



UNIONE EUROPEA
Fondo Sociale Europeo
Fondo Europeo di Sviluppo Regionale



Avviso 1735 del 13.07.2017 MIUR

Progetti di Ricerca Industriale e Sviluppo Sperimentale nelle 12 Aree di Specializzazione individuate dal PNR 2015-2020

Modelli previsionali di carico in ottica DR

Rapporto Tecnico di Ricerca Industriale D5.3b



Avviso	Avviso 1735 del 13.07.2017 MIUR
Codice progetto	ARS01_01259
Nome del progetto	Community Energy Storage Gestione Aggregata di Sistemi di Accumulo dell'Energia in Power Cloud
Acronimo	ComESto
Documento	D5.3b
Tipologia	Rapporto Tecnico
Data di Rilascio	08/06/2020
Obiettivo Realizzativo	OR5
Attività Realizzativa	AR5.3
Soggetti Beneficiari Proponenti	UNICAL, TEN, ENEA
Elaborato (Nome, Cognome – Soggetto Beneficiario)	Martina Caliano, Amedeo Buonanno, Antonino Pontecorvo, Gianluca Sforza - ENEA
Verificato (Nome, Cognome – Soggetto Beneficiario)	Alfonso Senatore, Nicola Sorrentino - UNICAL
Approvato (Nome, Cognome – Soggetto Beneficiario)	Membri del PEB

Indice

1.INTRODUZIONE E OBIETTIVO DI PROGETTO	6
2.ANALISI DEI REQUISITI	7
3.METODI	7
3.1. STATO DELL'ARTE	7
3.2. TEST SU UN DATASET DI LETTERATURA.....	8
3.3. ANALISI DEI DATASET RELATIVI ALLE UTENZE UNICAL	10
3.3.1. UtENZE di tipo domestico.....	11
3.3.2. UtENZE di tipo non domestico	16
3.3.2.1. SM 253 – Falegnameria	16
3.3.2.2. SM 103 – Piccolo hotel	17
3.3.2.3. SM 262 – Software house	17
3.4. MODELLI E METODI APPLICATI.....	17
3.4.1. Persistenza.....	17
3.4.2. Regressione lineare	18
3.4.3. FFNN	18
3.4.4. CNN	19
3.4.5. LSTM.....	20
3.4.6. XGBoost	22
3.4.7. SVR.....	22
3.5. MODELLI SVILUPPATI PER LA PREVISIONE DEL CARICO A 1 H.....	24
3.5.1. Persistenza.....	24
3.5.2. Regressione lineare	24
3.5.3. FFNN	26
3.5.4. CNN	28
3.5.5. LSTM.....	29
3.5.6. XGBoost	30
3.5.7. SVR.....	31
3.6. MODELLI SVILUPPATI PER LA PREVISIONE DEL CARICO A 24 H.....	32
3.6.1. Regressione lineare	32
3.6.2. FFNN	33

3.6.3. XGBoost	34
4.RISULTATI.....	35
4.1. RISULTATI DEI TEST EFFETTUATI SUL DATASET DI LETTERATURA	36
4.2. RISULTATI DELLA PREVISIONE A 1H	36
4.2.1. Utenze di tipo domestico.....	37
4.2.1.1. SM 3.....	37
4.2.1.2. SM 112	38
4.2.1.3. SM 232	40
4.2.2. Utenze di tipo non domestico	41
4.2.2.1. SM 253 – Falegnameria	41
4.2.2.2. SM 103 – Piccolo hotel	42
4.2.2.3. SM 262 – Software house	43
4.3. RISULTATI DELLA PREVISIONE A 24H	44
4.3.1. Utenze di tipo domestico.....	44
4.3.1.1. SM 3.....	44
4.3.1.2. SM 112	44
4.3.1.3. SM 232	45
4.3.2. Utenze di tipo non domestico	46
4.3.2.1. SM 253 – Falegnameria	46
4.3.2.2. SM 103 – Piccolo hotel	46
4.3.2.3. SM 262 – Software house	46
5.DESCRIZIONE DEL TOOL.....	47
6.DISCUSSIONE E CONCLUSIONI	49
7.BIBLIOGRAFIA.....	51

Indice delle figure

Figura 1. Forecasting dipendente da: (a) tipologia di edificio; (b) orizzonte temporale di previsione; (c) tipo di carico previsto; (d) utilizzo di dati misurati, simulati o di dataset pubblici [22]	8
Figura 2. Consumi medi orari per i tre casi residenziali relativi ad una settimana	11
Figura 3. Valori medi orari (blu) e giornalieri (verde) della potenza elettrica consumata e i <i>missing data</i> sostituiti con i template weekday-weekend (arancione) nell'intero periodo esaminato per lo SM 112	12
Figura 4. Andamento del consumo di potenza in ciascun mese nel periodo osservato per lo SM 112.....	13
Figura 5. Istogrammi per ciascun mese nel periodo osservato per lo SM 112	13
Figura 6. Template rispettivamente sull'intera settimana, solo sui giorni feriali, solo sui weekend per lo SM 112	14
Figura 7. Istogramma della distribuzione dei consumi per lo SM 112	14
Figura 8. Box e whisker plot (diagramma a scatola e baffi) dei consumi della settimana, dal lunedì (mon) alla domenica (sun) per lo SM 112.....	15
Figura 9. Distribuzione dei consumi su ciascuna ora della giornata per lo SM 112	15
Figura 10. Consumi medi orari per i tre casi non residenziali relativi ad una settimana	16
Figura 11. Schema di una generica rete di tipo <i>feed forward con un solo livello intermedio</i>	19
Figura 12. Esempio di Livello Convolutivo [38]	20
Figura 13. Esempio di Pooling [38]	20
Figura 14. Schema di massima di una RNN nella sua forma <i>folded</i> (sinistra) e <i>unfolded</i> (destra) [40].....	21
Figura 15. LSTM Cell [42]	21
Figura 16. Esempio di regressione lineare con indicazione della Zona Epsilon-Insensitive [47].....	23
Figura 17. Scatterplot dello storico al tempo t , $t-1$, $t-2$ per lo SM 253: (A) vista 3D completa; (B) zoom in 2D ...	26
Figura 18. (A) Struttura del modello con due variabili di input; (B) Struttura del modello con 7 variabili di input	27
Figura 19. Schema di funzionamento di un modello ricorsivo	34
Figura 20. Potenza elettrica assorbita effettiva e prevista nel caso dello SM 112 ottenuta con il metodo XGBoost	39
Figura 21. Potenza elettrica assorbita effettiva e prevista nel caso dello SM 112 ottenuta con il metodo FFNN	45
Figura 22. Schema a blocchi relativo alle fasi eseguite nel tool di forecasting del carico.....	47

1. INTRODUZIONE E OBIETTIVO DI PROGETTO

Nel presente rapporto tecnico sono descritte le attività di ricerca svolte dal gruppo di lavoro ENEA impegnato, nell'ambito dell'AR 5.3 del progetto di ricerca ComESTo, nella definizione di modelli di previsione del carico. L'obiettivo di tale linea di attività è stato quello di sviluppare dei modelli di previsione, su diverse scale temporali, dei carichi elettrici associati a sei utenze italiane di diverso tipo, i cui consumi sono in corso di misurazione da parte dell'Università della Calabria (UNICAL), dal mese di gennaio 2019. Queste, nel dettaglio, fanno riferimento a utenze domestiche e non, di tipo *consumer* o *prosumer*. L'attività ha permesso, quindi, la previsione del carico a 1h e a 24 ore, consentendo di avere sia previsioni dell'intero profilo giornaliero (day-ahead), sia previsioni del carico nelle 24 ore successive all'ora corrente.

Al fine di poter sviluppare opportuni modelli di previsione del carico elettrico, l'attività ha previsto una prima fase di ricerca e analisi di dataset di consumo pubblici, che ha condotto all'identificazione di un dataset di consumo riferito ad un edificio di un campus universitario statunitense. Rispetto a questo, sono stati sviluppati tre modelli di previsione del carico e sono stati analizzati i primi risultati, al fine di poter creare uno scenario di riferimento utile alla validazione dei modelli sviluppati.

Successivamente, è stata effettuata un'esplorazione del database di consumo messo a disposizione da UNICAL. La fase di studio dei dataset ha evidenziato molteplici problematiche connesse alla campagna di misurazione condotta, evidenziando per ciascuna utenza, buchi di misura, dati duplicati e/o mancanti; pertanto l'attività ha richiesto una fase di risoluzione di tali problematiche, mirata e particolareggiata per ciascuna utenza, al fine di poter effettuare una previsione del carico più efficace e affidabile. L'analisi ha, inoltre, mostrato una forte variabilità dei carichi misurati, soprattutto nel caso dei dataset di consumo riferiti alle utenze di tipo domestico, a conferma della spiccata variabilità dei carichi su scala residenziale, derivante, innanzi tutto, dal comportamento degli utenti, spesso imprevedibile e non dipendente da fattori comuni.

Contemporaneamente all'analisi dei dataset di consumo, l'attività si è concentrata, quindi, sullo sviluppo dei modelli di previsione del carico, a 1h e a 24 h, delle sei utenze considerate. Nel dettaglio, il forecasting del carico a 1h è stato eseguito mediante lo sviluppo di 11 modelli di diverso tipo, basati su 7 metodi statistici classici o di *machine learning*, al fine di poter effettuare un confronto tra i vari modelli sviluppati e identificare, tra questi, quello più opportuno ai fini del forecasting dello specifico caso analizzato. Gli approcci utilizzati per lo sviluppo di tali modelli sono stati, quindi, il metodo della persistenza; regressione lineare; reti neurali di tipo *feed forward*, convoluzionale e ricorrente (Long Short Term Memory network); il metodo dell'eXtreme Gradient Boosting e infine il Support Vector Regression.

A causa dell'elevato costo computazionale, la previsione del carico a 24 h è stata effettuata scegliendo, tra i vari modelli considerati, gli approcci mediamente più performanti nel caso della previsione del carico a 1 h, ovvero il modello di regressione lineare, l'eXtreme Gradient Boosting e il modello di rete neurale *feed forward*. Nel dettaglio, i primi due sono metodi ricorsivi, i quali, pertanto, consentono sia la previsione day-ahead, sia la previsione del carico delle 24 ore successive a quella corrente. Il modello di rete neurale *feed forward* implementato permette, invece, solo la previsione day-ahead, rendendola disponibile alla prima ora del giorno da prevedere. Tuttavia, variando opportunamente le variabili di input, la previsione day-ahead potrà essere fornita in tempo utile ai fini della piattaforma ComESTo per la gestione della comunità.

Il presente report risulta così suddiviso: il capitolo 2 riporta l'analisi dei requisiti, il capitolo 3 la descrizione di tutta l'attività svolta, inerente allo stato dell'arte, all'analisi sul dataset di letteratura, all'esplorazione dei dataset di consumo forniti e allo sviluppo dei modelli a 1h e a 24 h; il capitolo 4 riporta i risultati ottenuti per tutti i casi analizzati e il capitolo 5 riporta la descrizione del tool sviluppato. Infine, il capitolo 6 riporta le principali conclusioni.

2. ANALISI DEI REQUISITI

Per la previsione dei carichi elettrici assorbiti dalle singole utenze, ottenuta mediante i codici di calcolo implementati nell'ambito del presente progetto di ricerca, è stata necessaria l'acquisizione dei seguenti dati di input:

- potenza assorbita per n ore precedenti a quella oggetto della previsione, con n dipendente dal modello utilizzato;
- dati meteo, quali temperatura, umidità, velocità del vento ed irraggiamento solare, i quali, in generale, possono aiutare il modello di calcolo a prevedere in modo più accurato la potenza assorbita dalla singola utenza;
- data (giorno, mese, anno) e ora di acquisizione, sia dei carichi elettrici che dei dati meteo, in quanto il carico elettrico assorbito può dipendere fortemente sia dal giorno della settimana (lavorativo, week-end, festivo), sia dall'ora in cui si desidera effettuare la previsione.

Per quanto concerne le risorse di calcolo necessarie per la realizzazione delle simulazioni è stato utilizzato un processore Intel(R) Core (TM) i7-4790s con una CPU di 3.20GHz, con una memoria installata (RAM) di 8,00 GB e sistema operativo Windows 10 a 64 bit.

3. METODI

Di seguito è riportato un breve stato dell'arte relativo all'utilizzo delle tecniche di forecasting del carico più utilizzate in letteratura. A seguire, è mostrata l'analisi svolta con riferimento ad un dataset di consumo riferito ad un edificio di un campus universitario statunitense, e successivamente sono riportati la descrizione dei dataset di consumo oggetto della previsione prevista in questo AR, degli approcci utilizzati e dei modelli di previsione del carico sviluppati.

3.1. STATO DELL'ARTE

Il forecasting dei profili di carico degli edifici, basato su un approccio data-driven, ha avuto negli anni una grande diffusione in quanto, consentendo un adattamento ottimale della domanda e dell'offerta di energia, può favorire, tra l'altro, una maggiore penetrazione delle fonti rinnovabili nei sistemi energetici. Sono molti, infatti, gli autori che hanno analizzato e applicato varie tecniche di forecasting del carico ad edifici di tipo residenziale [1] [2] [3] [4] e non residenziale [5] [6] [7] [8]; utilizzando orizzonti temporali differenti: orario [9] [10], sub-orario [11] [12], mensile e annuale [13]; utilizzando dati simulati [14] [15] o dati reali [16] [17]; effettuando previsioni su diversi tipi di carico: elettrico, di riscaldamento e raffrescamento o combinazione di questi [18] [19] [20] [17] [21]. La Figura 1 mostra la percentuale di sviluppo dei modelli di forecasting, dipendente dalla tipologia di edificio; dall'orizzonte temporale di previsione; dal tipo di carico previsto; dall'utilizzo di dati misurati, simulati o di dataset pubblici.

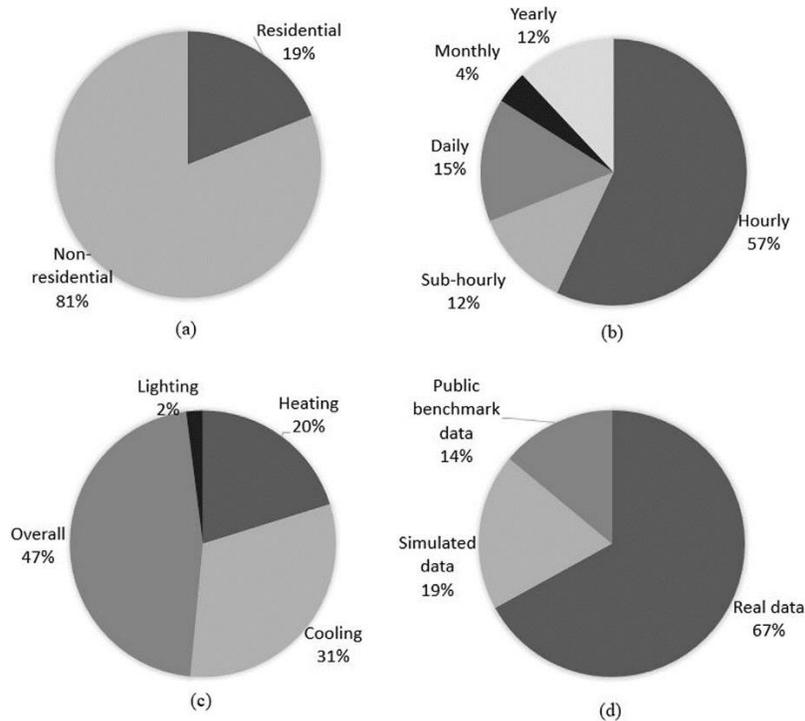


Figura 1. Forecasting dipendente da: (a) tipologia di edificio; (b) orizzonte temporale di previsione; (c) tipo di carico previsto; (d) utilizzo di dati misurati, simulati o di dataset pubblici [22]

Il settore della ricerca relativo alla previsione dei carichi ha visto la diffusione di diversi metodi, approssimativamente classificabili in metodi convenzionali o basati sull'intelligenza artificiale [23]. Sebbene i secondi siano nettamente più utilizzati, i metodi convenzionali sono in grado di fornire risultati soddisfacenti nel caso di problemi lineari, e pertanto, sono spesso utilizzati ai fini del forecasting del carico come modelli baseline [24]. Tra questi, vi sono i metodi statistici basati sulla regressione [22-27]. Tra i metodi basati sull'intelligenza artificiale, invece, quelli senza dubbio più diffusi, sono le reti neurali [23] [25] e tra queste, le reti neurali *feed forward* (FFNN) [26] [27] [28], le reti convoluzionali (CNN) [29], e le reti ricorrenti (LSTM) [30]. Sebbene le reti neurali siano tra le più utilizzate, il metodo del Support Vector Regression (SVR), in alcuni contesti residenziali, ha condotto a risultati mediamente migliori [2]. Sono diversi gli autori che hanno utilizzato tale metodo ai fini del forecasting del carico elettrico degli edifici [31] [32]. Infine, un ulteriore metodo, recentemente utilizzato per problemi di Short Term Load Forecasting è il metodo dell'eXtreme Gradient Boosting (XGBoost) [33], in precedenza utilizzato per previsioni in altri contesti.

3.2. TEST SU UN DATASET DI LETTERATURA

Il primo step dell'attività svolta ha previsto una fase di ricerca e analisi di dataset di consumo presenti in letteratura, al fine di poter effettuare dei test sui primi modelli sviluppati. La ricerca ha evidenziato l'assenza di dataset di consumo pubblici relativi ad utenze sia commerciali che residenziali poste sul territorio italiano, e pertanto, i primi test sono stati eseguiti su un dataset di consumo relativo ad un edificio di un campus universitario del Texas (USA). Nel dettaglio, tali test sono stati eseguiti utilizzando il dataset pubblico *ASHRAE Great Energy Predictor Shootout I* (ASHRAE 93) [34] [35], elaborato e fornito dalla American Society of Heating, Refrigeration and Air Conditioning Engineers. Tale dataset raccoglie, con frequenza oraria, dati riconducibili alle

grandezze meteo del luogo e i consumi elettrici e termici assorbiti dall'edificio per un periodo che intercorre tra settembre 1989 e febbraio 1990. Nel dettaglio, i dati forniti sono i seguenti:

- temperatura esterna;
- velocità del vento;
- grado di umidità;
- irradianza solare;
- ora del giorno;
- carico elettrico assorbito dall'edificio;
- carico frigorifero assorbito dall'edificio;
- carico di riscaldamento assorbito dall'edificio.

