



UNIVERSIDAD DE BUENOS AIRES
Facultad de Ciencias Exactas y Naturales
Departamento de Ciencias de la Atmósfera y los Océanos

Cuantificación de la incertidumbre en los pronósticos usando redes neuronales: aplicación en asimilación de datos

Tesis presentada para optar por el título de Doctor de la Universidad de Buenos Aires
en el área Ciencias de la Atmósfera y los Océanos

Lic. Maximiliano Antonio Sacco

Director de Tesis: *Dr. Juan Jose Ruiz*

Co-Director de Tesis: *Dr. Manuel Pulido*

Consejero de estudios: *Dr. Enrique Carlos Segura*

Lugar de Trabajo: Centro de Investigaciones del Mar y de la Atmósfera. CONICET-UBA
Servicio Meteorológico Nacional Argentino.

Buenos Aires, Agosto 2023

Cuantificación de la incertidumbre en los pronósticos usando redes neuronales: aplicación en asimilación de datos

Resumen

Hasta el momento, la metodología más usada para la estimación de la incertidumbre en los pronósticos meteorológicos es la técnica de ensambles. En esta tesis buscamos estimar la incertidumbre usando redes neuronales artificiales (RNAs) a partir de un único pronóstico determinístico (una única integración del modelo). Para esto se entrenaron RNAs usando una nueva función de costo basada en el error, en inglés extended Mean Squared Error (eMSE), y se estudió la calidad de la incertidumbre estimada comparándola con entrenamientos con funciones de costo tradicionales (función de verosimilitud) y con la técnica de ensambles. En primera medida, se evaluó la estimación de la varianza de los pronósticos. Las redes mostraron un buen desempeño para capturar la incertidumbre (varianzas) en un entorno de prueba basado en el modelo dinámico multi-escala de Lorenz'96 que simula las características de la predictibilidad atmosférica, tanto en un escenario de modelo perfecto (donde se analiza el impacto del error presente en las condiciones iniciales), como en un escenario de modelo imperfecto (cuando el modelo numérico no puede representar todas las escalas del problema e introduce error en el pronóstico). En un desarrollo posterior, se entendió la función de costo eMSE para la estimación de la matriz de covarianza completa. Esta metodología resultó ser mucho más estable numéricamente y más rápida de entrenar comparado con funciones de costo basadas en la verosimilitud que necesitan invertir la matriz de covarianza. Se diseñó un método de asimilación de datos, en inglés Data Assimilation (DA) que combina la técnica del filtro de Kalman con la estimación de matriz de covarianza del error de pronóstico (MCEP) usando RNAs que denominamos UnnKF. Esta metodología nos permitió generar largas secuencias de ciclos de DA (más de 15.000 ciclos consecutivos) usando observaciones parciales del estado del sistema. Se obtuvo una secuencia de análisis cuyo error cuadrático medio (MSE) es comparable al obtenido por la técnica del filtro de Kalman por ensambles, en inglés Ensemble Kalman Filter (EnKF) usando grandes ensambles y logra disminuir considerablemente el MSE de los análisis comparado con un ensamble pequeño.

Palabras Clave: dinámica caótica, modelos de alta dimensionalidad, red neuronal, cuantificación de la incertidumbre, estimación de la covarianza, asimilación de datos

Quantification of forecast uncertainty using neural networks: application to data assimilation

Abstract

The ensemble technique is currently the most widely used method for estimating uncertainty in weather forecasts. The drawback of this technique is that it requires multiple numerical integrations of models and thus a lot of computational power. In this thesis we attempt to estimate uncertainty using Artificial Neural Networks (RNA) and a single deterministic prediction (a single model integration). For this purpose, RNA were trained with a new cost function (eMSE) and the quality of the uncertainty estimated in this way was studied by comparing it with training with traditional cost functions (likelihood function) and with the ensemble technique. First, the estimation of the variance of the forecasts was evaluated. The networks showed good performance in capturing uncertainty (variances) in a test environment based on the Lorenz'96 multiscale dynamic model that simulates the characteristics of atmospheric predictability, both in a perfect model scenario (where the impact of the error present in the initial conditions is analysed) and in an imperfect model scenario (when the numerical model cRNAot represent all the scales of the problem and introduces errors in the forecast). We also investigated extending the eMSE cost function to estimate the full covariance matrix. This method proved to be much more numerically stable and faster to train than likelihood-based cost functions, which require the covariance matrix to be inverted. In addition to analysing the structural characteristics of the estimated covariances, a hybrid data assimilation system (UnnKF) based on the Kalman filter was designed to provide an analysis of the system state using the covariances estimated by the RNA, partial observations of the system and a deterministic forecast. The results obtained are very encouraging. The covariances estimated by RNA trained with the eMSE cost function and the hybrid UnnKF methodology robustly estimate long assimilation sequences (more than 15,000 consecutive cycles), obtaining an analysis sequence whose mean squared error (MSE) is comparable to that obtained by the ensemble assimilation technique (EnKF) for large ensembles, and manages to significantly reduce the MSE of the analysis compared to a small ensemble.

Keywords: chaotic dynamic, high-dimentional models, neural network, uncertainty quantification, covariance estimation, data assimilation

A mis padres, Teresa y Antonio, por todo lo que hicieron a lo largo de mi vida y a Simón por acompañarme en estos años de mucho crecimiento.

Agradecimientos

Especial agradecimiento a la Dra. Yanina Garcia Skabar que fue quien me invito a doctorarme.

Y también al Programa ECOS-Sud por su apoyo financiero a través del proyecto A17A08. Esta investigación también ha sido apoyada por la Agencia Nacional de Promoción Científica y Tecnológica de Argentina (subvención n° PICT-2233-2017, PICT-SERIEA-01168, PICT-2021-CAT-I-130), la Universidad de Buenos Aires (subvención n° UBACyT - 20020170100504). Un agradecimiento especial a la Directora del Servicio Meteorológico Nacional de Argentina, la Dra. Celeste Saulo por su apoyo y confianza en este trabajo.

Índice

Resumen	III
Abstract	V
Agradecimientos	IX
1. Introducción	5
2. Marco teórico	13
2.1. Modelos dinámicos estocásticos	13
2.2. Asimilación de datos	15
2.2.1. Asimilación de datos secuencial	15
2.2.2. Filtro de Kalman por ensambles	18
2.3. Redes neuronales artificiales	20
2.3.1. Redes completamente conectadas	21
2.3.2. Entrenamiento de redes neuronales	23
2.3.3. Redes neuronales convolucionales	30
3. Estimación de la incertidumbre	33
3.1. Entrenar directamente de un ensamble	34
3.2. Entrenar usando la verosimilitud de los datos	38
3.3. Entrenar usando la métrica CRPS	43
3.4. Propuesta: entrenar directamente del error	45
3.4.1. Estimación basada en el error de pronóstico	47
3.5. Análisis comparativo	49
3.5.1. Análisis de convergencia	49

3.5.2.	Ejemplo de estimación de la incertidumbre usando redes neuronales	50
4.	Pronósticos Probabilísticos	55
4.1.	Diseño experimental	55
4.1.1.	Escenario de modelo perfecto	56
4.1.2.	Escenario de modelo imperfecto	57
4.2.	Metodología	58
4.2.1.	Entrenamiento de la red de estados medios	59
4.2.2.	Entrenamiento de la red de varianza	60
4.2.3.	Arquitectura y entrenamiento de la red	62
4.3.	Métricas de verificación	64
4.3.1.	Error cuadrático medio	64
4.3.2.	Correlación desviación estándar/error absoluto	65
4.3.3.	Histogramas PIT	65
4.3.4.	Probabilidad de cobertura	66
4.4.	Resultados	66
4.4.1.	Sensibilidad a los datos de entrada	66
4.4.2.	Experimentos con modelo perfecto	68
4.4.3.	Experimentos con modelo imperfecto	74
4.4.4.	Análisis del rendimiento de las estimaciones	77
4.5.	Aplicación a modelos de pronóstico numérico de la atmósfera	80
4.5.1.	Conjunto de datos	80
4.5.2.	Red neuronal	81
4.5.3.	Resultados	81
4.6.	Conclusiones	84
5.	Estimación de la matriz de covarianza	87
5.1.	Estimación de la covarianza usando verosimilitud	89
5.2.	Estimación de la covarianza usando análogos	91
5.3.	Estimación de la covarianza usando el error	94
5.4.	Análisis comparativo	95
5.4.1.	Experimentación	95
5.4.2.	Resultados	96
5.4.3.	Conclusiones	97

6. Asimilación de datos	99
6.1. Método híbrido de asimilación de datos (UnnKF)	100
6.2. Diseño experimental	102
6.2.1. Modelo dinámico y conjunto de entrenamiento	103
6.2.2. Arquitectura de las RNA	104
6.2.3. Entrenamiento de las RNAs	105
6.2.4. Estimación del error de pronóstico	106
6.3. Resultados	108
6.3.1. Sensibilidad a la aproximación del error del pronóstico	110
6.3.2. Sensibilidad a la localización	112
6.3.3. Sensibilidad a la dimensión del vector de estados	113
6.3.4. Sensibilidad a la calidad del objetivo de entrenamiento	115
6.4. Conclusiones	117
7. Conclusiones	119
Bibliografía	126

Siglas

η_{lr} tasa de aprendizaje, en inglés Learning Rate. 24, 25, 62, 106

3Dvar variacional tridimensional. 8, 18, 100, 101

4Dvar variacional cuatridimensional. 8, 125

ACCRUE función de costo basada en la precisión y la confiabilidad, en inglés ACCurate and Reliable Uncertainty Estimate. 43, 45

AnDA asimilación de datos por análogos, en inglés Analog Data Assimilation. 91

BS BS, del inglés Brier Score. 7

CI condición inicial. 33

CNN red neuronal convolucional, en inglés Convolutional Neural Network. 6, 7, 32, 35, 123

CNNs redes neuronales convolucionales, en inglés Convolutional Neural Networks. 30, 35, 36

CRPS CRPS, del inglés Continuous Ranked Probability Score. 7, 37, 43, 44, 46, 74

DA asimilación de datos, en inglés Data Assimilation. 5, 6, 8–16, 18, 34, 87, 91, 99, 100, 108, 109, 120, 122, 125

EMI escenario de modelo imperfecto. 57, 62–64, 67, 74, 75, 77, 103, 108, 119

EMP escenario de modelo perfecto. 56, 58, 62–64, 68, 73–75, 95, 119, 121

- eMSE** función de costo basada en el error, en inglés extended Mean Squared Error. 47–54, 87, 94–97, 108, 118–120, 123
- EnKF** filtro de Kalman por ensambles, en inglés Ensemble Kalman Filter. 8, 18–20, 100, 101, 103, 108, 110, 114–118, 125
- GAT** del inglés Graph Attention Network. 124
- GN** del inglés Graph Network. 124
- IRN** Implicit Neural Representations. 124
- KF** filtro de Kalman, del inglés Kalman Filter. 19, 99
- LEnKF** filtro de Kalman local por ensambles ([Brusdal et al., 2003](#); [Evensen, 2003](#); [Ott et al., 2004](#)). 93
- LKHD** función de costo basada en la verosimilitud gaussiana, del inglés Likelihood. 42–44, 46, 49, 50, 53, 54, 87, 88, 94–97, 120
- LSTM** del inglés Long-Short Term Memory. 42, 124
- MCEP** matriz de covarianza del error de pronóstico. 87–91, 94–97, 99–102, 104–116, 118–120, 122
- ML** aprendizaje automático, en inglés Machine Learning. 6, 8, 9, 18, 26, 124, 125
- MLP** del inglés multi-layer Perceptron. 123, 124
- MOOC** cursos en línea masivos, del inglés Masive Open Online Courses. 25–27
- MPN** modelo de pronóstico numérico. 41
- MSE** función de costo basada en el error cuadrático medio, en inglés Mean Squared Error. 48, 59–61
- ODE** ecuaciones diferenciales ordinarias, del inglés Ordinary Differential Equations. 95
- OI** interpolación óptima, en inglés Optimal Interpolation. 8, 18, 88, 100, 122

- PDF** función de densidad de probabilidad, en inglés Probability Density Function. 9, 16, 37, 39, 71, 94
- RMSE** raíz cuadrada del error cuadrático medio, en inglés Root Mean Squared Error. 36, 103, 107–119
- RNA** red neuronal artificial. 6–10, 12, 18, 20, 21, 23–28, 30, 34–43, 45, 48, 51, 63, 64, 66–69, 71, 76, 77, 79, 80, 84, 85, 87, 89, 94–96, 99–106, 108, 110–113, 115, 116, 118, 120, 121, 124
- RNAs** redes neuronales artificiales. 6, 9, 13, 33, 34, 50, 51, 54, 55, 58, 61, 62, 68, 71, 73–75, 77, 81, 83, 84, 87, 92, 95, 97, 99, 103, 106, 115, 116, 119, 120, 122
- RS** del inglés Reliability Score. 43–45
- SIREN** del inglés Implicit Neural Representations with Periodic Activation Functions. 124
- UnnKF** método híbrido de asimilación de datos basado en el filtro de Kalman y usando redes neuronales. 99–103, 105, 106, 108, 110–114, 116–118, 120, 122
- VAE** autoencoders variacionales, del inglés Variational Auto Encoder. 6, 92–94
- WD** decaimiento de pesos, en inglés Weight Decay. 29, 41, 62, 106
- WIRE** del inglés Wavelet Implicit Neural Representations. 124

Acronyms

Capítulo 1

Introducción

El pronóstico oficial del tiempo para la República Argentina se genera en el Servicio Meteorológico Nacional (SMN). Es en la oficina de pronóstico dependiente de la Dirección Nacional de Pronósticos y Servicios para la Sociedad, donde un grupo de expertos pronostican eventos meteorológicos futuros basándose en múltiples fuentes de información que describen el estado actual de la atmósfera, por ejemplo, satélites, radares y estaciones meteorológicas. También se utilizan modelos numéricos que emplean las ecuaciones físicas que rigen el comportamiento de la atmósfera y las resuelven mediante sofisticados métodos matemáticos para simular estados futuros de la misma. Al momento de escribir esta tesis, hace más de ocho años que trabajo en la Dirección de Productos de Modelación Ambiental y Sensores Remotos (DMSR) que es donde, entre otras cosas, se diseña, planifica y desarrolla el sistema de pronóstico numérico del SMN. Como informático, entre otras tareas, desarrollé y automaticé la ejecución operativa del modelo de pronóstico numérico que se ejecuta cuatro veces al día todos los días del año. Así fue como comencé a interesarme en la problemática de la incertidumbre en los pronósticos, los pronósticos probabilísticos, el costo computacional asociado y la asimilación de datos, en inglés Data Assimilation (DA). Y así comencé mi trabajo doctoral, motivado por la idea de desarrollar nuevas técnicas que permitan cuantificar de manera más precisa y con menor costo computacional la incertidumbre asociada a los pronósticos.

La magnitud y la estructura de la incertidumbre en los pronósticos meteorológicos son dependientes del estado del sistema y se atribuye principalmente al error presente en las condiciones iniciales y a los errores del modelo. El primero está asociado al conocimiento imperfecto del estado inicial del sistema. Los últimos se producen por errores de truncamiento y una representación inexacta de los procesos de pequeña escala en

los modelos numéricos de pronóstico. Mientras que los errores del modelo pueden reducirse mediante una mayor resolución horizontal y mejores parametrizaciones físicas, es decir mejorando el modelo, los errores en las condiciones iniciales pueden reducirse mejorando las redes de observación y los sistemas de DA (Carrassi et al., 2018).

