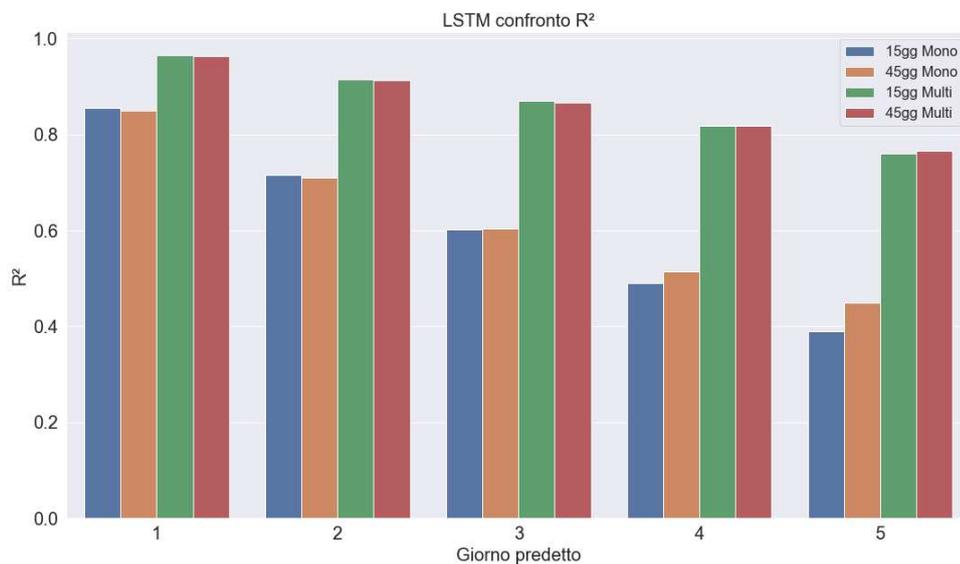


Figura 5.11: L'istogramma che illustra l' R^2 calcolato su ciascun giorno predetto, addestrando la rete per la ddp.dc con ID 8984 rispetto al modello univariabile e multivariabile, considerando, sul dataset iniziale, una finestra scorrevole di 15 e 45 giorni

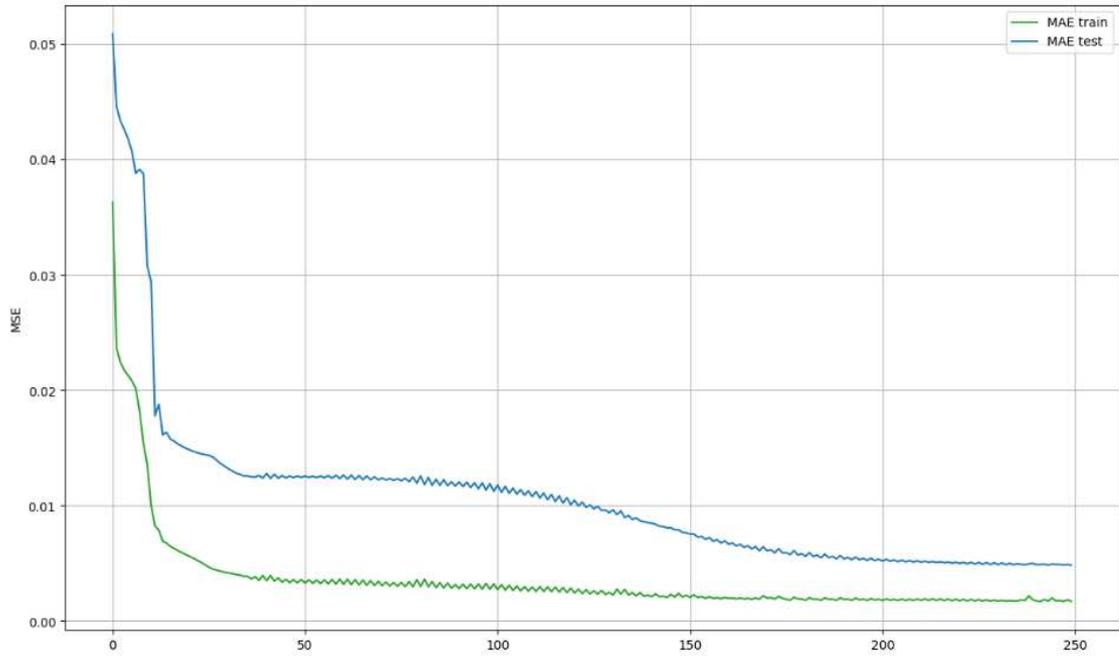


Figura 5.12: L'andamento dell'MSELoss durante il train ed il test effettuato su 250 epoche – modello multivariabile con finestra di 45 campioni

