Previsione consumi di elettricità usando modelli ARIMA, UCM e Machine Learning

Giorgio Carbone matr. 811974[1]

[1] Università degli Studi di Milano Bicocca. Dipartimento di Informatica, Sistemistica e Comunicazione. CdM Data Science

E-mail: g.carbone8@campus.unimib.it

Abstract

La previsione di serie temporali di consumi di elettricità riveste un ruolo di rilievo nell'ambito della gestione efficiente delle risorse e della pianificazione strategica nel settore energetico. In questo progetto si analizza una serie storica univariata, a istanti omogenei e ad alta frequenza, composta da consumi elettrici misurati ogni 10 minuti dal 01/01/2017 al 30/11/2017, confrontando approcci statistici (ARIMA e UCM), di Machine Learning (Random Forest e k-NN) e Deep Learning (GRU Recurrent Neural Network) per modellizzare la serie storica e prevedere i consumi relativi al mese di dicembre 2017. I più performanti tra i modelli di ogni famiglia, addestrati sui dati campionati tra il 01/01/2017 al 31/10/2017 (o su una parte di essi) e validati sui dati di novembre 2017, sono risultati nelle seguenti misure di errore: $\mathrm{MAE}_{\mathrm{ARIMA}} = 1010.08$, $\mathrm{MAE}_{\mathrm{UCM}} = 1183.40$ e $\mathrm{MAE}_{\mathrm{ML}} = 1184.07$, portando l'approccio ARIMA ad essere il più accurato in previsione tra quelli testati.

Keywords: Time Series Forecasting, Time Series Analysis, ARIMA, UCM, Machine Learning, Deep Learning

1. Introduzione

L'analisi, la modellizzazione e infine la previsione di serie storiche relative ai consumi energetici sono attività di cruciale importanza nel settore energetico. La possibilità di catturare e modellare i pattern di consumo passati in una determinata area geografica, e quindi di fornire previsioni affidabili dei consumi futuri, consente alle aziende energetiche di sviluppare strategie consapevoli per una gestione efficiente delle risorse, garantendo maggiore stabilità e affidabilità delle reti e minimizzando l'impatto ambientale.

In questo contesto, il seguente progetto ha come obiettivo analizzare e modellizzare una serie storica di consumi energetici campionati ogni 10 minuti nel periodo compreso tra il 1° gennaio 2017 e il 30 novembre 2017, confrontando l'accuratezza di previsione (fissato un orizzonte di previsione di un mese in avanti) di diversi modelli delle famiglie ARIMA, UCM e Machine Learning. Il modello che dovesse garantire le migliori performance in validazione (in termini di Mean Absolute Error) verrà utilizzato per prevedere i consumi energetici ogni 10 minuti tra il 1° dicembre 2017 e il 30 dicembre 2017 (4320 osservazioni).

2. Data Exploration

Viene fornito un dataset in formato tabellare *.csv*, in cui ogni riga è associata a una delle 48096 osservazioni della serie storica univariata e omogenea in esame, campionate ogni 10 minuti nel periodo compreso tra le 00:00 del 1° gennaio 2017 e le 23:50 del 30 novembre 2017. Il dataset è composto da due variabili (prive di valori duplicati o nulli): la variabile *power* contenente i valori della serie storica, associata alle misurazioni dei consumi energetici, e

date, contenente le *label* data-ora relative agli istanti di misurazione, codificate in formato *dd/mm/yyyy HH:MM:SS*.

La serie storica, visualizzata nella sua interezza in Figura 1, mostra consumi particolarmente elevati nei mesi estivi di luglio e agosto, associati probabilmente all'utilizzo di condizionatori per ambienti (energicamente onerosi), la relativa stagionalità intra-anno associata non risulta però visibile (avendo solo un anno di dati) e per questo non verrà modellata.

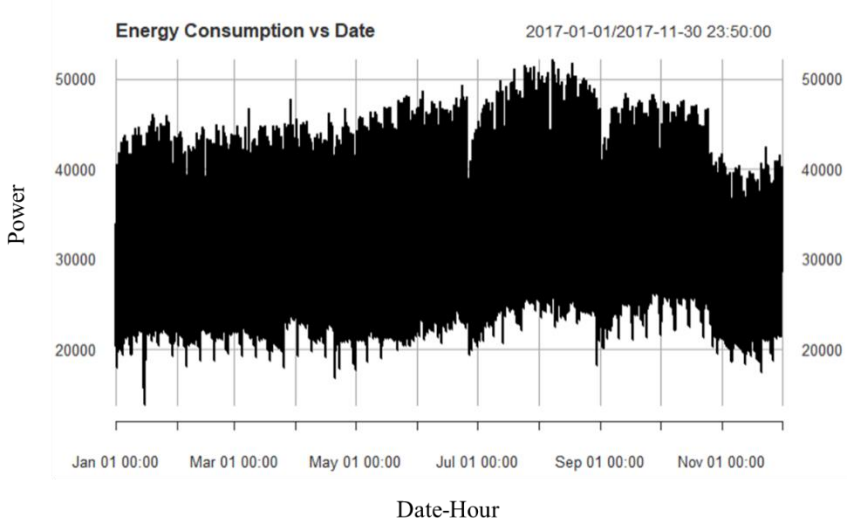

Figura 1: Serie Storica completa dei consumi energetici ogni 10 minuti.

Visualizzando l'andamento della serie storica (***Figura 2*** e ***Figura 3***) per le osservazioni relative alla prima settimana e al primo mese della serie storica è possibile apprezzare due forme di stagionalità che interagiscono tra loro: una evidente stagionalità giornaliera (ogni 24 ore, con un periodicità stagionale di 144 osservazioni), associata al ciclo giorno-notte, che vede i consumi minimi delle prime ore notturne crescere fino a massimizzarsi la sera (momento in cui le persone tendono ad essere a casa); e una stagionalità settimanale (ogni 7 giorni, con una periodicità stagionale di 1008 osservazioni), associata alla massimizzazione dei consumi nei giorni infra-settimanali e una loro forte riduzione nel week-end, probabilmente causata dalla chiusura degli impianti industriali.

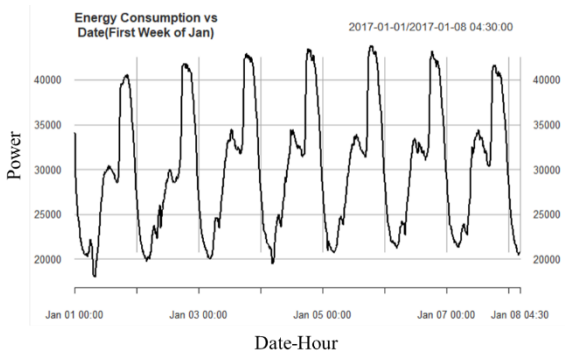

Figura 2: Serie Storica dei consumi energetici ogni 10 minuti nella prima settimana di gennaio.

