PREVISIONE SERIE STORICA DI MONOSSIDO DI CARBONIO

PROGETTO DEL CORSO STREAMING DATA MANAGEMENT AND TIME SERIES ANALYSIS

Lorgna Lorenzo
Corso di Laurea Magistrale in Data Science
Università degli Studi di Milano-Bicocca
l.lorgna@campus.unimib.it - 829776

Giugno 2022

ABSTRACT

In questo elaborato vengono presentati una serie di modelli appartenenti a tre grandi famiglie, ARIMA, UCM e machine learning. L'obiettivo primario è quello di affrontare un task di previsione di serie temporale riguardante l'andamento dei valori di monossido di carbonio. I modelli verranno valutati e confrontati sulla base della metrica MAPE, tramite anche l'ausilio di grafici.

Keywords Time series · Preprocessing · Data exploration · Time series forecasting · ARIMA · UCM · machine learning · KNN · LSTM

Contents

1 Introduzione

Il progetto si pone come obiettivo l'implementazione di diversi algoritmi con lo scopo di effettuare una previsione di valori per quanto concerne una serie storica rappresentante le rilevazioni di monossido di carbonio (CO). Per perseguire il seguente obiettivo sono stati considerati differenti algoritmi appartenenti a tre categorie diverse: ARIMA, UCM e machine learning. Al fine di individuare i modelli dalle migliori performance è stata utilizzata come metrica il Mean Absolute Percentage Error (MAPE), che indica l'errore percentuale assoluto medio. Tra i vantaggi offerti dall'utilizzo di tale metrica vi è la sua facile comprensione anche per persone non esperte del dominio.

2 Dataset

Il dataset fornito rappresenta una serie storica univariata relativa a misurazioni orarie di CO. In particolare i dati sono organizzati nelle seguenti 3 colonne:

- *Date*: stringa codificante la data della misurazione, in formato yyyy-mm-dd
- *Hour*: intero indicante l'ora della misurazione. I valori vanno da 0 a 23, con 0 che rappresenta l'intervallo 00:00 - 00:59, 1 che rappresenta l'intervallo 01:00 - 01:59, ... e 23 che rappresenta l'intervallo 23:00 - 23:59
- *CO*: valore di CO rilevato

Le misurazioni sono relative al periodo compreso tra il 10 marzo 2004 (00:00) e il 28 febbraio 2005 (23:00). L'intento è quello di prevedere i valori di CO per il mese di marzo 2005, dunque dal 1 marzo 2005 (00:00) al 31 marzo 2005 (23:00).

Il numero totale di osservazioni a disposizione è pari a 8526 mentre i valori che devono essere previsti sono 744.

2.1 Preprocessing

Prima di procedere con le analisi è stato necessario applicare alcune tecniche di preprocessing ed esplorare il dataset per avere una maggiore comprensione del dato.

Inizialmente è stata verificata la presenza o meno di record duplicati evidenziandone tuttavia l'assenza.

A seguire è stata creata una feature con l'obiettivo di unire le informazioni temporali a disposizione, *Date* e *Hour*. Inoltre a partire da quest'utlima feature appena creata tramite una funzione apposita sono state generate altre fetaure potenzialmente utili, in particolar modo nella fase di esplorazione, quali ad esempio: *dayofweek*, *month*, *year* e *dayofmonth*.

Si è proceduto con l'analisi dei valori mancanti all'interno del dataset: sono stati rilevati 365 valori mancanti per la feature *CO*. Per cercare di risolvere la seguente problematica, non avendo informazioni in merito a come trattare tali valori e non notando una specifica regolarità di essi, si è inizialmente optato per una interpolazione lineare. Visti, tramite anche l'ausilio di grafici, i risultati scadenti si è deciso di procedere con un approccio differente e dunque sostituire i valori mancanti con una media dei due valori precedenti e i due valori successivi, presi alla medesima ora e giorno della settimana. In questo caso il risultato è stato migliore.

Nella fase finale di preprocessing il dataset è stato diviso in train e test, come mostrato in figura 14 in Appendice A, suddivisione necessaria per le fasi successive di modellazione al fine di avere delle metriche con le quali giudicare la bontà dei modelli creati. Si è deciso di considerare come test set l'ultimo mese a disposizione (7.9% delle osservazioni), dunque il mese di febbraio e di prendere tutti i dati precedenti per il training del modello (92.1% delle osservazioni).

2.2 Esplorazione

In una prima fase di esplorazione è stata analizzata in termini generali la serie storica a disposizione, come è possibile osservare in figura 1. Osservando la serie storica prendendo in considerazione diversi tipi di granularità (giornaliera, settimanale, mensile), come mostrato in figura 15 in Appendice A, risulta difficile individuare un trend evidente.

A seguire aggregando i dati e utilizzando dei boxplot si è analizzata nel dettaglio la distribuzione della variabile di interesse CO. L'utilizzo dei boxplot permette di analizzare facilmente alcune importanti statistiche come la

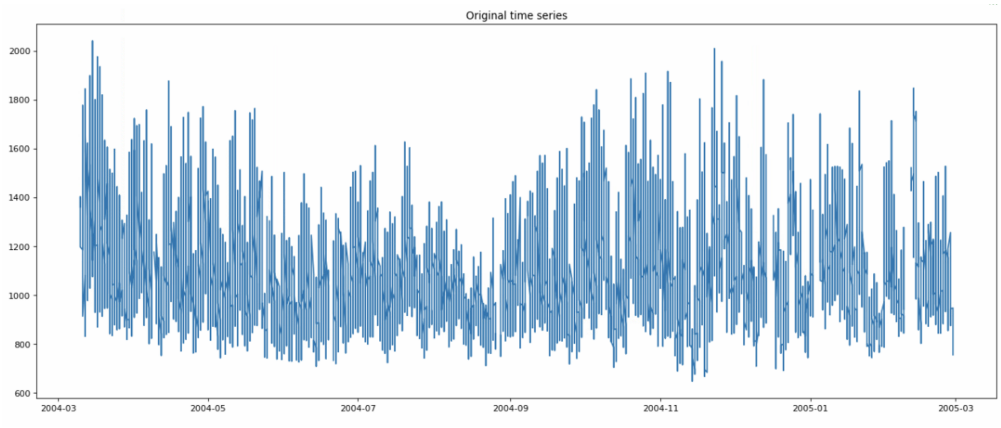

Figure 1: Serie storica originale a livello orario

media, la mediana e la presenza di eventuali outliers.