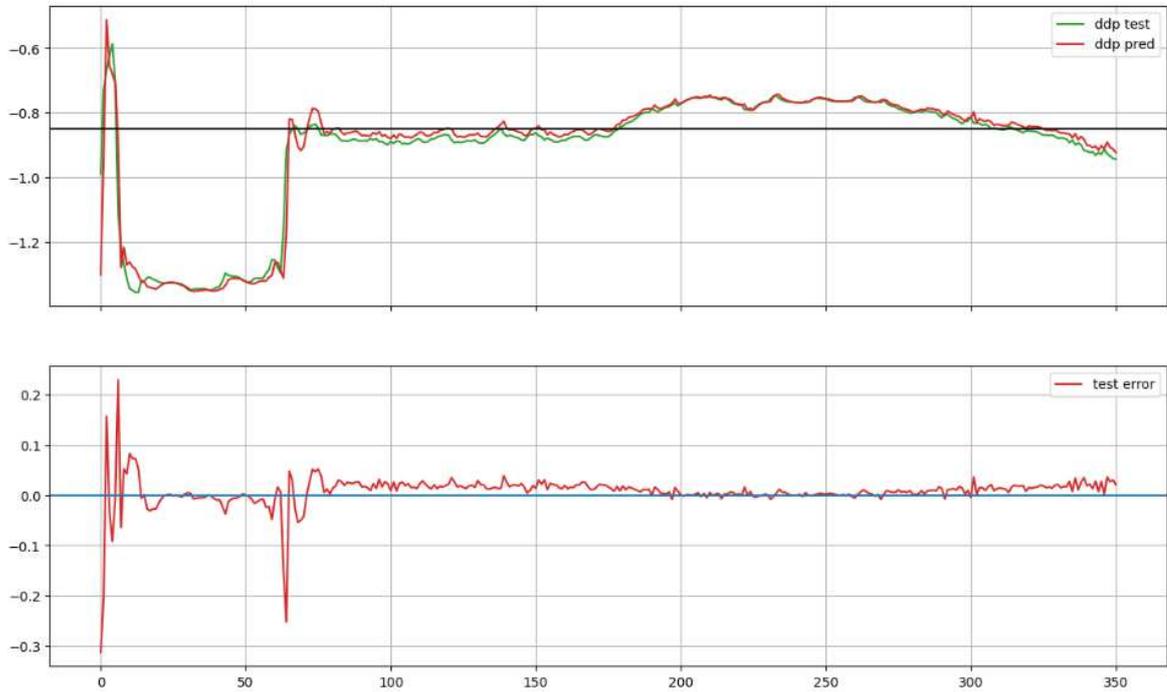


Figura 5.13: L'andamento nel tempo del primo giorno di previsione - modello multivariabile e finestra 45 giorni

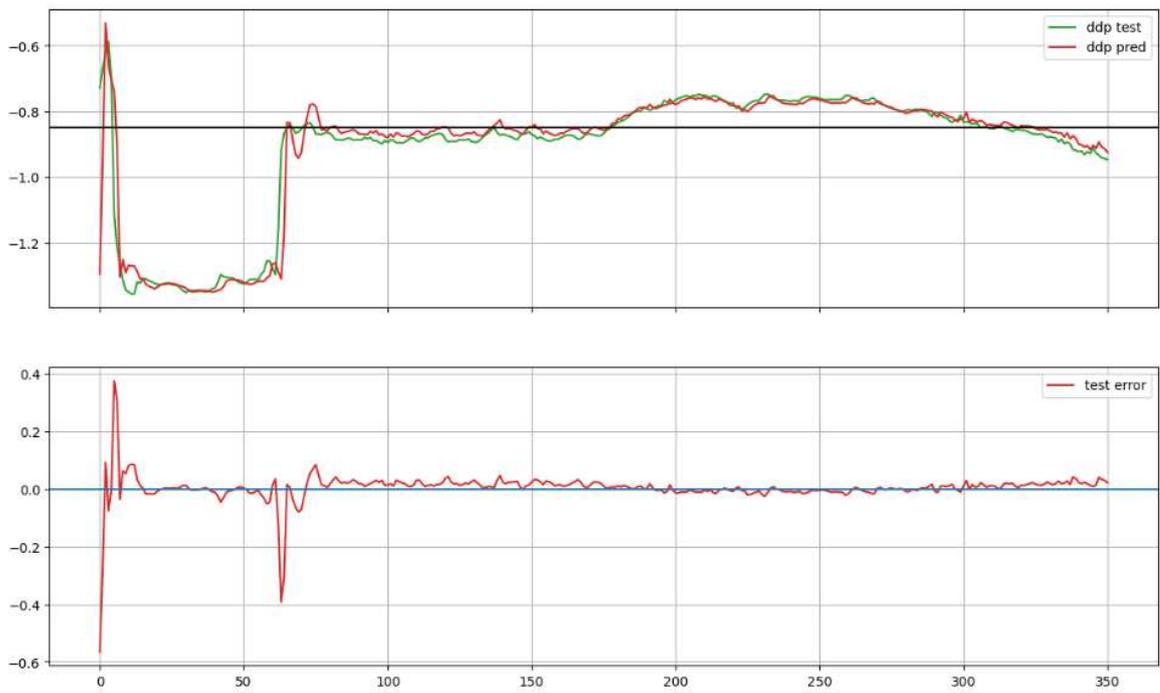


Figura 5.14: L'andamento nel tempo del secondo giorno di previsione - modello multivariabile e finestra 45 giorni