L'analisi è stata eseguita considerando come riferimento il lavoro di Edward et al. [2]. Pertanto, in linea con la metodologia seguita nel suddetto lavoro, l'attività è stata impostata suddividendo il dataset in *training set* e *test set*. Il *training test* è stato costruito considerando i dati misurati, con cadenza oraria, tra settembre e dicembre 1989, ovvero considerando 2926 record, mentre il *test set* è stato costruito considerando i dati misurati, con cadenza oraria, nei restanti 54 giorni, ovvero con 1282 record. Basandosi su tale dataset, sono stati realizzati quattro diversi modelli. I primi tre, basati su reti neurali di tipo *feed forward*, sono stati implementati mediante ambiente Matlab, mentre il modello basato sulla regressione lineare, mediante l'utilizzo della libreria Python Scikit-learn.

In una prima fase, sono stati realizzati due modelli, uno basato su una regressione lineare, l'altro su rete neurale di tipo *feed forward*, entrambi costruiti considerando due sole variabili di input, ovvero il carico elettrico assorbito nelle due ore precedenti, ossia negli istanti di tempo $t-1$ e $t-2$ e richiedendo al modello come output il carico elettrico assorbito dallo user all'ora corrente, ossia all'istante di tempo t .

Nel caso di utilizzo di reti neurali di tipo *feed forward*, al fine di prevedere il carico all'istante di tempo t sono stati sviluppati due ulteriori modelli basati rispettivamente sull'utilizzo di cinque e di sette variabili di input. Nel primo, sono state considerate le cinque seguenti variabili:

- carico elettrico all'istante $t-1$;
- carico elettrico all'istante $t-2$;
- irradianza solare;
- temperatura esterna;
- day-code.

Nel secondo, oltre le cinque appena presentate, sono state utilizzate le ulteriori due variabili di input di seguito presentate:

- $\sin(2\pi \text{ hour}/24)$;
- $\cos(2\pi \text{ hour}/24)$.

La variabile *day-code*, utilizzata in entrambi i casi, indica se il giorno cui appartiene l'ora di cui prevedere il consumo è un giorno festivo o feriale.

Nel caso di utilizzo di tutti i modelli di rete neurale *feed forward* è stata condotta, altresì, un'analisi parametrica variando il numero di neuroni dello strato *hidden* della rete tra 5 e 50, al fine di migliorare le performance della rete stessa. Il secondo strato della rete è stato caratterizzato da un solo neurone e le funzioni di trasferimento di ogni strato successivo al primo sono state rispettivamente la *tansig* e la *purelin*, la prima di tipo sigmoidale,

la seconda di tipo lineare. Infine, l'algoritmo di apprendimento scelto è stato il *traingd*, ossia il metodo di retro-propagazione dell'errore tramite gradiente decrescente.

I risultati delle analisi svolte e appena presentate sono riportati nel paragrafo 4.1 del presente report.

3.3. ANALISI DEI DATASET RELATIVI ALLE UTENZE UNICAL

Uno degli scopi dell'attività prevista dall'AR 5.3 del progetto ComESTo, come già detto, è stato quello di curare la messa a punto di modelli di previsione del carico di determinate utenze ubicate sul territorio italiano, su diverse scale temporali. Tali utenze, nel report indicate come "utenze UNICAL", fanno riferimento a utenze domestiche e non, di tipo *consumer* o *prosumer*, i cui consumi sono in corso di misurazione da parte dell'Università della Calabria (UNICAL), dal mese di gennaio 2019. Nel dettaglio, il database *MySQL* messo a disposizione da UNICAL raccoglie sei dataset di consumo, tre riferiti ad utenze di tipo domestico e tre ad utenze di vario tipo, non raggruppabili, e per questo indicate con la dicitura "non domestico". Ciascun utente considerato presenta uno o due canali misurati dallo stesso *smart meter* (SM), e le misure di potenza ed energia presentano una frequenza di campionamento di 5 secondi.

Al fine di procedere con l'attività di previsione, sono stati analizzati i dati relativi all'energia assorbita da ciascun utente. Per gli utenti *consumer* questa coincide con l'energia prelevata dalla rete elettrica nazionale ed è quindi un dato misurato; per gli utenti *prosumer* è stata calcolata come la differenza tra l'energia prelevata dalla rete, considerata per convenzione positiva, e l'energia generata dal sistema di autoproduzione, considerata per convenzione negativa.

Durante l'esplorazione dei dati sono state riscontrate alcune problematiche connesse principalmente a buchi di misura, ritardi nell'inserimento delle misure all'interno del database, inserimento di dati duplicati e in generale anomalie di misura. Pertanto, è stata richiesta in prima battuta, una fase di individuazione e correzione di tali errori. Per tutti gli *Smart Meter* è stato, quindi, utilizzato il seguente approccio di pre-elaborazione dei dati:

- i record, per i quali è stato riscontrato un ritardo di inserimento all'interno del database superiore alla soglia stimata con il 95-esimo quantile, solitamente inferiore a 10 s, sono stati cancellati;
- i dati duplicati sono stati eliminati;
- i *missing values* sono stati risolti usando come metodo di imputazione un template giornaliero, ottenuto mediante il calcolo della mediana per ogni ora. Per tener conto della diversità delle distribuzioni nei giorni feriali e nei giorni del fine settimana, sono stati usati due template differenziati. Sono stati valutati altri metodi di imputazione, ma, in presenza di giornate intere di dati mancanti, l'uso del template è sembrata la soluzione migliore.

Infine, i dati necessari alla previsione sono stati aggregati su base oraria, al fine di conformarli alla granularità della previsione obiettivo dell'AR in questione. Inoltre, sono state considerate mancanti quelle ore in cui il numero di misure acquisite è risultato essere inferiore al 30% delle misure attese, ovvero inferiore al 30% di 720. Per ciascun *smart meter* analizzato è stata effettuata un'attività di analisi dati di tipo esplorativo al fine di identificare gli approcci di forecasting più indicati. Sebbene cinque dei sei dataset siano in continuo aggiornamento, essendo tuttora in corso la campagna di misurazione, al fine di eseguire le prove necessarie alla previsione del carico, per ciascuna utenza è stata fissata una data ultima di lettura dei dati.

Nei paragrafi che seguono, è riportata la descrizione dei diversi dataset, indicati ognuno con il codice dello *smart meter* ad esso associato e, solo per lo SM 112, sono riportati alcuni grafici a scopo esemplificativo risultanti dall'analisi dati. Il paragrafo 3.3.1 riporta la descrizione dei dataset riferiti alle utenze di tipo

domestico, mentre il paragrafo 3.3.2 riporta la medesima descrizione per i dataset relativi alle utenze non domestiche.

3.3.1. Utenze di tipo domestico

Le utenze di tipo domestico sono indicate con le sigle SM 3, SM 112 e SM 232. Di seguito, sono riportate le principali caratteristiche di ognuna di esse. A titolo esemplificativo, in Figura 2 sono riportati i consumi di una settimana (25 marzo - 01 aprile) per tutti i casi residenziali analizzati. Da tale grafico è possibile notare per tutti gli SM una forte variabilità dei carichi, anche tra due ore successive.

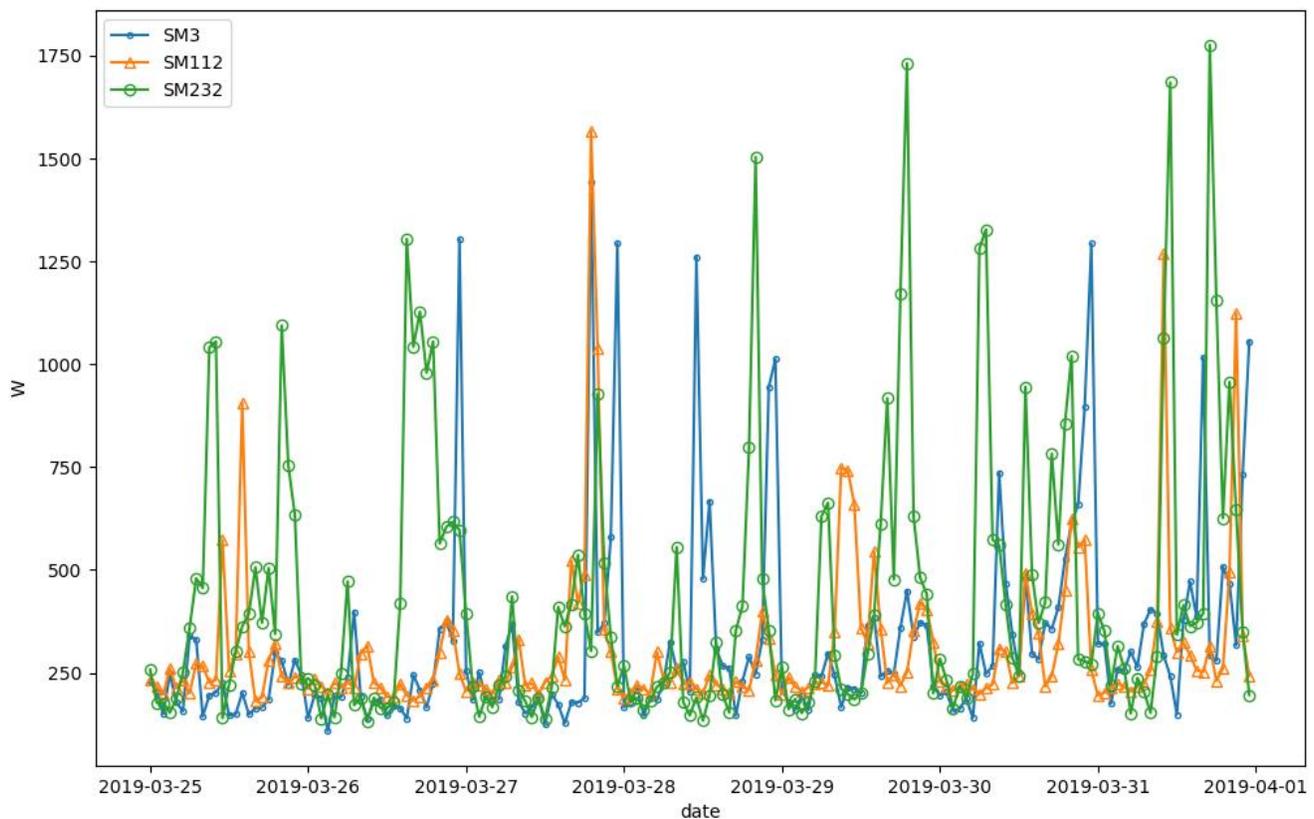


Figura 2. Consumi medi orari per i tre casi residenziali relativi ad una settimana

3.3.1.1. SM 3

L'utente SM 3 è un utente domestico di tipo *consumer*, con potenza elettrica installata di 4.5 kW, localizzato nel comune di Falerna Marina in provincia di Cosenza. La campagna di misurazione per tale utente ha avuto inizio in data 01/01/2019 ed è tutt'ora in corso. Al fine di poter effettuare le prove, il periodo di misura che è stato considerato va dal 06/01/2019 al 06/07/2019. Andando a valutare la *Sample Entropy*¹ della serie temporale per ogni mese, si è osservato che i primi tre mesi dell'anno risultano avere un valore alto, situazione

¹ La *Sample Entropy* cerca di quantificare l'imprevedibilità delle fluttuazioni in una serie temporale. Tale misura è stata introdotta per la prima volta in [46].

che evidenzia una bassa predicibilità. Alla metà del mese di aprile si verifica un cambio pronunciato del pattern di consumo.

3.3.1.2. SM 112

L'utente SM 112 è un utente domestico di tipo *prosumer*, con potenza elettrica installata di 3 kW e potenza del sistema di generazione pari a 2.25 kW, localizzato nel comune di Belmonte Calabro, in provincia di Cosenza. La campagna di misurazione per tale utente ha avuto inizio in data 01/01/2019 ed è tutt'ora in corso. Al fine di poter effettuare le prove, il periodo di misura che è stato considerato va dal 06/01/2019 al 06/07/2019. Lo SM 112 presenta diversi giorni, anche consecutivi, con assenza di dati. Tale situazione, sebbene risolta mediante l'utilizzo del metodo di imputazione sopra descritto, inevitabilmente distorce la serie temporale.

Di seguito sono riportati alcuni plot esplorativi dei dati relativi allo SM in questione. Nel dettaglio, la Figura 3 riporta i valori medi orari e giornalieri della potenza elettrica consumata e i *missing data* sostituiti con i template weekday-weekend nell'intero periodo esaminato, mentre la Figura 4 e la Figura 5 riportano, rispettivamente, i consumi elettrici mensili per tutto il periodo di misurazione e gli istogrammi per ciascun mese del periodo osservato.

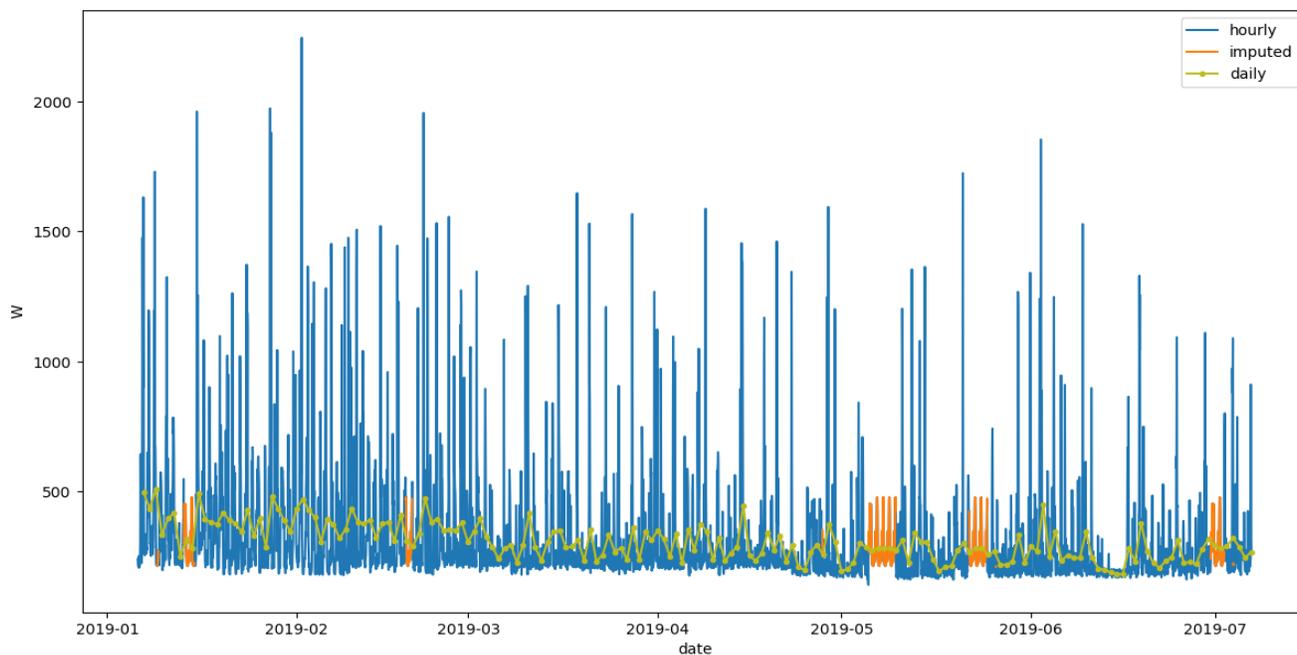


Figura 3. Valori medi orari (blu) e giornalieri (verde) della potenza elettrica consumata e i *missing data* sostituiti con i template weekday-weekend (arancione) nell'intero periodo esaminato per lo SM 112

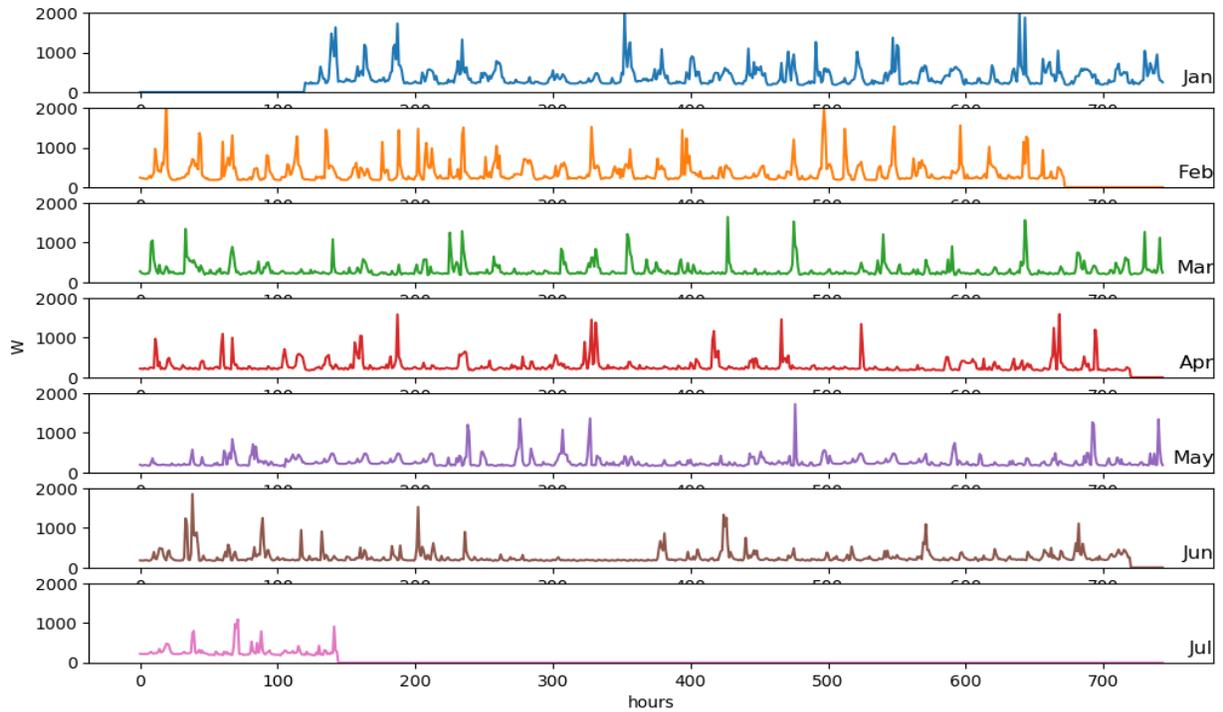


Figura 4. Andamento del consumo di potenza in ciascun mese nel periodo osservato per lo SM 112

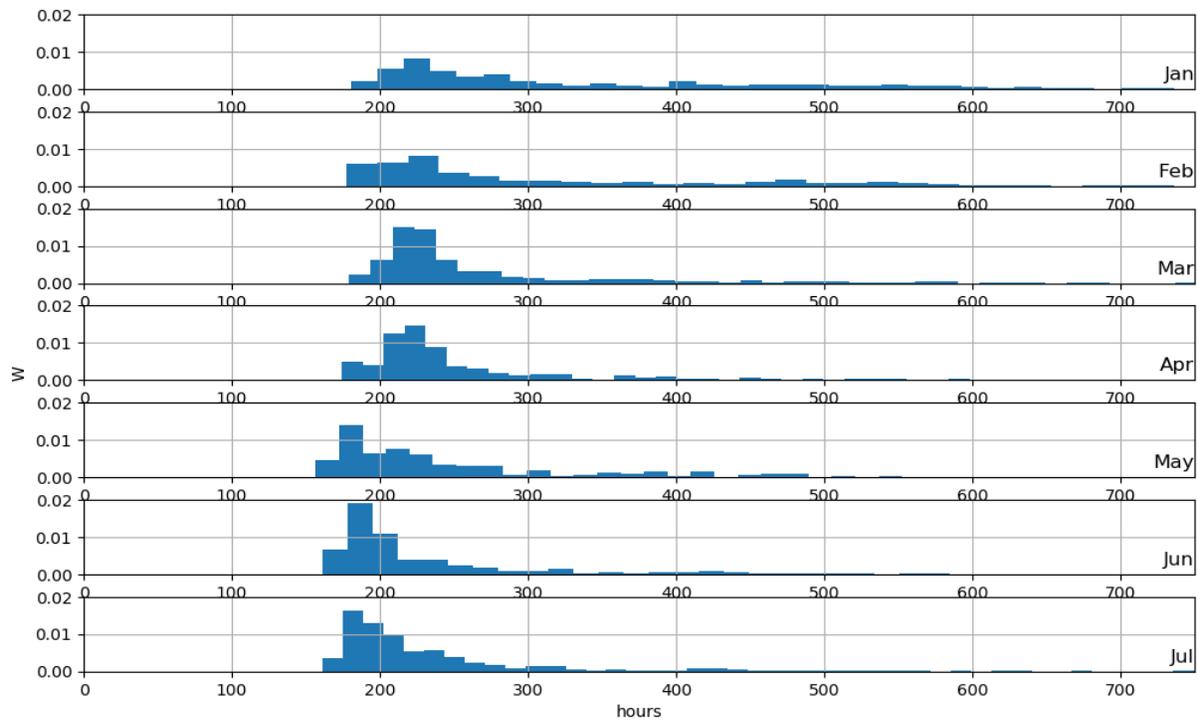


Figura 5. Istogrammi per ciascun mese nel periodo osservato per lo SM 112

La Figura 6 riporta gli andamenti del consumo per un'intera settimana, solo per i giorni feriali e solo per i weekend. Da tale figura, si nota una certa somiglianza tra il template settimanale e quello dei giorni feriali.

