

Treball Final de Màster

Màster en modelització per la Ciència i Enginyeria

Predictió de temperatura a curt termini utilitzant *machine learning*

Pau Rodrigo Parés



**Servei Meteorològic
de Catalunya**

Tutors

Sílvia Cuadrado (UAB)
Josep Ramon Miró (SMC)

Convocatòria: **Juny 2025**
Universitat Autònoma de Barcelona

Resum

Short-term temperature forecasting is crucial in meteorology and for services such as the Meteorological Service of Catalonia. Currently, these forecasts rely on numerical models that, while accurate, require high computational costs. This study analyzes the use of Recurrent Neural Networks (RNN), specifically Long Short-Term Memory (LSTM) architectures, to determine whether they can provide accurate hourly temperature forecasts with lower computational requirements.

To contextualize their performance, LSTM models are compared with classical ARIMA and SARIMA models, which have been widely used in time series modelling but may not be optimal with capturing complex short-term nonlinear patterns. In addition, a categorical Markov Chain model is used for qualitative comparison in a precipitation forecasting scenario. The analysis are conducted using real data from Meteocat's automatic weather station, aiming to assess both model accuracy and computational feasibility.

The results show that statistical models provide competitive performance in very short horizons, while LSTM models offer potential advantages in more challenging setups, particularly when using recursive prediction strategies. This work lays the groundwork for future research directions, including the integration of additional input features and the exploration of alternative neural architectures to build more robust forecasting systems.

La predicción de temperatura a curt termini és crucial en meteorologia i per a serveis com el Servei Meteorològic de Catalunya. Actualment, aquestes prediccions es basen en models numèrics que, tot i ser precisos, requereixen costos computacionals elevats. Aquest estudi analitza l'ús de Xarxes Neuronals Recurrents (RNN), concretament arquitectures de Memòria a Curt Termini Llarg (LSTM), per determinar si poden proporcionar prediccions de temperatura horàries precises amb requisits computacionals més baixos.

Per contextualitzar el seu rendiment, els models LSTM es comparen amb els models clàssics ARIMA i SARIMA, que s'han utilitzat àmpliament en la modelització de sèries temporals però que poden no ser óptims per capturar patrons no lineals complexos a curt termini. A més, s'utilitza un model de Cadena de Markov categòrica per a la comparació qualitativa en un escenari de predicción de precipitacions. L'anàlisi es duu a terme utilitzant dades reals de l'estació meteorològica automàtica de Meteocat, amb l'objectiu d'avaluar tant la precisió del model com la viabilitat computacional.

Els resultats mostren que els models estadístics proporcionen un rendiment competitiu en horitzons molt curts, mentre que els models LSTM ofereixen avantatges potencials en configuracions més desafiadores, especialment quan s'utilitzen estratègies de predicción recursiva. Aquest treball estableix les bases per a futures línies de recerca, incloent-hi la integració de funcions d'entrada addicionals i l'exploració d'arquitectures neuronals alternatives per construir sistemes de predicción més robustos.

Acrònims

ARIMA Model autoregressiu integrat de mitjana mòbil.

LSTM Long-Short Term Memory.

MAE Mean Absolute Error.

ML Aprenentatge automàtic (Machine Learning).

MSE Mean Squared Error.

RMSE Root Mean Squared Error.

RNN Recurrent Neural Network.

SARIMA Seasonal ARIMA.

SMC Servei Meteorològic de Catalunya.

UAB Universitat Autònoma de Barcelona.

XEMA Xarxa d'Estacions Meteorològiques Automàtiques.

Glossari de símbols i termes

- C_t Estat intern (memòria) actualitzat en el pas t .
- C_{t-1} Estat intern (memòria) de la cèl·lula en el pas anterior.
- P_{ij} Probabilitat de transició de l'estat i a j en un model de Markov.
- U_* Matriu de pesos aplicada a l'estat ocult h_{t-1} per a cada porta.
- W_* Matriu de pesos aplicada a l'entrada x_t per a cada porta ($*$ = f, i, o, c).
- $\sigma(\cdot)$ Funció sigmoide, amb sortida a l'interval $[0, 1]$.
- $\tanh(\cdot)$ Funció tangenta hiperbòlica, amb sortida a l'interval $[-1, 1]$.
- \tilde{C}_t Cèl·lula candidata: nova informació proposada per ser incorporada a la memòria.
- b_* Termes de biaix de cada porta.
- f_t Porta d'oblit: controla quina part de la memòria anterior es descarta.
- h_t Sortida de la cèl·lula en el pas actual.
- h_{t-1} Sortida (estat ocult) de la cèl·lula en el pas anterior.
- i_t Porta d'entrada: regula quina nova informació s'incorpora.
- o_t Porta de sortida: determina quina part de la memòria es propaga com a sortida.
- x_t Entrada de la cèl·lula en el pas temporal t .
- B Operador de retard (*backshift*): $By_t = y_{t-1}$. Permet expressar diferències i components estacionals..
- D Nombre de diferenciacions estacionals aplicades a la sèrie..
- $\Phi_P(B^s)$ Polinomi del component AR estacional, d'ordre P i període s ..
- $\Theta_Q(B^s)$ Polinomi del component MA estacional, d'ordre Q i període s ..
- ϕ_i Coeficients del component autoregressiu (AR), que ponderen la influència dels valors passats $y_{t-i}..$
- $\phi_p(B)$ Polinomi que representa el component AR no estacional d'ordre p , en funció de l'operador B ..
- θ_j Coeficients del component de mitjana mòbil (MA), que ponderen els errors passats $\varepsilon_{t-j}..$
- $\theta_q(B)$ Polinomi del component MA no estacional d'ordre q , en funció de B ..
- ε_t Soroll blanc: error aleatori amb mitjana zero i no correlacionat..
- ε_{t-j} Error de predicció comès en l'instant $t - j$, és a dir, la diferència entre el valor observat i el predit..
- c Constant o terme independent del model..
- d Nombre de diferenciacions no estacionals aplicades a la sèrie..
- s Període d'estacionalitat (p. ex., $s = 24$ per cicles diürns en dades horàries)..
- y_t Valor observat de la sèrie temporal en el temps t ..

Índex general

Resum	i
Índex general	v
1 Introducció	1
2 Obtenció i tractament de dades	2
2.1 Tractament previ de les dades	2
3 Modelització amb Cadenes de Markov	3
3.1 Aplicació de les cadenes de Markov en meteorologia	3
3.2 Plantejament del problema amb sèries de Markov	4
3.3 Predicció i resultats del model de Markov	5
4 Modelització amb models ARIMA	8
4.1 Aplicació dels models ARIMA	8
4.2 Fonaments dels models ARIMA	8
4.2.1 Identificació dels paràmetres del model	9
4.2.2 Extensió estacional: models SARIMA	9
4.3 Plantejament de models	10
4.4 Resultats	11
5 Modelització amb models LSTM	13
5.1 Aplicació dels models Long Short-Term Memory (LSTM)	13
5.2 Fonament de les RNN i LSTM	13
5.3 Estructuració del model	15
5.3.1 Descripció general del plantejament	15
5.3.2 Preparació de les dades	16
5.3.3 Arquitectura de la xarxa	17
5.3.4 Resum d'hiperparàmetres del model	17
5.3.5 Estratègies de predicció	18
5.4 Plantejament de models	19
5.5 Resultats	20
6 Avaluació comparativa dels models	23
7 Resum, conclusions i treball futur	24
Referències	26

1. Introducció

La predicción de la temperatura a curt termini és una tasca clau dins del camp de la meteorologia operativa, amb aplicacions que van des de la gestió del risc meteorològic fins a l'optimització de serveis logístics i energètics. Actualment, aquestes prediccions es basen principalment en models numèrics d'alta resolució com el WRF (*Weather Research and Forecasting model*), que ofereixen bons resultats però tenen un cost computacional elevat.

Aquesta necessitat obre la porta a explorar mètodes alternatius, menys costosos computacionalment, que puguin oferir prediccions raonablement precises amb menys recursos i temps. En aquest context, els models basats en tècniques d'aprenentatge automàtic (*machine learning*), com les arquitectures *Long Short-Term Memory* (LSTM), basades en xarxes neuronals recurrents (RNN), s'han popularitzat en els darrers anys per la seva capacitat d'adaptar-se a patrons complexos i no lineals de les sèries temporals, com els de l'evolució horària de la temperatura.

El projecte s'emmarca dins la línia de Recerca Aplicada i Modelització (RAM) del Servei Meteorològic de Catalunya (SMC), i té com a objectiu analitzar la viabilitat d'utilitzar models de *machine learning* per a la predicción horària de la temperatura a molt curt termini. L'estudi es presenta com un primer pas cap al desenvolupament de mètodes propis, eficients i adaptats a les necessitats específiques del SMC, orientats a la previsió a curt termini de variables meteorològiques com la temperatura o el vent.

L'estudi planteja una comparació entre tres enfocaments diferents: les cadenes de Markov (com a model categòric de base probabilística), els models estadístics ARIMA i SARIMA (com a referència clàssica per a sèries temporals) i les xarxes LSTM (com a model d'aprenentatge profund). Aquesta comparació es fa sobre dades reals horàries de temperatura de l'estació de la Bonaigua, en el període 1998–2024, amb especial atenció al comportament a curt termini i a la capacitat de cada model per superar el model de persistència, sovint emprat com a línia base simple però sorprendentment efectiva.

Tots els models s'han implementat amb llenguatge Python, utilitzant biblioteques especialitzades com `statsmodels` per als models ARIMA i `TensorFlow` per a les xarxes neuronals. A més, s'ha desenvolupat un flux de treball *pipeline* modular propi, que permet repetir els experiments de manera eficient i traçable. El codi complet d'aquest projecte està disponible públicament al repositori: github.com/Paurro/TFM_SCM_Temperature_Forecasting_ML.

Així doncs s'han plantejat com a objectius:

- Establir una línia base sòlida per a la predicción de temperatura a curt termini, a partir de dades horàries d'una estació de muntanya.
- Avaluar el comportament de diferents metodologies (Markov, ARIMA, LSTM) en termes de precisió i eficiència computacional.
- Analitzar la influència de diversos paràmetres estructurals i d'entrenament en el rendiment dels models LSTM.
- Explorar la viabilitat d'integrar models de *machine learning* dins els processos operatius del SMC.

Els capítols següents presenten el detall del tractament de dades, la formulació teòrica i pràctica dels diferents models, així com els resultats obtinguts i les conclusions extretes.

2. Obtenció i tractament de dades

El SMC disposa d'una Xarxa d'Estacions Meteorològiques Automàtiques (XEMA), que compta amb prop de 190 estacions distribuïdes per tot el territori català. Aquesta xarxa d'observació és la que recull totes les dades del territori que s'usen per calibrar i corregir les sortides dels models numèrics amb tècniques estadístiques.