Come si può osservare in figura 2 a livello annuale la media risulta essere leggermente più alta per il l'anno 2005 mentre considerando la distribuzione mensile è possibile notare che i valori più alti di misurazione di CO si verificano nei mesi di marzo e ottobre mentre i valori minori corrispondono al mese di agosto.

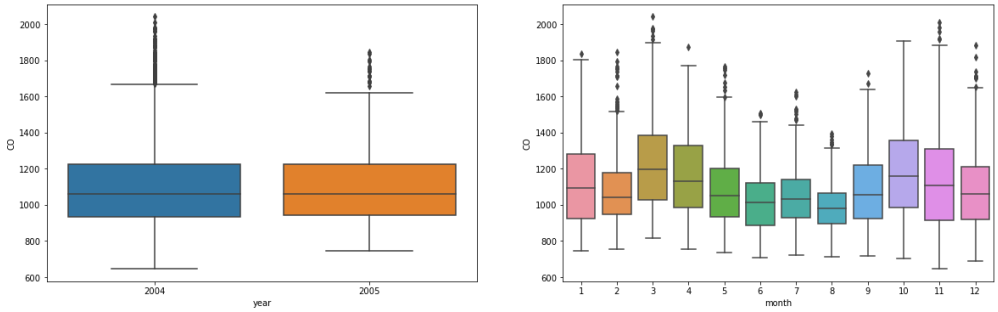

Figure 2: Box plot a livello annuale e mensile

Osservando la figura 3 a livello settimanale i giorni per cui si verifica un maggiore aumento di CO sono i giorni centrali della settimana mentre sembra emergere una diminuzione all'inizio e alla fine della settimana. Questo potrebbe indicare la presenza di una stagionalità settimanale, ovvero la presenza di osservazioni che si ripetono con pattern simili ogni 168 ore. A livello giornaliero, invece, si osservano valori più elevati ad inizio mattinata ed inizio serata, con valori inferiori nelle ore notturne. In questo caso la stagionalità giornaliera, ovvero la presenza di osservazioni simili che si ripetono ogni 24 ore, risulta essere molto più evidente.

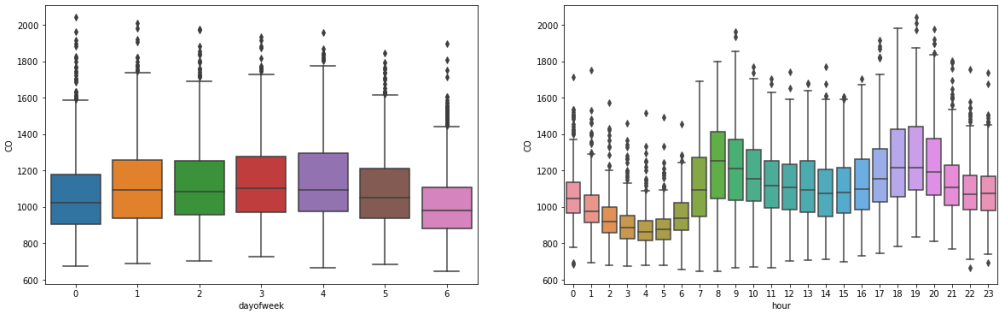

Figure 3: Box plot a livello settimanale e orario

3 Modelli ARIMA

Dopo le fasi precedenti di preprocessing ed esplorazione si è proceduto con la definizione dei modelli per essere in grado di prevedere i valori di CO per il mese di marzo 2005. Per tutti i modelli, come indicato precedentemente, come dati di training sono state considerate tutte le osservazioni comprese dal 10 marzo 2004 al 31 gennaio 2005 mentre come test set è stato utilizzato l'intero mese di febbraio.

Il primo modello ad essere considerato rientra nella famiglia dei modelli ARIMA. ARIMA sta per Auto-Regressive Integrated Moving Average e risulta essere uno degli strumenti più diffusi per effettuare previsioni di serie storiche.

Per l'implementazione dei vari modelli che sono stati testati è stata utilizzata la procedura proposta da Box e Jenkins, una procedura di tipo iterativo che si compone di alcune fasi fondamentali: verifica della stazionarietà della serie, identificazione del modello ARIMA, stima dei parametri e verifica del modello tramite analisi dei residui.

La stazionarietà della serie risulta essere un presupposto fondamentale in molte procedure statistiche utilizzate nell'analisi delle serie storiche tra cui appunto anche ARIMA. Per tale motivo è necessario adoperare alcune trasformazioni nel caso in cui questa condizione non venisse soddisfatta. Osservando i grafici precedenti è possibile ipotizzare che la serie in questione risulta essere non stazionaria.
Si procede nello specifico nell'analisi della non stazionarietà in varianza e della non stazionarietà in media. Partendo dalla prima, osservando la figura 4, si nota una chiara relazione lineare tra il livello e la deviazione standard. In questi casi la trasformazione consigliata è il logaritmo da applicare sui dati a nostra disposizione. Applicando il logaritmo e osservando la nuova serie storica sembra che la condizione di stazionarietà in varianza sia migliorata.