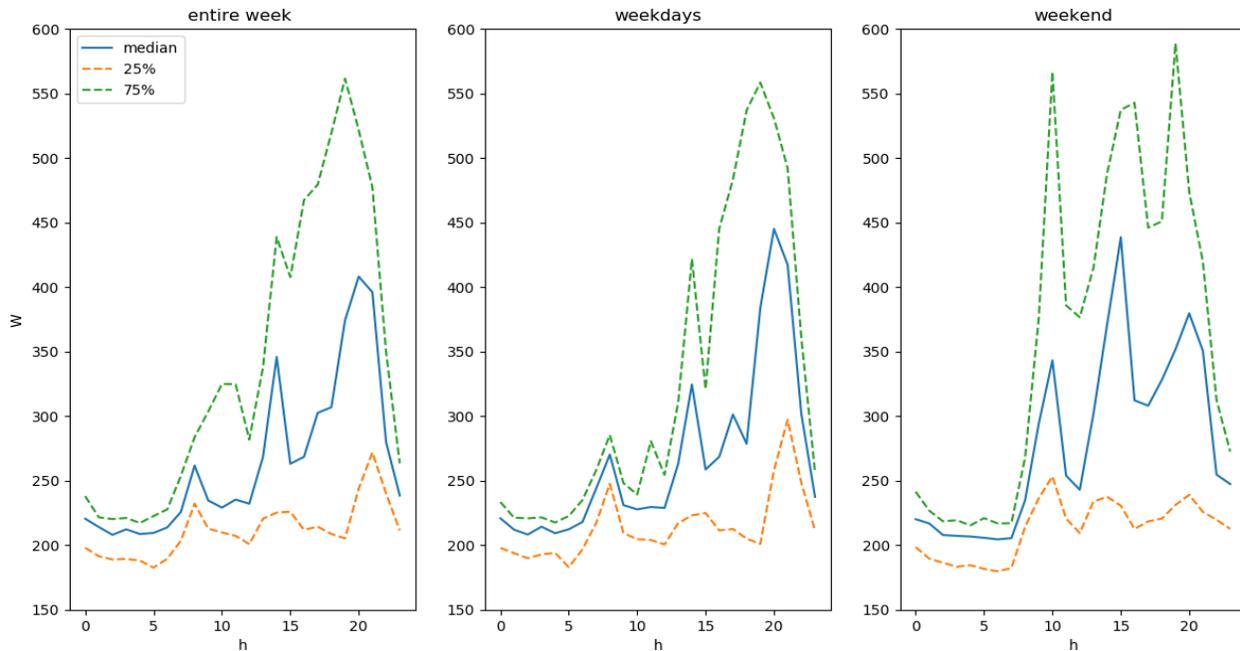


Figura 6. Template rispettivamente sull'intera settimana, solo sui giorni feriali, solo sui weekend per lo SM 112

La Figura 7 riporta l'istogramma della distribuzione dei consumi. Da tale figura, si nota la prevalenza di valori bassi, al di sotto di 500 W e la presenza di una lunga coda verso valori alti di potenza.

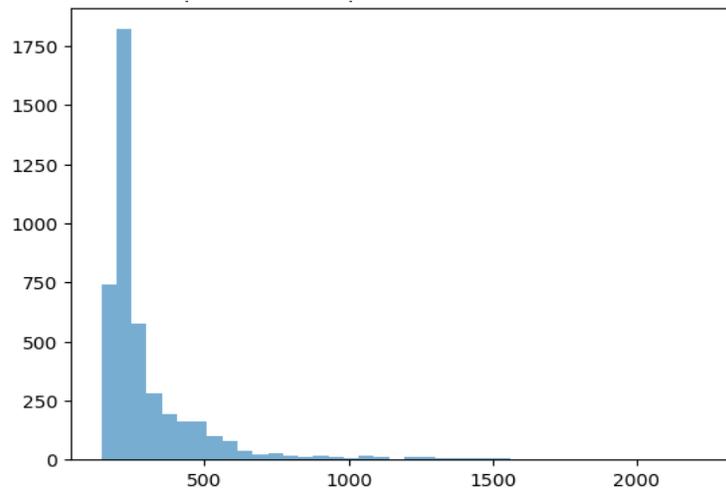


Figura 7. Iistogramma della distribuzione dei consumi per lo SM 112

La Figura 8 riporta box e whisker plot (diagramma a scatola e baffi) dei consumi di tutto il periodo di osservazione sulla settimana, dal lunedì (mon) alla domenica (sun). Tale figura descrive le caratteristiche

salienti della distribuzione in ciascun giorno: i bordi della scatola rappresentano il 25mo e 75mo percentile dei dati, mentre la linea che la divide in due parti rappresenta il 50mo, ovvero la mediana. I baffi che si estendono dalla scatola hanno lunghezza pari a 1.5 volte il range interquartile, ovvero l'altezza della scatola, mentre i punti cerchiati che li superano sono outlier. In questo caso si nota una larga presenza di outlier e una mediana simile in tutti i giorni. La Figura 9, invece, riporta la distribuzione dei consumi di tutto il periodo di osservazione sulle 24h. In questo caso, i boxplot mostrano, infatti, l'andamento dei consumi nelle diverse ore della giornata, evidenziando una maggiore variabilità nelle ore pomeridiane e serali.

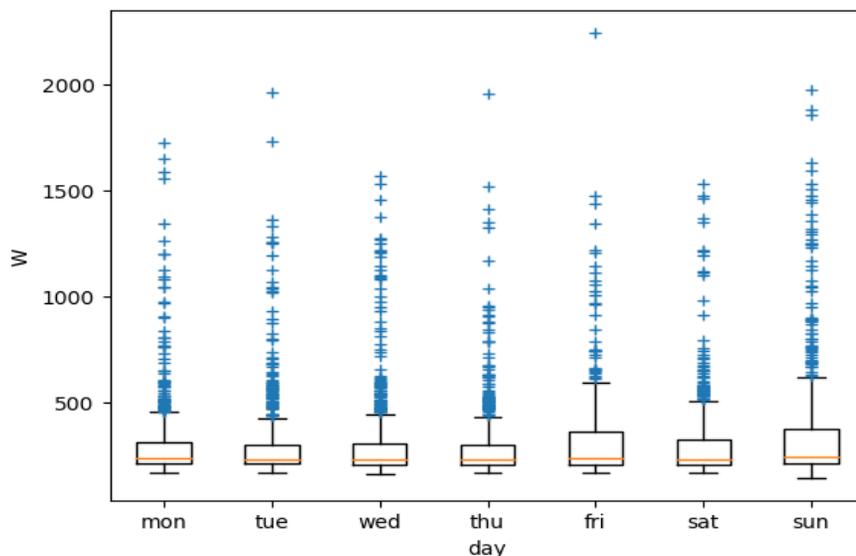


Figura 8. Box e whisker plot (diagramma a scatola e baffi) dei consumi della settimana, dal lunedì (mon) alla domenica (sun) per lo SM 112

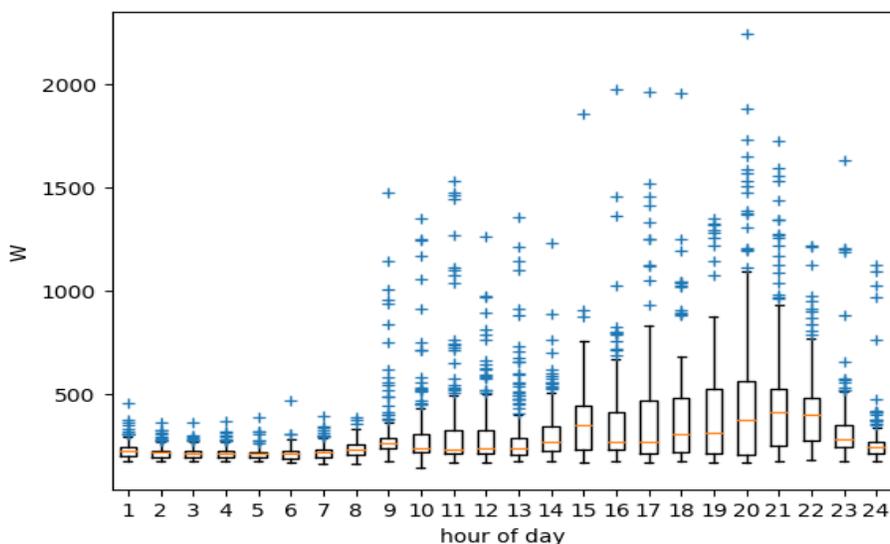


Figura 9. Distribuzione dei consumi su ciascuna ora della giornata per lo SM 112

3.3.1.3. SM 232

L'utente SM 232 è un utente domestico di tipo *prosumer*, con potenza elettrica installata di 4.5 kW e potenza del sistema di generazione pari a 6 kW, localizzato nella città di Reggio Calabria. La campagna di misurazione

per tale utente ha avuto inizio in data 07/01/2019 ed è tutt'ora in corso. Al fine di poter effettuare le prove, il periodo di misura che è stato considerato va dal 20/01/2019 al 06/07/2019. Lo SM 232 presenta anch'esso diversi giorni, anche consecutivi, con assenza di dati, soprattutto nel mese di gennaio. Anche in questo caso, sebbene sia stato utilizzato il metodo di imputazione di cui sopra, tale situazione inevitabilmente distorce la serie temporale.

3.3.2. Utenze di tipo non domestico

Le utenze di tipo non domestico sono indicate con le sigle SM 253, SM 103 e SM 262 e fanno riferimento, rispettivamente, ad una falegnameria, ad un piccolo hotel e ad una software house. Di seguito, sono riportate le principali caratteristiche di ognuna di esse. A titolo esemplificativo, in Figura 10 sono riportati i consumi di una settimana (25 marzo - 01 aprile) per tutti i casi non domestici analizzati. Da questa si nota come gli SM 103 e 262 presentino andamenti dei consumi meno variabili rispetto allo SM 253 e rispetto a quelli domestici, presentati in Figura 2.

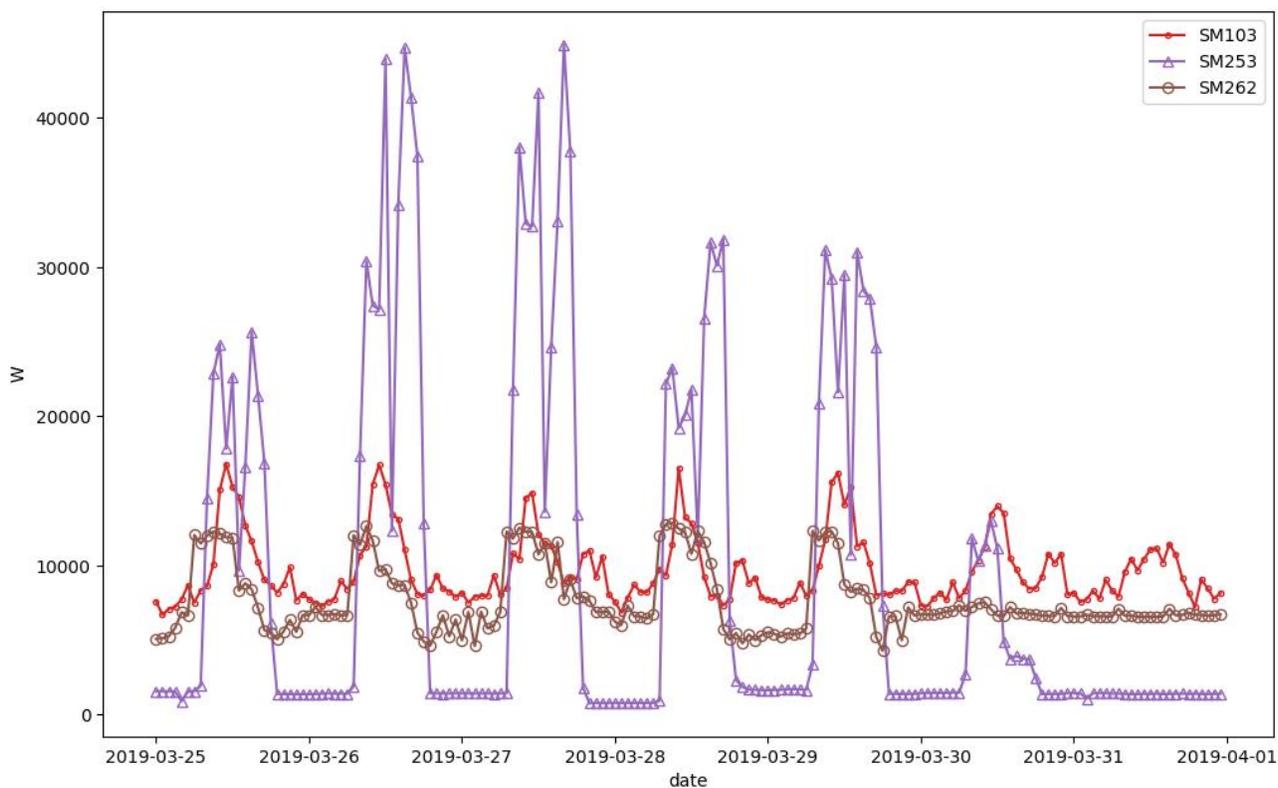


Figura 10. Consumi medi orari per i tre casi non residenziali relativi ad una settimana

3.3.2.1. SM 253 – Falegnameria

L'utente SM 253 è una falegnameria di tipo *consumer*, con potenza elettrica installata di 80 kW, localizzata nel comune di San Giorgio del Sannio in provincia di Benevento. La campagna di misurazione per tale utente ha avuto inizio in data 06/03/2019 ed è tutt'ora in corso. Al fine di poter effettuare le prove, il periodo di misura che è stato considerato va dal 24/03/2019 al 06/07/2019. Lo SM 253 presenta molti giorni consecutivi con assenza di dati (quasi 2 settimane). Anche in questo caso si è intervenuti con l'utilizzo del metodo di

imputazione sopra descritto ma, come per gli altri due casi, si assiste comunque ad una distorsione della serie temporale.

3.3.2.2. SM 103 – Piccolo hotel

L'utente SM 103 è un piccolo hotel di tipo *prosumer*, con potenza elettrica installata di 30 kW e potenza del sistema di generazione pari a 20 kW, localizzato nel comune di Falerna Marina in provincia di Cosenza. La campagna di misurazione per tale utente ha avuto inizio in data 01/01/2019. Per tale SM, sebbene la campagna di misurazione sia tuttora in corso, si riscontra un buco di registrazione di circa un mese, dal 09/06/2019 al 09/07/2019; pertanto, al fine di poter effettuare le prove, il periodo di misura considerato va dal 06/01/2019 al 01/06/2019.

3.3.2.3. SM 262 – Software house

L'utente SM 262 è una software house di tipo *consumer*, con potenza elettrica installata di 60 kW, localizzata nella città di Catanzaro. La campagna di misurazione per tale utente ha avuto inizio in data 01/02/2019, ed è stata bloccata al 30/04/2019. Pertanto, al fine di poter effettuare le prove, il periodo di misura che è stato considerato va dal 03/02/2019 al 27/04/2019.

3.4. MODELLI E METODI APPLICATI

Il forecasting del carico elettrico relativo alle utenze UNICAL, è stato eseguito mediante lo sviluppo di 11 modelli di diverso tipo, basati su 7 metodi statistici classici o di *machine learning*, al fine di poter effettuare un confronto tra i modelli stessi e individuare quello che maggiormente si presta e si adatta al forecasting dello specifico caso analizzato. Nel dettaglio, gli approcci utilizzati sono stati i seguenti:

- Persistenza;
- Regressione lineare;
- FFNN – Feed Forward Neural Network;
- CNN - Convolutional Neural Network;
- LSTM - Long Short-Term Memory network;
- XGBoost - Extreme Gradient Boosting;
- SVR - Support Vector Regression.

La scelta degli approcci da testare è scaturita da un'attenta analisi di letteratura, che ha permesso di selezionare tra i vari, gli approcci più indicati a questo tipo di previsione. In particolare, i primi due approcci descritti, ovvero quello della persistenza e della regressione lineare, sono stati utilizzati come approcci di riferimento con i quali andare a confrontare approcci più complessi.