Per a l'entrenament dels models predictius d'aquest estudi s'han utilitzat dades de l'estació de la Bonaigua, situada a 2266 metres d'altitud al municipi d'Alt Àneu, al Pallars Sobirà. S'ha escollit aquesta estació, ubicada en un dels punts més elevats dels Pirineus, per tal de poder observar amb claredat les oscil·lacions estacionals entre tardor-hivern i primavera-estiu, així com els cicles dia-nit. També s'han escollit les dades d'aquesta estació d'alta muntanya per tenir rangs de temperatura que permetin categoritzar les dades en tipus de precipitació (Neu, Aiguaneu, Pluja), la qual cosa s'aplicarà en la Modelització amb Cadenes de Markov.

Les dades utilitzades corresponen a la temperatura horària enregistrada des de l'1 de gener de 1998 fins al 31 de desembre de 2024. Aquesta extensió temporal és suficientment ampli per permetre l'estudi de tendències i l'entrenament de diversos models de predicción a curt termini.

2.1 Tractament previ de les dades

Per tal de garantir la qualitat de les dades, s'han eliminat valors nuls i inconsistents. També s'han filtrat les observacions per conservar només aquelles registrades exactament a l'inici de cada hora (és a dir, minuts iguals a zero). Aquesta transformació assegura una estructura temporal regular amb un valor per hora.

Altres transformacions específiques, com la discretització de la temperatura o l'agregació horària, es descriuràn a les seccions corresponents a cada model.

A més, s'ha estudiat la continuïtat de les dades a fi de detectar possibles discontinuitats en les dades, per tal d'acotar correctament els períodes de dades que s'utilitzaran. La Figura 2.1 mostra l'evolució de la temperatura des del 1998 fins al 2024, i permet observar trams amb dades absents, sobretot entre finals dels anys noranta i inicis dels 2000.

A partir de l'any 2007 la sèrie ja mostra estabilitat i continuïtat, amb només alguns buits puntuals de poques hores al 2008 i al 2019. Tenint això en compte, s'han identificat dos períodes especialment útils per entrenar models amb garantia de continuïtat temporal: del 2009 al 2018, i del 2020 al 2024. Aquests trams s'han considerat com a subconjunts de dades consistentes, lliures de grans discontinuitats, aptes per entrenar i desenvolupar models.

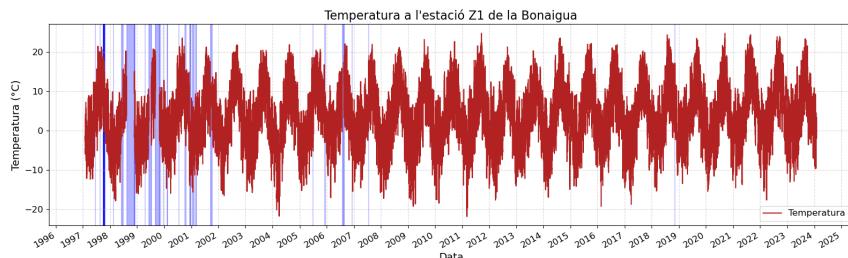


Figura 2.1: Sèrie de temperatura horària a l'Estació de la Bonaigua (1998–2024), amb buits de dades destacats.

3. Modelització amb Cadenes de Markov

3.1 Aplicació de les cadenes de Markov en meteorologia

Les cadenes de Markov són eines estadístiques basades en processos estocàstics que permeten modelitzar l'evolució dels estats d'un sistema en què el següent, depèn de l'actual. Aquest mètode ha demostrat ser útil per representar una gran varietat de sistemes reals amb estats discrets, com ara sistemes de cues, sistemes d'inventari o mercats de productes (Liu, 2010), així com per modelitzar variables meteorològiques categòriques com l'ocurrència o absència de precipitació de precipitació, intervals de temperatura o l'estat del cel (Gabriel & Neumann, 1962).

El funcionament de les cadenes de Markov es basa en la idea que la transició entre estats segueix una probabilitat condicionada. És a dir, la probabilitat que el sistema estigui en un estat j en el temps $t + 1$ depèn només de l'estat i en el temps t , i es pot escriure com:

$$\mathbb{P}(X_{t+1} = j \mid X_t = i) \quad (3.1)$$

Sent X una variable aleatòria que representa l'estat del sistema.

Aquesta probabilitat es defineix segons l'equació eq. (3.2), on donats dos possibles esdeveniments A, B , la probabilitat condicionada que es doni A , sabent que s'ha donat B es defineix com:

$$\mathbb{P}(A \mid B) = \frac{\mathbb{P}(A \cap B)}{\mathbb{P}(B)} \quad (3.2)$$

Quan es disposa d'una sèrie temporal discreta, et pot calcular, empíricament, la probabilitat de cada transició entre estats comptant el nombre de vegades que es produeix una seqüència concreta. Això permet estimar les probabilitats de transició com:

$$\mathbb{P}(X_{t+1} = j \mid X_t = i) = \frac{\text{nombre de transicions de } i \rightarrow j}{\text{nombre total de vegades que es troba l'estat } i} \quad (3.3)$$

Aquest conjunt de probabilitats condicionades es representen en forma de matriu de transició on cada fila correspon a l'estat actual i cada columna al següent. Sent i, j, k tres possibles estats del sistema, les diverses probabilitats condicionades es veurien com la següent matriu:

$$P = \begin{bmatrix} P_{ii} & P_{ij} & P_{ik} \\ P_{ji} & P_{jj} & P_{jk} \\ P_{ki} & P_{kj} & P_{kk} \end{bmatrix} \quad (3.4)$$

on $P_{xy} = \mathbb{P}(X_{t+1} = y \mid X_t = x)$, sent x, y qualsevol dels estats possibles (i, j, k), representa la probabilitat condicionada que el sistema es trobi en l'estat y en el temps $t + 1$, donat que en el temps t estava en l'estat x .

Tot i que existeixen cadenes de Markov d'ordre superior, en què l'estat futur depèn de més d'un estat anterior (per exemple, $\mathbb{P}(X_{t+1} \mid X_t, X_{t-1})$ en una cadena de segon ordre), aquestes impliquen una cost computacional superior i, per aquesta raó, sovint es prefereixen les cadenes de primer ordre, especialment quan l'objectiu és capturar la dependència immediata (Gutiérrez et al., 2004).

Aquest mètode, ha estat utilitzat en meteorologia per simular sèries binàries de precipitació (pluja / no pluja), com explica Gutiérrez et al. (2004). Aquest enfocament permet predir seqüències com, la

durada mitjana de períodes secs o humits, el primer dia de pluja després d'una ratxa seca, entre d'altres. A partir de les probabilitats de transició, es poden generar sèries temporals dels estats.

3.2 Plantejament del problema amb sèries de Markov

Una primera aproximació a l'estudi de la predicció de sèries temporals amb cadenes de Markov s'ha de fer per força amb variables categòriques, ja que aquesta metodologia es basa en un càlcul de probabilitats discretes. Per això, es proposa fer el càlcul de la probabilitat del tipus de precipitació basant-se en la discretització de la temperatura en diferents rangs, tal com es fa a, Lefebvre (2019), en el qual aquesta estratègia s'utilitza per modelitzar i predir variacions mensuals de temperatura a Itàlia.

Prenent aquest enfocament com a referència, s'han definit tres intervals de temperatura, cadascun associat a un tipus de precipitació:

$$\text{Tipus de precipitació} = \begin{cases} \text{Neu (N)} & \text{si } T < 1^\circ\text{C} \\ \text{Aiguaneu (A)} & \text{si } 1^\circ\text{C} \leq T \leq 2^\circ\text{C} \\ \text{Pluja (P)} & \text{si } T > 2^\circ\text{C} \end{cases} \quad (3.5)$$

En aquest cas, les dades dades són horàries i provinents de l'estació meteorològica de la Bonaigua, tal com s'indica al capítol 2. Tanmateix s'han considerat exclusivament els mesos corresponents a l'hivern climatològic (novembre, desembre, gener i febrer), amb l'objectiu de garantir la presència d'una distribució representativa entre les tres categories. Aquesta elecció evita augmentar el desbalanceig de la categoria de pluja en els mesos més càlids, que dificultaria l'aplicació de la metodologia.

Les probabilitats de transició s'han calculat a partir de les dades dels hiverns compresos entre les temporades 2000–2001 i 2022–2023. L'hivern 2023–2024 s'ha establert com a conjunt de test per tal de validar la capacitat predictiva del model. Les figures Fig. 3.1 i Fig. 3.2 mostren, respectivament, l'evolució de la temperatura i la classificació en categories durant aquest període.

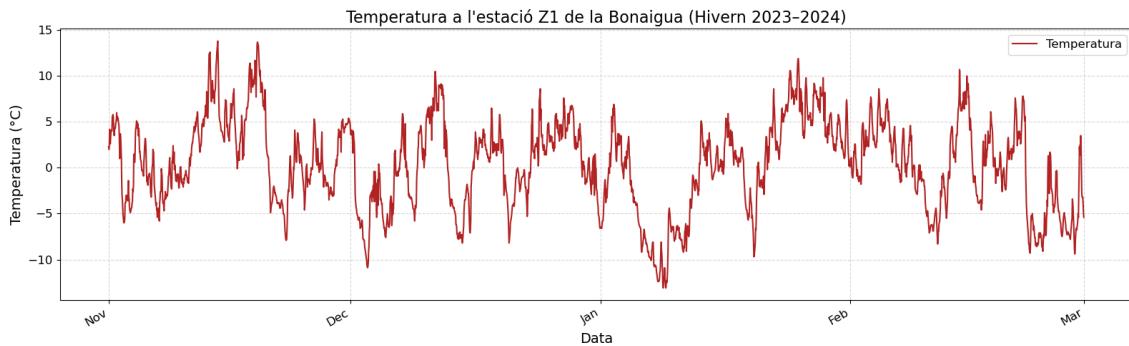


Figura 3.1: Evolució de la temperatura a l'estació de la Bonaigua durant l'hivern 2023–2024.

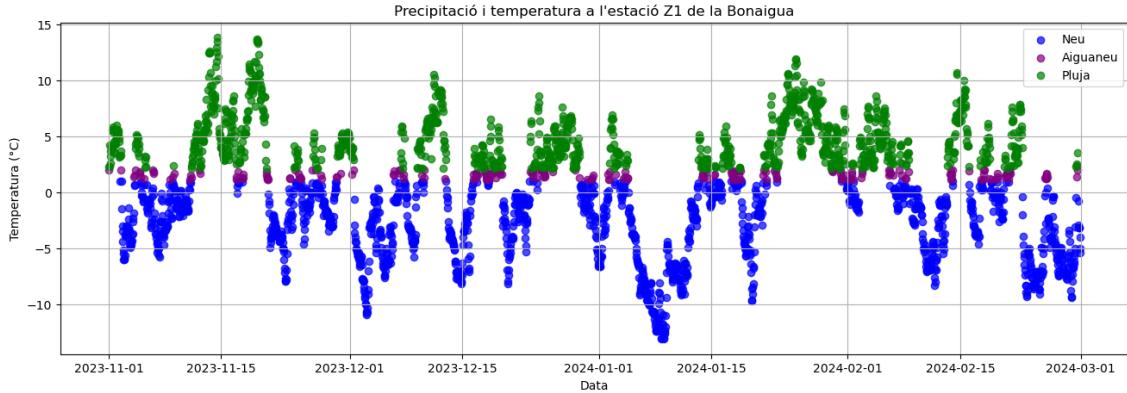


Figura 3.2: Discretització de la temperatura en categories de precipitació a l'estació de la Bonaigua durant l'hivern 2023–2024.