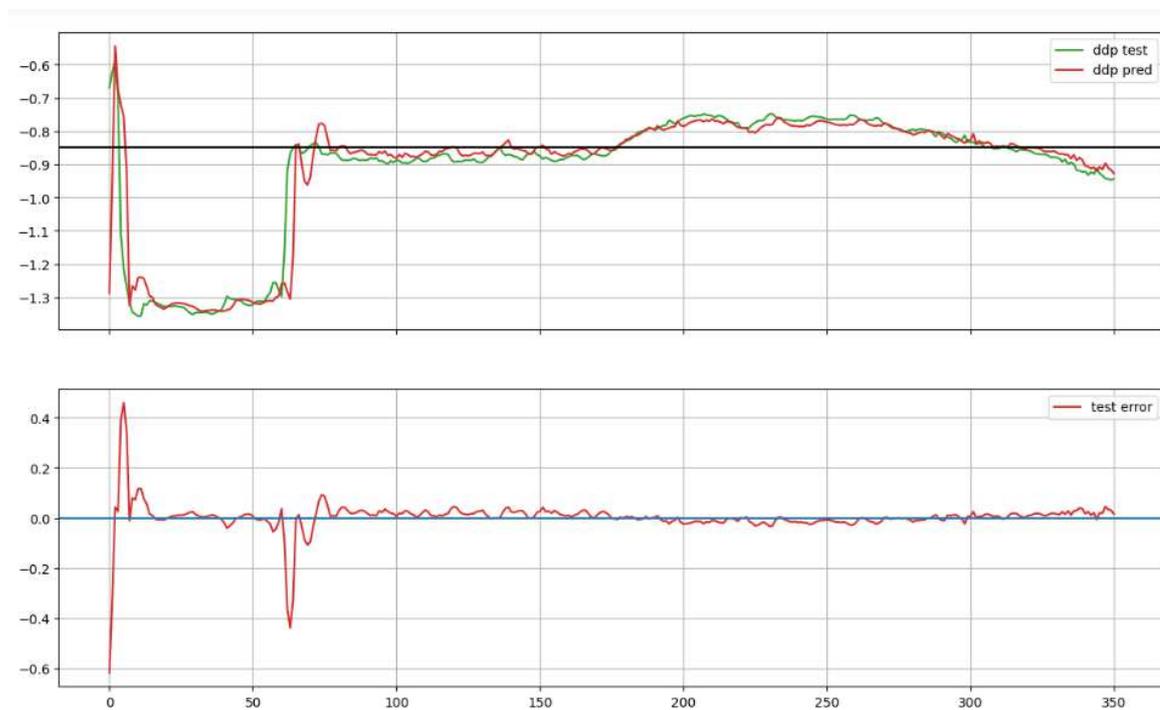


Figura 5.15: L'andamento nel tempo del terzo giorno di previsione - modello multivariabile e finestra 45 giorni