Nei paragrafi che seguono sono descritti brevemente gli approcci utilizzati. Nel dettaglio, il paragrafo 3.4.1 riporta il metodo della persistenza; il 3.4.2 della regressione lineare; il 3.4.3 è relativo alle reti neurali di tipo *feed forward*; il 3.4.4 alle reti neurali convoluzionali; il 3.4.5 riporta la *Long Short Term Memory network*; il 3.4.6 il metodo *eXtreme Gradient Boosting*; e infine, il paragrafo 3.4.7 riporta il *Support Vector Regression*.

3.4.1. Persistenza

Un modello di persistenza si basa sulla ripetizione dell'ultimo valore osservato per prevedere quello successivo, in accordo alla relazione seguente:

$$\hat{y}(t) = y(t - 1) \quad (1)$$

dove $y(t - 1)$ rappresenta il valore misurato del carico elettrico nell'ora precedente.

3.4.2. Regressione lineare

In un modello di *regressione lineare semplice* si assume che la variabile di output y sia funzione lineare della variabile di input x , secondo la seguente relazione:

$$y = b_0 + b_1x \quad (2)$$

Nel caso di n variabili di input x_1, \dots, x_n , si ha un modello di *regressione lineare multipla* in cui la relazione che sussiste tra output e input è la seguente:

$$y = b_0 + b_1x_1 + \dots + b_nx_n \quad (3)$$

La relazione tra variabili di input, dette predittori, e quelle di output, dette target, è modellata andando a stimare dai dati i coefficienti b_0, b_1, \dots, b_n della funzione, cercando quei valori che minimizzano l'errore nella predizione della variabile di output. A tale scopo, è usualmente utilizzato il metodo dei minimi quadrati, nel quale la funzione costo da minimizzare è data dalla somma del quadrato dei residui tra le risposte osservate nel dataset (y) e quelle predette dall'approssimazione lineare (\hat{y}).

La regressione lineare può essere usata anche sulle serie temporali, univariate o multivariate, ed è ideale per catturare il loro trend.

In questo caso, lo stimatore lineare userà le osservazioni della stessa variabile ai passi precedenti (lag) per predire il suo valore al passo successivo. Ad esempio, un modello lineare che predice il valore della variabile di interesse al tempo t utilizzando 2 lag di input, assume la forma:

$$\hat{y}(t) = b_0 + b_1y(t - 1) + b_2y(t - 2) \quad (4)$$

Nel caso in cui la serie venga generata da un processo autoregressivo, allora questo modello sarebbe anche il predittore ottimo.

Per una serie temporale multivariata, in cui si trovano più variabili di input in relazione tra loro, la predizione della variabile di interesse al tempo t usando ad esempio due variabili y_1, y_2 e due lag, ha la forma:

$$\hat{y}(t) = b_0 + b_1y_1(t - 1) + b_2y_1(t - 2) + b_3y_2(t - 1) + b_4y_2(t - 2). \quad (5)$$

3.4.3. FFNN

Uno degli approcci più utilizzati per la previsione dei carichi elettrici è sicuramente quello che implementa un modello di calcolo con le reti neurali di tipo *Feed Forward* [34] [36] [37]. Queste sono efficienti e potenti modelli computazionali, ispirati all'apparato nervoso umano, ossia all'insieme dei neuroni che fanno parte del cervello umano. Infatti, le FFNN, hanno la capacità di prevedere il comportamento non casuale e non lineare della fenomenologia oggetto di studio, mediante un opportuno sistema di apprendimento. In pratica, la rete neurale è in grado di apprendere la connessione o relazione che lega i valori di input di una funzione ai rispettivi valori di output, attraverso lo studio di noti valori di input/output.

In generale le reti neurali *Feed Forward* consistono di tre parti, come rappresentato in Figura 11. Schema di una generica rete di tipo *feed forward con un solo livello intermedio*:

- un livello di ingresso o *input layer* con un numero di neuroni pari al numero di input della rete;
- uno o più livelli intermedi o *hidden layers* con un numero di neuroni da assegnare;

- un livello di uscita costituito da tanti neuroni quanti sono gli output desiderati.

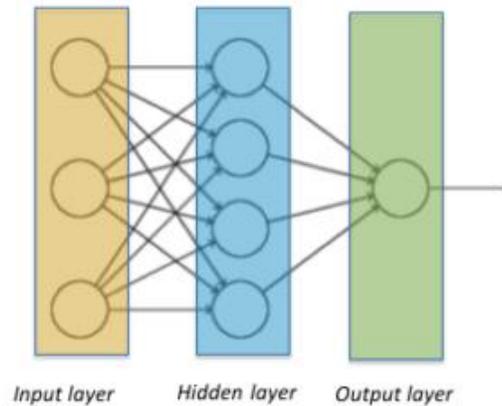


Figura 11. Schema di una generica rete di tipo *feed forward* con un solo livello intermedio

Osservando lo schema di Figura 11. Schema di una generica rete di tipo *feed forward* con un solo livello intermedio, si può notare che in tale tipologia di rete ogni neurone è connesso con tutti i neuroni dello strato precedente, ma non ha connessioni con i neuroni del suo stesso strato e il segnale si trasmette in un'unica direzione dal livello di ingresso a quello di uscita tramite gli strati nascosti.

Una FFNN con un solo strato nascosto, ha la seguente formulazione matematica:

$$y = g \left[\beta_0 + \sum_{m=1}^M \beta_m \cdot \phi \left(\alpha_{0m} + \sum_{i=1}^D \alpha_{im} x_i \right) \right] + \varepsilon \quad (6)$$

dove y è il valore di output della rete; M e D sono, rispettivamente, il numero di neuroni dello strato *hidden* e sullo strato di input, x_i è l' i -esimo input; α_{im} è il peso della connessione tra i -esima variabile di input e l' m -esima unità nascosta; α_{0m} è il bias dell' m -esima unità nascosta; ϕ è la funzione di trasferimento che lega l'input alle unità nascoste; β_m è il peso del collegamento tra l' m -esima unità nascosta e il vettore y di output; β_0 è il bias dell'output; g è la funzione di trasferimento che lega le unità nascoste all'output e ε è una variabile casuale.

Il cuore di una FFNN è rappresentato, quindi, dall'algoritmo di apprendimento o di addestramento, che consente di aggiornare i pesi per cercare di minimizzare la differenza tra il valore generato dalla rete e quello reale. Di solito la funzione che viene utilizzata per l'aggiornamento dei pesi si basa su metodi di discesa del gradiente (*gradient descent*). Nel nostro lavoro abbiamo utilizzato una rete di questo tipo, ossia con un solo strato nascosto.

3.4.4. CNN

Le reti neurali convolutive (*Convolutional Neural Network* – CNN [37]) sono reti neurali specializzate per il processamento di dati che hanno una topologia a griglia. Oltre che le immagini, per cui originariamente sono state pensate, anche le Serie Temporal possono essere viste come un insieme di campioni disposti su griglia monodimensionale rappresentante gli istanti di tempo. Mentre nelle reti neurali tradizionali vi sono tanti parametri distinti quante sono le interazioni tra unità di input ed unità di output di ogni livello, nelle CNN vi è una condivisione dei parametri, così che è necessario salvare solo un numero minimo di parametri (*Kernel* o *Filtri*). Per ogni livello, lo scorrimento del *Kernel* su tutto il volume di ingresso e il conseguente calcolo della convoluzione (a rigore si tratta di una correlazione), porta alla creazione della cosiddetta *Activation Map*. Il valore del pixel (i,j) della *Activation Map* è tanto più alto quanto maggiore sarà la correlazione tra il filtro e la parte di volume considerato in quel momento (*Receptive Field*). Il particolare tipo di condivisione dei parametri

introdotto nelle CNN, fornisce a queste ultime l'*equivarianza alla traslazione*². In Figura 12 è riportato un esempio di livello convolutivo.

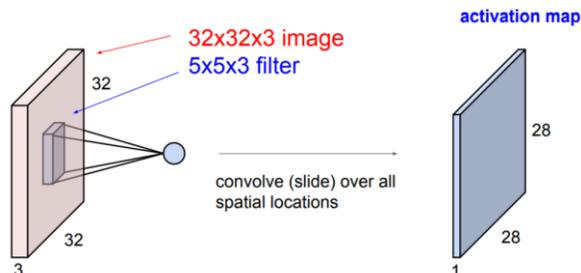


Figura 12. Esempio di Livello Convolutivo [38]

Di solito, all'interno delle reti convolutive, vengono inseriti anche dei livelli di *Pooling* che sostituiscono l'output di un certo livello con una statistica calcolata su un insieme di output vicini. Questa operazione effettua un *downsampling* dell'immagine riducendo quindi il carico computazionale richiesto e aumentando le caratteristiche di invarianza alla traslazione³. In Figura 13 è riportato un esempio di *Pooling*.

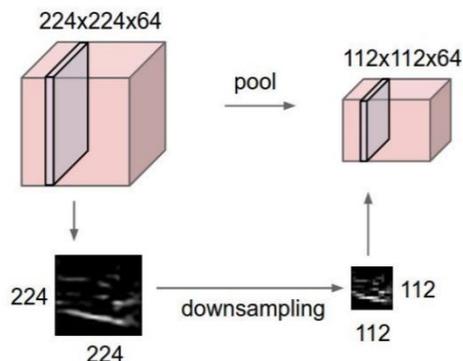


Figura 13. Esempio di Pooling [38]

3.4.5. LSTM

Le reti neurali ricorrenti (Recurrent Neural Network – RNN [39]) sono una famiglia di reti neurali utilizzate per il processamento di dati sequenziali. A differenza delle reti Feed Forward, presentano un loop interno che permette alle informazioni di persistere. La caratteristica principale che le rende adatte al modellamento delle sequenze, è la condivisione dei parametri tra step diversi. In Figura 14 è riportato uno schema di massima di una rete neurale ricorrente.

² Nel caso monodimensionale delle Time Series, se un evento viene traslato nel tempo, la sua rappresentazione, prodotta dalla CNN, verrà traslata nel tempo nello stesso modo.

³ Piccole traslazioni dell'input producono lo stesso output. Questa caratteristica è utile quando è più importante sapere se è presente una feature piuttosto che saperne la posizione esatta.

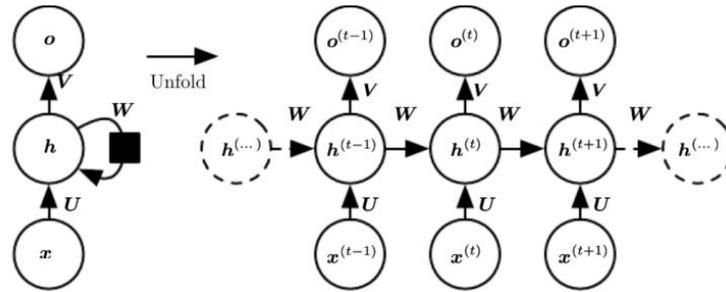


Figura 14. Schema di massima di una RNN nella sua forma folded (sinistra) e unfolded (destra) [40]

Ad ogni step è associato una cella che, dati l'ingresso ($x^{(t)}$) e lo stato allo step precedente ($h^{(t-1)}$), calcola lo stato ($h^{(t)}$) e l'uscita ($o^{(t)}$) secondo le seguenti formule.

$$h^{(t)} = \sigma_h (U \cdot x^{(t)} + W \cdot h^{(t-1)} + b_h) \quad (7)$$

$$o^{(t)} = \sigma_o (V \cdot h^{(t)} + b_o) \quad (8)$$

dove: σ_h e σ_o sono le funzioni di attivazione (di solito la tangente iperbolica) rispettivamente dello stato e dell'output; U è la matrice dei pesi che rappresenta la connessione input-stato; W è la matrice dei pesi che rappresenta la connessione ricorrente stato-stato; V è la matrice dei pesi che rappresenta la connessione stato-output; b_h e b_o sono i vettori di bias, rispettivamente dello stato e dell'output.

La LSTM [41] è una particolare rete ricorrente dove, per eliminare il problema del *vanishing* e *exploding gradient* (durante la fase di apprendimento), su sequenze lunghe, viene inserito un elemento di memoria ($c^{(t)}$). La Figura 15 descrive lo schema di una generica cella LSTM.

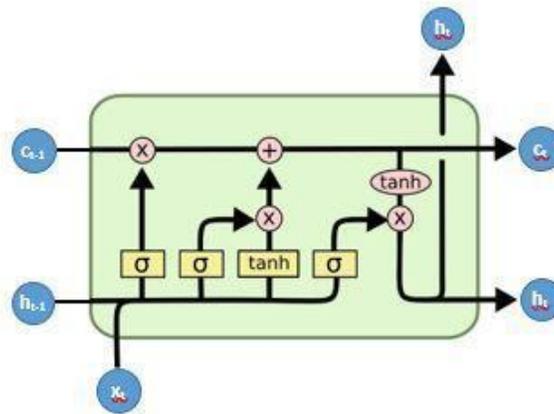


Figura 15. LSTM Cell [42]

La caratteristica principale della LSTM è il suo stato interno formato da due parti distinte. Una prima parte rappresenta una sorta di memoria a lungo termine, lo stato della cella ($c^{(t)}$), a cui vengono aggiunte o rimosse informazioni attraverso le tre gate: Forget Gate, Input Gate, Output Gate. La seconda parte è invece lo Stato Nascosto ($h^{(t)}$) che rappresenta una sorta di memoria a breve termine. Le operazioni effettuate dalle tre gate sono:

Forget Gate - decide quali informazioni vanno rimosse dallo Stato della Cella.

$$f^{(t)} = \sigma (W_f x^{(t)} + U_f h^{(t-1)} + b_f) \quad (9)$$

Input Gate - si compone di due parti, la prima, eq. (10), crea i valori candidati con cui aggiornare lo stato della cella; la seconda, eq. (11), decide quali valori, tra i candidati individuati dalla prima parte, dovranno essere aggiunti allo stato della cella.

$$\tilde{c}^{(t)} = \tanh (W_c x^{(t)} + U_c h^{(t-1)} + b_c) \quad (10)$$

$$i^{(t)} = \sigma (W_i x^{(t)} + U_i h^{(t-1)} + b_i) \quad (11)$$

Output Gate - decide il sottoinsieme di informazioni, presente nello stato della cella, da propagare agli stati successivi. Non tutto il contenuto della memoria deve infatti essere salvato nello stato nascosto.

$$o^{(t)} = \sigma (W_o x^{(t)} + U_o h^{(t-1)} + b_o) \quad (12)$$

I valori aggiornati di c_t e h_t sono ottenuti combinando opportunamente i valori ottenuti dalle singole gate:

$$c^{(t)} = f^{(t)} \otimes c^{(t-1)} + i^{(t)} \otimes \tilde{c}^{(t)} \quad (13)$$

$$h^{(t)} = o^{(t)} \otimes \tanh(c^{(t)}) \quad (14)$$

dove: $W_f, W_i, W_o, W_c, U_f, U_i, U_o, U_c, b_f, b_i, b_o, b_c$ sono le matrici e i bias utilizzati per calcolare il contributo dell'input e dello stato precedente al valore della gate corrispondente; σ è la funzione di attivazione sigmoideale $\left(\frac{1}{1+e^x}\right)$ che assume valori tra 0 e 1; \otimes è il prodotto di Hadamard (elemento per elemento).

3.4.6. XGBoost

Il metodo Extreme Gradient Boosting (XGBoost) [43] utilizza un insieme (ensemble) di predittori semplici (in questo caso Decision Tree) i cui risultati vengono combinati attraverso una tecnica di Boosting. Nel Boosting i diversi predittori vengono allenati in sequenza su un training set che tiene conto degli errori commessi dai precedenti predittori. Se un esempio viene predetto male questo avrà un maggior peso nell'apprendimento del predittore successivo. Una volta che tutti i predittori sono allenati, le loro predizioni vengono opportunamente combinate con un sistema di Voto [44].

L'XGBoost implementa i concetti sopra descritti utilizzando un modello opportunamente regolarizzato per evitare l'overfitting e migliorare i risultati.

3.4.7. SVR

La Support Vector Regression (SVR) è l'estensione della Support Vector Machine (SVM) [45] [46] ai problemi di regressione. Come funzione di Loss da minimizzare, invece di usare l'errore quadratico, si utilizza l'*Epsilon-Sensitive Loss* definita come:

$$e_\epsilon(y^{(t)}, \hat{y}^{(t)}) = \begin{cases} 0, & \text{se } |y^{(t)} - \hat{y}^{(t)}| < \epsilon \\ |y^{(t)} - \hat{y}^{(t)}| - \epsilon, & \text{altrove} \end{cases} \quad (15)$$

questo vuol dire che vengono tollerati errori fino ad ϵ , andando, in questo modo, a definire una *Zona Epsilon-Insensitive* all'interno della quale l'errore è nullo. Vengono poi introdotte delle variabili slack (ξ_+^t, ξ_-^t) per tener conto della deviazione intorno alla *Zona Epsilon-Insensitive*.

Il problema viene quindi visto come un problema di minimizzazione nella forma:

$$\begin{aligned} & \min \left\{ \frac{1}{2} \|w\|^2 + C \sum_t \xi_+^t + \xi_-^t \right\} \\ \text{s.t.} & \\ & y^{(t)} - (w^T x + w_0) \leq \epsilon + \xi_+^t \\ & (w^T x + w_0) - y^{(t)} \leq \epsilon + \xi_-^t \\ & \xi_+^t, \xi_-^t \geq 0 \end{aligned} \quad (16)$$

Andando a considerare il problema duale si ottiene:

$$\begin{aligned} & \max \left\{ -\frac{1}{2} \sum_t \sum_s (\alpha_+^{(t)} - \alpha_-^{(t)}) (\alpha_+^{(s)} - \alpha_-^{(s)}) (\mathbf{x}^{(t)})^T (\mathbf{x}^{(s)}) - \epsilon \sum_t (\alpha_+^{(t)} + \alpha_-^{(t)}) + \sum_t r^{(t)} (\alpha_+^{(t)} - \alpha_-^{(t)}) \right\} \\ \text{s.t.} & \\ & 0 \leq \alpha_+^{(t)} \leq C, \\ & 0 \leq \alpha_-^{(t)} \leq C, \\ & \sum_t (\alpha_+^{(t)} - \alpha_-^{(t)}) = 0 \end{aligned} \quad (17)$$

Dove $\alpha_+^{(t)}$ e $\alpha_-^{(t)}$ sono i moltiplicatori di Lagrange. In Figura 16 si può vedere la *Zona Epsilon-Insensitive* (quella racchiusa dalle due linee tratteggiate). Il punto (a) è interno alla zona e per esso vale: $\alpha_+^{(t)} = 0, \alpha_-^{(t)} = 0$. Il punto (b) è sul bordo e per esso vale $\alpha_+^{(t)} < C$. Il punto (c) è fuori dalla zona epsilon sensitive con un valore slack positivo ($\xi_+^t > 0$) e per esso vale $\alpha_+^{(t)} = C$.