3.3 Predicció i resultats del model de Markov

Amb les dades històriques, s'han calculat les probabilitats de transició entre estats i sintetitzat en una matriu (Taula 3.1), que són la base del model.

Estat anterior → Actual	Neu (N)	Aiguaneu (A)	Pluja (P)
Neu (N)	0.9717	0.0218	0.0065
Aiguaneu (A)	0.2809	0.4872	0.2320
Pluja (P)	0.0175	0.0745	0.9080

Taula 3.1: Matriu de transició estimada a partir del conjunt d'entrenament. Les files representen l'estat de precipitació a l'hora anterior, i les columnes l'estat a l'hora actual.

Abans de fer la simulació, les probabilitats de la matriu ja permeten anticipar alguns comportaments. L'estat de neu (N) i el de pluja (P) tenen una alta persistència, amb valors superiors al 90%, cosa que indica que, un cop s'hi entra, és molt probable que es mantinguin. En canvi, l'aiguaneu (A) és menys estable, i actua com a estat de transició: té gairebé la meitat de probabilitat de quedar-se, un 40%, i una probabilitat de mes del 20% de passar tant a neu com a pluja. Les transicions directes entre neu i pluja són molt baixes, fet que té sentit si es té en compte que representen situacions extremes separades per un marge tèrmic, i que la temperatura es una variable contínua. Aquests salts implicarien un canvi brusc de temperatura en menys d'una hora, cosa que és molt poc probable.

Seguidament, s'ha dut a terme una simulació estocàstica de l'evolució dels estats a partir de la matriu de transició. A cada pas de temps, donat l'estat actual en l'instant t , es genera l'estat següent $t + 1$ mitjançant un procés de mostreig aleatori, seguint la distribució de probabilitats associada a aquell estat en la matriu de transició.

Cal destacar que en cada pas s'utilitza sempre l'estat real observat en t per determinar la distribució de probabilitats del següent pas, i no l'estat simulat anterior, ja que l'objectiu és generar una seqüència de prediccions a partir de la sèrie real. La fig. 3.3 mostra la seqüència de prediccions generada, que s'ha comparat amb la sèrie real per valorar la capacitat del model de capturar els patrons temporals.

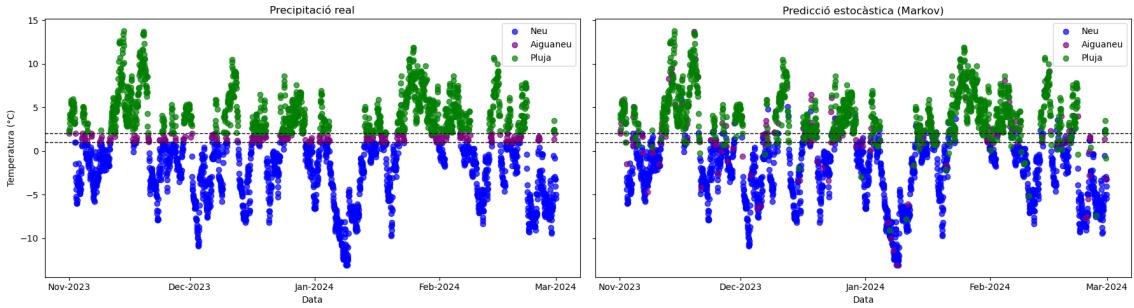


Figura 3.3: Comparació entre la sèrie real (esquerra) i la predició generada pel model de Markov (dreta).

La matriu de confusió de la Fig. 3.4 resumeix els encerts i errors de classificació respecte als valors reals observats. El model presenta un alt nivell d'encert en l'estat de neu (1.443 encerts) i pluja (979 encerts), mentre que l'estat intermedi d'aiguaneu presenta una menor taxa de classificació correcta (60 encerts sobre 219 casos).

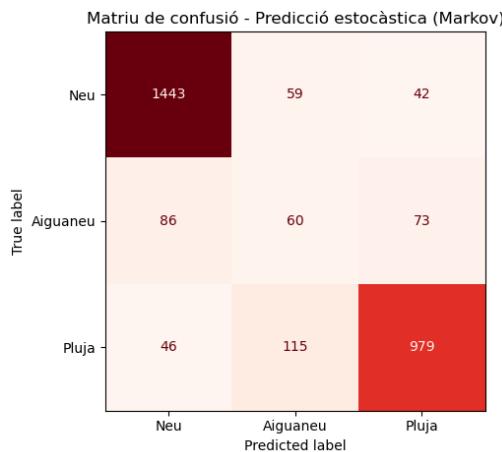


Figura 3.4: Matriu de confusió del model de Markov.

Les Fig. 3.5 i Fig. 3.6 permeten analitzar la distribució de freqüències i el valor de *recall* per classe. Pel que fa a les freqüències, s'observa que el model tendeix a mantenir la proporció entre real i predit, però respecte al *recall* per classes, els resultats mostren valors elevats per a neu (prop del 93%) i pluja (al voltant del 86%), mentre que per a aiguaneu el valor cau per sota del 30%.

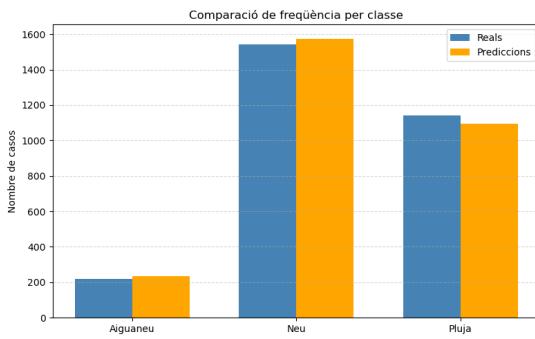


Figura 3.5: Comparació de freqüències entre classes reals i predites.

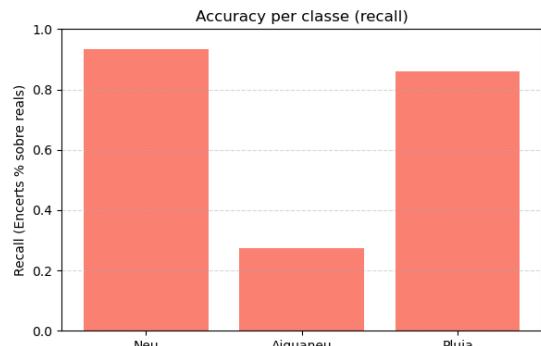


Figura 3.6: Valor de recall per classe.

En conjunt, els resultats mostren que el model de Markov és capaç de capturar amb eficàcia els patrons generals i els estats dominants del sistema (Neu i Pluja). Malgrat això, les transicions entre

estats similars, com ara entre neu i aiguaneu o entre aiguaneu i pluja, es prediuen amb dificultat i menys certesa. Aquestes limitacions poden atribuir-se a la proximitat dels llindars de classificació, que és bàsicament 1 grau, i al fet que el model no té memòria mes enllà d'un estat ni incorpora altres variables meteorològiques que podrien ajudar.

4. Modelització amb models ARIMA

4.1 Aplicació dels models ARIMA

Els models ARIMA (*Autoregressive Integrated Moving Average*) són una de les eines estadístiques més utilitzades per a l'anàlisi i la predicció de sèries temporals univariants. Es basen en la idea que el valor actual d'una sèrie pot expressar-se com una combinació lineal dels seus valors passats i dels errors de predicció anteriors, un cop eliminades les possibles tendències o estacionalitats de la sèrie. Aquesta combinació permet capturar l'estrucció temporal i generar prediccions a curt termini.

Els models ARIMA han estat aplicats amb èxit en nombrosos camps com l'economia, la medicina, la demografia i la meteorologia (Elseidi, 2024; Murat et al., 2018). En el cas concret de la temperatura, diversos estudis han mostrat que poden oferir una línia base sòlida per a la predicció a curt termini, especialment en contextos amb estacionalitat clara o tendències suaus (Koçak, 2023; Tugal & Sevgin, 2023).

Tot i això, els models ARIMA presenten certes limitacions que cal tenir en compte. D'una banda, tenen diversos punts forts: són estadísticament robustos, fàcilment interpretables i computacionalment eficients, cosa que els fa molt adequats per a prediccions ràpides o contextos on es disposa de recursos limitats. A més, són especialment útils com a línia base per comparar amb models més sofisticats, ja que proporcionen un referent simple però sòlid.

Tanmateix, també tenen limitacions conegeudes. Per una banda, assumeixen que les relacions entre els valors passats i futurs de la sèrie són lineals, i que aquesta és estacionària, cosa que pot limitar-ne l'aplicabilitat en casos amb dinàmiques no estacionàries o fortament no lineals. A més, els models ARIMA no permeten incorporar fàcilment informació exògena, com altres variables meteorològiques, a menys que s'utilitzin variants com els models ARIMAX. Finalment, també poden mostrar dificultats en presència de sorolls o anomalies, i tendeixen a perdre precisió en situacions amb canvis sobtats o no previsibles.

Tot i ser una eina senzilla, els models ARIMA i SARIMA ofereixen una base sòlida per a la predicció de sèries temporals, especialment quan es disposa de dades amb estacionalitat i tendències clares. Per aquest motiu, sovint s'utilitzen com a punt de partida en estudis on també es proven models més avançats, com les xarxes neuronals LSTM, servint com a referència per valorar la millora que aporten aquests enfocaments més complexos (De Saa & Ranathunga, 2020; Tugal & Sevgin, 2023).

4.2 Fonaments dels models ARIMA

Els models ARIMA van ser proposats originalment per Box et al. (2015) i combinen tres components diferenciats: autoregressiu (AR), integrat (I) i de mitjana mòbil (MA), que permeten capturar la dependència temporal, eliminar tendències i modelitzar la influència d'errors passats.

Un model ARIMA(p, d, q) combina:

- Un component autoregressiu d'ordre p (AR), que considera els valors passats de la pròpia sèrie.
- Una diferenciació d'ordre d (I), que transforma la sèrie per fer-la estacionària.
- Un component de mitjana mòbil d'ordre q (MA), que incorpora els errors de predicció anteriors.

La forma general del model és:

$$y_t = c + \sum_{i=1}^p \phi_i y_{t-i} + \sum_{j=1}^q \theta_j \varepsilon_{t-j} + \varepsilon_t \quad (4.1)$$

on y_t representa el valor observat de la sèrie temporal en el temps t , c és una constant (terme independent), i ϕ_i i θ_j són, respectivament, els coeficients dels components autoregressiu (AR) i de mitjana mòbil (MA). Els valors y_{t-i} corresponen als valors passats de la sèrie que intervenen en la part autoregressiva, mentre que els termes ε_{t-j} són els errors de predicció comesos en instants anteriors, és a dir, la diferència entre el valor observat i el valor predit pel model en aquell moment. Finalment, ε_t representa el soroll blanc, una seqüència d'errors aleatoris, independents i amb mitjana zero que modelitza la part imprevisible del sistema.

