ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

CORSO DI LAUREA MAGISTRALE IN
INGEGNERIA CIVILE CURRICULUM "IDRAULICA E TERRITORIO"

MODELLI DATA-DRIVEN PER LA PREVISIONE DELLA PRODUCIBILITÀ IDROELETTRICA IN GRUPPI DI BACINI A SCALA REGIONALE

Tesi di laurea magistrale in "Gestione degli invasi e degli impianti idroelettrici"

Relatrice

Prof.ssa Elena Toth

Presentata da

Simone Sanna

Correlatori

Ing. Mattia Neri

Ing. Ludovica Ruggeri

Sessione Marzo 2024

Anno Accademico 2022/2023

INDICE

1 - INTRODUZIONE

Questo elaborato di tesi parte da una collaborazione tra il Dipartimento di Ingegneria Civile, Chimica, Ambientale e dei Materiali (DICAM) dell'Università di Bologna ed Enel Green Power Italia s.r.l. Enel Green Power rappresenta una delle realtà più importanti in Italia e nel mondo per quanto riguarda la produzione di energia da fonti rinnovabili. In particolare in Italia gestisce circa 13 GW sui 22GW di potenza idroelettrica installata sul territorio nazionale.

L'insieme degli impianti idroelettrici gestititi da Enel Green Power sul territorio italiano (ognuno di essi legato al bacino imbrifero sotteso dalla sezione di chiusura in corrispondenza dell'impianto), è stato suddiviso in sei zone o gruppi, che Enel considera sufficientemente omogenei dal punto di vista del comportamento idrologico e climatico e per ognuno dei quali si intende modellare la producibilità aggregata.

Per necessità di mercato è infatti importante per l'azienda stimare l'energia producibile nel futuro prossimo. Dopo aver elaborato, a partire da banche dati europee, le stime per gli anni passati e le previsioni per il futuro delle principali variabili meteorologiche e idrologiche che influenzano la disponibilità idrica e di conseguenza, indirettamente l'entità dei volumi turbinabili, ci si è posti come obiettivo di questo elaborato la creazione di un modello che fornisca una stima dell'energia producibile inserendo in ingresso al modello tali variabili.

Il processo ha previsto lo sviluppo di diversi modelli per ciascuna zona, i quali sono stati confrontati tra loro per stabilire quale fosse il più adatto allo scopo sulla base di criteri di valutazione statistica. Per la creazione dei modelli si è utilizzato il software Matlab[1], andando ad esplorare numerosi modelli di regressione lineare e non lineare e utilizzando diverse tipologie di reti neurali. È stata attribuita particolare importanza alla "memoria" del modello, per cui, il modello che utilizza le reti neurali ricorrenti del tipo Long Short-Term Memory, che ha permesso le migliori prestazioni, è stato quello scelto per le modellazioni successive.

[1] MATLAB. (R2023b). Natick, Massachusetts: The MathWorks Inc.

2 – CONTESTO GEOGRAFICO E REGIONI DI STUDIO

Gli impianti idroelettrici gestiti da Enel Green Power sono presenti su tutto il territorio italiano. Ogni impianto è stato associato al bacino imbrifero sotteso dalla sezione di chiusura ipotizzata sull'impianto stesso. L'insieme dei bacini è stato poi suddiviso in 6 zone o gruppi per i quali il comportamento climatico e idrologico può essere considerato omogeneo. Le zone individuate possono essere riassunte come segue:

- Zona 1 – Bacini dell'Italia centro meridionale siti sul versante tirrenico degli Appennini.
- Zona 2 – Bacini dell'Italia centro meridionale siti sul versante adriatico degli Appennini.
- Zona 3 – Bacini alpini e prealpini lombardi e trentini, comprendono il bacino del Chiese, del Brembo, del Mincio, dell'Oglio, del Serio e dell'Adda.
- Zona 4 – Comprende tutti gli impianti sugli affluenti del Po in Valle d'Aosta e Piemonte. È chiusa dalla sezione del Po a Isola Serafini, che è l'impianto di Enel Green Power posto più a valle lungo il fiume.
- Zona 5 – Bacini dell'Italia meridionale e delle isole maggiori.
- Zona 6 – Bacini dell'alto Veneto e del Trentino-Alto Adige, comprende i bacini chiusi agli sbarramenti di Nervesa, Mignano, Cogollo, Zevio.

Come si può notare dalla figura 2.1 le varie zone sono composte da bacini molto eterogenei anche come dimensione e in alcune zone, geograficamente distanti l'uno dall'altro. In ciascuna zona inoltre sono presenti sia impianti a serbatoio che ad acqua fluente. Questi fattori rendono difficile stabilire un approccio univoco da applicare indiscriminatamente a tutte le zone. Quasi tutti i bacini si sviluppano in territorio montano, ci si aspetta quindi un contributo importante della precipitazione nevosa, con importanze maggiori all'aumentare della latitudine.

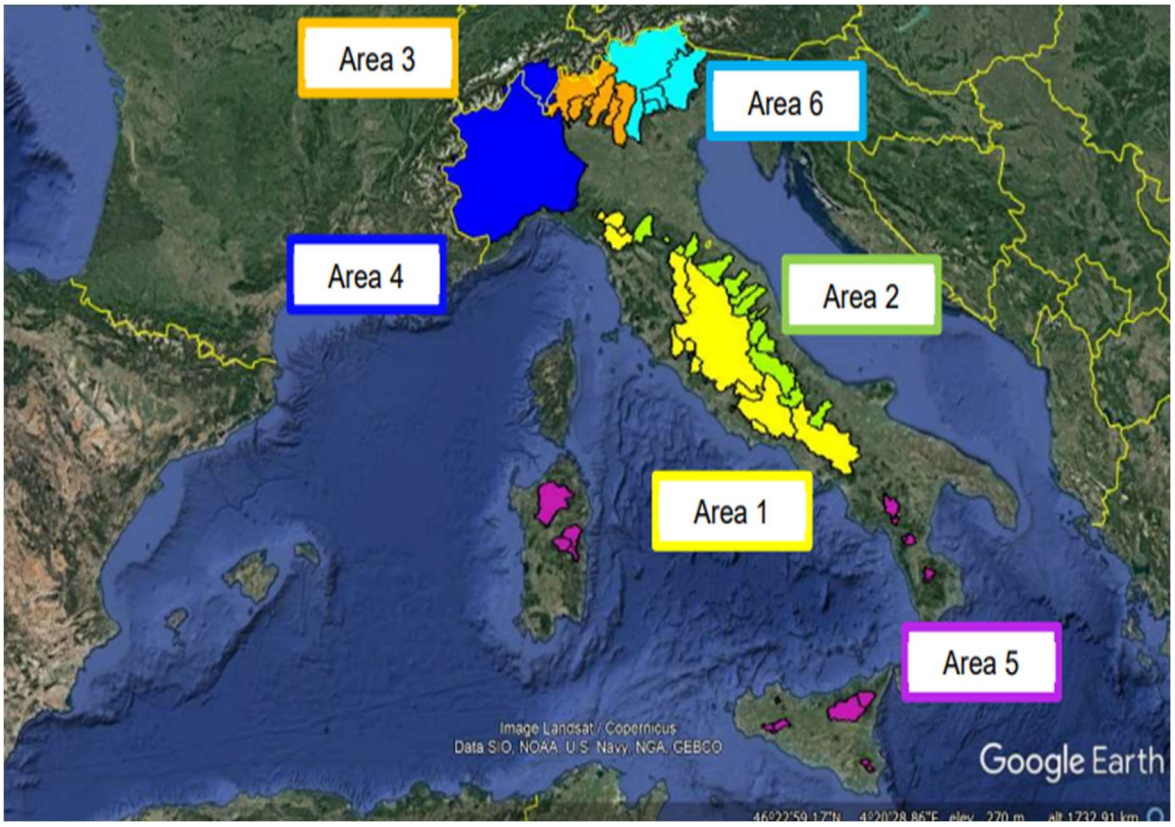


Fig. 2.1 – Suddivisione in zone.[2]

[2] Bonafè A., Ruggeri L., (2022). Effetti cronici del climate change sulla producibilità idroelettrica di EGP Italia

3 – DATI

3.1 – Dati a disposizione e scelta dei dati da raccogliere.

L'interesse di Enel Green Power è rivolto alla previsione dell'energia producibile nei mesi futuri rispetto a un determinato istante di previsione. In uscita dal modello verrà quindi usata una grandezza che sia rappresentativa di questa quantità. Come dato di output del modello ci è stato quindi fornito un valore definito producibilità, che rappresenta l'energia potenzialmente producibile da uno o più impianti in un determinato periodo di tempo. Ogni zona ha un valore unico rappresentativo dell'energia prodotta su tutta l'area e per ogni zona sono stati forniti i valori mensili dal 1990 al 2022. Questa grandezza è stata fornita in forma adimensionale, dividendone il valore mensile per la media (su tutti gli anni di osservazione) della producibilità totale annua. Poiché il dato di output è su scala mensile e disponibile dal 1990 al 2022, tutti i dati da inserire in input al modello sono stati raccolti per lo stesso intervallo di tempo e sulla stessa scala temporale.

Per la scelta dei dati da inserire in ingresso si è valutato quali fattori potessero incidere sulla produzione idroelettrica nelle zone interessate. Poiché la produzione di energia idroelettrica dipende strettamente dalle portate turbinate, è stato immediato considerare prioritarie le grandezze legate alla produzione dei deflussi fluviali, quindi innanzitutto la precipitazione, ovvero pioggia e neve che costituiscono il principale apporto ai volumi turbinati dagli impianti. L'evapotraspirazione è stata invece presa in considerazione come principale perdita di volume idrico disponibile. Il deposito nevoso è stato valutato per tenere conto dell'apporto idrico a disposizione per i mesi futuri e in qualche modo sottratto dall'apporto del mese corrente. La temperatura dell'aria inoltre gioca un ruolo fondamentale nel ciclo idrologico anche se il suo ruolo è già tenuto in conto nelle variabili che governano i processi evapotraspirativi, di accumulo e scioglimento nivale.

Tutte le variabili in ingresso, mediate spazialmente sulle aree di interesse, sono state ricavate dal data store di ERA5 del sito di Copernicus. Copernicus è la componente del programma spaziale europeo che si occupa dell'osservazione della Terra. Offre servizi di informazione che attingono dall'osservazione satellitare della Terra, da misure al suolo e da modelli meteorologici e climatici. La Commissione Europea gestisce il programma implementato in collaborazione con gli Stati Membri, l'Agenzia Spaziale Europea (ESA), l'Organizzazione

Europea per l'esercizio dei satelliti meteorologici (EUMETSAT), il Centro Europeo per le previsioni meteorologiche a medio termine (ECMWF), le Agenzie dell'UE e Mercator Océan ("About Copernicus", Copernicus).[3] In particolare il data store di ERA5 contiene la rianalisi dei dati meteorologici e climatici su tutto il globo dal 1940 al presente. La rianalisi di ERA5 combina i dati ricavati tramite un modello meteorologico con le osservazioni satellitari e misure al suolo provenienti da tutto il mondo utilizzando un approccio fisicamente basato. Questo metodo, chiamato *data assimilation* è basato su quello utilizzato dai centri di previsione meteorologica, in cui, il modello viene aggiornato costantemente ad intervalli di tempo prestabiliti, per esempio ogni 12 ore per l'ECMWF, combinandolo con le nuove osservazioni disponibili. La rianalisi, che si svolge in tempo differito, ha come vantaggio il fatto di non dover fornire previsioni in tempo reale per cui si ha la possibilità di raccogliere osservazioni di qualità più elevata. ERA5 fornisce dati orari relativi a grandezze atmosferiche, oceaniche e terrestri. Gli aggiornamenti giornalieri sono disponibili con circa 5 giorni di ritardo rispetto al tempo al quale si riferiscono. Questi dati, rilasciati a distanza di pochi giorni fanno pate del database ERA5T. Se vengono trovati degli errori importanti, questi dati possono essere diversi da quelli definitivi ERA5 rilasciati dopo circa 2 o 3 mesi. La risoluzione spaziale è di 0,25° di latitudine per 0,25° di longitudine (circa 28 km per 21 km). Per le variabili atmosferiche e i formati disponibili per il download sono il GRIB e il NetCDF. ("ERA5 hourly data on single levels from 1940 to present", Copernicus).[4]

Per le previsioni delle variabili meteorologiche Enel Green Power analizza le previsioni stagionali del Cento Europeo (ECMWF) e del Centro Americano (NOOA) per definire lo scenario più probabile sui territori di interesse. Tuttavia per questo studio sono state considerate le previsioni stagionali del Centro Europeo, scaricabili dal sito di Copernicus, tramite il database chiamato *Seasonal forecast monthly statistics on single levels*. Questo database fornisce dati a risoluzione temporale mensile delle previsioni per uso operativo delle principali variabili meteorologiche del Copernicus Climate Data Store (C3S) per i successivi sei mesi. I dati scaricabili dal database sono il frutto della combinazione degli output di diversi modelli climatici sviluppati indipendentemente. ("Seasonal forecast monthly statistics on single levels", Copernicus).[5] Questa tipologia di dati, a differenza delle previsioni meteorologiche a breve

[3] https://www.copernicus.eu/en/about-copernicus

[4] https://cds.climate.copernicus.eu/cdsapp#!/dataset/reanalysis-era5-single-levels?tab=overview

[5] https://cds.climate.copernicus.eu/cdsapp#!/dataset/seasonal-monthly-single-levels?tab=overview

termine o dei dati storici ricavabili da ERA5 è caratterizzata da un'incertezza notevolmente maggiore. I dati scaricabili sono previsioni delle variabili meteorologiche su scala temporale mensile per un periodo di tempo che va da uno a sei mesi nel futuro. Sono inoltre disponibili le previsioni effettuate dal 2017 al presente. La risoluzione spaziale è di 1° di latitudine per 1° di longitudine (circa 110km per 80km). I formati disponibili per il download sono il GRIB e il NetCDF.

Considerati i fattori che si ritengono maggiormente influenti sulla generazione del deflusso, e quindi sulla produzione idroelettrica e la disponibilità nei database sopra citati delle grandezze meteorologiche utili ai fini della modellazione, si è infine optato per l'utilizzo delle cinque grandezze in input elencate di seguito.

- Precipitazione totale (P) [mm/mese] – Rappresenta la somma dell'altezza di precipitazione liquida e solida (pioggia e neve), che è caduta sulla superficie terrestre durante l'intero intervallo temporale preso in considerazione. Non include nebbia, rugiada o precipitazione evaporata prima di toccare il suolo. La precipitazione solida è misurata in altezza d'acqua equivalente.
- Temperatura (T) [°C] – Temperatura media dell'aria due metri sopra la superficie terrestre.
- Snowfall (SF) [mm/mese] – Rappresenta l'altezza di precipitazione nevosa che cade sulla superficie terrestre. È misurata in altezza d'acqua equivalente.
- Snow depth (SD) [mm/mese] – Altezza d'acqua che si avrebbe se la neve accumulata sulla superficie considerata si sciogliesse e si distribuisse uniformemente.
- Evapotraspirazione (ET) [mm/mese] – Somma dell'evaporazione dalla superficie terrestre e della traspirazione dalla vegetazione. È indicata in altezza di acqua equivalente e ha valori negativi. In caso di valori positivi viene considerata condensa.

3.2 – Elaborazione dei dati.

Una volta stabiliti i parametri utili per la costruzione dei modelli si è proceduto scaricando e organizzando i dati da utilizzare. Enel Green Power ha inizialmente fornito i dati adimensionalizzati di producibilità dal 1990 al 2022, la serie storica di precipitazione e temperatura mensile dal 1990 al 2022 e le previsioni di precipitazione e temperatura, a 1, 2, 3, 4, 5 e 6 mesi di distanza per tutti e sei i gruppi da studiare dal 2017 al 2022 sotto forma di

anomalie rispetto alla media del periodo 1993-2016. Le grandezze rese disponibili da Enel Green Power sono tutte a scala temporale mensile e già rappresentative dell'intera superficie delle varie zone.

Dai database di Copernicus sono quindi stati scaricati i dati della serie storica e delle previsioni delle variabili aggiuntive e cioè di evapotraspirazione e dei dati relativi alla neve, sotto forma di snowfall e snow depth. La serie storica ha una risoluzione temporale oraria mentre le previsioni meteorologiche hanno una risoluzione temporale mensile. Per il download delle previsioni meteorologiche si sono utilizzati i seguenti parametri:

- Originating centre – ECMWF
- System – 51
- Product type – Ensemble mean
- Format - NetCDF

I parametri utilizzati per il download sono stati forniti da Enel Green Power e sono gli stessi parametri utilizzati per scaricare le previsioni di precipitazione e temperatura. L'originating centre rappresenta l'ente che fornisce i dati, il system invece è il numero identificativo del modello. Sia per le previsioni meteorologiche che per la rianalisi dei dati storici si sono scaricati i dati di tutte le celle comprese in un dominio compreso entro le latitudini tra 35° e 48° e longitudini tra i 6° e i 19°. Per via delle dimensioni dei file in download, è stato necessario suddividere i dati in 3 intervalli temporali successivi, ciascuno della durata di circa dieci anni per quanto riguarda la serie storica. I dati di previsione per i sei mesi successivi invece sono stati scaricati singolarmente per ogni mese. Il formato scelto è il NetCDF (Network Common Data Form). "NetCDF è un set di librerie software e dati autodescrittivi indipendenti dal sistema operativo, atti alla creazione, accesso, modifica e condivisione di dati array oriented".[6] Per elaborare questo tipo di dato è stato necessario utilizzare il software RStudio.[7] Per l'estrazione dei dati è stato necessario installare il pacchetto ncdf4.[8] Tramite questa estensione del software è stato possibile ottenere le grandezze in un formato vettoriale facilmente accessibile. Si è proceduto quindi estraendo i vettori contenenti i dati di latitudine, longitudine, istante temporale (ora) e valore della grandezza desiderata per ogni cella. Il dato temporale dei file scaricati è

[6] https://docs.unidata.ucar.edu/netcdf-c/current/index.html

[7] R Core Team (2023). _R: A Language and Environment for Statistical Computing_. R Foundation for Statistica l Computing, Vienna, Austria. <https://www.R-project.org/>.

[8] https://cirrus.ucsd.edu/~pierce/ncdf/

rappresentato da numeri progressivi nei quali l'1 rappresenta la prima ora del 01/01/1900; il dato è stato quindi riadattato trasformandolo in un formato più facilmente leggibile per poter verificare in elaborazione l'intervallo di tempo selezionato (aaaa-mm-gg, hh). I dati geografici sono quindi stati assemblati in una matrice associando ad ogni cella, il cui centro è rappresentato da una coppia di latitudine e longitudine, un indice di posizione. La matrice è stata poi trascritta su un file di testo. La matrice con le coordinate è stata in seguito importata sul software QGIS[9] tramite il quale si sono visualizzati i punti e gli shapefile dei gruppi nello stesso sistema di riferimento.

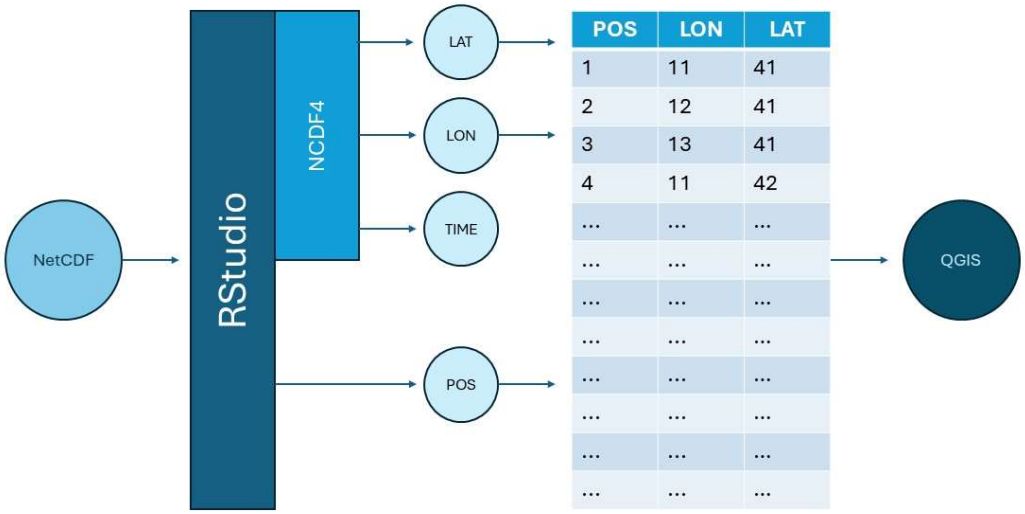


Fig. 3.1 – Schema di estrazione della matrice delle coordinate.

Tramite la funzione Voronoi polygons del software si sono associate a ciascun punto delle aree di pertinenza (che corrispondono all'incirca alle celle che hanno quel punto come centro), applicando il metodo dei poligoni di Thiessen.[10] Dopo questo passaggio tramite la funzione clip del software è stato ritagliato lo shapefile contenente i poligoni di Thiessen (cioè le celle coi valori delle variabili meteo), sui contorni dei gruppi di bacini in modo da ottenere per ogni cella l'estensione dell'area di propria competenza all'interno delle zone interessate. Il risultato finale per la griglia dei dati ERA5 è riportato in figura 3.2. I dati per ciascun gruppo inerenti indice di

[9] QGIS.org, 2023. QGIS Geographic Information System. QGIS Association. http://www.qgis.org

[10] Moisello U., (1999). Idrologia Tecnica.

posizione, latitudine, longitudine e area (all'interno del gruppo stesso) sono stati esportati in un documento di testo per poterli inserire nuovamente sul software RStudio per l'ultima elaborazione.

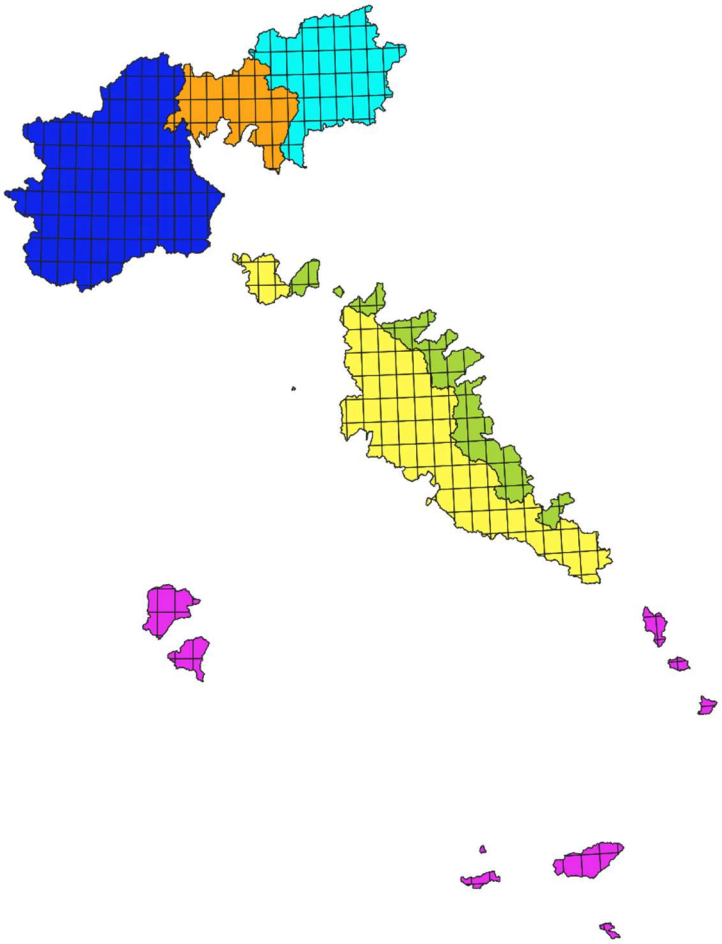


Fig. 3.2 – Poligoni di Thiessen.

Dalla figura 3.3 si può notare (prendendo come esempio l'area 4) la differenza nelle dimensioni delle aree di competenza per ciascun punto nel caso delle serie storiche (0,25°) e delle previsioni (1°). Per queste ultime infatti la risoluzione dei grigliati risulta nettamente inferiore.

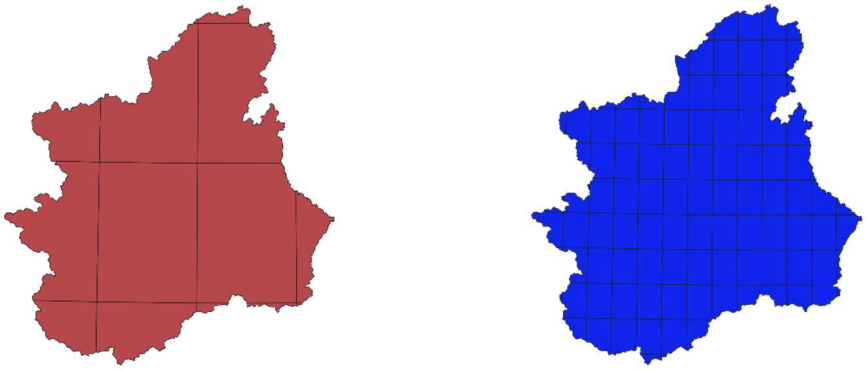


Fig. 3.3 – Confronto tra grigliati per il gruppo 4. A sinistra è rappresentato il grigliato associato alle previsioni, a destra il grigliato associato alle osservazioni.

Per ogni gruppo si è stabilito un peso da associare a ciascun punto, calcolato come l'area associata al punto stesso divisa per la somma delle aree contenute nel gruppo.

$$w_i = \frac{A_i}{\sum_i A_i}$$

Ogni valore puntuale della grandezza in analisi è stato poi moltiplicato per il peso corrispondente e si sono sommati i valori ottenuti, ottenendo così la media spaziale sulla superficie del gruppo considerato. Le serie temporali a scala oraria di tali medie areali per ogni variabile meteorologica ricavate tramite questo procedimento sono state poi convertite e organizzate in un foglio di calcolo. Per ottenere i valori mensili a partire da quelli orari, per i dati di snowfall ed evapotraspirazione della serie storica è stata effettuata la somma dei valori orari per ottenere il valore totale mensile (in mm/mese) mentre per lo snow depth della serie storica è stata effettuata la media temporale per ottenerne il valore medio mensile (mm). Le previsioni meteorologiche invece erano di due tipologie diverse; i dati forniti da Enel Green Power (precipitazione e temperatura) sono anomalie rispetto alla media mensile 1993-2016, per cui si sono calcolati inizialmente i valori medi e in seguito sono state sommate algebricamente

le anomalie per tornare ai valori mensili assoluti delle grandezze considerate. I dati di snowfall ed evapotraspirazione invece, ottenuti in m/s sono state trasformati in mm/mese.

3.3 – Analisi dei dati.

Per loro natura, i dati che influenzano l'idrologia di un territorio sono difficili da stimare anche quando si riferiscono al passato. I dati qui utilizzati per le serie storiche inoltre sono frutto di una combinazione di osservazioni e modelli, motivo per cui l'incertezza sul dato è ancora maggiore. L'incertezza è ancora superiore per grandezze più difficilmente misurabili come l'evapotraspirazione e lo snow depth, per le quali è difficile avere un valore corretto anche tramite uno strumento di misura. Quando poi la grandezza viene valutata solo tramite modello, come avviene per le previsioni meteorologiche, l'incertezza diventa estremamente rilevante e crescente con la distanza nel futuro della previsione stessa. Per quanto riguarda la serie storica, trattando zone di studio molto grandi ed eterogenee e considerando che questi saranno i dati in ingresso al modello, sui quali verranno tarati i parametri, si può considerare che l'incertezza del dato rientri all'interno dell'incertezza generale della modellazione, ipotizzando che la scelta dei parametri possa compensare eventuali errori sistematici dei dati osservati. Le previsioni meteorologiche spesso non rispecchiano il reale comportamento del sistema e possono compromettere le stime della producibilità indicando delle grandezze da inserire in input molto diverse da quelle che si verificheranno in realtà. Per valutare l'attendibilità delle previsioni e confrontarle con le serie storiche si è deciso di rappresentare graficamente la serie temporale dal 2017 al 2022 delle osservazioni per ogni gruppo e di volta in volta inserire nello stesso grafico i valori previsti per tutti i lead-time di previsione. Quindi, poiché a ogni mese, per ogni gruppo, sono associati sei differenti valori di previsione, per ogni grandezza si avranno sei grafici differenti per gruppo. Per ogni set inoltre si è calcolato un indicatore per valutare di quanto i valori di previsione si discostassero dal valore effettivamente osservato. Si è scelto di utilizzare la radice dell'errore quadratico medio (RMSE, root mean squared error), la quale aumenta all'aumento dello scostamento delle previsioni dai dati osservati.

$$RMSE = \sqrt{\frac{\sum_{i=1}^{N}(O_i - P_i)^2}{N}}$$

Si sono analizzati i risultati ottenuti dai grafici e dalla valutazione degli indicatori (in appendice A sono presenti le serie temporali previste e osservate per tutte le variabili, per tutti i lead-time).

La temperatura per tutti i gruppi e l'evapotraspirazione per i gruppi 3, 4 e 6 rimangono abbastanza affidabili per tutti e sei i mesi di previsione. La precipitazione invece è stimata con buona approssimazione per un mese di lead-time, anche se i valori di picco tendono ad essere minori di quelli misurati. Questo è probabilmente dovuto alla differente risoluzione spaziale con la quale i dati della serie storica e di previsioni meteorologiche sono stati ottenuti. La maggiore superficie delle celle utilizzate per mediare le grandezze meteorologiche previste potrebbe annullare gli effetti locali come ad esempio le precipitazioni orografiche. Per lead-time superiori a due mesi, in generale le grandezze previste, in modo particolare la precipitazione, tendono ad assumere valori che tendono al valore medio della variabile, appiattendo la curva e non prevedendo gli estremi, anche se si può riconoscere l'andamento stagionale delle variabili (come si può notare in appendice A e, a titolo di esempio per il gruppo 1, in figura 3.4).

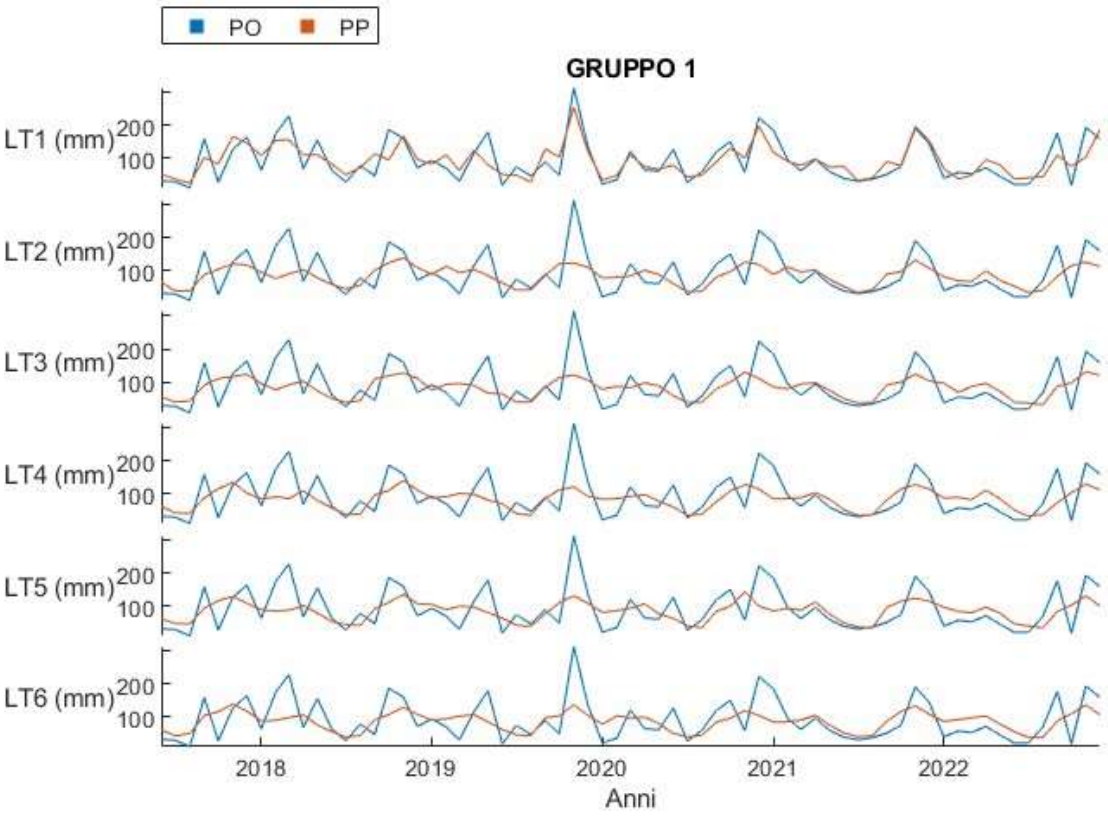


Fig. 3.4 – Serie temporale della serie storica (PO) e della serie prevista di precipitazione (PP) al variare dei mesi di lead-time (LT) per il gruppo 1.

In figura 3.5 è mostrato per ciascun gruppo l'RMSE all'aumentare del lead-time per la precipitazione. Le previsioni, già a due mesi di distanza nel futuro sono caratterizzate da errori non trascurabili, con un incremento dell'errore di circa il 30% rispetto alla previsione ad un mese, l'errore poi all'aumentare del lead-time tende ad aumentare leggermente o a mantenersi stabile.

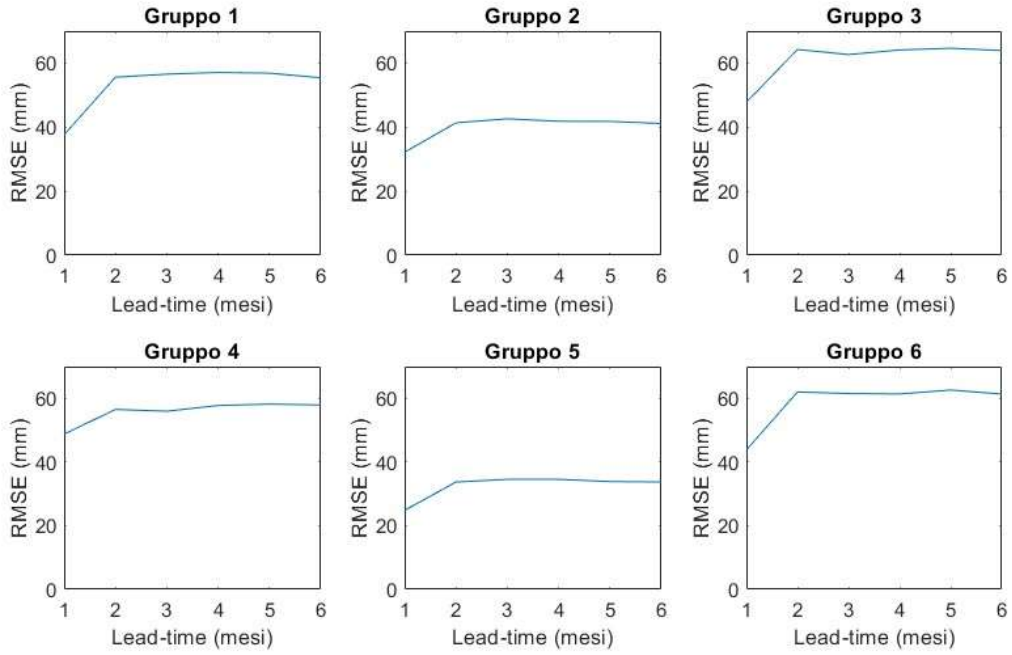


Fig. 3.5 – RMSE previsione di precipitazione al variare del lead-time.

Durante l'analisi dei dati si è inoltre notato che per i dati di evapotraspirazione si ha un problema con la stagionalità dei dati. Più il sistema sul quale l'evapotraspirazione è stata calcolata si trova a meridione, più la stagionalità risulta traslata in avanti nel tempo. Questo fenomeno è particolarmente evidente per il gruppo 5, per il quale il ciclo stagionale è completamente invertito, ma si può iniziare a notare già dai gruppi 1 e 2. Nella figura seguente (fig. 3.6) è mostrata la serie temporale dei dati di previsione di evapotraspirazione per i sei gruppi, per tre mesi di lead-time delle previsioni, a dimostrazione di quanto detto.

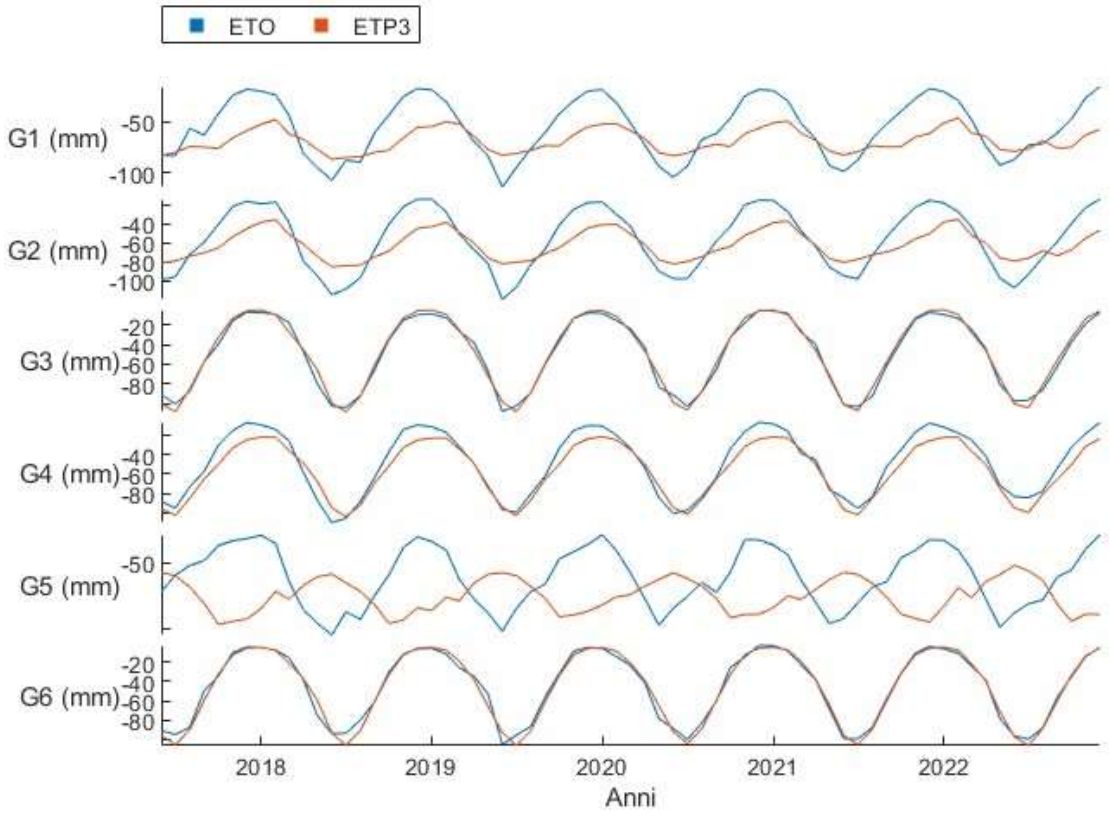


Fig. 3.6 – Serie temporale della serie storica (ETO) e della serie prevista tre mesi in anticipo dell'evapotraspirazione (ETP3) per i sei gruppi.

Per sopperire a questo problema, si è deciso di creare un set di dati alternativo per l'evapotraspirazione dei gruppi 1, 2 e 5 da utilizzare nei modelli di previsione di producibilità per paragonarne i risultati con il set ottenuto dai data base di Copernicus. Questo nuovo set è costituito semplicemente dalla media mensile della serie storica. Le previsioni su diversi orizzonti temporali dunque non cambiano. Nella figura seguente (fig. 3.7) è riportata la stessa serie temporale illustrata in precedenza ma utilizzando le medie mensili per i gruppi meridionali 1, 2, 5 anziché i dati del forecast di Copernicus.

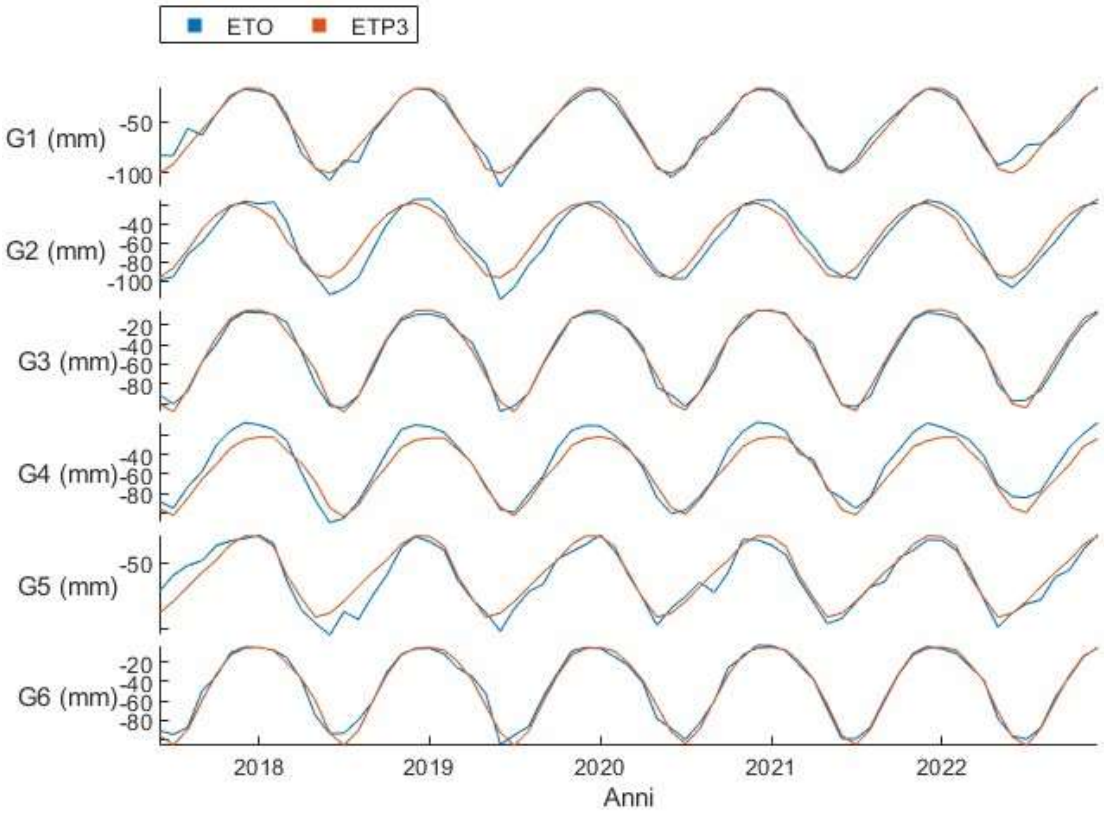


Fig. 3.7 – Serie temporale della serie storica (ETO) e della serie prevista tre mesi in anticipo dell'evapotraspirazione (ETP3) per i sei gruppi (medie mensili).

È evidente come questa soluzione approssimi meglio l'andamento osservato dell'evapotraspirazione.