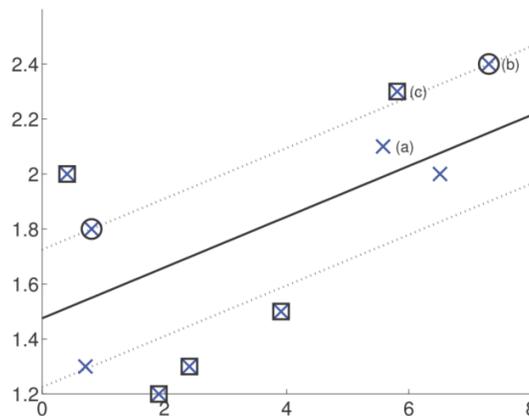


Figura 16. Esempio di regressione lineare con indicazione della Zona Epsilon-Insensitive [47]

Per realizzare un fitting non lineare degli ingressi è sufficiente utilizzare il *Kernel Trick* [44], sostituendo $(x^{(t)})^T(x^{(s)})$ con un'opportuna funzione Kernel, $k(x^{(t)}, x^{(s)})$ (ad es. Polinomiale, RBF ecc.).

3.5. MODELLI SVILUPPATI PER LA PREVISIONE DEL CARICO A 1 H

Nel seguito sono riportati i modelli sviluppati ai fini della previsione all'ora dopo del carico delle utenze UNICAL sopra descritte. Gli approcci di seguito presentati sono stati utilizzati in modo analogo per ciascun caso, al fine di poter confrontare i risultati ottenuti. In gran parte dei modelli sviluppati sono state eseguite differenti analisi, al fine di determinare le migliori configurazioni di opportuni parametri. Tale attività, detta *Hyperparameter Tuning*, è stata effettuata valutando i risultati in termini di minor Coefficiente di Variazione (CV) sui dati non imputati del *test set*, ovvero su quei dati del *test set* che non sono stati soggetti a modifiche secondo il template riportato precedentemente e, cioè, considerando solo i dati effettivamente misurati.

3.5.1. Persistenza

Il modello di persistenza è stato implementato mediante linguaggio Python. Nel dettaglio, è stato testato un modello che fa persistere il valore della potenza nell'ora precedente per la previsione di quella corrente, $\hat{y}(t) = y(t - 1)$ eq. 1. Inoltre, è stato valutato il potere predittivo della media dell'ultimo quarto d'ora sull'ora corrente, $\hat{y}(t) = y(t - 1/4)$.

3.5.2. Regressione lineare

Tutti i modelli sviluppati basati sulla regressione lineare sono stati implementati mediante linguaggio Python e libreria Scikit-learn. Nel dettaglio, questi sono stati:

- Regressione Lineare su serie univariata;
- Regressione Lineare su serie multivariata;
- Regressione Lineare a tratti;
- Regressione Lineare ibrida.

Nel caso di Regressione Lineare su serie univariata è stato utilizzato il modello descritto dall'eq. (4), con 2 lag. In questo caso, è stato testato, anche, l'effetto di detrending che si ottiene sottraendo il template giornaliero dai dati originali, con l'obiettivo di ricavare una serie più stazionaria⁴, da cui dipende la bontà della predizione del modello lineare. Questa operazione non ha prodotto variazioni significative del CV nei casi domestici e altrettanto poco utile si è rivelato lavorare sulle differenze di primo ordine dei residui, una volta sottratto il template.

Nel caso di Regressione Lineare su serie multivariata il modello è stato sviluppato usando 2 lag e i seguenti input, riferiti a grandezze relative al carico orario:

- media;
- scarto interquartile;
- mediana;
- deviazione standard;
- indici di asimmetria;
- curtosi.

⁴ Una serie temporale è stazionaria se la sua media e varianza sono costanti nel tempo, e l'auto co-varianza non dipende dal tempo.

Il modello di regressione lineare a tratti è stato utilizzato nel caso degli SM 3, 112, 232 e 262. Per ciascun caso i dati di input sono stati suddivisi in due gruppi, definiti in base a una soglia di carico medio orario. La scelta di testare tale modello è scaturita dal fatto che gli SM indicati presentano due grandi gruppi di consumo di energia. Pertanto, in questi particolari casi, sono stati specializzati due modelli, ciascuno su un gruppo di dati, al di sotto e al di sopra del valore di soglia, in accordo alla seguente equazione:

$$\hat{y}(t) = \begin{cases} b_0 + b_1y(t-1) + b_2y(t-2), & \text{se } y(t-1) < s \\ b'_0 + b'_1y(t-1) + b'_2y(t-2), & \text{se } y(t-1) \geq s \end{cases} \quad (18)$$

dove $s \in \{y(t)\}$ è la soglia definita ed $y(t) \geq 0$.

Di seguito, in Tabella 1 sono indicati i valori di soglia definiti per ciascuno degli SM per i quali è stato utilizzato tale modello.

Tabella 1. Valori di soglia per applicazione del modello lineare a tratti

SM	Valore di soglia (W)
3	400
112	500
232	700
262	13000

Infine, è stato implementato un modello ibrido, che consiste in un modello lineare per i giorni feriali e di uno basato sulla persistenza per i giorni del fine settimana. La scelta di testare tale modello deriva dall'osservazione di alcuni casi - come la falegnameria (SM 253) - che nei giorni del fine settimana presentano pattern molto diversi rispetto ai giorni feriali. Con riferimento allo SM 253, la Figura 17 mostra lo scatterplot dello storico al tempo t , $t-1$, $t-2$: nella Figura 17 (A) è riportata la vista 3D completa, nella Figura 17 (B) uno zoom 2D. La prima mostra la diffusione dei dati storici dei primi due lag rispetto al passo successivo, evidenziando la scarsa correlazione, visibile in particolare dalla formazione delle righe orizzontale e verticale; mentre dalla seconda si osserva una buona correlazione tra le serie t e $t-1$ dei valori bassi di potenza nel weekend, motivando l'uso della persistenza per questi giorni.

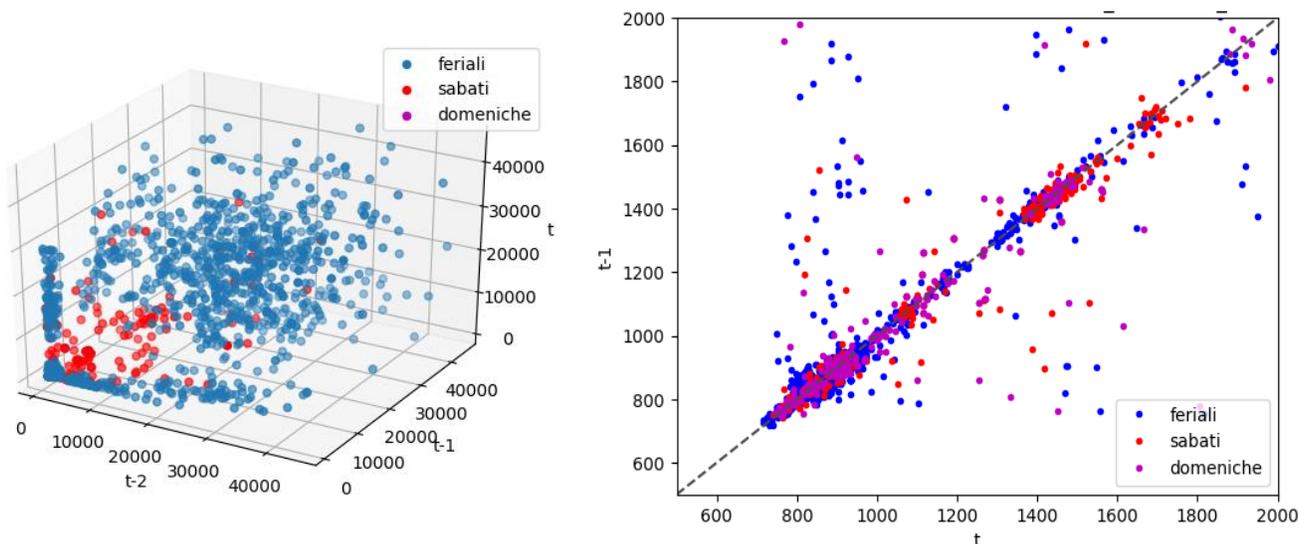


Figura 17. Scatterplot dello storico al tempo t , $t-1$, $t-2$ per lo SM 253: (A) vista 3D completa; (B) zoom in 2D

3.5.3. FFNN

Il modello di rete neurale *feed forward* è stato implementato in ambiente Matlab. Nel dettaglio, l'apprendimento della rete è stato basato sull'algoritmo di *Lavenberg-Marquardt*, mentre le funzioni di attivazione e di trasferimento utilizzate nello strato *hidden* e in quello di output sono state, rispettivamente, la *tansig*, ossia la funzione tangente iperbolica, e la *poslin*, ossia una funzione di trasferimento lineare positiva. Dopo un'attenta analisi, il numero di epoche di apprendimento è stato settato a 30000 per tutti i casi analizzati. Una volta definita l'architettura della rete sono stati, quindi, esaminati due modelli di calcolo, corrispondenti a due set di possibili variabili di input. Il primo modello è stato caratterizzato da un set di due variabili di input, ovvero:

- il carico elettrico all'ora precedente, ovvero all'istante $t-1$;
- il carico elettrico due ore prima, ovvero all'istante $t-2$.

Successivamente, allo scopo di esaminare l'effetto dovuto all'introduzione di ulteriori parametri di input, come l'ora e lo specifico giorno della settimana sulla previsione del carico elettrico all'ora corrente, sono state introdotte ulteriori variabili. In particolare, in questo caso, le variabili di input prese in considerazione sono state le seguenti:

- carico elettrico all'ora precedente, ovvero all'istante $t-1$;
- carico elettrico due ore prima, ovvero all'istante $t-2$;
- seno dell'ora, implementato come $\sin(2\pi \text{ hour}/24)$;
- coseno dell'ora, implementato come $\cos(2\pi \text{ hour}/24)$;
- seno del giorno della settimana;
- coseno del giorno della settimana;
- un coefficiente che tiene conto se è un giorno festivo o un fine settimana, *day-code*.

La Figura 18 mostra la struttura della rete sviluppata per entrambi i casi analizzati.

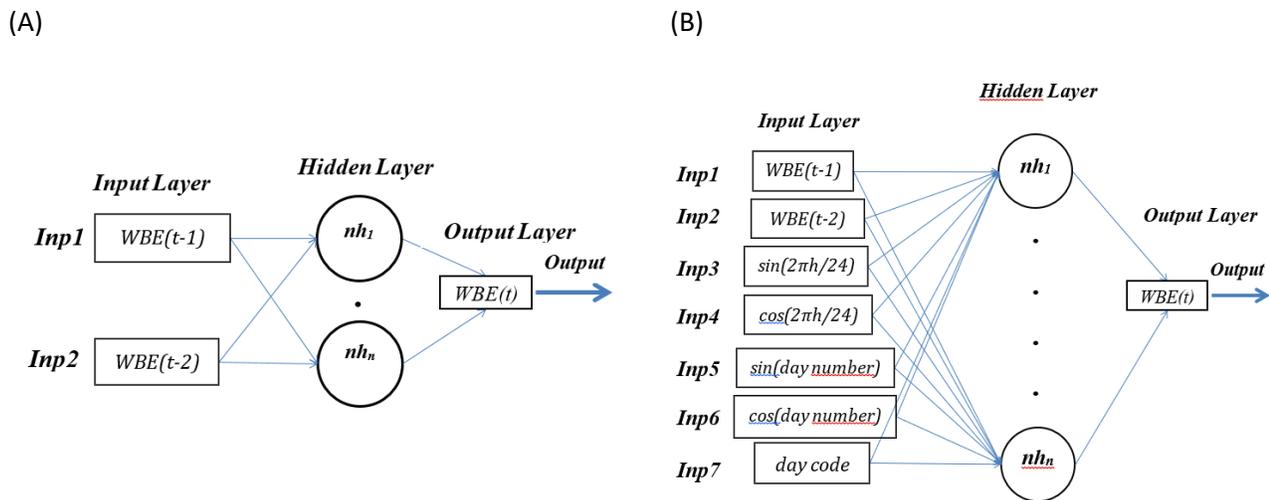


Figura 18. (A) Struttura del modello con due variabili di input; (B) Struttura del modello con 7 variabili di input

Solo per gli SM 232 e 262 è stata testata l’influenza dei dati meteo misurati. Nel dettaglio, per lo SM 232, al modello con 7 variabili di input sono stati aggiunte le seguenti tre variabili, relative a dati meteo misurati della specifica località:

- temperatura dell’ambiente esterno;
- umidità relativa dell’ambiente esterno;
- irradianza solare.

Per lo SM 262, al modello con 7 variabili di input è stata aggiunta la variabile relativa alla temperatura dell’ambiente esterno.

Per ciascun modello sviluppato, è stata condotta un’analisi parametrica variando il numero di neuroni dello strato *hidden* della rete tra 5 e 50, al fine di migliorare le performance della rete stessa. Nella Tabella 2 sono riportati, per ciascun *smart meter* e per ciascuno dei due modelli di rete neurale di tipo *FFNN*, il numero di neuroni ottimale risultante dall’analisi parametrica, ovvero quel determinato numero di neuroni dello strato *hidden* che ha condotto ad ottenere i più bassi valori del coefficiente di variazione. Solo per questi particolari casi, saranno riportati e discussi, nei paragrafi che seguono, i risultati ottenuti dai modelli di previsione del carico elettrico assorbito.

Tabella 2. Numero di neuroni ottimale per ciascun SM

SM	Numero di neuroni ottimale		
	FFNN - 2 input	FFNN - 7 input	FFNN – 7 input + dati meteo
3	35	10	-
112	44	21	-
232	9	8	9
103	14	12	-
253	24	31	-
262	12	35	44

3.5.4. CNN

Il modello di rete neurale di tipo convoluzionale è stato implementato mediante linguaggio Python utilizzando le seguenti librerie: Keras su Tensorflow backend, Scikit-learn, Numpy, pandas, Scipy. Nel dettaglio, la rete implementata è caratterizzata da: un Livello Convolutivo 1D con funzione di attivazione ReLU, un Livello di Batch Normalization e un livello Fully Connected con un unico neurone di uscita e funzione di attivazione ReLU. Al fine di individuare la configurazione migliore, in termini di minor CV ottenuto, sono stati testati i seguenti parametri:

- *hidden dimension* = [5, 25, 50, 100, 150, 200, 250, 300];
- numero di lag da considerare = [2, 3, 6, 12, 24, 48].

Per ogni configurazione, il numero di epoche di apprendimento è stato selezionato attraverso un approccio *Early Stopping* che consente di ridurre la probabilità di *overfitting* e nel contempo riduce l'onere computazionale.

Gli altri parametri sono stati settati in base a delle valutazioni a campione. Tra questi i più importanti sono:

- Kernel Size: 2;
- Batch Size: 32;
- Algoritmo di ottimizzazione: Adam (Adaptive Moment Estimation) con Learning Rate di $1e^{-3}$;
- Funzione di loss minimizzata: Mean Squared Error.

Definite le caratteristiche della rete, sono stati impostati i seguenti input del modello:

- carico elettrico in w ore precedenti, ovvero al tempo $t-1, t-2, \dots, t-w$;
- ora di interesse (codificata con il seno e coseno);
- indicatore se il giorno corrente è festivo o meno;
- giorno della settimana corrente (codificato con il seno e coseno);
- *day code* (1 se il giorno è feriale, -1 se è festivo o domenica, 0.6 se è un giorno feriale vicino a meno di 3 giorni da un giorno festivo).

In Tabella 3. Iperparametri ottimali per ciascun SM sono riportati i parametri ottimali per i singoli *smart meters*, ovvero i valori degli iperparametri che consentono di avere i minori valori di CV. Come nel caso delle *FFNN*, solo per questi particolari casi, saranno riportati e discussi i principali risultati della previsione.

Tabella 3. Iperparametri ottimali per ciascun SM

SM	Lag (ore indietro)	Hidden Dimension
3	2	100
112	48	150
232	2	5
103	48	150
253	24	200
262	2	50

3.5.5. LSTM

Il modello di rete *Long Short-Term Memory* è stato implementato mediante linguaggio Python utilizzando le seguenti librerie: Keras su Tensorflow backend, Scikit-learn, Numpy, pandas, Scipy. Nel dettaglio, la rete implementata è caratterizzata da: un Livello LSTM, un Livello di Batch Normalization e un livello Fully Connected con un unico neurone di uscita e funzione di attivazione ReLU.

Al fine di individuare la configurazione migliore, in termini di minor CV ottenuto, sono stati testati i seguenti parametri:

- *hidden dimension* = [5, 25, 50, 100, 150, 200, 250, 300];
- numero di lag da considerare = [2, 3, 6, 12, 24, 48].

Come per il caso CNN, per ogni configurazione, il numero di epoche di apprendimento è stato selezionato attraverso un approccio *Early Stopping*, e gli altri parametri sono stati settati in base a delle valutazioni a campione. Tra questi i più importanti sono:

- Batch Size: 32;
- algoritmo di ottimizzazione: Adam (Adaptive Moment Estimation) con Learning Rate di $1e^{-3}$;
- Funzione di loss minimizzata: Mean Squared Error.

Definite le caratteristiche della rete, sono stati impostati i seguenti input del modello:

- carico elettrico in w ore precedenti, ovvero al tempo $t-1, t-2, \dots, t-w$;
- ora di interesse (codificata con il seno e coseno);
- indicatore se il giorno corrente è festivo o meno;
- giorno della settimana corrente (codificato con il seno e coseno);
- *day code* (1 se il giorno è feriale, -1 se è festivo o domenica, 0.6 se è un giorno feriale vicino a meno di 3 giorni da un giorno festivo).

In Tabella 4 sono riportati i parametri ottimali per i singoli *smart meters*, ovvero i valori degli iperparametri che consentono di avere i minori valori di CV. Come negli altri casi, solo per questi particolari casi, saranno riportati e discussi i principali risultati della previsione.

Tabella 4. Iperparametri ottimali per ciascun SM

SM	Lag (ore indietro)	Hidden Dimension
3	3	250
112	24	5
232	3	5
103	48	250
253	24	250
262	24	150

3.5.6. XGBoost

Il modello XGBoost è stato implementato mediante linguaggio Python utilizzando le seguenti librerie: xgboost, Scikit-learn, Numpy, pandas, Scipy. Al fine di individuare la configurazione migliore, in termini di minor CV ottenuto, è stato considerato un numero di lag variabile = [2, 3, 6, 12, 24, 48], mentre gli altri parametri sono stati settati in base a delle valutazioni a campione. Tra questi i più importanti sono:

- numero di stimatori: 1000;
- massima profondità dell'albero: 2;
- Learning Rate: 0.1.