4.2.1 Identificació dels paràmetres del model

Per ajustar un model ARIMA(p, d, q) cal determinar tres paràmetres: p (ordre autoregressiu), d (nombre de diferenciacions) i q (ordre de la mitjana mòbil). Aquest procés de selecció s'inicia habitualment amb la detecció de no estacionaritat, seguida per l'anàlisi de la dependència temporal mitjançant autocorrelacions i, finalment, la comparació entre models amb criteris estadístics.

Estacionaritat i diferenciació. Els models ARIMA assumeixen que la sèrie temporal és estacionària, és a dir, que les seves propietats estadístiques, com la mitjana, la variància o l'autocorrelació, es mantenen estables en el temps. Aquesta condició no es compleix sovint en dades reals, especialment en sèries meteorològiques com la temperatura, que presenten tendències i estacionalitats (De Saa & Ranathunga, 2020).

Per tal de fer la sèrie estacionària, s'aplica una operació de diferenciació:

$$y'_t = y_t - y_{t-1} \quad (4.2)$$

Aquest procés es pot repetir (diferenciació d'ordre d) fins que la sèrie resulti aparentment estacionària. El valor de d és, doncs, un dels paràmetres fonamentals del model. L'estacionaritat es pot verificar amb tests com l'ADF (Augmented Dickey-Fuller), que permeten determinar la necessitat de diferenciació (Koçak, 2023).

Determinació de p i q . Els paràmetres p i q , que defineixen respectivament l'ordre dels components autoregressiu AR (nombre de valors anteriors de la sèrie que s'utilitzen per fer la predicció) i de mitjana mòbil MA (nombre d'errors anteriors que es tenen en compte per corregir la predicció), es poden tempejar mitjançant mètodes gràfics i estadístics.

Gràficament, es pot representar d'autocorrelació (ACF) i autocorrelació parcial (PACF), el que permet identificar l'estructura temporal de la sèrie: un tall brusc a la PACF pot indicar la presència d'un component AR, mentre que un tall a l'ACF pot suggerir un component MA.

Estadísticament es pot recórrer els càlculs de l'AIC (Akaike Information Criterion) i el BIC (Bayesian Information Criterion) són útils per comparar diferents configuracions de models i escollir-ne la més adequada. (Murat et al., 2018)

Tot i això, l'elecció final dels paràmetres sovint requereix una validació empírica basada en l'ajust del model i les seves mètriques d'error, respecte l'observació.

4.2.2 Extensió estacional: models SARIMA

Quan la sèrie presenta una estacionalitat clara (com cicles diürns o estacionals), el model ARIMA pot resultar insuficient. En aquests casos, s'utilitza una extensió anomenada SARIMA (*Seasonal ARIMA*), queafegeix components estacionals:

$$\text{SARIMA}(p, d, q)(P, D, Q)_s \quad (4.3)$$

on (P, D, Q) són els ordres dels components estacionals, i s representa el període de la repetició (per exemple, $s = 24$ per cicles diürns en dades horàries o $s = 365$ per dades diàries anuals).

Aquest model combina les diferències estacionals i no estacionals per capturar tant les tendències generals com les oscil·lacions periòdiques. Estudis recents mostren que el model SARIMA pot ser especialment útil en la predicción de temperatura quan la sèrie presenta una doble estacionalitat marcada (Koçak, 2023; Murat et al., 2018).

La formulació matemàtica completa és la següent:

$$\Phi_P(B^s) \phi_p(B) (1 - B)^d (1 - B^s)^D y_t = \Theta_Q(B^s) \theta_q(B) \varepsilon_t \quad (4.4)$$

El model SARIMA es formula com una combinació dels components estacionals i no estacionals. L'element B representa l'operador de retard , de manera que $By_t = y_{t-1}$. Els polinomis $\phi_p(B)$ i $\theta_q(B)$ corresponen als components autoregressiu (AR) i de mitjana mòbil (MA) no estacionals, d'ordre p i q , respectivament. Els termes estacionals s'expressen mitjançant els polinomis $\Phi_P(B^s)$ i $\Theta_Q(B^s)$, que representen els components autoregressiu i de mitjana mòbil estacionals, amb període s i ordres P i Q .

Les transformacions $(1 - B)^d$ i $(1 - B^s)^D$ introdueixen la diferenciació no estacional i estacional, amb l'objectiu de garantir l'estacionariedad de la sèrie a curt i llarg termini. Finalment, ε_t representa el soroll blanc, és a dir, una seqüència d'errors aleatoris amb mitjana zero i no auto-correlacionats.

Això permet capturar simultàniament les estructures a curt i llarg termini, així com la periodicitat intrínseca de la sèrie. En el cas de dades meteorològiques com la temperatura, resulta útil per modelitzar tant l'evolució diària (per exemple, cicles nit-dia amb $s = 24$) com les variacions estacionals anuals ($s = 365$ en dades diàries).

4.3 Plantejament de models

Tal com s'ha desenvolupat anteriorment, els paràmetres controlables dels models ARIMA són els que es poden observar a la taula 4.1

Paràmetre	Valors provats	Tipus
p, q (ordre AR i MA)	[2, 4, 6, 10]	A optimitzar
d (diferenciació)	1	Fixat
P, Q (ordre estacional)	[0, 1, 2]	A optimitzar (SARIMA)
D (diferenciació estacional)	0	Fixat
s (periode estacional)	24	Fixat
dies d'entrenament	[30, 60, 90]	A optimitzar

Taula 4.1: Resum de paràmetres considerats en els experiments amb models ARIMA i SARIMA.

Els valors escollits per als paràmetres responen a una combinació de criteris pràctics, intuïcions basades en proves preliminars i referències de la literatura (Elseidi, 2024; Koçak, 2023; Murat et al., 2018). En el cas de p i q , s'han considerat valors d'entre 2 i 10 per explorar models des de molt senzills fins a d'altres amb més capacitat de memòria. Tot i que en alguns casos podrien utilitzar-se ordres més alts, això incrementa molt la complexitat del model i el risc de sobre-ajustament, sobretot en finestres d'entrenament curtes. La diferenciació $d = 1$ s'ha fixat, ja que és un valor habitual per eliminar tendències suaus, i en les sèries de temperatura acostuma a ser suficient per assolir l'estacionariedad.

Pel que fa a la component estacional, s'ha optat per mantenir el període fix a $s = 24$ per capturar l'estacionalitat diària pròpia de dades horàries. Es prova l'efecte d'afegir components estacionals

(P, Q) fins a valor 2, però es manté $D = 0$ per simplicitat, ja que la diferenciació no estacional sol ser suficient per eliminar la major part de les tendències i oscil·lacions lentes. Les combinacions estacionals permeten veure si afegir estructura cíclica millora realment la capacitat predictiva del model o si n'hi ha prou amb una formulació ARIMA més senzilla.

Finalment, també s'ha volgut analitzar l'impacte de la quantitat d'historial disponible a l'hora d'entrenar els models, tot variant la mida de la finestra entre 30 i 90 dies. Aquesta variació és especialment rellevant en el context d'aquest projecte, on l'estrategia de predicción serà un esquema de *rolling forecast*, en què el model es re-entrena a un nombre de passos determinat (en aquest cas cada 5 passos), utilitzant la finestra d'entrenament més recent. Aquest enfocament permet adaptar-se millor a canvis en la dinàmica de la sèrie, però també és més exigent en termes computacionals i de robustesa del model.

ID	p	d	q	P	D	Q	s	dies_ent	Tipus	Notes
A0	2	1	2	0	0	0	24	60	ARIMA	Bàsic sense estacionalitat
A1	4	1	4	0	0	0	24	60	ARIMA	Més memòria autoregressiva i MA
A2	10	1	10	0	0	0	24	60	ARIMA	Model complex per contrast
A3	2	1	2	0	0	0	24	30	ARIMA	Bàsic amb poques dades
A4	2	1	2	0	0	0	24	90	ARIMA	Bàsic amb moltes dades
B0	2	1	2	1	0	1	24	60	SARIMA	Amb estacionalitat suau
B1	2	1	2	2	0	2	24	60	SARIMA	Estacionalitat més marcada
C0	2	1	2	1	0	1	24	30	SARIMA	Finestra curta, reentrenament constant
C1	2	1	2	1	0	1	24	60	SARIMA	Configuració intermèdia
C2	2	1	2	1	0	1	24	90	SARIMA	Finestra llarga, més estabilitat

Taula 4.2: Configuracions experimentals dels models ARIMA i SARIMA amb tots els paràmetres considerats. Es mostren tant les estratègies de predicción directa com les de rolling forecast.

4.4 Resultats

La taula 4.3 recull les mètriques d'error (RMSE, MSE i MAE) que permeten avaluar els models amb les configuracions provades. En general, els models SARIMA mostren un rendiment superior als ARIMA purs, amb una reducció clara de l'error en tots els indicadors. Aquesta millora es manté consistent fins i tot en escenaris amb diferents longituds de finestra d'entrenament, cosa que reforça la robustesa dels models amb component estacional.

ID	RMSE	MSE	MAE
A0	1.5372	2.3631	1.0516
A1	1.4075	1.9812	0.9793
A2	1.4203	2.0174	0.9768
A3	1.5351	2.3565	1.0499
A4	1.5164	2.2994	1.0405
B0	1.3145	1.7279	0.9371
B1	1.3146	1.7283	0.9335
C0	1.3243	1.7485	0.9434
C1	1.3145	1.7279	0.9371
C2	1.3155	1.7306	0.9300

Taula 4.3: Resultats de les mètriques d'error obtingudes per a cada configuració experimental.

Es pot observar en els experiments de la sèrie A com és la influència dels paràmetres i la finestra

d'entrenament. Augmentar l'ordre (p i q) o l'historial d'entrenament, comporta algunes millors, però aquestes no són gaire significatives i sovint no compensen l'increment de cost computacional.

Per altra banda, els models SARIMA de les sèries B i C ofereixen resultats més sòlids. Fins i tot amb configuracions estacionals senzilles, com $(1, 0, 1)$, el model és capaç d'adaptar-se de manera efectiva al cicle diari de la temperatura, millorant l'ajust global. En aquests casos, la mida de la finestra d'entrenament no té un impacte substancial en el rendiment, fet que permet una certa flexibilitat operativa. Això es reflecteix en el model C1 (SARIMA $(2, 1, 2)(1, 0, 1)$ amb període 24) (fig. 4.1) que amb només 60 dies d'entrenament assoleix les millors mètriques, sent flexible i senzill.