4 – MODELLI

4.1 – Generalità

La producibilità è legata ai volumi d'acqua turbinati negli impianti idroelettrici, si è ipotizzato perciò di poterla stimare utilizzando un approccio simile alla modellazione idrologica afflussi-deflussi. In generale un modello idrologico può essere rappresentato tramite una funzione del tipo:

$$\boldsymbol{Y} = F(\mathbf{X}, P)$$

Dove Y è l'insieme degli output del modello, X è l'insieme delle variabili in ingresso al modello e che caratterizzano il comportamento degli output, P è il vettore contenente i parametri e F è l'insieme delle funzioni che legano input e output. Questo tipo di approccio per la previsione delle variabili in uscita viene chiamato regressione. A seconda del tipo di semplificazione che viene effettuato nella scelta del modello esso può essere di tre tipi diversi:

- Modello fisicamente basato

  Analizzando la fisica del sistema, il modello viene costruito tramite le equazioni che lo governano. Questi modelli sono spesso complessi e cercano di riprodurre la natura degli eventi in maniera rigorosa. Sia le variabili in ingresso che i parametri rappresentano un fenomeno fisico specifico. Questo genere di modello, per la sua natura complessa e dettagliata è di difficile applicazione, specialmente in sistemi di grandi dimensioni nei quali la conoscenza di tutti gli input non è spesso possibile e per i quali la difficoltà di applicazione e di computazione del modello potrebbe essere troppo elevata. Solitamente quindi non vengono utilizzati a scopo previsionale.

- Modello concettuale

  Sono modelli nei quali la realtà fisica viene semplificata, per cui si ipotizza la conoscenza delle leggi che ne determinano il comportamento. I parametri, in questo caso, non rappresentano grandezze fisiche reali ma vengono tarati tramite la calibrazione del modello.

- Modelli data driven (black box)

Sono modelli nei quali le funzioni che trasformano gli input in output, così come i parametri, non sono noti a priori. Nella prima fase, detta di training, si effettua la scelta delle variabili in ingresso e in uscita e tramite approccio numerico si cerca una relazione tra input e output. La relazione trovata è poi riapplicabile a nuovi dati di input. Non essendo dipendenti da leggi o modelli fisici predefiniti, sono adatti a situazioni nelle quali la conoscenza della fisica del sistema non è ottimale, sono inoltre facilmente applicabili. Di contro, essendo calibrati su un sistema in particolare, non sono riutilizzabili in maniera affidabile in altri contesti ma occorrerà effettuare un nuovo training.

La natura dei sistemi per i quali si vuole conoscere la producibilità, ovvero, di grandi dimensioni, composti da bacini anche molto distanti tra loro e di natura completamente differente, oltre al fatto che la natura stessa della producibilità all'interno dei gruppi è diversa a seconda che l'impianto sia a serbatoio o ad acqua fluente, ha reso evidente l'impossibilità di affrontare il problema tramite un approccio fisicamente basato. Anche un approccio concettuale non sembra la scelta migliore per i problemi esposti, per cui si è optato per effettuare una modellazione data driven.

Scelte le variabili predittrici del modello, che verranno usate come input e nota la grandezza che si vuole ottenere in output è necessario raccogliere i dati di queste grandezze relativi allo stesso periodo e alla stessa superficie geografica. Normalmente, un modello viene creato effettuando inizialmente una fase di calibrazione dei parametri seguita da una fase di validazione su dati indipendenti. Nel contesto dei modelli data driven, la fase di calibrazione viene chiamata training, mentre la fase di validazione è detta testing. I dati raccolti vengono quindi suddivisi in due set in modo da avere un set per ciascuna fase. Il training del modello è il procedimento che stabilisce le relazioni e i parametri che legano le variabili in ingresso, dette variabili predittrici o esplicative e le grandezze che si vogliono ottenere come output, dette target o in generale variabili dipendenti. L'obiettivo è quello di minimizzare la differenza tra la grandezza predetta dal modello e il suo valore osservato nel sistema (target). Il procedimento di training del modello in questo caso viene detto supervisionato poiché il target viene fornito in partenza; se non venisse fornito un target l'algoritmo sarebbe non supervisionato e procederebbe autonomamente a trovare relazioni tra le variabili (ad esempio algoritmi di clustering). Dopo il training si procede con una fase di testing tramite la quale si verifica che il modello riproduca in maniera adeguata i dati di output su un set indipendente. Per verificare la bontà di un modello si misura di quanto si discostano i valori predetti dai valori reali per dati non utilizzati per la fase di calibrazione. Questo discostamento tra dati predetti e reali viene

calcolato tramite indici statistici. Per migliorare la qualità del modello, solitamente si utilizza anche un validation set che viene spesso estratto dal set di training. In questo modo il totale dei dati a disposizione è suddiviso in 3 gruppi: training, validation e test. Questo avviene per evitare il fenomeno del sovradattamento (overfitting), ovvero il fenomeno per il quale il modello si adatta troppo ai dati usati per l'addestramento, per cui in fase di previsione dell'output, esso rischia di riprodurre relazioni che esistono solo nei dati di training e di perdere invece le relazioni di carattere generale. In fase di addestramento, dunque, si verifica che la previsione del modello addestrato sui dati del training sia performante sui dati del validation set.

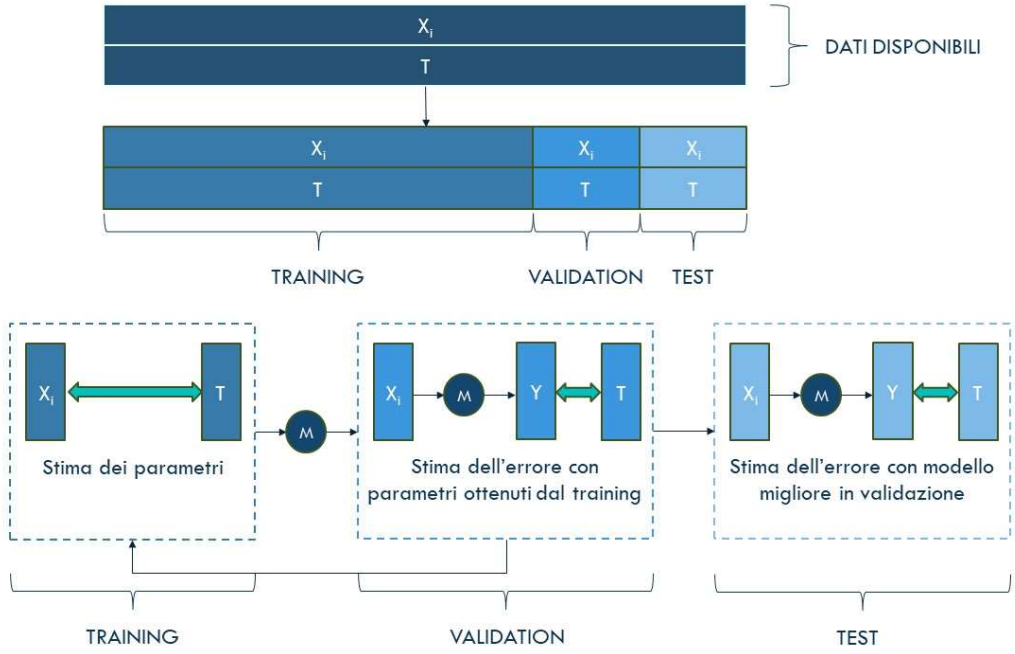


Fig. 4.1 – Schema generale del processo di training, test e validazione. Le Xi rappresentano le variabili esplicative, le T i dati di target e le Y i valori in uscita dal modello.

Per lo studio dei gruppi di bacini si è optato per suddividere i dati a disposizione, i quali vanno dal 1990 al 2022, in due set nei quali il set di training è composto da circa l'80% dei dati mentre il set di test è composto dal restante 20%. Per semplificare la trattazione si è utilizzato lo stesso set di training e di test per tutti i modelli e per tutti i gruppi e si è deciso di utilizzare un numero intero di anni. Il set di training dunque comprende tutti i dati dal 1990 al 2016 mentre il set di test comprende i restanti anni dal 2017 al 2022. I dati utilizzati per la validazione sono stati sempre scelti all'interno del set di training in maniera casuale autonomamente dal calcolatore, variandone le dimensioni a seconda del modello utilizzato.

Avendo a che fare con variabili in ingresso estremamente diverse tra loro, i cui valori assoluti divergono anche di quattro ordini di grandezza, si è deciso di standardizzare tutti i dati che verranno utilizzati. In questo modo tutti i dati saranno comparabili tra loro e saranno tutti caratterizzati da media 0 e varianza pari a 1. Per farlo si è utilizzata la formula:

$$Z_t = \frac{X_t - \mu_X}{\sigma_X}$$

Dove il pedice t è relativo al mese della misurazione, Z è la variabile standardizzata, X è la grandezza generica, μ è la media di X e σ la deviazione standard di X entrambe calcolate sull'intero periodo di training. Il processo di standardizzazione è stato effettuato sia sui dati in input che sui dati di output.

Per valutare l'efficacia e la bontà del modello si sono calcolate diverse stime dell'errore. Nelle formule seguenti P rappresenta il valore stimato dal modello mentre O è il valore osservato della stessa grandezza, N invece è la dimensione del set che si sta valutando. I valori soprassegnati rappresentano il valore medio del set considerato. Per ogni gruppo si sono quindi valutati i seguenti indicatori:

- Radice dell'errore quadratico medio (root mean square error). Stima la deviazione standard dei valori residui. Ha la stessa unità di misura della variabile sulla quale si applica (nel caso in esame è comunque adimensionale). Maggiore è il suo valore, più la stima della grandezza si discosta dai dati osservati.

$$RMSE = \sqrt{\frac{\sum_{i=1}^{N}(O_i - P_i)^2}{N}}$$

- Index of agreement.[11] Rappresenta il rapporto tra l'errore quadratico medio e l'errore potenziale e varia tra 0 e 1 dove 1 rappresenta la stima perfetta dei valori osservati mentre 0 rappresenta un modello totalmente inefficace.

$$d = 1 - \frac{\sum_{i=1}^{N}(O_i - P_i)^2}{\sum_{i=1}^{N}\left(\left|P_i - \bar{\bar{O}}\right| + \left|O_i - \bar{\bar{O}}\right|\right)^2}$$

[11] Willmott C. J., (1981). On the validation of models, physical geography

- Coefficiente di efficienza del modello di Nash-Sutcliffe.[12] Questo indicatore è quello che si è tenuto in conto maggiormente e che ha influenzato la scelta dei modelli e dei loro parametri. Esso ha valore massimo 1, il quale corrisponderebbe alla perfetta modellazione dei dati osservati. Il diminuire del coefficiente rappresenta una minore affidabilità del modello. Per un valore pari a 0 si ha che il modello è equivalente all'utilizzo della media della serie di dati come stimatore della variabile in uscita. Se il modello è meno affidabile della media i valori del coefficiente diventano negativi.

$$NSE = 1 - \frac{\sum_{i=1}^{N}(O_i - P_i)^2}{\sum_{i=1}^{N}\left(O_i - \bar{\bar{O}}\right)^2}$$

Per testare inizialmente i modelli e arrivare alla scelta del modello da utilizzare successivamente anche in previsione, si è optato per un approccio per cui i dati in input e i dati in output sono relativi allo stesso intervallo temporale. Per cui alle variabili predittive del mese generico corrisponde il target o l'output dello stesso mese, come illustrato in figura 4.2. Si utilizza quindi un lead-time pari a 0 per effettuare una simulazione.

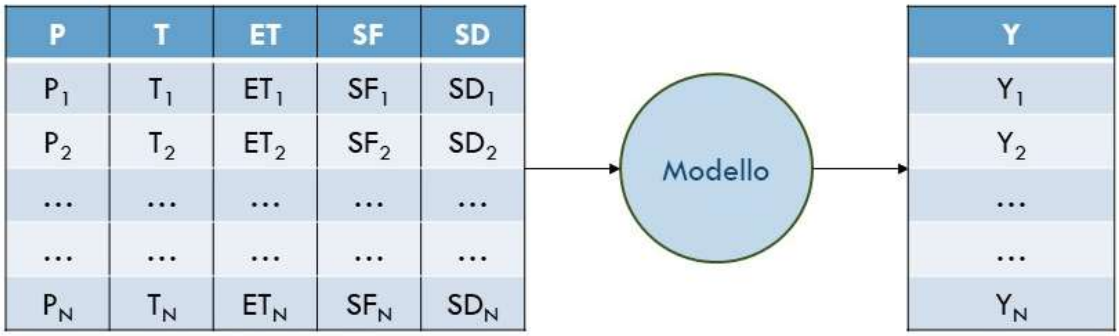


Fig. 4.2 – Schema temporale delle variabili, il pedice indica l'intervallo temporale.

Tutti i modelli sono stati creati tramite l'utilizzo del software Matlab, utilizzando le funzioni e le applicazioni interne al programma stesso.

[12] Nash J. E., Sutcliffe, J. V., (1970). River flow forecasting through conceptual models part I.

4.2 – Benchmark: simulazioni pari alla mediana dei valori sullo stesso mese

Per verificare l'utilità dei modelli creati per questa tesi, è stato necessario partire da una stima di base della producibilità. In assenza di un modello di previsione della producibilità, Enel Green Power utilizza come stima il 50° percentile (ovvero la mediana) della serie storica mensile. I valori di producibilità sono quindi stati raggruppati per mese e per gruppo, si è poi calcolata la mediana dei valori relativi al periodo di training (1990-2016). Si è in seguito organizzata la serie di mediane mensili in un vettore della lunghezza del set di test per ogni gruppo, ripetendone i valori per ogni anno. Ogni gruppo ha quindi come previsione di producibilità per il periodo di test un vettore composto da valori che si ripetono per lo stesso mese, costituiti dalle mediane relative a ciascun mese. Si è utilizzato dunque questo vettore come se fosse il risultato di un modello per calcolarne gli indicatori e valutare l'efficienza del metodo. Di seguito, sono riportati i risultati ottenuti (tab. 4.A) e la serie temporale per ciascun gruppo (fig. 4.3) e, come esempio, per il gruppo 4, il grafico più dettagliato della stima paragonata ai dati osservati (fig. 4.5) e il grafico di dispersione (scatterplot) (fig. 4.4), tutti per il periodo di test.

	GRUPPO 1	GRUPPO 2	GRUPPO 3	GRUPPO 4	GRUPPO 5	GRUPPO 6
rmse tr (-)	0.780	0.647	0.448	0.454	0.670	0.827
NSE tr	0.832	0.887	0.947	0.946	0.877	0.805
rmse test (-)	0.817	0.885	0.946	0.897	0.898	0.799
NSE test	0.388	0.621	0.788	0.635	0.616	0.311

Tab. 4.A – Indici di errore per il modello benchmark (mediana) sul training set (1990-2016) e sul test set (2017-2022).

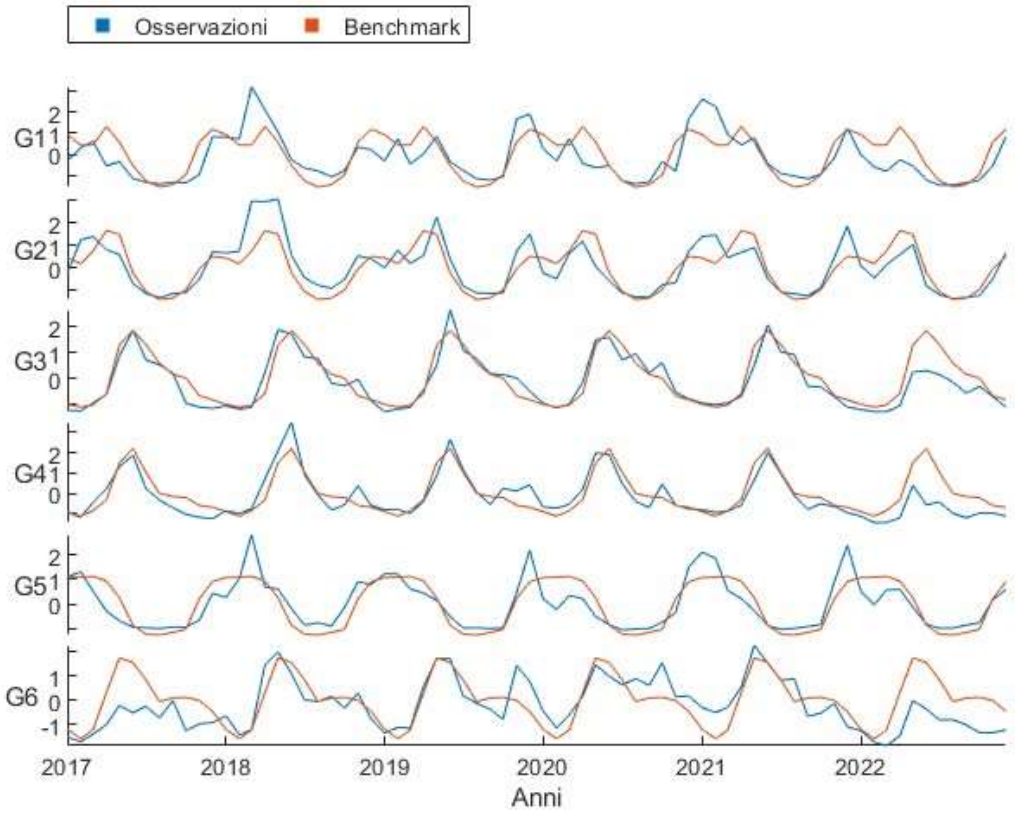


Fig. 4.3 – Confronto tra stima effettuata tramite il modello benchmark e producibilità osservata per tutti i gruppi di bacini sul set di test.

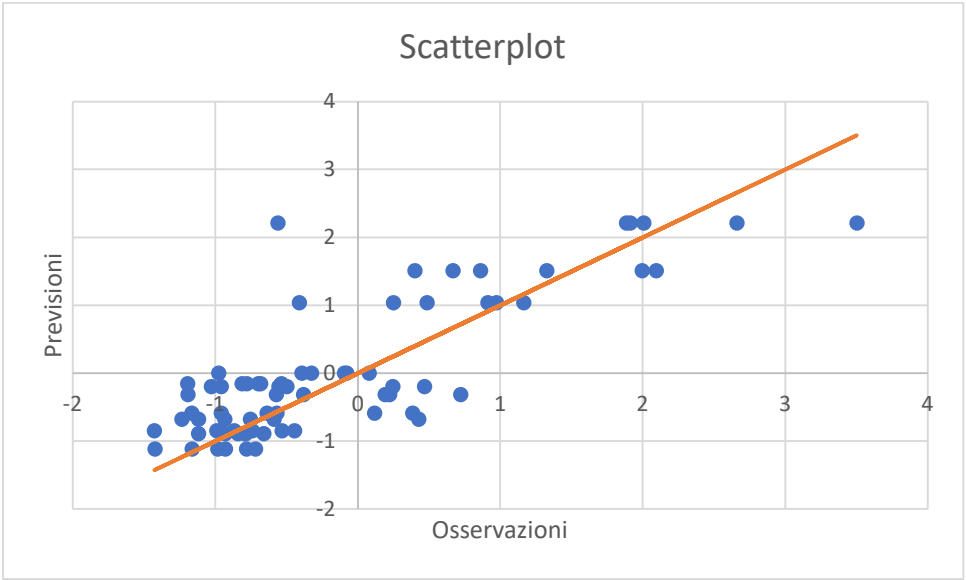


Fig. 4.4 – Scatterplot modello benchmark sul test set (2017-2022) – Gruppo 4.

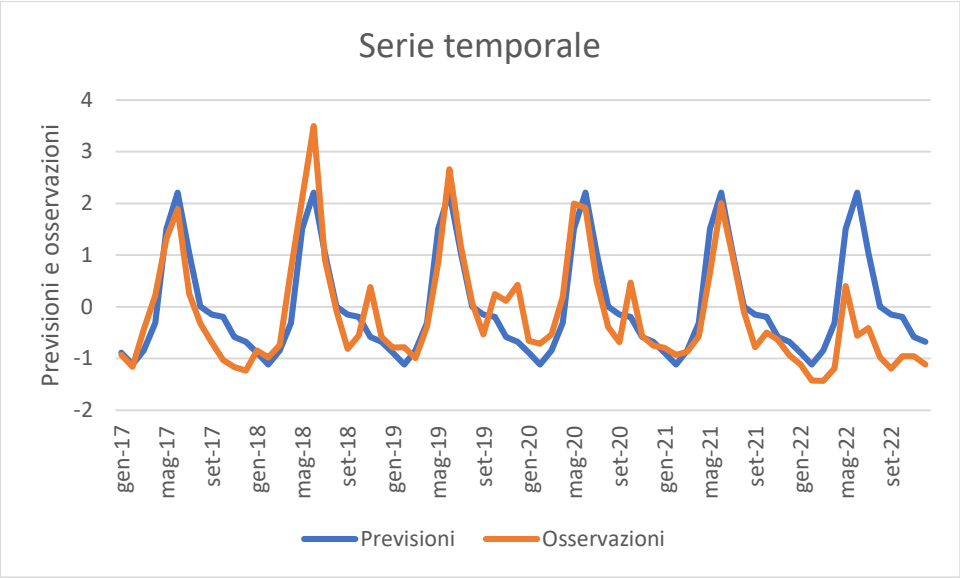


Fig. 4.5 – Serie temporale modello benchmark sul test set (2017-2022) – Gruppo 4.

4.3 – Stepwise regression

Il primo approccio per la modellazione è stato il più semplice in assoluto. Si è ipotizzata una relazione lineare tra le le variabili predittrici e la variabile dipendente sempre tutte corrispondenti allo stesso istante temporale. Avendo a disposizione più di una variabile esplicativa il tipo di relazione potrebbe essere descritto da una regressione lineare multipla. Il rapporto tra le variabili assumerebbe quindi una relazione del tipo:

$$\boldsymbol{Y(t) = \beta_0 + \beta_1 X_1(t) + \beta_2 X_2(t) + \cdots + \beta_m X_m(t) + \varepsilon}$$

Dove Y è il valore della variabile dipendente, i β sono i coefficienti dell'equazione che devono essere calibrati, le X sono i valori delle m variabili esplicative, t è il tempo ed ε è l'errore. Avendo a disposizione 5 variabili esplicative e considerando che al diminuire delle variabili si ha un aumento della componente erratica ma una diminuzione della varianza, si è cercato di trovare per ciascun gruppo il sottoinsieme ottimo delle variabili esplicative. Un tipo di regressione che permette di effettuare questa selezione è la stepwise regression. Questa funzione esegue sui dati una regressione stepwise convenzionale. Questo tipo di regressione può essere:

- Forward stepwise regression

Il modello parte senza variabili predittrici e le aggiunge una alla volta partendo da quella maggiormente correlata. Viene effettuato un test sul modello e se risulta positivo la variabile entra a far parte di esso, poi si continua per le variabili successive fino a raggiungere il criterio di arresto.

- Backward stepwise regression

Il modello parte includendo tutte le variabili esplicative e si procede eliminando la variabile meno correlata. Il procedimento si interrompe quando si raggiunge il criterio di arresto.

- Stepwise regression convenzionale

Essa rappresenta una combinazione delle due precedenti. Si parte con una stepwise forward ma alcune variabili che sono inizialmente state inserite possono essere poi rimosse durante il procedimento backward nel caso in cui la presenza di nuove variabili possa averne determinato una perdita di significatività.

Per stimare i coefficienti β della relazione lineare si è deciso di utilizzare la funzione stepwisefit di Matlab. Una volta inseriti in ingresso alla funzione la matrice delle variabili esplicative ed il vettore della variabile dipendente, essa restituisce in uscita i coefficienti della regressione lineare per tutte le variabili esplicative tra i quali può essere presente qualche zero. Questo è dovuto al fatto che il procedimento ha escluso le variabili associate ai coefficienti nulli. Dal software inoltre si possono ottenere in output anche alcuni indicatori statistici utili alla valutazione della qualità del modello stesso. La funzione è stata applicata separatamente su ciascun gruppo sul set dei dati di training. I coefficienti calibrati in questo modo si sono poi moltiplicati per il set di test delle variabili esplicative ottenendo un vettore delle stime della producibilità il quale è stato utilizzato per valutare gli indicatori per la valutazione del modello. Di seguito (tab. 4.B) sono riportati i risultati per ciascun gruppo nei quali sono presenti anche le variabili ritenute significative dal modello ed effettivamente utilizzate per la regressione. Inoltre sono riportati lo scatterplot (fig. 4.6) e il grafico temporale (fig. 4.7) relativi al gruppo 4. Dai risultati ottenuti si può notare un netto miglioramento rispetto alla stima tramite mediana anche se ancora i risultati non sono ottimali. La pioggia per tutte e sei le zone è risultata la principale variabile esplicativa. Per il gruppo 5, il quale è quello più meridionale, il procedimento ha escluso i dati di neve (snow depth, SD e snowfall, SF) dalle variabili in input. Si può supporre che questo sia dovuto a causa della scarsità di contributo da parte dell'accumulo nevoso alle portate dei corsi d'acqua per via delle alte temperature e di conseguenza scarsa percentuale di precipitazione nevosa. I gruppi dall'1 al 4, invece, utilizzano tutte le tipologie di

input anche se nei gruppi 1 e 3 il modello seleziona una sola delle due variabili relative alla neve. Il gruppo 6 (bacini del Veneto e del Trentino Alto Adige) è il più particolare in quanto vengono escluse sia la temperatura che l'evapotraspirazione, le quali sono state rilevanti per tutte le altre zone. Questo potrebbe essere dovuto alle basse temperature e di conseguenza al suo minore impatto sul ciclo dell'acqua.

	GRUPPO1	GRUPPO2	GRUPPO3	GRUPPO4	GRUPPO5	GRUPPO6
Variabili	P-T-ET-SD	P-T-SF-ET-SD	P-T-SF-ET	P-T-SF-ET-SD	P-T-ET	P-SF
rmse tr (-)	0.657	0.709	0.427	0.542	0.601	0.823
NSE tr	0.577	0.509	0.821	0.713	0.644	0.492
rmse test (-)	0.685	0.7051	0.448	0.565	0.620	0.798
NSE test	0.555	0.470	0.779	0.694	0.608	0.312

Tab. 4.B – Indici di errore per la modellazione con stepwise regression sui set di training (1990-2016) e test (2017-2022).

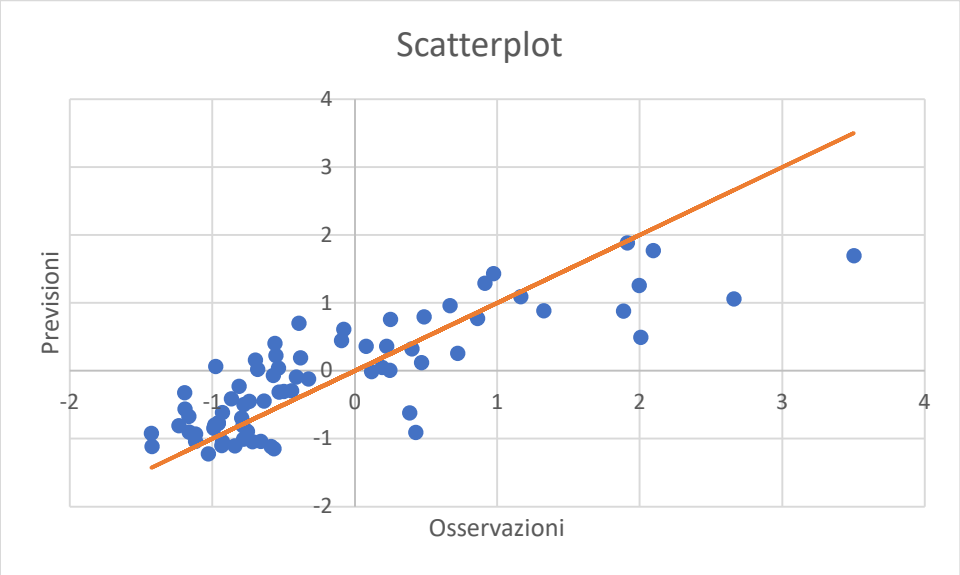


Fig. 4.6 – Scatterplot modello con stepwise regression sul test set (2017-2022) – Gruppo 4.

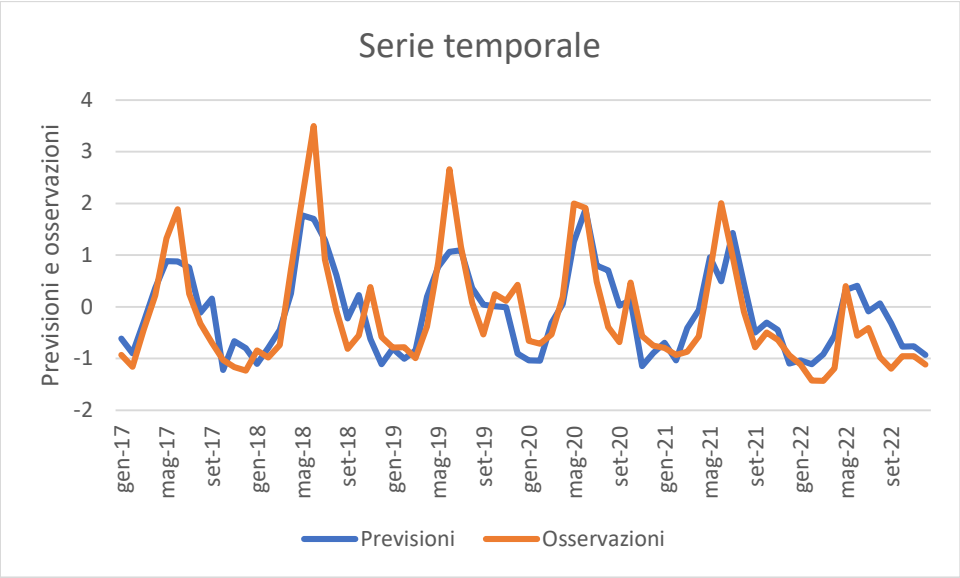


Fig. 4.7 – Serie temporale modello con stepwise regression sul test set (2017-2022) – Gruppo 4.

4.4 – Regression Learner

Lo "Statistics and machine learning" toolbox di Matlab permette di effettuare delle regressioni sulle serie di dati in maniera semplice tramite l'applicazione Regression Learner. Tramite questa interfaccia, inserendo in input la matrice delle variabili esplicative e il vettore dei target, è possibile effettuare parallelamente numerose tipologie di regressione differenti. Queste ultime vengono in seguito comparate tra di loro e tra le possibili soluzioni viene evidenziata quella che presenta il valore minore della funzione di errore scelta, nel nostro caso l'RMSE. Tra le opzioni di regressione utilizzate dall'interfaccia ci sono le regressioni lineari, gli alberi di regressione, la regressione gaussiana (GPR da Gaussian process regression), support vector machines (SVM), metodi kernel per l'approssimazione, metodi ensemble di alberi di regressione e reti neurali artificiali (ANN).

I modelli utilizzati sono elencati di seguito:

- Boosted e bagged trees[13]

  Sono entrambi derivati dal modello predittivo supervisionato detto albero decisionale o decision tree. Questo algoritmo è composto da nodi, rami e foglie, nei nodi i dati vengono

[13] https://it.mathworks.com/help/stats/fitrensemble.html

splittati. Il nodo rappresenta un "test", i rami rappresentano i risultati del test e le foglie invece sono le soluzioni intermedie o finali, ovvero la trasformazione finale dei dati. Il criterio di selezione in un albero decisionale di regressione è una misura dell'errore come ad esempio l'MSE. Unendo più di un albero decisionale si ottengono i cosiddetti ensemble trees; tra questi ci sono: i bagged trees, composti da più alberi decisionali in parallelo addestrati su sottoinsiemi indipendenti del training set, poi combinati nel modello finale e i boosted trees nei quali gli alberi sono organizzati in maniera sequenziale ottenendo in input il risultato dell'albero precedente. Mentre il primo metodo aumenta la robustezza del modello, riducendo la varianza e la possibilità di overfitting, il secondo ne aumenta la precisione riducendo l'errore.

- Gaussian Process Regression (GPR)[14]

Un processo gaussiano è un processo stocastico che prende come valori variabili aleatorie che hanno tra di loro una distribuzione congiunta gaussiana. Esso viene completamente descritto dalla funzione media m(X), spesso posta pari a zero e dalla funzione covarianza o kernel $k(x_i,x_j)$.[15] Gli indici i e j variano da 1 ad n e devono essere diversi tra di loro. La funzione kernel deve essere massima per x_i vicino a x_j e tendere a zero per x_i molto lontano da x_j. Dopo l'addestramento le funzioni di covarianza utilizzate per i modelli GPR sono state due:

$$Exponential\ kernel = k(x_i, x_j|\theta) = \sigma_f^2 \cdot exp\left(-\frac{r}{\sigma_l}\right)$$

$$Matern\ \frac{5}{2}\ kernel = \sigma_f^2 \cdot \left(1 + \frac{\sqrt{5}\cdot r}{\sigma_l} + \frac{5\cdot r^2}{\sigma_l^2}\right) \cdot exp\left(-\frac{\sqrt{5}\cdot r}{\sigma_l}\right)$$

Dove σ_f è la deviazione standard e σ_l è la scala di lunghezza caratteristica del segnale. Inoltre:

$$r = \sqrt{(x_i - x_j)^T (x_i - x_j)}$$

$$\theta_1 = log(\sigma_l)$$

[14] https://it.mathworks.com/help/stats/gaussian-process-regression-models.html

[15] Tosadori F. M., (2019). Gaussian Process Regression una tecnica di machine learning per il pricing veloce di derivati.

$$\theta_2 = log(\sigma_f)$$

- Support Vector Machine (SVM)[16]

  Anche questo tipo di machine learning utilizza funzioni di tipo kernel per effettuare una regressione tramite algoritmo supervisionato. Questo algoritmo impone una tolleranza ε tale per cui se i dati di target si discostano dalla funzione di regressione per una quantità maggiore di ε vengono considerati errori. L'algoritmo dunque cerca di minimizzare l'entità del quadrato di questi errori modificando i parametri interni.

Una volta aperta l'applicazione si inserisce la matrice contenente le variabili predittrici e il vettore contenente i target. L'applicazione effettua automaticamente la procedura di traininig e validazione secondo lo schema scelto e offre la possibilità di utilizzare una parte dei dati in ingresso per effettuare il test autonomamente sui dati indipendenti. Per l'obiettivo della tesi si è scelto di effettuare il test manualmente per poter comparare i risultati dei test di tutti i modelli sullo stesso set di dati, con le stesse modalità. Per la validazione si è scelto di utilizzare lo schema di cross-validation. Una volta scelto il numero k di *folds* (nel caso di studio sempre pari a 5) il procedimento prevede la divisione del set di dati di training in k parti, poi una alla volta ognuna delle parti è utilizzata per la validazione (set di validazione). Viene effettuato l'addestramento sui dati non appartenenti al set di validazione, valutandone poi l'efficienza calcolando l'RMSE sul set di validazione. Alla fine del procedimento il valore di RMSE di training fornito è rappresentato dalla media dei valori calcolati nelle k procedure di training. In questo modo il modello fornito dovrebbe essere protetto dal fenomeno di overfitting. Il procedimento di cross-validation genera un onere computazionale maggiore ma poiché il set di dati a disposizione non è eccessivamente grande questo non risulta un problema per la modellazione. Dopo aver impostato gli input e le impostazioni di validazione è stato effettuato il training in parallelo su tutti i modelli disponibili nell'applicazione. Una volta terminato il processo, il programma evidenzia il modello che ha generato l'errore quadratico medio minore (mediato sulle parti di training usate in validazione). Selezionando il modello è possibile generare una funzione interna a Matlab stesso che può essere salvata, la quale, riproduce il modello creato, addestrato sul set da noi scelto, su dati nuovi. Applicando quindi questa funzione al set dei dati di test è stato possibile stimare la producibilità per il periodo di test (2017-2022), che non è stato usato in nessun modo in fase di messa a punto dei modelli, e in seguito calcolarne gli indici per valutarne l'efficacia. Il processo di addestramento è stato

[16] https://it.mathworks.com/discovery/support-vector-machine.html

effettuato su tutti i gruppi, sia utilizzando in ingresso tutte le variabili esplicative, sia selezionando solo le variabili selezionate dalla stepwise regression effettuata in precedenza. Di seguito sono riportati i risultati ottenuti per entrambe le combinazioni di variabili in input (tabb. 4.C e 4.D) e i grafici relativi al gruppo 4 per il modello che ha permesso le migliori prestazioni utilizzando in ingresso tutte le variabili (figg. 4.8 e 4.9). Nelle tabelle sono indicati anche i modelli utilizzati dall'applicazione per effettuare la regressione.

	GRUPPO1	GRUPPO2	GRUPPO3	GRUPPO4	GRUPPO5	GRUPPO6
Modello	Boosted trees	Exponential GPR	Exponential GPR	Exponential GPR	Boosted trees	Matern 5/2 GPR
Variabili	TUTTE	TUTTE	TUTTE	TUTTE	TUTTE	TUTTE
rmse tr (-)	0.616	0.610	0.385	0.436	0.537	0.629
NSE tr	0.749	0.891	0.905	0.882	0.845	0.495
rmse test (-)	0.662	0.564	0.404	0.532	0.575	0.694
NSE test	0.555	0.619	0.809	0.713	0.683	0.524

Tab. 4.C - Indici di errore per la modellazione con il regression learner sui set di training (1990-2016) e test (2017-2022) con tutte le variabili in input.

	GRUPPO1	GRUPPO2	GRUPPO3	GRUPPO4	GRUPPO5	GRUPPO6
Modello	Bagged trees	Exponential GPR	Exponential GPR	Exponential GPR	Quadratic SVM	Quadratic SVM
Variabili	P-T-ET-SD	P-T-SF-ET-SD	P-T-SF-ET	P-T-SF-ET-SD	P-T-ET	P-SF
rmse tr (-)	0.601	0.610	0.396	0.436	0.519	0.698
NSE tr	0.732	0.891	0.906	0.882	0.628	0.166
rmse test (-)	0.617	0.564	0.413	0.532	0.579	0.803
NSE test	0.595	0.619	0.794	0.713	0.664	0.328

Tab. 4.D – Indici di errore per la modellazione con il regression learner sui set di training (1990-2016) e test (2017-2022) con le variabili selezionate dalla stepwise regression.

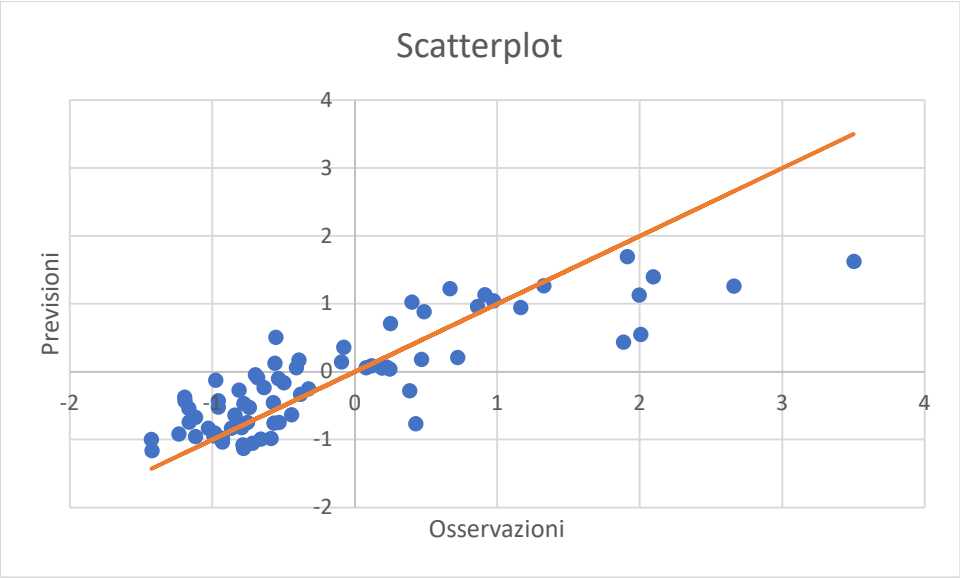


Fig. 4.8 – Scatterplot modello con modello exponential GPR sul test set (2017-2022) – Gruppo 4.

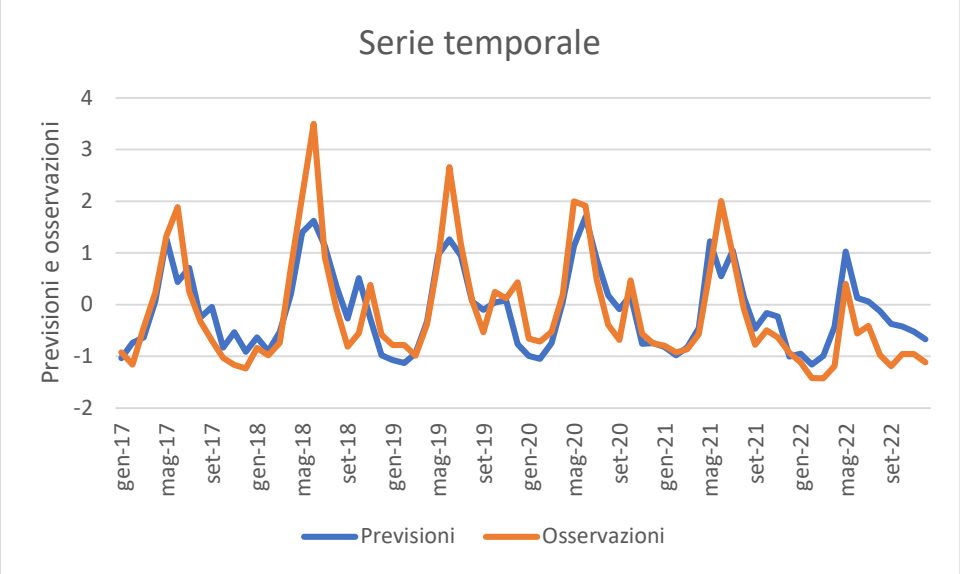


Fig. 4.9 – Serie temporale con modello exponential GPR sul test set (2017-2022) – Gruppo 4.

I risultati mostrano che utilizzando tutte le variabili si ottengono risultati migliori rispetto all'uso delle sole variabili selezionate dalla stepwise regression in tutti i gruppi tranne che per il primo. Inoltre gli indici mostrano prestazioni migliori rispetto ai modelli creati tramite stepwise regression.