In questo caso gli input al modello sono stati i seguenti:

- carico elettrico in w ore precedenti, ovvero al tempo $t-1, t-2, \dots, t-w$;
- ora di interesse (codificata con il seno e coseno);
- indicatore se il giorno corrente è festivo o meno;
- giorno della settimana corrente (codificato con il seno e coseno);
- mese corrente;
- numero della settimana corrente.

In Tabella 5 sono riportati i parametri ottimali per i singoli *smart meters*, ovvero i valori degli iperparametri che consentono di avere i minori valori di CV. Come negli altri casi, solo per questi particolari casi, saranno riportati e discussi i principali risultati della previsione.

Tabella 5. Iperparametri ottimali per ciascun SM

SM	Lag (ore indietro)
3	6
112	24
232	48
103	48
253	3
262	48

3.5.7. SVR

Il *Support Vector Regression* è stato implementato mediante linguaggio Python utilizzando le seguenti librerie: Scikit-learn, Numpy, pandas, Scipy. Al fine di individuare la configurazione migliore, in termini di minor CV ottenuto, è stato considerato un numero di lag variabile = [2, 3, 6, 12, 24, 48]. Per ogni configurazione è stato selezionato il modello con gli iperparametri più promettenti secondo un *3-Fold Cross Validation*. In questo frangente, gli iperparametri valutati sono stati:

- kernel = ['rbf', 'linear', 'poly', 'sigmoid'];
- C: [1e0, 1e1, 1e2, 1e3].

Come per l'XGBoost, anche in questo caso gli input al modello sono stati i seguenti:

- carico elettrico in w ore precedenti, ovvero al tempo $t-1, t-2, \dots, t-w$;
- ora di interesse (codificata con il seno e coseno);
- indicatore se il giorno corrente è festivo o meno;
- giorno della settimana corrente (codificato con il seno e coseno);
- mese corrente;
- numero della settimana corrente.

In Tabella 6 sono riportati i parametri ottimali per i singoli *smart meters*, ovvero i valori degli iperparametri che consentono di avere i minori valori di CV. Come negli altri casi, solo per questi particolari casi, saranno riportati e discussi i principali risultati della previsione.

Tabella 6. Iperparametri ottimali per ciascun SM

SM	Lag (ore indietro)	C	Kernel
3	24	100	Linear
112	24	1000	RBF
232	24	1000	RBF
103	24	1000	Polynomial
253	24	1000	Polynomial
262	24	1000	Polynomial

3.6. MODELLI SVILUPPATI PER LA PREVISIONE DEL CARICO A 24 H

Al fine di ridurre il costo computazionale connesso alla previsione del carico a n ore, questa è stata effettuata mediante lo sviluppo dei modelli mediamente più performanti nel caso di previsione del carico a 1h. Pertanto, i modelli sviluppati sono stati quello della regressione lineare, il modello di rete neurale *FFNN* e il metodo *XGBoost*. È stato sperimentato un approccio iterativo alla previsione multistep con i modelli lineare e *XGBoost*, ed un approccio diretto multi-rete con il modello *FFNN*.

3.6.1. Regressione lineare

Nel caso di previsione del carico a 24h, è stato utilizzato il modello di Regressione Lineare Iterativa. Nel dettaglio, la strategia multi-step iterativa viene realizzata applicando un modello che prevede un singolo passo avanti (one-step) e dando il suo output in input al modello stesso per predire il passo successivo.

Generalizzando la formulazione di un predittore lineare one-step univariato di ordine p definita dall'eq. (4):

$$\hat{y}(t) = b_0 + \sum_{i=1}^p b_i y(t - i) \quad (19)$$

con p lag (numero di valori passati usati per prevedere quelli futuri) e $t \in \{p + 1, \dots, N\}$ con N osservazioni nella serie, un predittore lineare multistep può essere definito ricorsivamente come:

$$\hat{y}(t + h) = b_0 + \sum_{i=1}^p b_i \hat{y}(t + h - i) \quad (20)$$

dove $h > 0$ denota l'orizzonte della previsione (numero di passi avanti) e $\hat{y}(t - i) = y(t - i), \forall i$.

In questo studio, il modello nell'eq.2, con due lag in input, è stato applicato con $i=24$, portando a:

$$\begin{aligned} \hat{y}(t) &= b_0 + b_1 y(t - 1) + b_2 y(t - 2) \\ \hat{y}(t + 23) &= b_0 + b_1 \hat{y}(t + 22) + b_2 \hat{y}(t + 21) \end{aligned} \quad (21)$$

Il metodo iterativo (anche detto ricorsivo) risente dell'accumulo di errori nei vari passi di previsione. Usando infatti la previsione fornita in output come input al passo successivo, eventuali errori andranno a propagarsi, riflettendo un bias del metodo [48]. Ciò accade specialmente quando l'orizzonte della previsione supera il numero di lag e tutti gli input sono valori previsti, non reali [49].

3.6.2. FFNN

La previsione del carico a 24 h nel caso di utilizzo del metodo *FFNN* è stata di tipo day-ahead, al fine di prevedere il carico del giorno dopo, partendo dalle ore 23 del giorno corrente. A tal proposito, sono state sviluppate 24 reti del tipo presentato nel paragrafo 3.5.3. La differenza rispetto alla previsione del carico a 1h consiste nelle variabili di input utilizzate. Per la previsione a 24 h, infatti, gli input scelti sono stati i seguenti:

- carico elettrico alla stessa ora del giorno prima, ovvero all'istante $t-24$;
- carico elettrico alla stessa ora di due giorni prima, ovvero all'istante $t-48$;
- seno del giorno della settimana;
- coseno del giorno della settimana;
- *day-code*.

Anche in questo caso, per la scelta del numero di neuroni da assegnare allo strato *hidden* di ciascuna rete, è stata implementata un'analisi parametrica, facendo variare il numero di neuroni nello strato *hidden* da 5 a 50 con passo 1. Nella Tabella 7 sono riportati, per ciascun *smart meter* e per ciascuna rete (ora), il numero di neuroni ottimale risultante dall'analisi parametrica, ovvero quel determinato numero di neuroni dello strato *hidden* che ha permesso di ottenere i più bassi valori del coefficiente di variazione. Dai dati riportati in tabella, risulta evidente che la definizione del numero di neuroni da assegnare allo strato *hidden* non può essere eseguita a priori e richiede, pertanto, per ciascun caso, un processo di ottimizzazione. Questi infatti presentano una forte variabilità sia al variare dello *smart meter* che dell'ora considerata.

Come per il caso di previsione del carico a 1h, solo per questi particolari casi, saranno riportati e discussi i principali risultati.

Tabella 7. Numero di neuroni ottimali

SM	Numero di neuroni ottimale per ciascuna ora																							
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
3	13	25	8	48	5	36	5	15	33	40	13	5	22	26	5	40	43	40	42	44	30	8	46	41
112	38	10	28	5	12	10	12	28	30	9	15	47	13	30	50	18	17	20	44	8	13	27	6	45
232	20	12	18	30	29	17	13	12	10	13	9	19	5	9	18	16	26	27	32	6	10	12	17	20
103	28	25	14	33	33	25	26	46	43	45	44	37	12	30	5	34	39	48	18	34	19	17	39	45
253	23	42	18	9	6	6	38	36	12	28	41	6	33	15	38	32	13	17	45	6	17	9	5	36
262	21	22	21	31	49	6	12	10	48	5	44	29	39	5	34	45	22	10	18	18	47	27	12	50

3.6.3. XGBoost

Per la previsione del carico a 24 ore, il modello XGBoost è stato utilizzato come blocco costitutivo di un approccio *Recursive Multistep*. Pertanto, il modello di previsione a 1h, presentato nel paragrafo 3.5.5, è stato utilizzato in modo ricorsivo, cosicché la previsione del carico di una specifica ora possa utilizzare, come input, la previsione ottenuta per un certo numero di ore precedenti, come indicato nella Figura 19 sottostante.

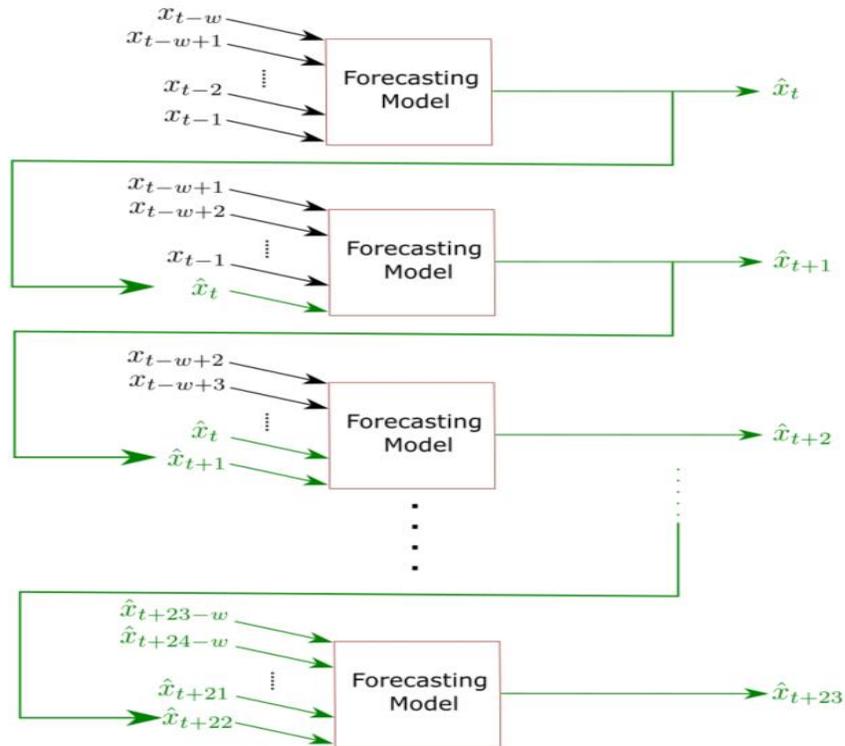


Figura 19. Schema di funzionamento di un modello ricorsivo

In questo caso, l'iperparametro testato è stato il numero di lag da considerare = [6, 12, 24, 48], mentre gli altri parametri e le variabili di input sono stati settati come riportato nel paragrafo 3.5.5.

Come consuetudine, nella Tabella 8 che segue sono riportati i parametri ottimali per i singoli *smart meters*, ovvero i valori degli iperparametri che consentono di avere i minori valori di CV. Come negli altri casi, solo per questi particolari casi, saranno riportati e discussi i principali risultati della previsione.

Tabella 8. Iperparametri ottimali per ciascun SM

SM	Lag (ore indietro)
3	12
112	48
232	48
103	48
253	12
262	24

4. RISULTATI

Nei paragrafi che seguono sono riportati i risultati delle simulazioni effettuate. Al fine di valutare e confrontare tutti i risultati ottenuti è stata definita una metrica, basata sulla valutazione dei seguenti parametri di prestazione, differenziati per training set e test set:

- Coefficiente di Prestazione (CV);
- CV medio giornaliero;
- Deviazione standard del CV giornaliero;
- Root Mean Square Error (RMSE);
- RMSE medio giornaliero;
- Deviazione standard dell'RMSE giornaliero.

Il CV e l'RMSE sono delle misure puntuali d'errore, calcolate rispettivamente con le eq. (22) e (23).

$$CV = \frac{\sqrt{\frac{1}{N-1} \sum_{i=1}^N (y_i - \hat{y}_i)^2}}{\bar{y}} * 100 \quad (22)$$

$$RMSE = \sqrt{\frac{\sum_{i=1}^N (\hat{y}_i - y_i)^2}{N}} \quad (23)$$

in cui, y_i è il valore di target assegnato, \hat{y}_i è il valore predetto, \bar{y} è il valore medio dei valori di target assegnati, N è il numero di dati considerati nel test.

Il CV è un parametro adimensionale, solitamente espresso in percentuale ed è una misura di dispersione dell'errore, che per la sua formulazione agevola il confronto tra risultati ottenuti con metodi o su dataset diversi. Sul CV hanno un gran peso gli *outlier*, ovvero le previsioni molto distanti dal valore reale, per effetto del termine al quadrato; pertanto, minore è il CV, migliore è il risultato della previsione.

L'RMSE è espresso con la stessa unità di misura della variabile considerata, che, in questo caso è la potenza espressa in Watt. Data la sua formulazione, minore è l'RMSE, migliore è il risultato della previsione.

Nel caso delle prove effettuate sul dataset ASHRAE 93, invece, oltre al CV calcolato sul test set e sul training set, sono stati valutati i seguenti indici di prestazione, anch'essi sia per il test set che per il training set, al fine di effettuare un confronto con il lavoro utilizzato come riferimento:

- Mean Bias Error (MBE);
- Mean Absolute Percentage of Error (MAPE).

L'MBE e il MAPE sono stati calcolati, rispettivamente, in accordo all'eq. (24) e (25):

$$MBE = \frac{1/(N-1) * \sum_{i=1}^N (y_i - \hat{y}_i)}{\bar{y}} * 100 \quad (24)$$

$$MAPE = \frac{1}{N} \sum_{i=1}^N \frac{|y_i - \hat{y}_i|}{y_i} * 100 \quad (25)$$

4.1. RISULTATI DEI TEST EFFETTUATI SUL DATASET DI LETTERATURA

La Tabella 9 riporta i risultati ottenuti nel caso della previsione del carico relativo al dataset di consumo ASHRAE 93. Questa evidenza come, nel caso analizzato, il modello di rete neurale di tipo *feed forward* a 7 input determini, tra i tre modelli sviluppati, il risultato migliore in termini di minor CV ottenuto. Inoltre, il confronto con i valori di letteratura, riportati nell'ultima riga, mostra un'ottima sovrapposizione dei risultati.

Tabella 9. Risultati per il dataset ASHRAE 93

Modello	Migliore Configurazione	CV Test Set (%)	MBE Test Set (%)	MAPE Test Set (%)	CV Training Set (%)	MBE Training Set (%)	MAPE Training Set (%)
Regressione lineare	-	3.1					
FFNN - 2 input	28 neuroni	3.8	-0.6	7.4*10 ⁻⁴	3.1	-8.7*10 ⁻⁴	9.9*10 ⁻⁵
FFNN - 7 input	20 neuroni	2.8	-0.4	0	2.0	2.4*10 ⁻⁹	4.8*10 ⁻⁴
Rif. Edwards [2] - FFNN - 7 input	10 - 15 neuroni	2.9	0.6	1.8			

4.2. RISULTATI DELLA PREVISIONE A 1H

Nei paragrafi che seguono, sono riportati tutti i risultati, per ciascun modello e per ciascun *smart meter* analizzato, nel caso del forecasting del carico a un'ora. Sono prima riportati i risultati, in formato tabellare, per i tre *smart meter* domestici e successivamente per quelli non domestici. Per ciascun SM sono evidenziati i migliori risultati ottenuti.

4.2.1. UtENZE di tipo domestico

4.2.1.1. SM 3

Tabella 10. Risultati della previsione a 1h per lo SM 3

Modello	RMSE Training Set (W)	CV Training Set (%)	RMSE Test Set (W)	CV Test Set (%)	RMSE Test Set (media +- devstd giornaliera)	CV Test Set (media +- devstd giornaliera)
Persistenza	297.3	78.9	415.1	74.4	349.7+-224.5	61.9+-24.0
Regressione lineare univariata	280.6	74.4	402.1	72.1	336.5+-220.8	59.0+-22.7
Regressione lineare multivariata	264	70	394.7	70.8	329.9+-217.4	58.3+-21.0
Regressione lineare + persistenza WE	285.6	75.8	412.8	74	344.8+-227.8	60.5+-23.9
Regressione lineare a tratti	296.7	78.7	406.5	72.9	347.5+-211.5	62.6+-23.7
FFNN - 2 input	251.9	66.9	380.0	68.2	315.5+-214.7	56.6+-21.6
FFNN - 7 input	257.9	68.5	389.7	69.9	325.7+-216.78	58.4+-21.3
CNN	254.6	67.5	391.6	70.2	325.1+-218.8	57.7+-21.7
LSTM	254.2	67.4	405.9	72.7	338.3+-224.8	61.3+-20.8
XGBoost	228.4	60.5	391.3	70.0	329.1+-211.7	59.9+-19.1
SVR	293.1	77.8	451.3	80.3	370.2+-258.8	62.6+-24.7

4.2.1.2. SM 112

Tabella 11. Risultati della previsione a 1h per lo SM 112

Modello	RMSE Training Set (W)	CV Training Set (%)	RMSE Test Set (W)	CV Test Set (%)	RMSE Test Set (media +- devstd giornaliera)	CV Test Set (media +- devstd giornaliera)
Persistenza	217.7	65.8	173.1	66.3	137.8+-102.7	48.5+-28.4
Regressione lineare univariata	190.8	57.7	153.2	58.7	130.8+-77.5	47.4+-19.0
Regressione lineare multivariata	182.9	55.3	150.6	57.7	129.2+-75.9	46.7+-19.0
Regressione lineare + persistenza WE	199.4	60.3	162.4	62.2	139.0+-82.5	50.5+-20.0
Regressione lineare a tratti	201.9	61	155.9	59.7	129.2+-85.6	46.0+-23.1
FFNN - 2 input	170.62	52.02	147.22	56.39	125.18+-76.63	49.30+-19.78
FFNN - 7 input	168.69	51.43	147.84	56.63	128.91+-71.31	48.24+-17.71
CNN	165.5	50.4	148	57.3	125.6+-78.6	46.9+-20.7
LSTM	170.8	51.9	147	56.7	128.4+-72.9	48.3+-18.9
XGBoost	161.9	49.2	144.9	55.8	123.8+-75.8	46.3+-20.7
SVR	165.2	50.2	147.1	56.7	119.3+-84.4	43.6+-22.0

La Figura 20 riporta l'andamento della potenza elettrica assorbita effettiva e prevista nel caso dello SM 112 ottenuta con il modello XGBoost.