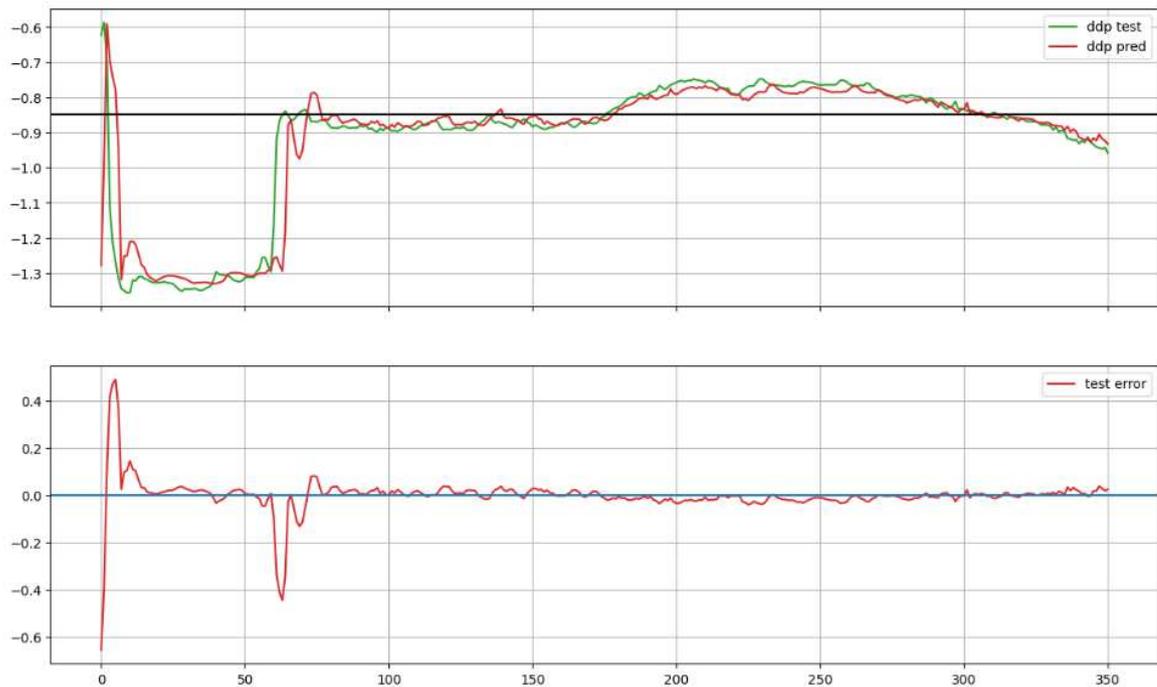


Figura 5.16: L'andamento nel tempo del quarto giorno di previsione - modello multivariabile e finestra 45 giorni

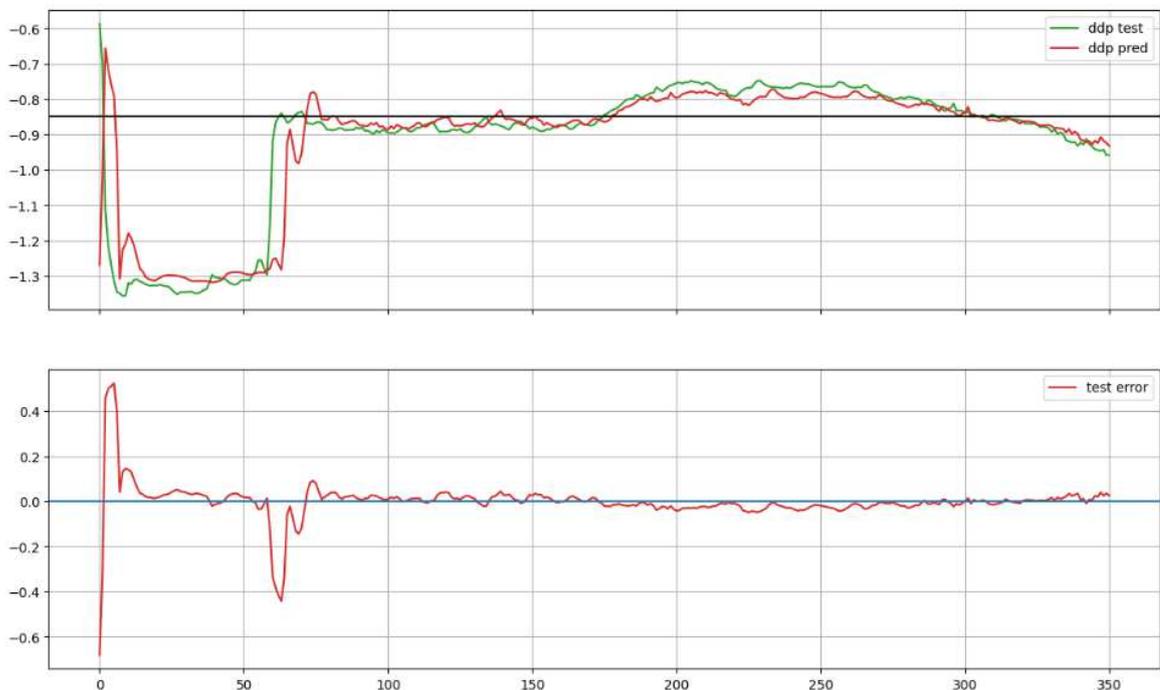


Figura 5.17: L'andamento nel tempo del quinto giorno di previsione - modello multivariabile e finestra 45 giorni

Infine, dopo aver classificato i valori precetti ed i valori reali come spiegato nel paragrafo 3.4, sono state calcolate, per il modello multivariabile (finestra 45 giorni) le metriche di Accuracy, Precision, Recall ed F1-score riportate in Figura 5.18, insieme alla relativa confusion matrix, in modo da verificare la sua capacità nel riconoscere le ddp.dc fuori soglia nel giorno

più distante da quello in cui viene effettuata la previsione. Si evidenzia che l'accuratezza per l'ultimo giorno è di circa l'90% e di tutti i dati fuori soglia il 4.8% dei campioni non è predetto correttamente.