En la actualidad, la incertidumbre asociada a los pronósticos suele cuantificarse mediante el uso de ensambles de pronóstico. Para producir un pronóstico por ensamble, se realizan varias simulaciones numéricas utilizando diferentes condiciones iniciales adecuadamente seleccionadas (p. ej. Kalnay, 2003) y distintos enfoques para representar la imperfección en el modelo, como la representación de errores paramétricos y estocásticos (p. ej. Hagedorn et al., 2012). Una de las principales limitaciones de los sistemas de pronóstico por ensamble radica en su elevado costo computacional y en la complejidad de su diseño para captar y cuantificar adecuadamente las diferentes fuentes de incertidumbre, en particular los errores del modelo.

Como ha señalado Vannitsem et al. (2021), el error sistemático en los valores pronosticados, así como las deficiencias en la cuantificación de su incertidumbre, pueden mitigarse mediante diferentes técnicas de posprocesamiento de los pronósticos. Varias técnicas, como la regresión por análogos (Hamill and Whitaker, 2006; Lguensat et al., 2017), el uso de bosques aleatorios (del inglés random forest) (Taillardat et al., 2016; Schlosser et al., 2019) y (Herman and Schumacher, 2018), las redes neuronales (Rasp and Lerch, 2018; Bremnes, 2020), redes neuronales difusas (Lu et al., 2021) y autoencoders variacionales, del inglés Variational Auto Encoder (VAE) (Grooms, 2021; Yang and Grooms, 2021), se han aplicado con éxito para el aumento de la precisión del pronóstico, así como para proporcionar una cuantificación fiable de su incertidumbre.

En Haupt et al. (2021); Marzban (2003); Amemiya et al. (2023) se utilizan redes neuronales artificiales (RNAs) para reducir el error sistemático, especialmente en situaciones en las que su relación el estado del sistema es altamente no lineal. Farchi et al. (2021a); Bonavita and Laloyaux (2020) utilizan técnicas de aprendizaje automático, en inglés Machine Learning (ML) para representar el error del modelo. En Farchi et al. (2021a) utilizan ML para representar los errores de modelo en un sistema de asimilación de datos y la propia asimilación para llevar adelante la actualización secuencial de los parámetros que definen el modelo de ML a partir de la información que proveen las observaciones.

Recientemente se han propuesto modelos de red neuronal artificial (RNA) para estimar de forma conjunta el componente sistemático del error del pronóstico y la incertidumbre del mismo. Scher and Messori (2018) diseñaron una metodología basada en una red neuronal convolucional, en inglés Convolutional Neural Network (CNN) que aprendió

la desviación estándar del error a partir de un sistema de ensamble existente. Las CNNs predicen la distribución espacial de la varianza del error del pronóstico utilizando un pronóstico determinista como entrada. El objetivo de entrenamiento es la varianza del error estimada a partir de un pronóstico por ensamble. La habilidad de la red para predecir la incertidumbre es menor que la obtenida con un ensamble de pronósticos, pero la red proporciona una cuantificación razonable de la incertidumbre a un coste computacional mucho menor que el ensamble de pronósticos. Además, la red funciona mejor que otros enfoques de bajo costo computacional, como una simple estimación de la incertidumbre basada en una regresión lineal múltiple. En la misma línea, [Grönquist et al. \(2021\)](#) utilizó una RNA para mejorar la cuantificación de la incertidumbre obtenida con un ensamble de pocos miembros. En este caso, se utiliza una RNA profunda para modelar la dispersión del ensamble generada con un ensamble grande utilizando un ensamble pequeño como entrada. Los resultados muestran una mejora significativa en la calidad de los pronósticos probabilísticos y, en particular, para los fenómenos meteorológicos extremos. Una de las desventajas de este enfoque es que la red modela la incertidumbre estimada por un ensamble de pronósticos, que puede estar sesgada, particularmente en presencia de errores de modelo.

Un enfoque diferente estima la incertidumbre del pronóstico directamente a partir de los datos sin requerir una estimación a priori de la misma como la que proporciona un ensamble de pronósticos. [D’Isanto and Polsterer \(2018\)](#); [Camporeale et al. \(2019\)](#); [Rasp and Lerch \(2018\)](#); [Veldkamp et al. \(2021\)](#); [Barnes and Barnes \(2021\)](#) y [Wang et al. \(2018\)](#) introdujeron arquitecturas de RNAs y metodologías de entrenamiento que cuantifican directamente la incertidumbre utilizando una gran base de datos de pronósticos y sus correspondientes observaciones. El aspecto clave es la definición de una función de costo sensible a la incertidumbre dependiente del estado del pronóstico. En [Wang et al. \(2018\)](#); [Barnes and Barnes \(2021\)](#) logran esto definiendo una función de costo inspirada en la verosimilitud de la observación, mientras que en [Camporeale and Carè \(2021\)](#), [D’Isanto and Polsterer \(2018\)](#), [Rasp and Lerch \(2018\)](#) y [Veldkamp et al. \(2021\)](#) definen una función de costo basada en las métricas CRPS, del inglés Continuous Ranked Probability Score (CRPS) y BS, del inglés Brier Score (BS) para la estimación de varianzas. Estos trabajos concluyen que es posible obtener una estimación poco costosa de la incertidumbre de los pronósticos, tanto para casos idealizados como para datos reales. Estos enfoques suponen la forma de la distribución de probabilidad del error del pronóstico (normalmente una distribución gaussiana), por lo que pueden considerarse enfoques paramétricos.

En [Scheuerer et al. \(2020\)](#); [Clare et al. \(2021\)](#) se propone un enfoque no paramétrico

para estimar la incertidumbre del pronóstico puramente a partir de los datos sin hacer suposiciones sobre la forma de la distribución de error del pronóstico subyacente. Así, discretizan el rango de valores de las variables pronosticadas en intervalos y calculan la probabilidad de que el valor pronosticado pertenezca a un intervalo dado, proporcionando una aproximación discreta de la función de densidad de probabilidad. En una línea similar, [Cannon \(2018\)](#); [Gasthaus et al. \(2019\)](#); [Bremnes \(2020\)](#) utilizan RNAs y regresión por cuantiles para cuantificar la incertidumbre del pronóstico sin asumir a priori la forma de su distribución.

La cuantificación de la incertidumbre en los pronósticos es un aspecto clave de los sistemas de DA. En particular, la mayoría de los métodos de DA se basan en una estimación precisa de la media de los pronósticos y de la matriz de covarianza de los errores. Juntas describen la función de densidad de probabilidad bajo el supuesto de que los errores son insesgados y gaussianos. Los enfoques de DA, como la interpolación óptima, en inglés Optimal Interpolation (OI) [Gandin 1965](#) o el método variacional tridimensional (3Dvar) [Parrish and Derber 1992](#), asumen que la matriz de covarianza del error de los pronósticos es independiente del estado del sistema.

En la actualidad, los métodos de DA que utilizan una matriz de covarianza dependiente del estado, ya sea implícita (p. ej., los variacional cuatridimensional (4Dvar), [Rabier et al. 2000](#)), o explícita (p. ej. filtro de Kalman por ensambles, en inglés Ensemble Kalman Filter (EnKF), [Houtekamer and Zhang 2016](#), o los filtros de partículas [van Leeuwen et al. 2019](#)) o métodos híbridos como en [Bannister \(2017\)](#), producen una notable mejora en la precisión de las condiciones iniciales y de la calidad de los pronósticos ([Kalnay, 2003](#); [Carrassi et al., 2018](#)). Sin embargo, estas mejoras se producen a expensas de un aumento significativo del coste computacional. Además, aun cuando la contribución de los errores en la condición inicial puede estar razonablemente bien representada, la contribución de los errores del modelo sigue siendo un desafío. Especialmente para los métodos que buscan representar la dependencia de la covarianza de los errores con el estado ([Tandeo et al., 2020](#)). Esto se debe, en parte, a que la forma en la que dichos errores surgen y se amplifican durante la integración de las ecuaciones es más difícil de representar, dado el desconocimiento que existe sobre la dinámica de los mismos.

El uso de técnicas de ML en el contexto de DA se ha debatido en varios trabajos. Las similitudes entre DA y ML y su potencial sinergia han sido introducidas en [Hsieh and Tang \(1998\)](#) y revisadas en [Cheng et al. \(2023\)](#). En [Bocquet et al. \(2019\)](#); [Brajard et al. \(2020\)](#); [Farchi et al. \(2021a, 2022\)](#) se ha propuesto un enfoque en el que se utiliza ML para la estimación de la dinámica del sistema y para representar los errores del modelo,

y DA para proporcionar una optimización continua *online* del modelo basado en datos. En la misma línea, [Brajar et al. \(2021\)](#); [Farchi et al. \(2021b\)](#), utilizan un enfoque de DA para entrenar una parametrización basada en ML del efecto de la dinámica de escala no resuelta dentro de un modelo numérico.

Otros enfoques buscaban la sustitución del sistema DA completo por una RNA, como en [Härter and de Campos Velho \(2008\)](#). En este enfoque los autores entrenan una RNA que aprende, a partir de un sistema DA dado, la magnitud y los patrones espaciales de la actualización del estado introducidos por las observaciones. [Buizza et al. \(2022\)](#) introdujo el nombre de *aprendizaje de datos* para describir varios ejemplos en los que ML y DA pueden combinarse para compensar mutuamente sus debilidades.

Relativamente pocos trabajos investigaron cómo la estimación de la covarianza de los errores del pronóstico basada en ML puede acoplarse con un sistema DA (p. ej., [Lguensat et al. 2017](#); [Ouala et al. 2018](#)). En particular, [Ouala et al. \(2018\)](#), acopló una estimación basada en RNAs de la covarianza del error del pronóstico con una actualización del análisis similar a la del filtro de Kalman en un espacio de estado de alta dimensión, y comparó los resultados con un enfoque de filtro de Kalman por ensamble y el enfoque basado en análogos de [Lguensat et al. \(2017\)](#) con resultados prometedores.

El objetivo general que nos planteamos fue investigar en qué medida se puede reconstruir la incertidumbre asociada a un estado determinado del sistema a partir del estado mismo y sin la necesidad de un ensamble. Es decir, si existe una relación que vincule la forma de la función de densidad de probabilidad, en inglés Probability Density Function (PDF), con el estado. En particular, si esta relación puede ser implementada mediante un algoritmo de ML y más específicamente mediante el entrenamiento de RNAs. Esto nos plantea también el interrogante de si los datos disponibles actualmente son suficientes para describir la relación existente entre el estado y la incertidumbre, tanto en términos de la calidad de información que proveen como en la cantidad necesaria.

Nuestra hipótesis es que mediante el uso de modelos de regresión basados en redes neuronales artificiales, junto con la información de un pronóstico determinístico, es posible obtener una estimación de la incertidumbre asociada a los pronósticos y de su dependencia con el estado del sistema. Asimismo, proponemos que la información producida por los sistemas de pronóstico del estado del arte es suficiente para entrenar el modelo de regresión y obtener resultados satisfactorios.

En este marco definimos los siguientes objetivos específicos:

- Proponer una metodología que permita entrenar una RNA capaz de estimar el error sistemático presente en un pronóstico determinístico.

- Proponer una metodología que permita entrenar una RNA capaz de estimar, a partir de un pronóstico determinístico, la varianza del error asociado al pronóstico.
- Diseñar y construir un conjunto de experimentos que permita analizar la calidad de la incerteza y compararla con otras metodologías del estado del arte.
- Extender la metodología propuesta para estimar matrices de covarianzas necesarias para la DA.
- Diseñar un sistema de DA que pueda hacer uso de las covarianzas estimadas con una RNA y evaluar su desempeño en el contexto de modelos caóticos simples.

El avance y la concreción de los objetivos específicos me permitió colaborar, interactuar y presentar los resultados parciales con otros miembros de la comunidad científica en las siguientes oportunidades:

- 2020 En el marco del proyecto de colaboración bilateral ECOS A17A08, se realizó una estadia de investigación en la Universidad IMT-Atlantique en la ciudad de Brest, Francia. Durante esta colaboración, se investigó junto al Dr. Pierre Tandeo y el Dr. Yicun Zhen las posibles ventajas de aplicar nuestra metodología de estimación de varianzas con redes neuronales en el espacio de estados transformados de las componentes principales.
- 2020 Los resultados obtenidos fueron presentados en formato póster en *Climate Informatics 2020 - 10th International Conference on Climate Informatics* con el título de **Quantification of forecast uncertainty using neural networks**.
- 2021 Los primeros resultados relativos a la estimación de covarianza se presentaron en *EGU General Assembly 2021* bajo el título **Machine learning-based uncertainty quantification for data assimilation: a simple model experiment**.
- 2021 El método de DA propuesto que utiliza la estimación de covarianza hecha por una RNA se presentó oralmente por primer vez en *International Symposium on Data Assimilation – Online* con el título **Forecast Uncertainty Estimation for Data Assimilation Using Neural Networks**.
- 2022 La metodología propuesta para entrenar redes neuronales que estimen la varianza del error de los pronósticos y el análisis comparativo con la técnica de ensambles fue publicado el 24 de noviembre del 2022 en la revista *Quarterly Journal of*

the Royal Meteorological Society con el título **Evaluation of Machine Learning Techniques for Forecast Uncertainty Quantification.**

- 2022 Los primeros resultados de la estimación de covarianza y la metodología propuesta de DA fue presentada en el taller *Machine learning and uncertainties in climate simulations* organizado por el instituto *Henri Lebesgue Center de Mathematiques* llevado a cabo entre el 6 y 9 de junio de 2022 en Moulin Mer, Logonna-Daoulas, Finistère, Francia.
- 2022 En una segunda estadía de investigación en el Instituto IMT-Atlantique en Brest, Francia en el marco del proyecto: A17A08 del programa ECOS-SUD a cargo de los Doctores Pierre Tandeo y Juan. J. Ruiz. Durante la estadía se realizaron experimentos para combinar la estimación de covarianza con la técnica de análogos, para el pronóstico del tiempo.
- 2022 Experimentos con datos reales meteorológicos (presión en superficie) de pronósticos probabilísticos fueron presentados en el *XIV Congreso Argentino de Meteorología - CONGREMET XIV* en modalidad oral con el título **Cuantificación de la incertidumbre en pronósticos numéricos usando redes neuronales**
- 2023 La metodología propuesta de estimación de covarianza y DA se encuentra en revisión en la revista *Quarterly Journal of the Royal Meteorological Society* con el título **Online machine-learning forecast uncertainty estimation for sequential data assimilation**
- 2023 Se expusieron los resultados de esta tesis en una presentación oral por el Dr. Juan J. Ruiz en: *10th International Congress on Industrial and Applied Mathematics, Tokyo, Japón*. Con el título: **Machine learning-based estimation of state-dependent forecast uncertainty.**
- 2023 Los resultados también fueron expuestos por Dr. Juan J. Ruiz como invitado a una charla en el *4th RIKEN Data Assimilation Workshop, Kobe, Japón*. Con el título: **Machine learning-based estimation of state-dependent forecast uncertainty: application to data assimilation.**
- 2023 Los conocimientos y la experiencia adquiridos en el tema durante mi trabajo de investigación doctoral fue compartido con la comunidad académica en dos clases-taller dictadas en el Curso Intensivo: *Aplicaciones de Machine Learning en Geofluidos* dictado del 27 de febrero al 10 de marzo de 2023 en el Departamento de

Ciencias de la Atmósfera y los Océanos - Facultad de Ciencias Exactas y Naturales
- Universidad de Buenos Aires

El resto de la tesis está organizado de la siguiente manera: en el capítulo 2 se introduce la teoría los tres ejes temáticos principales (modelado numérico, DA y RNA). En el capítulo 3 se estudian antecedentes y distintas técnicas de estimación de la incertidumbre (varianzas) sentando las bases de la experimentación que será llevada a cabo en el capítulo 4. El análisis de cómo extender la estimación de varianzas a covarianzas se realiza en el capítulo 5, donde se realiza una reseña y análisis de trabajos previos y se define la metodología que se usará en la experimentación del capítulo 6, donde se pondrá a prueba la calidad de las estimaciones en un contexto de DA. Finalmente, el capítulo 7 presenta las conclusiones generales y explora las líneas abiertas de investigación y posibles formas de continuar el trabajo.