5 – MODELLAZIONE TRAMITE LSTM

La modellazione effettuata finora prevedeva che in ingresso e in uscita dal modello ci fossero dati riferiti allo stesso intervallo temporale. I modelli utilizzati, dunque, sono di fatto privi di memoria. Il comportamento idrologico di un bacino e di conseguenza la producibilità idroelettrica, hanno invece una forte dipendenza dagli eventi meteorologici e dai processi idrologici avvenuti nei periodi precedenti. In particolare gli impianti a serbatoio hanno per definizione il pregio di poter immagazzinare i volumi idrici per poterli turbinare in periodi futuri. Anche la neve e le portate idriche sotterranee inoltre non generano volumi turbinabili consistenti nell'immediato ma consentono invece la produzione idroelettrica in periodi successivi al verificarsi della precipitazione. Per questi motivi, si è pensato di utilizzare un modello dotato di memoria rimanendo comunque in un ambito di modellazione data driven. L'evoluzione del machine learning ha reso affidabile e poco dispendioso dal punto di vista computazionale l'utilizzo del deep learning per la regressione. In particolare, tra le soluzioni di deep learning, la long short-term memory (LSTM) si distingue per la sua spiccata capacità di memorizzare le dipendenze a lungo termine.[17]

5.1 – Stato dell'arte sulle applicazioni di LSTM nella modellazione afflussi-deflussi

Gli algoritmi di tipo data driven sono ampiamente utilizzati per la realizzazione di modelli afflussi e deflussi e sono spesso caratterizzati da un'ottima affidabilità. Con lo sviluppo tecnologico e il miglioramento della capacità computazionale dei calcolatori si è passati a modelli sempre più complessi, passando per reti neurali artificiali e reti neurali ricorrenti. Recentemente si sono iniziate ad utilizzare per i modelli idrologici anche le reti di tipo LSTM. Per via della capacità di memorizzare dipendenze a lungo termine, queste reti sembrano particolarmente adatte a descrivere i comportamenti dei corsi d'acqua. Kratzert et al. nel 2018 hanno applicato per primi questo tipo di reti per la modellazione afflussi-deflussi.[18] Il database utilizzato è stato il CAMELS (Catchment Attributes for Large-Sample Studies), un data set di bacini americani per i quali i dati sono liberamente disponibili. Di questo dataset hanno

[17] Hochreiter, S., Schmidhuber, J. (1997). Long short-term memory.

[18] Kratzer et al., (2018), Rainfall-Runoff modelling using Long Short-Term Memory (LSTM) networks.

utilizzato 241 bacini suddivisi in 4 unità idrologiche. Sono state effettuati tre tipi di modellazione differenti. Il primo modello è stato messo a punto singolarmente su ciascun bacino, nel secondo esperimento si è implementato un unico modello per tutti i bacini di ciascuna unità idrologica e il terzo ha unito le informazioni dei primi due, addestrando il modello prima sull'unità idrologica e poi ricalibrandolo singolarmente per ciascun bacino. L'output da simulare è rappresentato dalla portata giornaliera e in input sono stati forniti precipitazione, temperatura minima e massima, radiazione solare e umidità. Per il training si sono utilizzati 15 anni di misure. L'architettura della rete è strutturata con un vettore di input di lunghezza n=365, due hidden layers da 20 unità e l'RMSE come funzione errore (loss). Nello studio questo tipo di reti ha fornito risultati paragonabili al modello concettuale utilizzato come metro di paragone (Sacramento Soil Moisture Accounting Model o SAC-SMA[19] associato al Snow-17 snow routine[20]), inoltre i modelli creati regionalmente sono stati adattati con successo sui singoli bacini, ricalibrando il modello con l'utilizzo di pochi dati. L'anno successivo, il gruppo di Kratzert (2019) ha ampliato lo studio aumentando il numero dei bacini utilizzati a 531 (sempre dal data base CAMELS).[21] Questa volta al modello con le cinque variabili meteorologiche in input si sono affiancati altri due modelli con altre 27 grandezze in input le quali caratterizzano il bacino (ad esempio area, pendenza media, indice di aridità). Queste nuove informazioni sono statiche, cioè costanti per ciascun bacino. I due modelli con le informazioni aggiuntive si differenziano in quanto uno è un modello che viene addestrato con la portata di target di tutti i bacini, mentre l'altro è un modello che dovrebbe riprodurre la portata per gruppi di bacini non strumentati, per cui viene addestrato su 12 gruppi casuali di bacini per poi essere testato sul tredicesimo gruppo di cui si ipotizza che non siano disponibili le portate osservate. Lo studio ha mostrato una migliore risposta del modello data driven rispetto ai modelli concettuali (SAC-SMA e U.S. National Water Model NWM)[22] sulla maggior parte dei bacini, inoltre il modello si è comportato meglio anche sui gruppi di bacini non strumentati. In particolare, l'aggiunta delle caratteristiche del bacino ha reso il modello più performante sui bacini non strumentati. Dopo Kratzert, altri hanno contribuito alla diffusione di informazioni

[19] Burnash f. j., Fernal R. L., (1973), A generalized streamflow simulation system: Conceptual modeling for digital computers.

[20] Anderson E., (2006). Snow Accumulation and Ablation Model – SNOW-17.

[21] Kratzert et al., (2019). Toward improved predictions in ungauged basins: Exploiting the power of machine learning.

[22] Salas et al., (2018). Towards real-time continental scale streamflow simulation in continuous and discrete space.

legate all'utilizzo di questo tipo di reti per la previsione di portate. Tian et al., nel 2019, hanno utilizzato dei modelli LSTM su due bacini di differenti dimensioni nel Sud-Est della Cina.[23] Anche in questo caso i dati erano giornalieri. Lo studio, in questo caso, ha comparato l'efficacia della rete LSTM rispetto al modello concettuale GR4J[24] e ad altri tipi di RNN. Lo studio inoltre, ha combinato il modello concettuale con la rete neurale. La LSTM è risultata il modello più performante e in generale i modelli ibridi hanno fornito risultati più soddisfacenti. In Europa, Lees et al. (2021) hanno replicato l'esperienza di Kratzert creando un modello unico regionale per la Gran Bretagna, utilizzando un data set equivalente per l'isola Britannica (CAMELS-GB) con un data base di 669 bacini.[25] Anche in questo caso la rete LSTM performava meglio del modello utilizzato come benchmark (4 modelli dal FUSE framework).[26] Sebbene i dati giornalieri siano quelli più utilizzati in letteratura scientifica per l'argomento in esame, è interessante analizzare lo studio di Song et al. del 2019.[27] Lo scopo del modello in questo caso era di prevedere le piene improvvise (flash floods). L'area di studio è stata il bacino dell'Anhe (251 kmq) nella Cina sudorientale. Avendo uno scopo differente, i dati sono stati classificati e organizzati per piene. Al modello sono stati quindi forniti solo i dati relativi agli eventi di piena, partendo dalle 5 ore precedenti il picco di portata, con l'intento di prevedere fino alle 10 ore successive all'istante di previsione. Nonostante i pochi dati usati per il training (45 piene di training, 15 di validazione e 15 di test), i risultati sono stati soddisfacenti, in particolare per piene importanti. Alcuni studi hanno invece analizzato l'efficacia dei modelli addestrati su set di dati meteorologici ottenuti tramite osservazioni satellitari, in particolare, Ouma et al. (2021)[28] hanno modellato su scala temporale mensile per il bacino del fiume Nzoia nel Kenya occidentale, mentre Choi et al. (2022) [29] su scala temporale giornaliera su 13 bacini Sud

[23] Tian et al., (2018). Integration of a parsimonious hydrological model with recurrent neural networks for improved streamflow forecasting.

[24] Perrin et al., (2001). Does a large number of parameters enhance model performance? Comparative assessment of common catchment model structures on 429 catchments.

[25] Lees et al., (2021). Benchmarking data-driven rainfall–runoff models in Great Britain: a comparison of long short-term memory (LSTM)-based models with four lumped conceptual models.

[26] Clark, M. P. et al., (2008). Framework for Understanding Structural Errors (FUSE): A modular framework to diagnose differences between hydrological models.

[27] Song et al., (2019). Flash flood forecasting based on long short-term memory networks.

[28] Ouma et al., (2021). Rainfall and runoff time-series trend analysis using LSTM recurrent neural network and wavelet neural network with satellite-based meteorological data: case study of Nzoia hydrologic basin.

[29] Choi et al., (2022). Utilization of the Long Short-Term Memory network for predicting streamflow in ungauged basins in Korea.

Coreani con l'intento di applicazione su bacini non strumentati (quindi con addestramento su tutti i bacini ad esclusione del k-esimo sul quale viene effettuato il test). In entrambi i casi si sono ottenuti buoni risultati per le reti LSTM anche se lo studio coreano ha mostrato la dipendenza dell'algoritmo da una mole notevole di dati, motivo per cui l'applicazione a bacini non strumentati potrebbe essere di difficile attuazione. Recentemente Frame et al. (2022) hanno verificato l'efficacia della LSTM per eventi estremi o fuori dal campione di training, inoltre, nello stesso studio hanno testato una rete nella quale sono stati inseriti dei limiti fisicamente basati (conservazione della massa).[30] Il data set e l'impostazione ricalca quella di Kratzert et al. del 2019. L'efficacia delle reti è stata confermata anche in questo caso però l'imposizione dei limiti fisici ha compromesso i risultati del modello. Lo studio di Majeske et al. del 2021 ha portato due ulteriori novità nel campo delle previsioni idrologiche tramite reti neurali.[31] Le reti utilizzate sono infatti di tipo BLSTM ovvero reti LSTM bidirezionali. In ingresso gli intervalli temporali sono stati ridotti, andando ad inserire nella rete solo informazioni significative, inoltre in uscita al modello non c'è soltanto la portata ma anche il soil water content. L'esperimento, condotto sul bacino del fiume Wabash nel Midwest degli Stati Uniti, ha mostrato che, nonostante la riduzione degli intervalli temporali, il modello performa allo stesso livello di modelli più complessi dello stesso tipo, inoltre è stata mostrata la possibilità di riutilizzo del modello in altri bacini per prevedere la portata anche se per il soil water content non si sono ottenuti risultati altrettanto convincenti. Uno degli ultimi studi sull'argomento è quello effettuato da Clark et al. su 496 bacini australiani.[32] È stato creato un modello per ciascuno dei 496 bacini utilizzati. La sfida dello studio è dovuta al fatto che il periodo di training è molto più umido del periodo di test. La stima delle portate della rete LSTM però è sempre alla pari o migliore del modello concettuale usato come benchmark (WAPABA[33]). Nella tabella seguente (tab 5.A) sono riportati sinteticamente i dati principali degli studi analizzati in questo paragrafo.

[30] Frame et al., (2022). Deep learning rainfall–runoff predictions of extreme events.

[31] Majeske et al., (2021). Inductive predictions of extreme hydrologic events in the wabash river watershed.

[32] Clark et al., (2023). Deep learning for monthly rainfall-runoff modelling: a comparison with classical rainfall-runoff modelling across Australia.

[33] Wang et al., (2011). Monthly versus daily water balance models in simulating monthly runoff.

	Kratzert et al. (2018)	Kratzert et al. (2019)	Tian et al. (2018)	Lees et al. (2021)	Song et al. (2019)
Area di indagine	241 bacini CAMELS (USA) raggruppati in 4 unità idrologiche	531 bacini CAMELS (USA) raggruppati in unità idrologiche	Bacino del fiume Xiangjiang (81600 kmq) e del fiume Quijiang (5290 kmq)	669 bacini CAMELS (GB) raggruppati in una unità idrologica	Bacino del fiume Anhe (251 kmq)
Variabili in input	• Precipitazione • Temperatura minima • Temperatura massima • Radiazione ad onda corta • Umidità	• Precipitazione • Temperatura minima • Temperatura massima • Radiazione solare incidente • Pressione di vapore • 27 caratteristiche del bacino	• Precipitazione • Evapotraspirazione potenziale	• Precipitazione • Temperatura minima • Temperatura massima • Radiazione solare incidente • Pressione di vapore • 22 caratteristiche del bacino	• Precipitazione
Informazioni temporali	• Scala giornaliera • 15 anni di training	• Scala giornaliera • 15 anni di training • 15 anni di test	• Scala giornaliera • 15 anni di training • 10 anni di validazione • 5 anni di test	• Scala giornaliera • 9 anni di training • 10 anni di test	• Scala oraria • 45 piene di training • 15 piene di validazione • 15 piene di test
Architettura della rete neurale	• 2 layers da 20 HU • LF = RMSE • Input n=365	• 1 layer da 256 HU	• 1 layer da 50 HU • LF = RMSE • Adam solver	• 1 layer da 64 HU • 30 epoche • Input n=365	• 1 layer da 5 HU • LF = MAE
	Ouma et al. (2021)	**Choi et al. (2022)**	**Frame et al. (2022)**	**Majeske et al. (2021)**	**Clark et al. (2023)**
Area di indagine	Bacino del fiume Nzoia (12700 kmq)	13 bacini in Corea del Sud	498 bacini CAMELS (USA) raggruppati in unità idrologiche	Bacino del fiume Wabash	496 bacini in Australia
Variabili in input	• Precipitazione • Temperatura • Radiazione solare	• Precipitazione • Temperatura minima • Temperatura massima • Velocità del vento • Umidità relativa • Temperatura di rugiada • Informazioni da satellite • Caratteristiche del bacino	• Precipitazione • Temperatura minima • Temperatura massima • Radiazione solare incidente • Pressione di vapore • 27 caratteristiche del bacino	• Precipitazione • Temperatura minima • Temperatura massima	• Precipitazione • Temperatura • Evapotraspirazione
Informazioni temporali	• Scala mensile • 30 anni di training	• Scala giornaliera • 4 anni di training	• Scala giornaliera • 15 anni di training • 15 anni di test	• Scala giornaliera • 84 anni di training	• Scala mensile • 35/70 anni a seconda del bacino divisi tra training e test
Architettura della rete neurale	• 4 layers da 30 HU	• 2 layer da 30 HU • Input n=365	• 1 layer da 64 HU • 1 layer da 128 HU • 30 epoche • Adam solver	• 1 layer da 64 HU • 30 epoche • Input n=365	• 1 layer da 10 HU • 100 epoche • Input n=6

Tab. 5.A – Sintesi articoli su modelli A/D basati su reti LSTM.

5.2 – Reti neurali ricorrenti di tipo Long Short-Term Memory

Una rete neurale artificiale (Artificial Neural Network ANN) è un modello matematico creato a somiglianza delle reti neurali biologiche. Queste reti sono ampiamente utilizzate nel contesto delle intelligenze artificiali e per migliorarne la qualità sono addestrate tramite il machine learning. La rete è costituita da nodi detti neuroni suddivisi in tre tipi di livelli diversi, detti layers. Ogni rete ha un layer di input attraverso il quale le informazioni entrano nella rete sotto forma di un vettore numerico, uno o più livelli nascosti nei quali l'informazione viene processata (hidden layers) e trasmessa infine al livello finale che è rappresentato dal layer di output il quale fornisce il risultato dell'elaborazione della rete neurale. Lo schema generale della rete neurale di tipo feedforward con un singolo livello nascosto è rappresentato in figura 5.1.

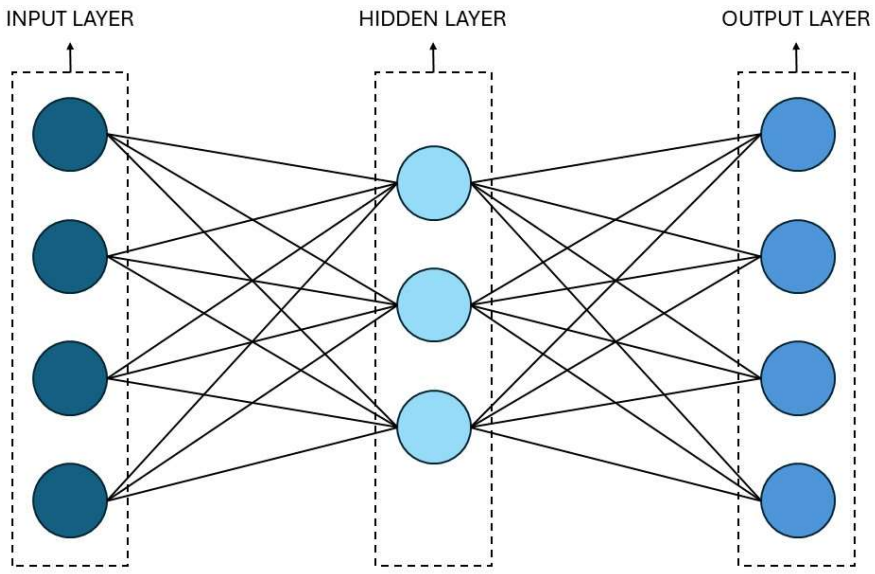


Fig. 5.1 – Schema rete neurale artificiale.

Le informazioni x_i contenute nell'input layer vengono moltiplicate per i pesi w_i associati a ciascuna connessione, poi i valori in ingresso ai layer nascosti vengono sommati. All'interno dei nodi degli hidden layer i valori vengono trasformati tramite una funzione soglia, spesso nella forma di una sigmoide:

$$f(x) = \frac{1}{1 + e^{-x}} \qquad 0 < f(x) < 1$$

Originando un valore dato da:

$$O_j = \frac{1}{1 + e^{-\sum w_i \cdot x_i}}$$

Questo procedimento è ripetuto passando attraverso le connessioni per tutti gli hidden layer fino a esplicitare la risposta della rete nell'output layer.[34] Le reti neurali impostate in questo modo non hanno comunque modo di conservare una memoria. Per sopperire a questo problema sono state proposte le reti neurali ricorrenti (Recurrent Neural Network RNN). Esse sono caratterizzate da un ciclo al loro interno che permette la persistenza dell'informazione. L'output all'n-simo intervallo temporale è determinato dall'elaborazione degli input associati all'intervallo temporale n stesso e di tutti quelli relativi agli intervalli temporali precedenti.

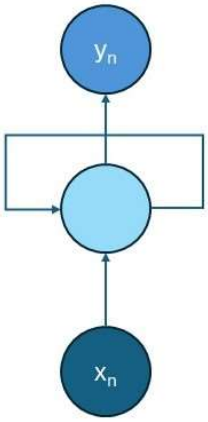


Fig. 5.2 – Schema generico rete neurale ricorrente.

Se il ciclo venisse "srotolato" si vedrebbe come ogni istante precedente contribuisce al successivo. Il risultato dell'ultimo layer ricorrente e dell'ultimo intervallo temporale arrivano in un layer denso (dense layer) che effettua l'elaborazione finale per fornire l'output finale per l'intervallo temporale n-simo.

[34] Minns A. W., Hall M. J., (1996). Artificial neural networks as rainfall-runoff models

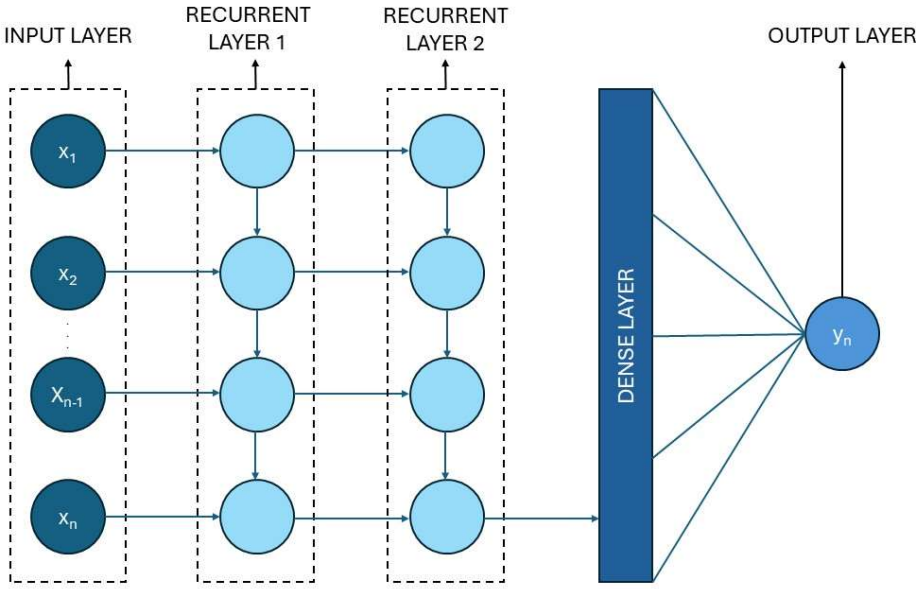


Fig. 5.3 – Schema rete neurale ricorrente esplicitando il ciclo.

In teoria la struttura di questo tipo di rete neurale dovrebbe essere in grado di mantenere una memoria a lungo termine, in realtà studi come quelli riportati da Bengio et al. nel 1994[35] hanno dimostrato che le RNN hanno difficoltà a ricordare sequenze di lunghezza maggiore di 10. Per sopperire al problema della memoria a lungo termine sono state create le Long Short-Term Memory networks, che per semplicità d'ora in poi saranno chiamate LSTM. Esse sono un tipo di RNN e differiscono dalle RNN tradizionali per le operazioni svolte all'interno del layer ricorrente. In una RNN tradizionale esiste un solo tipo di stato, ovvero lo interno o nascosto (hidden state h_t) che è ricalcolato ad ogni intervallo temporale.

$$h_t = g(W \cdot x_t + U \cdot h_{t-1} + b)$$

Dove g è la funzione di attivazione, spesso una tangente iperbolica, W e U sono matrici di pesi regolabili relativi rispettivamente al vettore di input x e allo stato interno h, mentre b è il vettore di bias, anch'esso regolabile. Lo stato nascosto inizialmente è posto uguale a un vettore nullo, la cui lunghezza è un iperparametro deciso dall'utente. Il confronto tra operazioni interne in una RNN e in una LSTM è illustrato nelle figure seguenti (figg. 5.4 e 5.5).

[35] Bengio et al., (1994). Learning long-term dependencies with gradient descent is difficult.

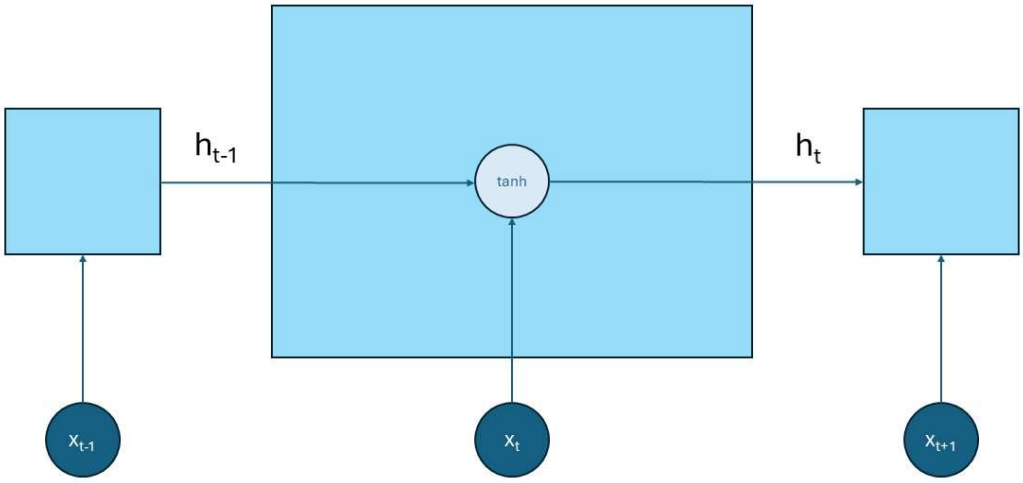


Fig. 5.4 – Schema operazioni interne RNN.

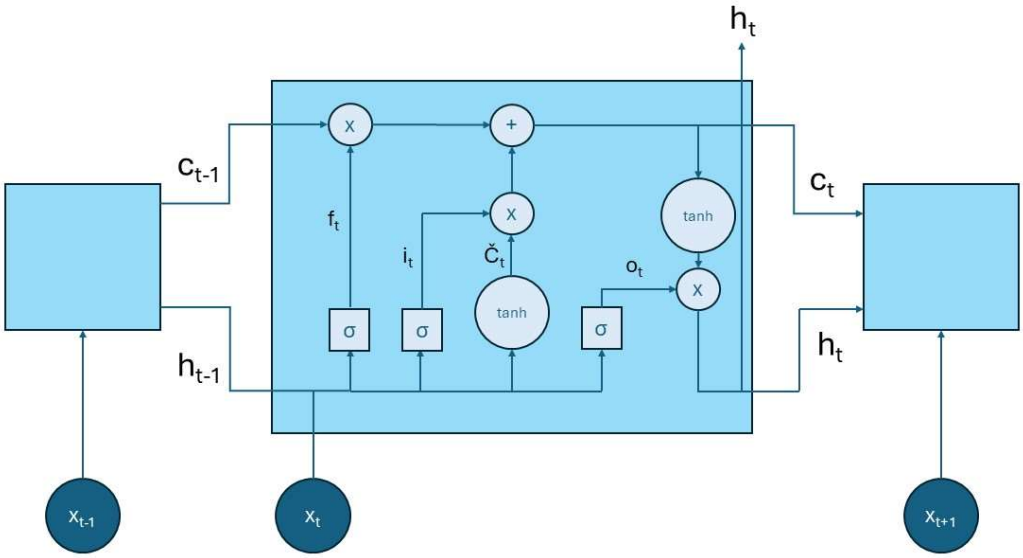


Fig. 5.5 – Schema operazioni interne LSTM. Il + corrisponde alla somma, la X alla moltiplicazione elemento per elemento, tanh è la tangente iperbolica. f, i e o sono i vettori in uscita rispettivamente dal forget, input e output gate. h è l'hidden state mentre c è il cell state.

La prima differenza sostanziale tra i due schemi è la presenza di uno stato interno in più detto stato di cella (c_t). Attraverso lo stato di cella l'informazione passa attraverso il ciclo in maniera diretta con qualche operazione lineare che permette di filtrare i dati di passaggio. Questo processo è quello fondamentale per garantire la memoria a lungo termine della rete neurale. Lo stato di cella è un vettore della stessa lunghezza dell'hidden state e anch'esso è inizializzato con un vettore di soli zeri. Nelle LSTM l'informazione passa attraverso dei gate nei quali essa viene controllata e selezionata. In molte delle formule che interessano i processi interni delle LSTM saranno presenti le matrici dei pesi per il vettore x e per il vettore h e il vettore dei bias b, tutti riferiti al gate nel quale vengono utilizzati. Questi parametri sono regolabili e differenti per ciascun gate. Il primo passaggio è il forget gate attraverso cui si decide quali informazioni del cell state al tempo t-1 verranno dimenticate. L'informazione in arrivo attraverso l'hidden state al tempo t-1 e attraverso l'input x al tempo t viene elaborata tramite una sigmoide (σ) per ottenere un vettore con elementi compresi tra 0 e 1.

$$f_t = \sigma\left(W_f \cdot x_t + U_f \cdot h_{t-1} + b_f\right)$$

Il passo successivo è la creazione di un vettore di aggiornamento potenziale per lo stato di cella, utilizzando l'informazione in arrivo dallo stato nascosto e dall'input dell'intervallo temporale attuale:

$$\check{C}_t = tanh(W_{\check{C}} \cdot x_t + U_{\check{C}} \cdot h_{t-1} + b_{\check{C}})$$

Essa è compresa tra -1 e 1. In seguito l'input gate decide quale informazione di $\check{C}_t$ debba essere usata per aggiornare lo stato di cella attraverso la funzione:

$$i_t = \sigma(W_i \cdot x_t + U_i \cdot h_{t-1} + b_i)$$

Che fornisce un vettore di valori compresi tra 0 e 1. Lo stato di cella dunque viene aggiornato attraverso la relazione:

$$c_t = f_t \odot c_{t-1} + i_t \odot \check{C}_t$$

Il simbolo ⊙ rappresenta in questo caso una moltiplicazione elemento per elemento. L'ultimo gate è detto di output e controlla quali informazioni dello stato di cella relativo al tempo t verranno trasmesse all'hidden state. Anch'esso varia tra 0 e 1 e si calcola come:

$$o_t = \sigma(W_o \cdot x_t + U_o \cdot h_{t-1} + b_o)$$

Infine l'hidden state da tramettere al ciclo successivo è calcolato come:

$h_t = \tanh(c_t) \odot o_t$

L'informazione quindi viene modificata passaggio per passaggio come in una RNN ma, attraverso il cell state, parte dell'informazione viaggia indisturbata per tutto il ciclo, subendo solo modifiche tramite operazioni lineari semplici.[36]Le reti LSTM sembrano quindi il compromesso ideale per la creazione di un modello data driven con memoria a lungo termine. L'architettura della rete e la scelta degli iperparametri dipendono dal caso di studio, non esistono formule a priori, per cui si è deciso di procedere con un approccio di tipo trial and error, scegliendo i parametri in base alla migliore risposta del modello in termini di indicatori di errore sul primo modello testato (vd sez. 5.3).

5.3 – Modelli addestrati separatamente per ciascun gruppo e scelta degli iperparametri.

Il primo passo è stata la creazione di un modello per ogni gruppo di bacini. Si è scelto quindi di effettuare il training della rete neurale sugli stessi dati utilizzati per la creazione dei modelli esposti in precedenza (vedi sz. 4.3 e 4.4). Per il training si è utilizzato nuovamente il software Matlab, il quale attraverso i toolbox di deep learning fornisce la possibilità di un addestramento automatico della rete neurale fornendo in ingresso le caratteristiche della stessa. Innanzitutto si è dovuto cambiare il formato dei dati. Se fino ad ora in ingresso per il training si aveva una matrice *Nxm* con *N* numero di record (uno per ogni istante temporale da simulare, per noi pari a tutti i mesi del training set, quindi 324) ed *m* numero di variabili esplicative (5 per noi, corrispondenti alle 5 variabili meteo-idrologiche di forcing), ora occorre che in input ci sia un'unità denominata cell, ovvero un vettore di lunghezza *N* in cui ogni elemento è composto da una matrice di *m* righe ed *n* colonne, dove n rappresenta gli istanti temporali forniti come memoria al modello. Uno schema di impostazione di una matrice in input al tempo i-esimo è riportato in fig. 5.6.

[36] Kratzert, et al., (2018). op. cit.

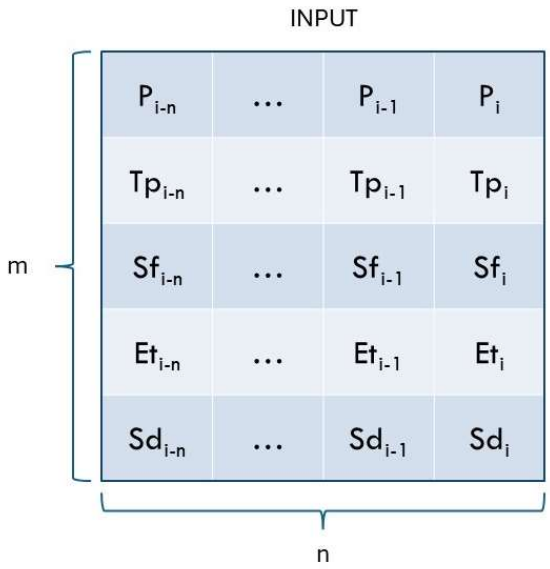


Fig. 5.6 – Schema di un'unità della cella in input a modello LSTM al tempo i-esimo.

La figura seguente (fig. 5.7) mostra invece schematicamente la struttura dei dati in ingresso e in uscita dell'intero modello per una memoria *n* pari a 3 mesi. Gli indici *i* rappresentano gli intervalli temporali e variano da 1 a *N-n*. Le *Y* sono i valori delle grandezze in output. La colonna n-sima di ogni cella in input contiene le variabili esplicative dello stesso mese che si ottiene in output.

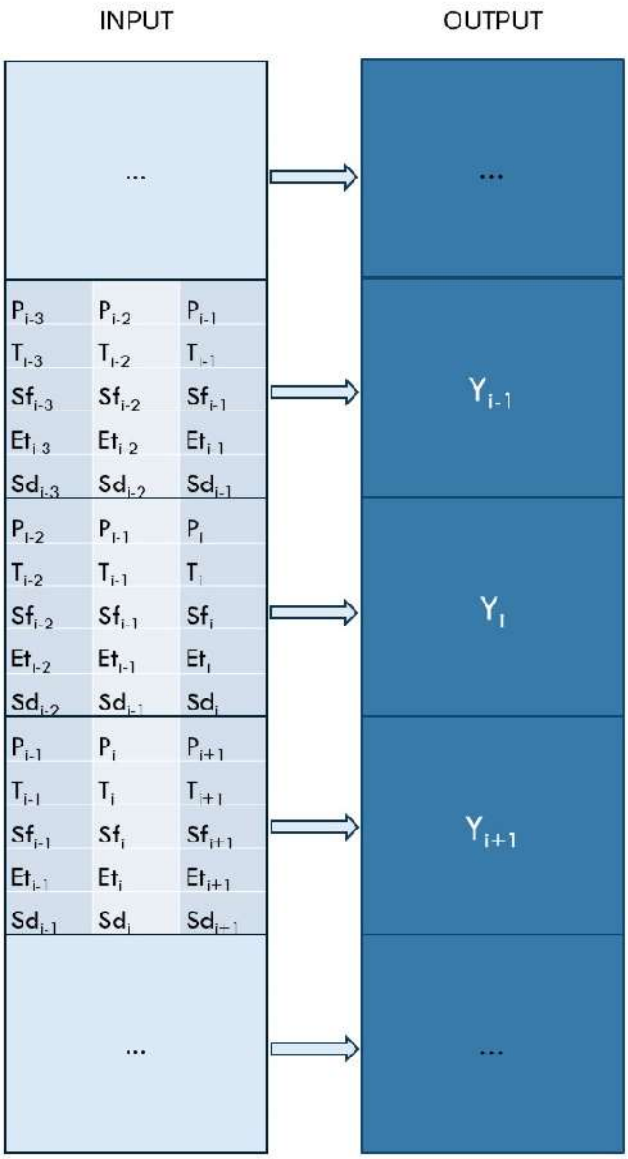


Fig. 5.7 – Schema modelli LSTM con memoria n pari a 3 mesi.

Questo tipo di struttura è stata creata su Matlab in modo da poter automatizzare il processo cambiando di volta in volta solo il gruppo sul quale si vuole effettuare la previsione e il numero di mesi di memoria da fornire al modello. Il processo è stato svolto per tutti i gruppi fornendo (per ogni record) in ingresso i valori delle variabili meteo corrispondenti un numero di mesi precedenti l'istante di previsione (memoria), variabile da 1 a 24. In generale l'architettura della rete è stata scelta dopo un processo di tipo trial and error in cui sono stati fatti variare gli iperparametri caratterizzanti la rete. Gli iperparametri scelti sono stati quelli che hanno fornito i risultati migliori in termini di indicatori di errore sul set di test del gruppo 1. Alla rete si sono

assegnate il numero di variabili esplicative, nel nostro caso 5, il numero di hidden layers, nel nostro caso pari a 1 e di unità nascoste (hidden units), assunto pari a 128, che rappresenta la lunghezza del vettore di hidden state e di conseguenza la quantità di informazione memorizzabile tra un intervallo temporale ed il successivo. Il valore di questi parametri non è determinante nel risultato ma se troppo basso la rete avrebbe difficoltà a ottenere dei risultati ottimali mentre se troppo grande si rischia di avere overfitting. La scelta del numero di hidden units e hidden layers è stata effettuata testando la rete con vari valori utilizzati nella letteratura scientifica consultata utilizzando la coppia di parametri che forniva i migliori risultati in test. Sono state testate reti con un hidden layer con hidden cells variabili da 32 a 254 (per multipli di due), poi si è provato ad utilizzare due hidden layer prima con 20, poi con 30 hidden units in ciascuno di essi. Per evitare l'overfitting si è provato ad aggiungere un dropout, il quale, durante il training, imposta a zero il valore di una percentuale di neuroni, nel nostro caso si è utilizzato il 10% e il 40% come in letteratura (Kratzert et al. 2018-2019) e del 20% (default di Matlab). In questo modo si forza l'algoritmo a trovare rapporti tra le variabili in maniera più robusta.[37] Il dropout è stato rimosso in seguito al processo di trial and error poiché la rete forniva indicatori di errore migliori sui dati indipendenti usati per il test senza di esso. Si inserisce in seguito la dimensione del layer in output, per noi pari a 1 (la producibilità all'istante desiderato) e si imposta il layer per la regressione (comando regressionlayer). In seguito si stabilisce la dimensione dei mini-batch da utilizzare per ogni iterazione dell'addestramento. Il mini-batch è un sottoinsieme del set di training che viene utilizzato per il calcolo del gradiente della funzione perdita e per aggiornare i pesi. Un'iterazione è un passo intrapreso dall'algoritmo di discesa del gradiente verso la minimizzazione della funzione di perdita usando un mini-batch. Dunque ad ogni iterazione l'algoritmo viene addestrato sui dati contenuti nel mini-batch, si aggiornano i parametri e si passa al mini-batch successivo. Si è deciso di utilizzare mini-batch di 27 elementi come di default sul software, considerando anche che non si discostava troppo dai valori utilizzati in letteratura. Si sceglie poi il numero massimo di epoche. Un'epoca è il passaggio completo dell'algoritmo di addestramento lungo l'intero set di addestramento. Il numero massimo di epoche si è stabilito inizialmente pari a 30 come in letteratura, poi è stato aumentato a 40 migliorando la qualità del modello ma ci si è resi conto che con l'aumentare della memoria cresceva anche la mole di dati in input per il training, per cui, a parità di epoche, la rete tendeva all'overfitting sui set con maggiori informazioni in ingresso, restituendo indici notevolmente migliori per il set di training rispetto a quelli del set di test. Il numero massimo di epoche si è

[37] Srivastava et al., (2014). Dropout: a simple way to prevent neural networks from overfitting.

quindi posto pari al valore fisso di 40 al quale si è sottratto n, in modo da ridurne il valore all'aumentare della memoria. In figura 5.8 sono riportati gli indicatori di errore per il gruppo 1 al variare della memoria per numero di epoche pari a 40 e pari a 40-n.

TRAINING

n	ME=40	ME=(40-n)
6	0.907	0.900
12	0.942	0.926
15	0.947	0.930
18	0.943	0.929
21	0.954	0.933

TEST

n	ME=40	ME=(40-n)
6	0.895	0.909
12	0.898	0.900
15	0.898	0.924
18	0.869	0.879
21	0.877	0.881

Fig. 5.8 – Coefficiente di Nash-Sutcliffe per set di training (1990-2016) e di test (2017-2022) al variare della memoria del modello (n) per 40 e 40-n epoche massime (ME).

Infine, si sceglie il risolutore per l'addestramento che, nel caso in esame, è di tipo a*dam* (adaptive moment estimation).[38] Per quanto riguarda le funzioni di attivazioni di stato e per i gate si sono mantenute quelle di default, ovvero, rispettivamente tangente iperbolica e sigmoide. Gli stati nascosti e di cella iniziali sono stati impostati con le caratteristiche di default. Nella tabella sottostante (5.B) sono riportate le caratteristiche della rete per l'addestramento.

[38] Kingma D. P., Ba J., (2014). Adam: A method for stochastic optimization.

Numero di varibili esplicative (m)	5
Numero di hidden units (HU)	128
Numero di output	1
Epoche massime	40-n
Dimensioni dei mini-batch	27
Risolutore	adam
Funzione di attivazione (gate)	tanh
Funzione di attivazione (state)	σ
Memoria (n)	Variabile da 1 a 24

Tab. 5.B – Impostazione rete LSTM.

Al termine del processo di training e di test si sono raccolti i risultati ottenuti. Gli indicatori utilizzati per il test sono risultati ottimali per un numero di mesi di input intermedio, come si può notare dalla figura seguente (fig. 5.9) che mette in relazione il coefficiente di efficienza di Nash-Sutcliffe con il numero di mesi in input per il gruppo 2.

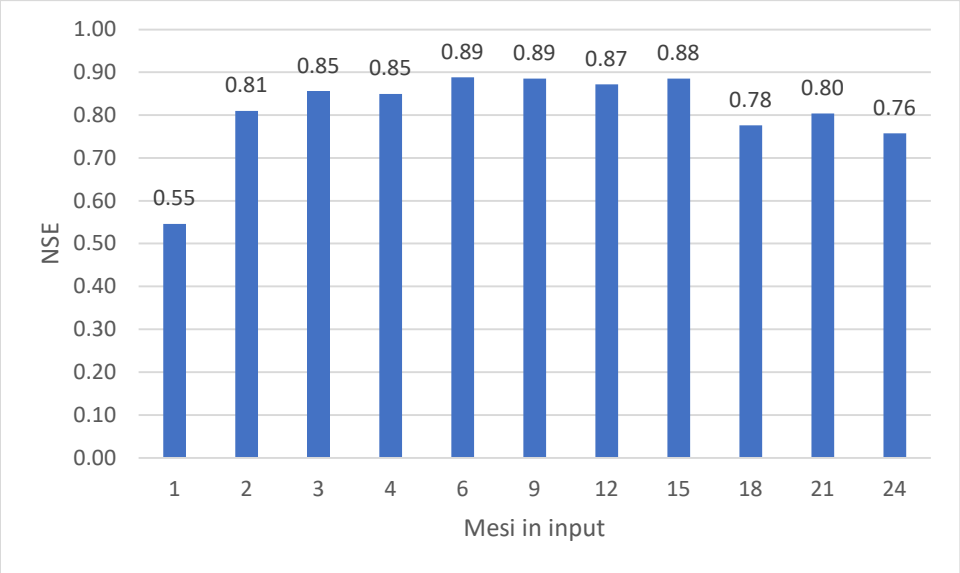


Fig. 5.9 – Andamento dell'NSE sul test set al variare della memoria n (numero di mesi precedenti forniti in input) per il gruppo 2.

La tabella seguente (5.C) mostra i risultati per ciascun gruppo. Si è deciso di riportare solo i risultati per il numero di mesi di memoria che ha permesso le migliori prestazioni in testing. Sono inoltre riportati, a titolo di esempio lo scatterplot e la serie temporale per il gruppo 4 (bacini padani) con memoria pari a 12 mesi (figg.5.10 e 5.11). In appendice C.2 sono riportate le serie temporali per il periodo di test per tutti i gruppi per il modello con la memoria che ha garantito indici di errore migliori.

	GRUPPO1	GRUPPO2	GRUPPO3	GRUPPO4	GRUPPO5	GRUPPO6
Mesi in input	15	6	6	12	4	15
rmse tr (-)	0.264	0.264	0.265	0.202	0.353	0.332
NSE tr	0.930	0.931	0.929	0.959	0.876	0.889
rmse test (-)	0.305	0.373	0.317	0.351	0.317	0.477
NSE test	0.924	0.888	0.897	0.892	0.894	0.788

Tab. 5.C – Indici di errore per la modellazione tramite LSTM sui set di training (1990-2016) e test (2017-2022) per modelli addestrati separatamente per ciascun gruppo.

.

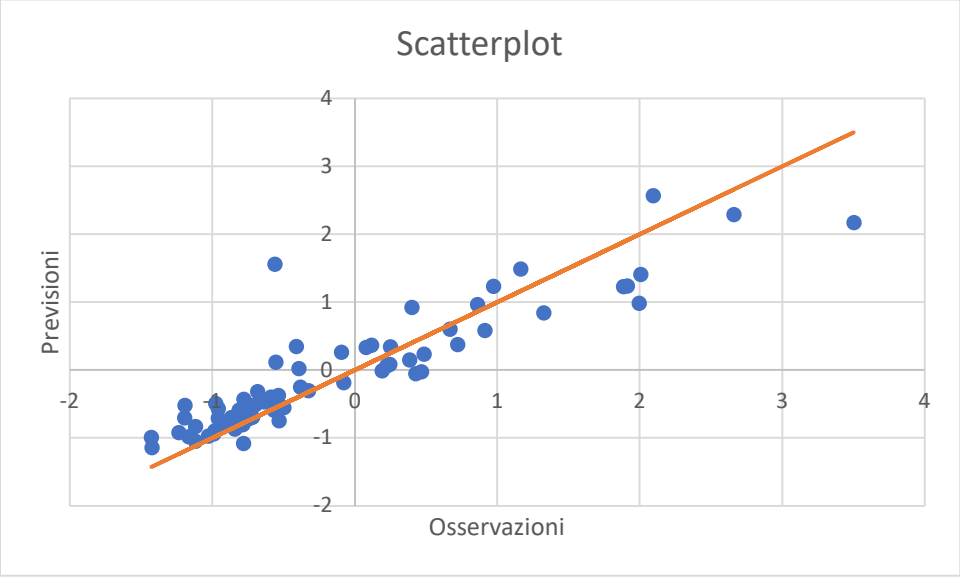


Fig. 5.10 – Scatterplot modello tramite LSTM con n = 12 mesi sul test set (2017-2022) – Gruppo 4.