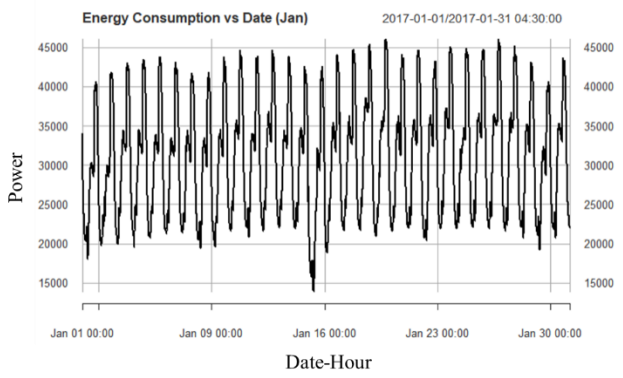

Figura 3: Serie Storica completa dei consumi energetici ogni 10 minuti nel mese di gennaio.

Inoltre, analizzando l'andamento delle differenze tra due misurazioni consecutive (in *Figura 5*) emerge la presenza di alcuni bruschi cali nei consumi (differenze in valore assoluto superiori a 7500), che risultano anomali data l'elevata frequenza della serie storica. Tali anomalie, delle quali la più rilevante è riportata in *Figura 4*, non sono attribuibili alle forme di stagionalità (giornaliera e settimanale) individuate, e sono considerate quindi come *outliers*, scegliendo però di non modellarle perché presenti in numero esiguo.

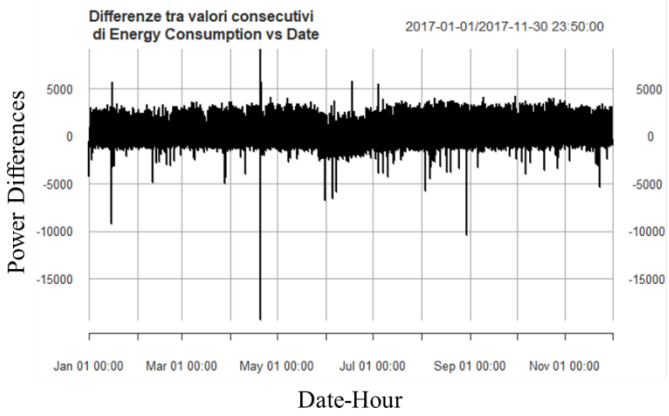

Figura 5: Differenze tra valori consecutivi di consumi ogni 10 minuti.

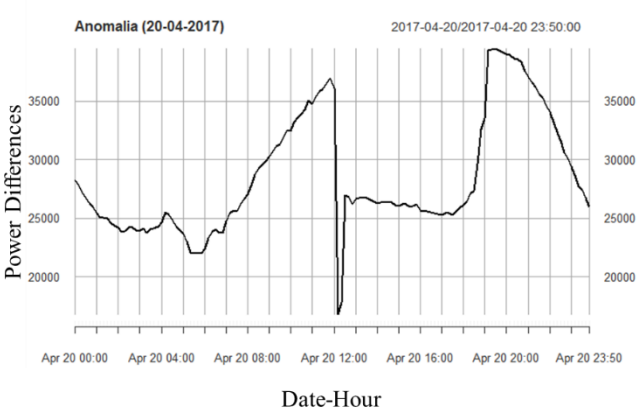

Figura 4: Anomalia: brusco calo nei consumi alle 12:10 del 20-04-2017.

3. Modeling e out-of-sample evaluation (Hold-out)

Con il fine di poter confrontare l'accuratezza in previsione dei modelli proposti, e poter scegliere quelli più performanti (per ognuna delle classi ARIMA, UCM e Machine Learning) da utilizzare per prevedere i consumi relativi a dicembre 2017, si è scelto di effettuare uno *splitting* della serie storica mediante procedura di hold-out. Un primo *subset* della serie storica, che include le 43776 osservazioni con *label* nell'intervallo [01/01/2017 00:00:00, 31/10/2017 23:50:00] è stato utilizzato come *training set* per addestrare i modelli, i quali sono stati poi testati su un *validation set* composto dalle 4320 osservazioni relative al mese di novembre (nell'intervallo [01/11/2017 00:00:00, 30/11/2017 23:50:00]). Questo ha permesso di poter valutare le performance di previsione dei modelli fissando un numero di *step* in avanti di previsione coerente con l'orizzonte di previsione previsto per gli obiettivi del progetto, scegliendo i tre modelli (per ogni famiglia) che minimizzassero la metrica Mean Absolute Error (MAE) per le previsioni validation set, riaddestrandoli poi sull'intera serie storica (concatenando training e validation set) e utilizzandoli per prevedere le 4320 misurazioni nell'intervallo [01/12/2017 00:00:00, 30/12/2017 23:50:00].

3. Modelli ARIMA

3.1 Stazionarietà debole: diagnosi e trasformazioni

Prima di modellare la serie storica, si è provveduto a verificare l'omogeneità (in termini di stazionarietà debole) della stessa, requisito del *modeling* ARMA, valutando poi la necessità di applicare eventuali trasformazioni per stabilizzare il processo. Si è inizialmente studiata la presenza di una possibile non stazionarietà in varianza, non direttamente apprezzabile dall'andamento della serie storica in *Figura 1*, andando quindi ad analizzare la relazione tra la media e la varianza del processo (calcolate raggruppando le 144 osservazioni giornaliere) mediante un grafico di *Box-Cox*. Nel grafico, riportato in Figura 7, o almeno nella parte sinistra dello stesso, sembra emergere una leggera correlazione lineare, sintomo di una possibile non stazionarietà in varianza. Al fine

di stabilizzare il processo in varianza si è quindi scelto inizialmente di applicare una *trasformazione di Box-Cox*, stimando il parametro $\lambda = 0.22$ ottimo mediante il metodo della massima verosimiglianza (il grafico di Box-Cox in *Figura 6*, ottenuto sulla serie storica trasformata, non mostra più una correlazione lineare, la serie storica risulta quindi stabilizzata in varianza). L'approccio però non ha dato buoni risultati, le performance dei modelli stimati sulla serie storica trasformata sono peggiori in fase di validazione rispetto a quando non viene applicata, per questo motivo tutti i modelli mostrati sono stimati sulla serie storica non trasformata.

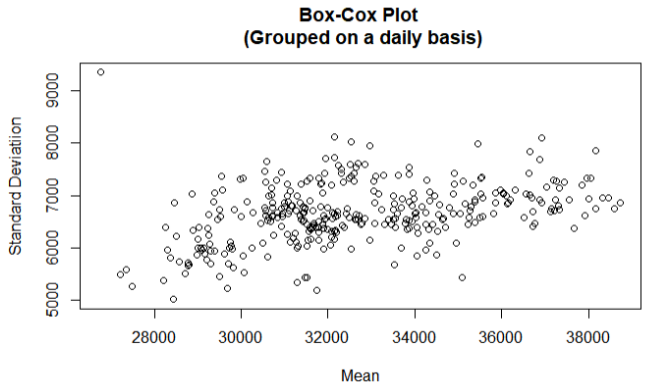

Figura 7: Box-Cox Plot (su dati aggregati giornalmente)

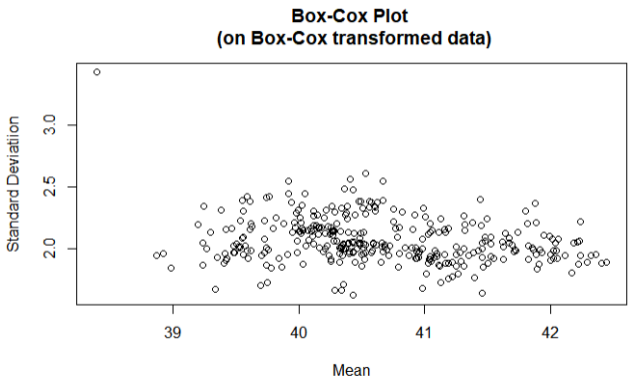

Figura 6: Grafico di Box-Cox su serie storica trasformata (Box-Cox).