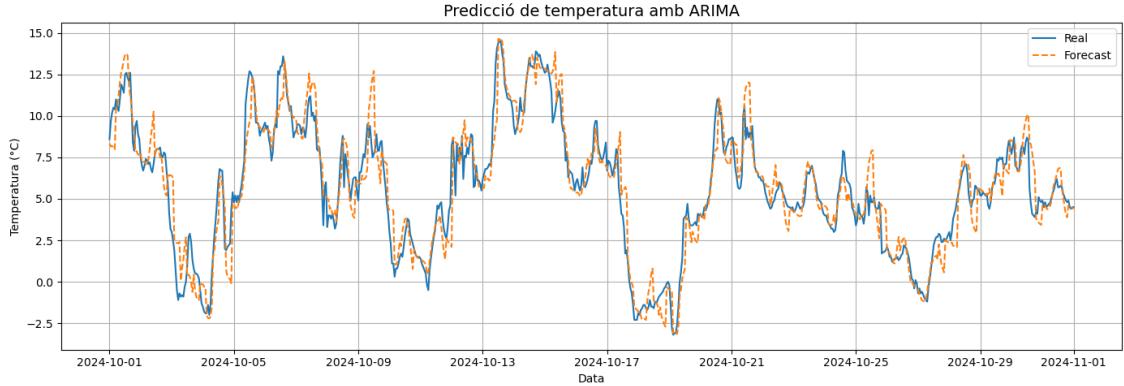


Figura 4.1: Visualització de les prediccions del model C1

5. Modelització amb models LSTM

5.1 Aplicació dels models Long Short-Term Memory (LSTM)

En la predicción de sèries temporals, s'ha popularitzat l'ús de tècniques basades en *deep learning* per la seva capacitat de captar relacions i patrons de dependència més complexos i no lineals, sobretot a mitjà i llarg termini, difícilment captables pels models estadístics clàssics com serien els ARIMA. Un dels models més utilitzats és les xarxes LSTM (*Long Short-Term Memory*).

En l'aplicació per predicción de sèries meteorològiques, els models LSTM solen ser comparats amb models clàssics com els ARIMA per avaluar el que pot aportar el *deep learning* com a millora. En general els resultats mostren que les LSTM són particularment útils en sèries no lineals amb estacionalitats (De Saa & Ranathunga, 2020; Lim & Zohren, 2021).

Diversos estudis han comparat el rendiment dels dos tipus de model aplicats a la predicción de temperatura. Tugal i Sevgin, 2023 compara els models per a casos de predicción diària d'una regió de Turquia, obtenint resultats més acurats en els models LSTM, segons les mètriques més habituals (RMSE, MSE i MAE). Tanmateix Ozbek et al., 2021 també fan aquesta comparació en regions de Turquia, però afinant a escales temporals més petites; en intervals de 10 minuts i de 1 hora, obtenint la mateixa conclusió. No obstant aquesta metodologia no sempre es clarament superior, sinó que probablement està lligada a la situació, tendència de les dades i configuració de paràmetres, ja que en l'estudi de Koçak, 2023 els models ARIMA presenten una error lleugerament més baix que les LSTM. Els autor apunten que podria ser a causa del volum baix de dades, i que aquestes presenten una tendència suau de canvi i estacionalitat marcades, per el que en aquest context podria ser millor els models auto-regressius, ja que les xarxes LSTM són més flexibles i capten millor la tendència en presència de sorolls i patrons amb canvis mes bruscos, però que en cas contrari subestimen els valors màxims i sobreestimen els mínims.

5.2 Fonament de les RNN i LSTM

Els models LSTM es basen en l'arquitectura de les *Recurrent Neural Network* (RNN), xarxes neuronals desenvolupades especialment per tractar dades en forma de seqüències, on l'ordre de les observacions és rellevant, a diferència de les altres arquitectures de xarxes neuronals, on les entrades son independents, les RNN incorporen una connexió recursiva que permet transmetre informació d'un pas temporal al següent. Això les fa especialment útils per tasques de predicción temporal, processament llenguatge natural, entre d'altres (Lim & Zohren, 2021).

Malgrat la seva utilitat, les RNN convencionals presenten una limitació important quan s'apliquen a seqüències llargues: el conegut problema del *vanishing* o *exploding gradient*. Aquest fenomen està relacionat amb el mètode d'entrenament habitual de les xarxes neuronals, basat en la retropropagació (*backpropagation*). Aquesta tècnica ajusta els pesos del model a partir de les derivades de la funció de pèrdua, amb l'objectiu de minimitzar l'error de predicción.

En el cas de les RNN, aquestes derivades es propaguen cap enrere a través de cada pas temporal. Quan les seqüències són molt llargues, aquest procés implica multiplicar moltes derivades entre si. Si aquestes tenen valors menors que 1, el producte es redueix progressivament fins a valors gairebé nuls, fet que impedeix actualitzar els pesos de les primeres etapes: és el que es coneix com a *vanishing gradient*. En canvi, si les derivades són majors que 1, el valor del gradient pot créixer exponencialment, provocant canvis inestables en els pesos ,el que s'anomena *exploding gradient*. En ambdós casos, el resultat és que la xarxa perd la capacitat d'aprendre patrons que depenen de contexts temporals llargs De Saa i Ranathunga, 2020; Lim i Zohren, 2021.

Per solucionar aquest problema, es van desenvolupar les arquitectura LSTM, que afegeixen un mecanisme específic per regular la memòria. La unitat o neurona LSTM inclou una cèl·lula de memòria i un conjunt de portes que regulen quina informació s'emmagatzema, s'actualitza o s'oblida en cada pas de temps. Aquest disseny les fa especialment robustes per capturar dependències estacionals o cicles llargs, com els que es poden donar en dades meteorològiques Tugal i Sevgin, 2023, solucionant així el problema del *vanishing / exploding gradient*.

Cada cèl·lula LSTM conté una estructura interna que li permet conservar informació rellevant durant períodes prolongats. Aquesta estructura es basa en un conjunt de mecanismes anomenats **portes**, que regulen el flux d'informació dins de la cèl·lula, com es transforma i manté la informació en cada pas temporal.

Els operadors i les funcions que el defineixen de cada cèl·lula LSTM són:

- **Porta d'oblit (f_t):** controla quina part de la memòria anterior s'ha d'eliminar.

$$f_t = \sigma(W_f \cdot x_t + U_f \cdot h_{t-1} + b_f) \quad (5.1)$$

- **Porta d'entrada (i_t):** regula quina informació nova s'incorpora a la memòria.

$$i_t = \sigma(W_i \cdot x_t + U_i \cdot h_{t-1} + b_i) \quad (5.2)$$

- **Cèl·lula candidata (\tilde{C}_t):** genera una proposta d'actualització per a la memòria.

$$\tilde{C}_t = \tanh(W_c \cdot x_t + U_c \cdot h_{t-1} + b_c) \quad (5.3)$$

- **Actualització de la memòria (C_t):** combina el que es conserva i el que s'incorpora.

$$C_t = f_t \cdot C_{t-1} + i_t \cdot \tilde{C}_t \quad (5.4)$$

- **Porta de sortida (o_t):** decideix quina part de la memòria es propaga com a sortida.

$$o_t = \sigma(W_o \cdot x_t + U_o \cdot h_{t-1} + b_o) \quad (5.5)$$

- **Sortida de la cèl·lula (h_t):** resultat final del pas actual, filtrat per la porta de sortida.

$$h_t = o_t \cdot \tanh(C_t) \quad (5.6)$$

En aquest conjunt d'equacions, x_t representa l'entrada en el pas temporal t , i h_{t-1} és la sortida (o estat ocult) del pas anterior. L'estat intern de memòria es denota com C_t , i es va actualitzant al llarg del temps en funció de tres mecanismes de control: la porta d'oblit f_t , que filtra la memòria anterior C_{t-1} ; la porta d'entrada i_t , que determina quina nova informació \tilde{C}_t s'incorpora; i la porta de sortida o_t , que regula quina part de la memòria es propaga com a sortida h_t .

Les matrius W_* i U_* contenen els pesos associats a les entrades i als estats anteriors, respectivament, i b_* són els termes de biaix de cada porta (on $* = f, i, o, c$ segons el cas). Aquest conjunt de components treballa conjuntament per controlar el flux d'informació dins la cèl·lula i facilitar l'aprenentatge de dependències temporals.

Totes aquestes operacions són regulades per les anomenades funcions d'activació $\sigma(\cdot)$ i $\tanh(\cdot)$. La funció sigmoide $\sigma(\cdot)$ s'utilitza en les portes per limitar la sortida a l'interval $[0, 1]$, el que permet interpretar el seu valor com una fracció de la informació que passa al següent estat. En canvi, la funció tangent hiperbòlica $\tanh(\cdot)$ s'aplica a les actualitzacions internes de la memòria, mantenint els valors dins l'interval $[-1, 1]$ i afavorint així l'estabilitat numèrica durant l'entrenament.

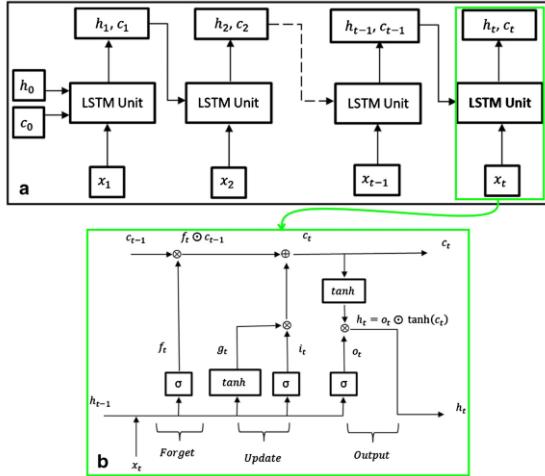


Figura 5.1: Arquitectura d'una xarxa LSTM: (a) Xarxa LSTM (b) Flux intern d'informació en una cèl·lula LSTM. Font: Ozbek et al., 2021.

A cada pas temporal t , la cèl·lula LSTM rep com a entrada x_t i l'estat ocult anterior h_{t-1} , (també anomenat estat ocult) i actualitza la seva memòria interna C_t combinant la informació conservada del pas anterior (modulada per la porta d'oblit f_t) amb la nova informació rellevant (determinada per la porta d'entrada i_t i la cèl·lula candidata \tilde{C}_t). La sortida del pas actual h_t es calcula a partir d'aquest estat intern actualitzat i és filtrada per la porta de sortida o_t , per transmetre només la informació rellevant cap a la següent cèl·lula o capa.

Aquest mecanisme flexible (fig fig. 5.1 (b)) permet que cada unitat LSTM aprengui a decidir quina informació recordar, quina oblidar i quina transmetre en funció del context seqüencial de les dades. Gràcies a aquesta capacitat, les xarxes LSTM s'han consolidat com una eina potent per modelitzar dependències temporals complexes.

Una xarxa neuronal LSTM completa està formada per múltiples d'aquestes cèl·lules organitzades seqüencialment i sovint disposades en capes (fig fig. 5.1 (a)). Cada capa rep com a entrada les sortides temporals de la capa anterior, cosa que permet a la xarxa capturar patrons cada vegada més complexos a diferents nivells de representació temporal.