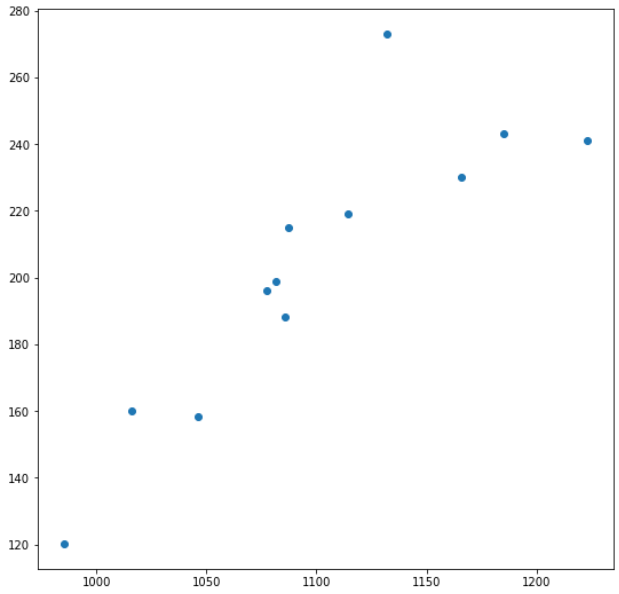

Figure 4: Relazione lineare livello e deviazione standard

A seguire, considerando la condizione di stazionarietà in media, viene eseguito il Dickey-Fuller test, un test la cui ipotesi nulla è che ci sia una radice unitaria. Analizzando i risultati ottenuti dall'esecuzione del test è possibile giungere alla conclusione che l'ipotesi nulla non è possibile rifiutarla. In questo caso per cercare di risolvere la seguente problematica si procede con una differenziazione stagionale, di ordine 24. In questo caso analizzando nuovamente la serie storica su cui è stata applicata la differenziazione è possibile notare che tale operazione ha ridotto parzialmente l'effetto della stagionalità. Questo viene confermato dal grafico 16 in Appendice A, dove si può notare il diverso comportamento della stagionalità prima e dopo le due operazioni messe in atto, applicazione del logaritmo e differenziazione stagionale. Per conferma, viene nuovamente eseguito il Dickey-Fuller test sui dati differenziati che in questo caso porta a rifiutare l'ipotesi nulla, indicando dunque che la serie non ha radici unitarie. Tale conclusione porta ad ipotizzare che la serie ora soddisfi anche la condizione di stazionarietà in media.

Per l'implementazione dei modelli si è proceduto gradualmente partendo da modelli più semplici per ar-

rivare poi a modelli più complicati, in grado di modellare aspetti più complessi. Sono stati fatti diversi tentativi facendo variare i parametri del modello ARIMA:

- p: il numero di osservazioni di lag nel modello, noto anche come ordine di ritardo
- i: il numero di volte in cui le osservazioni grezze vengono differenziate, noto anche come grado di differenziazione
- q: la dimensione della finestra della media mobile, noto anche come ordine della media mobile

Ad essere considerati anche i parametri P, Q, I, analoghi a quelli precedenti ma a livello stagionale. I vari modelli ARIMA sono stati confrontati tra loro prendendo in considerazione la metrica Akaike Information Critera (AIC) oltre al MAPE e osservando il comportamento dei residui tramite l'analisi dei correlogrammi ACF (AutoCorrelation Function) e PACF (Partial AutoCorrelation Function). Si tratta di grafici che permettono di studiare l'autocorrelazione e l'autocorrelazione parziale tra osservazioni.

Si è partiti con un primo modello base che includesse le trasformazioni considerate precedentemente, dunque logaritmo e differenziazione: *ARIMA (0, 1, 0)(0, 1, 0)[24]*.
Trattandosi di un modello molto semplice, osservando i grafici ACF e PACF in figura 5, si nota un'evidente stagionalità ogni 24 osservazioni, ovvero ogni giorno.

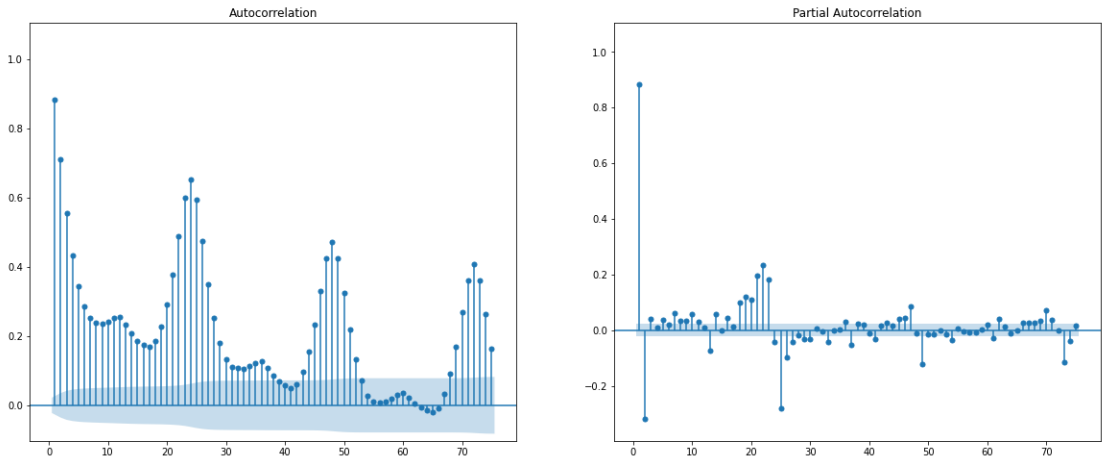

Figure 5: ACF/PACF *ARIMA (0, 1, 0)(0, 1, 0)[24]*

Come è solito fare dopo aver fatto eseguito una differenza stagionale si opta per un MA stagionale: *ARIMA (0, 1, 0)(0, 1, 1)[24]*. Osservando i grafici ACF e PACF in figura 6 i risultati sono nettamente migliorati, tuttavia rimangono alcuni lag ancora significativi.