Dal grafico della serie storica nel tempo in Figura 1 sembra inoltre emergere la presenza di un andamento persistente (le osservazioni tendono a rimanere sopra o sotto la media nel lungo periodo), sintomo di una possibile non stazionarietà in media dovuta alla presenza di un trend. Inoltre, come precedentemente accennato, la serie storica è caratterizzata da una forma di non-stazionarietà in media dovuta a stagionalità multiple (giornaliera e settimanale).

Sono stati quindi applicati applicati i test *Augmented Dickey-Fuller* e *KPSS* (non stagionali e stagionali) con significatività al 5%, i quali risultati suggeriscono che non sia necessario applicare una differenza semplice al processo ma che sia invece necessario differenziare stagionalmente (con periodicità stagionale di 144 osservazioni, un giorno) per stabilizzare il processo in media. Tale bisogno è confermato dall'andamento delle stime di ACF (Figura 8), che presenta una ciclicità con periodo di 144 osservazioni, e PACF (Figura 8Figura 9) del processo contro il ritardo, che presenta memoria per ritardi multipli di s.

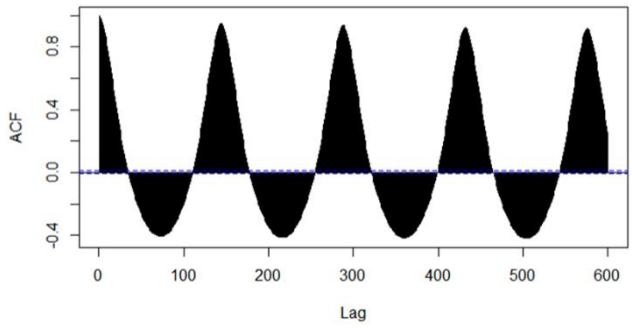

Figura 8: Stima della funzione di autocorrelazione contro il lag.

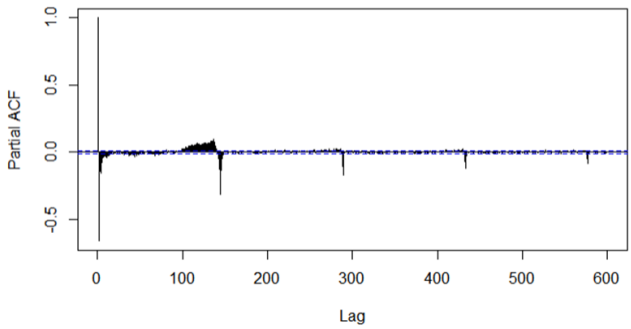

Figura 9: Stima della funzione di autocorrelazione parziale contro il lag.

I modelli proposti di seguito hanno come base la modellizzazione della non stazionarietà in media dovuta alla stagionalità giornaliera, proseguendo iterativamente modificando i parametri $(p, d, q)(P, D, Q)_s$ del modello

ARIMA, integrando eventuali regressori e aggregando le osservazioni, cercando il compromesso che minimizzasse l'errore di previsione MAE sul validation set e quindi massimizzasse le capacità di generalizzazione del modello, piuttosto che la capacità dello stesso di catturare la memoria lineare dei dati di training.

3.2 Modelli ARIMA stagionali senza regressori

I primi modelli testati hanno avuto come scopo modellare la stagionalità giornaliera (periodicità di 144 osservazioni), che risulta dominante dai grafici di ACF e PACF, differenziando stagionalmente. Il primo modello addestrato, il più semplice, ha visto l'applicazione di una differenza stagionale e l'introduzione di una componente SMA di ordine 1, considerando il rientro graduale per lag multipli di 144 osservabile in ***Figura 9***.

$$ARIMA(0,0,0)(0,1,1)_{144} \ (1.1)$$

Il modello addestrato presenta un coefficiente MA significativo e raggiunge un $MAE_{validation} = 1415.85$. L'autocorrelogramma dei residui è sicuramente ancora incompatibile con l'ipotesi di white noise, l'ACF presenta picchi estremamente elevati in valore assoluto (intorno a 1) nei primi ritardi, che rientrano gradualmente, fenomeno associato alla presenza di memoria lineare nel breve periodo non catturata, inoltre si segnalano picchi ogni 1008 osservazioni associati probabilmente alla stagionalità settimanale. La PACF presenta memoria lineare significativamente diversa da 0 nei primi 3 ritardi, uscendo poi a ritardi multipli di 144 con valori molto piccoli di correlazione (la componente SMA(1) sembra aver catturato in parte la memoria stagionale giornaliera).

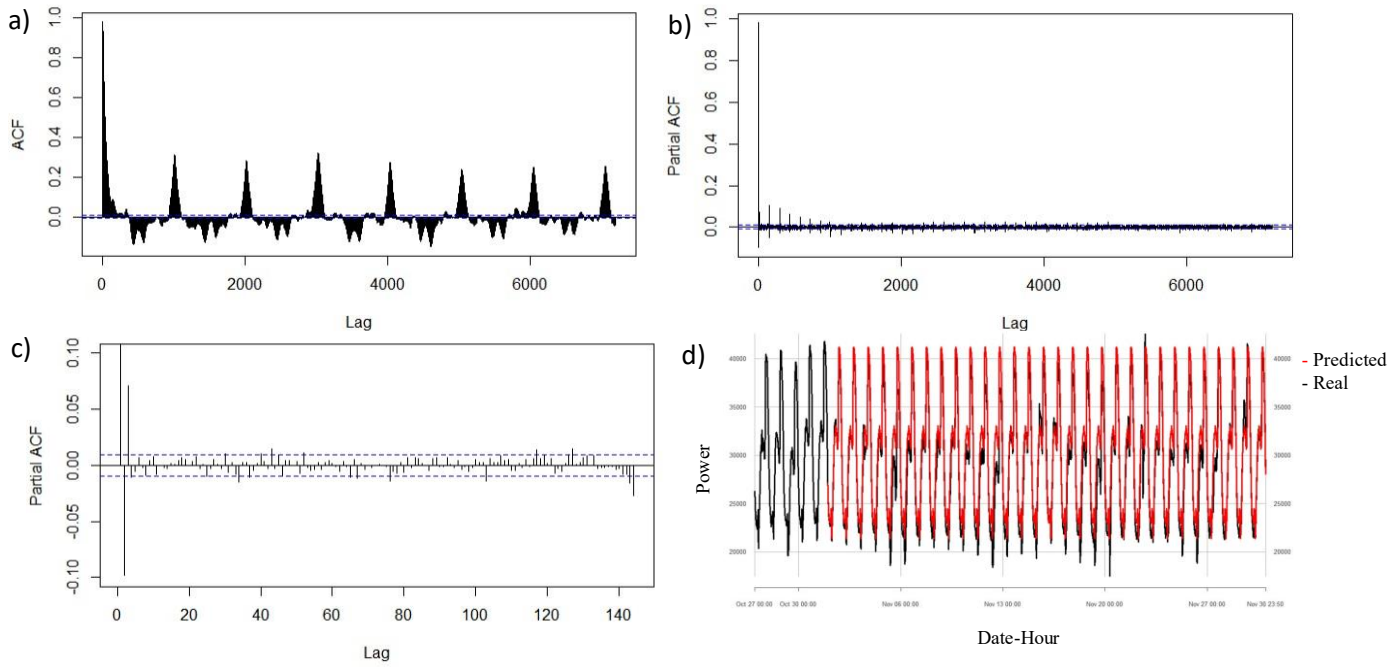

Figura 10: Autocorrelogramma dei residui del modello Arima (1.1): a) ACF, b) PACF e c) PACF zoomata e d) previsioni sul validation set.