Capítulo 2

Marco teórico

En este capítulo se presentará una introducción a los principales ejes temáticos de esta tesis. Primero se introduce el estudio de los sistemas dinámicos desde un contexto formal. En la sección 2.2 se presentarán los conceptos básicos de la asimilación de datos, en inglés Data Assimilation (DA), y finalmente en la sección 2.3 se explicarán las ideas fundamentales, su formulación y el entrenamiento de las redes neuronales artificiales (RNAs) que serán utilizadas a lo largo del trabajo.

2.1. Modelos dinámicos estocásticos

La evolución de un sistema dinámico, como la atmósfera o el océano, se representa en este trabajo mediante un proceso de Markov

$$\mathbf{x}_k = \mathcal{M}_{k:k-1}(\mathbf{x}_{k-1}) + \boldsymbol{\eta}_k(\mathbf{x}_k), \quad (2.1)$$

donde $\mathbf{x}_k$ es un vector N_x -dimensional que representa el estado del sistema en el tiempo k , $\mathcal{M}_{k:k-1}$ es un modelo subrogado de la dinámica del sistema que asumimos conocido, por ejemplo, un modelo numérico de predicción meteorológica o un modelo de simulación climática. Este evoluciona el estado del sistema desde el tiempo $k - 1$ hasta el tiempo k . Se supone que este modelo es no lineal y que captura, al menos parcialmente, el comportamiento caótico del sistema. La discrepancia entre $\mathbf{x}_k$ y $\mathcal{M}_{k:k-1}(\mathbf{x}_{k-1})$ debido a la imperfección del modelo está representada por $\boldsymbol{\eta}_k(\mathbf{x}_k)$ que es una realización de una distribución de probabilidad desconocida condicionada en $\mathbf{x}_k$ (los parámetros de la

distribución suelen ser dependientes del estado del sistema). La media de esta distribución corresponde al error sistemático del modelo, y al no ser necesariamente igual cero, introduce un sesgo en el pronóstico. La evolución del sistema en un período de tiempo arbitrario se puede aproximar mediante la aplicación recursiva del modelo a lo largo de varios pasos de tiempo. En este caso, los errores del modelo interactúan con la dinámica caótica del sistema:

$$\mathbf{x}_{k+l} = \underbrace{\mathcal{M}(\mathcal{M}(\cdots(\mathcal{M}(\mathbf{x}_k) + \boldsymbol{\eta}_{k+1}) + \boldsymbol{\eta}_{k+2}) + \cdots)}_{l-\text{veces}} + \boldsymbol{\eta}_{k+l}(\mathbf{x}_k). \quad (2.2)$$

Vamos a suponer que estas interacciones se pueden representar con un término de error aditivo dependiente del estado

$$\mathbf{x}_{k+l} = \mathcal{M}_{k+l:k}(\mathbf{x}_k) + \boldsymbol{\delta}_{k+l}(\mathbf{x}_k, \boldsymbol{\eta}_{k+1}, \dots, \boldsymbol{\eta}_{k+l}), \quad (2.3)$$

donde $\boldsymbol{\delta}_{k+l}$ representa el efecto acumulado de los errores de modelo y su evolución bajo la dinámica no lineal del sistema entre los tiempos k y $k+l$ y $\mathcal{M}_{k+l:k}$ es el operador no lineal resultante de aplicar, en forma recursiva, el modelo subrogado l veces. Para el pronóstico numérico del tiempo, se suele obtener un pronóstico determinístico ($\mathbf{x}^d$) integrando el modelo dinámico a partir de la condición inicial más probable que suele denominarse análisis ($\mathbf{x}^a$), tal que

$$\mathbf{x}_{l,k+l}^d = \mathcal{M}_{k+l:k}(\mathbf{x}_k^a), \quad (2.4)$$

donde $\mathbf{x}_{l,k+l}^d$ es el estado pronosticado para el plazo l (primer subíndice) válido en el tiempo $k+l$ (segundo subíndice) y $\mathbf{x}_k^a$ es el análisis en el tiempo k (es decir, una estimación puntual del estado del sistema en el tiempo k dadas todas las observaciones disponibles hasta el tiempo k). Esta estimación suele obtenerse mediante un proceso de DA (por ejemplo [Kalnay, 2003](#); [Carrassi et al., 2018](#)).

El error de pronóstico resultante se define como

$$\boldsymbol{\epsilon}_{l,k+l}^d = \mathbf{x}_{l,k+l}^d - \mathbf{x}_{k+l}^t, \quad (2.5)$$

donde $\mathbf{x}_{k+l}^t$ es el estado real del sistema que suele denominarse estado verdadero y lo asumiremos desconocido. El error de pronóstico se debe a la presencia de errores en las condiciones iniciales (errores presentes en el análisis $\mathbf{x}_k^a$) y errores en la formulación del modelo ($\boldsymbol{\delta}_{k+l}$). Ambos errores son propagados a través de la dinámica del sistema como

lo indica la ecuación 2.2. Este efecto depende del estado, lo que significa que la tasa de crecimiento del error depende del estado del sistema, lo cual constituye un aspecto particularmente desafiante del pronóstico de corto y mediano plazo de los fenómenos atmosféricos. El error es una muestra de una distribución desconocida, por lo que proponemos cuantificarlo estimando sus dos primeros momentos, su media ($\bar{\epsilon}_{l,k+l}$) y su covarianza ($\mathbf{Q}_{l,k+l}$) dependientes del estado del sistema. Para simplificar la notación, durante el resto de esta sección, eliminamos el índice de tiempo (segundo subíndice) siempre que sea posible y sólo mantenemos el subíndice del plazo de pronóstico, es decir, $\mathbf{x}_l^d = \mathbf{x}_{l,k+l}^d$. En la actualidad, la forma más común de estimar la media y la covarianza del error del pronóstico y su dependencia del estado es mediante el pronóstico por ensambles. En este enfoque, el estado inicial se supone incierto y, por tanto, viene dado por una distribución de probabilidad. Esta distribución inicial se representa mediante una muestra, es decir, un conjunto de estados iniciales. A continuación, se realiza un ensamble de integraciones del modelo inicializado a partir de las diferentes condiciones iniciales, es decir, el n -ésimo miembro del ensamble $\mathbf{x}^{f,(n)}$ se genera integrando el modelo a partir de la n -ésima condición inicial, $\mathbf{x}^{a,(n)}$:

$$\mathbf{x}_l^{f,(n)} = \mathcal{M}_{l;0}^{(n)}(\mathbf{x}^{a,(n)}) + \hat{\delta}_{l;0}^{(n)}, \quad (2.6)$$

donde $\mathbf{x}_l^{f,(n)}$ es el vector de estado pronosticado dado por el n -ésimo miembro del ensamble en el plazo l , $\hat{\delta}_{l;0}^{(n)}$ es una representación del efecto de los errores del modelo, y $\mathcal{M}_{l;0}^{(n)}$ es el modelo numérico. En el caso de los modelos dinámicos no lineales caóticos, estas integraciones se dispersan entre sí en algunas direcciones del espacio de fases. La magnitud de la dispersión se cuantifica, típicamente, a través de la desviación estándar de los estados pronosticados en un tiempo determinado. Es importante notar que puede utilizarse un modelo numérico diferente para la integración de cada miembro del ensamble para representar el efecto de los errores de modelo. Técnicas como la perturbación estocástica de los parámetros (Christensen et al., 2015) o los ensambles multi-modelo (Hou et al., 2001) se basan en esta idea.

2.2. Asimilación de datos

2.2.1. Asimilación de datos secuencial

En un ciclo de DA secuencial, se busca estimar el estado de un sistema dinámico a intervalos regulares de tiempo, combinando la información proporcionada por un modelo

numérico subrogado y un conjunto de observaciones parciales y afectadas por ruido (Carrassi et al., 2018).

Consideremos un sistema dinámico caótico representado mediante un proceso de Markov como en la Ec. 2.1 con un error del pronóstico definido en la Ec. 2.5. Los métodos de DA se basan en la suposición de que este error es gaussiano y con media cero, lo que generalmente no suele ser el caso.

El estado del sistema se relaciona con las magnitudes observables mediante la ecuación de observación,

$$\mathbf{y}_k = \mathcal{H}(\mathbf{x}_k^t) + \boldsymbol{\nu}_k, \quad (2.7)$$

donde $\mathbf{y}_k$ es el vector N_y -dimensional que contiene las cantidades observables, $\mathcal{H}$ es el operador de observación, es decir la función que proyecta las variables de estado en el espacio de observación y $\boldsymbol{\nu}_k$ es el error de observación que se supone extraído de una distribución gaussiana con media cero y covarianza conocida denotada $\mathbf{R}_k$. Dado una estimación a priori del estado del sistema, típicamente un pronóstico realizado utilizando el modelo numérico, $(\mathbf{x}_k^f)$, un conjunto de observaciones $(\mathbf{y}_k)$ y suponiendo que sus errores son insesgados, el estimador lineal que minimiza el error cuadrático medio con respecto al estado verdadero del sistema viene dado por:

$$\mathbf{x}_k^a = \mathbf{x}_k^f + \mathbf{K}(\mathbf{y}_k - \mathcal{H}(\mathbf{x}_k^f)), \quad (2.8a)$$

$$\mathbf{K} = \mathbf{P}_k^f \mathbf{H}^\top (\mathbf{H} \mathbf{P}_k^f \mathbf{H}^\top + \mathbf{R}_k)^{-1}, \quad (2.8b)$$

donde $\mathbf{x}_k^a$ es la estimación del estado del sistema en el tiempo k , $\mathbf{H}$ es la aproximación lineal tangente del operador de observación y $\mathbf{K}$ es la matriz de ganancia de Kalman que proyecta la discrepancia entre las observaciones y los valores observables pronosticados, ponderando en base a la incertidumbre asociada de cada uno. Esta estimación es también la estimación de máxima verosimilitud del estado del sistema bajo el supuesto de que la función de densidad de probabilidad, en inglés Probability Density Function (PDF), de los errores del pronóstico y la de los errores de observación sean ambas gaussianas con media cero (Carrassi et al., 2018).

El panel izquierdo de la Fig. 2.1 esquematiza tres ciclos de DA donde se representa la evolución temporal real del estado de un sistema dinámico (línea azul) y su simulación mediante un modelo numérico (línea verde), también se muestra cómo se combina el pronóstico $\mathbf{x}^f$ (círculo verde) y la observación $\mathbf{y}$ (estrella azul) junto a sus respectivas incertidumbres $\mathbf{P}^f$ y $\mathbf{R}$ (sombreados) para obtener un análisis $\mathbf{x}^a$ (cuadrado rojo) y su

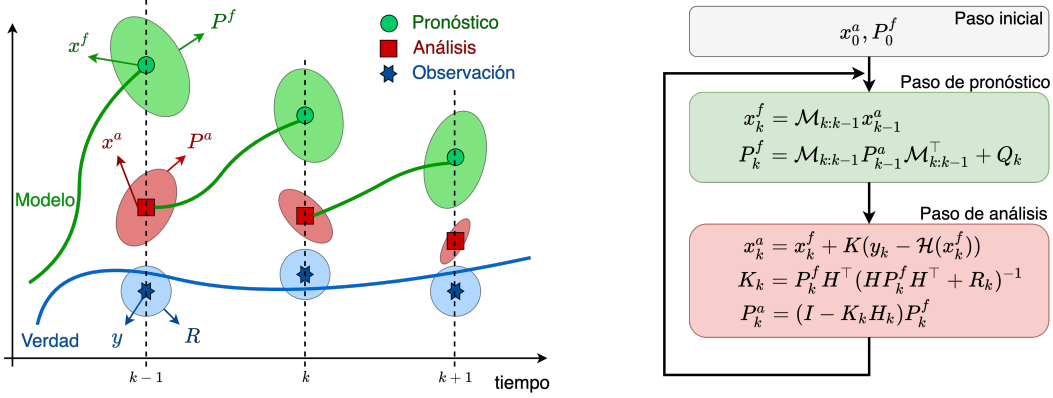


Figura 2.1: Esquema del ciclo de asimilación de datos secuencial Carrassi et al. (2018).

incertidumbre asociada $\mathbf{P}^a$. El algoritmo secuencial que genera los análisis se muestra en la parte derecha de la Fig. 2.1 y consiste en tres etapas fundamentales:

Paso inicial Debemos comenzar el proceso con una condición inicial que describa el estado del sistema ($\mathbf{x}_0^a$) y la incertidumbre asociada ($\mathbf{P}_0^f$). Si al comienzo del ciclo se desconoce la incertidumbre del estado inicial se puede aproximar mediante una covarianza climatológica.

Paso pronóstico Se utiliza el modelo dinámico $\mathcal{M}$ para generar un pronóstico $\mathbf{x}_1^f$ a partir de la condición inicial $\mathbf{x}_0^a$. Este pronóstico suele denominarse *first guess* o *background*. También se requiere estimar la incertidumbre de dicho pronóstico. Este es un punto clave de la metodología, y cuando la dimensión del estado es grande y su dinámica es no lineal, estimar $\mathbf{P}^f$ puede ser un gran desafío. Más adelante se discuten distintas técnicas de estimación.

Paso análisis En este paso se calcula la ganancia de Kalman K que pondera la incertidumbre asociada a la observación ($\mathbf{R}$) y al pronóstico ($\mathbf{P}^f$). La ganancia de Kalman se usa para determinar, según el nivel de confianza, cuánto será el aporte del pronóstico y cuánto el de la observación para la construcción del análisis. También se utiliza para estimar la incertidumbre asociada al análisis combinando la del pronóstico y la de la observación.

Una vez obtenida una estimación del estado del sistema en el momento k , el modelo numérico (Ec. 2.4) puede utilizarse para predecir el estado del sistema al tiempo $k+1$, y el ciclo puede repetirse cada vez que se disponga de un nuevo conjunto de observaciones.

La exactitud de la estimación del estado depende en gran medida de la exactitud de las matrices de covarianza de los errores $\mathbf{P}_k^f$ y $\mathbf{R}_k$, cuya determinación es posiblemente uno de los aspectos más desafiantes de los sistemas de asimilación (Tandeo et al., 2020).

La forma en la que se estima la covarianza de los errores difiere entre los diferentes métodos de asimilación secuenciales. En métodos como la interpolación óptima, en inglés Optimal Interpolation (OI), y los métodos variacional tridimensional (3Dvar), $\mathbf{P}_k^f$ se asume fija, mientras que en el filtro de Kalman, el filtro extendido y el filtro de Kalman por ensambles, en inglés Ensemble Kalman Filter (EnKF), se intenta capturar la dependencia de dicha covarianza con el estado del sistema. Por otra parte, los métodos híbridos se basan en la suposición de que la matriz de covarianza de los errores puede ser descripta como la suma de una covarianza independiente del estado y una covarianza dependiente del estado. Esto busca combinar los aspectos más ventajosos de ambos métodos. Uno de los objetivos específicos de este trabajo de tesis es lograr una estimación de la incertidumbre asociada al pronóstico ($\mathbf{P}^f$) que dependa del estado del sistema usando red neuronal artificial (RNA) para aplicaciones en DA.