Visti i lag significativi, in particolar modo a livello stagionale, si prova con un *ARIMA (0, 1, 0)(1, 1, 1)[24]*. Osservando i risultati si nota un leggero miglioramento.

Per modellare al meglio i parametri p e q è stata effettuata una grid search e tenendo in considerazione anche il miglioramento in termini di AIC si è optato per i valori 2 e 2, per p e q. Dunque si è ottenuto il seguente modello: *ARIMA (2, 1, 2)(1, 1, 1)[24]*.

Non avendo ottenuto ancora dei risultati soddisfacenti si è deciso di considerare dei regressori esterni, in particolare delle sinusoidi per cercare di catturare la stagionalità settimanale, osservata anche nella prima fase di esplorazione della serie storica. Dopo alcuni test il numero ottimale di sinusoidi a periodo 168 per modellare questo tipo di stagionalità è stato 3. Anche in questo caso il modello ha registrato un leggero miglioramento in termini di AIC.

Viene riportata di seguito la tabella 1 con i modelli ARIMA implementati durante la fase di modellazione e i relativi valori di performance misurati sul test set.

In definitiva il modello migliore ottenuto è: *ARIMA (2, 1, 2)(1, 1, 1)[24]* con regressori esterni per model-

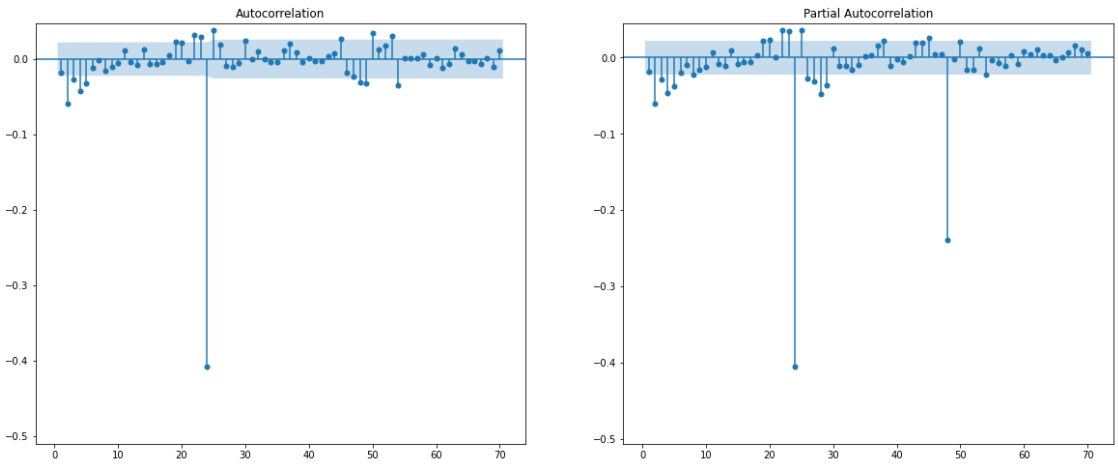

Figure 6: ACF/PACF *ARIMA (0, 1, 0)(0, 1, 1)[24]*

Table 1: Modelli ARIMA

Modello	**AIC**
ARIMA (0, 1, 0)(0, 1, 0)[24]	-15544
ARIMA (0, 1, 0)(0, 1, 1)[24]	-19685
ARIMA (0, 1, 0)(1, 1, 1)[24]	-19742
ARIMA (2, 1, 2)(1, 1, 1)[24]	-20260
ARIMA (2, 1, 2)(1, 1, 1)[24] con regressori	-20297

lare la stagionalità settimanale, con valori di AIC e MAPE rispettivamente di -20297 e 10.62. La figura 7 riporta graficamente come il modello si è adattato ai dati di testing.

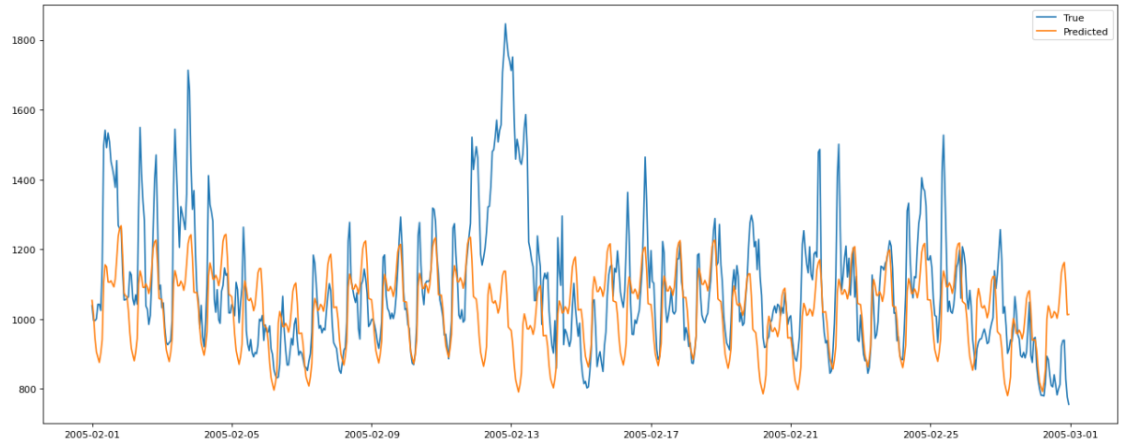

Figure 7: Previsioni sul test set ARIMA

4 Modelli UCM

L'altra famiglia di modelli ad essere considerata è stata quella dei modelli a componenti non osservabili (UCM). Questi modelli, detti anche Modelli Strutturali per Serie Storiche, sviluppati principalmente da Harvey (1989), si basano sull'idea di pensare una serie storica come la somma di componenti stocastiche, oltre ad eventuali regressori.