Si è quindi proseguito iterativamente modificando i parametri della parte non stagionale del modello (1.1), testando anche l'applicazione di una differenza semplice e cercando in alcuni di essi di catturare la memoria lineare nel breve periodo introducendo una componente AR(p). In particolare, sono stati testati:

$ARIMA(0,1,1)(0,1,1)_{144}$ (1.2) *Airline* $ARIMA(0,0,1)(0,1,1)_{144}$ (1.3) $ARIMA(0,1,0)(0,1,1)_{144}$ (1.4)

$ARIMA(1,0,0)(0,1,1)_{144}$ (1.5) $ARIMA(1,1,0)(0,1,1)_{144}$ (1.6) $ARIMA(3,1,0)(0,1,1)_{144}$ (1.7)

$ARIMA(2,1,1)(0,1,1)_{144}$ (1.8) $ARIMA(1,1,1)(0,1,1)_{144}$ (1.9)

Parte dei modelli testati descrive meglio la serie storica di training, riuscendo a catturare una quantità maggiore di memoria lineare rispetto al modello (1.1), portando a ACF e PACF dei residui con valori di autocorrelazione molto inferiori al modello (1.1) e $MAE_{training}$ migliori, ma le performance in validazione (quindi la capacità di generalizzazione) sono peggiori.

Il modello (1.7) riesce a catturare globalmente più memoria degli altri, con valori massimi assoluti in ACF e PACF dei residui (***Figura 11***) di circa 0.06, ma comunque mantenendo povere abilità di generalizzazione sul validation set.

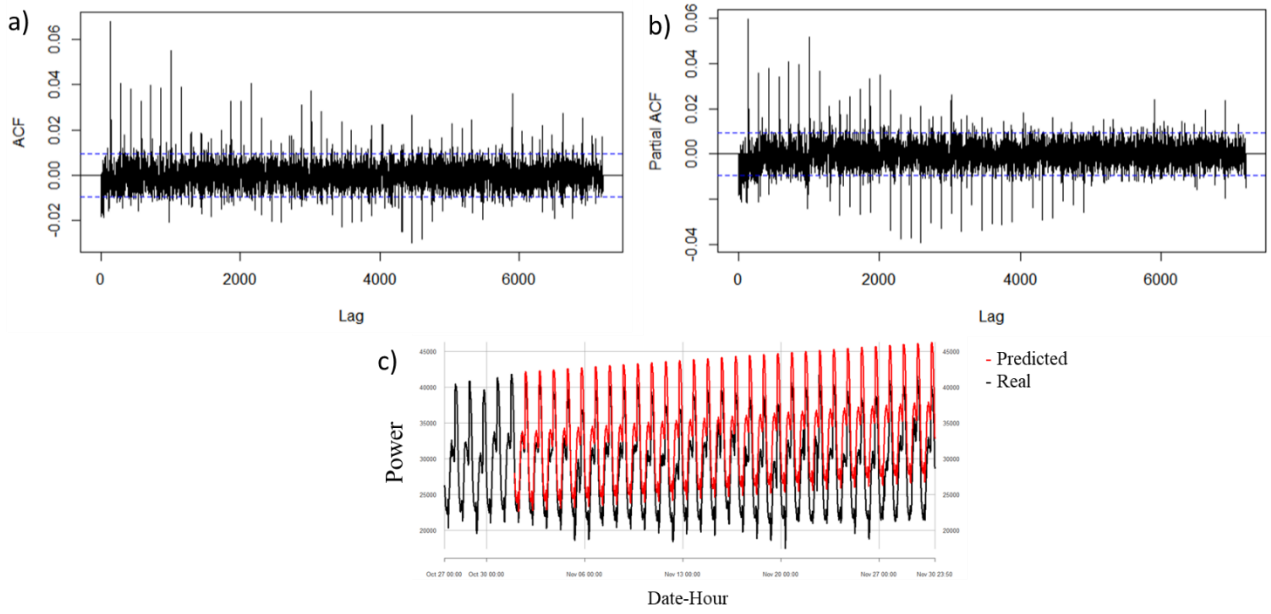

Figura 11: Autocorrelogramma dei residui del modello Arima (1.7): a) ACF, b) PACF e d) previsioni sul validation set.

3.3 Modelli ARIMAX

Provando a catturare la stagionalità settimanale sono stati valutati ulteriori modelli aggiungendo sei variabili ***dummy*** per modellare i sette giorni della settimana, utilizzate come regressori esterni, continuando a trattare la stagionalità giornaliera mediante differenza stagionale.

Anche in questo caso il modello trovato che meglio cattura la memoria della serie storica, ovvero un:

$$ARIMA(2,1,0)(0,1,1)_{144} + 6\ dummy\ settimanali \quad (1.10)$$

ottenuto testato automaticamente diverse configurazioni dei parametri ARIMA, usando la funzione *auto.arima* con criterio di selezione AIC e valori massimi $(3,2,1)(1,1,1)$, risulta in basse capacità predittive, paragonabili a quelle del modello (1.7), con $MAE_{training} = 220.06$ e $MAE_{validation} = 4116.936$. Sono quindi state testate configurazioni più semplici dei parametri ARIMA, ottenendo le performance migliori in validazione con il

modello (1.11) privo di differenze stagionali, che cattura poca memoria del processo ma generalizza bene su novembre ($MAE_{training} = 1220.55$ e $MAE_{validation} = 1065.45$).

$$ARIMA(0,0,0)(1,0,0)_{144} + 6\ dummy\ settimanali \quad (1.11)$$

Successivamente si è provato a modellare la stagionalità settimanale usando dei regressori alternativi alle variabili dummy, introducendo **8 sinusoidi settimanali** (un numero superiore non portava a aumenti di performance significativi) di periodo 1008 come regressori esterni, ottenendo $\boldsymbol{MAE_{validation} = 1010.08}$ e $MAE_{training} = 1082.68$, e come visibile in *Figura 12* questo modello non cattura molta della memoria a breve termine e stagionale del processo.

$$ARIMA(0,0,0)(1,0,0)_{144} + 8\ sinusoidi\ settimanali\ (p = 1008) \quad (1.12)$$