2.2.2. Filtro de Kalman por ensambles

El EnKF es uno de los métodos más utilizados para aproximar $\mathbf{P}_k^f$ y su dependencia con el estado del sistema en aplicaciones de DA. En esta tesis, el EnKF se utiliza para generar la base de datos para el entrenamiento de los métodos basados en aprendizaje automático, en inglés Machine Learning (ML), y también se utiliza como base de referencia para la evaluación de los mismos. A continuación describimos brevemente esta técnica.

Dadas las muestras $\mathbf{x}_{k-1}^{a,(n)}$ ($n \in 1 \dots N_e$ donde N_e el tamaño del ensamble) extraídas de la distribución de probabilidad del estado del sistema al tiempo $k - 1$, el estado del sistema en el tiempo k puede estimarse evolucionando los miembros individuales del ensamble desde el tiempo $k - 1$ al tiempo k mediante las ecuaciones del modelo no lineal:

$$\mathbf{x}_k^{f,(n)} = \mathcal{M}_{k:k-1}^{(n)} \left(\mathbf{x}_{k-1}^{a,(n)} \right) + \hat{\boldsymbol{\eta}}_k^{(n)}, \quad (2.9)$$

donde $\mathbf{x}_k^{f,(n)}$ son los miembros evolucionados del ensamble. En este punto es importante notar que algunas implementaciones de EnKF asumen $\hat{\boldsymbol{\eta}}_k = 0$ y el error de modelo se representa en forma multiplicativa en lugar de aditiva.

Ahora, podemos usar estas evoluciones para proporcionar una estimación puntual del

estado calculando la media del ensamble en el tiempo k ,

$$\bar{\mathbf{x}}_k^f = \frac{1}{N_e} \sum_{n=1}^{N_e} \mathbf{x}_k^{f,(n)}. \quad (2.10)$$

En esta línea, la covarianza de los errores del pronóstico puede estimarse a partir del muestreo del estado pronosticado,

$$\hat{\mathbf{P}}_k^f = \frac{1}{(N_e - 1)} \sum_{n=1}^{N_e} \left(\mathbf{x}_k^{f,(n)} - \bar{\mathbf{x}}_k^f \right) \left(\mathbf{x}_k^{f,(n)} - \bar{\mathbf{x}}_k^f \right)^\top. \quad (2.11)$$

En este trabajo de tesis se utilizaron dos implementaciones del filtro de Kalman, del inglés Kalman Filter (KF), una versión determinística (LETKF [Hunt et al. \(2007\)](#)) y una versión estocástica del EnKF ([Burgers et al., 1998](#)). En la implementación determinística, todos los miembros se actualizan utilizando la observación del estado $\mathbf{y}_k$, mientras que en la implementación estocástica, cada miembro del ensamble se actualiza independientemente usando la Ecuación 2.8a, en la que $\mathbf{y}_k$ se sustituye por $\mathbf{y}_k^{(n)} = \mathbf{y}_k + \nu_k^{(n)}$, con $\nu_k^{(n)} \sim \mathcal{N}(\mathbf{0}, \mathbf{R}_k)$ y con $\mathbf{P}_k^f$ dado por la Ec. 2.11. Luego, para ambas implementaciones, se puede aplicar el algoritmo de la figura 2.1 donde para cada miembro del ensamble se construye un análisis, y el análisis medio se considera la mejor aproximación al estado verdadero.

Uno de los desafíos más importantes en estas técnicas es la corrección del sesgo en los pronósticos. Los ensambles pueden verse afectados por sesgos sistemáticos, es decir, tendencias persistentes en los errores de pronóstico que no se deben al azar. Para abordar este problema, se utilizan técnicas de inflación y localización.

La inflación es un proceso mediante el cual se ajustan las desviaciones estándar de las perturbaciones en el ensamble. Esto se hace para corregir el sesgo en los pronósticos y garantizar que las perturbaciones reflejen adecuadamente la verdadera variabilidad atmosférica. La inflación se basa en la idea de que si las perturbaciones son demasiado pequeñas, el ensamble tiende a subestimar la incertidumbre, mientras que si son demasiado grandes, sobreestima la incertidumbre. Ajustar las desviaciones estándar de las perturbaciones de manera apropiada es esencial para obtener un ensamble bien calibrado.

La localización, por otro lado, se refiere a la influencia que una observación tiene en la actualización de un pronóstico en el análisis de datos. En la asimilación de datos, es importante determinar cuánto peso se debe dar a una observación en función de su

ubicación y de la distancia entre la ubicación de la observación y la ubicación del punto de pronóstico. La localización ayuda a evitar que las observaciones lejanas tengan un impacto excesivo en el pronóstico y contribuyan a errores no deseados.

En los sistemas geofísicos, la covarianza entre variables correspondientes a puntos alejados en el espacio físico es prácticamente nula. En la metodología EnKF, debido a la presencia de errores de muestreo, las covarianzas entre variables distantes pueden ser significativamente distintas de 0, sobre todo cuando se utiliza un ensamble pequeño. En este caso, se puede utilizar un método de localización de covarianzas para amortiguar la magnitud de las covarianzas espurias. Estos métodos suelen multiplicar las covarianzas estimadas por un factor que disminuye con la distancia física entre las dos variables (Hamill et al., 2001).

La inflación y la localización son componentes esenciales en los métodos de ensamble en meteorología. Ayudan a corregir el sesgo en los pronósticos y a garantizar que los ensambles sean una herramienta confiable para estimar la incertidumbre en los pronósticos meteorológicos. En esta trabajo de tesis, los valores óptimos de estos parámetros se encontraron mediante una búsqueda exhaustiva en el espacio de los parámetros.

2.3. Redes neuronales artificiales

Las RNA surgen como un intento de encontrar una representación matemática de los sistemas de procesamiento de información biológicos (McCulloch and Pitts, 1943; Widrow and Hoff, 1960; Rosenblatt, 1962; Rumelhart et al., 1986). Desde entonces fueron evolucionando, y nuevas variantes y arquitecturas se desarrollaron para resolver problemas de clasificación y regresión prácticamente en todas las áreas de la ciencia. Pero, en esencia, siguen siendo un conjunto de unidades (neuronas) interconectadas entre sí para transmitir información.

En el panel izquierdo de la figura 2.2 se muestra el diagrama de una neurona, que es el elemento constituyente de las RNA. Una neurona k está conectada a muchas señales de entrada $x_{j \in \{1, \dots, m\}}$ a través de los pesos sinápticos $w_{k,j}$ que ponderan la suma del aporte de cada señal de entrada a la excitación general de la neurona. Es importante notar que los pesos sinápticos son los parámetros que serán ajustados durante la etapa de entrenamiento, es decir, la parte de la neurona que codifica el aprendizaje. La excitación de la neurona puede representarse como una componente $u_k = \sum_{j=1}^m w_{k,j} \cdot x_j$ que es una transformación lineal de la entrada, más una entrada adicional de desplazamiento o bias resultando en una transformación afín ($u_k + b_k$) que permite desplazar la salida del

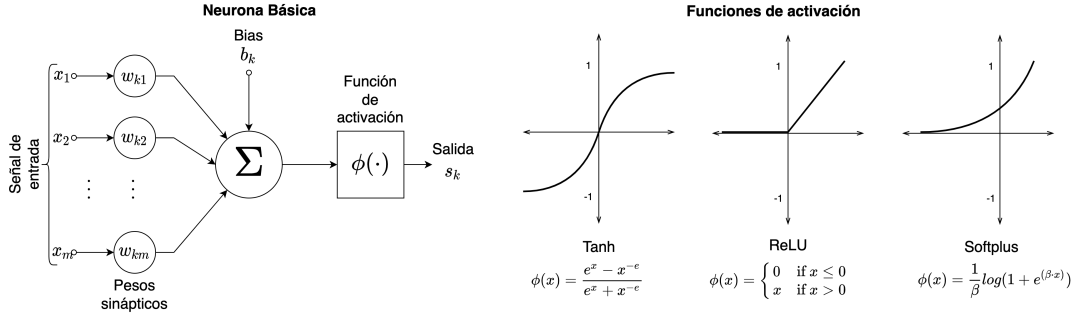


Figura 2.2: Esquema de una neurona de McCulloch & Pitts (izquierda) y las distintas funciones de activación $\phi(\cdot)$ utilizadas en este trabajo (derecha)

sumador u_k del origen de coordenadas. El valor de activación s_k (la salida de la neurona) está determinado por una función de activación $\phi(\cdot)$ sensible a la excitación de la neurona $s_k = \phi(u_k + b_k)$. Las funciones de activación sirven para limitar la amplitud de la salida y para incorporar un componente no lineal en el comportamiento de la neurona. Elegir la función de activación es una decisión de diseño de la que depende, en la práctica, la velocidad y la convergencia a los valores óptimos de los pesos sinápticos durante el entrenamiento. La parte derecha de la figura 2.2 muestra tres ejemplos de funciones de activación (Tangente hiperbólica, rectificador lineal y softplus) que fueron usadas en la experimentación a lo largo de este trabajo.

Dependiendo cómo se interconecten estas neuronas puede obtenerse una gran variedad de arquitecturas de RNA, en particular en este trabajo se utilizaron dos configuraciones que describiremos a continuación.

2.3.1. Redes completamente conectadas

En esta arquitectura, las RNA están organizadas como una sucesión de capas (conjunto de neuronas) en donde la salida de cada capa es la entrada de la capa siguiente (las conexiones son direccionales) y las neuronas pertenecientes a una misma capa no están conectadas entre sí. Se llaman redes completamente conectadas (*fully-connected*) o Perceptron multi-capas (MLP) porque cada neurona de una capa está conectada a todas las neuronas de la capa siguiente. Pero sólo se permiten conexiones entre capas vecinas (las conexiones no se saltan capas) y además se prohíben las conexiones dirigidas hacia capas anteriores (arquitectura *feed-forward*).

En la Fig. 2.2 se muestra un ejemplo de una red MPL con dos capas y se muestra la

notación que utilizaremos para describirlas. Las neuronas de salidas están identificadas con $s_{i \in \{1,2\}}$ y las neuronas de la capa oculta son $V_{k \in \{1,2,\dots,k\}}$, $x_{j \in \{1,2,\dots,j\}}$ son los datos de entrada de la red. Los pesos sinápticos $w_{k,j}^{(1)}$ conectan las entradas con las neuronas de la primera capa, mientras que $w_{i,k}^{(2)}$ vincula la capa oculta con las salidas de la red.

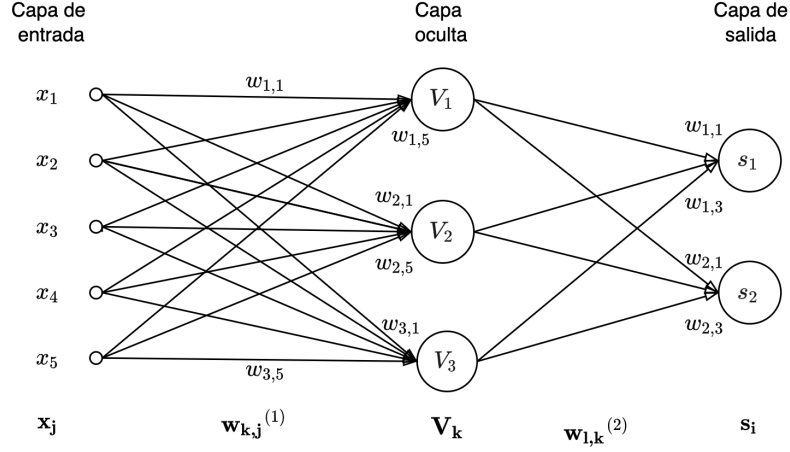


Figura 2.3: Esquema de una red multi-capas completamente conectada, donde las únicas conexiones son entre una capa y la siguiente, y todas las salidas de las neuronas de una capa están conectadas a las entradas de las neuronas en la capa siguiente.

Dado el patrón de entrada $x_{j \in \{1,2,\dots,j\}}$, las neuronas de la primera (y única en este caso) capa oculta reciben como entrada

$$h_k^{(1)} = \sum_j w_{k,j}^{(1)} \cdot x_j \quad (2.12)$$

y generan a la salida

$$V_k = \phi \left(h_k^{(1)} + b_k^{(1)} \right) = \phi \left(\sum_j w_{k,j}^{(1)} \cdot x_j + b_k^{(1)} \right) \quad (2.13)$$

que se propaga a la capa de salida como

$$h_i^{(2)} = \sum_k w_{i,k}^{(2)} \cdot V_k = \sum_k w_{i,k}^{(2)} \cdot \phi \left(\sum_j w_{k,j}^{(1)} \cdot x_j + b_k^{(1)} \right) \quad (2.14)$$

que produce a la salida de la red

$$\begin{aligned}
 s_i &= \phi \left(h_i^{(2)} + b_i^{(2)} \right) \\
 &= \phi \left(\sum_k w_{i,k}^{(2)} \cdot V_k + b_i^{(2)} \right) \\
 &= \phi \left(\sum_k w_{i,k}^{(2)} \cdot \phi \left(\sum_j w_{k,j}^{(1)} \cdot x_j + b_k^{(1)} \right) + b_i^{(2)} \right)
 \end{aligned} \tag{2.15}$$

Este procedimiento de calcular la entrada de una capa usando la salida de la capa anterior se puede repetir tantas veces como capas tenga la red, permitiéndonos calcular los valores de salida de la red a partir de un patrón de entrada dado.

2.3.2. Entrenamiento de redes neuronales

Todas las RNA utilizadas en este trabajo fueron entrenadas de manera supervisada, esto es, se dispone de un conjunto de entrenamiento μ que consiste en muestras formadas por pares (x^μ, y^μ) donde x^μ es el valor de entrada a la red y y^μ es la salida esperada. Durante el entrenamiento, se busca en general que la salida de la red s^μ correspondiente a la entrada x^μ se parezca a y^μ . Es decir, buscamos minimizar una función de similitud o distancia $\mathcal{L}$ entre y^μ y s^μ . A esta función se le suele llamar también función de costo o de pérdida y es fundamental para determinar la convergencia del entrenamiento de la RNA. Entonces, entrenar una RNA significa encontrar el valor óptimo de los pesos sinápticos $w_{i,j}$ que minimizan la función de costo $\mathcal{L}$ para todos los pares μ del conjunto de datos de entrenamiento.

Gradiente descendente y retropropagación (backpropagation)

El gradiente descendente es un método que se utiliza para encontrar un mínimo de una función. Consiste en calcular el gradiente de la función en un punto, e ir moviéndose en la dirección de descenso del gradiente hasta que éste se haga cero. Así como está planteado, este método no garantiza alcanzar el mínimo global de la función, debido a que la condición de que el gradiente se haga cero se puede cumplir para un mínimo local. No obstante, en la práctica existen técnicas que nos ayudan, en algunos casos, a reponernos de esta situación como se verá más adelante.

El algoritmo de retropropagación es una metodología para calcular el gradiente en una RNA, comienza con la derivada de la función de costo y va hacia atrás, utilizando la regla de la cadena de las derivadas, hasta llegar a todos los pesos sinápticos de los cuales depende la predicción. Esto nos permite aplicar el gradiente descendente para definir

una regla de ajuste de los pesos de la red (entrenamiento) basándose en la evaluación de todo el conjunto de datos (a esto se lo denomina una **época**). Así, el entrenamiento es simplemente repetir este procedimiento por varias épocas hasta alcanzar la convergencia a cero del gradiente a lo que se espera sea el mínimo global.