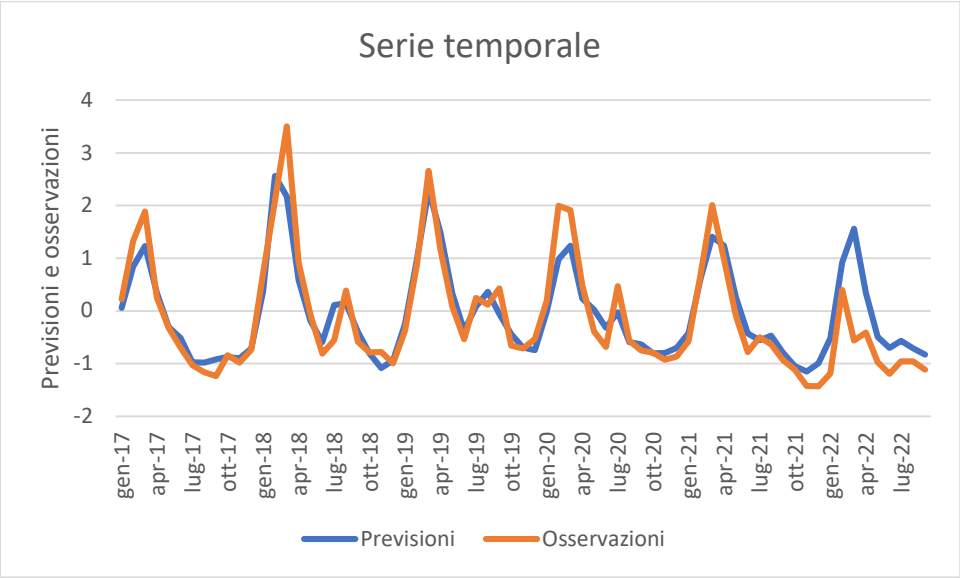


Fig. 5.11 – Serie temporale modello tramite LSTM con n = 12 mesi sul test set (2017-2022) – Gruppo 4.

5.4 – Modelli addestrati separatamente per ciascun gruppo (memoria unitaria)

È interessante confrontare il risultato del modello per memoria $n = 1$ con i risultati dei modelli di regressione effettuati in precedenza (vd sezz. 4.3 e 4.4) . La rete con in memoria un singolo mese infatti ha una struttura di input e output paragonabile a quella dei modelli di regressione, anch'essi relativi a una modellazione in cui le variabili in input sono fornite solo in corrispondenza dello stesso istante di simulazione del target. Nella tabella e nella figura seguente (tab 5.D, figg. 5.12 e 5.13), sono stati messi a confronto gli indicatori ottenuti per la stepwise regression, per il regression learner con i modelli migliori, e per la rete LSTM descritta in questo paragrafo sia per memoria unitaria che per la memoria ottimale. Come si può notare la prestazione della rete LSTM con memoria unitaria è circa paragonabile ai modelli di regressione precedenti, la rete LSTM con memoria ottimale invece permette risultati nettamente superiori, soprattutto sul set indipendente. Si è dunque deciso di utilizzare questo modello per effettuare le previsioni future di producibilità.

NSE TRAINING	GRUPPO1	GRUPPO2	GRUPPO3	GRUPPO4	GRUPPO5	GRUPPO6
Stepwise regression	0.577	0.509	0.821	0.713	0.644	0.492
Regression learner	0.749	0.891	0.905	0.882	0.845	0.495
LSTM n = 1	0.684	0.634	0.831	0.763	0.767	0.575
LSTM n migliore	0.930	0.931	0.929	0.959	0.876	0.890
Benchmark	0.390	0.580	0.799	0.793	0.550	0.315
NSE TEST	**GRUPPO1**	**GRUPPO2**	**GRUPPO3**	**GRUPPO4**	**GRUPPO5**	**GRUPPO6**
Stepwise regression	0.555	0.470	0.779	0.694	0.608	0.312
Regression learner	0.555	0.619	0.809	0.713	0.683	0.524
LSTM n = 1	0.672	0.546	0.766	0.723	0.705	0.414
LSTM n migliore	0.924	0.888	0.897	0.892	0.894	0.788
Benchmark	0.388	0.621	0.788	0.635	0.616	0.311

Tab. 5.D – Confronto tra coefficienti di efficienza di Nash-Sutcliffe per la modellazione tramite stepwise regression, regression learner, modello LSTM sui set di training e benchmark (1990-2016) e test (2017-2022).

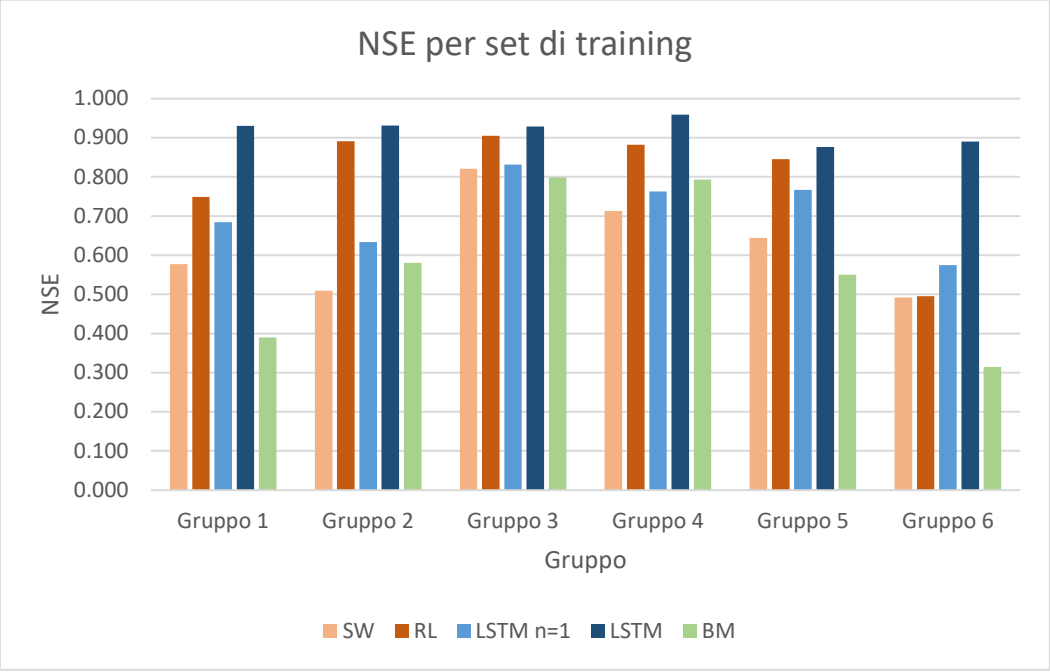


Fig. 5.12 – Confronto tra coefficienti di efficienza di Nash-Sutcliffe per la modellazione tramite stepwise regression(SW), regression learner (RL), modelli LSTM e modello benchmark (BM)sui set di test (2017-2022).

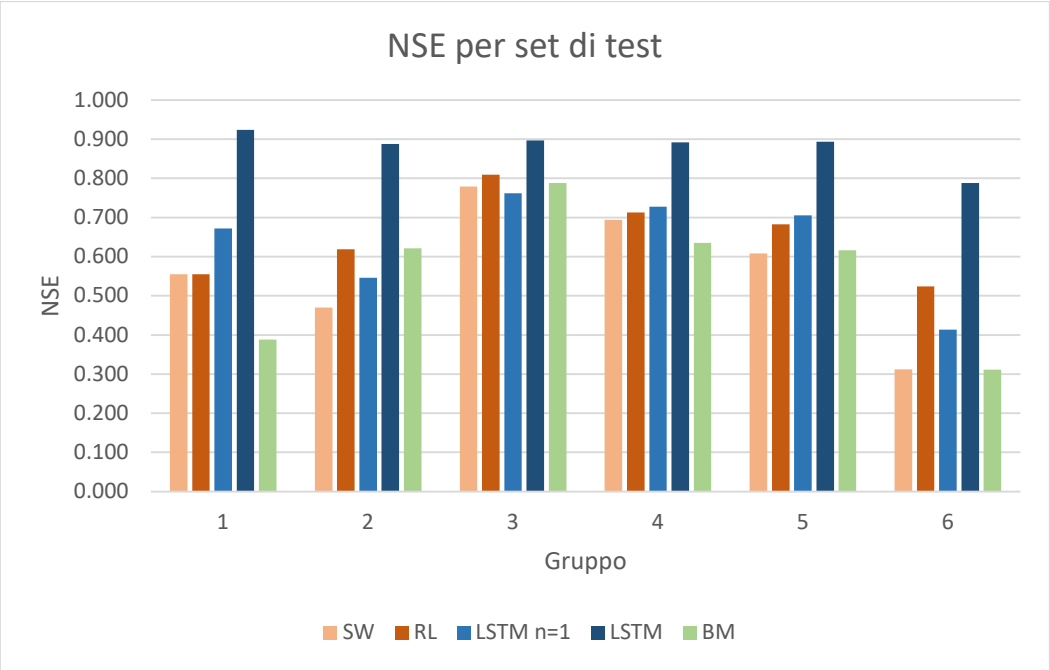


Fig. 5.13 – Confronto tra coefficienti di efficienza di Nash-Sutcliffe per la modellazione tramite stepwise regression(SW), regression learner (RL), modelli LSTM e modello benchmark (BM)sui set di test (2017-2022).

5.5 – Modello unico per tutti i gruppi

Come spiegato sopra, per ognuno dei 6 gruppi di bacini è stato messo a punto ("trained") un modello diverso, seppure con la stessa architettura e le stesse impostazioni (vd Tab 5.B) e facendo poi variare solo l'estensione della memoria, che viene identificata con un'analisi trial-and-error separata per ogni gruppo.

Per provare a semplificare la modellazione, è stato anche effettuato un tentativo di stima della producibilità mettendo a punto un unico modello per tutti e sei i gruppi, inserendo come input del training della rete tutti i record dei set di training dei 6 gruppi contemporaneamente: viene quindi "persa", in questo schema, l'appartenenza di ogni record (matrice *mxn* in input e valore di target in uscita) a uno specifico gruppo regionale. Lo schema è sintetizzato in figura 5.14:

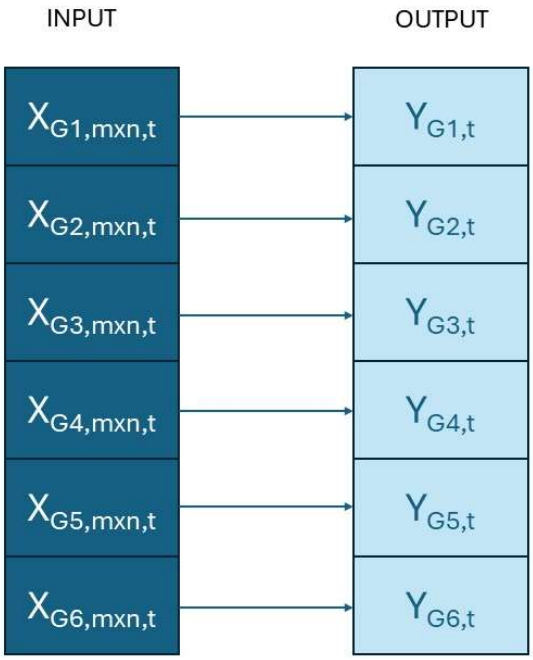


Fig. 5.14 – Schema LSTM con tutti i gruppi. G è il pedice per i gruppi, t quello per il tempo, m sono le caratteristiche, n la memoria.

L'indice *m* nell'immagine si riferisce alle variabili esplicative e vale 5 nel nostro caso, l'indice *G* invece è riferito ai gruppi, *n* rappresenta la memoria fornita al modello, nel caso in esame pari a 12 mesi e infine *t* è il pedice temporale che varia da 1 al numero di record (324-*n* per il training e 72-*n* per il test). I parametri utilizzati per la rete sono rimasti gli stessi riportati in tabella 5.B, nella quale *n* è stato posto pari a 12. Una volta effettuata la modellazione si sono calcolati gli indicatori della bontà della stima utilizzando i dati di test. Gli indicatori sono stati calcolati sia sul set intero dei dati di test (denominato totale in tab 5.E) sia singolarmente per ciascun gruppo. I risultati non sono stati soddisfacenti, con valori dell'indice di efficienza di Nash-Sutcliffe quasi sempre negativi. Per questo motivo le analisi successive si sono sempre svolte con modelli messi a punto separatamente per ciascun gruppo.

	GRUPPO1	GRUPPO2	GRUPPO3	GRUPPO4	GRUPPO5	GRUPPO6	TOTALE
NSE training	-0.840	-0.668	0.720	0.774	-0.588	0.664	0.797
NSE test	-0.960	-0.716	0.744	0.634	-0.710	0.551	0.739

Tab. 5.E – Coefficienti di Nash-Sutcliffe per la modellazione tramite LSTM sui set di training (1990-2016) e test (2017-2022) per modello unico per tutti i gruppi (memoria di 12 mesi).

6 – PREVISIONI DI PRODUCIBILITÀ

Fino ad ora i modelli sviluppati hanno utilizzato in input i dati storici delle forzanti meteorologiche che includevano anche il mese t di cui stava modellando la producibilità, ipotizzando quindi di conoscerne l'evoluzione per l'intero mese di previsione. Per l'applicazione dei modelli per la previsione futura della producibilità (che viene emessa nei primi giorni del mese) invece sono disponibili le misure delle forzanti meteorologiche solo fino al mese che precede quello su cui si vuole prevedere. Per prevedere la producibilità futura si è pensato di ricorrere a due approcci differenti. Un primo metodo è stato quello di progettare una nuova rete affinché prevedesse la producibilità dei mesi futuri (lead-time da 1 a 6 mesi successivi all'istante di previsione) solo in base alle osservazioni meteorologiche storiche disponibili fino al momento in cui si emette la previsione.

Un secondo metodo è stato quello di utilizzare la rete addestrata in precedenza, in cui si era fatta l'ipotesi di conoscere sempre l'effettiva evoluzione delle variabili meteo fino al mese su cui si stava prevedendo, ma utilizzando questa volta in input al posto dei dati storici ERA5 che nella realtà non sono disponibili, i dati di previsione meteorologica per i mesi successivi all'istante di previsione.

6.1 – Previsione della producibilità per i mesi futuri senza le previsioni meteorologiche.

Per poter prevedere una variabile di output per un istante futuro, senza utilizzare in ingresso le previsioni delle variabili in ingresso fino a tale istante, occorre addestrare la rete su un set impostato in maniera differente. Se in precedenza il rapporto temporale tra input e target era parallelo, associando all'input i-esimo il target i-esimo, per poter effettuare una previsione, il target deve invece essere traslato nel futuro di f intervalli temporali, dove *f*, o lead-time, rappresenta i mesi nel futuro di cui si vuole prevedere la producibilità, ovvero l'orizzonte temporale della previsione. Per questo è stata addestrata una nuova rete per lead-time. Infatti la rete restituirà in output solo la producibilità per il lead-time per il quale è stata addestrata. Per maggior chiarezza, immaginando di essere alla fine di Luglio/inizio Agosto e di voler prevedere la producibilità di Ottobre, occorrerà utilizzare la rete addestrata per prevedere su un lead-time di tre mesi, la quale fornirà una stima, basata sulla serie storica disponibile solo fino a fine Luglio, della producibilità di Ottobre. Di seguito è riportato uno schema (fig. 6.1) che mostra

l'organizzazione dei dati di input, per una singola cella, in cui *i* è l'indice che rappresenta l'istante temporale di previsione per la producibilità del mese *i+f*. Per esempio se *i* fosse Luglio ed *f* pari a 3, nella cella sarebbero presenti i dati di Maggio, Giugno e Luglio e in uscita si avrebbe la producibilità di Ottobre.

P_{i-2}	P_{i-1}	P_i
Tp_{i-2}	Tp_{i-1}	Tp_i
Sf_{i-2}	Sf_{i-1}	Sf_i
Et_{i-2}	Et_{i-1}	Et_i
Sd_{i-2}	Sd_{i-1}	Sd_i

Fig. 6.1 – Schema previsione producibilità senza dati meteorologici. Con memoria n pari a 3. L'indice *i* è l'istante di previsione per prevedere la producibilità per il mese i+f.

Per ogni gruppo si è deciso di utilizzare le stesse impostazioni della rete utilizzate nel modello precedente esposto in sezione 5.2. In ingresso per ogni output è stato fornito solo il numero di mesi di memoria che ha performato meglio nel test sui dati storici. In tabella 6.A sono riportati i coefficienti di Nash-Sutcliffe per il test dei modelli. Inoltre è riportata la serie temporale per il gruppo 4, utilizzando 12 mesi di memoria in input (fig. 6.2).

	GRUPPO1	GRUPPO2	GRUPPO3	GRUPPO4	GRUPPO5	GRUPPO6
Mesi di memoria	15	6	6	12	4	15
Lead-time 1 mese	0.681	0.749	0.825	0.831	0.778	0.579
Lead-time 2 mesi	0.475	0.472	0.813	0.753	0.572	0.474
Lead-time 3 mesi	0.441	0.461	0.812	0.751	0.576	0.317
Lead-time 4 mesi	0.326	0.413	0.793	0.700	0.532	0.253
Lead-time 5 mesi	0.312	0.314	0.804	0.668	0.393	0.203
Lead-time 6 mesi	0.267	0.240	0.787	0.607	0.268	0.144
Benchmark	0.388	0.621	0.788	0.635	0.616	0.311

Tab. 6.A – Coefficienti di Nash-Sutcliffe per la modellazione tramite LSTM sui set di test (2017-2022) per modello di previsione di producibilità al variare del lead-time senza fornire le previsioni meteorologiche in ingresso.

.

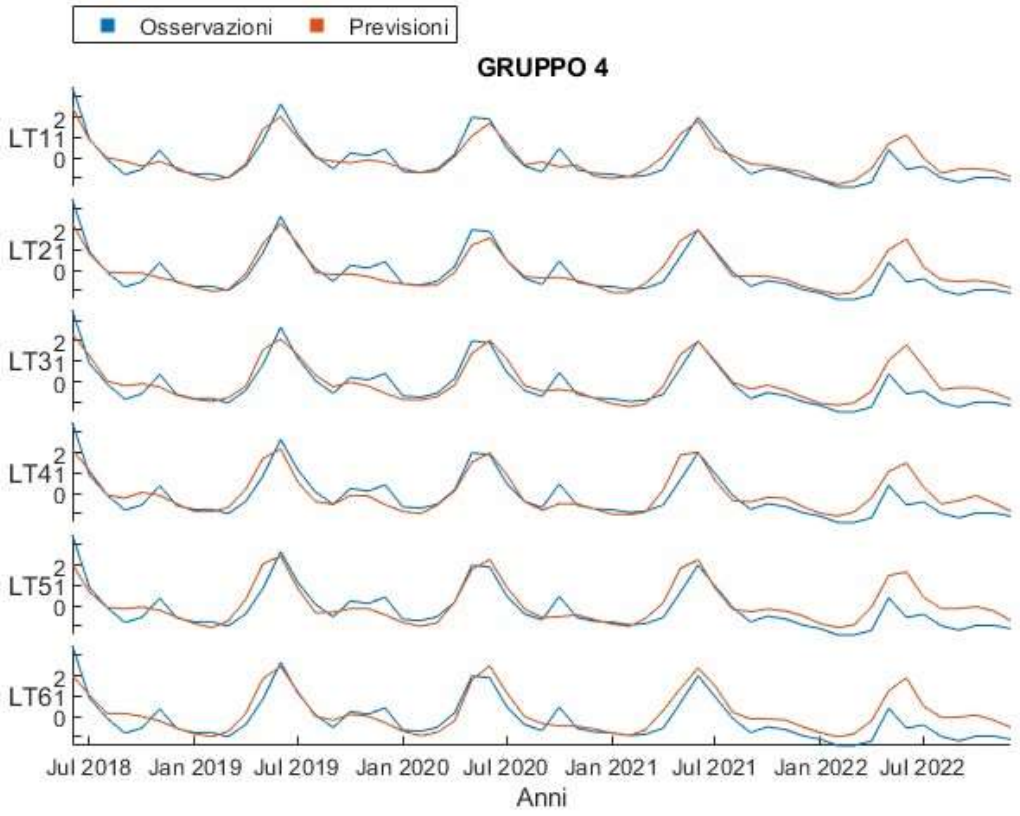


Fig. 6.2 – Serie temporale modello di previsione di producibilità senza previsioni meteorologiche tramite LSTM con n =12 per lead time variabile sul test set (2017-2022) – Gruppo 4.

Come si può notare dalla tabella 6.A, in generale, come atteso, la qualità dei modelli diminuisce all'aumentare del lead-time, e in modo drastico per tutti i gruppi a parte il 3 e il 4. Poiché il modello è migliorativo rispetto alla prassi attualmente utilizzata se il coefficiente di Nash-Sutcliffe è maggiore di quello ottenuto tramite il modello benchmark, il modello risulta utile per il primo mese di lead-time su tutti i gruppi. All'aumentare del leadtime, solo i gruppi 3 e 4 garantiscono valori dell'efficienza superiori alla soglia di quel gruppo fino a 5 mesi di lead-time. Per i gruppi 1 e 6, in cui le previsioni col benchmark sono particolarmente insoddisfacenti, la previsione del modello permette un leggero miglioramento per leadtime fino a 3 mesi. I

gruppi 2 e 5 invece risultano i peggiori fornendo un valore migliore del benchmark solo per il primo mese.

6.2 – Previsione della producibilità per i mesi futuri tramite i dati meteorologici

In questo paragrafo verrà esposto il secondo metodo tramite il quale si è deciso di stimare la producibilità futura. Se in precedenza si è provato a conoscere il comportamento futuro solo tramite la serie storica delle variabili di input disponibili fino all'istante di previsione, in questo caso si utilizzeranno invece in input le previsioni meteorologiche delle variabili esplicative disponibili dall'istante di previsione fino al lead-time voluto. I modelli utilizzati sono gli stessi addestrati sulla serie storica, la differenza sta nel set utilizzato in input per la previsione. Alla rete addestrata sui soli dati storici viene infatti fornito in input un set di dati contenenti, in questo caso, sia dati storici che previsioni meteorologiche delle variabili esplicative. Essendo *f* il numero di mesi nel futuro di cui si vuole prevedere la producibilità, in input al modello verranno forniti elementi costituiti da matrici *mxn* dove le ultime *f* colonne sono dati previsti, mentre le *(n-f)* colonne iniziali sono dati storici. Un esempio dello schema utilizzato è riportato in figura 6.3.

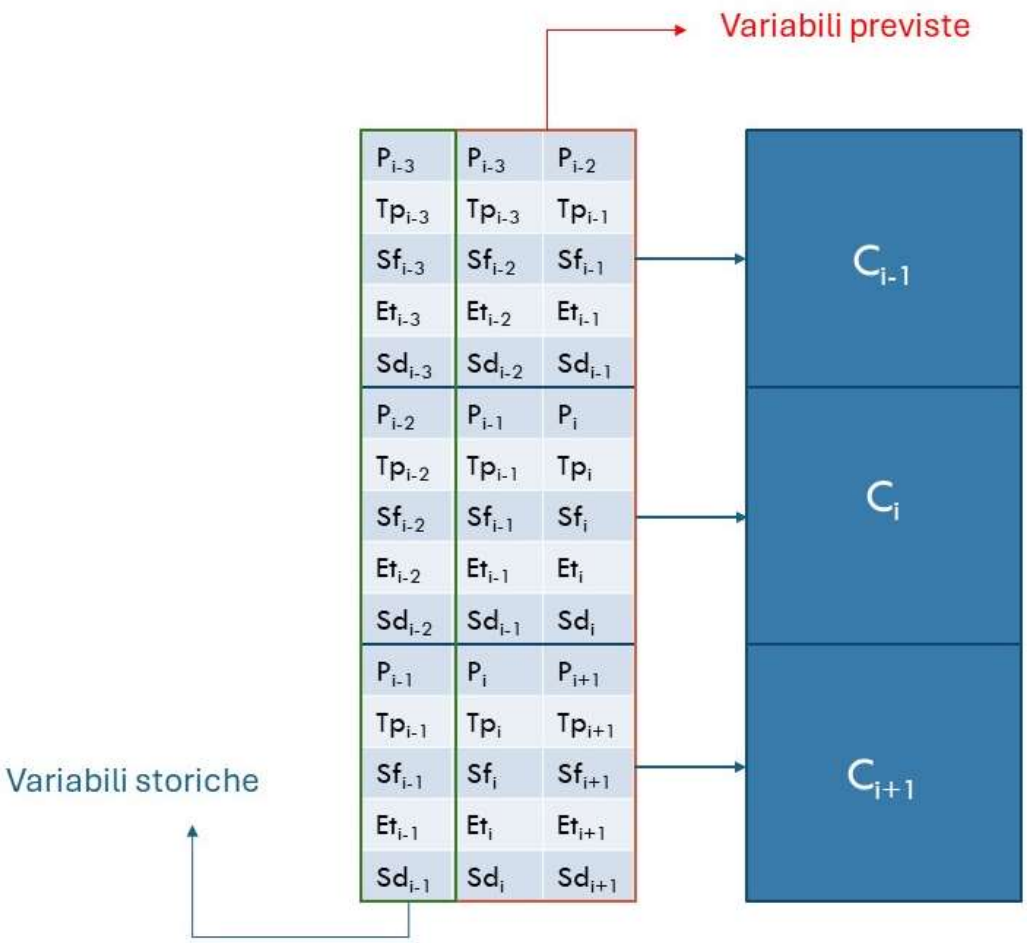


Fig. 6.3 – Schema training previsione producibilità con dati meteorologici per memoria n pari a 3 mesi e lead-time f pari a 2 mesi. Le C rappresentano i target del modello (producibilità) e l'indice *i* rappresenta l'intervallo temporale.

Nello schema è riportato l'esempio del training di una previsione utilizzando *n=3* mesi di memoria in input per ogni variabile con lo scopo di stimare la producibilità per i due mesi successivi, quindi con lead-time *f=2.* Per ogni nuova previsione, in input basta l'inserimento di un singolo elemento, impostato come la cella i-esima. Per ogni gruppo si è effettuato l'addestramento inserendo in input lo stesso numero di mesi n (memoria) utilizzati in precedenza e per ogni gruppo si è testata la rete anche per gli intervalli di memoria adiacenti (per esempio se in simulazione la memoria ottimale era risultata *n=4* si è testata la rete anche per *n=3* ed *n=6*). Per tutti i gruppi la memoria che ha fornito in test indicatori migliori è risultata la stessa risultante dalla simulazione sui dati storici, ma per il gruppo 5 è risultato migliore il

risultato ottenuto utilizzando *n=6* mesi di memoria anziché *n=4* mesi come ottenuto in fase di simulazione.

	GRUPPO1	GRUPPO2	GRUPPO3	GRUPPO4	GRUPPO5	GRUPPO6
Mesi di memoria	15	6	6	12	6	15
Lead-time 1 mese	0.822	0.816	0.837	0.849	0.850	0.681
Lead-time 2 mesi	0.597	0.678	0.821	0.777	0.533	0.580
Lead-time 3 mesi	0.481	0.601	0.819	0.765	0.356	0.571
Lead-time 4 mesi	0.478	0.596	0.807	0.725	0.342	0.579
Lead-time 5 mesi	0.447	0.612	0.793	0.714	0.404	0.524
Lead-time 6 mesi	0.433	0.600	0.790	0.682	0.266	0.517
Benchmark	0.388	0.621	0.788	0.635	0.616	0.311

Tab. 6.B – Coefficienti di Nash-Sutcliffe per la modellazione tramite LSTM sui set di test (2017-2022) per modello di previsione di producibilità al variare del lead-time fornendo le previsioni meteorologiche in ingresso e confronto con benchmark.

Si può notare che in generale il modello che utilizza anche le previsioni meteorologiche funzioni meglio del precedente anche per previsioni a più lungo termine (5 o 6 mesi), con efficienze superiori al benchmark su tutti i mesi di lead-time per quattro gruppi su sei e paragonabili al benchmark per tutti i leadtime per il gruppo 2. Il gruppo 5 però ha riscontrato dei problemi, fornendo risultati nettamente inferiori al benchmark gia' dal leadtime=2 anche con questo tipo di previsione. Esaminando i dati si è notato che, come anticipato in sezione 3.3, le previsioni di evapotraspirazione hanno un andamento sospetto, con massimi in inverno e minimi in estate. Questo andamento, soprattutto per il gruppo più meridionale non sembra corretto, per cui i risultati delle previsioni potrebbero essere stati compromessi dal valore di evapotraspirazione. Per sopperire a questo problema si sono effettuate nuovamente le previsioni utilizzando, per i dati futuri, la media mensile dell'evapotraspirazione storica ERA5 in sostituzione alle previsioni fornite da Copernicus. I dati sono stati sostituiti per i gruppi 1, 2 e 5, ovvero quelli più meridionali, nei quali i valori previsti di evapotraspirazione apparivano traslati (seppure in modo molto più marcato per il gruppo 5) nel tempo rispetto alle osservazioni. I risultati delle previsioni di producibilità ottenuti con i nuovi dati sono decisamente migliori dei precedenti sul gruppo 5 come si può notare nella tabella riassuntiva sottostante (tab. 6.C), mentre per i

gruppi 1 e 2 il cambio di stima dell'evapotraspirazione (non essendo, per tali gruppi, le due stime troppo lontane) non influisce in modo significativo sui risultati.

	GRUPPO1	GRUPPO2	GRUPPO3	GRUPPO4	GRUPPO5	GRUPPO6
Mesi in input	15	6	6	12	6	15
NSE 1 mese	0.822	0.817	0.837	0.849	0.864	0.681
NSE 2 mesi	0.626	0.694	0.821	0.777	0.673	0.580
NSE 3 mesi	0.514	0.612	0.819	0.765	0.591	0.571
NSE 4 mesi	0.480	0.606	0.807	0.725	0.605	0.579
NSE 5 mesi	0.449	0.623	0.793	0.714	0.624	0.524
NSE 6 mesi	0.443	0.616	0.790	0.682	0.629	0.517
Benchmark	0.388	0.621	0.788	0.635	0.616	0.311

Tab. 6.C – Coefficienti di Nash-Sutcliffe per la modellazione tramite LSTM sui set di test (2017-2022) per modello di previsione di producibilità al variare del lead-time fornendo le previsioni meteorologiche in ingresso ed evapotraspirazione media per i gruppi 1, 2, 5 e confronto con benchmark.

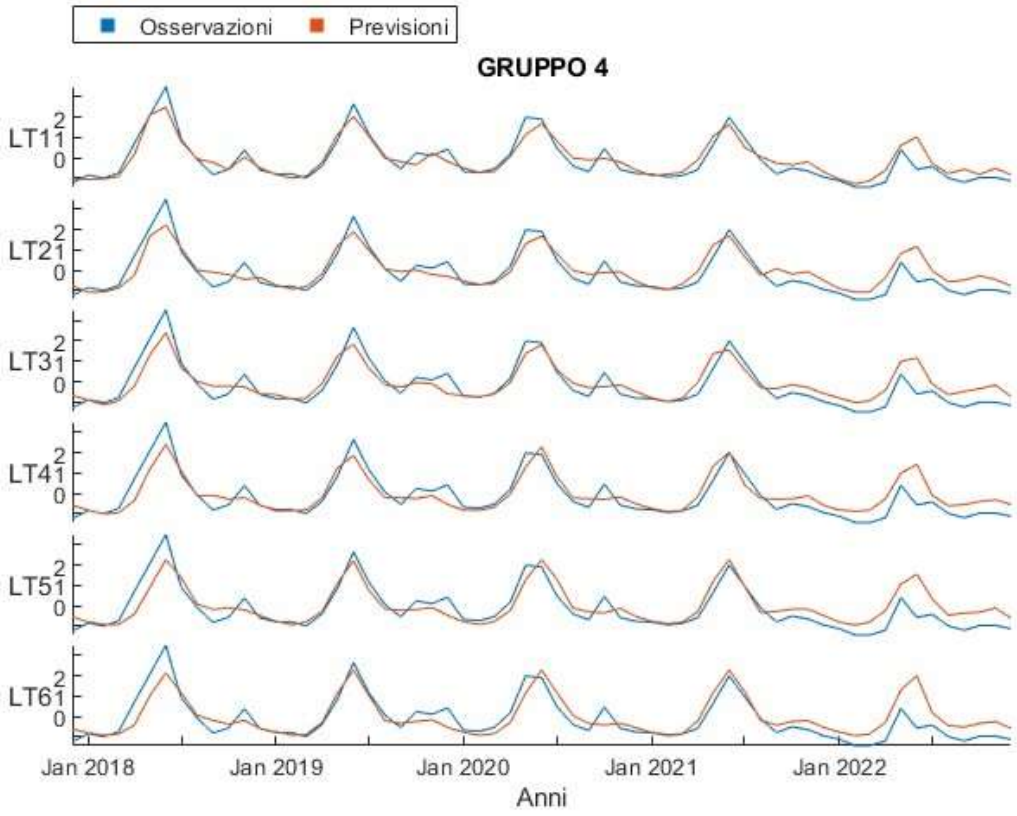


Fig. 6.4 – Serie temporale modello di previsione di producibilità con previsioni meteorologiche al variare del lead-time tramite LSTM con n = 12 mesi sul test set (2017-2022) – Gruppo 4.

7 – CONCLUSIONI

Prevedere la producibilità idroelettrica è di difficile attuazione in assenza di informazioni esaustive sulle caratteristiche del bacino e senza un'accurata misura delle forzanti meteorologiche che determinano i volumi turbinabili. Il compito si complica ulteriormente se la previsione non viene effettuata su un impianto singolo ma su un insieme di impianti di diversa natura e siti in bacini eterogenei. I modelli data-driven ed in particolare le reti neurali ricorrenti di tipo LSTM sono riuscite a sopperire in parte all'assenza di informazioni importanti e alla bassa qualità dei dati a disposizione. L'utilizzo dei dati meteorologici di Copernicus sia in training che in test ha permesso alla rete di trovare autonomamente le connessioni tra le variabili che governano il deflusso e quindi la producibilità idroelettrica.

Il modello che ha permesso le migliori prestazioni in simulazione (target al tempo t, dando in ingresso le variabili meteorologiche fino al tempo t incluso) e che dunque è stato scelto come modello definitivo è la rete neurale ricorrente di tipo LSTM con le caratteristiche elencate in tab. 5.B (sez. 5.3). Per ogni gruppo il modello ha utilizzato un numero differente di mesi di memoria:

1. n = 15 mesi
2. n = 6 mesi
3. n = 6 mesi
4. n = 12 mesi
5. n = 6 mesi
6. n = 15 mesi

Questo tipo di rete garantisce in test un risultato soddisfacente su tutti i gruppi, anche se per il gruppo 6 (Trentino Alto-Adige e Veneto) i risultati sono sempre stati di qualità minore per tutte le modellazioni effettuate. Nella figura 7.1 sono riportati gli indicatori di errore (NSE) in simulazione per il modello utilizzato e sono stati confrontati per gli stessi indicatori del modello benchmark.

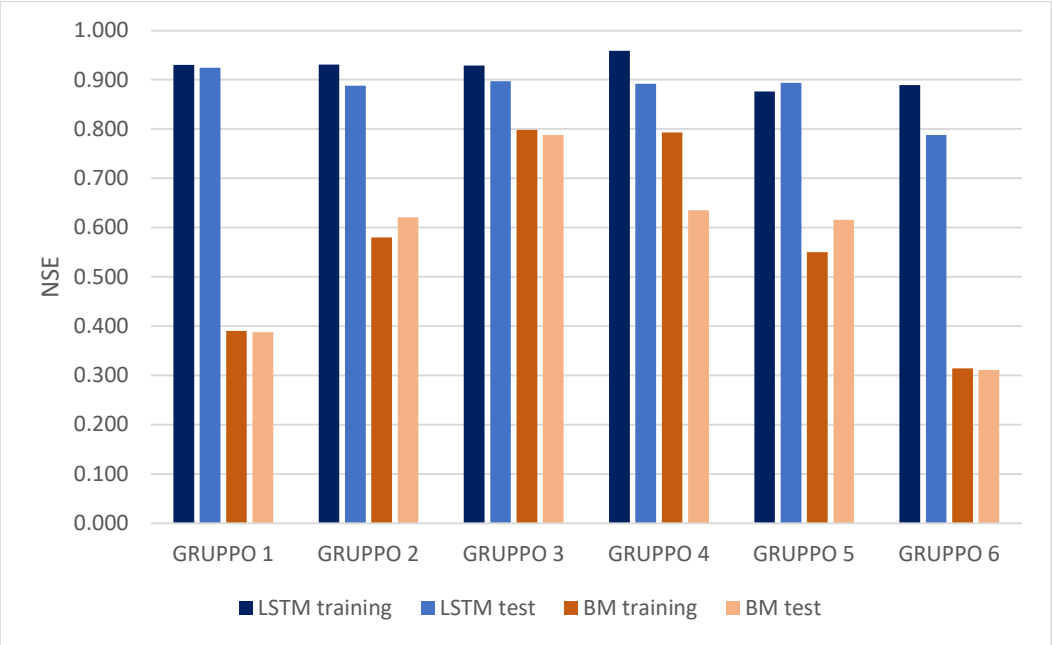


Fig. 7.1 – Coefficienti di Nash-Sutcliffe per modello LSTM in simulazione e confronto con modello benchmark sui set di training (1990-2016) e di test (2017-2022).

La previsione di producibilità (per i mesi successivi all'istante di previsione) è stata testata sui sei anni (2017-2022) per il quale erano disponibili le previsioni delle grandezze meteorologiche. Per prevedere la producibilità per i mesi futuri il metodo più efficace è stato tramite l'utilizzo in ingresso al modello messo a punto in simulazione delle previsioni delle variabili meteorologiche, modificando però il valore dell'evapotraspirazione per i gruppi meridionali (1, 2, 5) per via della poca affidabilità delle serie previste scaricate tramite il data base di Copernicus. I risultati sono esposti in figura 7.2.

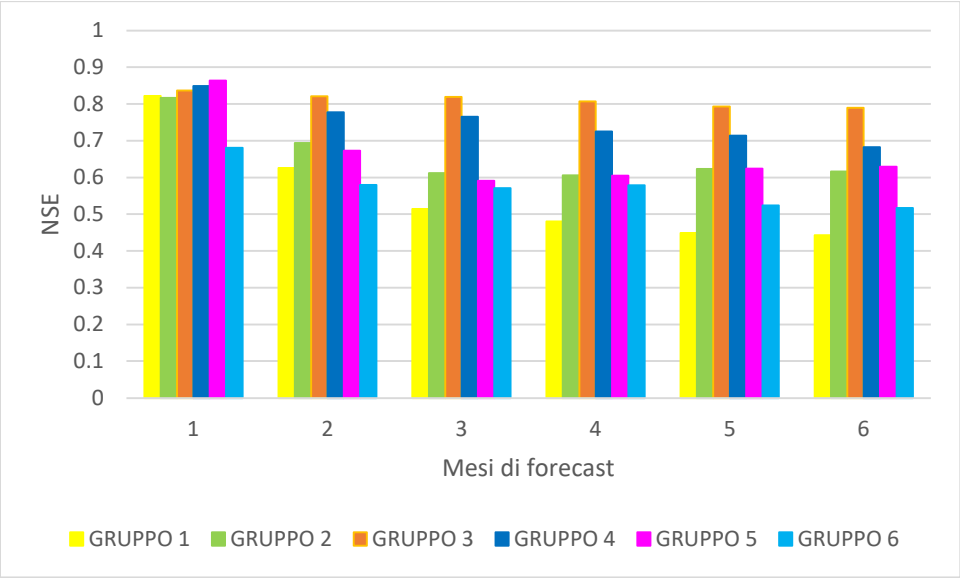


Fig. 7.2 – Coefficienti di Nash-Sutcliffe per modello LSTM della producibilità futura (con utilizzo previsioni meteorologiche) al variare del lead-time sul set di test (2017-2022).

Per stabilire l'applicabilità o meno del modello è stato comparato l'indice di Nash-Sutcliffe ottenuto dai modelli LSTM con lo stesso indicatore calcolato per il modello benchmark. Di seguito, nelle figg. da 7.3 a 7.8 sono riportati i valori dei coefficienti per tutti i gruppi sul set di test, al variare del lead-time, per il modello di previsione che utilizza le previsioni meteorologiche (LSTM A), per quello che utilizza solo la serie storica ERA5 (LSTM B) e per il modello benchmark (BM).

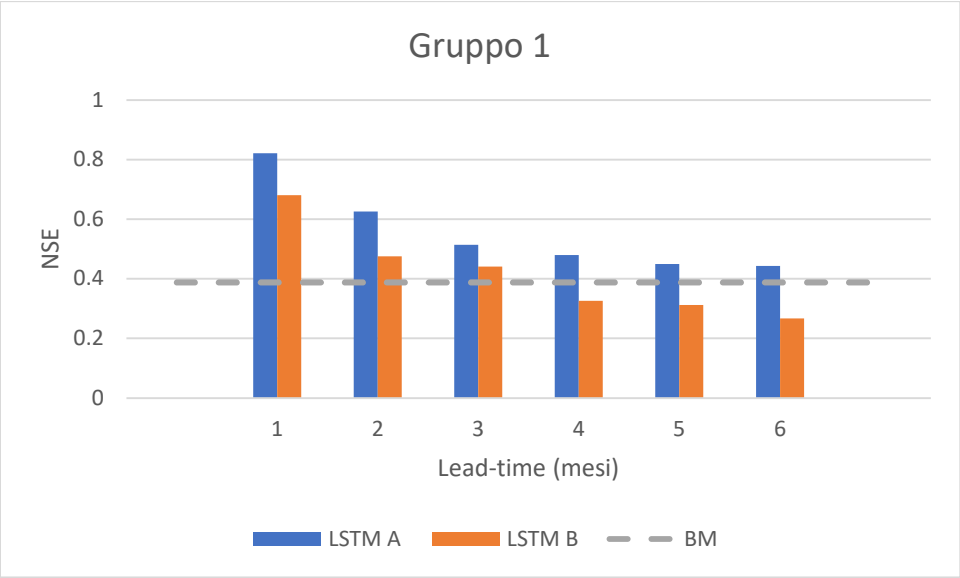


Fig. 7.3 – Coefficienti di Nash-Sutcliffe per modello LSTM della producibilità futura con utilizzo di previsioni meteorologiche (LSTM A), senza utilizzo di previsioni meteorologiche (LSTM B) al variare del lead-time sul set di test (2017-2022) e confronto con il benchmark (BM) per il gruppo 1.