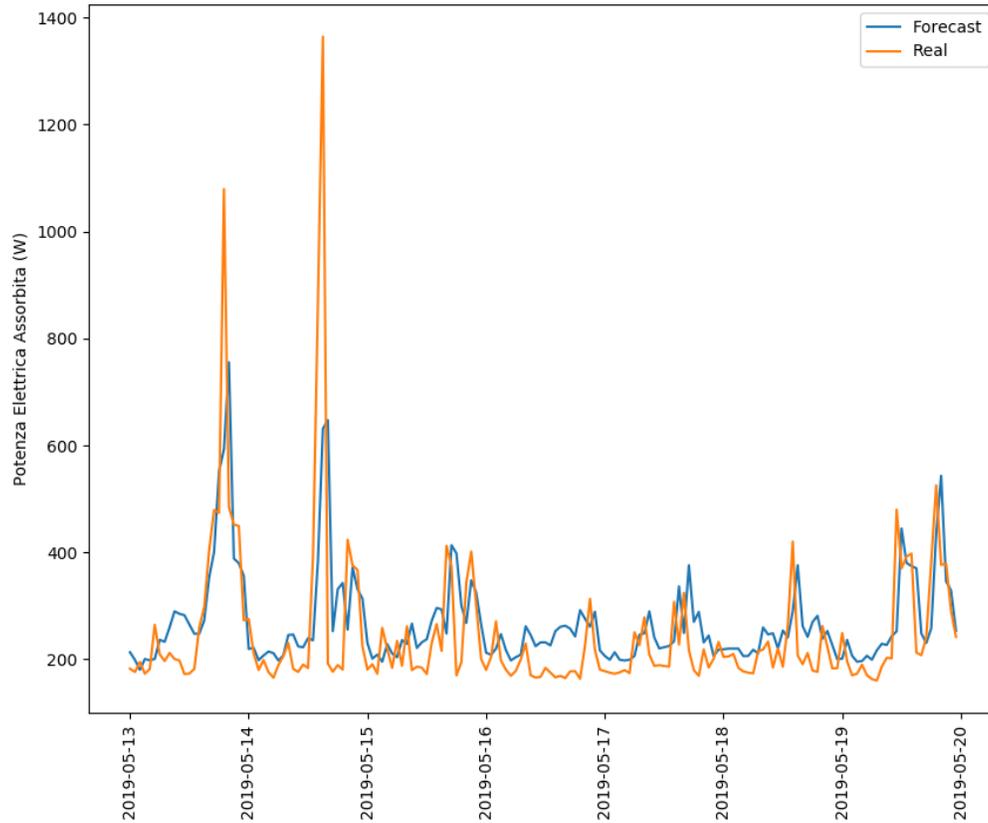


Figura 20. Potenza elettrica assorbita effettiva e prevista nel caso dello SM 112 ottenuta con il metodo XGBoost

4.2.1.3. SM 232

Tabella 12. Risultati della previsione a 1h per lo SM 232

Modello	RMSE Training Set (W)	CV Training Set (%)	RMSE Test Set (W)	CV Test Set (%)	RMSE Test Set (media +- devstd giornaliera)	CV Test Set (media +- devstd giornaliera)
Persistenza	317.3	72.3	248.2	53.9	224.8+-98.5	49.7+-21.0
Regressione lineare univariata	278.4	63.4	224.1	48.6	208.0+-77.1	46.7+-15.0
Regressione lineare multivariata	267.2	60.9	219.6	47.7	203.6+-75.6	45.7+-16.0
Regressione lineare + persistenza WE	273.6	62.4	232.6	50.5	214.4+-84.6	47.3+-13.4
Regressione lineare a tratti	300.5	68.5	264	57.3	245.1+-93.3	54.0+-20.1
FFNN - 2 input	251.8	57.7	216.7	47.0	200.9+-75.0	46.4+-16.6
FFNN - 7 input	242.2	55.5	221.3	48.0	207.7+-69.8	47.9+-14.5
FFNN – 7 input + dati meteo	237.0	54.3	216.01	46.9	150.2+-81.9	39.3+-14.8
CNN	243.1	55.4	219.3	47.6	204.4+-75.0	47.4+-15.8
LSTM	231.2	52.7	216.2	46.9	202.4+-70.2	46.6+-13.6
XGBoost	217	49.7	207.1	44.9	192.7+-71.2	44.4+-14.5
SVR	227.6	51.9	229.2	49.9	206.8+-94.5	45.9+-14.2

4.2.2. Utenze di tipo non domestico

4.2.2.1. SM 253 – Falegnameria

Tabella 13. Risultati della previsione a 1h per lo SM 253

Modello	RMSE Training Set (W)	CV Training Set (%)	RMSE Test Set (W)	CV Test Set (%)	RMSE Test Set (media +- devstd giornaliera)	CV Test Set (media +- devstd giornaliera)
Persistenza	6963.5	82.1	6227.6	80.2	5361.2+-3226.3	57.3+-26.9
Regressione lineare univariata	6580.5	77.6	5905.7	76	5339.9+-2575.9	82.4+-38.2
Regressione lineare multivariata	6042.7	71.2	5638	72.6	5028.0+-2639.1	68.1+-13.8
Regressione lineare + persistenza WE	3789.5	44.7	3046.7	39.2	2632.7+-1564.1	31.0+-17.8
FFNN - 2 input	5241.1	63.3	5321.0	68.5	4745.3+-2516.0	63.5+-29.8
FFNN - 7 input	3542.6	42.8	3949.0	50.9	3622.5+-1629.0	65.1+-55.3
CNN	3006.2	34.9	3309.1	41.6	3043.3+-1321.6	51.8+-38.3
LSTM	3074.3	35.7	3197.9	40.2	2883.9+-1408.7	45.5+-46.4
XGBoost	3380.2	39.8	3320.3	42.6	3117.8+-1191.0	64.0+-51.6
SVR	4512.8	52.4	3929.5	49.4	3538.0+-1763.4	53.8+-43.6

4.2.2.2. SM 103 – Piccolo hotel

Tabella 14. Risultati della previsione a 1h per lo SM 103

Modello	RMSE Training Set (W)	CV Training Set (%)	RMSE Test Set (W)	CV Test Set (%)	RMSE Test Set (media +- devstd giornaliera)	CV Test Set (media +- devstd giornaliera)
Persistenza	1622.1	15.9	1666.3	20.7	1598.2+-463.5	19.7+-4.6
Regressione lineare univariata	1531	15	1646.1	20.4	1603.4+-364.7	19.9+-3.8
Regressione lineare + persistenza WE	1169.1	11.5	1396.7	17.3	1358.1+-322.4	17.0+-4.0
FFNN - 2 input	1480.22	14.58	1684.73	20.91	1630.3+-425.16	20.65+-4.7
FFNN - 7 input	1167.19	11.49	1572.62	19.57	1548.9+-297	20.06+-5.3
CNN	1305.5	12.9	1749.1	21.8	1704.9+-389.6	22.0+-5.7
LSTM	1511.4	14.9	1338.6	16.7	1285.0+-373.0	16.4+-4.4
XGBoost	1122.5	11.1	1871.9	23.3	1829.4+-396.1	24.0+-7.4
SVR	1087.7	10.7	1700.6	21.1	1649.5+-411.7	21.4+-6.6

4.2.2.3. SM 262 – Software house

Tabella 15. Risultati della previsione a 1h per lo SM 262

Modello	RMSE Training Set (W)	CV Training Set (%)	RMSE Test Set (W)	CV Test Set (%)	RMSE Test Set (media +- devstd giornaliera)	CV Test Set (media +- devstd giornaliera)
Persistenza	1465.6	18.1	1491.9	21.3	1368.9+-629.7	18.6+-8.3
Regressione lineare univariata	1410.0	17.4	1441.7	20.6	1366.6+-501.8	19.1+-6.1
Regressione lineare multivariata	1387.3	17.1	1485.8	21.2	1401.0+-526.1	19.8+-7.0
Regressione lineare + persistenza WE	966.8	11.9	1467.2	20.9	1331.7+-655.1	18.3+-9.2
Regressione lineare a tratti	1407.3	17.3	1433.5	20.5	1366.2+-477.7	19.2+-5.9
FFNN - 2 input	1356.2	16.8	1397.0	20.0	1307.4+-546.0	18.3+-6.7
FFNN - 7 input	1154.1	14.3	1368.2	19.6	1335.6+-352.1	20.0+-7.2
FFNN - 7 input + dati meteo	1138.74	14.07	1353,00	19.32	1326.1+-301.92	19.87+-6.82
CNN	1006	12.4	1244	17.8	1193.2+-367.6	17.6+-5.9
LSTM	791.2	9.7	1369.3	19.5	1314.2+-473.2	18.8+-5.3
XGBoost	752.6	9.3	1313	18.8	1260.6+-385.5	18.7+-6.4
SVR	915.5	11.3	1200.8	17.1	1161.4+-333.5	17.2+-5.9

4.3. RISULTATI DELLA PREVISIONE A 24H

Nei paragrafi che seguono, sono riportati tutti i risultati, per ciascun modello e per ciascun *smart meter* analizzato, nel caso del forecasting del carico a 24 ore di tipo *day-ahead*. Come sopra, sono prima riportati i risultati, in formato tabellare, per i tre *smart meter* domestici e successivamente per i tre non domestici.

4.3.1. Utenze di tipo domestico

4.3.1.1. SM 3

Tabella 16. Risultati della previsione *day-ahead* per lo SM 3

Modello	RMSE Training Set (W)	CV Training Set (%)	RMSE Test Set (W)	CV Test Set (%)
Modello Lineare	409.7	108.7	697.3	125.1
XGBoost	401.9	107.9	689.7	122.7
FFNN	564.0	134.8	396.0	85.5

4.3.1.2. SM 112

Tabella 17. Risultati della previsione *day-ahead* per lo SM 112

Modello	RMSE Training Set (W)	CV Training Set (%)	RMSE Test Set (W)	CV Test Set (%)
Modello Lineare	225.4	68.1	181.3	69.4
XGBoost	193.0	58.6	174.3	67.4
FFNN	130.8	48.8	220.9	56.7

La Figura 21 riporta l'andamento della potenza elettrica assorbita effettiva e prevista nel caso dello SM 112 ottenuta con il modello FFNN.

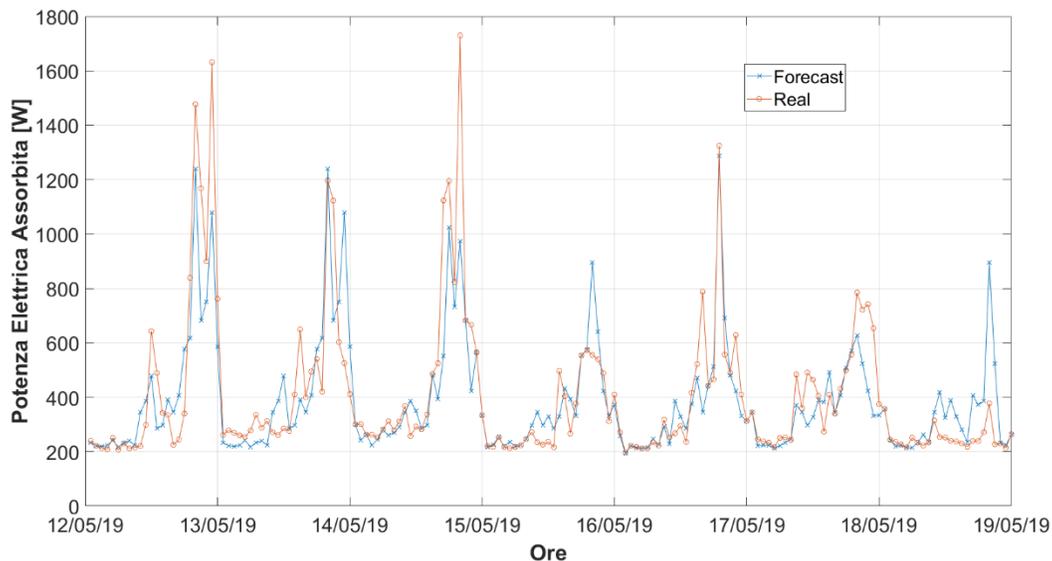


Figura 21. Potenza elettrica assorbita effettiva e prevista nel caso dello SM 112 ottenuta con il metodo FFNN

4.3.1.3. SM 232

Tabella 18. Risultati della previsione *day-ahead* per lo SM 232

Modello	RMSE Training Set (W)	CV Training Set (%)	RMSE Test Set (W)	CV Test Set (%)
Modello Lineare	327.5	74.6	279.2	60.6
XGBoost	275.5	63.2	258.7	56.2
FFNN	230.1	60.1	301.3	51.1

4.3.2. Utenze di tipo non domestico

4.3.2.1. SM 253 – Falegnameria

Tabella 19. Risultati della previsione *day-ahead* per lo SM 253

Modello	RMSE Training Set (W)	CV Training Set (%)	RMSE Test Set (W)	CV Test Set (%)
Modello Lineare	10714.2	126.3	9831.7	126.6
XGBoost	6070.9	69.8	5068.0	63.7
FFNN	6571.8	64.5	5947.2	71.6

4.3.2.2. SM 103 – Piccolo hotel

Tabella 20. Risultati della previsione *day-ahead* per lo SM 103

Modello	RMSE Training Set (W)	CV Training Set (%)	RMSE Test Set (W)	CV Test Set (%)
Modello Lineare	2440.0	24.0	2904.6	36.0
XGBoost	1345.6	13.2	2190.7	27.3
FFNN	1597.9	16.5	1446.5	15.8

4.3.2.3. SM 262 – Software house

Tabella 21. Risultati della previsione *day-ahead* per lo SM 262

Modello	RMSE Training Set (W)	CV Training Set (%)	RMSE Test Set (W)	CV Test Set (%)
Modello Lineare	2672.2	32.9	2445.0	34.9
XGBoost	1250.1	15.4	1583.5	22.4
FFNN	1838.2	21.8	1001.6	15.7

5. DESCRIZIONE DEL TOOL

Nel presente capitolo, è riportata la descrizione dell'intera procedura adottata ai fini della previsione del carico elettrico delle sei utenze UNICAL. Nel dettaglio, la Figura 22 riporta uno schema a blocchi concettuale rappresentativo di tutte le fasi eseguite e le funzioni associate, relativamente a tutti i modelli di calcolo sviluppati. I modelli di forecasting sono stati implementati o mediante linguaggio di programmazione Python (*file.py*) o Matlab (*file.m*). Pertanto, la dicitura *file.script* va intesa come un *file.py* se il modello è stato implementato in Python, come un *file.m* se è stato implementato in Matlab.

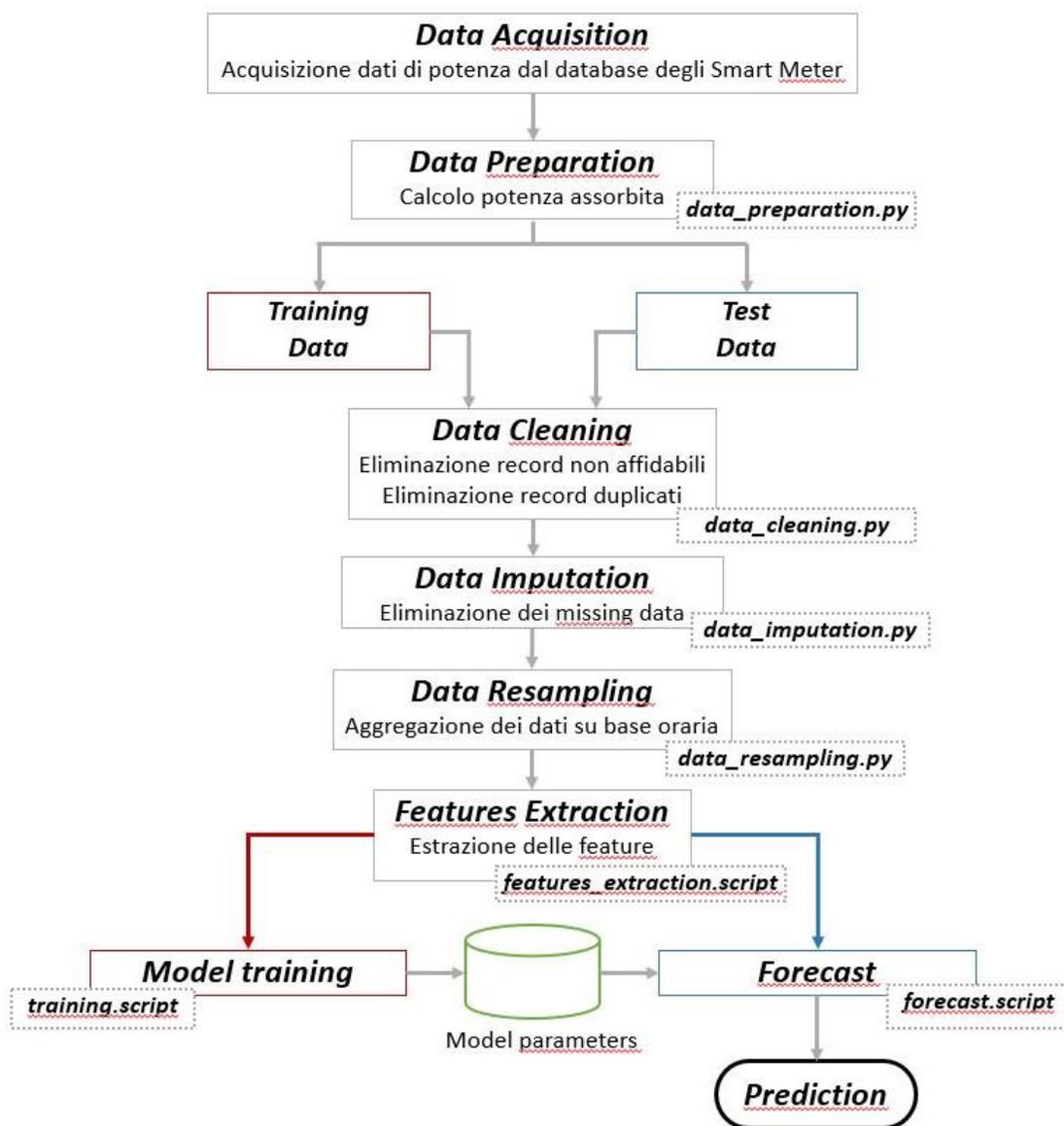


Figura 22. Schema a blocchi relativo alle fasi eseguite nel tool di forecasting del carico

L'intero sistema comprende, quindi, le seguenti fasi:

1. **Acquisizione dati (*Data Acquisition*):** i dati acquisiti dai diversi *Smart Meter* ogni 5 secondi, sono prelevati dal Database *MySQL* messo a disposizione da UNICAL.
2. **Preparazione dei dati (*Data Preparation*):** eseguita mediante la funzione *data_preparation.py*. La preparazione dei dati consiste nell'estrazione della grandezza di interesse, ovvero il carico relativo a ciascun SM.
3. **Pulizia dei dati (*Data Cleaning*):** eseguita mediante la funzione *data_cleaning.py*. Tale procedura consiste nell'eliminazione dei record non affidabili e dei record duplicati.
4. **Applicazione dell'algoritmo di imputazione (*Data Imputation*):** il template utilizzato per la gestione dei *missing data* viene stimato a partire dai soli dati di training. Successivamente, è applicato a qualsiasi dato in ingresso.
5. **Ricampionamento dei dati (*Data Resampling*):** eseguito mediante la funzione *data_resampling.py*, consiste nell'aggregazione dei dati su base oraria.
6. **Estrazione delle variabili di input (*Feature Extraction*):** eseguita a seconda del modello sviluppato, mediante funzione Python o Matlab e pertanto indicata come *feature_extraction.script*. Le variabili di input necessarie a ciascun modello sviluppato sono riportate in modo dettagliato nel report ai paragrafi 3.5 e 3.6.
7. **Training del modello (*Model Training*):** eseguito mediante la funzione *train.script*. Tale procedura, relativa all'apprendimento vero e proprio, adatta i parametri del modello in esame, al fine di ridurre l'errore commesso sui dati di training.
8. **Previsione (*Forecast*):** eseguito mediante la funzione *forecast.script*. Tale procedura utilizza il modello allenato in precedenza e, usando i dati in ingresso, effettua la previsione del carico all'orizzonte temporale richiesto.