Supongamos entonces que usamos el error cuadrático definido sobre la salida de la RNA como función de costo:

$$\mathcal{L} = \frac{1}{2} \sum_{\mu, i} (y_i^\mu - s_i^\mu)^2 \quad (2.16)$$

reemplazando s_i^μ por la Eq. 2.15:

$$\mathcal{L} = \frac{1}{2} \sum_{\mu, i} \left[y_i^\mu - \phi \left(\sum_k w_{i,k}^{(2)} \cdot V_k + b_i^{(2)} \right) \right]^2 \quad (2.17)$$

Así, podemos utilizar la regla de la cadena para derivar respecto de cada peso sináptico, empezando por los pesos de la capa de salida hasta llegar a los de la primera capa. Por ejemplo, si consideramos los pesos de la última capa:

$$\begin{aligned} \Delta w_{i,k}^{(2)} = \frac{\partial \mathcal{L}}{\partial w_{i,k}^{(2)}} &= \sum_{\mu, i} (y_i^\mu - s_i^\mu) \phi' \left(h_i^{(2), \mu} \right) V_k^\mu \\ &= \sum_{\mu, i} \delta_i^\mu V_k^\mu \end{aligned} \quad (2.18)$$

donde definimos

$$\delta_i^\mu = \phi' \left(h_i^{(2), \mu} \right) (y_i^\mu - s_i^\mu). \quad (2.19)$$

Para obtener el gradiente respecto de los pesos entre la entrada y la capa oculta se debe diferenciar respecto de $w_{k,j}^{(1)}$, que está más profundo en la Eq. 2.15.

Ahora, para actualizar los pesos podemos usar el gradiente descendente escalado por un valor η_{lr} , que llamaremos tasa de aprendizaje, que determina la proporción con que los pesos de la red son actualizados y controla la velocidad a la que los pesos se adaptan al problema (convergencia).

$$w_{p,q}^{\text{nuevo}} = w_{p,q}^{\text{viejo}} - \eta_{lr} \Delta w_{p,q} \quad (2.20)$$

El valor de η_{lr} es quizás uno de los hiperparámetros de entrenamiento más importantes

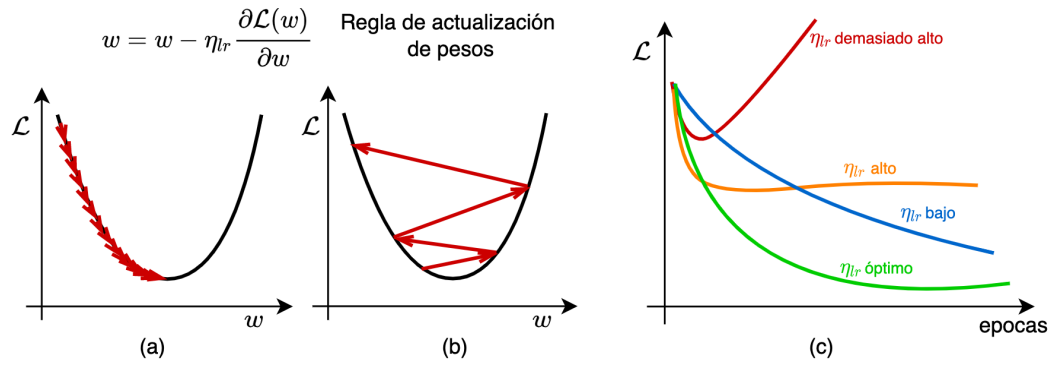


Figura 2.4: Los paneles (a) y (b) muestran el efecto del entrenamiento según el valor del parámetro η_{lr} sea muy pequeño o muy grande respectivamente. El panel (c) muestra la evolución del valor de la función de costo a lo largo del entrenamiento (épocas) para distintos valores de η_{lr} . Las imágenes (a,b) están basadas en [Andrew Ng](#), MOOC y la (c) en [CS231n](#), MOOC

([Goodfellow et al., 2016](#), pag. 429), valores muy bajos requieren mucho tiempo innecesario de convergencia a un mínimo que podría evitarse (Fig. 2.4a y 2.4c línea azul) mientras que los η_{lr} muy altos pueden hacer que los pesos converjan muy rápidamente a una solución subóptima (Fig. 2.4c línea naranja) o hasta ocasionar la divergencia del método (Fig 2.4b y 2.4c línea roja). Entonces, como se puede ver en la figura 2.4, el método no garantiza la convergencia del algoritmo de minimización ni siquiera a un mínimo local, es por eso que el parámetro η_{lr} debe ser cuidadosamente seleccionado. Pero aun si pudiéramos encontrar un η_{lr} óptimo, la metodología tampoco nos garantiza que logremos alcanzar un mínimo global, y lo más común es que el método se estanque en un mínimo local. La presencia de mínimos locales es común en las funciones de costo asociadas al entrenamiento de las RNA debido a la relación no lineal que existe entre la entrada y la salida de la red dada por las funciones de activación.

Una manera de mitigar este problema es extendiendo el método del gradiente descendente con una técnica llamada momento (o momentum) que consiste en añadir historia a la ecuación de actualización de pesos utilizando actualizaciones previas. La figura 2.5a ilustra esta técnica donde se combina la dirección e intensidad del momento (el gradiente del paso anterior escalado) con el gradiente del paso actual. El efecto de esta técnica asemeja a incorporar una componente de inercia al descenso por gradiente, como se ilustra en la figura 2.5b. Entonces, el momento es el hiperparámetro que determina qué porcentaje de la historia del gradiente se incorpora al gradiente actual.

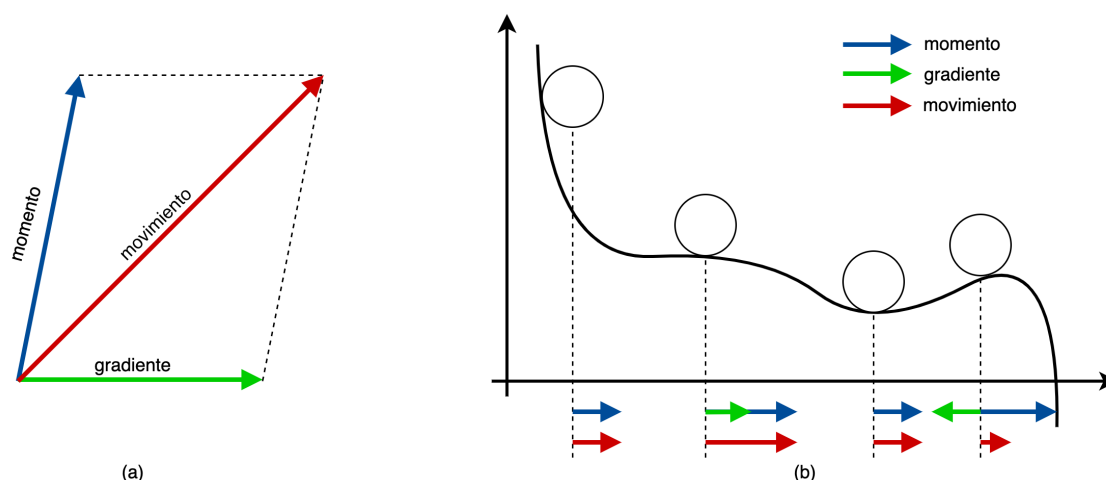


Figura 2.5: (a) Dirección efectiva de descenso (movimiento) basada en el gradiente y el momento, imagen basada en [CS231n](#), MOOC. (b) Interpretación del momento como inercia, imagen basada en [James Le](#), MOOC

En resumen, el algoritmo de retropropagación nos permite modificar los pesos de la RNA de manera tal que, con cada ajuste, se vaya convergiendo a un mínimo local de la función de costo $\mathcal{L}$ (entrenamiento). Es decir, la función de costo es lo que determina qué se va a aprender de los datos durante el entrenamiento. En los capítulos 3 y 5 se desarrollará este tema con mayor profundidad.

Entrenamiento y conjunto de datos

Como vimos, el conjunto de entrenamiento de RNA es simplemente un conjunto de pares (*entrada, objetivo*), donde el valor *entrada* es el dato que recibe la red en su entrada y el valor *objetivo* (en inglés *target*) es un valor conocido que se considera correcto y que se espera que la red aprenda a asociar al valor *entrada* durante el entrenamiento.

Entonces, cuando el método del gradiente descendente utiliza el gradiente medio entre todas las muestras del conjunto de entrenamiento, se lo denomina entrenamiento determinístico o por lotes (en inglés *batch*). Un aspecto característico de las técnicas de ML es que las funciones de costo generalmente pueden descomponerse como una suma sobre cada muestra del conjunto de entrenamiento. Esto permite calcular una actualización de los pesos sinápticos basándose en una estimación del valor esperado del gradiente usando sólo un subconjunto de muestras (entrenamiento por mini-lotes o mini-batches). Cuando el tamaño del mini-lote es igual a un elemento, entonces se lo denomina en-

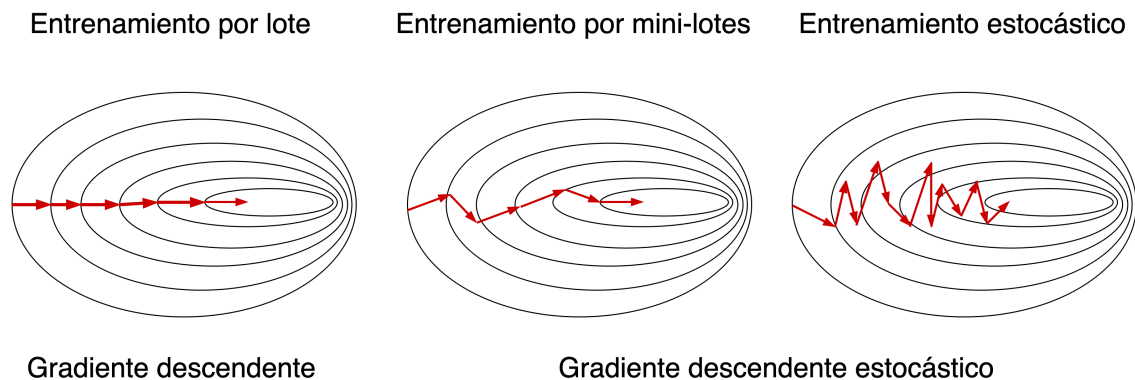


Figura 2.6: Efecto del tamaño del lote en la convergencia del algoritmo de retropropagación. Imagen adaptada de [Andrew Ng](#), MOOC

trenamiento estocástico. El uso de mini-lotes supone una convergencia más rápida (en épocas) comparado con el entrenamiento determinista porque éste realiza una única actualización de pesos por época mientras que el entrenamiento por mini-lotes actualiza los pesos usando una estimación del gradiente basada en procesar un mini-lote. En general, a esta técnica se la conoce como gradiente descendente estocástico y la idea de introducir *estocasticidad* en la actualización de pesos es también para ayudar al entrenamiento a salir de mínimos locales que puedan estar presentes en un mini-lote pero no en otro. La figura 2.6 muestra un diagrama idealizado de como sería el camino de descenso usando las distintas técnicas de lotes y mini-lotes. El tamaño del mini-lote es otro hiperparámetro de entrenamiento que hay que definir.

Para evaluar o comparar la calidad del entrenamiento de una RNA, se mide la capacidad de generalización de la misma, es decir, la habilidad para predecir muestras nuevas. Entonces, el conjunto total de muestras disponibles se divide en: un grupo que será usado exclusivamente para entrenar la RNA y se denomina conjunto de entrenamiento; y un segundo grupo con muestras que no tuvieron ninguna participación en el entrenamiento, que se usará para probar la capacidad de generalización y se lo denomina conjunto de testeo o pruebas. Así, de varios modelos entrenados, el mejor será el que alcance el menor valor medio de la función de costo evaluada en el conjunto de pruebas. Es importante notar que, evaluar el valor medio de la función de costo en el conjunto de entrenamiento no alcanza para medir la calidad del entrenamiento, ya que una RNA puede alcanzar un valor de la función de costo muy bajo en el conjunto de entrenamiento, pero muy alto para el conjunto de prueba. A esta situación se la llama sobreentrenamiento o sobre-

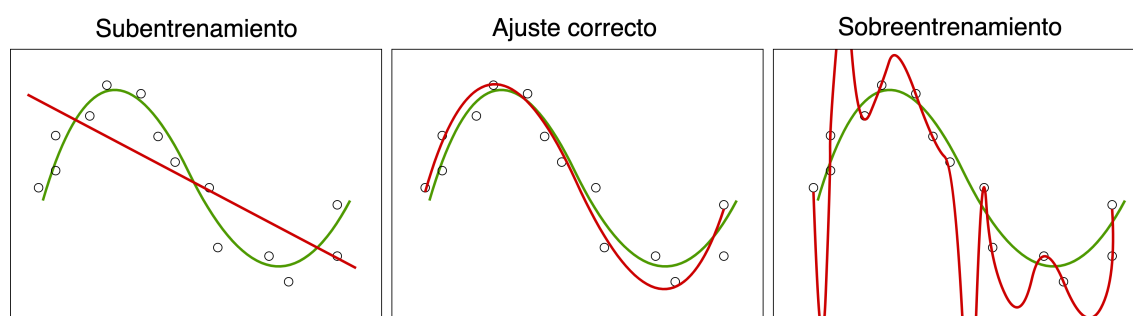


Figura 2.7: Ejemplos en una dimensión del ajuste de un conjunto de datos afectados por ruido (puntos verdes) por parte de una función (p.e. una red neuronal, línea roja). La línea verde muestra la verdadera relación entre la variable de entrada y la variable de salida. Los diagramas ilustran situaciones en las que la red incurre en un subentrenamiento de los datos (panel de la izquierda), en un ajuste óptimo (panel central) y en un sobreajuste de los datos (panel de la derecha). Diagramas basados en [Bishop \(2006\)](#)

ajuste (en inglés *overfitting*). Cuando hay sobreentrenamiento, la red aprende a estimar demasiado bien los datos con los que fue entrenada, pero pierde la capacidad de estimar muestras nuevas. Esto sucede cuando la capacidad de representación de la RNA es muy grande o el conjunto de datos es muy pequeño (panel derecho de la Fig. 2.7). Lo contrario también es un problema y se denomina subentrenamiento, y es cuando la capacidad de la RNA no alcanza para ajustar la complejidad de los datos y generaliza pobremente (panel izquierdo de la Fig. 2.7), la solución a este problema es simplemente aumentar la capacidad de la RNA expandiendo su arquitectura, ya sea agregando neuronas en las capas o agregando más capas, y así generar modelos más profundos hasta encontrar la capacidad óptima y, en consecuencia, el ajuste correcto del problema (panel central de la Fig. 2.7).

Mitigar el sobreajuste no es tan sencillo, pero existen varias opciones y la mejor es siempre agrandar el conjunto de entrenamiento cuando sea posible. Otra forma es reducir la capacidad de la RNA para evitar que pueda aprender *de memoria* los datos del conjunto de entrenamiento. Hay varias formas de hacer esto, una es simplificar la arquitectura de la red hasta encontrar la arquitectura óptima, pero esto es una tarea muy compleja. Otra forma de limitar la capacidad de la RNA es imponiendo restricciones en los pesos de la red (regularizaciones) limitando los valores que puedan adoptar. Existen muchas

técnicas de regularización y hay mucha bibliografía al respecto. En esta introducción, sólo mencionaremos una de las más comunes que se denomina decaimiento de pesos, en inglés Weight Decay (WD), y consiste en incorporar un término en la función de costo que penalice la norma cuadrática de los pesos de cada capa de la red. Entonces la función de costo modificada queda determinada por

$$\mathcal{L}_{WD} = \mathcal{L} + \frac{\lambda}{2m} \sum_{l=1}^L \|w^l\|_2^2 \quad (2.21)$$

Donde w^l es la matriz de pesos de la capa l , m la cantidad de elementos de w^l , L es la cantidad de capas de la red y λ es un hiperparámetro que escala el término de regularización y su valor óptimo suele encontrarse en forma empírica.