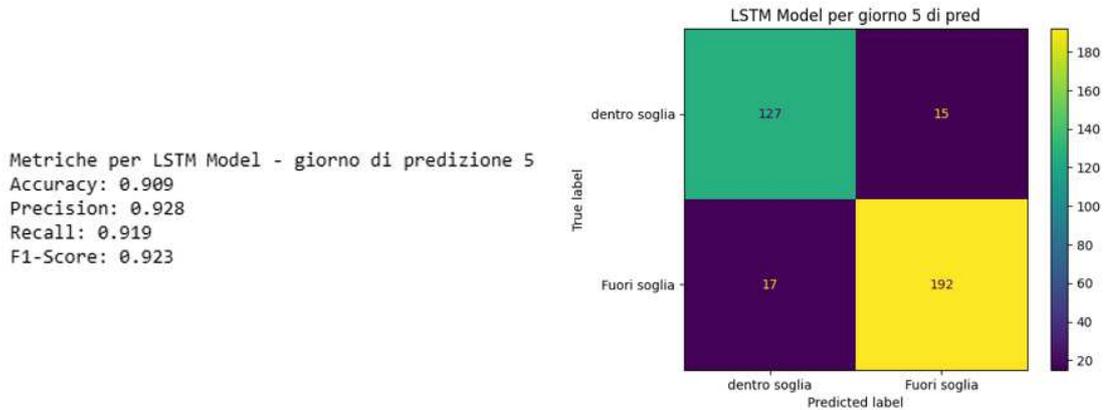


Figura 5.18: Le metriche e la confusion matrix relativa al quinto giorno di previsione per il modello multivariabile con finestra di 45 giorni

5.3 Confronto dei risultati delle ddp.dc

La ddp.dc con *ID 17271* rispetto alla ddp.dc con *ID 8984* presenta più dati fuori soglia e meno valori *NaN*, ma è descritta da una serie temporale costituita da meno campioni. Infatti, come precedentemente analizzato nel capitolo 2, la seconda ddp.dc ha circa cinque anni in più rispetto alla prima, ma sapendo che 604 dati sono dei *NaN* consecutivi, tali campioni non sono stati presi in considerazione per l'addestramento ed il test della rete. Di conseguenza, la ddp.dc con *ID 17271* ha circa quattro anni e mezzo in meno rispetto alla ddp.dc con *ID 8984*.

In Tabella 5.2 e Tabella 5.3 sono riportati i risultati ottenuti dai modelli migliori monovariati della rete *LSTM* per la ddp.dc con *ID 17271* e la ddp.dc con *ID 8984*; il primo ha performance migliori con una finestra di 45 ed il secondo con una finestra di 15. Confrontando i risultati, per ciascun sistema, si osserva che aumentando i campioni su cui si addestra la rete, non è detto che migliori la previsione, infatti, il secondo ha valori di *MAE* più alti e valori di R^2 più bassi rispetto al secondo.

Giorno	ddp.dc – ID 17271	ddp.dc – ID 8984
Modello	<i>Univariabile - 45 gg</i>	<i>Univariabile - 15 gg</i>
MAE - Giorno 1	0.062	0.113
MAE - Giorno 5	0.137	0.287

Tabella 5.2: Valori di *MAE* ottenuti dai modelli migliori monovariati della rete *LSTM* per la ddp.dc con *ID 17271* e la ddp.dc con *ID 8984*

In Tabella 5.4 e Tabella 5.5 sono riportati i risultati ottenuti dai modelli migliori (multivariati) della rete *LSTM* per la ddp.dc con *ID 17271* e la ddp.dc con *ID 8984*. Entrambi hanno ottime performance considerando una finestra di 45, ma è bene specificare che i risultati rispettivi di ciascun sistema sono paragonabili anche considerando una finestra di 15; pertanto,

Giorno	ddp.dc – ID 17271	ddp.dc – ID 8984
Modello	<i>Univariabile - 45 gg</i>	<i>Univariabile - 15 gg</i>
R^2 - Giorno 1	0.919	0.855
R^2 - Giorno 5	0.593	0.390

Tabella 5.3: I valori di R^2 ottenuti dai modelli migliori monovariati della rete *LSTM* per la ddp.dc con ID 17271 e la ddp.dc con ID 8984

Giorno	ddp.dc – ID 17271	ddp.dc – ID 8984
Modello	<i>Multivariabile - 45 gg</i>	<i>Multivariabile - 45 gg</i>
MAE - Giorno 1	0.063	0.018
MAE - Giorno 5	0.108	0.037