Funzione	Input	Output
<i>data_preparation.py</i>	Dati acquisiti in forma tabellare dalla fase 1 di Data Acquisition: <ul style="list-style-type: none"> • Timestamp; • per gli utenti <i>consumer</i> energia prelevata dalla rete elettrica ogni 5 sec; • per gli utenti <i>prosumer</i> energia prelevata dalla rete e energia generata dal sistema di generazione ogni 5 sec. 	Dati preparati in forma tabellare: <ul style="list-style-type: none"> • Timestamp; • Carico elettrico ogni 5 sec.
<i>data_cleaning.py</i>	<ul style="list-style-type: none"> • Output del data preparation 	Dati ripuliti in forma tabellare: <ul style="list-style-type: none"> • Timestamp; • <u>Carico elettrico ogni 5 secondi</u> trattato come riportato nella descrizione del data cleaning.

<i>data_imputation.py</i>	<ul style="list-style-type: none"> • Output del data cleaning 	Dati imputati in forma tabellare: <ul style="list-style-type: none"> • Timestamp; • <u>Carico elettrico ogni 5 secondi</u> trattato come riportato nella descrizione del data imputation
<i>data_resampling.py</i>	<ul style="list-style-type: none"> • Output del data imputation 	Dati ricampionati in forma tabellare: <ul style="list-style-type: none"> • Timestamp con periodo 1h; • <u>Carico ricampionato ogni ora</u> come riportato nella descrizione del data resampling.
<i>train.script</i>	<ul style="list-style-type: none"> • Output del data resampling; • <u>Data inizio training e Data fine training</u> (o numero di record appartenenti al training); • Eventuali dati meteo misurati. 	<ul style="list-style-type: none"> • <u>Parametri del modello allenato</u>, salvati come file MAT (per i modelli implementati in Matlab), o HDF5 (per i modelli implementati in Python)
<i>forecast.script</i>	<ul style="list-style-type: none"> • Parametri del modello; • <u>Dati necessari</u> per effettuare la previsione del carico (dipendenti dal tipo di modello) 	Previsione del carico elettrico. In dipendenza del tipo di modello scelto, la funzione restituirà il valore del carico predetto per 1h successiva, per le 24 h successive o per l'intera giornata successiva.

6. DISCUSSIONE E CONCLUSIONI

Il presente rapporto tecnico riporta le attività di ricerca svolte dal gruppo di lavoro ENEA impegnato, nell'ambito dell'AR 5.3 del progetto di ricerca ComESTo, nell'attività di definizione di modelli di previsione del carico. Nel dettaglio, tale attività ha previsto lo sviluppo di opportuni modelli di previsione dei carichi elettrici associati a sei utenze italiane, tre domestiche e tre non domestiche, di tipo *consumer* o *prosumer*, i cui consumi sono stati forniti dall'Università della Calabria. Con riferimento a queste utenze, l'attività ha richiesto, quindi, la messa a punto, su due differenti orizzonti temporali, di modelli previsionali di carico, allo scopo di ottenere previsioni a 1h, dell'intero profilo giornaliero (day-ahead), e delle 24 ore successive all'ora corrente, necessarie all'implementazione della piattaforma ComESTo. I risultati di tale previsione consentiranno, infatti, l'implementazione di modelli e tecniche per la gestione di programmi avanzati di DR necessari per la gestione efficiente ed integrata dei vettori energetici con l'ottica di favorire il ruolo attivo dell'utente finale nel mercato elettrico e dei servizi di dispacciamento.

L'attività di ricerca condotta ha richiesto lo svolgimento di opportune analisi empiriche specifiche per ciascuna utenza, presentando, queste, andamenti e pattern di consumo differenti.

Dallo studio sull'autocorrelazione delle serie è emerso in tutti i casi un legame debole tra i valori storici della variabile dipendente (potenza assorbita). In particolare, il peso maggiore è rivestito dal valore della grandezza all'istante precedente a quello da predire, mentre il valore assunto a 2 lag ha mostrato un'incidenza minore. Sono state, quindi, implementate diverse strategie di scelta dei lag: nella previsione *one-step*, è stato utilizzato un lag fisso pari a 2, sia per i modelli lineari che per il modello di rete neurale, come riportato al ref [2], mentre sono stati sperimentati vari valori nell'intervallo 2 - 48 con altri metodi non lineari; nella previsione multi-step, invece, per i casi di reti neurali multiple è stato utilizzato il valore storico della grandezza nello stesso istante dei due giorni precedenti.

Dai risultati dell'analisi condotta, riportati in forma tabellare, non emerge un modello predominante. Nei casi domestici, in particolare, è stata rilevata una ridotta capacità di fitting della serie da parte dei modelli implementati; i valori di CV, infatti, risultano piuttosto elevati rispetto a quelli ottenuti nel caso riportato al ref [2]. Tuttavia, la baseline tracciata nel presente studio dal modello lineare risulta anch'essa superiore, ed i risultati ottenuti dai modelli migliori, se letti rispetto a questa linea di partenza, presentano riduzioni del CV confrontabili a quelle del lavoro [2]. Infatti, sebbene in tale riferimento siano analizzati contesti domestici presi singolarmente, i dati analizzati provengono da un contesto meccanizzato, in cui l'utilizzo di elettrodomestici in casa è gestito da controlli automatici, che simulano il comportamento tipico di un utente americano, rimuovendo così la componente dinamica del comportamento umano. Al contrario, i dati analizzati nel presente studio provengono da ambienti reali, con alta volatilità.

Il problema di previsione analizzato risulta influenzato prevalentemente da due fattori: la quantità di record a disposizione per la fase di training e la regolarità, o similmente, la variabilità dei carichi orari. Come riportato ai paragrafi 3.3.1 e 3.3.2, i dati messi a disposizione sono relativi al più ad un periodo di 6 mesi; inoltre, essi mostrano, per gran parte degli SM analizzati, una forte variabilità dei consumi registrati, ponendo l'attività di forecasting in condizioni sfavorevoli.

Gli *smart meter* per cui si registra una previsione più accurata sono gli SM 103 e 262. Per tali SM è stata osservata una minore variabilità dei consumi misurati; pertanto, anche se il numero di record orari è piuttosto esiguo, la previsione ha fornito risultati nettamente migliori rispetto agli altri casi.

I dati meteo misurati hanno mostrato uno scarso impatto sui risultati di previsione per gli SM per cui sono stati testati (SM 232 e SM 262). Questa bassa influenza può essere in parte connessa alla provenienza dei dati meteo, misurati da stazioni meteo non solidali con l'utenza in esame.

Alla luce di ciò, è possibile affermare che tale attività potrebbe condurre a risultati più promettenti se condotta su dataset orari più ampi e proveniente da un maggior numero di *smart meter* omogenei. La disponibilità di un maggior numero di dati, infatti, potrebbe ridurre l'effetto negativo connesso alla variabilità della grandezza in esame, determinando una previsione più accurata. Allo scopo di riuscire ad appianare tali problematiche, potrebbe essere vantaggioso disporre di informazioni circa la natura dei consumi elettrici misurati, mediante la misurazione dei consumi a livello del singolo utilizzatore. Inoltre, potrebbe risultare utile ottenere informazioni sulle condizioni termo-igrometriche degli ambienti interni, mediante installazione di un'opportuna sensoristica.

7. BIBLIOGRAFIA

- [1] J.S. Chou, D.K. Bui. Modeling heating and cooling loads by artificial intelligence for energy-efficient building design. *Energy and Buildings*, 82 (2014), pp. 437–46.
- [2] R.E. Edwards, J. New, L.E. Parker. Predicting future hourly residential electrical consumption: a machine learning case study. *Energy and Buildings*, 49 (2012), pp. 591–603.
- [3] R.K. Jain, K.M. Smith, P.J. Culligan, J.E. Taylor. Forec. energy cons. of multi-family resid. buildings using support vector regression: inv. the impact of temporal and spat. monit. granularity on perform. accuracy. *Applied Energy*, 123 (2014), pp. 168–78.
- [4] R.K. Jain, T. Damoulas, C.E. Kontokosta. Towards data-driven energy consumption forecasting of multi-family residential buildings: feature selection via The Lasso. *Comput Civ Build Eng 2016*.
- [5] L. Xuemei, D. Lixing, L. Jinhu, X. Gang, L. Jibin. A Novel Hybrid Approach of KPCA and SVM for Building Cooling Load Prediction. *2010 Third Int Conf Knowl Discov Data Min 2010:1*.
- [6] L. Jinhu, L. Xuemei, D. Lixing, J. Liangzhong. Applying principal component analysis and weighted support vector machine in building cooling load forecasting. *2010 Int Conf Comput Commun Technol Agric Eng 2010;1:434–7*.
- [7] Q. Li, Q. Meng, J. Cai, H. Yoshino, A. Mochida. Applying support vector machine to predict hourly cooling load in the building. *Applied Energy*, 86 (2009), pp. 2249–56.
- [8] Q. Li, Q. Meng, J. Cai, H. Yoshino, A. Mochida. Predicting hourly cooling load in the building: a comparison of support vector machine and different artificial neural networks. *Energy Conversion and Management*, 50 (2009), pp. 90–6.
- [9] R. Mena, F. Rodríguez, M. Castilla, M.R. Arahál. A prediction model based on neural networks for the energy consumption of a bioclimatic building. *Energy Buildings*, 82 (2014), pp. 142–55.
- [10] S.S.K. Kwok, E.W.M. Lee. A study of the importance of occupancy to building cooling load in prediction by intelligent approach. *Energy Conversion and Management*, 52 (2011), pp. 2555–64.
- [11] M. Georgescu, E. Eccles, V. Manjunath, E. Swindle, I. Mezic. MACHINE LEARNING METHODS FOR SITE-LEVEL BUILDING ENERGY FORECASTING AND DATA RECTIFICATION. *Build. Simul. Optim. - Second IBPSA, London; 2014*.
- [12] S. Paudel, M. Elmtiri, W.L. Kling, O.L. Corre, B. Lacarrière. Pseudo dynamic transitional modeling of building heating energy demand using artificial neural network. *Energy Buildings*, 81 (2014), pp. 81–93.
- [13] A. Yezioro, B. Dong, F. Leite. An applied artificial intelligence approach towards assessing building performance simulation tools. *Energy Buildings*, 40 (2008), pp. 612–20.
- [14] K. Yun K, R. Luck, P.J. Mago, H. Cho. Building hourly thermal load prediction using an indexed ARX model. *Energy Building*, 54 (2012), pp. 225–33.
- [15] C. Turhan, T. Kazanasmaz, I.E. Uygun, K.E. Ekmen, G.G. Akkurt. Comparative study of a building energy performance software (KEP-IYTE-ESS) and ANN-based building heat load estimation. *Energy Buildings*, 85 (2014), pp. 115–25.
- [16] B. Dong, C. Cao, S.E. Lee. Applying support vector machines to predict building energy consumption in tropical region. *Energy Buildings*, 37 (2005), pp. 545–53.
- [17] D. Popescu, F. Ungureanu. Prediction of space heating consumption in district heated apartments. *Energy ASME 2013;6B*.
- [18] G.Escrivá-Escrivá, C. Álvarez-Bel, C. Roldán-Blay, M. Alcázar-Ortega. New artificial neural network

- prediction method for electrical consumption forecasting based on building end-uses. *Energy Buildings*, 43 (2011), pp. 3112–9.
- [19] V.A. Kamaev, M.V. Shcherbakov, D.P. Panchenko, N.L. Shcherbakova, A. Brebels. Using connectionist systems for electric energy consumption forecasting in shopping centers. *Autom Remote Control*, 73 (2012), pp. 1075–84.
- [20] S.L. Wong, K.K.W. Wan, T.N.T. Lam. Artificial neural networks for energy analysis of office buildings with daylighting. *Applied Energy*, 87 (2010), pp. 551–7.
- [21] J.C. Lam, K.K.W. Wan, S.L. Wong, T.N.T. Lam. Principal component analysis and long-term building energy simulation correlation. *Energy Conversion and Management*, 51 (2010), pp. 135–9.
- [22] K. Amasyali, N.M. El-Gohary. A review of data-driven building energy consumption prediction studies. *Renewable and Sustainable Energy Reviews*, 81 (2018), pp. 1192–1205.
- [23] M.A. Mat Daut, M.Y. Hassan, H. Abdullah, H.A. Rahman, M.P. Abdullah, F. Hussin. Building electrical energy consumption forecasting analysis using conventional and artificial intelligence methods: A review. *Ren. Sust. Energy Rev.* 70 (2017), pp. 1108–11.
- [24] H. Zhao, F. Magoulès. A review on the prediction of building energy consumption. *Renewable and Sustainable Energy Reviews*, 16 (2012), pp. 3586– 3592.
- [25] A.S. Ahmad, M.Y. Hassan, M.P. Abdullah, H.A. Rahman, F. Hussin, H. Abdullah, R. Saidur. A review on applications of ANN and SVM for building electrical energy consumption forecasting. *Renew Sustain Energy Rev*, 33 (2014).
- [26] R. Platon, V.R. Dehkordi, J. Martel. Hourly prediction of a building's electricity consumption using case-based reasoning, artificial neural networks and principal component analysis. *Energy Buildings*, 92 (2015), pp. 10–8.
- [27] S. Karatasou, M. Santamouris, V. Geros. Modeling and predicting building's energy use with artificial neural networks: methods and results. *Energy Buildings*, 38 (2006), pp. 949–58.
- [28] A.H. Neto, F.A.S. Fiorelli. Comparison between detailed model simulation and artificial neural network for forecasting building energy consumption. *Energy Buildings*, 40 (2008), pp. 2169–76.
- [29] P.H. Kuo, C.J. Huang. A high precision artificial neural networks model for short-term energy load forecasting. *Energies*, 11 (2018), p. 213.
- [30] J. Zheng, C. Xu, Z. Zhang, and X. Li. Electric load forecasting in smart grids using long-short-term-memory based recurrent neural network. *Information Sciences and Systems (CISS), 2017 51st Annual Conference on, IEEE, 2017*, pp. 1–6.
- [31] F. Zhang, C. Deb, S.E. Lee, J. Yang, K. W. Shah. Time series forecasting for building energy consumption using weighted support vector regression with differential evolution optimization technique. *Energy and Buildings*, 126 (2016), pp. 94–103.
- [32] X.M. Zhang, K. Grolinger, M.A.M. Capretz, L. Seewald. Forecasting Residential Energy Consumption: Single Household Perspective. *2018 17th IEEE International Conference on Machine Learning and Applications*.
- [33] R.A. Abbasi, N. Javaid, M.N.J. Ghuman, Z.A. Khan, S.U. Rehman, Amanullah. Short Term Load Forecasting Using XGBoost, (2019)..
- [34] J. Kreider, J. Haberl. Predicting hourly building energy use: the great energy predictor shootout – overview and discussion of results, *ASHRAE Transactions*, 100 (1994), pp. 1104–1118.
- [35] Dataset ASHRAE93, disponibile al link: <https://www.kaggle.com/c/great-energy-predictor-shootout-i/data>.
- [36] A.G. Bakirtzis, J.B. Theocharis, S.J. Kiartzis, and K.J. Satsois. Short term load forecasting using fuzzy neural

networks. IEEE Trans. Power Syst. 10, pp. 1518–1524.

- [37] S. Karatasou, M. Santamouris, V. Geros. Modeling and predicting building's energy use with artificial neural networks: methods and results. *Energy and Buildings*, 38 (2006), pp. 949–958.
- [38] CS231n: Convolutional Neural Networks for Visual Recognition, <http://cs231n.stanford.edu/> - Lecture Spring 2019.
- [39] D. Rumelhart, G. Hinton, R. Williams. Learning representations by back-propagating errors. *Nature*, 1986.
- [40] I. Goodfellow, Y. Bengio and A. Courville, "Deep Learning", MIT Press, 2016.
- [41] S. Hochreiter, J. Schmidhuber. Long short-term memory. *Neural Computation*, 1997.
- [42] C. Olah, Understanding LSTM Networks, 2015, colah.github.io.
- [43] <https://xgboost.readthedocs.io/en/latest/>.
- [44] C. M. Bishop. *Pattern Recognition and Machine Learning*. Springer Verlag, 2006.
- [45] C. Cortes, V.N. Vapnik. Support vector networks. *Machine Learning*, 1995.
- [46] B. E. Boser, I.M. Guyon, and V. N. Vapnik. A training algorithm for optimal margin classifiers. In COLT '92: Proceedings of the fifth annual workshop on Computational learning theory, New York, 1992.
- [47] E. Alpaydin. *Introduction to Machine Learning*. MIT Press, 2014.
- [48] M. Marcellino, J. H. Stock and M. W. Watson, "A comparison of direct and iterated multistep AR methods for forecasting macroeconomic time series," *Journal of Econometrics*, vol. 135, no. 1-2, pp. 499-526, 2006.
- [49] S. B. Taieb, G. Bontempi, A. F. Atiya and A. Sorjamaa, "A review and comparison of strategies for multi-step ahead time series forecasting based on the NN5 forecasting competition," *Expert Systems with Applications*, vol. 39, no. 8, pp. 7067-7083, 2012.
- [50] J. S. Richmand, J. R. Moorman. Physiological time-series analysis using approximate entropy and sample entropy. *American Journal of Physiology-Heart and Circulatory Physiology*, 2000.
- [51] M. Marcellino, J. H. Stock e M. W. Watson, «A comparison of direct and iterated multistep AR methods for forecasting macroeconomic time series,» *Journal of Econometrics*, vol. 135, n. 1-2, pp. 499-526, 2006.
- [52] S. B. Taieb, G. Bontempi, A. F. Atiya e A. Sorjamaa, «A review and comparison of strategies for multi-step ahead time series forecasting based on the NN5 forecasting competition,» *Expert Systems with Applications*, vol. 39, n. 8, pp. 7067-7083, 2012.
- [53] S. Makridakis, E. Spiliotis e V. Assimakopoulos, «Statistical and Machine Learning forecasting methods: Concerns and ways forward,» *PloS one*, vol. 13, n. 3, 2018.
- [54] S. Makridakis, E. Spiliotis and V. Assimakopoulos, "Statistical and Machine Learning forecasting methods: Concerns and ways forward," *PloS one*, vol. 13, no. 3, 2018.