Otra manera de lidiar con el sobreajuste es permitir que suceda e intentar detectarlo evaluando la capacidad de generalización de la red utilizando otro conjunto de datos que se denomina conjunto de validación. Esta metodología consiste en utilizar primero el conjunto de entrenamiento para modificar los pesos de la red durante una cantidad fija f de épocas para luego usar el conjunto de validación para determinar la capacidad de generalización de la red. Si la capacidad de generalización aumentó, continuamos con otro paso de entrenamiento; si, por el contrario, la capacidad de generalización disminuyó, entonces estamos en presencia de un posible sobreajuste y entonces se detiene el entrenamiento. A esta técnica se la llama parada temprana (en inglés, *early stop*) y es el criterio que se utilizó en la experimentación realizada a lo largo esta tesis. El hiperparámetro f es la frecuencia de validación y es otro criterio de diseño que debe ser ajustado. La figura 2.8 muestra un ejemplo de la evolución de la función de costo evaluada sobre el conjunto de validación y entrenamiento como función de la cantidad de épocas. La línea roja a trazos indica el número óptimo de épocas (mínimo valor en la curva de validación) y es donde se debería detener el entrenamiento, ya que entrenar por más épocas generaría sobreentrenamiento. Siempre es preferible mantener tres conjuntos de datos (entrenamiento, validación y testeo), porque, aunque el conjunto de validación no se utilizó para actualizar los pesos, participó durante el entrenamiento como criterio de selección y es mejor evaluar la capacidad de generalización con un conjunto independiente que no haya tenido ninguna participación en el proceso de entrenamiento. En la experimentación realizada para este trabajo, como en la mayoría de los experimentos se utilizaron datos sintéticos, siempre se usaron tres conjuntos de datos y todas las métricas fueron evaluadas en un conjunto de testeo independiente del entrenamiento.

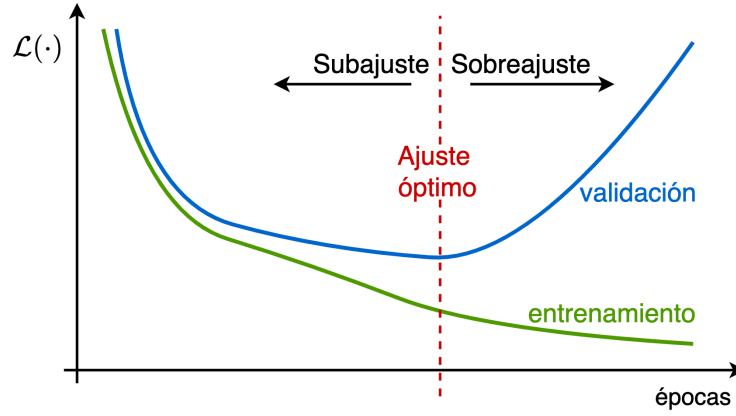


Figura 2.8: Evolución del valor de la función de costo $\mathcal{L}$ en función de la época de entrenamiento para el conjunto de validación (azul) y el conjunto de entrenamiento (verde). El ajuste óptimo coincide con el mínimo valor de la función de costo para el conjunto de validación

2.3.3. Redes neuronales convolucionales

Las redes neuronales convolucionales, en inglés Convolutional Neural Networks (CNNs) (Lecun, 1989), son un tipo de RNA especializadas en procesar datos con topología reticular. Esta característica las hace especialmente atractivas para procesar datos provenientes de modelos dinámicos donde las variables espaciales están discretizadas en una red (como es el caso de los modelos que analizaremos en este trabajo). Reciben el nombre de *convolucionales* debido a que implementan una operación matemática denominada convolución que se define, en términos generales, como la integral del producto de dos funciones donde una de ellas fue reflejada y desplazada respecto del eje de ordenadas. La convolución entre dos funciones f y g es otra función que suele escribirse como $(f * g)$ y esta dada por

$$(f * g)(t) = \int f(t - a)g(a) da. \quad (2.22)$$

En el campo específico de las RNA, la función f suele denominarse como la *entrada*, la función g es el *filtro* y el resultado se lo llama *mapa de características*. La convolución descrita en la Eq. 2.22 está definida para un dominio continuo, lo que no suele ser el caso de las aplicaciones de RNA, por lo que suele utilizarse la versión discreta. Por ejemplo, la

convolución entre dos campos bidimensionales y discretos F y G dará como resultado un mapa de características M bidimensional y discreto de la forma:

$$M(i, j) = (F * G)(i, j) = \sum_m \sum_n F(i - m, j - m) G(m, n) \quad (2.23)$$

donde m y n son las dimensiones del filtro. El efecto de incrementar i y j representa el movimiento del filtro a lo largo de la primera y segunda dimensión respectivamente. La Fig. 2.9 muestra gráficamente la operación de convolución entre un campo de entrada de dimensión (5×5) (cuadro rojo) y el filtro de dimensión (3×3) . Los recuadros de color intenso muestran los elementos involucrados para el cálculo del punto $(i = 3, j = 2)$. Las capas convoluciones pueden tener tantos filtros como se quiera. Por ejemplo, si se tuvieran p filtros en la capa de la Fig. 2.9, el mapa de características sería un volumen de profundidad p . Nótese que la dimensión del mapa de características es menor que el campo de entrada. Esto se debe a que el filtro debe *solaparse* con los elementos del campo de entrada, es decir, que si el primer elemento de campo de entrada es $(i = 0, j = 0)$, la primera evaluación del filtro sería en $(i = 2, j = 2)$. Se puede evitar este efecto ampliando el campo de entrada (en inglés *padding*), por ejemplo agregando filas y columnas con el valor de cero o con algún valor apropiado para el problema en cuestión.

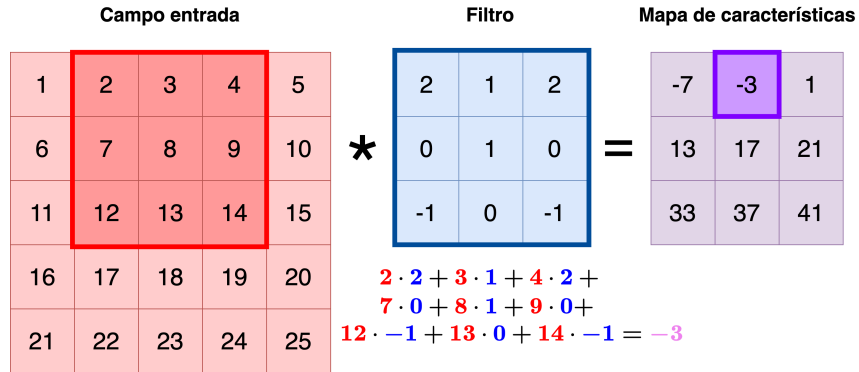


Figura 2.9: Ejemplo de convolución aplicada a un campo bidimensional (rojo) con un filtro bidimensional (azul). El resultado de la operación es el cuadro violeta. Se muestra un ejemplo de las operaciones involucradas para el cálculo del elemento del mapa de características remarcado en la figura (violeta intenso). Los cuadros de color intenso marcan los elementos involucrados en este cálculo.

Finalmente, en una CNN donde se tiene una secuencia de capas convolucionales, el mapa de característica de una capa se vuelve el campo de entrada de la siguiente y así sucesivamente.

Capítulo 3

Estimación de la incertidumbre

Una característica distintiva de los modelos de pronóstico de sistemas dinámicos caóticos es su sensibilidad a la condición inicial, y en particular, cómo esta sensibilidad varía o depende del estado. Así, la incertidumbre asociada a los pronósticos está vinculada a esta sensibilidad quedando determinada también por la dinámica del sistema, y su variabilidad también depende del estado, tal como se muestra en la Fig. 3.1. En el panel de la izquierda, la condición inicial (CI) está situada en una zona de baja incertidumbre del atractor y entonces sus pronósticos son predecibles, es decir, el error introducido en la CI crece muy lentamente y todas las trayectorias (pronósticos) se mantienen unidas. Por el contrario, el panel de la derecha muestra pronósticos inicializados en una zona de alta incertidumbre y puede verse cómo las perturbaciones introducidas en las CI crecen muy rápidamente generando que los pronósticos diverjan en el corto plazo. Es importante distinguir entre esta incertidumbre, que es intrínseca y propia de la dinámica del sistema, y la varianza estadística asociada a la técnica de estimación que se utilice y a la calidad y cantidad de muestras disponibles para efectuar dicha estimación. En este trabajo de tesis se busca fundamentalmente estudiar la capacidad de las redes neuronales artificiales (RNAs) de aprender la relación existente entre el estado del sistema y su incertidumbre intrínseca, es decir, la que está asociada exclusivamente a la dinámica del sistema. También se analizará la convergencia y la varianza de la estimación asociada principalmente al diseño de la arquitectura de las RNAs, la elección de los hiperparámetros de entrenamiento y el conjunto de datos disponible para entrenar, es decir, la varianza asociada a la técnica de estimación. Como veremos en este capítulo, ambas fuentes de incertidumbre están acopladas en la estimación de sistemas reales de pronóstico, y en consecuencia es difícil lograr independizarlas completamente.

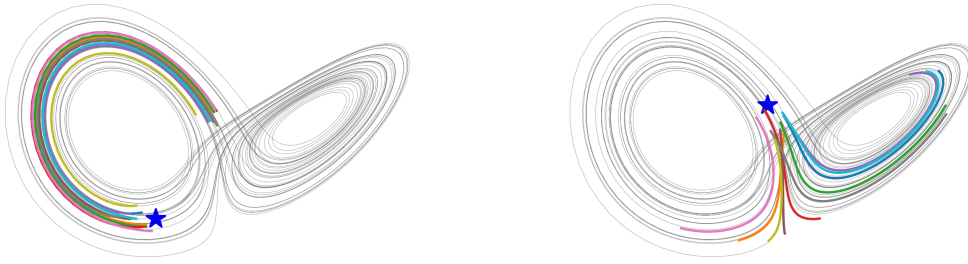


Figura 3.1: Los diagramas muestran el atractor Lorenz'63 en líneas gris claro. Las líneas de color muestran las trayectorias de pronósticos inicializados a partir de perturbaciones de la condición inicial (estrella azul) situada en una zona de baja incertidumbre (izquierda) y en una zona de alta incertidumbre (derecha).

También se estudiarán otras fuentes de incertidumbre, por ejemplo, cuando las RNAs son entrenadas en forma supervisada, es necesario armar un conjunto de entrenamiento donde se indique explícitamente qué debe aprender la red (objetivo de entrenamiento) para cada ejemplo. Esto supone un desafío, puesto que, en general, no se dispone del estado verdadero del sistema y, en cambio, solo tenemos datos parciales y con errores de observación y representación. Además, se pretende que la red estime los parámetros de una función de densidad de probabilidad que, en principio, son desconocidos. Esta es un área bastante activa de investigación y varias estrategias fueron planteadas para abordar una solución. A continuación analizaremos algunas de estas soluciones y, con base en estos antecedentes, propondremos una nueva metodología de entrenamiento en la sección 3.4 y analizaremos su comportamiento en la sección 3.5

3.1. Entrenar directamente de un ensamble

En el capítulo 2 se introdujo la técnica de ensambles para generar pronósticos probabilísticos (Sec. 2.1) y asimilación de datos, en inglés Data Assimilation (DA) (Sec. 2.2.2). Así, la metodología de ensambles es, al momento de escribir esta tesis, la técnica más robusta y más utilizada para estimar incertidumbre en el contexto de los pronósticos meteorológicos. Los ensambles proveen una estimación directa de la incertidumbre (ej. la dispersión del ensamble) que puede ser utilizada para entrenar una red neuronal artificial (RNA) buscando, por ejemplo, que la red sea capaz de aprender la relación entre los datos de entrada (ej. pronóstico determinístico) y la incertidumbre estimada por el

ensemble para el mismo plazo de pronóstico.

[Scher and Messori \(2018\)](#) usaron esta idea. Basándose en la hipótesis de que la incertidumbre de los pronósticos depende en gran medida del estado inicial, suponen que, dada una serie suficientemente larga de pronósticos históricos, debería ser posible inferir cuáles de estas condiciones iniciales conducen a un pronóstico con un alto nivel de incertidumbre o lo contrario.

Ellos proponen un método basado en aprendizaje profundo usando redes neuronales convolucionales, en inglés Convolutional Neural Networks (CNNs), que entrenan para determinar un valor escalar que represente el grado de incertidumbre asociado al estado de la atmósfera en la entrada de la red. Entonces, la entrada está dada por una selección de campos de variables meteorológicas en distintos niveles de presión y la salida es un valor escalar que resume en forma global la incertidumbre vinculada a la situación meteorológica descrita por los campos en la entrada.

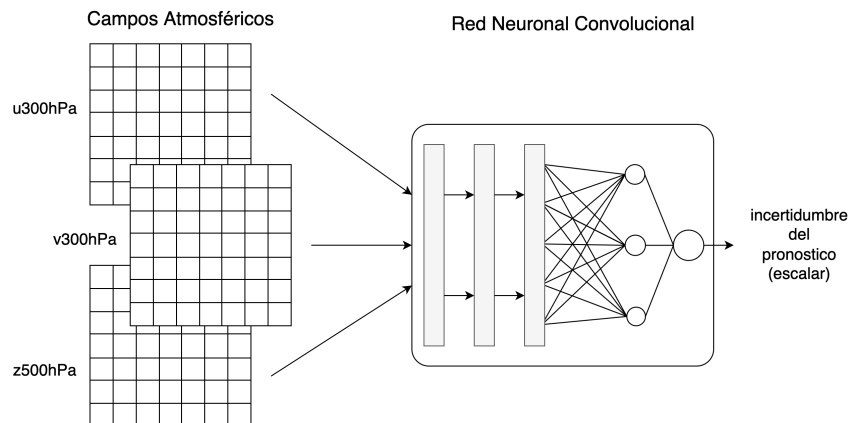


Figura 3.2: Esquema de RNA convolucional propuesta en [Scher and Messori \(2018\)](#) para estimar un indicador de incertidumbre en los pronósticos.

El conjunto de datos utilizado por [Scher and Messori](#) para las CNN son pronósticos retrospectivos generados con la segunda versión del modelo GEFS ([Hamill et al., 2013](#)). Los pronósticos corresponden al período comprendido entre los años 1985 y 2016 contando con mas de de 10.000 inicializaciones diarias. Los datos de entrada de las CNN son campos meteorológicos para un dominio espacial del hemisferio norte que abarca Europa y Estados Unidos. En particular utilizan campos de altura geopotencial de 500hPa y el viento zonal y meridional en 300hPa. Y como objetivo de entrenamiento evalúan dos estimaciones distintas: una basada en la dispersión del ensemble del geopotencial de

500hPa del GEFS y la otra basada en el RMSE de los diferentes miembros del ensamble respecto del miembro control. Para eliminar la variabilidad estacional en los datos, el error y la dispersión siempre son considerados sobre las anomalías.

Para los experimentos usaron una red con una capa convolucional de 32 canales y una capa oculta de 32 neuronas. Las tres variables de entrada se representan como canales separados en la capa de entrada, como se muestra en el esquema de la Fig. 3.2. Se usó el error cuadrático medio como función de costo y las redes se entrenaron durante 30 épocas.