5.3 Estructuració del model

5.3.1 Descripció general del plantejament

En base als fonaments teòrics de les xarxes LSTM i la naturalesa del nostre problema, s'ha dissenyat un flux de treball i una estructuració d'hiperparàmetres específica amb l'objectiu d'entrenar i confeccionar un model capaç de predir la temperatura horària amb una precisió acceptable.

Aquest flux segueix una estructura modular que permet gestionar de manera flexible i escalable totes les etapes del procés: des del preprocessament de les dades fins a la generació de prediccions. El *pipeline* implementat inclou la càrrega i tractament de les dades, l'escalat dels valors, la generació de seqüïències temporals per a l'entrenament, la definició del model LSTM amb diferents configuracions, i finalment, les estratègies de predicció i avaluació.

L'objectiu principal d'aquest esquema és desenvolupar un model robust per a la predicció a curt termini de la temperatura, tot explorant diferents configuracions arquitectòniques i hiperparàmetres que poden afectar el rendiment del model. Així, s'ha establert un conjunt d'experiments per avaluar com canvia la capacitat predictiva del model en funció de la mida de la finestra temporal, el nombre de sortides, la profunditat de la xarxa i altres paràmetres rellevants.

5.3.2 Preparació de les dades

Per a l'entrenament del model s'han escollit el dataset que s'inclou la temperatura horària des del 1-01-2020 fins a 31-12-2024, sense cap interrupció. Aquest es l'interval més recent que no mostra cap interrupció en les dades, tal com s'indica en el capítol 2.

Les dades s'han separat en tres conjunts:

- *Test*: per avaluar el rendiment real del model un cop entrenat. (3 mesos, últims del conjunt)
- *Validation*: per monitoritzar l'error. (3 mesos, entre Test i Train)
- *Train*: per ajustar els pesos del model. (42 mesos, resta de dades)

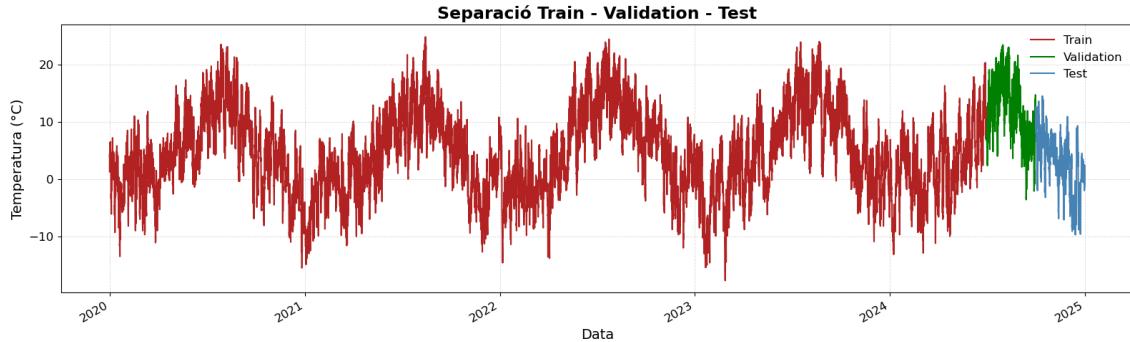


Figura 5.2: Data Splits (Train, Validation, Test)

Amb l'objectiu d'afavorir l'estabilitat numèrica durant l'entrenament i evitar valors descompensats en les funcions d'activació, una pràctica habitual és escalar els valors de les dades en l'interval $[0,1]$, utilitzant una funció escaladora (eq. (5.7)), que converteix el valor mínim en 0, el màxim en 1 i la resta en el rang intermig.

$$x_{scaled} = \frac{x - x_{min}}{x_{max} - x_{min}} \quad (5.7)$$

Aquest procés de transformació es fa només a partir del conjunt de train, i s'extrapolà la funció d'escalat als conjunts de validació i test. Aquest detall és important per evitar el fenomen conegut com a *data leaking*, en el que informació del conjunt de test influeixi en el procés d'entrenament del model. Si els valors mínim i màxim s'obtinguessin tenint en compte tot el conjunt de dades, s'estaria introduint informació futura dins la transformació, cosa que falsejaria l'avaluació real del rendiment del model. Finalment, es farà servir la funció inversa per desfer la transformació en els valors predits a la sortida.

Els models LSTM aprenen les relacions a partir de seqüències de dades per predir la seva continuació. Per això, és necessari transformar la sèrie temporal en un conjunt de seqüències d'entrada i sortida. Aquest procés consisteix a aplicar una finestra móbil de mida fixa sobre les dades escalades, i generar parelles de mostres (X, y) que permetin a la xarxa aprendre les relacions temporals.

Cada seqüència d'entrada X està formada per un conjunt de valors consecutius de la variable temperatura amb una longitud fixa de *mida de finestra* (paràmetre `window size`) hores. L'objectiu és que el model, a partir d'aquesta finestra d'informació passada, sigui capaç de predir el valor dels *horitzons de predicció* següents (paràmetre `num outputs` o `mostres de sortida`) que segueixen la sèrie de temperatura. Per tant, per a cada seqüència d'entrada, hi ha una seqüència de sortida y amb els valors que es volen predir.

Formalment, el procés de generació de seqüències es pot descriure mitjançant la construcció de parelles (X_i, y_i) , on:

$$\begin{aligned} X_i &= [x_i, x_{i+1}, \dots, x_{i+ws-1}] \\ y_i &= [x_{i+ws}, x_{i+ws+1}, \dots, x_{i+ws+nout-1}] \end{aligned} \quad (5.8)$$

En aquesta expressió, ws és la mida de la finestra d'entrada (`window size`), i $nout$ és el nombre d'horitzons temporals a predir (`num outputs`). Cada parella (X_i, y_i) representa un exemple d'entrenament, format per una seqüència de valors consecutius i la seva corresponent predicció.

L'estructura final de les dades és un tensor de la forma:

$$\mathbf{X} : (n_{\text{mostres}}, \text{window size}, 1) \quad \mathbf{y} : (n_{\text{mostres}}, \text{num outputs}) \quad (5.9)$$

Aquesta estructura s'aplica als tres conjunts de *Train*, *Validation* i *Test*, consolidant els valors de *Windows Size* i *Num Outputs* com els primers hiperparàmetres del model, ja que condicionaran l'entrenament d'aquest.

5.3.3 Arquitectura de la xarxa

L'arquitectura de la xarxa neuronal amb cel·les LSTM es defineix a partir d'un conjunt de paràmetres estructurals i d'entrenament que condicionen tant la capacitat del model com el seu comportament durant l'ajust:

- **Nombre de capes LSTM (n_{layers})**: determina la profunditat de la xarxa i la seva capacitat per captar patrons temporals complexos.
- **Nombre de neurones per capa (n_{units})**: indica quantes cel·les LSTM operen en paral·lel dins de cada capa, aportant capacitat del model.
- **Taxa de dropout**: fracció de neurones que es desactiven aleatòriament en cada iteració de l'entrenament per reduir el risc de sobre-ajustament.

Pel que fa al procés d'entrenament, els principals paràmetres que influeixen en la dinàmica de l'optimització són:

- **Nombre d'èpoques (*Epochs*)**: Nombre de vegades que el model recorre tot el conjunt d'entrenament.
- **Mida del lot (*Batch Size*)**: Nombre de mostres utilitzades en cada actualització dels pesos.
- **Aturada anticipada (*Patience*)**: Nombre d'èpoques sense millora del rendiment, sense disminució de la funció de pèrdua per interrompre l'entrenament i concloure'l a fi d'estalviar recursos computacionals

Cal destacar que, tot i que valors més elevats d'aquests paràmetres poden afavorir un millor ajustament, això no sempre es tradueix en un model més robust ni eficient. L'objectiu és trobar un equilibri entre rendiment predictiu, capacitat generalitzadora i eficiència computacional.

5.3.4 Resum d'hiperparàmetres del model

Per tal de dissenyar i ajustar el model LSTM de manera sistemàtica, s'ha definit un conjunt d'hiperparàmetres que controlen el flux de dades, l'arquitectura i el procés d'entrenament. Alguns d'aquests paràmetres s'han mantingut fixos per simplicitat computacional i consistència metodològica, mentre que els altres han estat objecte d'experimentació per analitzar el seu impacte en el model (taula 5.1).

Hiperparàmetre	Valors provats	Tipus
window size	[24, 48, 72, 96, 120, 168]	A optimitzar
n outputs	[1, 6, 12]	A optimitzar
n layers	[1, 2, 3]	A optimitzar
n units	[32, 64, 128]	A optimitzar
dropout	0.2	Fixat
batch size	256	Fixat
epochs	50	Fixat
patience	5	Fixat

Taula 5.1: Resum dels hiperparàmetres considerats en el disseny i entrenament dels models LSTM.

Els valors seleccionats per a cada hiperparàmetre s'han escollit combinant criteris pràctics i referències de la literatura. Els paràmetres com el nombre de capes, de neurones i la mida de finestra s'han variat per analitzar com afecten la capacitat del model de capturar patrons a diferents escales temporals. D'altra banda, paràmetres com *batch size* o el nombre d'èpoques s'han mantingut constants per reduir la complexitat de l'espai de cerca i facilitar la comparació entre models.

Aquesta selecció permet explorar un conjunt raonable d'arquitectures sense sobrecarregar el cost computacional ni introduir massa variabilitat entre experiments.

5.3.5 Estratègies de predicció

A l'hora d'avaluar el model sobre el conjunt de *test*, es poden aplicar diferents estratègies de predicció, cadascuna amb característiques pròpies que permeten explorar com afecta la forma de generar les prediccions a la qualitat dels resultats obtinguts. Aquestes estratègies es basen en la manera com s'utilitzen les sortides del model per predir múltiples valors consecutius de temperatura.

És important distingir entre els models amb una sola sortida ($n_{outputs} = 1$) i els que prediuen múltiples valors a la vegada ($n_{outputs} > 1$). En el primer cas s'han implementat tres estratègies diferenciades: la *Predictió Batch*, la *Predictió Iterativa* i la *Predictió Iterativa amb reinjecció*. En canvi, per als models amb múltiples sortides, només s'ha considerat la *Predictió Batch*, ja que aquests models ja generen els diversos horitzons temporals en una única passada, i les altres estratègies no aporten informació addicional.