Sono infine stati effettuati due ultimi tentativi: nel modello 1.13 ($MAE_{validation} = 1186.09$) sono introdotte 8 sinusoidi per modellare la stagionalità giornaliera (p = 144) mentre nel modello 1.14 ($MAE_{validation} = 1075.07$) sono utilizzate 8 sinusoidi giornaliere e 6 dummy settimanali. Il modello 1.12 resta il migliore in termini di performance in validazione.

$$ARIMA(0,0,0)(1,0,0)_{144} + 8\ sinusoidi\ giornaliere\ (p = 144) \quad (1.13)$$

$$ARIMA(0,0,0)(1,0,0)_{144} + 8\ sinusoidi\ giornaliere\ (p = 144) + 6\ dummy\ settimanali \quad (1.14)$$

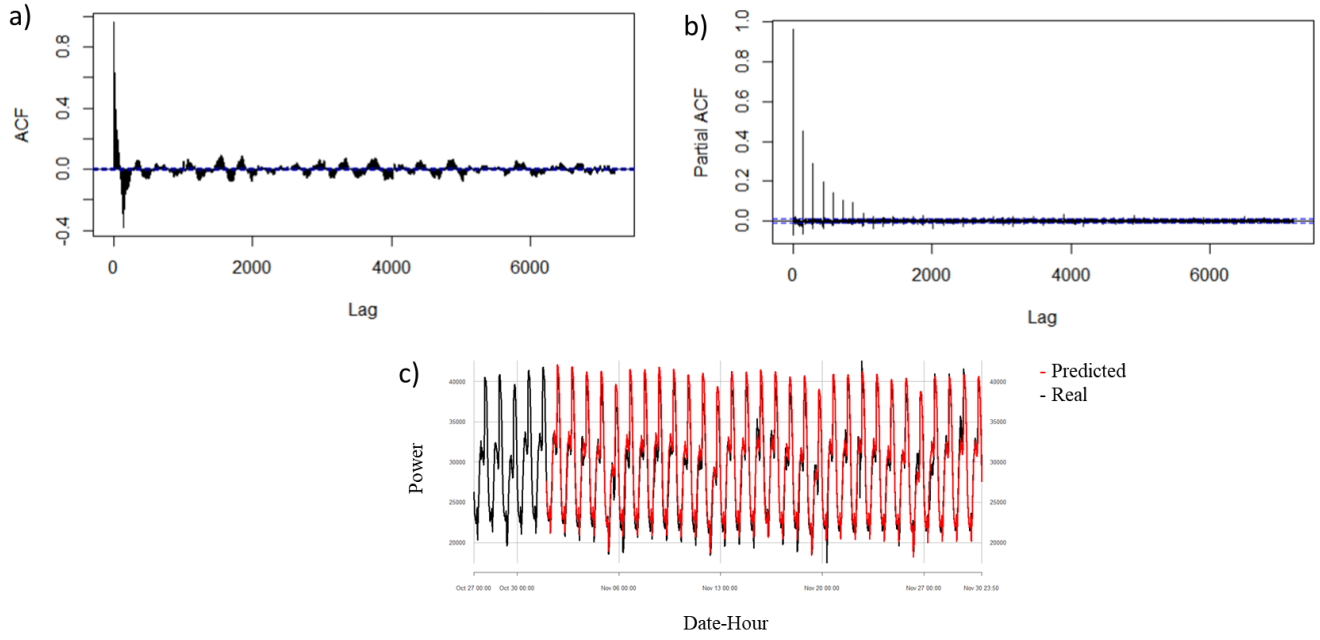

Figura 12: Autocorrelogramma dei residui del modello Arima (1.12): a) ACF, b) PACF e d) previsioni sul validation set.

3.4 Modelli ARIMA su serie storica aggregata a livello orario

Quest'ultimo tentativo di *modeling* ARIMA ha visto la creazione di 24 serie storiche indipendenti associate alle 24 ore della giornata, con lo scopo di eliminare la stagionalità giornaliera e migliorare l'efficienza di stima, raggruppando le osservazioni per giorno e per ora, mediando i valori all'interno dei gruppi e suddividendo le osservazioni in funzione dell'ora associata.

La ricerca del modello migliore è stata effettuata in modo iterativo, modellando una delle 24 serie storiche, e modificando i parametri fino a ottenere un modello che riuscisse a catturare buona parte della memoria della serie storica (osservando ACF e PACF dei residui) e al contempo massimizzasse le performance in fase di validazione. La valutazione delle performance è avvenuta fittando il medesimo modello su ognuna delle 24 serie storiche, effettuando le previsioni, concatenandole al fine di ottenere un'unica serie storica oraria e ripristinando un campionamento ogni 10 minuti mediante interpolazione cubica spline, usando la funzione *na.spline* di *zoo* (è testata anche l'approssimazione lineare con risultati peggiori). In questo contesto il miglior modello ottenuto è:

$$ARIMA(0,1,1)(0,0,1)_7 \qquad (1.15)$$

che garantisce un $MAE_{validation} = 1142.87$ e un autocorrelogramma dei residui vicino all'ipotesi di white noise. (*Figura 13*)

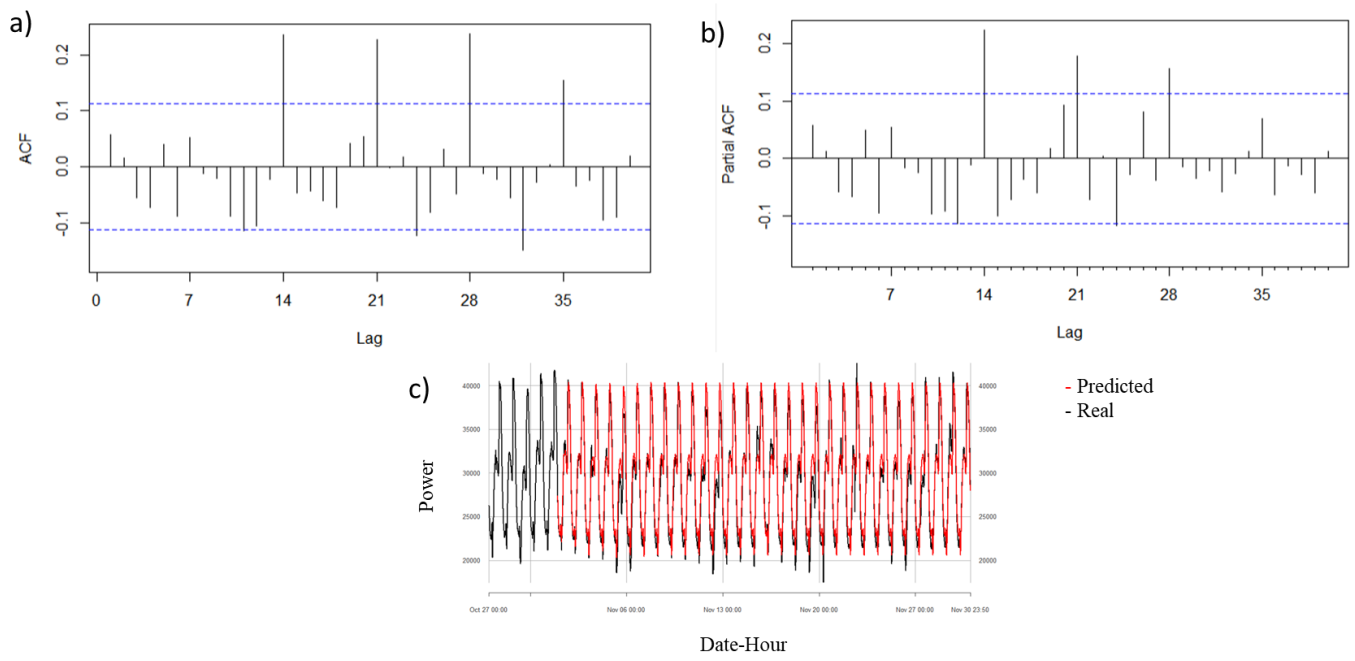

Figura 13: Autocorrelogramma dei residui del modello Arima 1.15 ore 12:00 a) ACF, b) PACF e d) previsioni complete sul validation set.