Se evaluaron diferentes plazos de pronósticos entre 3 y 6 días. Se realizaron experimentos con el objetivo de mostrar las propiedades estadísticas de las estimaciones y compararlas con otras técnicas como la *persistencia* y la *dimensión local* (Faranda et al., 2017), el *cluster de tipos meteorológicos* (Ferranti et al., 2015) y la técnica de *vecinos más cercanos*. Los resultados mostraron que, en general, para todos los plazos las RNA fueron capaces de predecir la incertidumbre de los pronósticos significativamente mejor que las otras técnicas evaluadas, con excepción de los ensambles. También se realizó un experimento de sensibilidad al tamaño del conjunto de entrenamiento, que mostró que el desempeño de las redes aumenta junto con el tamaño del conjunto sin alcanzarse ningún aplanamiento en la curva, lo cual sugiere que una limitación de la metodología es la falta de más datos históricos.

Los autores concluyen que, si bien el método propuesto no es tan hábil para predecir la incertidumbre de los pronósticos como la dispersión del ensamble, supera significativamente a los métodos existentes. En términos de eficiencia computacional, el entrenamiento de la red tarda unos 30 minutos en 2 GPU NVidia K80 y la inferencia en casos de prueba (presentar a la red un campo de entrada y predecir la incertidumbre) es del orden de segundos, y por tanto varios órdenes de magnitud más barato que generar un ensamble de simulaciones numéricas.

Continuando en esta misma línea de trabajo, en Grönquist et al. (2021) se usaron RNA para estimar campos de incertidumbre asociados a variables meteorológicas.

Los autores entrenan CNNs para aprender la relación entre los pronósticos y su incertidumbre (datos de entrada) dados por un ensamble pequeño, por ende relativamente limitado para proveer una buena estimación de la incertidumbre, y la dispersión estimada con un ensamble grande de pronósticos (objetivo de entrenamiento), que se asume capaz de generar una estimación más precisa de la incertidumbre.

El conjunto de datos que utilizan son los pronósticos retrospectivos por ensamble del Centro Europeo (European Centre for Medium-Range Weather Forecasts - ECMWF) que

consta de 10 miembros (ENS10) más un miembro de control. Para entrenar y contrastar los resultados usaron los reanálisis ERA5 también producidos por el Centro Europeo.

Entonces plantean tres tareas diferentes: cuantificar la incertidumbre, corregir el sesgo y calibrar la función de densidad de probabilidad, en inglés Probability Density Function (PDF), resultante y cada una de estas tareas se aborda con una RNA diferente. Para cuantificar de la incertidumbre entrenaron una RNA usando como objetivo de entrenamiento la desviación estándar de un ensamble de 10 miembros y como entrada un ensamble pequeño de 5 miembros. Para la corrección del sesgo entrenan una RNA utilizando como objetivo de entrenamiento los reanálisis (ERA5) y en la entrada los campos medios del ensamble de 5 miembros. Con esta metodología obtienen una red que corrige parte del sesgo sistemático presente en la media del ensamble y otra red que representa la incertidumbre estimada con el ensamble. Y finalmente un tercer entrenamiento, esta vez usando como entrada las salidas de las redes previamente entrenadas. Esta última red se entrena con una función de costo derivada de la métrica CRPS, del inglés Continuous Ranked Probability Score (CRPS), similar a la definida en [Camporeale and Carè \(2021\)](#) (ver Sec. 3.3) que da una medida de cuán representativa es una observación respecto de una PDF. El uso del CRPS en este caso indica que no se está aprendiendo la incertidumbre únicamente a partir del ensamble, sino que hay un ajuste posterior en base a los análisis.

Los modelos se entrenaron para corregir y cuantificar la incertidumbre de pronósticos a 48 horas de plazo para la temperatura al nivel de 850hPa y el geopotencial de 500hPa. La entrada de la red consiste en los campos de temperatura, geopotencial, viento, divergencia, velocidad vertical y humedad específica en el nivel seleccionado y en tres plazos de pronósticos 0, 24 y 48 horas. Los resultados mostraron que la red calibrada, usando información del ensamble de 5 miembros, resultó en un aumento de la eficacia de más del 14,5 % comparado con el ensamble de 10 miembros.

Los autores concluyen que la corrección de la media del ensamble y la estimación de la varianza a partir de un ensamble de 5 miembros, usando RNA, permite obtener valores de CRPS que son inferiores (mejores) a los obtenidos a partir de la media y la dispersión del ensamble de 5 miembros e incluso inferiores a los obtenidos a partir de la media y dispersión calculados con el ensamble de 10 miembros. Esto último probablemente se deba a la calibración final que se introduce con la red entrenada usando al CRPS como función de costo.

3.2. Entrenar usando la verosimilitud de los datos

La aproximación de funciones consiste en obtener una función $\tilde{\mu}(x)$ que aproxime a otra función desconocida $\mu(x)$ de la cual sólo se tiene un conjunto de valores observados $y(x)$ que, en la mayoría de los casos, suelen estar sujetos a errores intrínsecos a la obtención del dato, por ejemplo: errores de medición. Asumiendo que dichos errores tienen distribución gaussiana, podemos describir las observaciones mediante el siguiente modelo

$$y(x) \sim \mathcal{N}(\mu(x), \sigma(x)) \quad (3.1)$$

donde $\sigma(x)$ representa la desviación estándar del error y puede ser interpretada como errores que desplazan a la observación $y(x)$ de su valor medio $\mu(x)$.

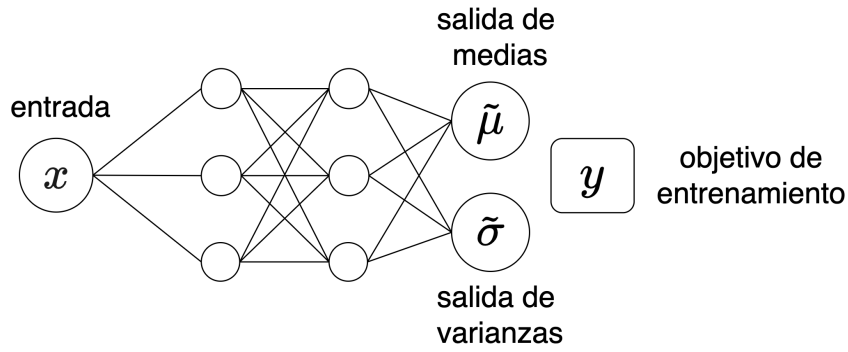


Figura 3.3: Esquema de la RNA propuesta para estimar $\tilde{\mu}(x) \approx \mu(x)$ y $\tilde{\sigma}(x) \approx \sigma(x)$ entrenada usando las observaciones $y(x)$.

Cuando usamos RNA para aproximar una función, utilizando las observaciones como objetivo de entrenamiento, se puede interpretar que la salida de la red $\tilde{\mu}(x)$ es un estimador de la media poblacional $\mu(x)$ de la distribución de las observaciones y condicionada en x . En este trabajo, nuestro objetivo es, además de estimar la media de la distribución ($\tilde{\mu}(x)$), cuantificar la incertidumbre asociada a dicha predicción ($\tilde{\sigma}(x)$) basándonos únicamente en el conjunto de observaciones $y(x)$. Cuando la desviación estándar del ruido $\tilde{\sigma}(x)$ depende de x el problema se denomina heterocedástico y podemos definir σ como la desviación estándar poblacional del ruido presente en las observaciones condicionada a la entrada x . Cuando el ruido no depende del estado del sistema, se denomina homocedástico. A lo largo de esta tesis supondremos siempre que el error es heterocedástico y el desafío es encontrar la relación entre el estado del sistema y el ruido o la incertidumbre.

Este problema se introduce en [Nix and Weigend \(1994\)](#), siendo uno de los trabajos seminales en el uso de RNA para la estimación de PDF. En ese trabajo se propone maximizar el logaritmo de la verosimilitud de las observaciones como método para aproximar de manera conjunta $\tilde{\mu}(x) \approx \mu(x)$ y $\tilde{\sigma}^2(x) \approx \sigma(x)^2$ (esquema de la Fig. 3.3). Si asumimos que el ruido en las observaciones sigue una distribución gaussiana, entonces podemos obtener la verosimilitud de la i -ésima observación $y_i = y(x_i)$ condicionada a x y a los pesos de la red w como:

$$P(y_i|x_i, w) = (2\pi\tilde{\sigma}_i^2)^{-\frac{1}{2}} \exp \left[-\frac{(y_i - \tilde{\mu}_i)^2}{2\tilde{\sigma}_i^2} \right] \quad (3.2)$$

Dado que maximizar una función es equivalente a minimizar el negativo de la misma, podemos obtener la siguiente función de costo tomando el negativo del logaritmo de la verosimilitud de todas las observaciones presentes en el conjunto de entrenamiento:

$$\mathcal{L} = \sum_i \frac{1}{2} \left[\frac{(y_i - \tilde{\mu}_i)^2}{2\tilde{\sigma}_i^2} + \ln \tilde{\sigma}_i^2 \right] \quad (3.3)$$

Notar que en el caso homocedástico en donde $\tilde{\sigma}$ se asume independiente de x , su valor se puede remover de la función de costo, ya que pasa a ser una constante que no influye en la ubicación del mínimo de la misma. En este caso obtenemos el error cuadrático medio. Dicho de otra manera, el error cuadrático medio nos provee el estimador de máxima verosimilitud bajo la hipótesis de que el ruido de los datos es independiente de la entrada. [Nix and Weigend](#) usan la función de costo 3.3 para entrenar una red totalmente conectada similar a la del esquema de la Fig. 3.3 donde un grupo de neuronas en la capa de salida produce la estimación de las medias $\tilde{\mu}(x)$ y otro grupo estima las varianzas $\tilde{\sigma}^2(x)$. Los autores sugieren que cada salida esté completamente conectada a su propia capa oculta, es decir, que las salidas de medias y las de varianza no compartan neuronas en la capa oculta, ya que esta configuración les dio mejores resultados. Además, para garantizar el correcto entrenamiento de la RNA es importante que la función de costo 3.3 esté definida para todo x , es decir, que la varianza no tome el valor cero. Para evitar esto, inicializan los pesos de la red de manera aleatoria pero asegurándose de que la estimación inicial de las varianzas sea uno para todo el espacio de los datos de entradas. Esta elección además tiene la ventaja de no asumir un comportamiento determinado de las varianzas antes de que la estimación de la media empiece a converger. Para reforzar esta idea, proponen no actualizar los pesos de las varianzas durante los primeros pasos del entrenamiento hasta lograr obtener una estimación de las medias relativamente buena. Para entrenar la red

usan la función de costo 3.3 y el algoritmo de retropropagación definido en la sección 2.3.2. Entonces, reemplazando Eq. 3.3 en Eq. 2.18 nos queda determinada la ecuación de actualización de los pesos de salida de la red de la siguiente manera:

$$\Delta w_{\mu,j} = \eta_{lr} \frac{(y_i - \tilde{\mu}_i)^2}{\tilde{\sigma}_i^2} h_j(x_i) \quad (3.4)$$

donde η_{lr} es la tasa de aprendizaje, $w_{y,j}$ son los pesos de la capa de salida y $h_j(x_i)$ representa el resto de las capas de la red neuronal que no dependen de $\tilde{\sigma}^2$.

Una interpretación adicional del gradiente de la función de costo dada por 3.4 consiste en asociar la varianza con la tasa de aprendizaje η_{lr} resultando en el factor $\frac{\eta_{lr}}{\sigma_i^2}$. Es decir que, en patrones de entrada para los cuales σ_i^2 es menor, la tasa de aprendizaje se ve amplificada respecto a los casos en donde ésta es mayor. Así, dada una RNA con una cierta capacidad, el ruido heterocedástico implica darle más peso a determinadas instancias del conjunto de entrenamiento (aquellas con menor error) por sobre otras (con error mayor). Finalmente, [Nix and Weigend](#) advierten que esta metodología es muy sensible al sobreentrenamiento y que sólo debería usarse con conjuntos de entrenamiento grandes. Esto es fácil de ver si pensamos que, cuando tenemos un conjunto de entrenamiento muy pequeño, la red puede aproximar perfectamente el conjunto de entrenamiento, lo cual implicaría que las varianzas tiendan a cero y no logren estimar la incertidumbre asociada a las observaciones. Además, es importante notar que estamos infiriendo la varianza del ruido por lo que se requiere de más muestras que en el caso de la media.

Este problema también es abordado en [Bishop and Qazaz \(1997\)](#) pero desde un enfoque diferente. Ellos señalan que la estimación de la varianza basada en máxima verosimilitud es sesgada porque busca maximizar la probabilidad conjunta de μ y σ . Tal como se mencionó anteriormente, debido a que la media (μ) se estima a partir de los datos en el conjunto de entrenamiento D , esta puede incorporar parte del ruido presente en las observaciones y esto conduce a una subestimación de la varianza. Ellos analizan el problema desde un enfoque bayesiano y usan la propuesta de [MacKay \(1992\)](#) de, en vez de maximizar la verosimilitud de los datos, maximizar la verosimilitud marginal integrando la dependencia sobre μ

$$p(D|\sigma^2) = \int p(D|\sigma^2, \mu) p(\mu) d\mu \quad (3.5)$$

De esta manera, se logra una estimación insesgada de σ^2 , o de manera equivalente, de

los pesos de la RNA correspondientes a la estimación de varianzas (w_σ). Esto conduce a un análisis bayesiano jerárquico donde los parámetros $\tilde{\mu}$ y $\tilde{\sigma}^2$ son considerados variables aleatorias que dependen de los pesos de las redes w_μ y w_σ respectivamente (usan redes distintas para aproximar cada uno de los parámetros). Ahora nos queda definir el término $p(\mu)$ que es la probabilidad a priori (o *prior*) del parámetro μ o de forma equivalente de los pesos w_μ . Bishop and Qazaz definen para cada red priors gaussianos de la siguiente manera:

$$p(w|\alpha_w) = \left(\frac{\alpha_w}{2\pi}\right)^{\frac{1}{2}} \exp\left\{-\frac{\alpha_w}{2}\|w\|^2\right\} \quad (3.6)$$

donde α_w (α_{w_μ} para los pesos de la red de medias y α_{w_σ} para los pesos de la red de varianzas) son hiperparámetros de entrenamiento. Al desarrollar las ecuaciones del modelo bayesiano jerárquico, suceden dos cosas interesantes. La primera es que las hiperpriors aparecen en las ecuaciones a minimizar como términos independientes de la forma $(\frac{\alpha_w}{2}\|w\|^2)$, es decir, como regularizaciones sobre los pesos de la red. En efecto, la regularización por decaimiento de pesos, en inglés Weight Decay (WD), puede obtenerse a partir de suponer un prior gaussiano para los pesos de la red con media cero y varianza igual al factor de decaimiento. Lo segundo es el algoritmo de entrenamiento de los pesos w_μ y w_σ que Bishop and Qazaz lo describen como un bucle externo en el que el valor más probable ($w_{\sigma MP}$) de los pesos w_σ de la red de varianzas se encuentra por minimización de la *posterior* ($p(w_\sigma|D, w_{\mu MP}, \alpha_{w_\sigma})$) y un bucle interior que encuentra el valor óptimo para $w_{\mu MP}$ minimizando la *posterior* ($p(w_\mu|D, w_{\sigma MP}, \alpha_{w_\mu})$). Entonces, w_μ evoluciona en una escala de tiempo rápida y w_σ en una escala de tiempo lenta.