- **Predictió batch:** el model genera diversos valors futurs d'una sola vegada a partir d'una finestra d'entrada \mathbf{X} . Aquesta estratègia és compatible tant amb models de sortida única ($n_{outputs} = 1$) com múltiple ($n_{outputs} > 1$), i permet obtenir la seqüència $\hat{\mathbf{y}}$ directament.

$$\hat{\mathbf{y}} = \text{Model}(\mathbf{X}) \quad (5.10)$$

- **Predictió iterativa:** en aquest cas el model prediu únicament un pas temporal ($n_{outputs} = 1$), i les prediccions s'encadenen de forma recursiva: la sortida \hat{y}_t del model es reutilitza com a nova entrada per calcular \hat{y}_{t+1} , i així successivament. Aquest enfocament acumula errors amb el temps i tendeix a convergir a un valor constant, però permet explorar horitzons temporals més llargs.

$$\hat{y}_{t+1} = \text{Model}([x_{t-n+1}, \dots, x_t]) \Rightarrow \hat{y}_{t+2} = \text{Model}([x_{t-n+2}, \dots, \hat{y}_{t+1}]) \quad (5.11)$$

- **Predictió iterativa amb reinjecció:** aquesta estratègia és similar a l'anterior, però incorpora una actualització periòdica amb les observacions reals. És a dir, cada cert nombre de prediccions (per defecte, cada 5 hores), els valors predictius es substitueixen pels valors reals

disponible. Això simula un escenari real amb retroalimentació externa i pot ajudar a corregir la deriva del model a llarg termini. A eq. (5.12) es pot veure un exemple de l'estratègia.

$$\begin{aligned}
 \hat{y}_{t+1} &= \text{Model}([x_{t-3}, x_{t-2}, x_{t-1}, x_t]) \\
 \hat{y}_{t+2} &= \text{Model}([x_{t-2}, x_{t-1}, x_t, \hat{y}_{t+1}]) \\
 \hat{y}_{t+3} &= \text{Model}([x_{t-1}, x_t, \hat{y}_{t+1}, \hat{y}_{t+2}]) \\
 \hat{y}_{t+4} &= \text{Model}([x_t, \hat{y}_{t+1}, \hat{y}_{t+2}, \hat{y}_{t+3}]) \\
 \hat{y}_{t+5} &= \text{Model}([\hat{y}_{t+1}, \hat{y}_{t+2}, \hat{y}_{t+3}, \hat{y}_{t+4}]) \quad (\text{reinjecció}) \\
 \hat{y}_{t+6} &= \text{Model}([\textcolor{red}{x_{t+2}}, \textcolor{red}{x_{t+3}}, \textcolor{red}{x_{t+4}}, \textcolor{red}{x_{t+5}}])
 \end{aligned} \tag{5.12}$$

5.4 Plantejament de models

Amb l'objectiu d'avaluar el rendiment dels models LSTM en la predicció de temperatura horària, s'ha dissenyat una sèrie d'experiments que permeten analitzar l'efecte dels diferents paràmetres sobre la qualitat de les prediccions. La idea és identificar configuracions que ofereixin un bon equilibri entre precisió, simplicitat i robustesa.

Els experiments s'han dissenyat amb l'objectiu de respondre diverses qüestions pràctiques sobre l'arquitectura i el comportament dels models. Per tal d'estruccurar aquesta anàlisi de manera clara i sistemàtica, els experiments s'han agrupat en blocs segons el paràmetre principal que es vol estudiar en cada cas (taula 5.2).

- Analitzar com influeix la mida de la finestra d'entrada (`window_size`) en la capacitat del model per captar les dinàmiques temporals, combinat amb diverses sortides i profunditats de xarxa.
- Comparar models amb sortida única (`n_outputs = 1`) i models multioutput de 6 o 12 hores.
- Avaluar com afecta la profunditat de la xarxa (`n_layers`) i el nombre de neurones per capa (`n_units`) al rendiment.

ID	window_size	n_outputs	n_layers	n_units	Notes
BSL	24	1	1	32	Capacitat mínima
Exp0	24	1	2	64	Model senzill, horitzó curt
Exp1	96	1	3	128	Model potent 1 hora
Exp2	96	6	2	64	Multioutput 6 hores
Exp3	96	12	3	128	Multioutput 12 hores
Exp4	168	1	3	64	Seqüència llarga
Exp5	48	1	1	32	Model lleuger, realista
Exp6	120	1	3	64	Context llarg
A0	96	1	1	64	Xarxa senzilla
A1	96	1	2	64	Xarxa mitjana
A2	96	1	3	64	Xarxa profunda
B0	96	1	2	32	Xarxa petita
B1	96	1	2	64	Capacitat intermèdia
B2	96	1	2	128	Xarxa gran
C0	96	1	1	32	Model lleuger
C1	96	1	1	64	Equilibri senzill
C2	96	1	1	128	Alta capacitat plana

Taula 5.2: Configuracions experimentals dels models LSTM. S'identifiquen les sèries d'experiments (finals, A, B i C) i les variants de paràmetres més rellevants.

5.5 Resultats

En la taula 5.3 es poden veure les mètriques per avaluar els diferents models LSTM entrenats tant amb l'estrategia de predicció *batch* (directa) com amb la de *reinjecció* (recursiva corregida). En general, en els models amb una sola sortida, la predicció *batch* sol donar molt bons resultats, amb $\text{RMSE} < 1$. Això és esperable, ja que el model disposa de tota la seqüència passada per predir només una hora endavant, la següent dada, cosa que té poc valor pràctic en el context real del problema. Per això, resulta més rellevant avaluar el rendiment en l'escenari de *reinjecció*, on les prediccions del model es van reutilitzant i acumulant, simulant un escenari real.

S'observa que el model base (BSL) presenta un rendiment força limitat, amb valors d'error relativament alts, especialment en la predicció amb reinjecció, on l'error creix de forma important. Aquest model, amb una arquitectura mínima, serveix com a punt de referència per quantificar les millores obtingudes la resta de models, establint una mena de llindar mínim de predicció.

ID	Predicció batch			Predicció reinjecció		
	RMSE	MSE	MAE	RMSE	MSE	MAE
BSL	0.99267	0.8588	0.7212	2.3256	5.4085	1.7917
Exp0	0.7883	0.6214	0.542	1.6214	2.6289	1.1492
Exp1	0.8358	0.6986	0.6026	1.7257	2.9779	1.2662
Exp2	1.7069	2.9136	1.2301	-	-	-
Exp3	2.8366	8.0462	2.1109	-	-	-
Exp4	0.8239	0.6788	0.5941	1.8156	3.2965	1.3427
Exp5	0.8696	0.7563	0.6538	2.0572	4.2319	1.5592
Exp6	0.7685	0.5906	0.5166	1.52	2.3104	1.0436
A0	0.8468	0.7171	0.6298	2.047	4.1901	1.5608
A1	0.8003	0.6450	0.5611	1.6095	2.5906	1.1565
A2	0.7955	0.6328	0.5488	1.5005	2.2516	1.0535
B0	0.847	0.7175	0.6395	1.9343	3.7415	1.4798
B1	0.8003	0.6405	0.5611	1.6095	2.5906	1.1565
B2	0.7994	0.6391	0.5965	1.6566	2.7443	1.1955
C0	0.9597	0.9211	0.747	2.1274	4.9169	1.7294
C1	0.8443	0.7128	0.6268	1.9396	3.7619	1.4767
C2	0.7538	0.5683	0.5220	1.5440	2.3840	1.0966

Taula 5.3: Resultats dels models LSTM: mètriques de predicción per a batch i reinjecció

Els models de la sèrie Exp representen una exploració diversa de configuracions: des de arquitectures lleugeres (com Exp0 i Exp5) fins a models profunds i amb múltiples sortides (Exp2 i Exp3). Es pot destacar que els models amb una sola sortida (`n_outputs = 1`) i més profunditat (Exp1, Exp4, Exp6) milloren considerablement el rendiment respecte al model base.

Contràriament, les configuracions amb sortides múltiples (Exp2, Exp3) mostren valors d'error clarament superiors. En aquests casos, tot i que la tendència general de les prediccions sembla seguir la sèrie real, sovint es produeix un desfasament temporal, clarament visible en els gràfics de predicción (fig. 5.3b). Probablement aquest fenomen es causat per una acumulació d'error entre les varietats de previsions que surten del model, cosa que li dificulten anticipar canvis al anar predint en bloc, de manera que per minimitzar l'error aquest acaba assimilant valors pròxims als anteriors, fent, sense planificar-ho, una estratègia similar a la que duen a terme els models de persistència, cosa que segurament origina el desfasament.

Per intentar abordar aquest problema, es va experimentar amb el paràmetre `lookahead`, desplaçant la finestra d'entrada temporalment per ajudar al model a situar-se millor en el context. Tot i això, els resultats no van mostrar una millora clara, cosa que suggerix que el desfasament no es corregeix només ajustant l'entrada, sinó que pot ser necessari augmentar la capacitat del model o bé incorporar més informació contextual (per exemple, afegir dades sobre el dia i la hora).

El model Exp6 (fig. 5.3a), amb finestra llarga, 3 capes, i un nombre de neurones mig (64), ofereix els millors resultats globals amb RMSE de 0.768 en batch i 1.52 amb reinjecció.

Les sèries d'experiments A, B i C exploren de manera sistemàtica l'efecte de la profunditat i la capacitat del model. La Sèrie A es basa en una profunditat de xarxa fixa, on els resultats mostren millora progressiva en les mètriques a mesura que augmenta la profunditat (fig. 5.3c). L'arquitectura amb 3 capes (A2) presenta els millors resultats dins la sèrie, evidenciant que pot captar patrons més complexos.

Per altra banda, la sèrie B i C es basen en l'extensió de les capes. En cas de la B, igual profunditat (2 capes), es comprova que augmentar les neurones millora lleugerament l'error, tot i que a partir de 64 unitats (model B1) el guany addicional és reduït. En contrapartida, la sèrie C, representa

models plans, on es manté la profunditat mínima (1 capa) i es varia la capacitat. Els resultats indiquen que una única capa amb prou neurones (C2) pot obtenir resultats molt competitius, fins i tot comparables amb models més complexos (fig. 5.3d).

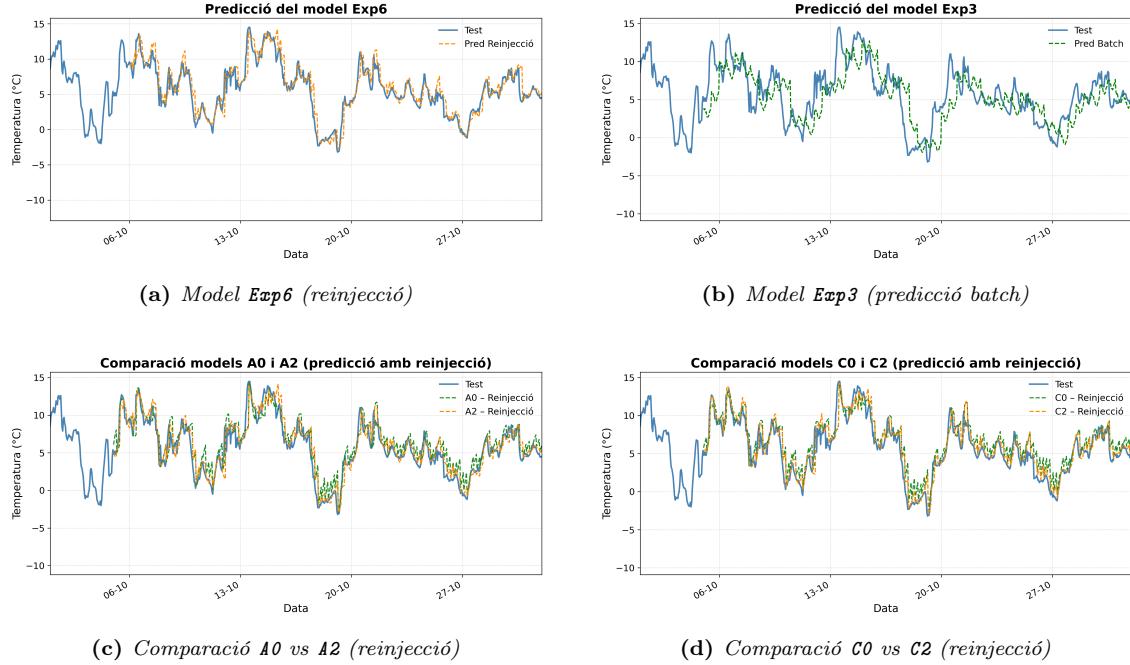


Figura 5.3: Visualització de les prediccions obtingudes pels diferents models LSTM destacats.