Anche in questo caso sono stati testati differenti modelli al fine di ottenere quello che meglio si adatta ai dati. Sono stati dunque considerati i parametri level e seasonal. Il parametro cycle non è stato considerato dal momento

in cui le fluttuazioni cicliche sono solitamente della durata di almeno 2 anni. Considerata la limitata dimensione dei dati a disposizione si è preferito non considerare tale parametro.

Si è considerato inizialmente il parametro level, paramento che modella la componente di trend. Considerando la stagionalità giornaliera, il tipo di stagionalità più evidente all'interno dei dati a disposizione, sono stati testati diversi tipi di modelli facendo variare il parametro level tra i seguenti valori: No trend, Deterministic constant, Local level, Random walk, Deterministic trend, Local linear deterministic trend, Random walk with drift, Local linear trend, Smooth trend. Prendendo in considerazione come metrica il MAPE e secondariamente l'AIC il tipo di level che ha permesso di ottenere i risultati migliori è stato rwdrift, ovvero un random walk con drift. È possibile osservare i risultati ottenuti nella seguente tabella 2.

Table 2: Modellazione componente trend modelli UCM

Modello	**MAPE (%)**	**AIC**
No trend	100.73	132294.86
Deterministic constant	12.16	104027.66
Local level	12.20	91661.99
Random walk	12.20	91569.99
Deterministic trend	11.95	104028.77
Local linear deterministic trend	12.04	91654.41
Random walk with drift	12.04	91652.40
Local Linear trend	12.04	91656.40
Smooth trend	2412.27	94506.95
Random trend	1487.71	96516.59

A seguire per quanto riguarda la componente stagionale, facendo diversi tentativi, si è deciso di optare per 3 sinusoidi di periodo 24 e 2 sinusoidi di periodo 168 per modellare rispettivamente la stagionalità giornaliera e quella settimanale.

Considerando dunque il modello finale, i migliori risultati si sono ottenuti con level pari a random walk con drift e la parte stagionale modellata da 3 sinusoidi di periodo 24 e 2 sinusoidi di periodo 168. Il modello addestrato ha registrato sul test set un valore di MAPE pari a 11.02 %. Il risultato dal punto di vista grafico dell'adattamento del modello sui dati di testing è osservabile in figura 8.

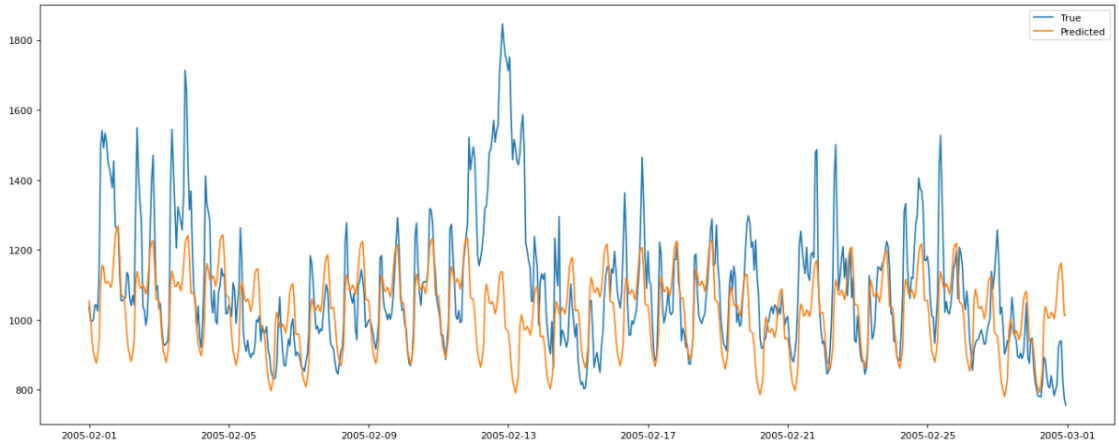

Figure 8: Previsioni sul test set UCM

5 Modelli Machine Learning

Per quanto riguarda i modelli rientranti nella famiglia del machine learning sono stati implementati due differenti modelli, k nearest neighbor (KNN) e Long Short-Term Memory (LSTM) network. Tra i due poi è stato preferito quello che ha registrato le migliori performance sul test set.

5.1 KNN

L'idea che sta alla base dell'algoritmo KNN in termini generali è che data un'istanza come input, il rispettivo output è predetto sulla base delle k istanze più simili presenti nel dataset a disposizione. Se si considera che il futuro è in qualche maniera collegato al passato, allora nel caso in cui si identificano delle finestre temporali che sono simili a quella corrente ma che sono avvenute nel passato si è nella condizione di conoscere come queste si sono evolute e quindi di ipotizzare come evolverà la sequenza in questione. Dunque se si individuano nel passato k sottosequenze che sono simili alla situazione attuale è lecito pensare e usare i loro futuri per stimare il futuro della finestra corrente presa in considerazione.

Sono state fatte diverse prove con l'obiettivo di identificare i valori migliori degli iperparametri. Gli iperparametri in questione sono: p, il numero di valori passati che vengono considerati e k, il numero di sottosequenze simili da prendere in considerazione. In questo caso è stata utilizzata l'implementazione multi-step multi-output (MIMO).

Nei test effettuati sono stati provati differenti valori per p (2,4,6, ... ,36,38,40) e k (24*7, 24*14, 24*21).

La combinazione ottimale dei parametri risulta essere 504 e 18, rispettivamente per gli iperparametri p e k. In questo caso il modello addestrato ha registrato sul test set un valore di MAPE pari a 10.44 %. L'adattamento del modello ai dati di test è visibile in figura 9. Per completezza nonostante poi il seguente modello non verrà selezionato tra i modelli finali migliori, vengono riportare in Appendice A le previsioni ottenute per il mese di marzo in figura 17.