Tabella 5.4: Valori di *MAE* ottenuti dai modelli migliori della rete *LSTM* per la ddp.dc con ID 17271 e la ddp.dc con ID 8984, considerando una finestra scorrevole di 45 giorni

Giorno	ddp.dc – ID 17271	ddp.dc – ID 8984
Modello	<i>Multivariabile - 45 gg</i>	<i>Multivariabile - 45 gg</i>
R^2 - Giorno 1	0.857	0.964
R^2 - Giorno 5	0.609	0.767

Tabella 5.5: I valori di R^2 ottenuti dai modelli migliori della rete *LSTM* per la ddp.dc con ID 17271 e la ddp.dc con ID 8984, considerando una finestra scorrevole di 45 giorni

in questo caso, la scelta della finestra migliore risulta quasi indifferente. Probabilmente le performance di ciascun sistema, come nella rolling window, sono fortemente dipendenti dall'andamento nel tempo del punto di misura, in particolare, dalla presenza o meno di numerose escursioni di differenza di potenziale tra campioni consecutivi. Infatti, anche per questo non è possibile scegliere una finestra a priori per una qualsiasi ddp.dc di un qualsiasi sistema, perché appunto varia in base ai dati; pertanto, la finestra deve essere scelta ad hoc per ogni punto di misura. Inoltre, è bene osservare che, la seconda ddp.dc riporta un *MAE* più basso ed un R^2 più alto rispetto alla prima, ma questo, probabilmente, è dovuto sempre alla tipologia di dati testata; in questo caso specifico, nonostante si è provato a testare la rete su un set di dati il più diverso possibile da quello su cui è stato addestrato, i dati presentano, in questo caso dei pattern ricorrenti e di conseguenza non è possibile garantire la generalizzazione della rete, soprattutto se l'andamento riporta dei picchi. Questo potrebbe spiegare anche il motivo per cui il primo sistema ha buone performance, ma comunque inferiori rispetto alle performance del secondo: i dati della ddp.dc con ID 17271 hanno una forte variabilità rispetto alla ddp.dc con ID 8984.

Nella presente tesi è stata eseguita un'analisi predittiva per conto dell'azienda *Automa s.r.l.*, la quale si occupa di monitoraggio remoto in ambito Oil e Gas. E' stato analizzato e, successivamente predetto, l'andamento temporale della differenza di potenziale misurata (*ddp.dc*) in uno specifico punto di un sistema di *protezione catodica*. In questo studio ci si è concentrati sui sistemi di protezione *a corrente impressa*, ovvero su una rete metallica interrata, predisposta in una determinata area geografica, gestita da uno o più alimentatori che erogano una corrente affinché il sistema sia protetto, ovvero, affinché tutti i punti di misura abbiano una differenza di potenziale al di sotto di $-0,85$ V. Tali informazioni sono raccolte in uno storico dati reperibile dalla loro piattaforma aziendale *WebProcat* a cui si è fatto riferimento per lo sviluppo di tale progetto; in particolare, i dati soggetti a tale studio appartengono all'azienda *Centrias.r.l.*, una società che gestisce il vettoriamento di gas naturale, nonché cliente di *Automa s.r.l.*

Il primo passo per lo sviluppo del seguente elaborato è stato lo studio del contesto di applicazione, in particolare del generale funzionamento di un sistema di protezione catodica *a corrente impressa* e degli strumenti utilizzati dall'azienda per il monitoraggio della stessa. Dunque, dopo aver compreso il problema legato alla corrosione e la tecnica elettrochimica utilizzata per il suo rallentamento, è stato studiato il meccanismo di gestione di un sistema di protezione catodica, concentrandosi sul controllo degli alimentatori di corrente effettuato da *Automa s.r.l.*

Successivamente, sono stati analizzati i dati della società *Centria s.r.l.* tramite il linguaggio di programmazione *Python*, sulla piattaforma *Jupyter Notebook*. in questa fase sono stati filtrati tutti i sistemi con un solo alimentatore, zero drenaggi o attraversamenti ferroviari, in modo da potersi concentrare su sistemi semplici soggetti il meno possibile a correnti interferenti. In seguito sono stati presi in considerazione tutti quei sistemi che riportano una differenza di potenziale al di sopra di -0.85 V per almeno 15 giorni consecutivi. Tale filtraggio è stato eseguito per porre maggiore attenzione su sistemi con possibili guasti all'alimentatore ed i sistemi presi in esame sono l'*MV11* ed il *S04ML*, i quali hanno, rispettivamente, una *ddp.dc* con molti punti di misura fuori soglia. Il primo riporta la *ddp.dc* con *ID 17271* con molti *eventi prevedibili* ma dati a partire, solo, dal 2018, mentre, il secondo riporta la *ddp.dc* con *ID 8984* con meno *eventi prevedibili* rispetto alla precedente ma con campioni a partire dal 2013. Queste *ddp.dc* sono state quelle soggette alla previsione.