3.4 ARIMA: Miglior modello

Il miglior modello ARIMA, in termini di performance in fase di valutazione sul mese di novembre, risulta essere il modello 1.12, con un $MAE_{validation} = 1010.08$:

$$ARIMA(0,0,0)(1,0,0)_{144} + 8\ sinusoidi\ settimanali\ (p = 1008) \quad (1.12)$$

4. Modelli UCM

Una seconda classe di modelli testati è stata quella degli Unobserved Components Models (UCMs), che non necessitano di gestire una eventuale non stazionarietà del processo e si basano sull'idea che una serie storica sia modellabile come la combinazione di componenti non osservabili, come eventuali tendenze, cicli e stagionalità.

4.1 UCM su serie storica aggregata a livello orario

Inizialmente i modelli sono stati addestrati sulla serie storica aggregata a livello orario, ottenendo le osservazioni relative a ogni ora come media delle 6 osservazioni campionate ogni 10 minuti, al fine di ridurre gli elevati tempi di computazione. Una volta ottenute le 720 previsioni orarie per il mese di novembre, è stato rigenerato il campionamento ogni 10 minuti, caratteristico della serie storica originaria, mediante interpolazione cubica spline.

Nel caso specifico la serie storica la serie storica è stata modellata come combinazione di un *trend*, una stagionalità giornaliera (ogni 24 ore) e una settimanale (ogni 168 ore), ottenendo un modello:

$$\hat{y} = TREND + STAGIONALITÀ_{giornaliera}(24\ ore) + STAGIONALITÀ_{settimanale}(168\ ore)$$

Anche in questo caso la configurazione ottima per il modello è stata trovata mediante un processo iterativo, cercando di massimizzare le performance in previsione (in termini di MAE), dei modelli testati, sul validation set composto dalle osservazioni di novembre 2017.

Inizialmente si è cercata la migliore modalità per modellare la stagionalità giornaliera (periodo = 24 ore), testando un modello composto da un Local Linear Trend (SSMtrend di ordine 2) e una stagionalità settimanale (periodo = 168 ore) modellata come stagionalità a sinusoidi stocastiche (usando una sinusoide a 10 armoniche). Sono quindi verificate le performance in validazione di due variazioni dello stesso, usando 24 dummy stocastiche e sinusoidi stocastiche per modellare la stagionalità giornaliera (testando diversi valori per il numero di sinusoidi da utilizzare): i modelli che generalizzano meglio risultano essere quelli con stagionalità giornaliera modellata con dummy stocastiche.

Si è inoltre provato a utilizzare un ciclo stocastico come alternativa alle sinusoidi stocastiche (con periodo di 168 osservazioni) per modellare la stagionalità settimanale, ottenendo però $MAE_{validation}$ molto elevati.

Si è infine valutato il numero ottimo di sinusoidi da utilizzare per modellare la stagionalità settimanale, testando iterativamente da 1 a 16 sinusoidi, e confrontando le performance sul validation set, usando modelli le cui altre componenti sono Local Linear Trend e 24 dummy stocastiche. Le migliori performance di previsioni di ottengono rispettivamente con 2 ($\boldsymbol{MAE_{validation} = 1210.08}$) e con 6 sinusoidi stocastiche ($\boldsymbol{MAE_{validation} = 1299.41}$).

3.4 UCM su serie storica originale campionata ogni 10 minuti

Cercando di migliorare ulteriormente le performance sul validation set dei modelli UCM, si è modellata la serie storica originale campionata ogni 10 minuti, usando come dati di training le osservazioni di settembre e ottobre 2017 (perché la stima sulla serie storica completa risultava troppo onerosa) e valutando i modelli prevedendo le 4320 osservazioni di novembre 2017, per poi addestrare il modello finale sulle osservazioni di Settembre, Ottobre e Novembre 2017, prevedendo quelle di Dicembre 2017. La scelta del modello è stata effettuata tramite un processo iterativo del tutto analogo a quello effettuato sulla serie storica oraria, ottenendo buone performance in validazione con un modello composto da:

- Trend modellato come Local Linear Trend
- Stagionalità settimanale (p = 1008) modellata con 10 sinusoidi stocastiche
- Stagionalità giornaliera (p = 144) modellata con 10 sinusoidi stocastiche.

Inizializzando (dopo vari tentativi) le varianze del modello (data vy come varianza della serie storica) come:

- $logVarEta$: $vy/10$ (varianza livello)
- $logVarZeta$: $vy/10000$ (varianza slope)
- $logVarOmega144$: $vy/1000$ (varianza stagionalità giornaliera)
- $logVarOmega1008$: $vy/10000$ (varianza stagionalità settimanale)
- $logVarEpsilon$: $vy/10$ (varianza white noise)

E impostando:

- $a_1 = mean(serie\ storica)$
- $P_1 = vy * 10$

Il modello così configurato risulta in un $\boldsymbol{MAE_{validation} = 1323.07}$, ma quando applicato al logaritmo della serie storica (invertito in fase di previsione), raggiunge un $\boldsymbol{MAE_{validation} = 1183.40}$, risultando il miglior modello testato per la classe dei modelli UCM.

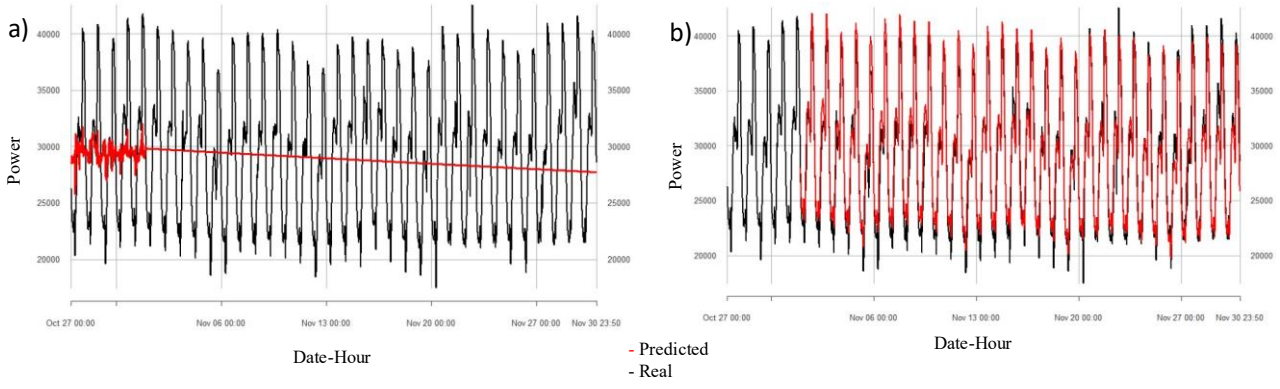

Figura 14: a) stima del trend e b) previsioni sul validation set per il miglior modello UCM ottenuto.