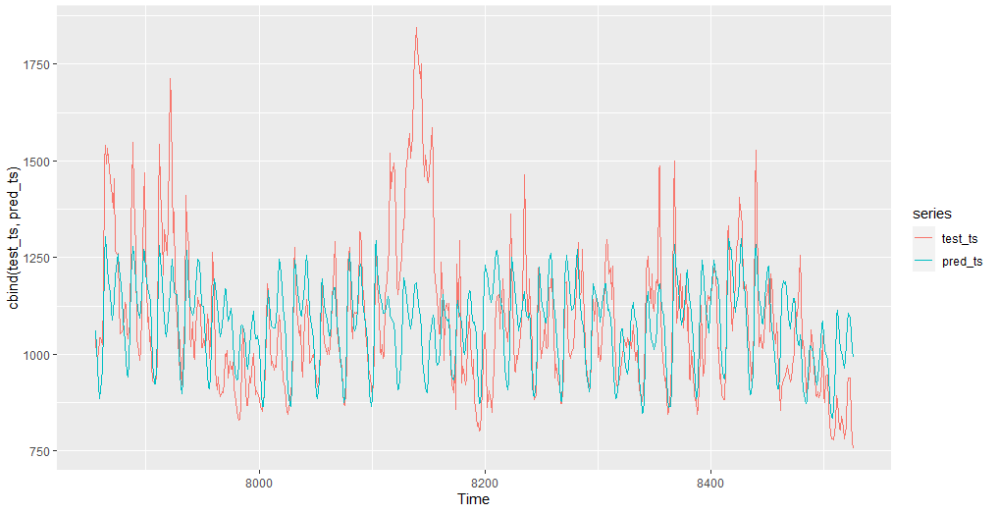

Figure 9: Previsioni sul test set KNN

5.2 LSTM

Le reti LSTM appartengono alla famiglia delle Recurrent Neural Networks (RNNs). Le LSTM insieme alle Gated Recurrent Units (GRU) appartengono alla famiglia più ampia delle Gated RNNs che risolvono il problema della scomparsa del gradiente, ovvero la dimenticanza di dipendenze a lungo termine.

Prima di applicare la rete neurale è stato necessario procedere con alcune operazioni di preprocessing. Nello specifico è stato inizialmente normalizzato il dataset (utilizzando la funzione MinMaxScaler con range 0,1). A seguire si è proceduto con la nuova divisione in train e test mantenendo le stesse proporzioni utilizzate in precedenza. Infine, come ultima operazione prima dell'addestramento del modello, è stato necessario rimodellare il dataset tramite funzioni apposite per ottenere una particolare tipologia di struttura dati richiesta dalla rete.

Anche nel caso delle LSTM sono stati testati diversi modelli in modo tale da ottenere la migliore combinazione degli iperparametri. In questo caso gli iperparametri su cui è stata posta l'attenzione sono: lookback, il numero di osservazioni del passato da utilizzare, il numero di layer e il numero di neuroni contenuti in ciascuno di essi, il

numero di epoche e la dimensione del batchsize.

L'architettura della rete realizzata è così definita in maniera sequenziale:

- Layer LSTM da 10 neuroni
- LeakyReLU come funzione di attivazione
- Layer di Droput con dropout rate di 0.2
- Layer LSTM da 5 neuroni
- LeakyReLU come funzione di attivazione
- Layer di output di tipo Dense da 1 neurone

Il modello è stato compilato utilizzando come funzione di perdita il mean squared error e come ottimizzatore adam ed è stato allenato per 100 epoche con un batchsize pari a 32. Non da ultimo, dal momento in cui l'oggetto delle analisi è una serie storica e dunque l'ordine delle osservazioni risulta fondamentale, è stato considerato il parametro shuffle, impostando il rispettivo valore a False in modo tale che ad ogni epoca si eviti lo shuffle dei dati di training.

Inoltre per cercare di ridurre ulteriormente l'overfitting del modello che inizialmente ha rappresentato un problema nella fase di modellazione della rete, oltre al dropout che permette di ignorare alcuni unità neurali, è stato preso in considerazione l'early stopping: in questo il training del modello viene interrotto quanto la metrica monitorata (loss) non registra più alcun miglioramento. In particolare l'early stopping è stato implementato in modo tale che se dopo due epoche successive non si registra un miglioramento in termini di loss allora la fase di training del modello verrà interrotta.

Il modello addestrato facendo uso di una funzione di previsione di tipo ricorsivo ha registrato sul test set un valore di MAPE pari a 6.45% e graficamente ha ottenuto il risultato osservabile in figura 10.

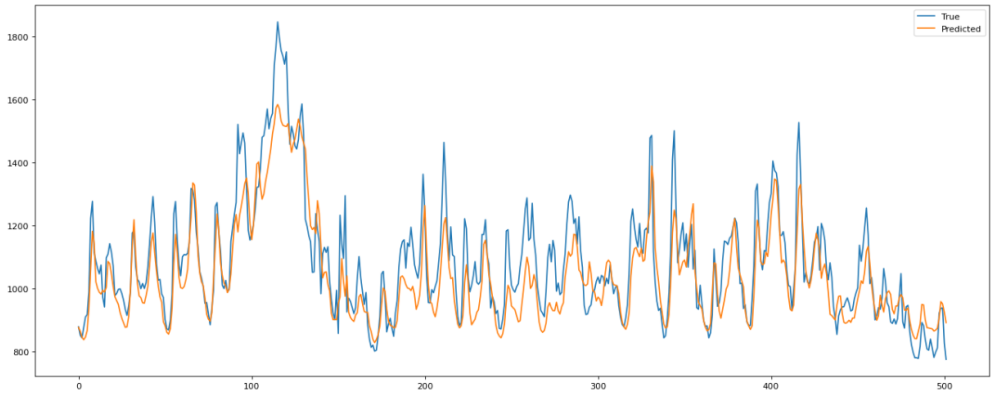

Figure 10: Previsioni sul test set LSTM

6 Conclusioni e Previsioni

In conclusione per ogni famiglia di modelli quelli finali che hanno registrato le migliori performance sono i seguenti:

ARIMA (2, 1, 2)(1, 1, 1)[24] con 3 sinusoidi di periodo 168.