6. Avaluació comparativa dels models

Un cop seleccionats els millors models de cada enfocament, es comparen directament els resultats obtinguts amb SARIMA C1 i LSTM Exp6. A la taula 6.1 es mostren les mètriques d'error per a cada model, mentre que la fig. 6.1 permet visualitzar les prediccions i l'error absolut acumulat al llarg del període.

Els resultats mostren que ARIMA dona millors mètriques en tots els casos, amb una diferència notable en RMSE i MSE. El gràfic també ho confirma: tots dos models segueixen força bé la tendència, però el SARIMA ajusta millor les oscil·lacions i genera menys error acumulat. L'LSTM, en canvi, tendeix a desviar-se lleugerament a mesura que reutilitza les seves pròpies prediccions.

Aquest comportament s'explica per la naturalesa de la sèrie, que és força estable, amb una estacionalitat diürna molt marcada i un component aleatori relativament suau. En aquest context, un model clàssic com el SARIMA ja és capaç de capturar les principals tendències amb una bona precisió. Ara bé, en escenaris més complexos, com poden ser estacions de fons de vall, amb major presència de fenòmens no lineals, inversions tèrmiques abruptes o canvis sobtats de règim meteorològic, és probable que els models LSTM puguin oferir avantatges més clars.

Model	ID	RMSE	MSE	MAE
ARIMA	C1	1.3144	1.7279	0.9371
LSTM	Exp 6	1.52	2.3104	1.0436

Taula 6.1: Comparativa de les mètriques d'error entre els millors models ARIMA i LSTM seleccionats.

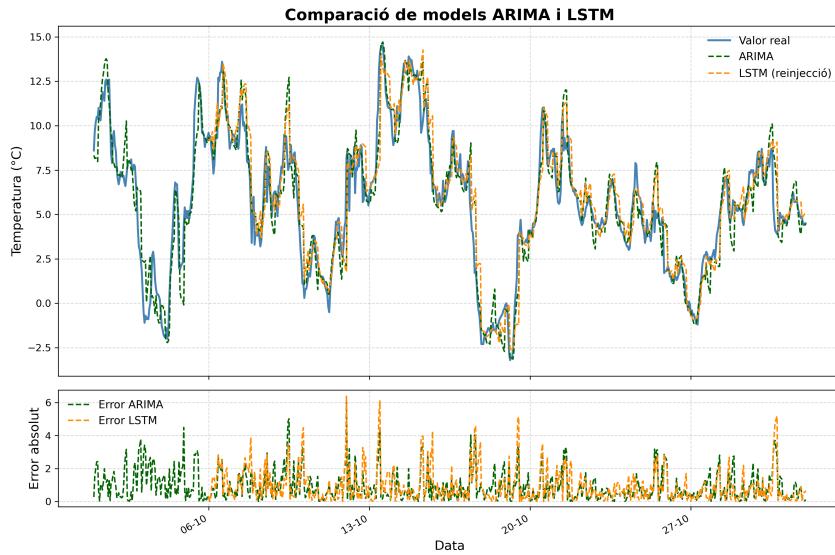


Figura 6.1: Comparació de les prediccions i errors absoluts entre models ARIMA i LSTM

7. Resum, conclusions i treball futur

Aquest treball ha explorat la viabilitat de diferents models per a la predicción horària de la temperatura a molt curt termini, utilitzant dades reals de l'estació de la Bonaigua. S'han implementat i comparat tres enfocaments representatius: les cadenes de Markov, per a una predicción categòrica de base probabilística; els models estadístics ARIMA/SARIMA, com a referència clàssica per a sèries temporals contínues, i les xarxes neuronals LSTM, com a alternativa basada en *deep learning*. De la seva aplicació se n'han extret diverses conclusions.

En primer lloc, les cadenes de Markov, tot i la seva simplicitat, han estat capaces de capturar els patrons dominants dels estats de precipitació en escenaris hivernals. No obstant això, presenten una capacitat predictiva limitada per a transicions fines, i no són aplicables directament a variables contínues com la temperatura, tot i que poden resultar útils com a aproximació per al càlcul d'estats meteorològics d'interès.

Els models ARIMA i SARIMA han ofert una base sòlida per a la predicción a curt termini, amb resultats competitius i un cost computacional moderat. En particular, els models SARIMA han demostrat capacitat per capturar l'estacionalitat diürna amb configuracions relativament senzilles, sense necessitat d'elevada complexitat.

Per altra banda els models LSTM, tot i aportar bons resultats, no han assolit l'avantatge esperat respecte als models estadístics que planteja la bibliografia. Això suggereix que, en problemes de predicción horària a molt curt termini, la dinàmica de la sèrie pot ser més difícil de capturar del que es podria pensar inicialment. Les dades són molt contínues, i és fàcil que els errors es propaguin i s'acumulin ràpidament, desviant les prediccions. Malgrat això, s'ha observat que arquitectures més senzilles, com les xarxes d'una sola capa, poden ser tan eficients com estructures més profundes, fet que posa en relleu la dificultat de calibrar l'equilibri entre simplicitat i robustesa, evitant tant el sobreajustament com l'esbiaix.

Tanmateix, és possible que en estacions amb una dinàmica més no lineal, com poden ser les de fons de vall o amb forts contrastos tèrmics, els models LSTM puguin mostrar un rendiment superior respecte als estadístics. L'anàlisi d'aquests entorns més complexos constitueix una línia de treball futura molt interessant.

Un aspecte important a considerar és que, tot i que valors de RMSE al voltant d'1 °C podrien considerar-se molt bons en contextos de predicción diària, en el cas d'aquest estudi, centrat en la predicción horària a molt curt termini, aquests resultats són relativament modestos. Donada la proximitat temporal entre observació i predicción, s'esperaria que els models fossin capaços d'assolir errors més baixos.

Aquest projecte s'ha centrat en la base d'un sistema de predicción horària de temperatura mitjançant tècniques d'aprenentatge automàtic, però obre la porta a múltiples línies de millora i ampliació. Una primera direcció clara seria la incorporació de noves variables predictives a l'entrenament dels models LSTM, com la codificació del dia de l'any, l'hora del dia o altres variables meteorològiques, que permetrien a la xarxa disposar de més context per fer una predicción acurada.

A més a més, també seria interessant explorar altres models com Prophet, àmpliament utilitzat en contextos de sèries temporals, o bé provar arquitectures alternatives basades en LSTM. Algunes opcions serien entrenar models específics per a diferents horitzons temporals i combinar-ne les prediccions, o bé desenvolupar models especialitzats per a mesos o estacions concretes de l'any, amb l'objectiu de capturar millor les variacions estacionals.

Paral·lelament, una millora clara seria l'automatització del procés de selecció d'hiperparàmetres, mitjançant tècniques de cerca eficient com la validació creuada o la cerca bayesiana, amb l'objectiu de trobar la combinació òptima per maximitzar el rendiment i evitar el sobre-ajustament.

Finalment, una línia especialment prometedora seria l'exploració de models més avançats com les *Neural Ordinary Differential Equations* (Neural ODEs), que permetrien modelitzar de manera més explícita la dinàmica contínua del sistema. Tot i no haver-se implementat en aquest estudi, la seva aplicació podria obrir nous camins per representar millor l'evolució temporal de variables meteorològiques i millorar la interpretabilitat dels models neuronals en contextos físics, constituint un possible pont entre els models basats en dades i els models meteorològics tradicionals.

Referències

- Box, G. E., Jenkins, G. M., Reinsel, G. C., & Ljung, G. M. (2015). *Time series analysis: forecasting and control*. John Wiley & Sons. (V. la pàg. 8).
- De Saa, E., & Ranathunga, L. (2020). Comparison between arima and deep learning models for temperature forecasting. *arXiv preprint arXiv:2011.04452* (v. les pàg. 8, 9, 13).
- Elseidi, M. (2024). A hybrid Facebook Prophet-ARIMA framework for forecasting high-frequency temperature data. *Modeling Earth Systems and Environment*, 10(2), 1855 - 1867 (v. les pàg. 8, 10).
- Gabriel, K., & Neumann, J. (1962). A Markov chain model for daily rainfall occurrence at Tel Aviv. *Quarterly Journal of the Royal Meteorological Society*, 88(375), 90-95 (v. la pàg. 3).
- Gutiérrez, J. M., Cano, R., Cofiño, A. S., & Sordo, C. M. (2004). *Redes probabilísticas y neuronales en las ciencias atmosféricas*. Ministerio de Medio Ambiente. (V. la pàg. 3).
- Koçak, H. (2023). Time Series Prediction of Temperature Using Seasonal ARIMA and LSTM Models. *Gazi Mühendislik Bilimleri Dergisi*, 9(3), 574-584 (v. les pàg. 8-10, 13).
- Lefebvre, M. (2019). Modelling and forecasting temperature and precipitation in Italy. *Atti della Accademia Peloritana dei Pericolanti-Classe di Scienze Fisiche, Matematiche e Naturali*, 97(2), 2 (v. la pàg. 4).
- Lim, B., & Zohren, S. (2021). Time-series forecasting with deep learning: a survey. *Philosophical Transactions of the Royal Society A*, 379(2019), 20200209 (v. la pàg. 13).
- Liu, T. (2010). Application of Markov chains to analyze and predict the time series. *Modern Applied Science*, 4(5), 162 (v. la pàg. 3).
- Murat, M., Malinowska, I., Gos, M., & Krzyszczak, J. (2018). Forecasting daily meteorological time series using ARIMA and regression models. *International agrophysics*, 32(2) (v. les pàg. 8-10).
- Ozbek, A., Sekertekin, A., Bilgili, M., & Arslan, N. (2021). Prediction of 10-min, hourly, and daily atmospheric air temperature: comparison of LSTM, ANFIS-FCM, and ARMA. *Arabian Journal of Geosciences*, 14, 1-16 (v. les pàg. 13, 15).
- Tugal, I., & Sevgin, F. (2023). Analysis and forecasting of temperature using time series forecasting methods a case study of Mus. *Thermal Science*, 27(4 Part B), 3081-3088 (v. les pàg. 8, 13, 14).