5. Modelli di Machine Learning (k-NN e RF) e Deep Learning (GRU RNN)

L'ultima classe di modelli testati è stata quella dei modelli di machine learning, che presentano il pregio, rispetto alle classi ARIMA e UCM, di non necessitare la definizione a priori di eventuali stagionalità nei dati (nel caso specifico stagionalità settimanale e giornaliera), riuscendo ad apprenderla automaticamente in fase di training. In particolare, sono stati addestrati due diversi modelli di Machine Learning (Random Forest e k-NN) e un modello di Deep Learning (GRU RNN) direttamente sulla serie storica originale con campionamento ogni 10 minuti.

5.1 GRU Recurrent Neural Network

Si è testata l'implementazione di reti neurali ricorrenti GRU (Gated Recurrent Units), variando iterativamente gli iperparametri dell'architettura (in funzione delle performance in termini di MAE in validazione) che includono: numero di layers GRU presenti nell'architettura e relativo numero di unità neuronali, presenza o meno di dropout dopo i layer GRU, batch size, numero di epoche, learning rate, valori di lag e lead, funzione di attivazione e dimensioni di eventuali layer densi di output.

La rete è stata addestrata sulla serie storica originale campionata ogni 10 minuti, standardizzata rispetto alla media (ottenendo valori con media 0 e deviazione standard 1) e differenziata con ordine di 144 osservazioni (invertendo poi le due trasformazioni sulle previsioni) utilizzando come regressori i soli ritardi della serie storica stessa. In particolare, le modalità con cui sono strutturati i dati in input alla rete e l'architettura della stessa, fanno si che la rete impari a prevedere le 30 osservazioni di un mese, associate a uno specifico minuto della giornata, date le 30 osservazioni per quel minuto nel mese precedente.

I parametri della configurazione finale sono stati:

- **Batch Size** = 144, **Lag** = 4320 (numero di osservazioni nel passato usate come regressori, 1 mese di dati), **Lead** = 4320 (orizzonte di previsione di 30 giorni, numero di osservazioni nel futuro da prevedere, si usa approccio multi-step)
- **Numero di epoche:** 40, **Learning Rate:** 0.001, **Ottimizzatore**: Adam, **Loss:** MAE

L'architettura del modello migliore è composta da:

1. GRU Layer (16 unità, dropout = 0, attivazione = tangente iperbolica)
2. GRU Layer (16 unità, dropout = 0, attivazione = tangente iperbolica)
3. Layer Denso/Fully Connected (30 neuroni, attivazione = lineare)

Il modello porta a buone performance di previsione in validazione, le cui previsioni (in ***Figura 15***) risultano in un $MAE_{validation} = 1184.07$.

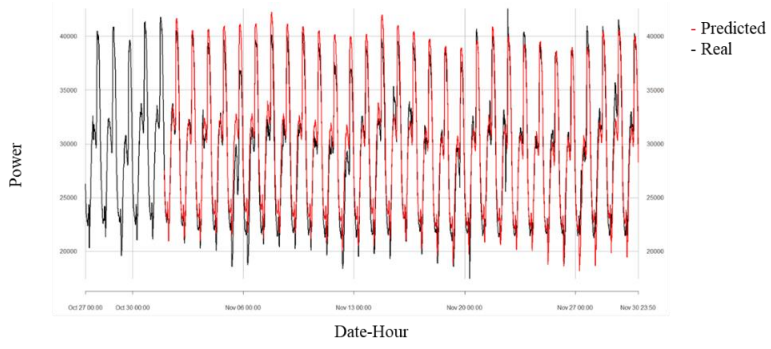

Figura 15: previsioni sul validation set (novembre 2017) per il modello GRU.

5.2 Random Forest e k-NN

Sono infine stati testati due modelli di machine learning, in particolare Random Forest e k-NN, applicandoli direttamente sulla serie storica originale con campionamento ogni 10 minuti, utilizzando come regressori i ritardi delle osservazioni.

Per quanto riguarda l'approccio Random Forest, sono state testate iterativamente diverse configurazioni, variando il numero di lag usati come regressori, il numero di alberi e introducendo regressori ausiliari (dummies relative all'orario e al giorno della settimana), ma i risultati in previsione (effettuata con approccio ricorsivo alla stima, introducendo le previsioni come nuovi regressori) non sono risultati particolarmente soddisfacenti. Il modello più performante, che utilizza 144*7 lag, dummy relative al giorno della settimana come osservazioni e 250 alberi, ottiene un $MAE_{validation} = 2350.14$.

L'implementazione della k-NN regression, usando 144*7 lag (una settimana) come osservazioni, un approccio alla previsione multi-step di tipo MIMO (multiple-input and multiple-output) e la mediana come funzione di combinazione per aggregare i target associati ai *nearest neighbors*, permette di ottenere delle performance in validazione superiori a Random Forest, risultando in un $MAE_{validation} = 1694.75$.

6. Conclusioni

L'analisi di serie storiche di consumi energetici ad alta frequenza permette una migliore comprensione delle tendenze e le dinamiche che caratterizzano i consumi energetici in un determinato contesto geografico, e la possibilità di catturare e modellare tali pattern di consumo passati per fornire previsioni affidabili dei consumi futuri risulta uno strumento strategico essenziale per le aziende energetiche.

Nonostante ciò, l'elevata frequenza e la persistenza della serie storica analizzata, così come gli innumerevoli fattori che influenzano i consumi energetici, rendono complessa la modellazione e quindi l'ottenimento di previsioni particolarmente accurate. I risultati dei migliori modelli di ogni famiglia, in termini di Mean Absolute Error (MAE) su un validation set composto dalle osservazioni di novembre 2017, sono riportati in ***Tabella 1*** e le relative previsioni sono mostrate in ***Figura 16***. Le performance dei modelli migliori per ogni tipologia sono paragonabili, ma il modello ARIMA risulta essere il più accurato. Possibili miglioramenti nelle performance potrebbero essere ottenuti: avendo a disposizione una serie storica più lunga (soprattutto per quanto riguarda i modelli di Machine Learning, che difficilmente riescono ad imparare fenomeni unici, come le festività, avendo a disposizione solo un anno di dati); integrando dati esterni che descrivano fenomeni potenzialmente legati ai consumi energetici (come le condizioni atmosferiche) oppure ancora combinando le previsioni di modelli addestrati sulla serie storica aggregata e non aggregata (cercando di combinare la robustezza dei primi e il maggiore dettaglio dei secondi) e modelli appartenenti a classi diverse (che riescono ad apprendere dai dati informazioni diverse).

Tabella 1: Risultati dei migliori modelli per ogni famiglia di modelli, in termini di MAE sul validation set.