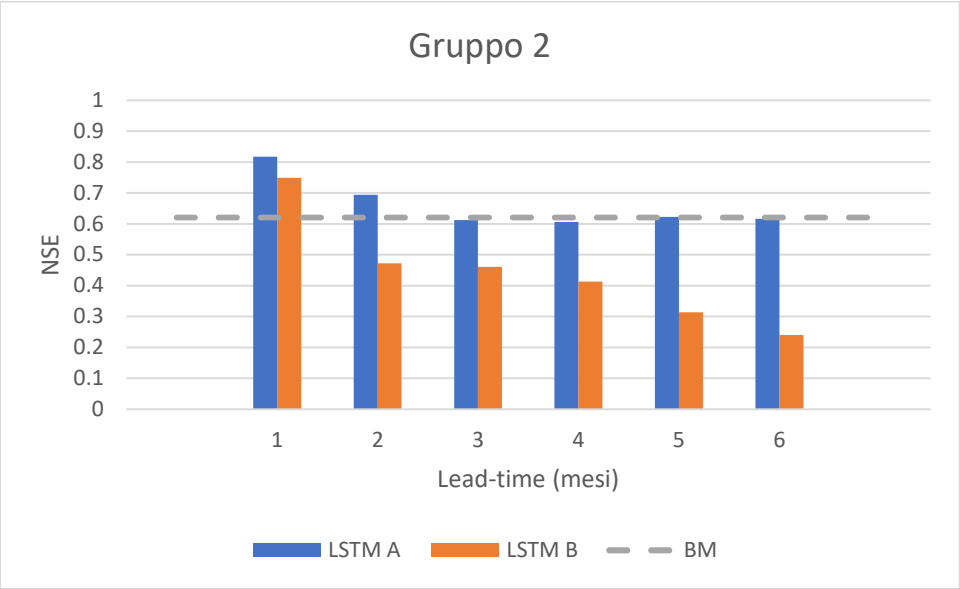


Fig. 7.4 – Coefficienti di Nash-Sutcliffe per modello LSTM della producibilità futura con utilizzo di previsioni meteorologiche (LSTM A), senza utilizzo di previsioni meteorologiche (LSTM B) al variare del lead-time sul set di test (2017-2022) e confronto con il benchmark (BM) per il gruppo 2.

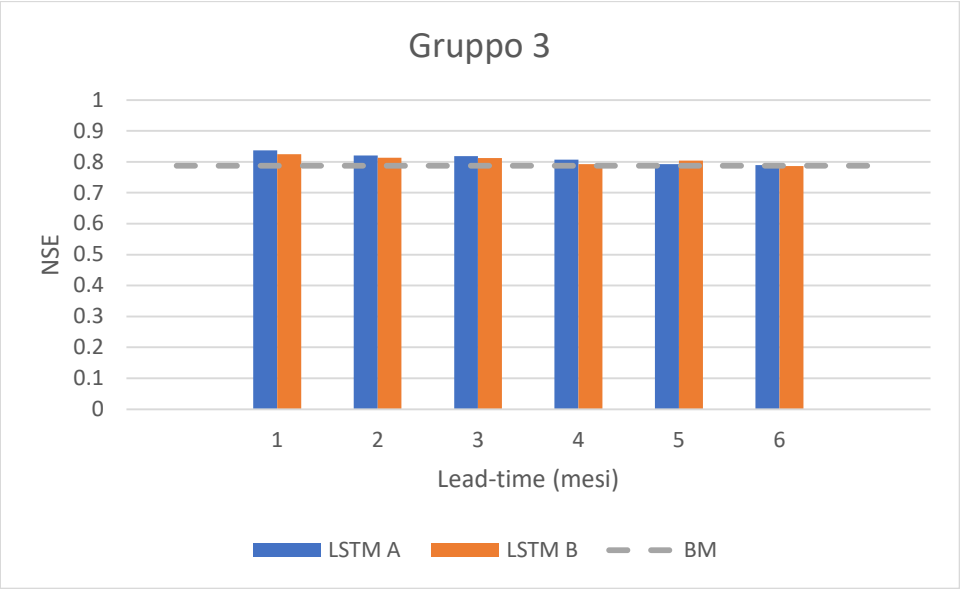


Fig. 7.5 – Coefficienti di Nash-Sutcliffe per modello LSTM della producibilità futura con utilizzo di previsioni meteorologiche (LSTM A), senza utilizzo di previsioni meteorologiche (LSTM B) al variare del lead-time sul set di test (2017-2022) e confronto con il benchmark (BM) per il gruppo 3.

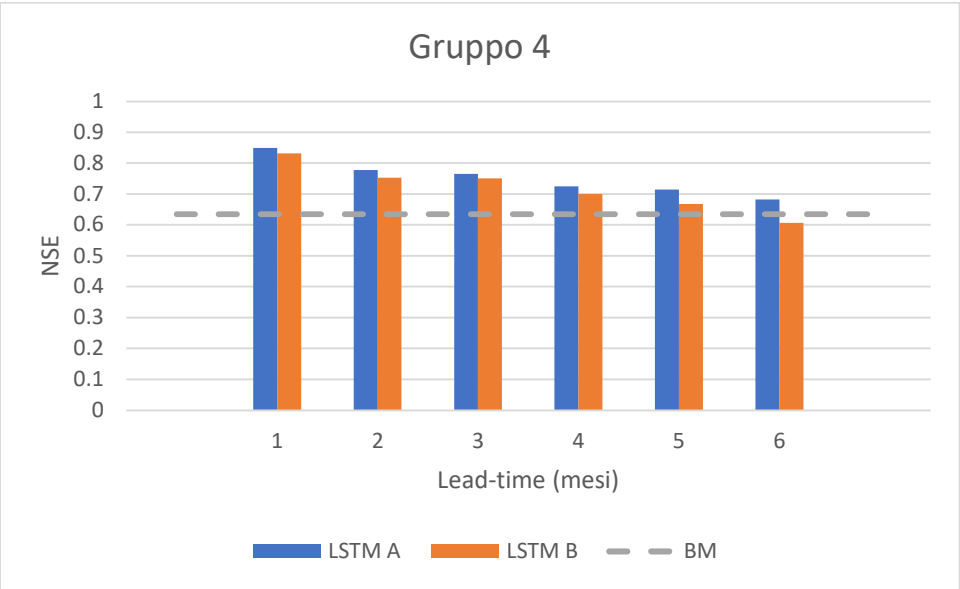


Fig. 7.6 – Coefficienti di Nash-Sutcliffe per modello LSTM della producibilità futura con utilizzo di previsioni meteorologiche (LSTM A), senza utilizzo di previsioni meteorologiche (LSTM B) al variare del lead-time sul set di test (2017-2022) e confronto con il benchmark (BM) per il gruppo 4.

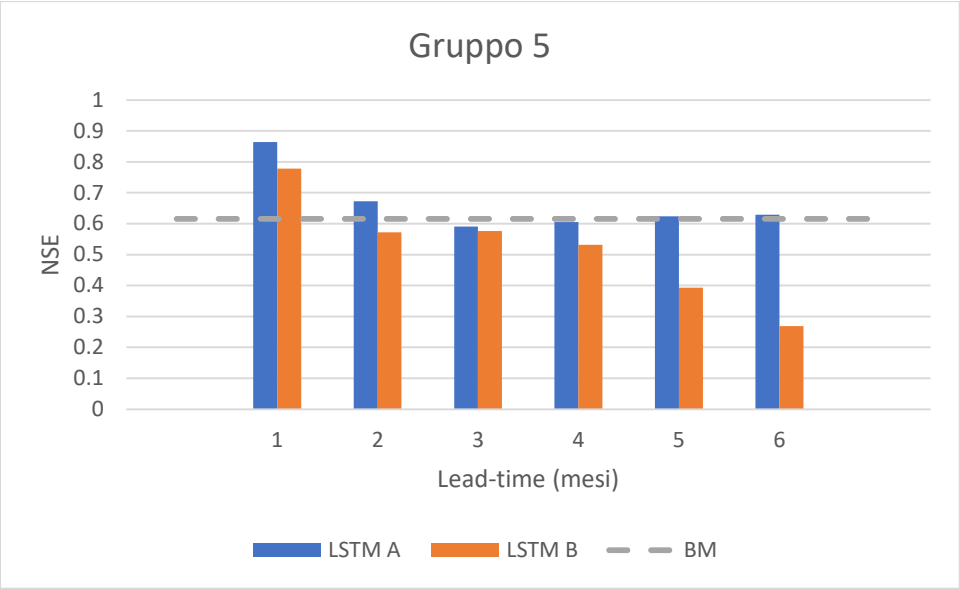


Fig. 7.7 – Coefficienti di Nash-Sutcliffe per modello LSTM della producibilità futura con utilizzo di previsioni meteorologiche (LSTM A), senza utilizzo di previsioni meteorologiche (LSTM B) al variare del lead-time sul set di test (2017-2022) e confronto con il benchmark (BM) per il gruppo 5.

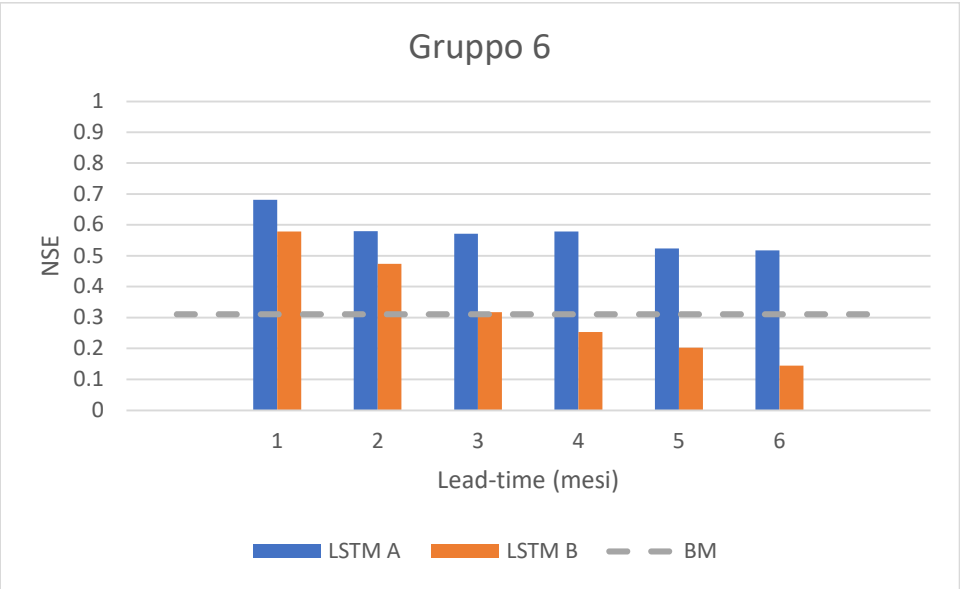


Fig. 7.8 – Coefficienti di Nash-Sutcliffe per modello LSTM della producibilità futura con utilizzo di previsioni meteorologiche (LSTM A), senza utilizzo di previsioni meteorologiche (LSTM B) al variare del lead-time sul set di test (2017-2022) e confronto con il benchmark (BM) per il gruppo 6.

Il modello LSTM che utilizza le previsioni meteorologiche è in generale il modello con i coefficienti più alti per tutti i gruppi e per tutti i mesi di lead-time, anche se sul gruppo 2 e sul gruppo 5 dal secondo mese di lead-time in avanti l'efficienza è paragonabile al benchmark.

Per tutti i gruppi la qualità della stima cala con l'aumentare del lead-time e in particolare già per lead-time superiori a 2 mesi si ha un calo netto degli indicatori: d'altra parte, soprattutto per i gruppi 1, 3 e 6, l'utilizzo del modello LSTM con in input le previsioni meteorologiche è risultata essere un'alternativa molto migliore rispetto all'utilizzo del modello benchmark.

Il modello della producibilità futura soffre purtroppo della bassa qualità delle previsioni meteorologiche, specialmente per quanto riguarda la precipitazione, la quale è la principale causa del deflusso. L'andamento appiattito e tendente alla media delle previsioni di precipitazione causa nel modello l'incapacità di prevedere i picchi di producibilità. Come già accennato in sezione 3.3 questo andamento è anche in parte da attribuirsi alla differente dimensione delle celle sulle quali sono state mediate le previsioni meteorologiche, le quali essendo più grandi nel caso delle previsioni, potrebbero trascurare effetti locali. Questo inconveniente potrebbe forse essere risolto riaddestrando la rete utilizzando i valori di previsione delle variabili di input invece delle osservazioni, in questo modo si permetterebbe alla rete di apprendere il pattern delle previsioni caratterizzato da assenza di estremi provando a compensarne gli errori sistematici. L'addestramento su questi dati non è stato possibile per l'assenza delle previsioni meteorologiche sul periodo di training.

BIBLIOGRAFIA

- ANDRESON E., (2006). Snow Accumulation and Ablation Model – SNOW-17.
- ABRAHART, R. J., Anctil, F., Coulibaly, P., Dawson, C. W., Mount, N. J., See, L. M. , ... & Wilby, R. L.(2012). Two decades of anarchy? Emerging themes and outstanding challenges for neural network river forecasting. Progress in Physical Geography, 36(4) , 480-513.
- BENGIO, Y., Simard, P., & Frasconi, P. (1994). Learning long-term dependencies wit h gradient descent is difficult. IEEE transactions on neural networks, 5(2), 157-166.
- BONAFÈ Alberto, Ruggeri Ludovica, (2022). Effetti cronici del climate change sulla producibilità idroelettrica di EGP Italia.
- BURNASH, R. J., & Ferral, R. L. (1973). A generalized streamflow simulation system : Conceptual modeling for digital computers. US Department of Commerce, National Weather Service, and State of California, Department of Water Resources.
- CHOI, J., Lee, J., & Kim, S. (2022). Utilization of the Long Short-Term Memory network for predicting streamflow in ungauged basins in Korea. Ecological Engineering, 182, 106699.
- CLARK, M. P., Slater, A. G., Rupp, D. E., Woods, R. A., Vrugt, J. A., Gupta, H. V., ... & Hay, L. E. (2008). Framework for Understanding Structural Errors (FUSE): A modular framework to diagnose differences between hydrological models. Water Resources Research, 44(12).
- CLARK, S. R., Lerat, J., Perraud, J. M., & Fitch, P. (2023). Deep learning for monthly rainfall-runoff modelling: a comparison with classical rainfall-runoff modelling across Australia. Hydrology and Earth System Sciences Discussions, 2023, 1-34.
- FRAME, J. M., Kratzert, F., Klotz, D., Gauch, M., Shalev, G., Gilon, O., ... & Nearing, G. S. (2022). Deep learning rainfall–runoff predictions of extreme events. Hydrology and Earth System Sciences, 26(13), 3377-3392.
- HERSBACH, H., Bell, B., Berrisford, P., Biavati, G., Horányi, A., Muñoz Sabater, J., Nicolas, J., Peubey, C., Radu, R., Rozum, I., Schepers, D., Simmons, A., Soci, C., Dee, D., Thépaut, J-N. (2023): ERA5 hourly data on single levels from 1940 to present. Copernicus Climate Change Service (C3S) Climate Data Store (CDS), DOI: 10.24381/cds.adbb2d47 (Accessed on 12-SEP-2023).

- HOCHREITER, S., & Schmidhuber, J. (1997). Long short-term memory. Neural computation, 9(8), 1735-1780.
- KINGMA, D. P., & Ba, J. (2014). Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980.
- KRATZERT, F., Klotz, D., Brenner, C., Schulz, K., & Herrnegger, M. (2018). Rainfall–runoff modelling using long short-term memory (LSTM) networks. Hydrology and Earth System Sciences, 22(11), 6005-6022.
- KRATZERT, F., Klotz, D., Herrnegger, M., Sampson, A. K., Hochreiter, S., & Nearing, G. S. (2019). Toward improved predictions in ungauged basins: Exploiting the power of machine learning. Water Resources Research, 55(12), 11344-11354.
- LEES, T., Buechel, M., Anderson, B., Slater, L., Reece, S., Coxon, G., & Dadson, S. J. (2021). Benchmarking data-driven rainfall–runoff models in Great Britain: a comparison of long short-term memory (LSTM)-based models with four lumped conceptual models. Hydrology and Earth System Sciences, 25(10), 5517-5534.
- MAJESKE, N., Abesh, B., Zhu, C., & Azad, A. (2021). Inductive predictions of extreme hydrologic events in the wabash river watershed. arXiv preprint arXiv:2104.14658.
- MINNS, A. W., & Hall, M. J. (1996). Artificial neural networks as rainfall-runoff models. Hydrological sciences journal, 41(3), 399-417.
- MOISELLO U., (1999). Idrologia Tecnica, Pavia, La Goliardica Pavese, pp. 109NASH, J. E., & Sutcliffe, J. V. (1970). River flow forecasting through conceptual models part I—A discussion of principles. Journal of hydrology, 10(3), 282-290.
- NASH, J. E., & Sutcliffe, J. V. (1970). River flow forecasting through conceptual models part I—A discussion of principles. Journal of hydrology, 10(3), 282-290.
- OUMA, Y. O., Cheruyot, R., & Wachera, A. N. (2021). Rainfall and runoff time-series trend analysis using LSTM recurrent neural network and wavelet neural network with satellite-based meteorological data: case study of Nzoia hydrologic basin. Complex & Intelligent Systems, 1-24.
- PERRIN, C., Michel, C., & Andréassian, V. (2001). Does a large number of parameters enhance model performance? Comparative assessment of common catchment model structures on 429 catchments. Journal of hydrology, 242(3-4), 275-301.
- SALAS, F. R., Somos-Valenzuela, M. A., Dugger, A., Maidment, D. R., Gochis, D. J., David, C. H., ... & Noman, N. (2018). Towards real-time continental scale streamflow

simulation in continuous and discrete space. JAWRA Journal of the American Water Resources Association, 54(1), 7-27.

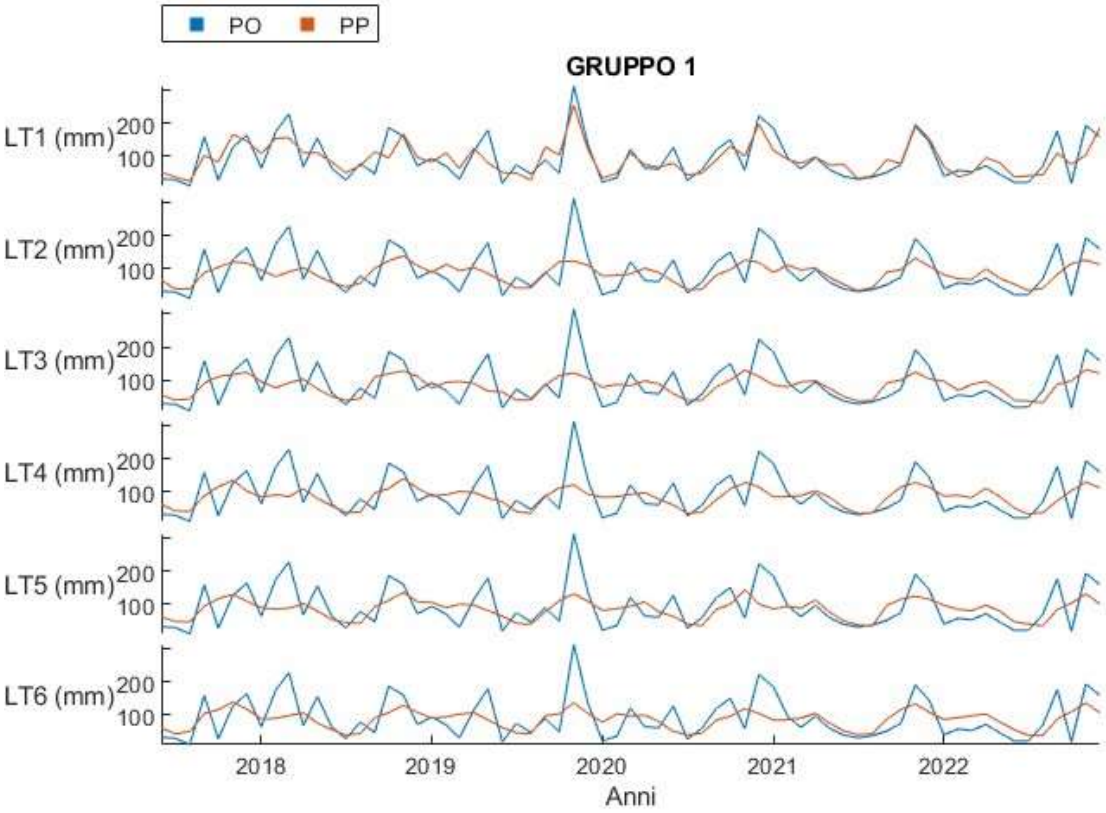
- SONG, T., Ding, W., Wu, J., Liu, H., Zhou, H., & Chu, J. (2019). Flash flood forecasting based on long short-term memory networks. Water, 12(1), 109.
- SRIVASTAVA, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: a simple way to prevent neural networks from overfitting. The journal of machine learning research, 15(1), 1929-1958.
- TIAN, Y., Xu, Y. P., Yang, Z., Wang, G., & Zhu, Q. (2018). Integration of a parsimonious hydrological model with recurrent neural networks for improved streamflow forecasting. Water, 10(11), 1655.
- TOSADORI F. M., (2019). Gaussian Process Regression una tecnica di machine learning per il pricing veloce di derivati, tesi di laurea magistrale, Politecnico di Milano, relatore: Marazzina D.
- TOTH, E. (2008). Data-driven streamflow simulation: the influence of exogenous variables and temporal resolution. Practical Hydroinformatics: Computational Intelligence and Technological Developments in Water Applications, 113-125.
- TOTH, E., & Brath, A. (2007). Multistep ahead streamflow forecasting: Role of calibration data in conceptual and neural network modeling. Water Resources Research, 43(11).
- WANG, Q. J., Pagano, T. C., Zhou, S. L., Hapuarachchi, H. A. P., Zhang, L., & Robertson, D. E. (2011). Monthly versus daily water balance models in simulating monthly runoff. Journal of Hydrology, 404(3-4), 166-175.
- WILLMOTT, C. J. (1981). On the validation of models. Physical geography, 2(2), 184-194.

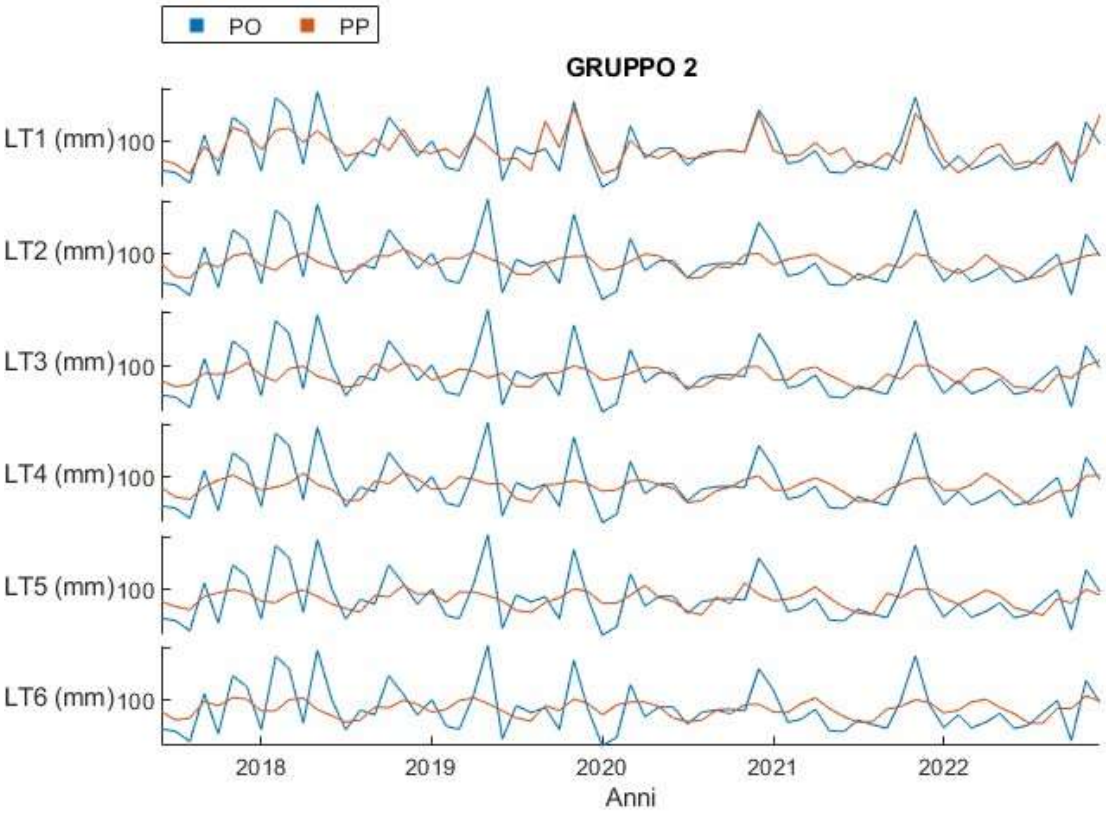
SITOGRAFIA

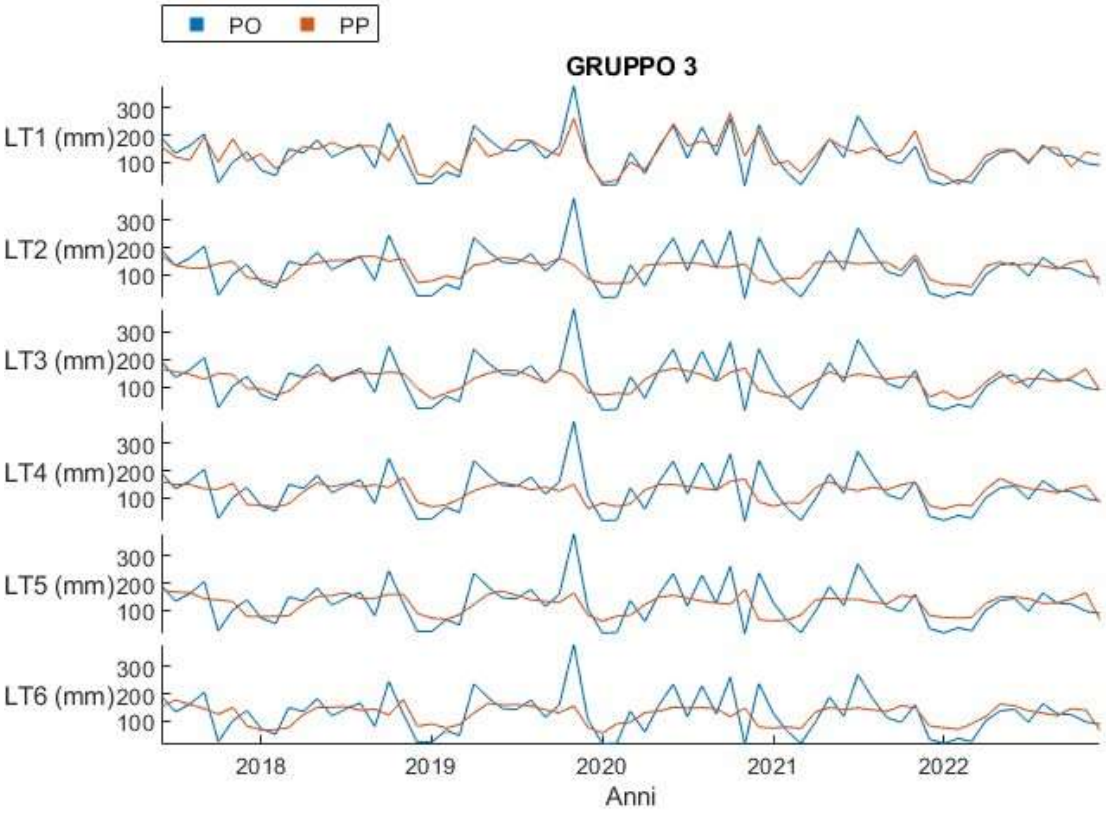
- Copernicus Climate Change Service, Climate Data Store, (2018): Seasonal forecast monthly statistics on pressure levels. Copernicus Climate Change Service (C3S) Climate Data Store (CDS). DOI: 10.24381/cds.0b79e7c5 (Accessed on 30-JAN-2024)
- https://cds.climate.copernicus.eu/cdsapp#!/dataset/reanalysis-era5-single-levels?tab=overview
- https://cds.climate.copernicus.eu/cdsapp#!/dataset/seasonal-monthly-single-levels?tab=overview
- https://cirrus.ucsd.edu/~pierce/ncdf/
- https://docs.unidata.ucar.edu/netcdf-c/current/index.html
- https://it.mathworks.com/discovery/support-vector-machine.html
- https://it.mathworks.com/help/stats/fitrensemble.html
- https://it.mathworks.com/help/stats/gaussian-process-regression-models.html
- https://www.copernicus.eu/en/about-copernicus
- MATLAB. (R2023b). Natick, Massachusetts: The MathWorks Inc. https://it.mathworks.com/products/matlab.html
- QGIS.org, 2023. QGIS Geographic Information System. QGIS Association. http://www.qgis.org
- R Core Team (2023). _R: A Language and Environment for Statistical Computing_. R Foundation for Statistical Computing, Vienna, Austria. <https://www.R-project.org/>.

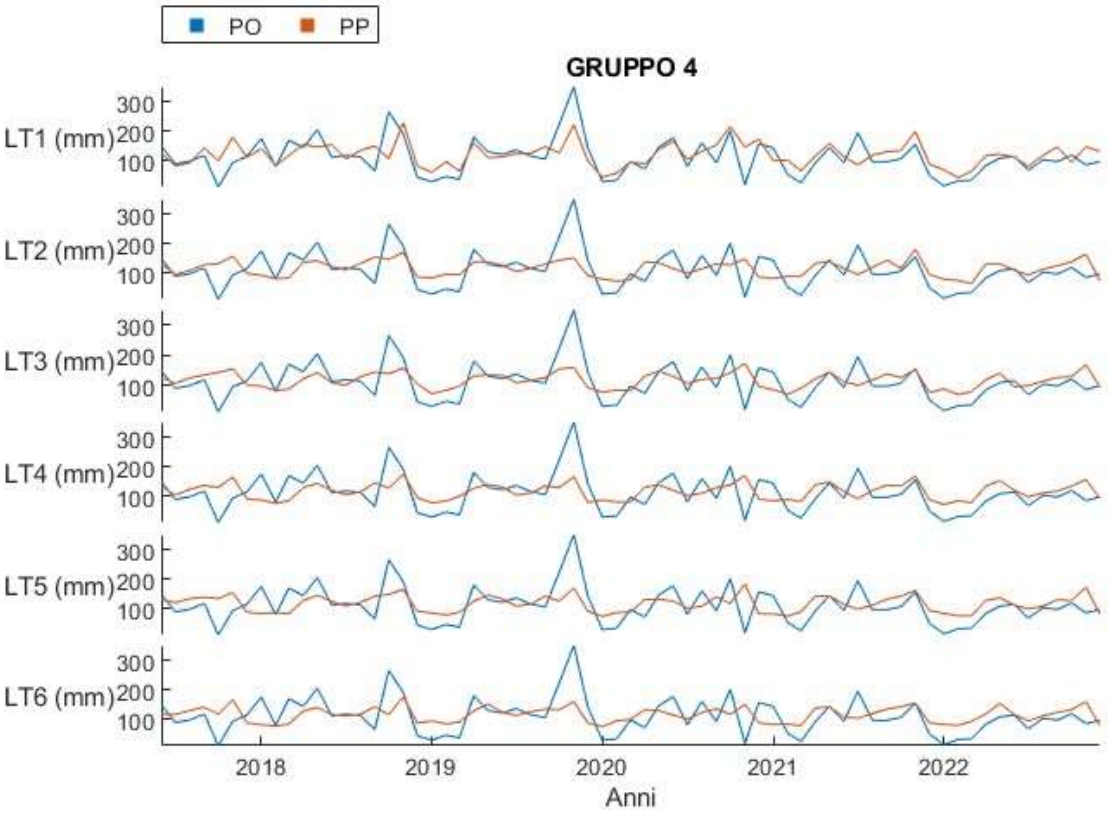
APPENDICE A

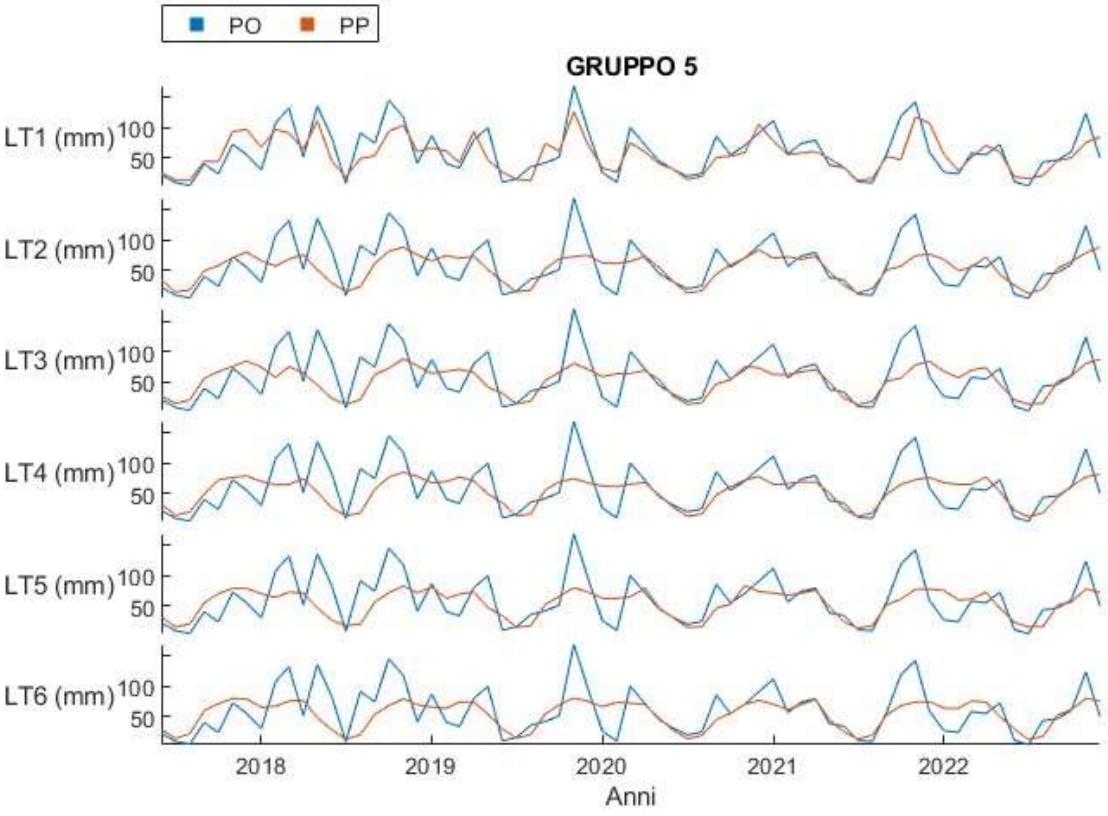
A.1 Previsioni di precipitazione al variare del lead-time

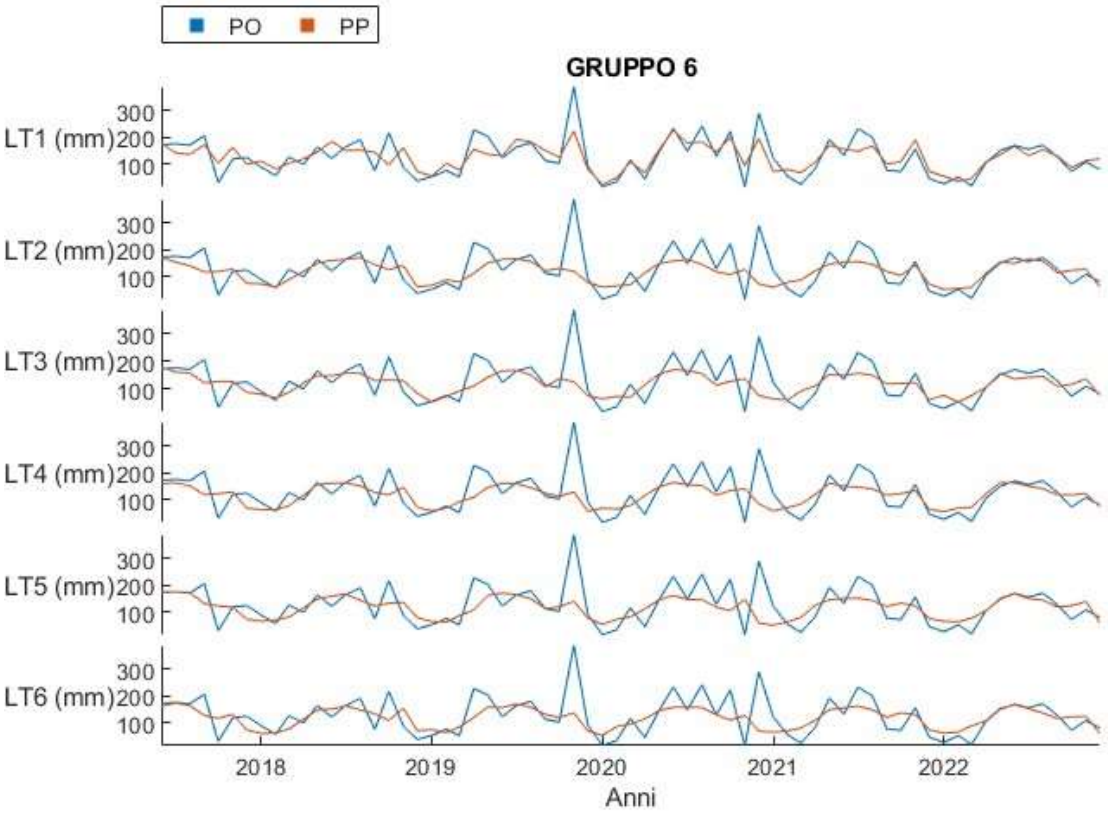



PO
PP
GRUPPO 2
LT1 (mm)
LT2 (mm)
LT3 (mm)
LT4 (mm)
LT5 (mm)
LT6 (mm)
100
2018
2019
2020
2021
2022
Anni

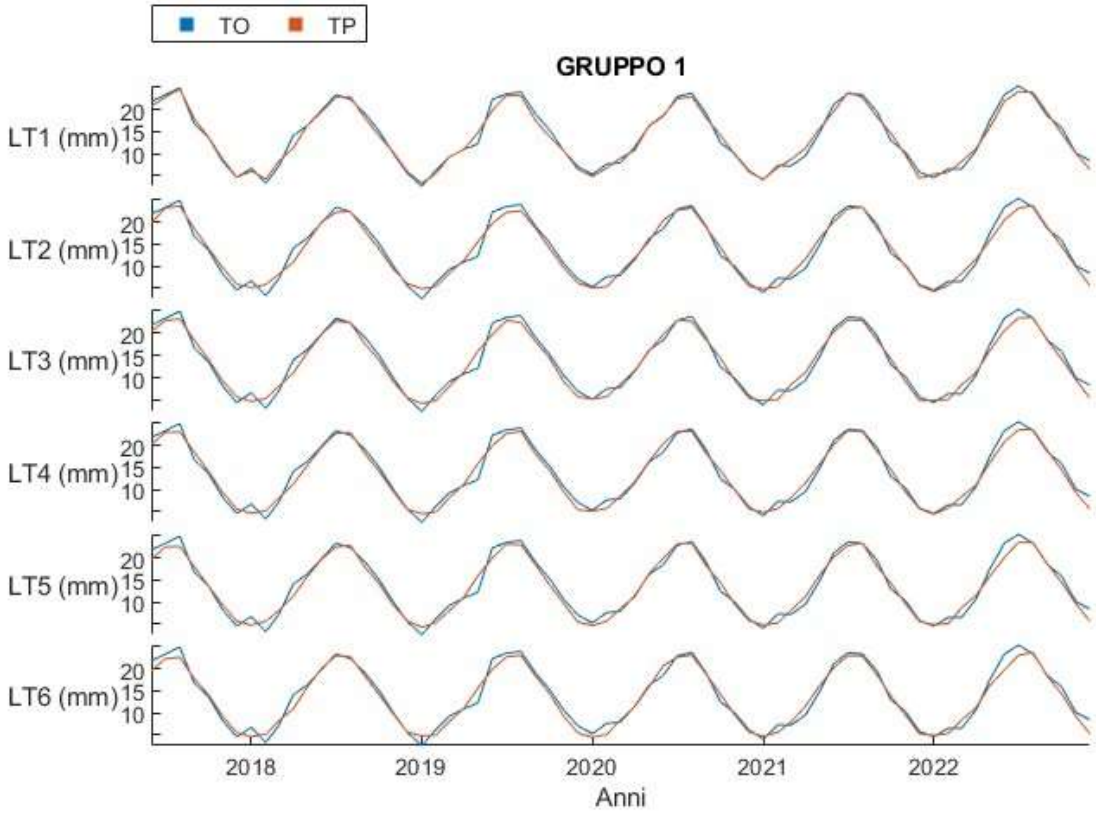

PO
PP
GRUPPO 3
LT1 (mm)
LT2 (mm)
LT3 (mm)
LT4 (mm)
LT5 (mm)
LT6 (mm)
300
200
100
2018
2019
2020
2021
2022
Anni

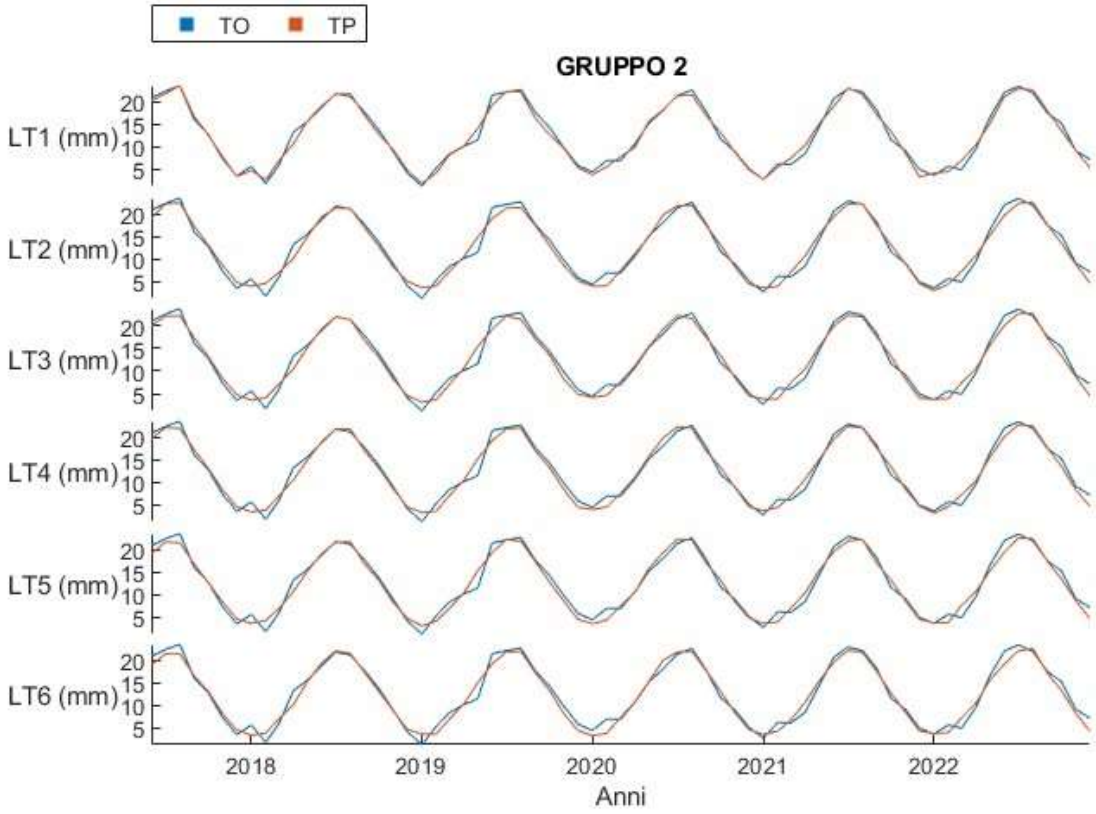

PO
PP
GRUPPO 4
LT1 (mm)
LT2 (mm)
LT3 (mm)
LT4 (mm)
LT5 (mm)
LT6 (mm)
300
200
100
2018
2019
2020
2021
2022
Anni