L'obiettivo è stato quello di prevedere i 5 giorni successivi delle *ddp.dc* considerate, rispetto al giorno in cui è effettuata la previsione; ciascuna differenza di potenziale è stata

predetta separatamente. Per il raggiungimento dello scopo sono state intraprese due strade tramite l'implementazione di due metodologie differenti di seguito riportate.

- *La rolling - window*: una previsione run - time, dunque locale, in cui si sfrutta la semplice relazione tra la ddp.dc e l'istante di campionamento; secondo questo metodo si fa scorrere una finestra temporale di lunghezza fissa su tutta la lunghezza della serie temporale e, ad ogni iterazione, su di essa è eseguita l'operazione di *fit*, in modo da prevedere i dati successivi tramite il "prolungamento" della funzione ottenuta.
- *La rete neurale lstm*: una previsione basata su un modello generale, tale per cui, sfruttando lo storico dati, è stato possibile addestrare una rete, con una specifica struttura, iperparametri e su di un determinato numero di epoche, per ciascuna ddp.dc considerata; inoltre, per ciascuna ddp.dc è stato addestrato un modello univariabile, considerando solo la misura di differenza di potenziale ed un modello multivariabile, considerando sia la differenza di potenziale che la corrente di alimentazione.

Una volta ottenuti i risultati delle relative previsioni, per ciascuna metodologia, sono state confrontate le metriche ottenute sulle differenti ddp.dc. Dunque, si è potuto constatare che, sia per la ddp.dc con ID 17271 che per la ddp.dc con ID 8984, la rolling window ha ottime performance sfruttando il modello *Voting Regressor* su una finestra di 30 giorni. Tale modello effettua una combinazione pesata del modello *Linear Regression* (*loss function* : OLS) e modello SVR con kernel radiale (*loss function*: ϵ -insensitive loss). Mentre, sia per la ddp.dc con ID 17271 che per la ddp.dc con ID 8984, il modello migliore risulta essere il multivariabile, considerando una finestra scorrevole di 45 campioni; ma, a differenza del primo metodo, questo ha un modello customizzato per ciascuna ddp.dc, infatti, si è riscontrato che non si avrebbero le migliori performance se venisse utilizzata, per ciascuna ddp.dc, una rete con lo stesso *hidden layer*, numero di *layer*, *learning rate* ed epoche.

Gli approcci utilizzati per la stima della ddp.dc sono entrambi validi. In particolare, l'approccio basato sulla *rolling window* è più semplice, può essere utilizzato anche senza avere uno storico dati ed il modello su cui si effettuerà il *fit* su ciascuna finestra scorrevole sarà sempre il *Voting Regressor*. Il limite, però, di questo approccio è che, quando si registreranno, tra un giorno e l'altro, escursioni di ddp.dc moderatamente alte, in quel caso la previsione sarà sempre soggetta ad un errore più o meno grande in base ai valori dei campioni stessi nella finestra temporale. L'approccio basato su rete neurale LSTM, necessita di dati storici e si dimostra comunque efficace ma, per ciascuna ddp.dc, è necessario addestrare un modello ad hoc e, dunque, è necessario effettuare un fine tuning per ciascun punto di misura di ciascun sistema. Se si aumentasse la granularità dei dati, quindi, se si effettuassero campionamenti più fitti, e se si introducessero informazioni aggiuntive correlate alla variazione di ddp.dc, come dati meteorologici o resistività del terreno, sicuramente entrambi gli approcci potrebbero restituire statistiche migliori. Con tali dati a disposizione si potrebbe valutare l'implementazione di un'unica rete per più punti di misura, in quanto, questi potrebbero aiutare la generalizzazione del modello. Inoltre, se si avessero a disposizione più informazioni legate al processo corrosivo ed una granularità maggiore nei dati, la previsione potrebbe essere messa in atto anche per sistemi più complessi, ovvero sistemi con più alimentatori, ma questo necessiterebbe anche di informazioni topologiche approfondite per comprendere al meglio la direzionalità della corrente.

A valle di queste analisi sviluppate, viste le potenzialità emerse, l'azienda ha deciso di muoversi verso campionamenti più fitti e non più giornalieri; questo sicuramente permetterà di migliorare entrambi gli approcci. Ad ogni modo, l'approccio *rolling window* verrà sfruttato per eseguire le future previsioni di dati fuori soglia e sarà adottato anche per altri loro progetti di intelligenza artificiale. Come precedentemente anticipato, essendo un modello

locale, necessita solo dei dati più recenti ed, attualmente, è la metodologia più adatta alle loro esigenze ed ai dati disponibili al momento. Concluso tale progetto, si può, dunque, affermare di aver ottenuto un buon punto di partenza per trasformare *l'azione correttiva*, attualmente eseguita in seguito a rilevamenti di differenza di potenziale fuori soglia, in un'*azione preventiva*.