Famiglia di Modelli	**Modello migliore**	**$MAE_{validation}$**
ARIMA	ARIMA(0,0,0) $(1,0,0)_{144}$+ 8 sinusoidi settimanali (p=1008)	1010.08
UCM	LLT + 10 sinusoidi stocastiche (giornaliere, p = 144) + 10 sinusoidi stocastiche (settimanali, p = 1008)	1183.40
ML	Gated Recurrent Units (GRU) Recurrent Neural Network	1184.07

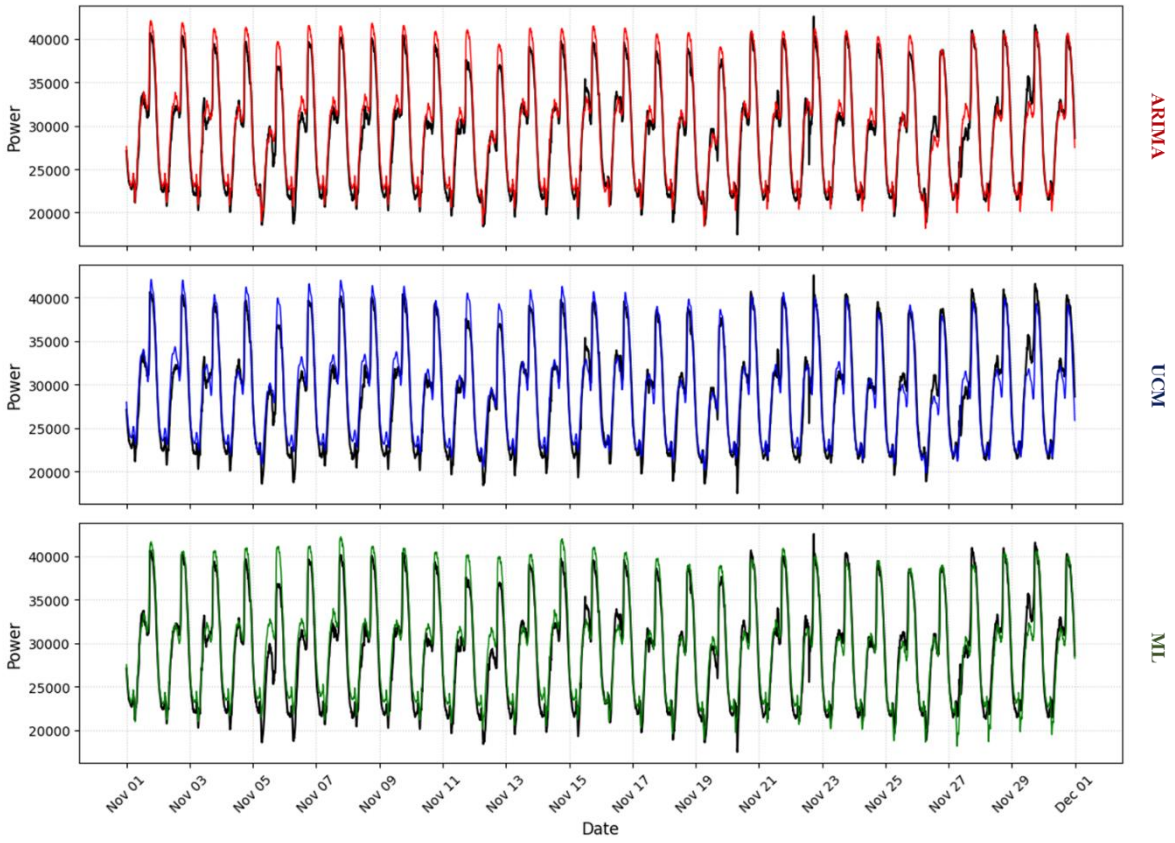

Figura 16: Previsioni per il validation set (novembre 2017) effettuate con i migliori modelli per ogni classe.

Sono di seguito riportate le previsioni di dicembre 2017 effettuate riaddestrando i migliori modelli per ogni categoria sulla combinazione del rispettivo training e validation (novembre 2017) set, prevedendo le 4320 osservazioni di dicembre 2017

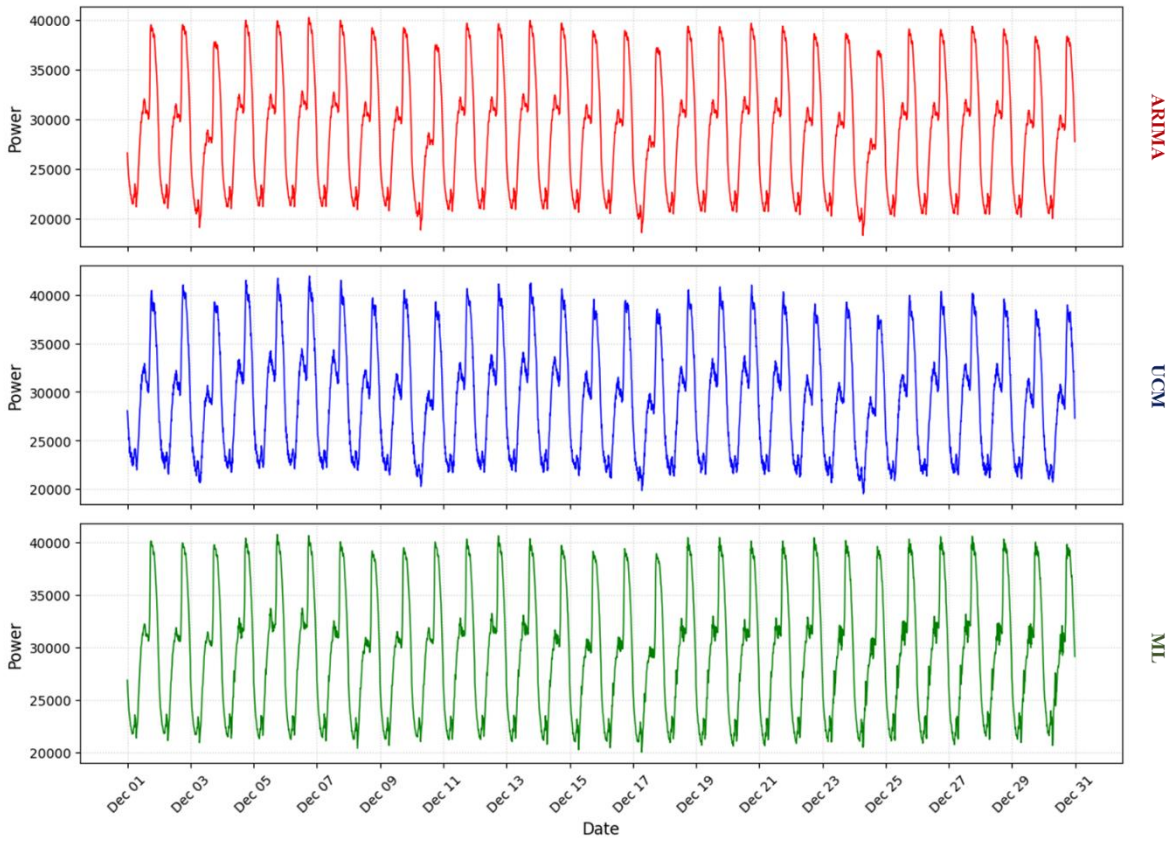

Figura 17: Previsioni per il mese di dicembre 2017 effettuate con i migliori modelli di ogni classe.