UCM con random walk con drift e stagionalità modellata da 3 sinusoidi di periodo 24 e 2 sinusoidi di periodo 168.

LSTM con la seguente architettura:

- Layer LSTM da 10 neuroni
- LeakyReLU come funzione di attivazione
- Layer di Droput con dropout rate di 0.2

- Layer LSTM da 5 neuroni
- LeakyReLU come funzione di attivazione
- Layer di output di tipo Dense da 1 neurone

I risultati in termini di MAPE ottenuti dai seguenti modelli sul test set sono riassunti nella tabella 3.

Table 3: Risultati finali

Modello	**MAPE (%)**
ARIMA	10.62
UCM	11.02
machine learning	6.45

Il miglior modello osservando le performance sul test set in termini di MAPE è stato ottenuto dalla rete LSTM, della famiglia machine learning. Per quanto riguarda invece i modelli ARIMA e UCM registrano performance molto simili.

Questi modelli sono stati perciò utilizzati per generare le previsioni per il mese di marzo. In quest'ultimo caso l'addestramento è stato fatto tuttavia su tutti i dati a disposizione. Nei grafici a seguire (figura 11, figura 12, figura 13) vengono riportati graficamente i risultati ottenuti.

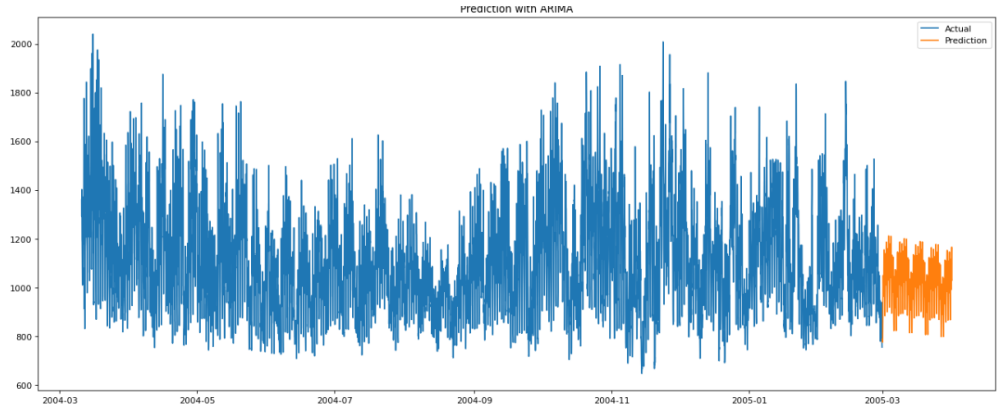

Figure 11: Previsioni marzo 2005 ARIMA

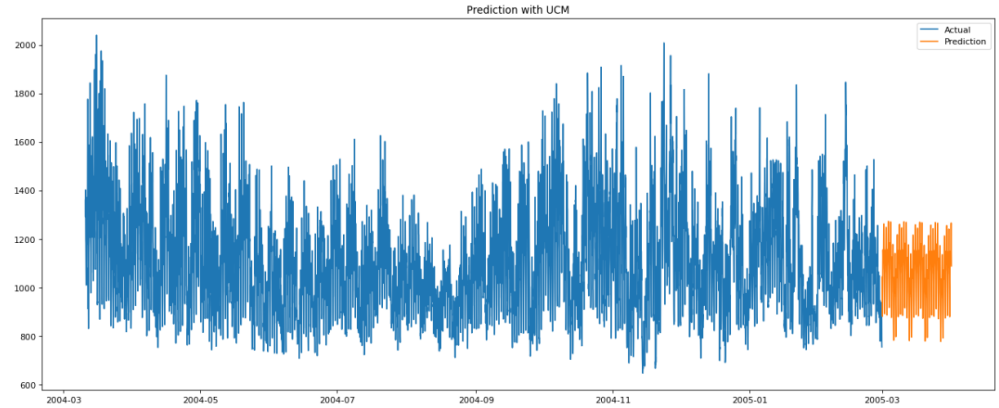

Figure 12: Previsioni marzo 2005 UCM

Ipotizzando alcuni sviluppi futuri si potrebbe porre una maggiore attenzione alla modellazione della stagionalità per cercare di avere dei risultati migliori e ai parametri delle reti neurali per evitare problemi di overfitting.