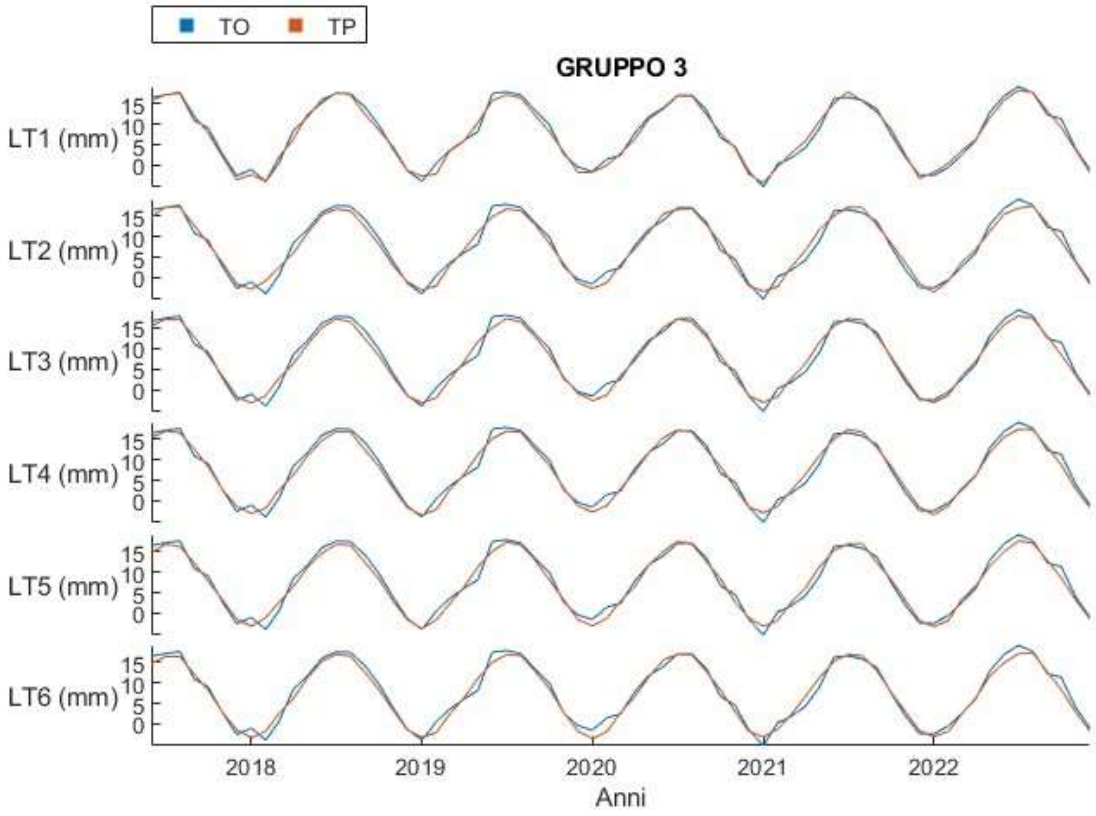

PO
PP
GRUPPO 5
LT1 (mm)
LT2 (mm)
LT3 (mm)
LT4 (mm)
LT5 (mm)
LT6 (mm)
100
50
2018
2019
2020
2021
2022
Anni

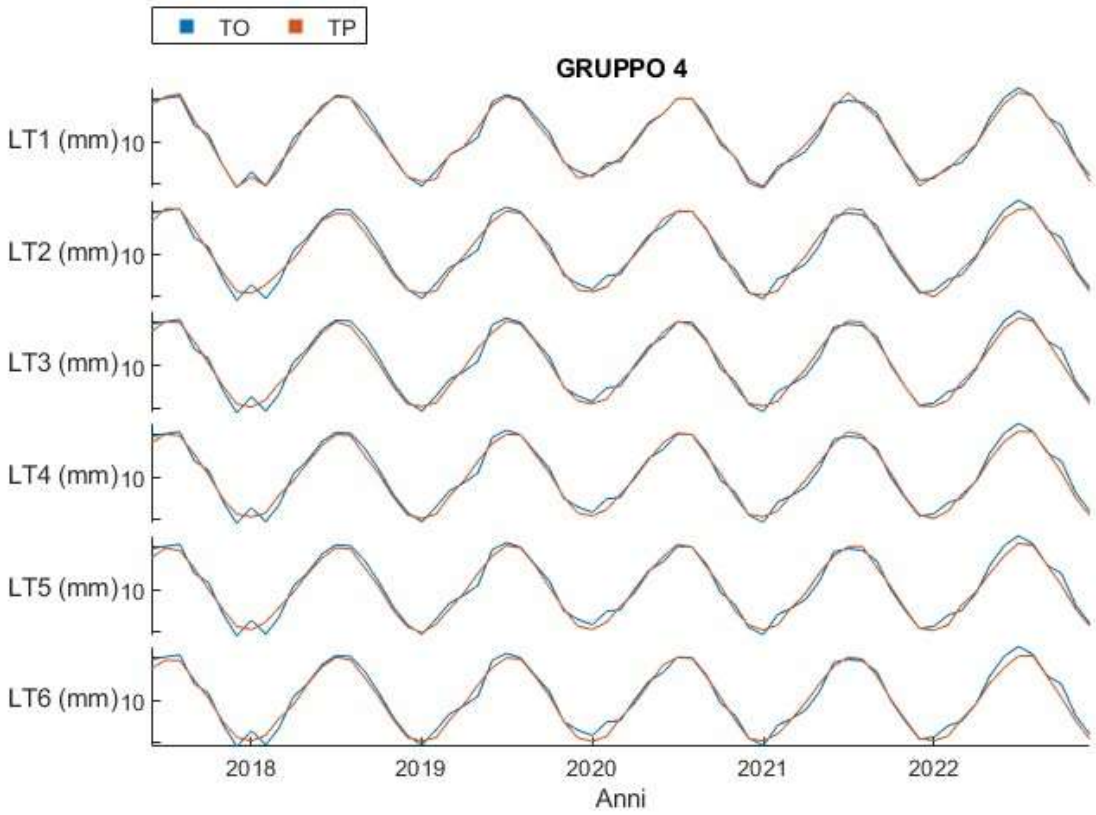

PO
PP
GRUPPO 6
LT1 (mm)
LT2 (mm)
LT3 (mm)
LT4 (mm)
LT5 (mm)
LT6 (mm)
300
200
100
2018
2019
2020
2021
2022
Anni

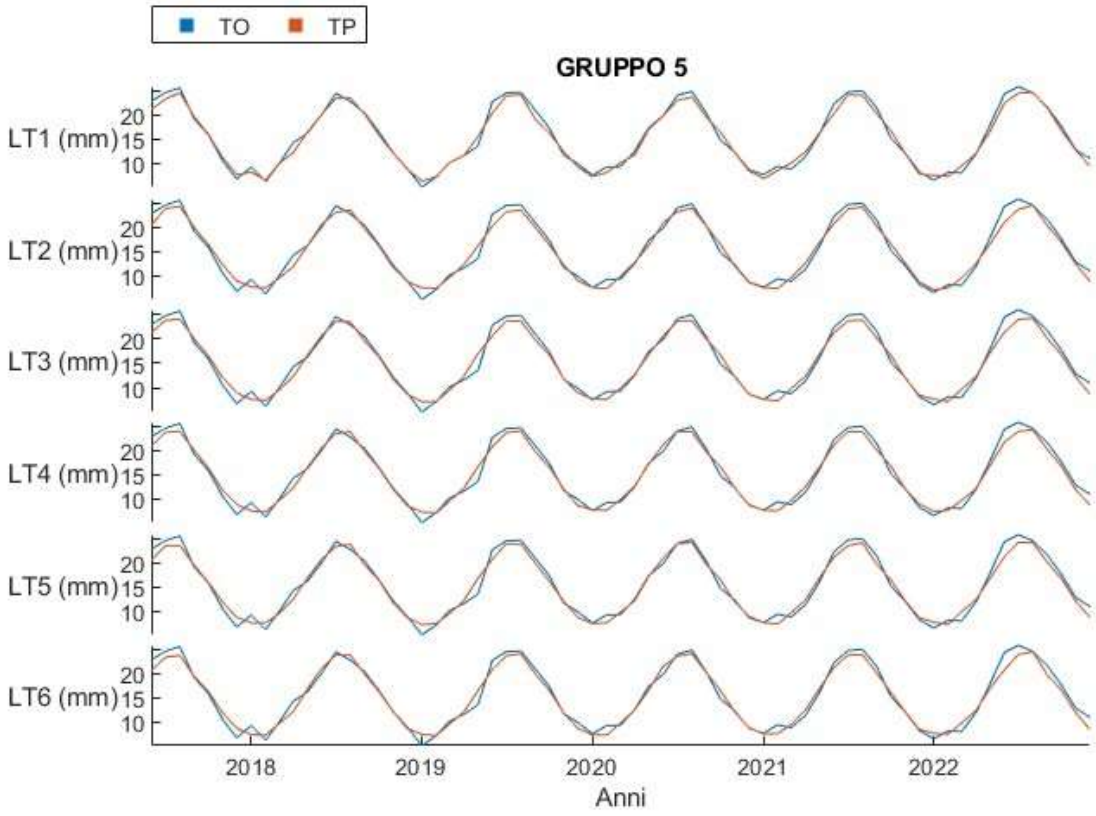
A.2 Previsioni di temperatura al variare del lead-time

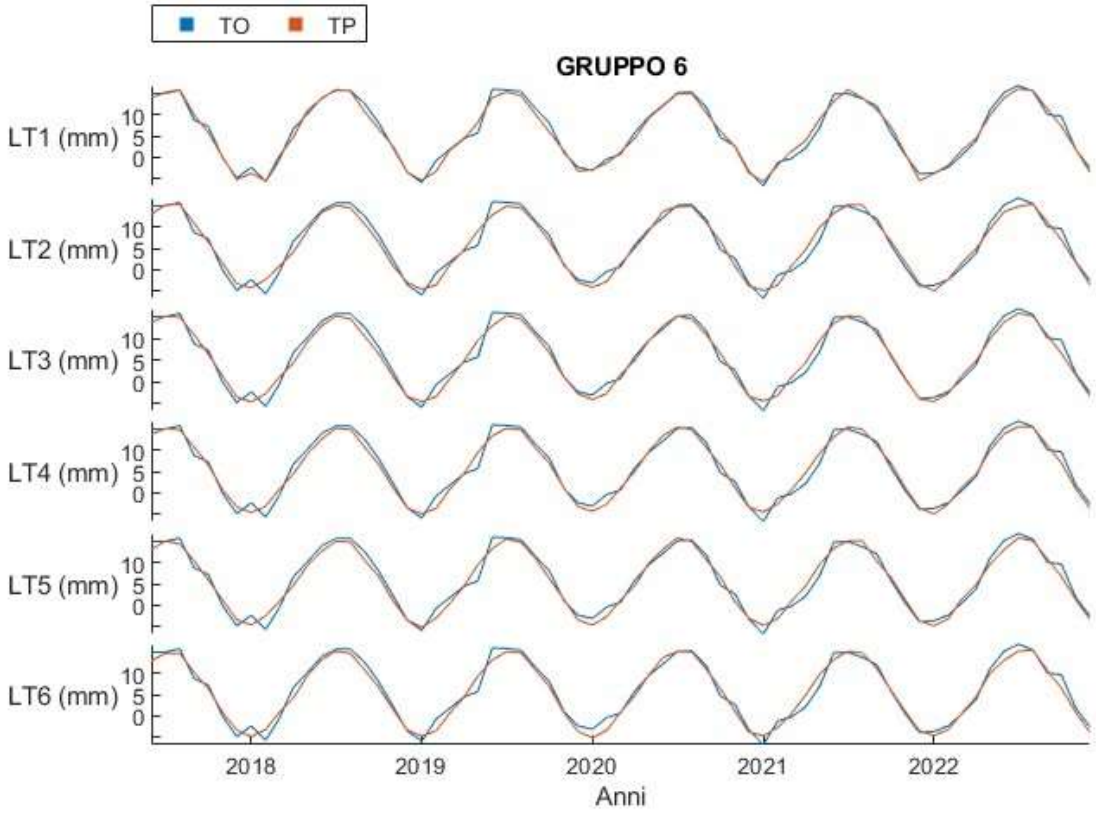




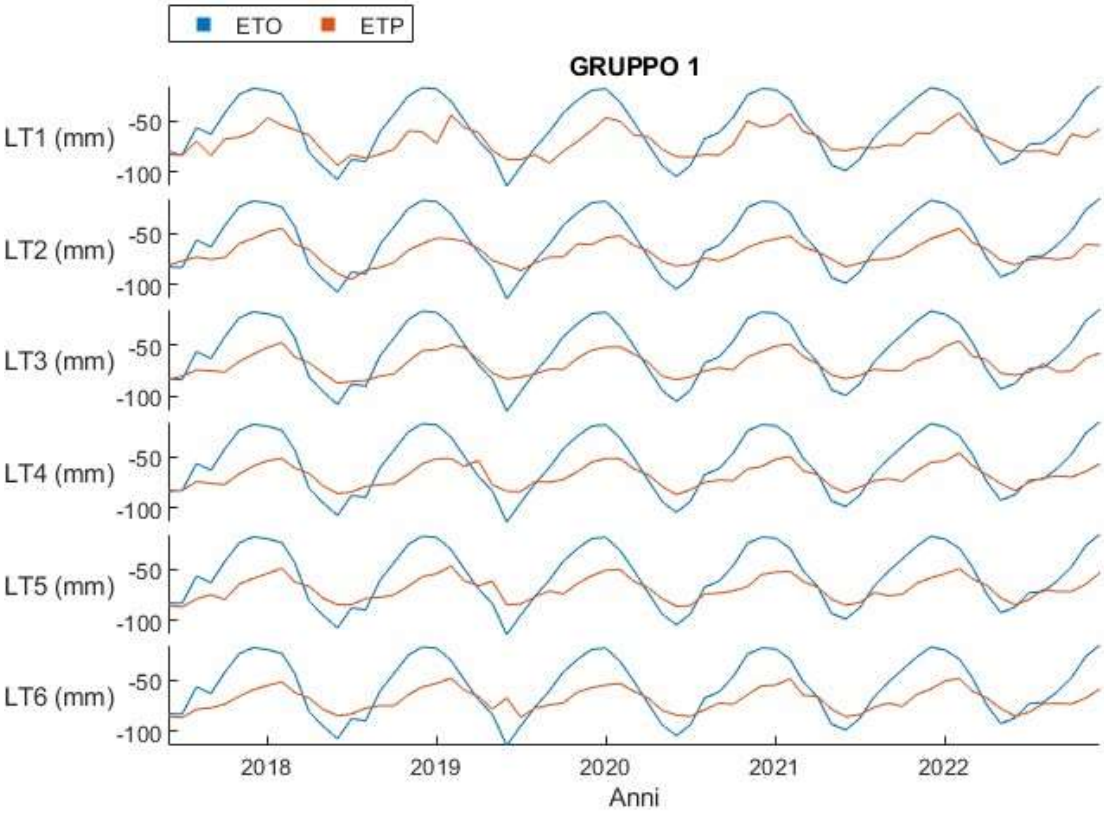

TO
TP
GRUPPO 3
LT1 (mm)
LT2 (mm)
LT3 (mm)
LT4 (mm)
LT5 (mm)
LT6 (mm)
15
10
5
0
2018
2019
2020
2021
2022
Anni

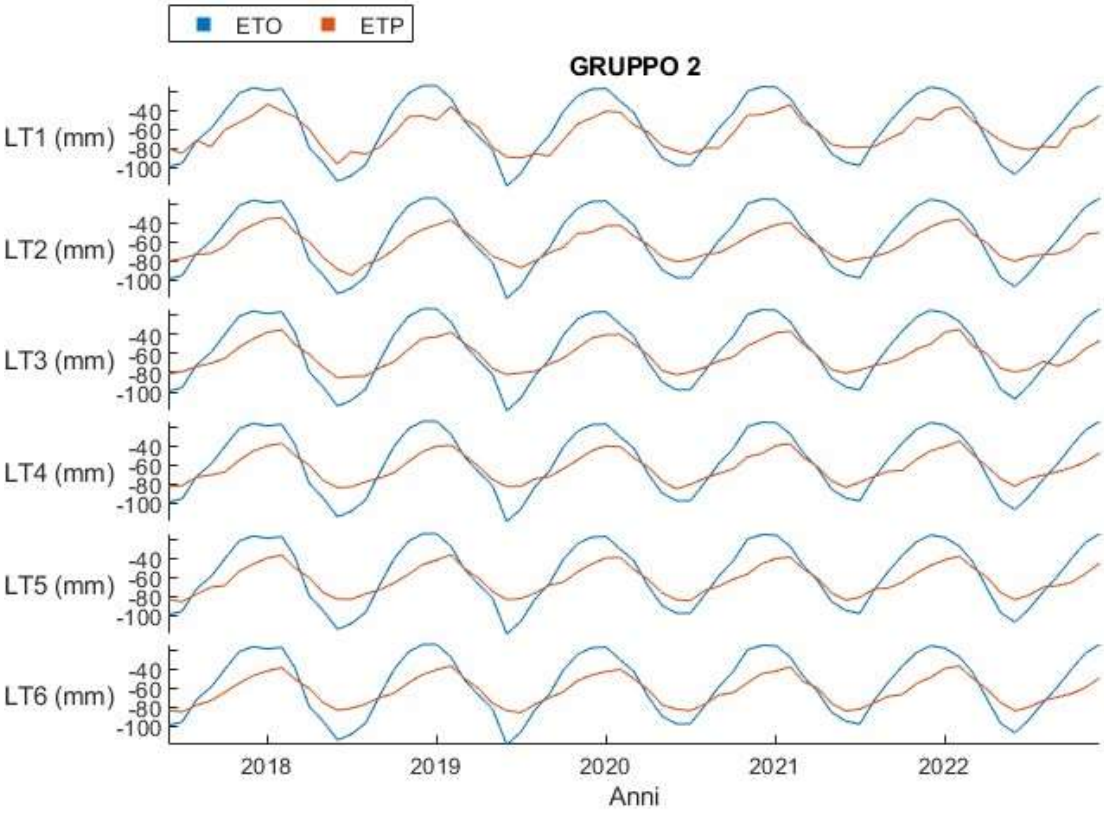

TO
TP
GRUPPO 4
LT1 (mm)
LT2 (mm)
LT3 (mm)
LT4 (mm)
LT5 (mm)
LT6 (mm)
10
2018
2019
2020
2021
2022
Anni

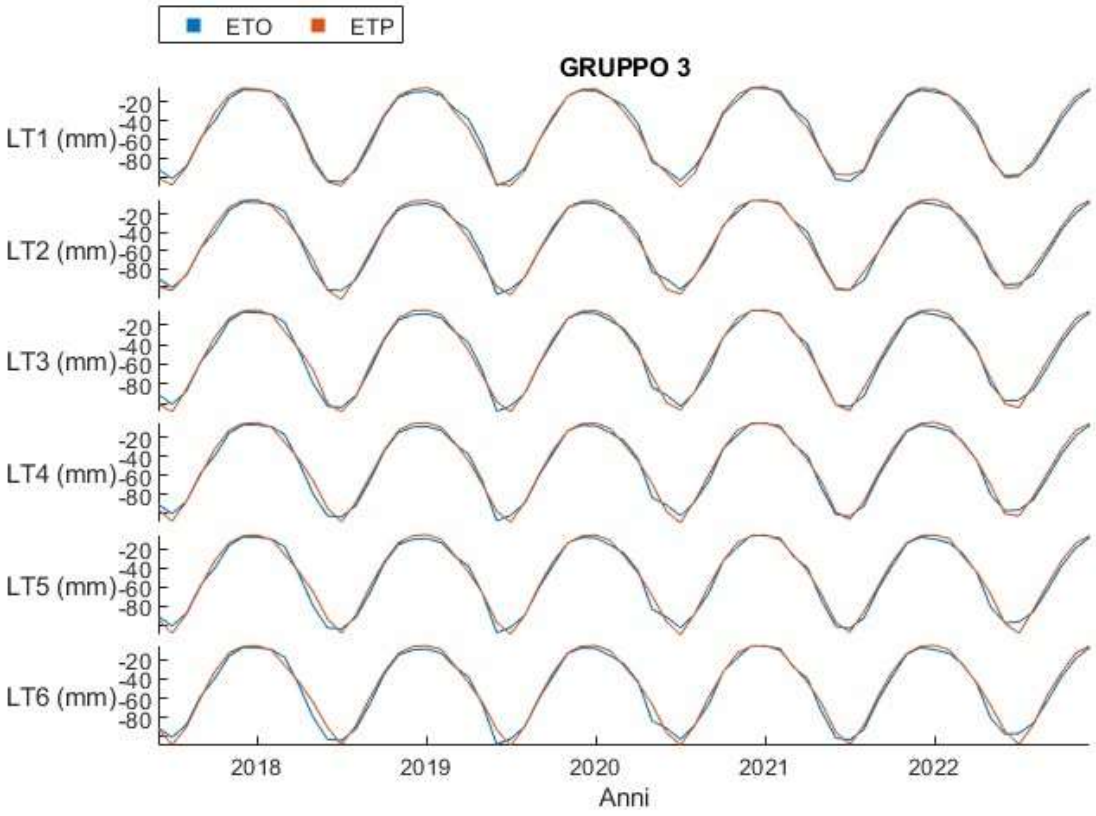

TO
TP
GRUPPO 5
LT1 (mm)
LT2 (mm)
LT3 (mm)
LT4 (mm)
LT5 (mm)
LT6 (mm)
20
15
10
2018
2019
2020
2021
2022
Anni

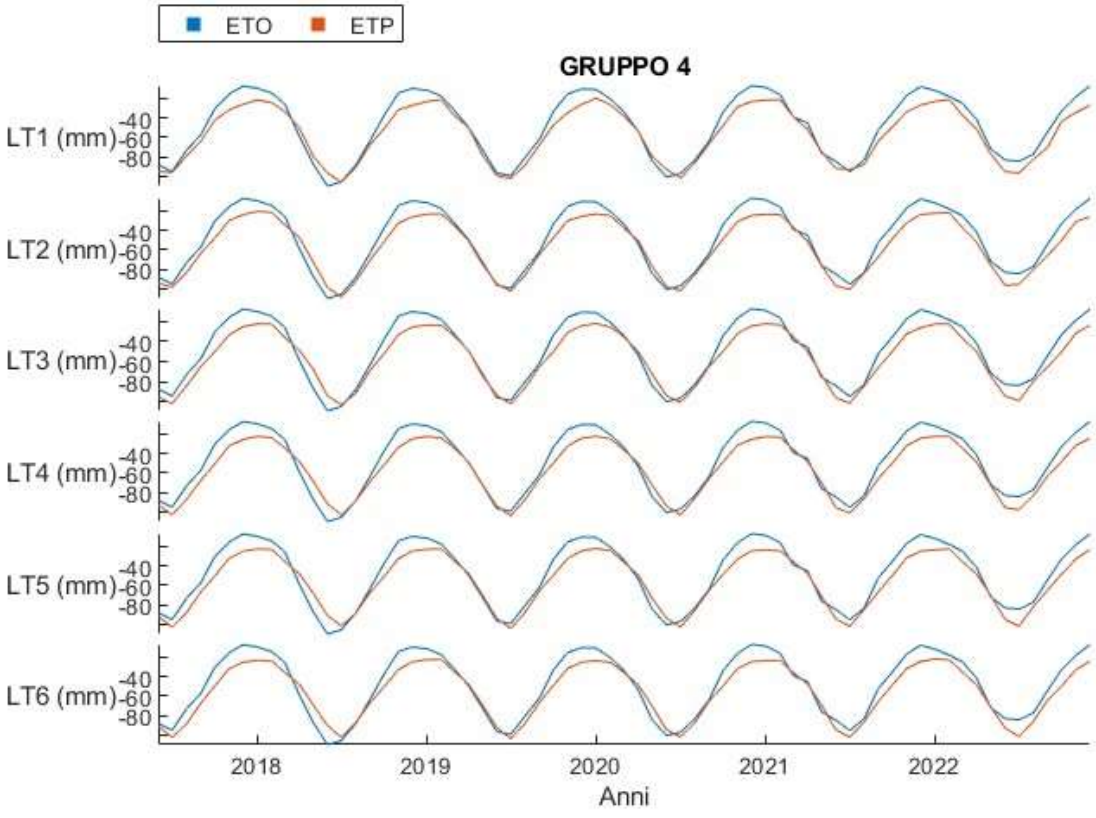

TO
TP
GRUPPO 6
LT1 (mm)
LT2 (mm)
LT3 (mm)
LT4 (mm)
LT5 (mm)
LT6 (mm)
10
5
0
2018
2019
2020
2021
2022
Anni

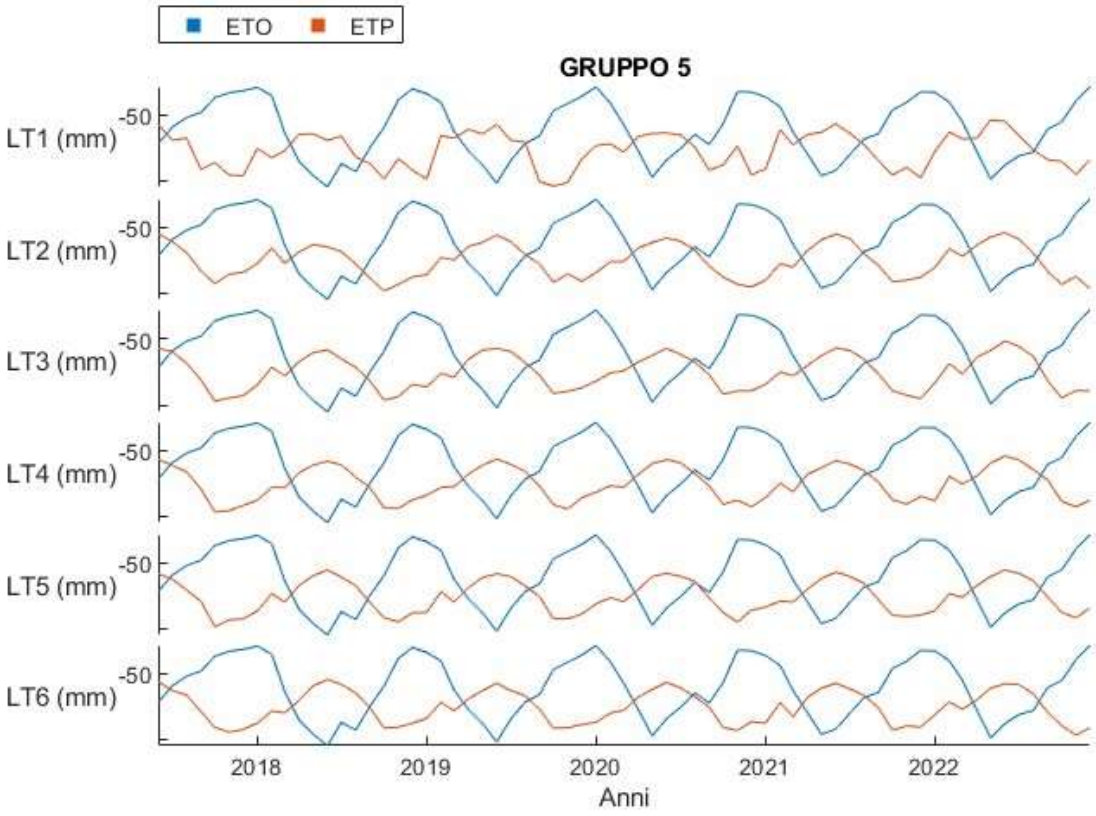
A.3 Previsioni di evapotraspirazione al variare del lead-time

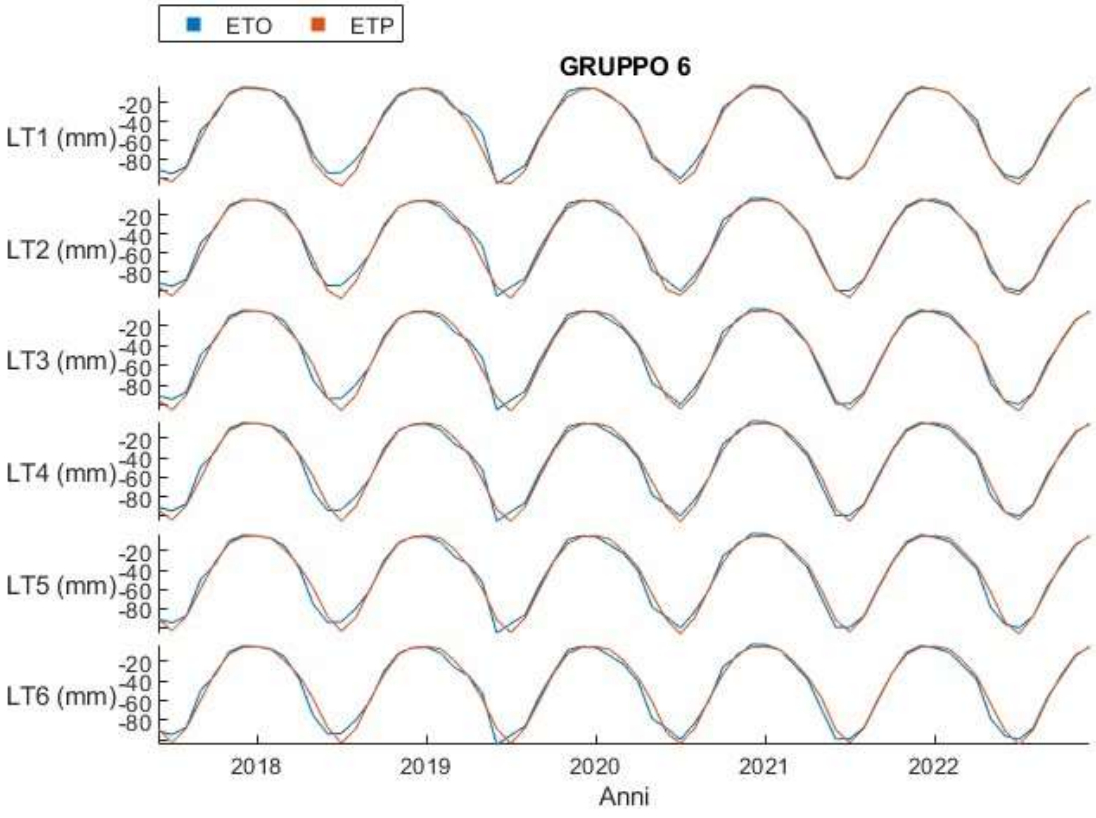




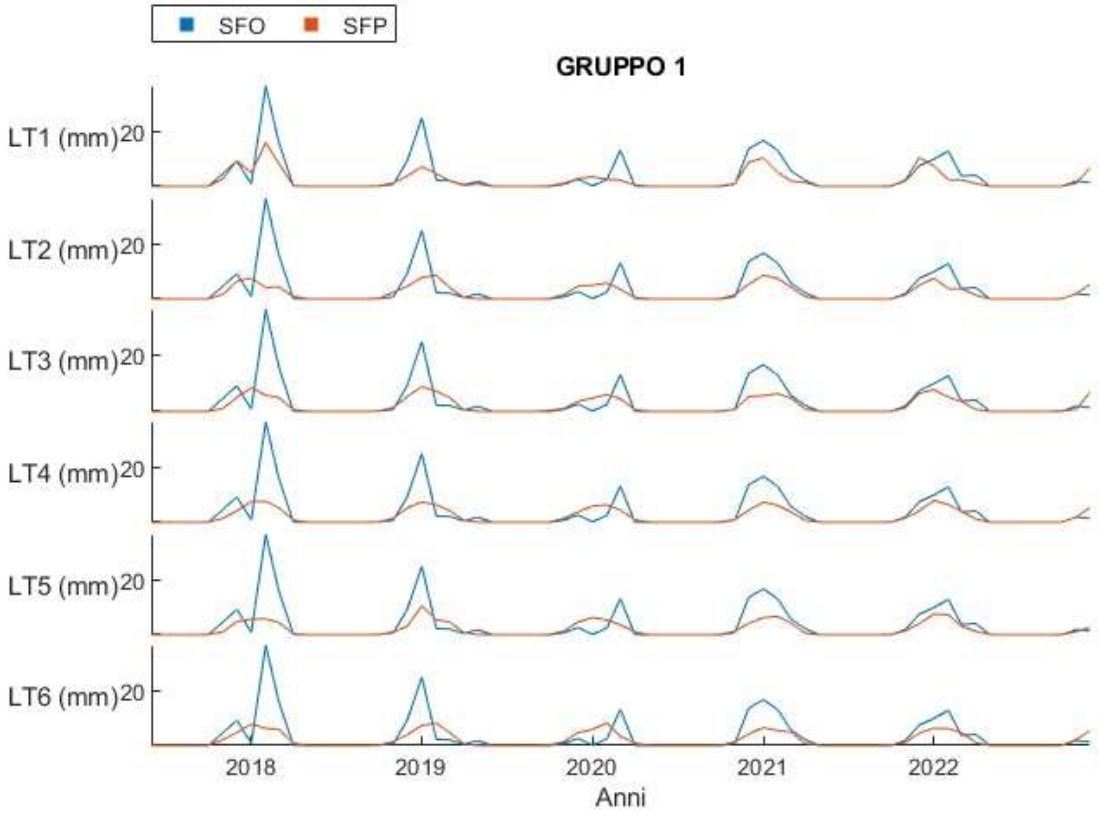

ETO
ETP
GRUPPO 3
LT1 (mm)
LT2 (mm)
LT3 (mm)
LT4 (mm)
LT5 (mm)
LT6 (mm)
-20
-40
-60
-80
2018
2019
2020
2021
2022
Anni

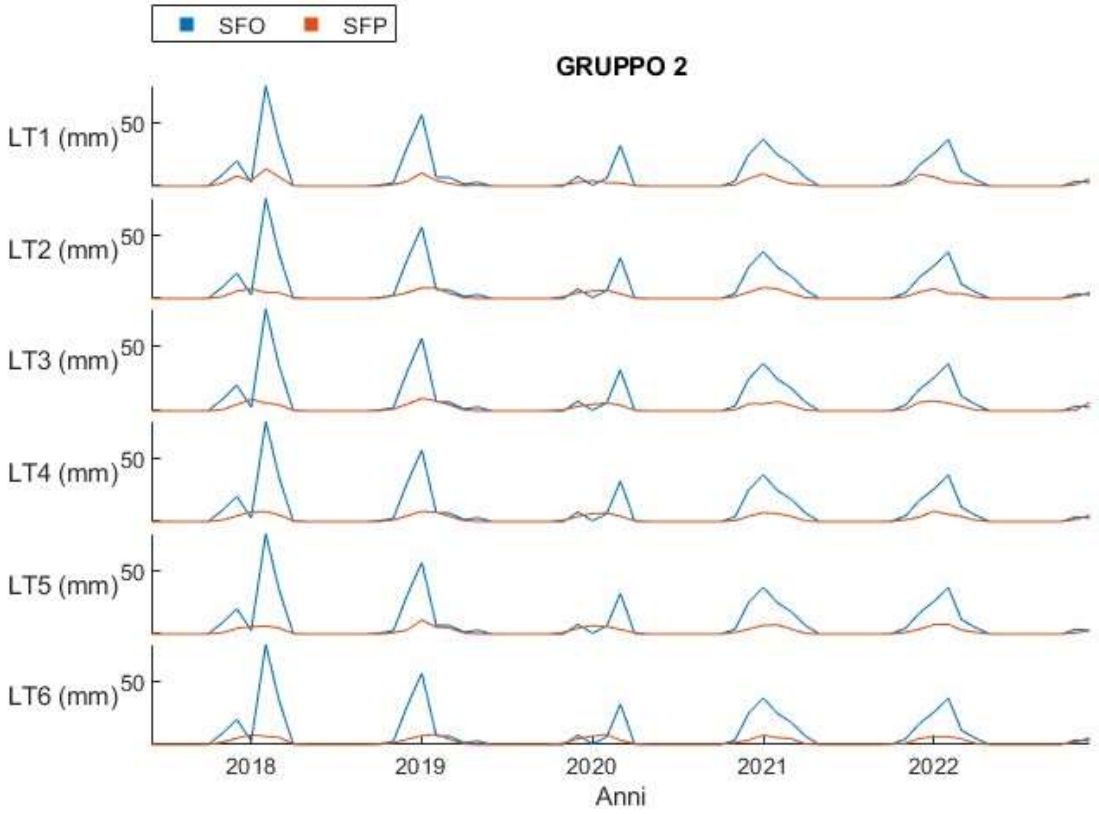

ETO
ETP
GRUPPO 4
LT1 (mm)
LT2 (mm)
LT3 (mm)
LT4 (mm)
LT5 (mm)
LT6 (mm)
-40
-60
-80
2018
2019
2020
2021
2022
Anni

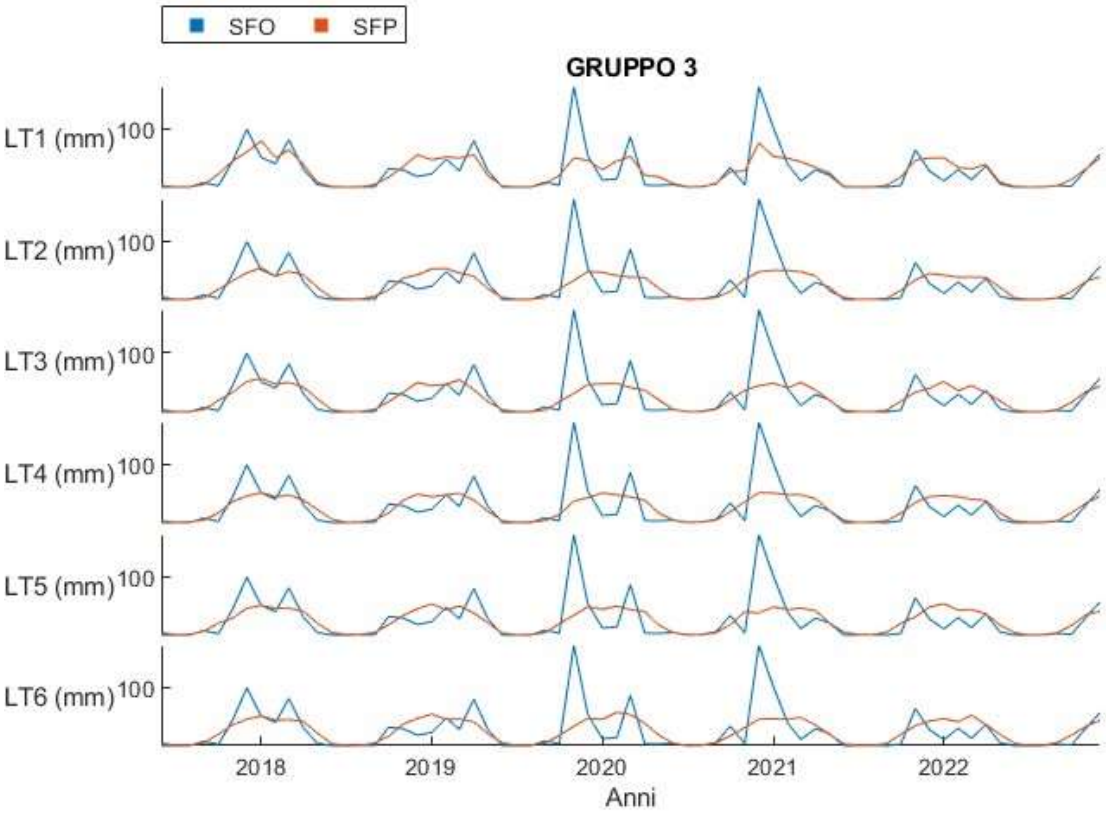

ETO
ETP
GRUPPO 5
LT1 (mm)
LT2 (mm)
LT3 (mm)
LT4 (mm)
LT5 (mm)
LT6 (mm)
-50
2018
2019
2020
2021
2022
Anni

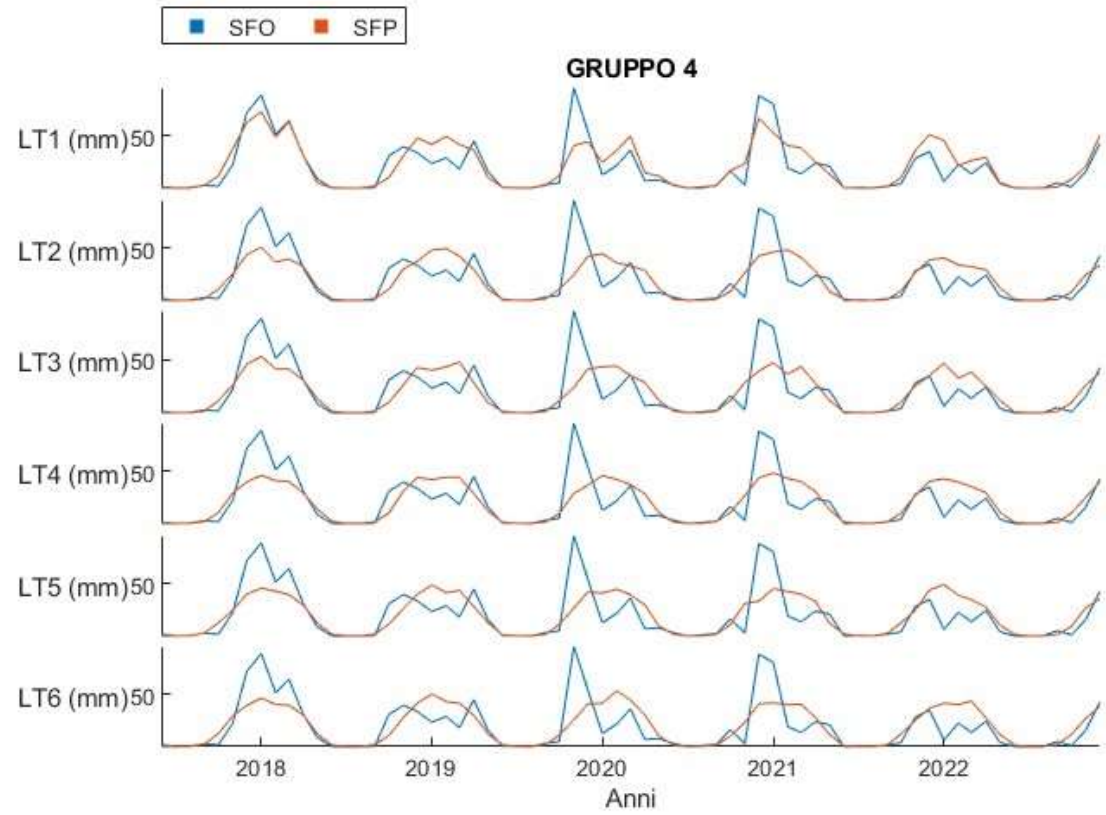

ETO
ETP
GRUPPO 6
LT1 (mm)
LT2 (mm)
LT3 (mm)
LT4 (mm)
LT5 (mm)
LT6 (mm)
-20
-40
-60
-80
2018
2019
2020
2021
2022
Anni