- BI, L., TSIMHONI, O. e LIU, Y. (2011), «Using the Support Vector Regression Approach to Model Human Performance», *ACM Transactions on Internet Technology*. (Cited at pages iv e 30)
- BROWNLEE, J. (2020), *Data preparation for machine learning – data cleaning, feature selection, and data transforms in python*, Machine Learning Mastery. (Cited at page 55)
- CORTES, C. e VAPNIK, V. (1995), «Support-vector networks», *International Journal of Web-Based Learning and Teaching Technologies*. (Cited at page 30)
- DRUCKER, H., BURGESS, C. J. C., KAUFMAN, L. e SMOLA, A. J. (1997), «Support vector regression machines», *International Journal of Applied Mathematics and Computer Science*. (Cited at page 29)
- HOCHREITER, S. e SCHMIDHUBER, J. (1997), «Long Short-Term Memory», *Neural Computation*.
- INTERNATIONAL ORGANIZATION FOR STANDARDIZATION (2015), «ISO 15589-1: Petroleum, petrochemical and natural gas industries – Cathodic protection of pipeline systems», Standard published by the International Organization for Standardization. (Cited at page 10)
- ISLAM, M., CHEN, G. e JIN, S. (2019), «An Overview of Neural Network», *Neural Networks*. (Cited at pages iv e 32)
- JAMES, G., WITTEN, D., HASTIE, T. e TIBSHIRANI, R. (2021), *An Introduction to Statistical Learning with Applications in R*, Springer.
- JOHNSON, K. e KUHN, M. (2013), *Data preparation for machine learning – data cleaning, feature selection, and data transforms in python*, Springer. (Cited at page 34)
- KUMAR, I., TRIPATHI, B. K. e SINGH, A. (2022), «Attention-based LSTM network-assisted time series forecasting models for petroleum production», *Engineering Applications of Artificial Intelligence*. (Cited at page 32)
- LAZZARRI, L., PEDEFERRI, P. e ORMELLESE, M. (2006), *Protezione catodica*, Polipress. (Cited at pages iii, 9, 10 e 11)
- PEDEFERRI, P. (2010), *Corrosione e Protezione dei Materiali Metallici*, Polipress. (Cited at pages iii, 4, 7 e 9)

- PEDEFERRI, P. (2018), *Corrosion Science and Engineering*, Springer. (Cited at page 6)
- PEDEFERRI, P. e LAZZARI, L. (2006), *Cathodic protection*, Polipress. (Cited at pages iii e 3)
- RAO, C. R. e TOUTENBURG, H. (1995), *Linear Models Least Squares and Alternatives*. (Cited at page 27)
- REVIE, R. W. (2016), *Uhlig's Corrosion Handbook*, Wiley. (Cited at page 8)
- ROBERGE, P. R. (2008), *Corrosion Engineering*, McGraw-Hill. (Cited at page 4)
- SCHÖLKOPF, B. e SMOLA, A. J. (2001), *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*, The MIT Press.
- SMOLA, A. J. e SCHÖLKOPF, B. (2004), «A tutorial on support vector regression», *Journal of Intelligent & Fuzzy Systems*. (Cited at page 29)
- STRUTZ, T. (2016), *Data Fitting and Uncertainty A practical introduction to weighted least squares and beyond*, Springer. (Cited at page 26)
- VAPNIK, V. N. (1995), *The Nature of Statistical Learning Theory*, Springer.
- WOODITCH, A., JOHNSON, N. J., SOLYMOSI, R., ARIZA, J. M. e LANGTON, S. (2021), *Ordinary Least Squares Linear Regression*, Springer. (Cited at page 27)
- ZHOU, Z.-H. (2012), *Ensemble Methods Foundations and Algorithms*, CRC Press. (Cited at page 31)
- ZIVOT, E. e WANG, J. (2006), *Modeling Financial Time Series with S-PLUS*, Springer. (Cited at page 26)

Siti Web consultati

- Automa s.r.l., il portale dell'azienda – <https://www.byautoma.com/>
- WebProcat, la piattaforma di Automa s.r.l. per la raccolta e gestione dei dati – <https://www.webprocat.com>
- G4C-Pro, il dispositivo di misurazione dell'azienda – <https://www.goliah.info/protezione-catodica/>
- Python – <https://python.org/>
- Numpy – <https://numpy.org/>
- Pandas – <https://pandas.org/>
- pyodbc – <https://pypi.org/project/pyodbc/>
- Matplotlib – <https://matplotlib.org/>
- Seaborn – <https://seaborn.pydata.org/>
- scikit-learn – <https://scikit-learn.org/stable/>