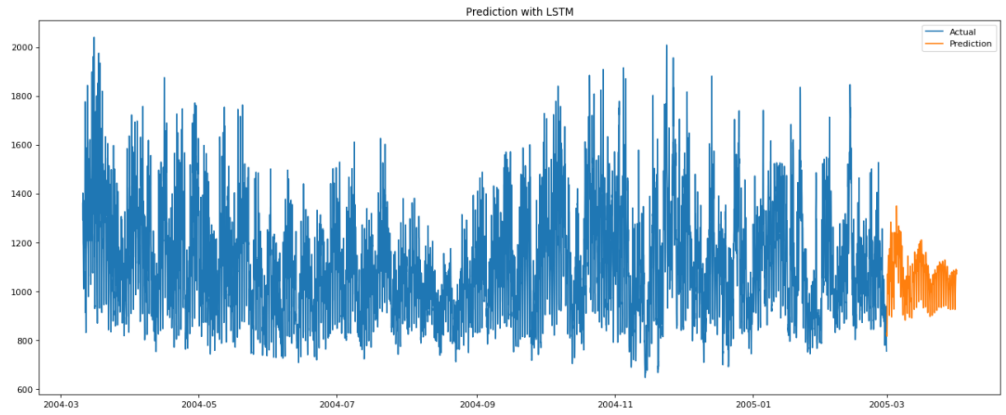

Figure 13: Previsioni marzo 2005 LSTM

Appendice A

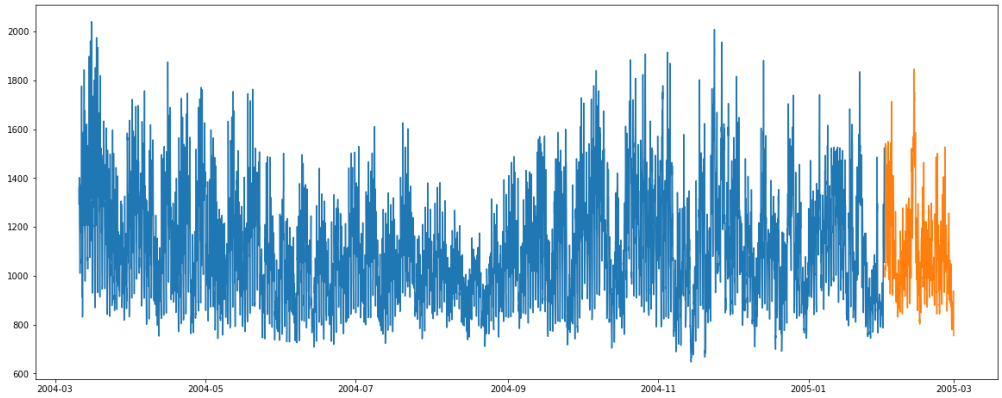

Figure 14: Split train test set

Figure 15: Serie storica a granularità giornaliera, settimanale, mensile

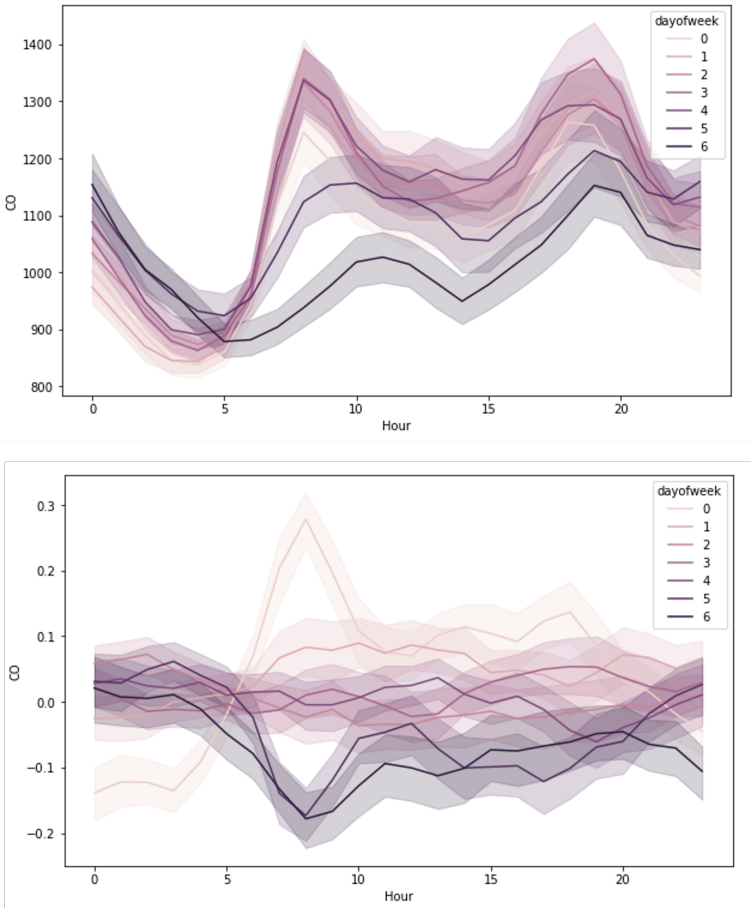

Figure 16: Stagionalità giornaliera pre e post preprocessing

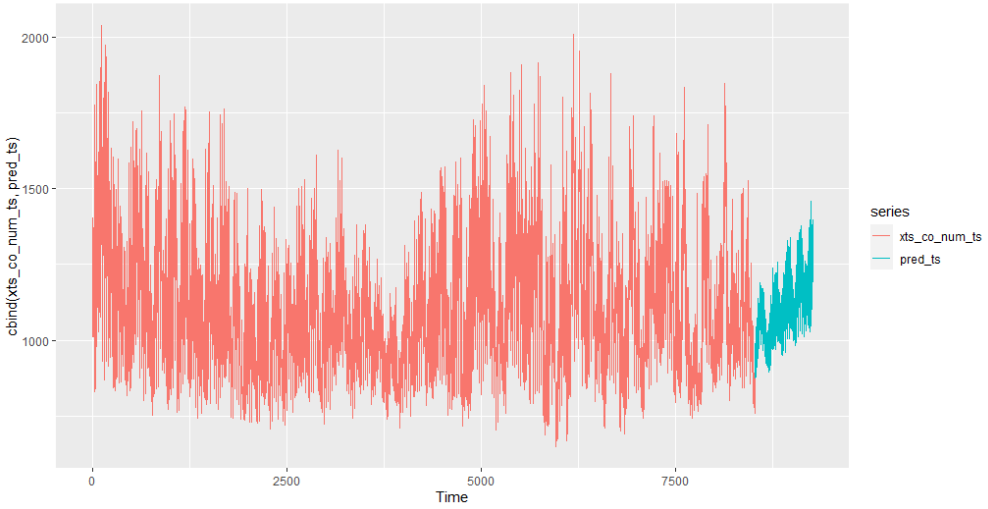

Figure 17: Previsioni marzo 2005 KNN