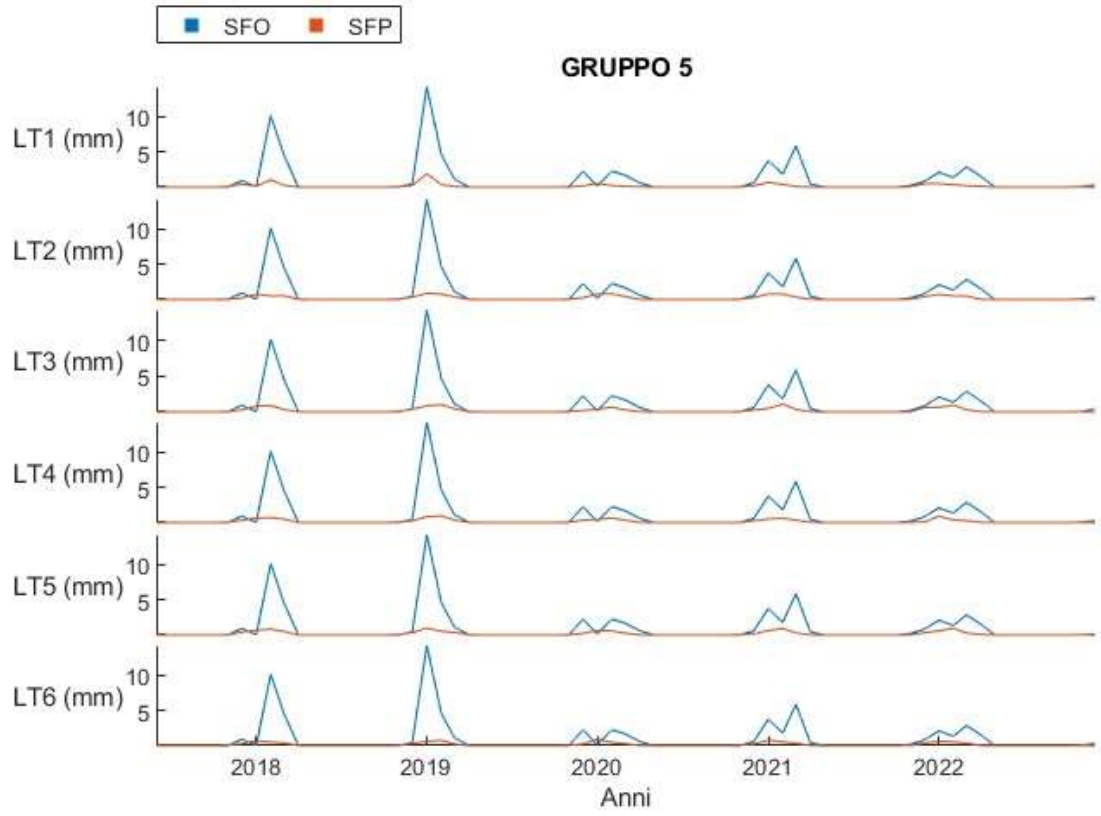
A.4 Previsioni di snowfall al variare del lead-time

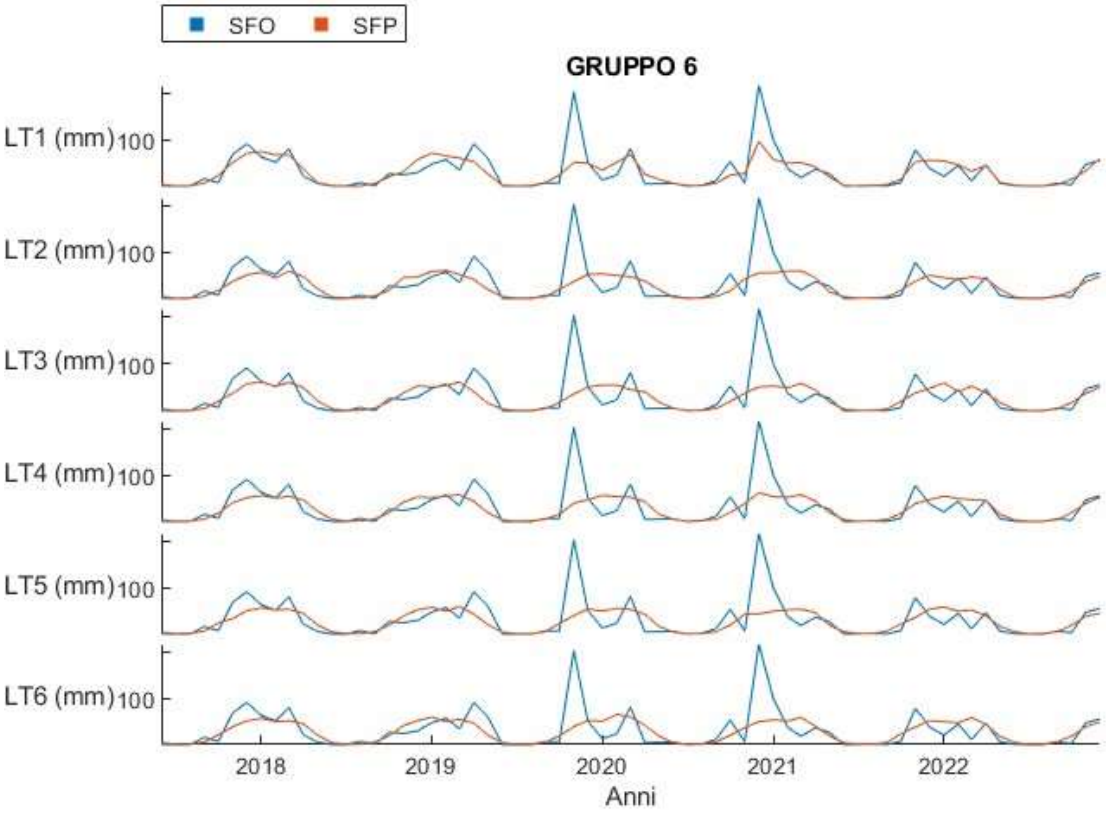




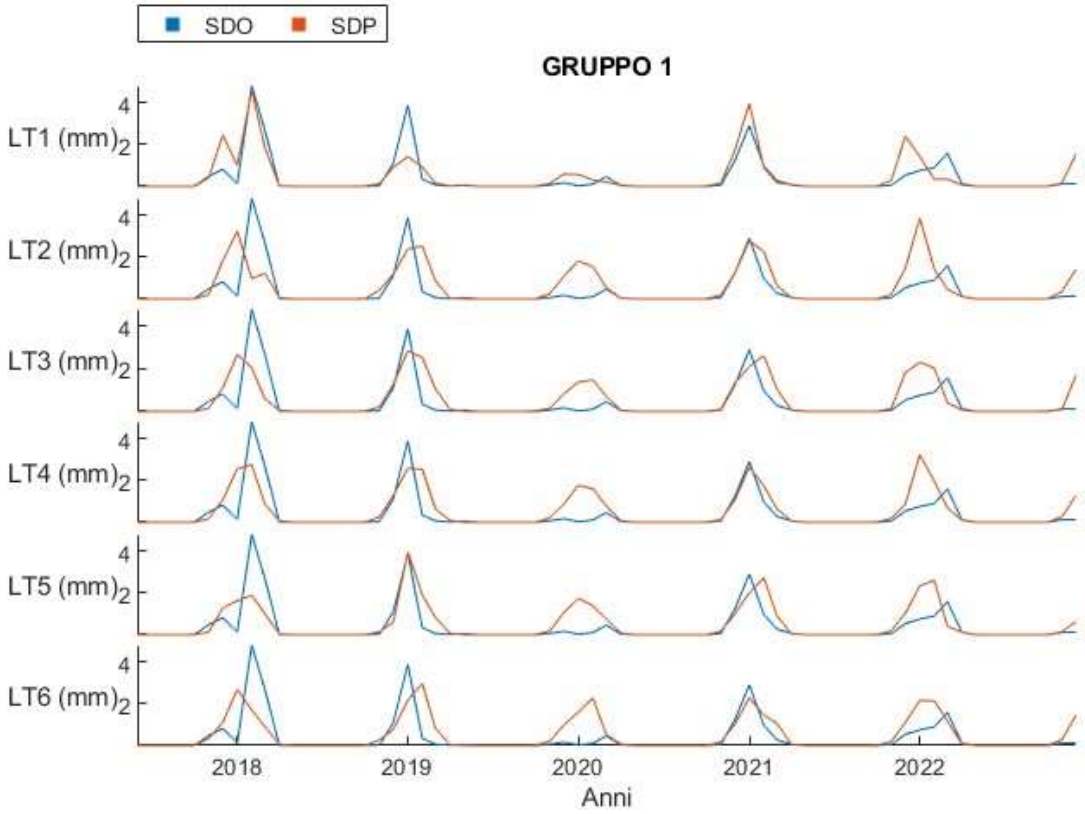

SFO
SFP
GRUPPO 3
LT1 (mm)
LT2 (mm)
LT3 (mm)
LT4 (mm)
LT5 (mm)
LT6 (mm)
100
2018
2019
2020
2021
2022
Anni

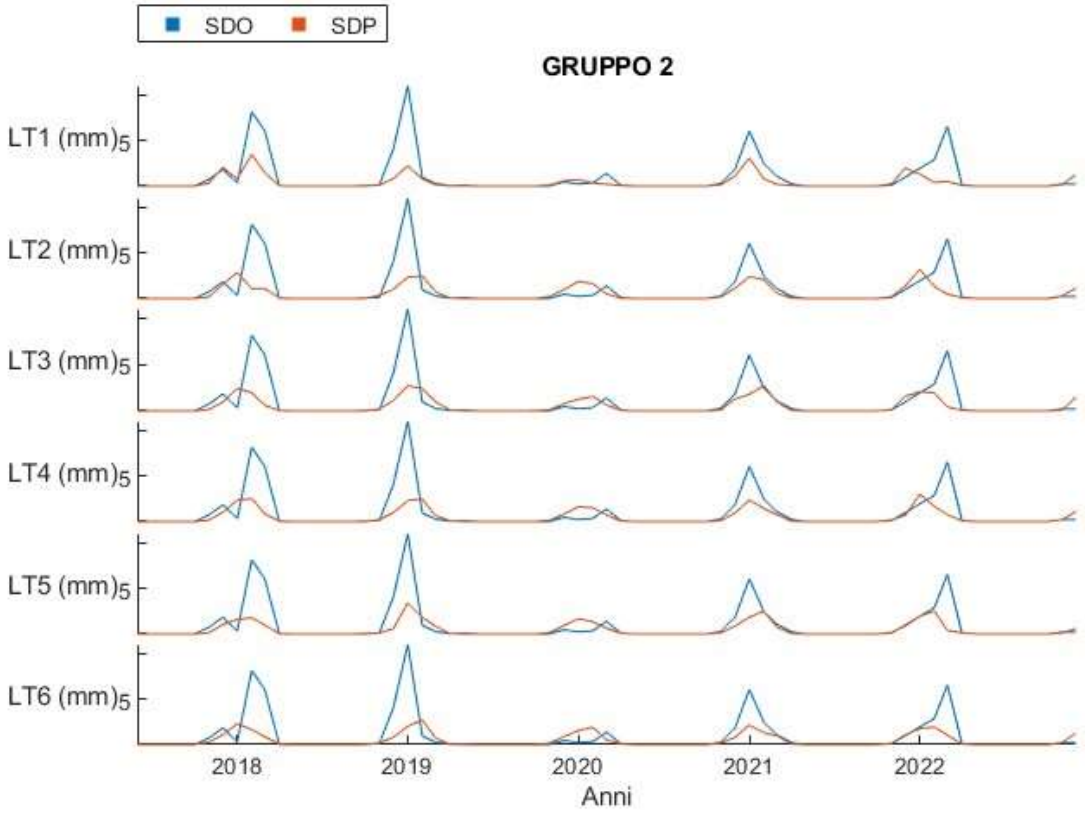

SFO
SFP
GRUPPO 4
LT1 (mm)
LT2 (mm)
LT3 (mm)
LT4 (mm)
LT5 (mm)
LT6 (mm)
50
2018
2019
2020
2021
2022
Anni

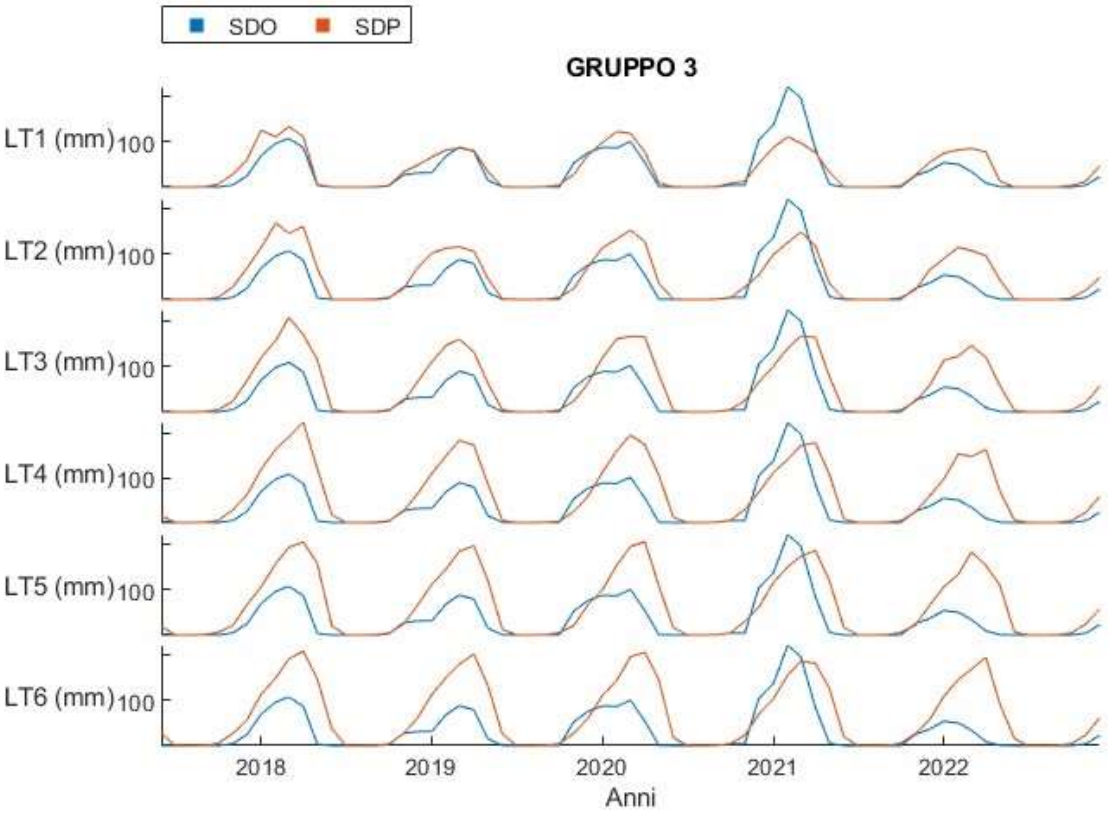

SFO
SFP
GRUPPO 5
LT1 (mm)
LT2 (mm)
LT3 (mm)
LT4 (mm)
LT5 (mm)
LT6 (mm)
10
5
2018
2019
2020
2021
2022
Anni

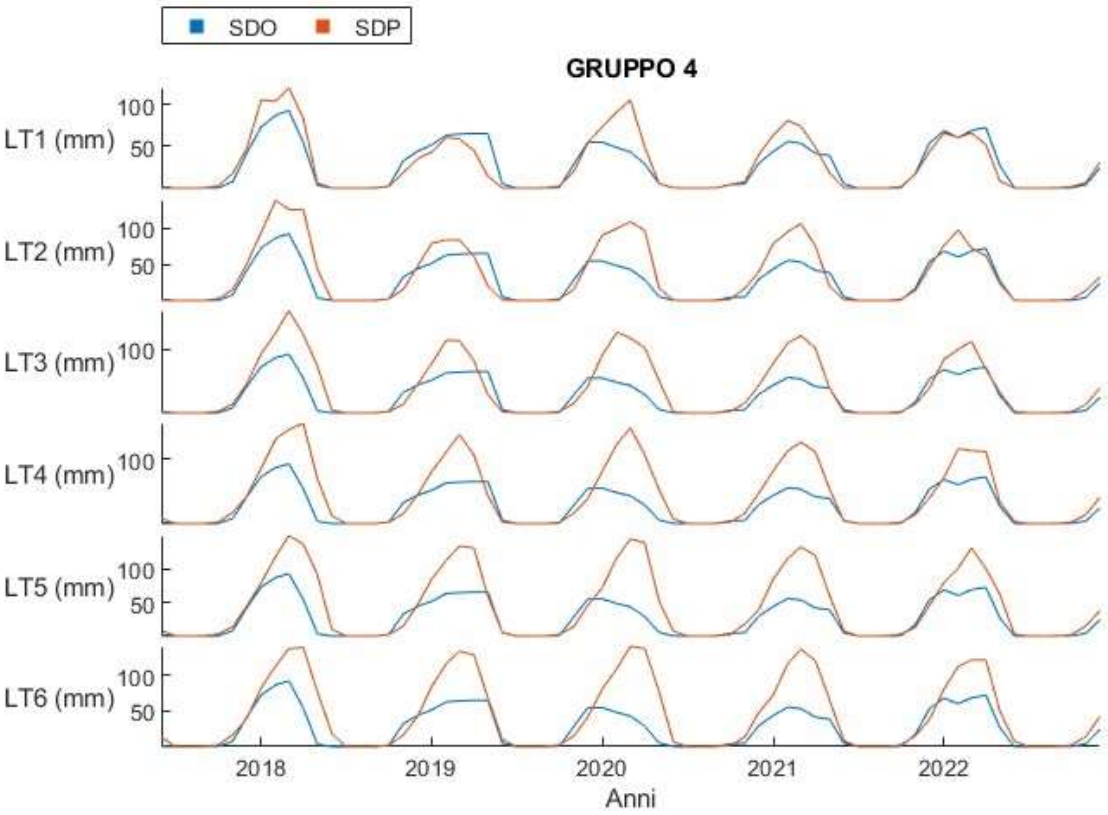

SFO
SFP
GRUPPO 6
LT1 (mm)
LT2 (mm)
LT3 (mm)
LT4 (mm)
LT5 (mm)
LT6 (mm)
100
2018
2019
2020
2021
2022
Anni

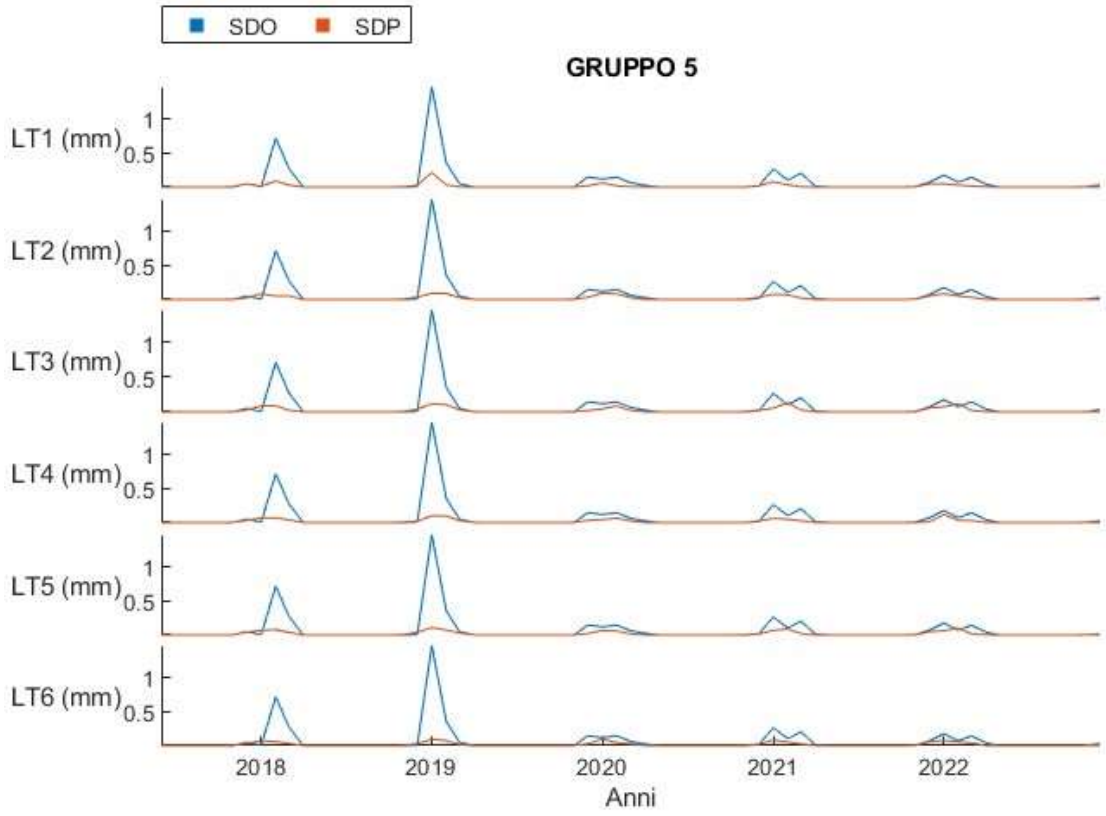
A.5 Previsioni di snow depth al variare del lead-time

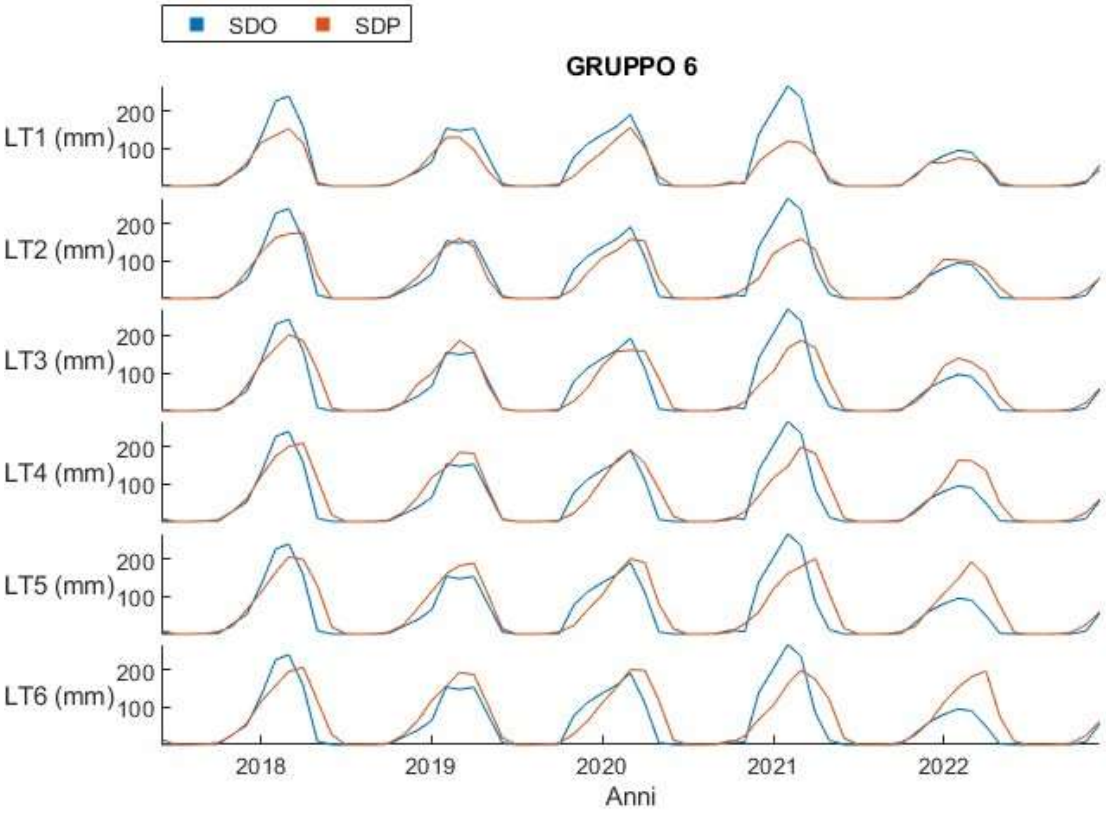





SDO
SDP
GRUPPO 3
LT1 (mm)
LT2 (mm)
LT3 (mm)
LT4 (mm)
LT5 (mm)
LT6 (mm)
100
2018
2019
2020
2021
2022
Anni


SDO
SDP
GRUPPO 4
LT1 (mm)
LT2 (mm)
LT3 (mm)
LT4 (mm)
LT5 (mm)
LT6 (mm)
100
50
2018
2019
2020
2021
2022
Anni


SDO
SDP
GRUPPO 5
LT1 (mm)
LT2 (mm)
LT3 (mm)
LT4 (mm)
LT5 (mm)
LT6 (mm)
1
0.5
2018
2019
2020
2021
2022
Anni


SDO
SDP
GRUPPO 6
LT1 (mm)
LT2 (mm)
LT3 (mm)
LT4 (mm)
LT5 (mm)
LT6 (mm)
200
100
2018
2019
2020
2021
2022
Anni

APPENDICE B

B.1 – Codice per l'elaborazione dei dati in formato NetCDF (dati ERA5)

```
#Esempio di elaborazione dei dati di evapotraspirazione
#Si carica il pacchetto necessario per l'elaborazione del formato NetCDF
library(ncdf4)
library(tidyverse)
#Estrazione dei dati contenuti nel file .nc
nc_data <- nc_open("ET_10_16.nc")
#Il dato viene visualizzato a schermo per identificare i nomi delle grandezze utili
print(nc_data)
#Si estraggono i dati singolarmente
lat <- ncvar_get(nc_data, "latitude")
lon <- ncvar_get(nc_data, "longitude")
time <- ncvar_get(nc_data, "time")
ET10_16_array <- ncvar_get(nc_data, "e")
#Il tempo viene trasformato in un formato facilmente leggibile
time2 <- time*3600
time_obs <- as.POSIXct(time2, origin = "1900-01-01", tz = "GMT")
#Si stampa il tempo a schermo per una verifica
range(time_obs)
#Il file con le coordinate dei punti viene creato e stampato su file di testo da inserire su QGIS
latlon <- cbind(1:(length(lat)*length(lon)),as.matrix(expand.grid(lon,lat)))
colnames(latlon) <- c('pos','lon','lat')
write.csv(as.data.frame(latlon), "latlon_ET90_99.txt", row.names=F)
#Si caricano i dati delle aree per ciascun punto, in questo caso per il gruppo 6 (ricavati tramite QGIS)
pesi <- read.table('Pesi6.txt',header=T,sep='\t')
#Si calcolano i pesi di ciascun punto
pesi$weight <- pesi$Area/sum(pesi$Area)
wmat <- matrix(0,nrow=length(lon),ncol=length(lat))
```

```
# Si effettua un ciclo per assegnare i pesi
for(i in 1:nrow(pesi)){
  pos_lon <- which(round(lon,2)==round(pesi$lon[i],2))
  pos_lat <- which(round(lat,2)==round(pesi$lat[i],2))
  wmat[pos_lon,pos_lat] <- pesi$weight[i]
}
#Si effettua la media pesata spaziale per l'area e si stampa il file .csv con i dati elaborati
ET10_16_bacino <- vector()
for(i in 1:dim(ET10_16_array)[3]) ET10_16_bacino[i] <- sum(ET10_16_array[,,i]*wmat)
write.csv(as.data.frame(ET10_16_bacino), "ET6_10_16.csv", row.names=F)
```

B.2 – Codice per l'elaborazione dei dati in formato NetCDF (previsioni meteorologiche)

```
#Esempio di elaborazione dei dati di evapotraspirazione per il gruppo 5
#Si carica il pacchetto necessario per l'elaborazione del formato NetCDF
library(ncdf4)
library(tidyverse)
#Si imposta un ciclo per elaborare tutti i dati relativi ad una grandezza per un singolo gruppo
for (a in 2017:2022){
  for (m in 1:12) {
#Vengono creati i nomi dei file di input e di output
  input<-paste("download",a,m,sep="_")
  input<-paste0(input,".nc")
  output<-paste("FC","ET5",m,a,sep="_")
  output<-paste0(output,".csv")
#Estrazione dei dati contenuti nel file .nc
nc_data <- nc_open(input)
#Il dato viene visualizzato a schermo per identificare i nomi delle grandezze utili
print(nc_data)
#Si estraggono i dati singolarmente
lat <- ncvar_get(nc_data, "latitude")
```

```
lon <- ncvar_get(nc_data, "longitude")
time <- ncvar_get(nc_data, "time")
ET_array <- ncvar_get(nc_data, "erate")
#Il tempo viene trasformato in un formato facilmente leggibile
time2 <- time*3600
time_obs <- as.POSIXct(time2, origin = "1900-01-01", tz = "GMT")
#Si stampa il tempo a schermo per una verifica
range(time_obs)
#Il file con le coordinate dei punti è stato preparato precedentemente come nel codice per i dati ERA5.
Si carica quindi il file con le aree estratte tramite QGIS
pesi <- read.table('Pesi_FC5.txt',header=T,sep='\t')
#Si calcolano i pesi di ciascun punto
pesi$weight <- pesi$Area/sum(pesi$Area)
wmat <- matrix(0,nrow=length(lon),ncol=length(lat))
# Si effettua un ciclo per assegnare i pesi
for(i in 1:nrow(pesi)){
  pos_lon <- which(round(lon,2)==round(pesi$lon[i],2))
  pos_lat <- which(round(lat,2)==round(pesi$lat[i],2))
  wmat[pos_lon,pos_lat] <- pesi$weight[i]
}
#Si effettua la media pesata spaziale per l'area e si stampa il file .csv con i dati elaborati
ET_bacino <- vector()
for(i in 1:dim(ET_array)[3]) ET_bacino[i] <- sum(ET_array[,,i]*wmat)
write.csv(as.data.frame(ET_bacino), output, row.names=F)
}}
```

B.3 – Codice del modello LSTM finale (creazione e test)

```
% Vengono estratti i dati dai file di testo preparati in precedenza
Data1=importdata("Dati_training.txt");
Data=Data1.data;
Data2=importdata("Dati_totale.txt");
```

```
Dtot=Data2.data;
N=length(Data);
Ntot=length(Dtot);
% Si impostano i mesi di memoria (lunghezza di input) n = w+1
w=5;
% Si imposta la struttura dei dati in input (cell)
Xtrain=cell(N-w,1);
Xtot=cell(Ntot-w,1);
Xtest=cell(Ntot-N-w,1);
% Numero della zona/gruppo
g=1;
% Numero di variabili esplicative
nf=5;
% Indice mobile per selezionare le variabili
k=6+g;
% Ciclo per calcolare la media e la deviazione standard delle variabili esplicative
for j=1:nf
   muI(j)=mean(Data(:,k));
   sdI(j)=std(Data(:,k));
   k=k+6;
end
% Media e deviazione standard della producibilità
muO=mean(Data(:,g));
sdO=std(Data(:,g));
% Ciclo per assegnare ad ogni cella la matrice corretta (training)
for i=(1+w):N
   k=6+g;
   for j=1:nf
      for l=0:w
      % Creazione della matrice e standardizzazione del dato
      D1(j,l+1)=(Data(i-w+l,k)-muI(j))/sdI(j);
      end
      k=k+6;
   end
   % Assegnazione della matrice nella cella di input
   Xtrain(i-w,:)={D1};
end
```

```
% Si ripete il procedimento per la totalità dei dati
for i=(1+w):Ntot
   k=6+g;
   for j=1:nf
      for l=0:w
      D2(j,l+1)=(Dtot(i-w+l,k)-muI(j))/sdI(j);
      % Si selezionano i dati per il test
      if i>N+w
         D3(j,l+1)=D2(j,l+1);
         Xtest(i-N-w,:)={D3};
      end
      end
      k=k+6;
   end
   Xtot(i-w,:)={D2};
end
% Si creano i vettori di target e di output per il test
Ttrain=(Data(w+1:N,g)-muO)/sdO;
Ttot=(Dtot(w+1:Ntot,g)-muO)/sdO;
for i=1:length(Xtest)
   Ttest(i,1)=Ttot(N+i);
end
% Si imposta l'architettura della rete neurale
numFeatures = nf;
numHiddenUnits = 128;
numResponses = 1;
layers = [ ...
   sequenceInputLayer(numFeatures)
   lstmLayer(numHiddenUnits,'OutputMode','last')
   fullyConnectedLayer(numResponses)
   regressionLayer];

maxEpochs = 25-w+numFeatures*3;
miniBatchSize = 27;

options = trainingOptions("adam", ...
   MiniBatchSize=miniBatchSize, ...
```

```
    MaxEpochs=maxEpochs, ...
    ExecutionEnvironment="cpu", ...
    Plots="none", ...
    Verbose=false);
% La rete viene addestrata
net = trainNetwork(Xtrain,Ttrain,layers,options);
% Si creano i vettori delle stime della rete su tutti i set
Ytest=predict(net,Xtest);
Ytrain=predict(net,Xtrain);
Ytot=predict(net,Xtot);
% Calcolo dell'RMSE
Etest=(Ttest-Ytest(:,1)).^2;
Etrain=(Ttrain-Ytrain(:,1)).^2;
Etest_q=sqrt(sum(Etest)/length(Ttest));
Etrain_q=sqrt(sum(Etrain)/length(Ttrain));
% Calcolo dell'index of agreement
dtest=1-(sum((Ttest-Ytest).^2)/sum((abs(Ytest-mean(Ttest))+abs(Ttest-mean(Ttest))).^2));
dtrain=1-(sum((Ttrain-Ytrain).^2)/sum((abs(Ytrain-mean(Ttrain))+abs(Ttrain-mean(Ttrain))).^2));
dtot=1-(sum((Ttot-Ytot).^2)/sum((abs(Ytot-mean(Ttot))+abs(Ttot-mean(Ttot))).^2));
% Calcolo dell'indice di efficienza di Nash-Sutcliffe
muY=mean(Ttrain);
NStest=1-(sum((Ttest-Ytest).^2)/sum((Ttest-muY).^2));
NStrain=1-(sum((Ttrain-Ytrain).^2)/sum((Ttrain-muY).^2));
% Scatter plot e grafico con serie temporale
tiledlayout (1,2)
nexttile
plot(Ttest,Ytest,".")
hold on
plot(Ttest,Ttest)
hold off
xlabel("Osservazioni")
ylabel("Previsione")
nexttile
plot(1:length(Ttest),Ttest,"b")
hold on
plot(1:length(Ttest),Ytest,"r")
hold off
```

```
xlabel("Mesi")
ylabel("Osservato e previsto")
legend ('Osservato','Previsto')
% Stampa a schermo degli indicatori
Stampa=sprintf("Gruppo %d\nMesi in input %d\nRMSE training\t%f\nD training\t%f\nNSE
training\t%f\nRMSE test\t%f\nD test\t%f\nNSE
test\t%f",g,w+1,Etrain_q,dtrain,NStrain,Etest_q,dtest,NStest);
disp(Stampa)
```

B.4 – Codice del modello LSTM finale (previsione)

```
% Vengono estratti i dati dai file di testo preparati in precedenza
Data1=importdata("Dati_training.txt");
Data=Data1.data;
Data2=importdata("Dati_totale.txt");
Dtot=Data2.data;
Data3=importdata("Prev_gruppi_nuovo.txt");
Dprev=Data3.data;
Data4=importdata("Storico_17_22.txt");
Dstor=Data4.data;
N=length(Data);
Ntot=length(Dtot);
Nprev=size(Dprev,1);
% Si impostano i mesi di memoria (lunghezza di input) n  = w+1
w=5;
% Si impostano i mesi di lead-time che si vogliono stimare
f=6;
% Si imposta la struttura dei dati in input (cell)
Xtrain=cell(N-w,1);
Xtot=cell(Ntot-w,1);
Xtest=cell(Nprev-w,1);
% Numero della zona/gruppo
g=5;
% Numero di variabili esplicative
nf=5;
% Indice mobile per selezionare le variabili
```

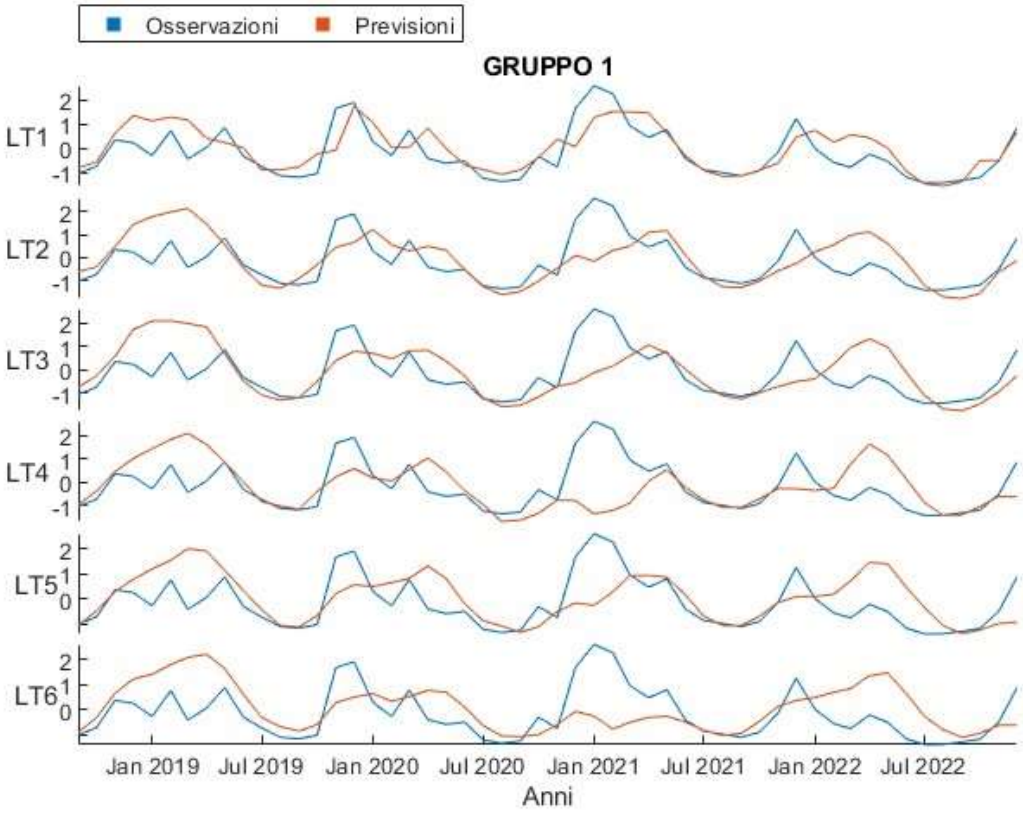
```
k=6+g;
% Ciclo per calcolare la media e la deviazione standard delle variabili esplicative
for j=1:nf
   muI(j)=mean(Data(:,k));
   sdI(j)=std(Data(:,k));
   k=k+6;
end
% Media e deviazione standard della producibilità
muO=mean(Data(:,g));
sdO=std(Data(:,g));
% Ciclo per assegnare ad ogni cella la matrice corretta (training)
for i=(1+w):N
   k=6+g;
   for j=1:nf
      for l=0:w
      % Creazione della matrice e standardizzazione del dato
      D1(j,l+1)=(Data(i-w+l,k)-muI(j))/sdI(j);
      end
      k=k+6;
   end
   % Assegnazione della matrice nella cella di input
   Xtrain(i-w,:)={D1};
end
% Si ripete il procedimento per la totalità dei dati
for i=(1+w):Ntot
   k=6+g;
   for j=1:nf
      for l=0:w
      D2(j,l+1)=(Dtot(i-w+l,k)-muI(j))/sdI(j);
      end
      k=k+6;
   end
   Xtot(i-w,:)={D2};
end
% Si crea l'input per il test (matrice contenente sia osservazioni che previsioni)
for i=(1+w):Nprev
   for j=1:nf
```

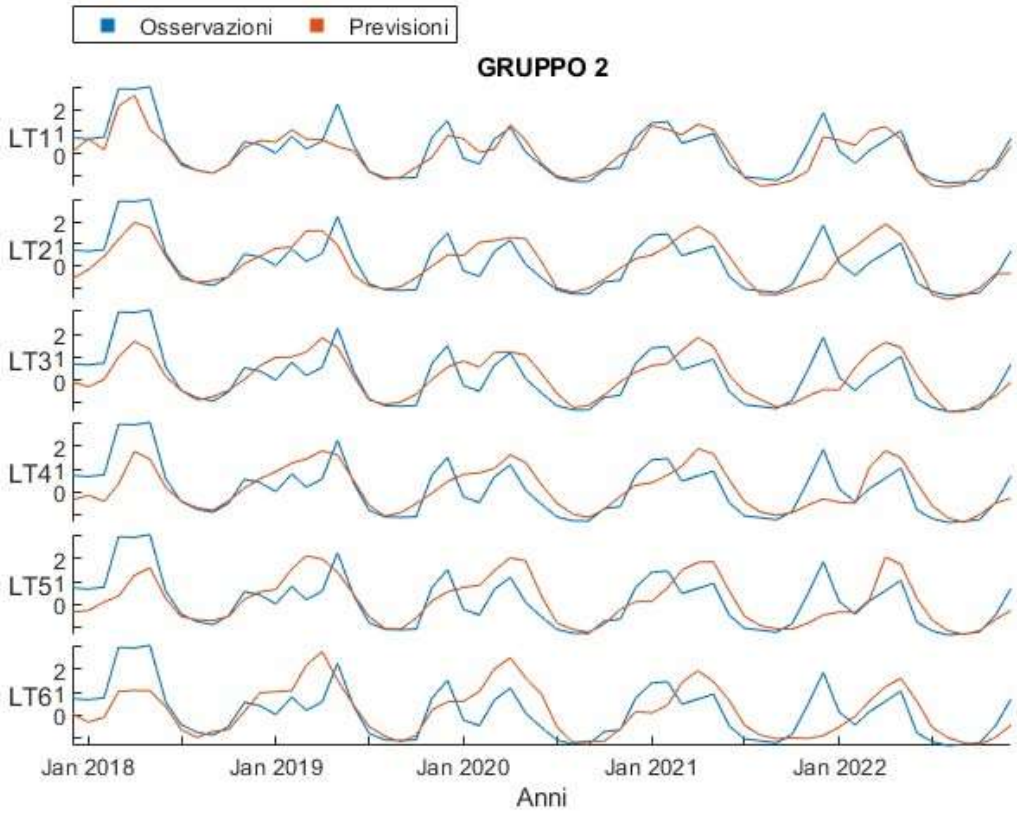
```
        for l=0:w
            if l>(w-f)
                D3(j,l+1)=(Dprev(i-w+l,(l-w+f-1)*nf+(g-1)*nf*6+j)-muI(j))/sdI(j);
                else
                D3(j,l+1)=(Dstor(i-w+l,6*(j-1)+g)-muI(j))/sdI(j);
            end
        end
    end
    Xtest(i-w,:)={D3};
end
% Si creano i vettori di target e di output per il test
Ttrain=(Data(w+1:N,g)-muO)/sdO;
Ttot=(Dtot(w+1:Ntot,g)-muO)/sdO;
for i=1:length(Xtest)
    Ttest(i,1)=Ttot(N+i);
end
% Si imposta l'architettura della rete neurale
numFeatures = nf;
numHiddenUnits = 128;
numResponses = 1;

layers = [ ...
    sequenceInputLayer(numFeatures)
    lstmLayer(numHiddenUnits,'OutputMode','last')
    fullyConnectedLayer(numResponses)
    regressionLayer];

maxEpochs = 25-w+numFeatures*3;
miniBatchSize = 27;

options = trainingOptions("adam", ...
    MiniBatchSize=miniBatchSize, ...
    MaxEpochs=maxEpochs, ...
    ExecutionEnvironment="cpu", ...
    Plots="none", ...
    Verbose=false);
% La rete viene addestrata
```

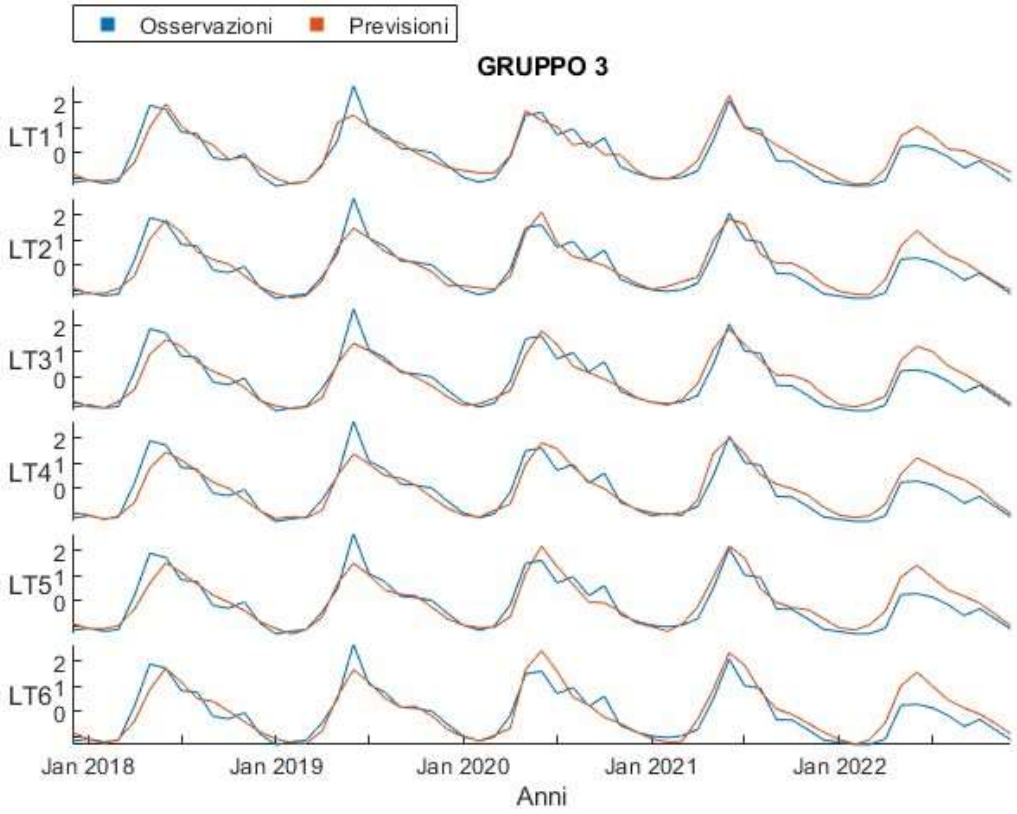
```
net = trainNetwork(Xtrain,Ttrain,layers,options);
% Si creano i vettori delle stime della rete su tutti i set
Ytest=predict(net,Xtest);
Ytrain=predict(net,Xtrain);
Ytot=predict(net,Xtot);
% Per il calcolo degli indicatori e per i plot dei risultati si può utilizzare lo stesso codice usato in precedenza
```

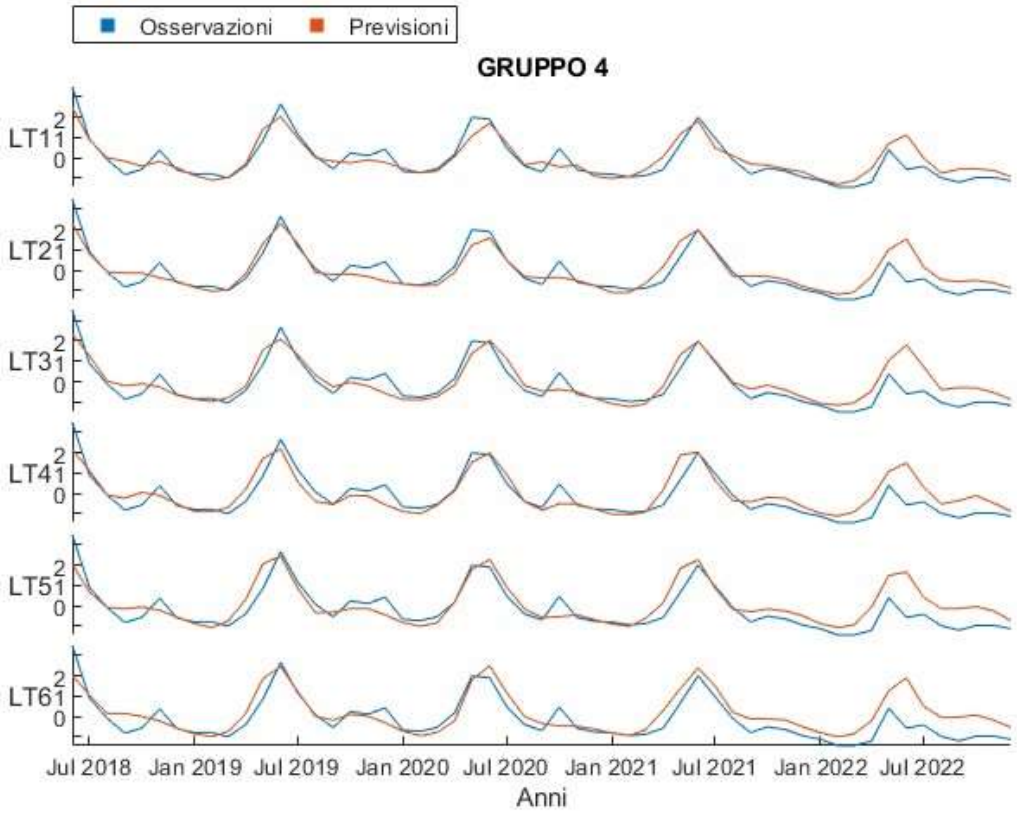
APPENDICE C

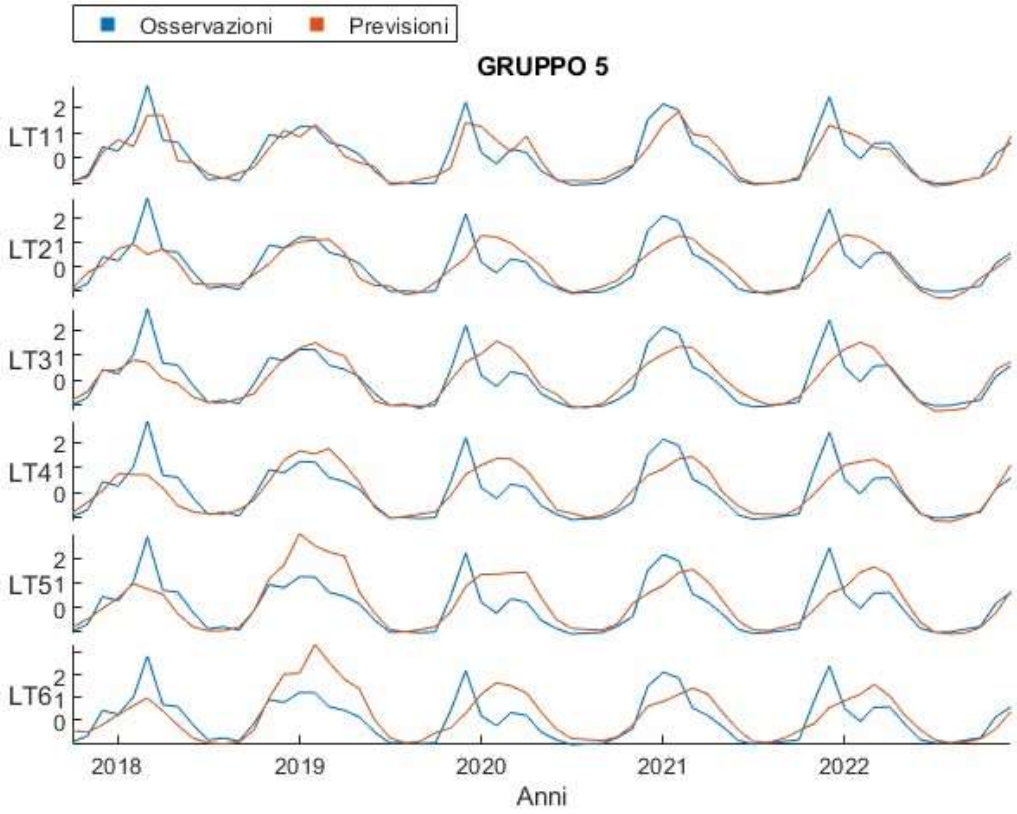
C.1 – Risultati del modello LSTM senza forzanti meteorologiche (previsione)

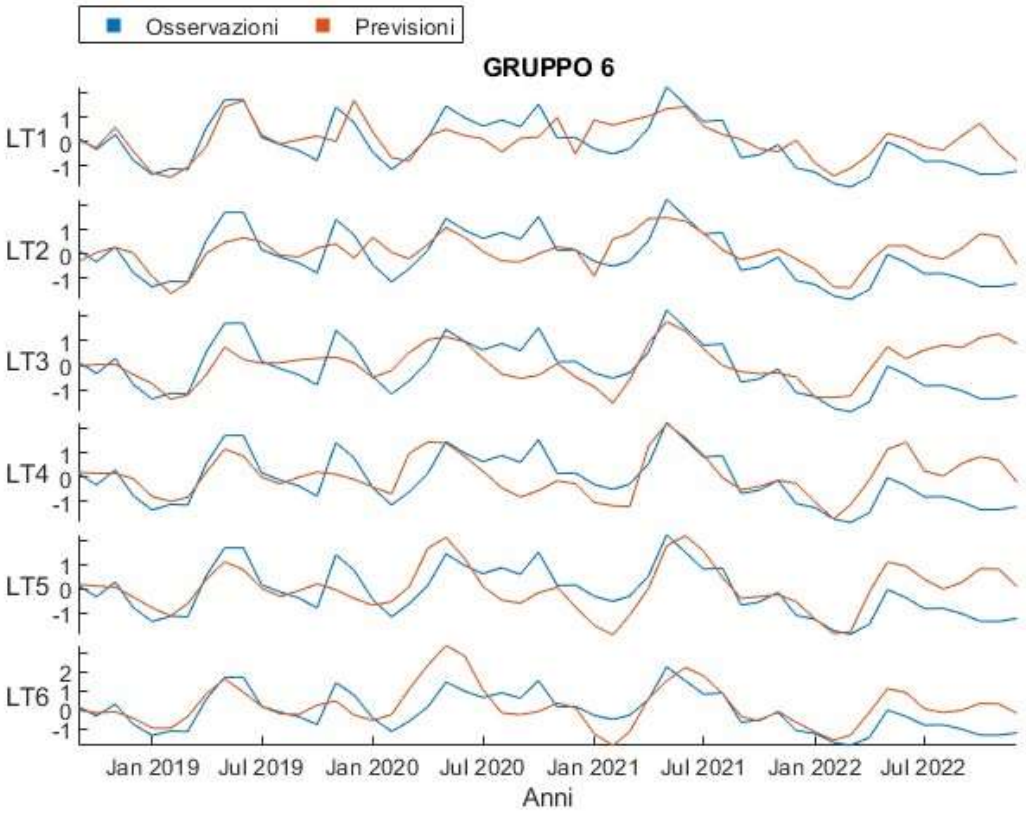



Osservazioni
Previsioni
GRUPPO 2
LT1
LT2
LT3
LT4
LT5
LT6
2
1
0
Jan 2018
Jan 2019
Jan 2020
Jan 2021
Jan 2022
Anni

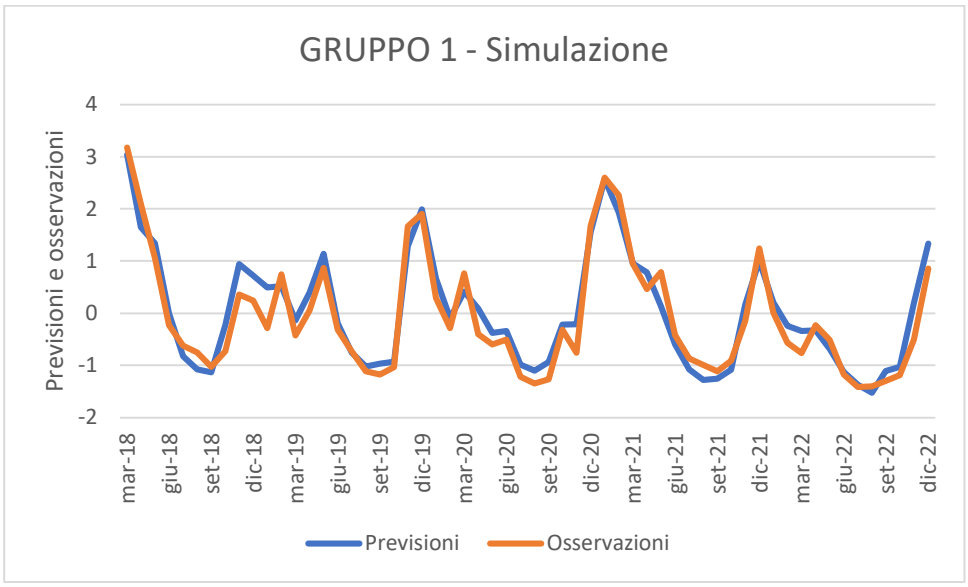

Osservazioni
Previsioni
GRUPPO 3
LT1
LT2
LT3
LT4
LT5
LT6
2
1
0
Jan 2018
Jan 2019
Jan 2020
Jan 2021
Jan 2022
Anni

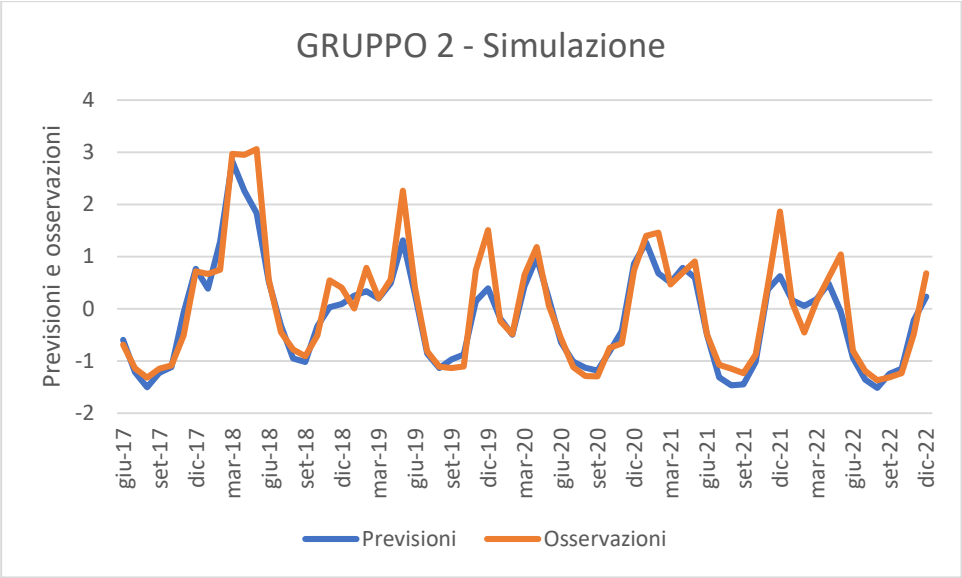

Osservazioni
Previsioni
GRUPPO 4
LT11
LT21
LT31
LT41
LT51
LT61
2
1
0
Jul 2018
Jan 2019
Jul 2019
Jan 2020
Jul 2020
Jan 2021
Jul 2021
Jan 2022
Jul 2022
Anni

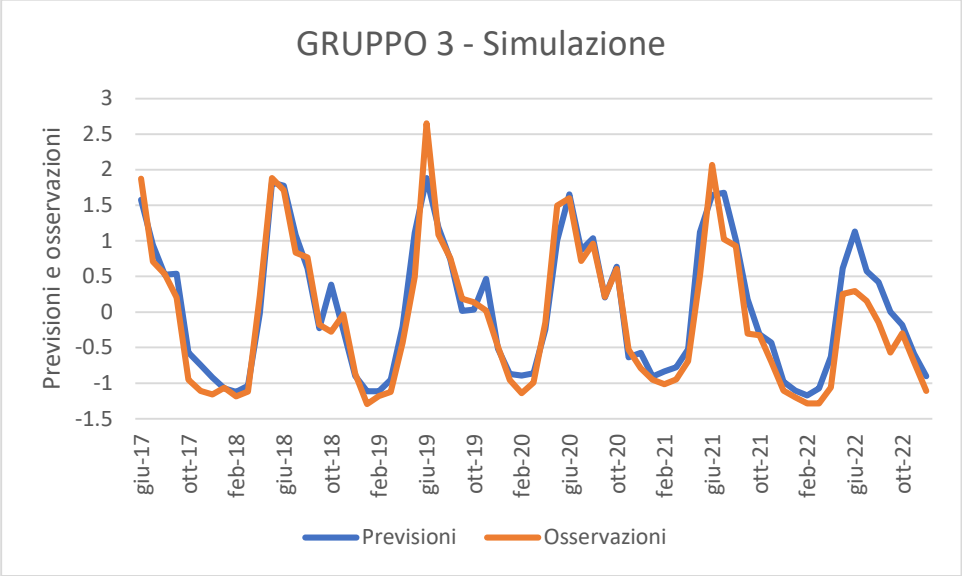

Osservazioni
Previsioni
GRUPPO 5
LT11
LT21
LT31
LT41
LT51
LT61
2
1
0
2018
2019
2020
2021
2022
Anni

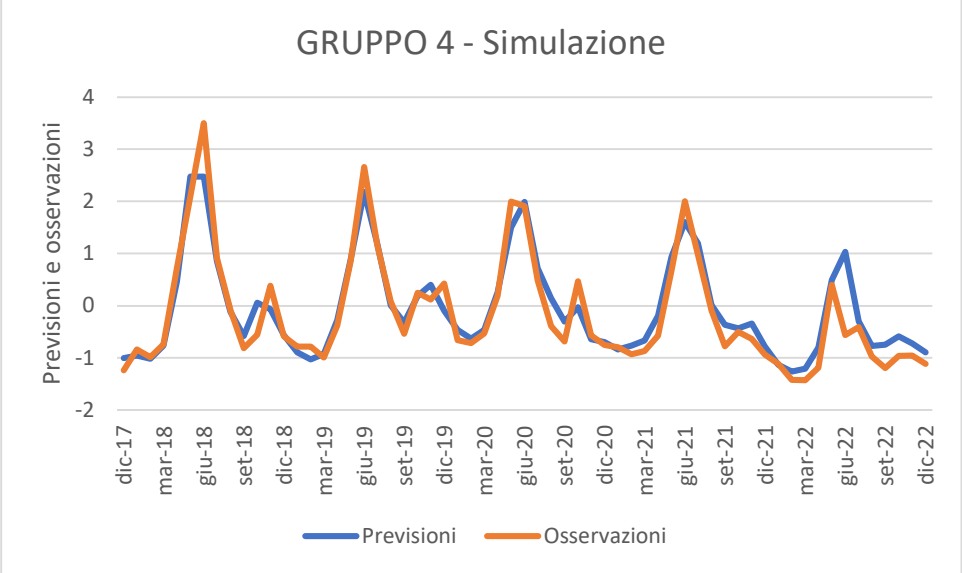


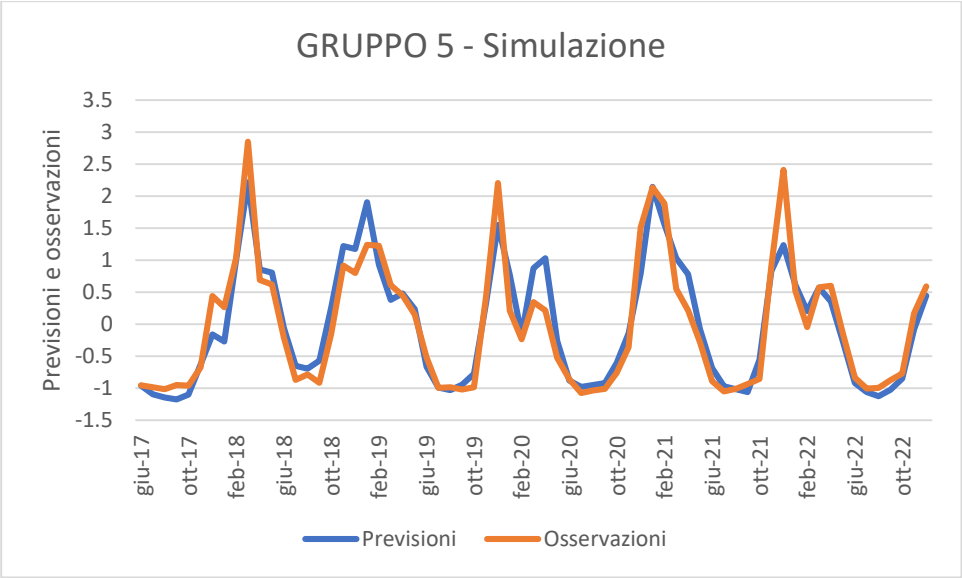
C.2 – Risultati del modello LSTM finale (test con dati osservati)

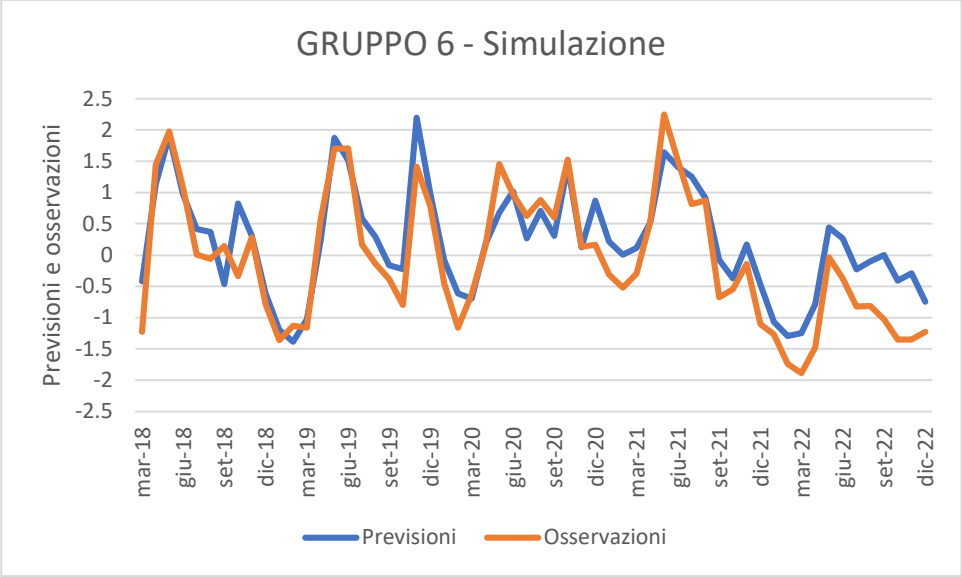



GRUPPO 2 - Simulazione
Previsioni e osservazioni
4
3
2
1
0
-1
-2
giu-17
set-17
dic-17
mar-18
giu-18
set-18
dic-18
mar-19
giu-19
set-19
dic-19
mar-20
giu-20
set-20
dic-20
mar-21
giu-21
set-21
dic-21
mar-22
giu-22
set-22
dic-22
Previsioni
Osservazioni

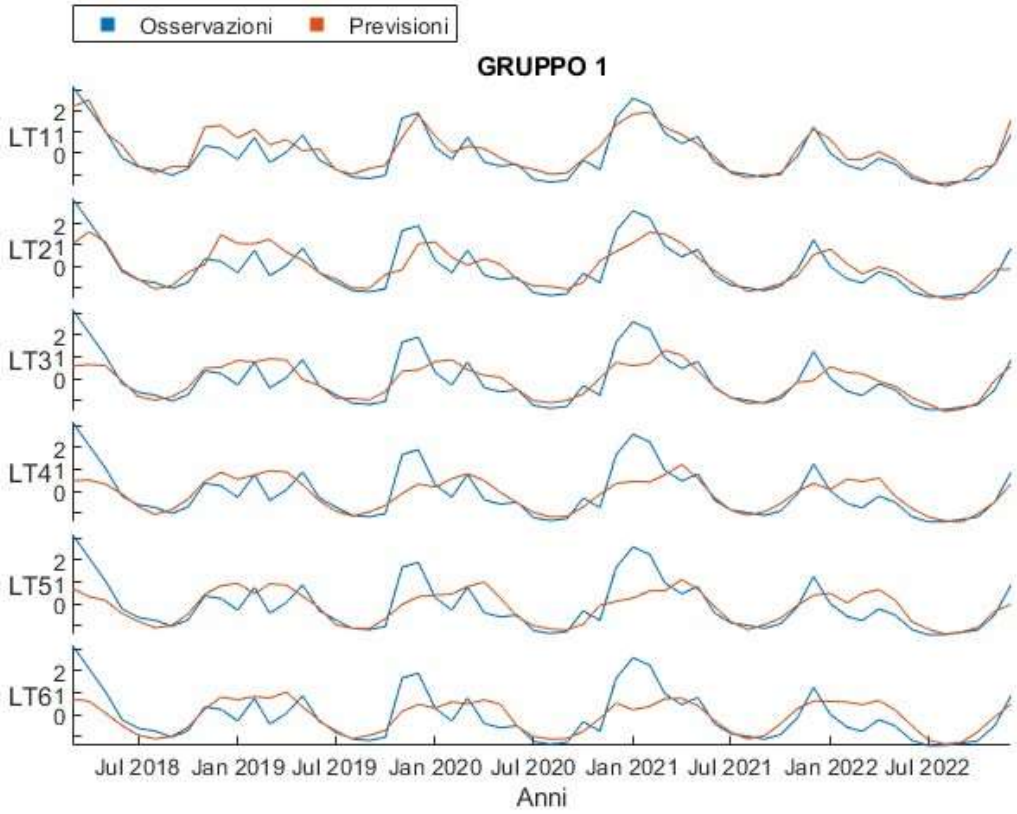

GRUPPO 3 - Simulazione
Previsioni e osservazioni
3
2.5
2
1.5
1
0.5
0
-0.5
-1
-1.5
giu-17
ott-17
feb-18
giu-18
ott-18
feb-19
giu-19
ott-19
feb-20
giu-20
ott-20
feb-21
giu-21
ott-21
feb-22
giu-22
ott-22
Previsioni
Osservazioni

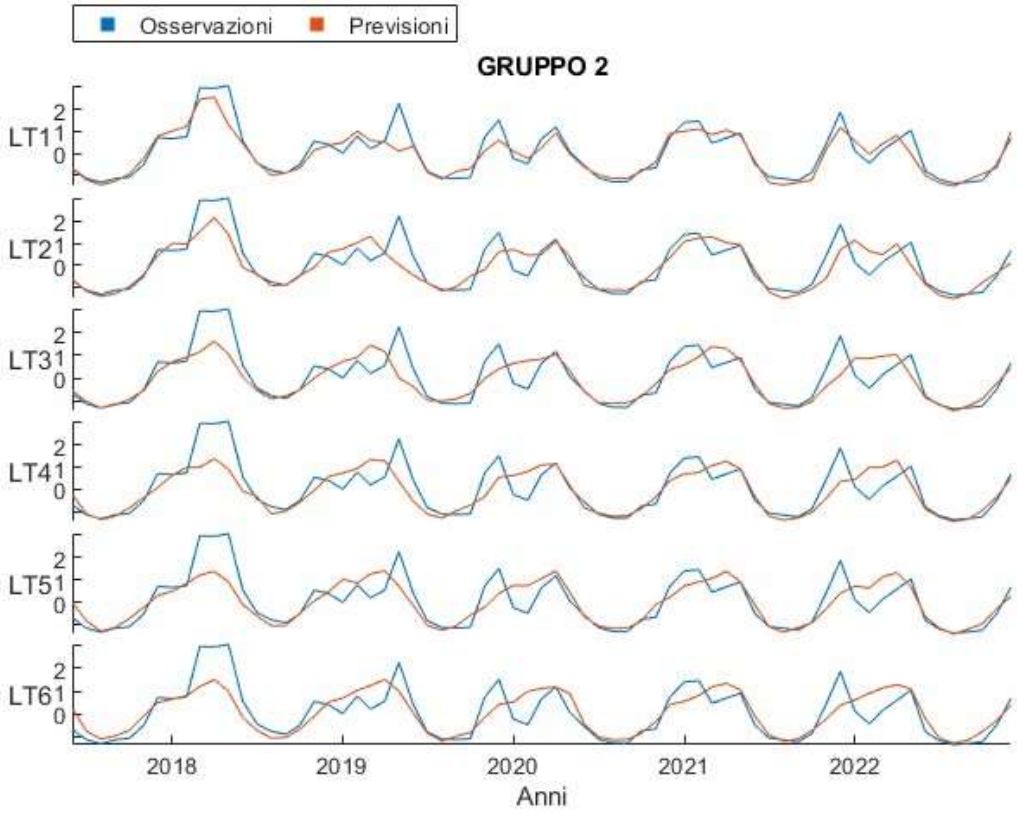

GRUPPO 4 - Simulazione
Previsioni e osservazioni
4
3
2
1
0
-1
-2
dic-17
mar-18
giu-18
set-18
dic-18
mar-19
giu-19
set-19
dic-19
mar-20
giu-20
set-20
dic-20
mar-21
giu-21
set-21
dic-21
mar-22
giu-22
set-22
dic-22
Previsioni
Osservazioni

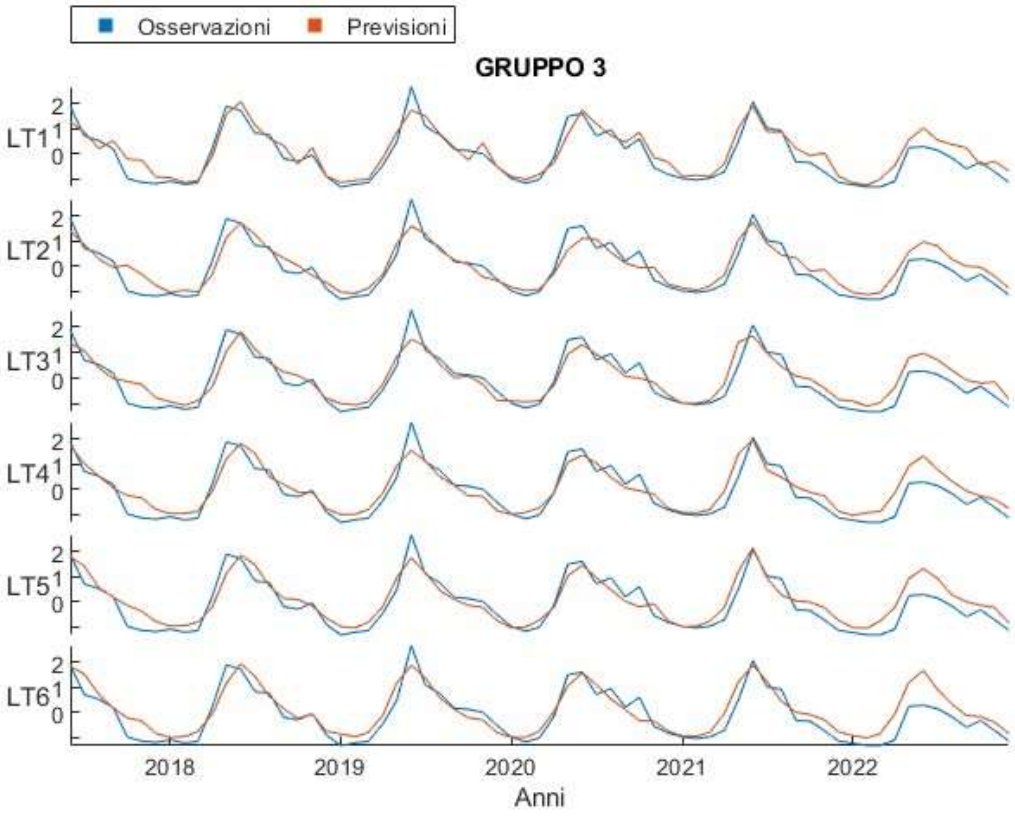

GRUPPO 5 - Simulazione
Previsioni e osservazioni
3.5
3
2.5
2
1.5
1
0.5
0
-0.5
-1
-1.5
giu-17
ott-17
feb-18
giu-18
ott-18
feb-19
giu-19
ott-19
feb-20
giu-20
ott-20
feb-21
giu-21
ott-21
feb-22
giu-22
ott-22
Previsioni
Osservazioni

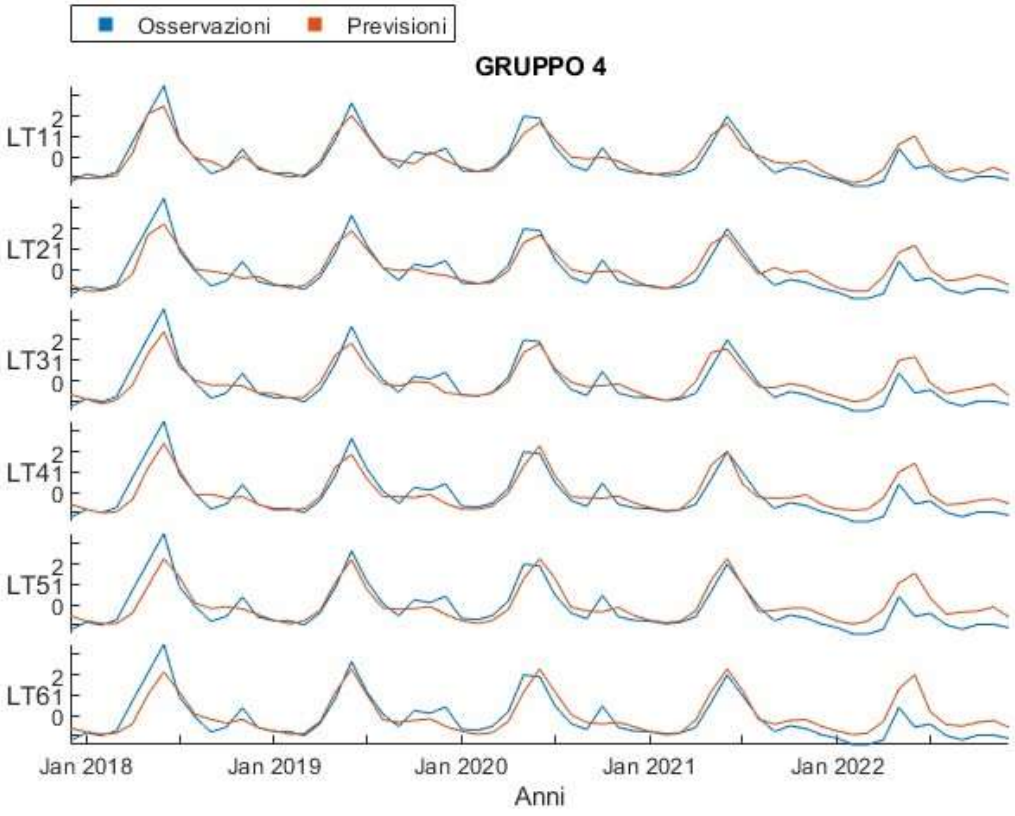

GRUPPO 6 - Simulazione
Previsioni e osservazioni
2.5
2
1.5
1
0.5
0
-0.5
-1
-1.5
-2
-2.5
mar-18
giu-18
set-18
dic-18
mar-19
giu-19
set-19
dic-19
mar-20
giu-20
set-20
dic-20
mar-21
giu-21
set-21
dic-21
mar-22
giu-22
set-22
dic-22
Previsioni
Osservazioni

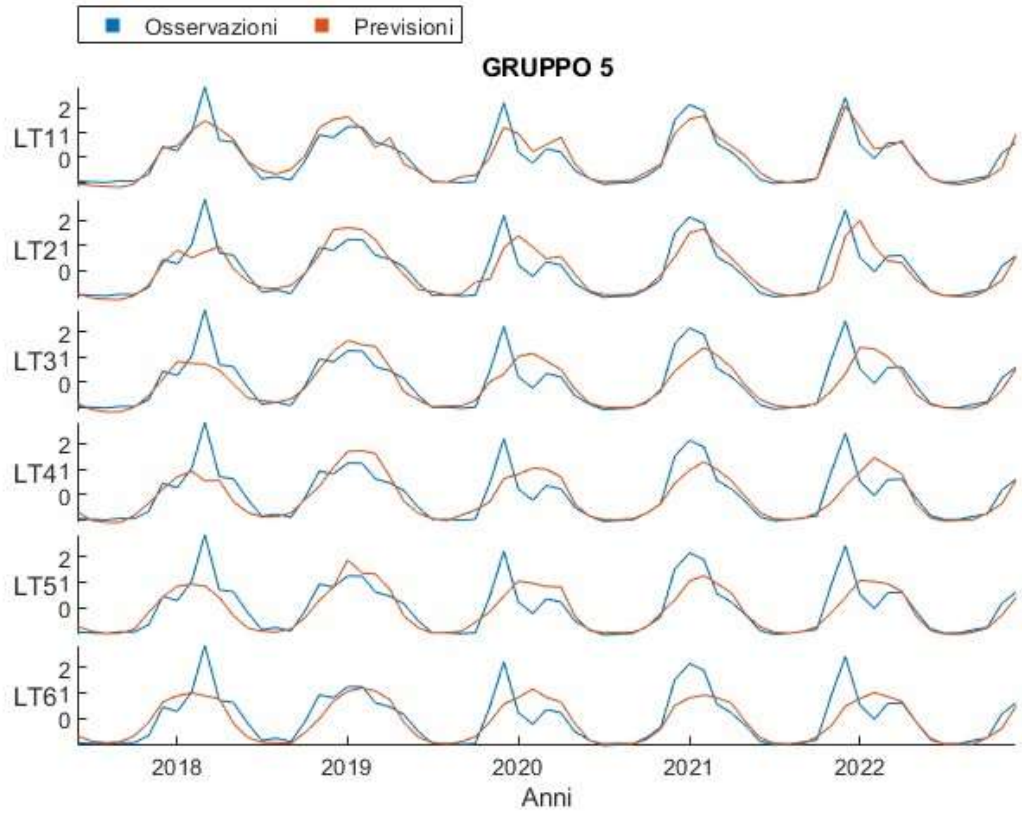
C.3 – Risultati del modello LSTM finale (previsione)

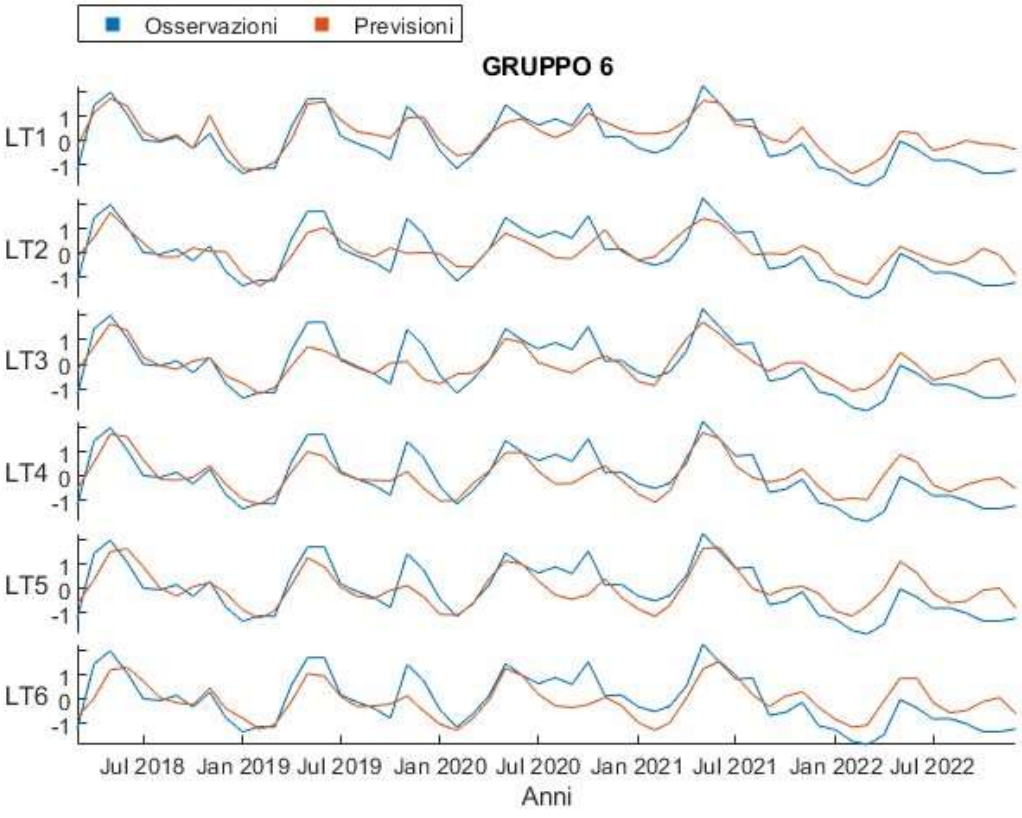





Osservazioni
Previsioni
GRUPPO 3
LT1
LT2
LT3
LT4
LT5
LT6
2
1
0
2018
2019
2020
2021
2022
Anni


Osservazioni
Previsioni
GRUPPO 4
LT1
LT2
LT3
LT4
LT5
LT6
2
1
0
Jan 2018
Jan 2019
Jan 2020
Jan 2021
Jan 2022
Anni


Osservazioni
Previsioni
GRUPPO 5
LT11
LT21
LT31
LT41
LT51
LT61
2
0
2018
2019
2020
2021
2022
Anni


Osservazioni
Previsioni
GRUPPO 6
LT1
LT2
LT3
LT4
LT5
LT6
1
0
-1
Jul 2018
Jan 2019
Jul 2019
Jan 2020
Jul 2020
Jan 2021
Jul 2021
Jan 2022
Jul 2022
Anni