Como ya se comentó, Nix and Weigend también concluyeron algo similar. Aunque su metodología permite estimar μ y σ^2 en forma conjunta, observaron que en la práctica los mejores resultados se obtienen entrenando primero los pesos de la red de medias sin actualizar los pesos de la red de varianzas y, cuando la estimación de medias es robusta, permitir actualizar los pesos de la red de varianzas. En esta tesis, se empleará un procedimiento similar a este en donde las redes son entrenadas por separado, primero la de medias y luego la de varianzas, no permitiendo la interdependencia de ambos parámetros como se verá en el capítulo siguiente.

Esta metodología, entrenar usando la verosimilitud, fue puesta en práctica con datos reales de observaciones de estaciones meteorológicas de superficie y con datos de un modelo de pronóstico numérico (MPN) en Wang et al. (2018). Ellos diseñaron un método

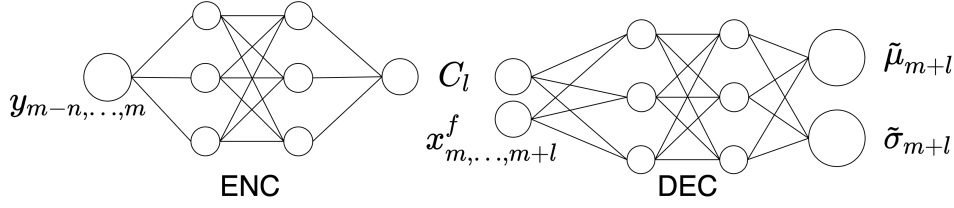


Figura 3.4: Esquema de fusión de datos de estaciones meteorológicas con pronósticos numéricos propuesto en Wang et al. (2018). La red neuronal ENC codifica la evolución dinámica de las observaciones a un espacio latente C_m y la red neuronal DEC combina el pronóstico numérico $x_{m, \dots, m+l}^f$ y C_m para obtener $(\tilde{\mu}_{m+l} \text{ y } \tilde{\sigma}_{m+l})$ que son la estimación del estado y la incertidumbre del pronóstico respectivamente.

donde fusionan observaciones del pasado y pronósticos numéricos para entrenar RNA profundas que aprenden a corregir parte del error presente en los pronósticos además de estimar la incertidumbre asociada a los mismos. Al igual que Nix and Weigend (1994), asumen que el error tiene una distribución normal y usan el logaritmo negativo de la verosimilitud gaussiana como función de costo (Ec. 3.3). Específicamente, construyen el conjunto de entrenamiento a partir de secuencias temporales de observaciones $y_{m-n, \dots, m}$ de longitud n y pronósticos numéricos $x_{m, \dots, m+l}^f$ inicializados en el tiempo m para un plazo de tiempo l .

La forma de combinar las observaciones puntuales de las estaciones meteorológicas con los pronósticos es mediante dos RNA, una es un codificador (ENC) y la otra un decodificador (DEC) ambos de capas LSTM del inglés Long-Short Term Memory (Srivastava et al., 2015) como se indica en la figura 3.4. Entonces, las observaciones $y_{m-n, \dots, m}$ son presentadas al codificador que aprende una caracterización de la evolución dinámica de las observaciones en el espacio latente $C_m = \text{ENC}(y_{m-n, \dots, m})$. El decodificador utiliza el espacio latente de las observaciones y los pronósticos numéricos $x_{m, \dots, m+l}^f$ para aprender una versión corregida del pronóstico $\tilde{\mu}_{m+l}$ y su correspondiente incertidumbre $\tilde{\sigma}_{m+l}$:

$$\tilde{\mu}_{m+l}, \tilde{\sigma}_{m+l} = \text{DEC}(C_m, x_{m, \dots, m+l}^f) = \text{DEC}(\text{ENC}(y_{m-n, \dots, m}), x_{m, \dots, m+l}^f)$$

En Wang et al. (2018) realizaron varios experimentos de comparación y sensibilidad donde las RNA entrenadas usando una función de costo basada en la verosimilitud gaussiana, del inglés Likelihood (LKHD), mostraron resultados superadores, aunque notaron que esta técnica requirió más iteraciones para converger y que esto puede deberse a que tiene que optimizar dos magnitudes simultáneamente.

Las conclusiones a las que llegan Wang et al. concuerdan con las predichas por Nix and Weigend y por Bishop and Qazaz dando a la estimación de la incertidumbre con RNA entrenadas con LKHD una fuerte base analítica y una demostración práctica.

3.3. Entrenar usando la métrica CRPS

En Camporeale and Carè (2021) también se analiza el problema de la asignación de incertidumbre a predicciones puntuales, pero haciendo especial hincapié en la calibración. Ellos plantean que, cuando se trata de una predicción probabilística, la calibración o confiabilidad es tan importante como la precisión y la definen como *el requisito de que la probabilidad estimada sea similar a la frecuencia de ocurrencia esperada del suceso*, es decir, que exista coherencia estadística entre las predicciones probabilísticas y la ocurrencia observada de un determinado suceso. En predicciones de sucesos discretos mide que, si la probabilidad pronosticada del suceso es p , el mismo ocurra en promedio con frecuencia p . Este concepto puede extenderse a los pronósticos de variables continuas, por ejemplo mediante diagramas de confiabilidad o curvas de CRPS.

Camporeale and Carè demuestran que el posprocesamiento de la calibración compite con la precisión y que, por lo tanto, hay que buscar un equilibrio óptimo entre estas dos cualidades igualmente importantes de un pronóstico probabilístico. Así, dos modelos con idéntica calificación para una métrica basada en el error (ellos lo muestran para LKHD, pero lo mismo ocurre con el CRPS) pueden tener diagramas de confiabilidad notablemente diferentes. Por lo que minimizar una función de costo como LKHD o CRPS para errores fijos no implica necesariamente lograr un modelo bien calibrado. Aunque los autores aclaran que no están sugiriendo que cualquier modelo generado mediante la minimización de LKHD esté inevitablemente mal calibrado, es necesario implementar algún procedimiento que la garantice ya sea de forma explícita incorporándola directamente en la función de costo, o implícitamente a través de algún mecanismo durante el proceso de entrenamiento.

Entonces introducen una función de costo que implementa este compromiso, y que denominaron función de costo basada en la precisión y la confiabilidad, en inglés ACCurate and Reliable Uncertainty Estimate (ACCRUE) y que utiliza la métrica CRPS para medir la precisión y derivaron una fórmula analítica para la confiabilidad RS del inglés Reliability Score basada en distribuciones gaussianas. Entonces, dado un pronóstico probabilístico

(μ, σ) y una observación y^o , el CRPS se define como:

$$\text{CRPS} = \int_{-\infty}^{\infty} [C(y) - H(y - y^o)]^2 dy \quad (3.7)$$

donde $C(\cdot)$ es la función de distribución acumulada del pronóstico y $H(\cdot)$ es la función Heaviside. Así, el CRPS para distribuciones gaussianas es:

$$\text{CRPS}(\epsilon, \sigma) = \sigma \left[\frac{\epsilon}{\sigma} \text{erf} \left(\frac{\epsilon}{\sqrt{2}\sigma} \right) + \sqrt{\frac{2}{\pi}} \exp \left(-\frac{\epsilon^2}{2\sigma^2} \right) - \frac{1}{\sqrt{\pi}} \right] \quad (3.8)$$

donde $\epsilon = y^o - \mu$ es la distancia entre la media del pronóstico y la observación. Esta ecuación, al igual que LKHD, no garantiza por sí sola la calibración, y entonces introducen una métrica de confiabilidad para pronósticos probabilísticos normalmente distribuidos que compara la distribución acumulada empírica del pronóstico con una distribución normal estándar. Los errores normalizados ε son calculados sobre una muestra de N predicciones-observaciones de manera que tengan una distribución normal estándar con cdf $\Theta(\varepsilon) = \frac{1}{2}(\text{erf}(\varepsilon) + 1)$. Y entonces definen:

$$\text{RS} = \int_{-\infty}^{\infty} [\Theta(\varepsilon) - C_e(\varepsilon)]^2 d\varepsilon \quad (3.9)$$

donde $C_e(\cdot)$ es la distribución acumulada empírica del error estándar ε :

$$C_e(\varepsilon) = \frac{1}{N} \sum_{i=1}^N H(\varepsilon - \varepsilon_i) \quad (3.10)$$

con $\varepsilon_i = \frac{(y_i^o - \mu_i)}{\sqrt{2}\sigma_i}$. Obsérvese que cada error $(y_i^o - \mu_i)$ se estandariza con respecto a una σ_i diferente (dependiente de los datos de entrada). La integral de la Ec. 3.9 se puede calcular analíticamente, obteniendo:

$$\text{RS} = \sum_{i=1}^N \left[\frac{\varepsilon_i}{N} (\text{erf}(\varepsilon_i) + 1) - \frac{\varepsilon_i}{N^2} (2i - 1) + \frac{\exp(-\varepsilon_i^2)}{N\sqrt{\pi}} \right] - \frac{1}{2} \sqrt{\frac{2}{\pi}} \quad (3.11)$$

Esto da lugar a un problema de optimización de dos objetivos. [Camporeale and Carè](#) demuestran en su trabajo que CRPS y RS no pueden alcanzar simultáneamente su valor mínimo, y entonces no existe una solución única para las varianzas que minimizan RS

y, por lo tanto, los autores consideran más adecuado interpretar a RS como un término de regularización en la función de costo ACCRUE. Para resolver este problema de doble optimización, [Camporeale and Carè](#) deciden escalar la funciones de la siguiente manera:

$$\text{ACCRUE} = \beta \cdot \overline{\text{CRPS}} + (1 - \beta)\text{RS} \quad (3.12)$$

con

$$\beta = \frac{\text{RS}_{\min}}{\overline{\text{CRPS}}_{\min} + \text{RS}_{\min}} \quad (3.13)$$

siendo $\overline{\text{CRPS}}_{\min} = (\sqrt{(\log 4)/2N}) \sum_{i=1}^N \varepsilon_i$ que es simplemente el error medio escalado y $\text{RS}_{\min}$ que es un valor que sólo depende de N .

Finalmente los autores sugieren desacoplar la función que aproxime el valor medio del estado $\mu(x)$ y utilizar ACCRUE para estimar únicamente la varianza $\sigma^2(x)$. Esto permite usar, por ejemplo, un modelo físico para predecir la dinámica y ACCRUE para estimar la incertidumbre de la predicción.

3.4. Propuesta: entrenar directamente del error

Entonces, hemos visto que las RNA pueden ser buenos estimadores de la incertidumbre de los pronósticos. Ya sea entrenándolas mediante una metodología *directa* usando las predicciones de un ensamble ([Scher and Messori 2018](#) y [Grönquist et al. 2021](#)) o mediante un entrenamiento *indirecto* donde se usan observaciones históricas como en [Nix and Weigend \(1994\)](#), [Bishop and Qazaz \(1997\)](#), [Wang et al. \(2018\)](#) y [Camporeale and Carè \(2021\)](#) sin requerir información a priori de la varianza del error. Ambos enfoques, como hemos analizado, tienen ventajas y desventajas. Los métodos basados en ensambles, como aprenden directamente la incertidumbre del ensamble, pueden entrenarse usando una función de costo basada en el error cuadrático medio, lo cual conduce a una convergencia más simple (puesto que no se realiza una optimización doble de media y varianza) y mas rápida (ya que resulta menos sensible a los hiperparámetros) y además no sufren, en principio, de inestabilidades numéricas ni son tan sensibles al sobreentrenamiento porque no hay inversión de la varianza durante el entrenamiento. Pero como los ensambles tienen dificultad para representar el error de modelo, a menos que sean diseñados cuidadosamente para minimizar este efecto, el error sistemático afecta a la estimación

de la varianza siendo necesaria una calibración adicional.

Mientras que los métodos que aprenden de las observaciones pueden considerar el efecto de los errores de modelo en la estimación de la incertidumbre, y simultáneamente estimar su componente sistemática, pero son más complejos de entrenar porque deben realizar una doble optimización. Además, las funciones de costo como LKHD y CRPS, que incluyen en su formulación la inversión de la varianza, son muy sensibles al sobreentrenamiento y generan inestabilidades numéricas durante el entrenamiento.

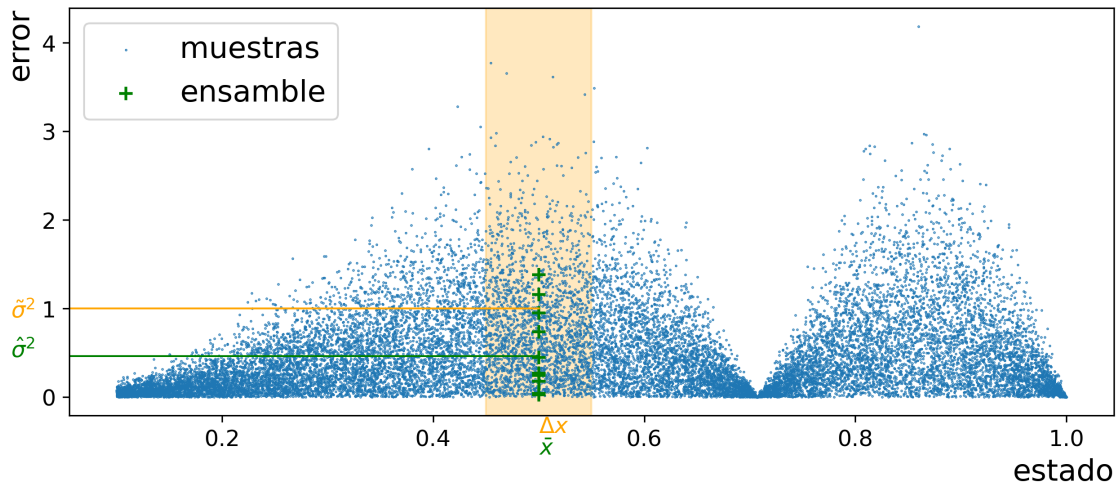


Figura 3.5: Diagrama de dispersión del valor de la variable de estado (eje x) y el error del pronóstico (eje y) asociado a cada muestra del conjunto de datos históricos (puntos azules), la banda naranja identifica las muestras pertenecientes a un intervalo Δx alrededor de un valor de estado determinado y $\tilde{\sigma}^2$ es la varianza empírica de las muestras dentro de la banda. Las cruces verdes señalan el error de cada miembro de un ensamble (10 miembros en este caso) respecto de la media y $\hat{\sigma}^2$ es la varianza empírica calculada sobre el ensamble.

En todo caso, ambos métodos no tienen porque converger al mismo valor de varianza para un estado dado. Aun si consideramos un ensamble y un conjunto de datos históricos lo suficientemente grande, el error de modelo presente en el ensamble no está representado en la varianza, lo que hace a ambas metodologías intrínsecamente distintas.

La Figura 3.5 muestra un esquema que busca aclarar las diferencias y semejanzas entre los dos métodos. En los ensambles de pronósticos, la varianza se estima a partir de la distancia entre los miembros (que son posibles realizaciones del estado del sistema a un dado tiempo) y la media del ensamble. Los miembros del ensamble se generan bajo

ciertas suposiciones respecto de los errores presentes en las condiciones iniciales del pronóstico y de los errores de modelo. La calidad de la estimación está condicionada por el tamaño del ensamble y por la validez de dichas suposiciones. Por otra parte, al usar una secuencia histórica de estados posibles, los errores del pronóstico se estiman a partir de la diferencia del estado pronosticado y de una estimación del estado verdadero del sistema (ej. un análisis). La estimación de la varianza surge de considerar dichas diferencias en una determinada región del espacio de fase, que esquemáticamente está representada por la banda naranja en la Fig. 3.5. En este caso, la calidad de la estimación está condicionada por la cantidad de trayectorias que atraviesan esa región del atractor que su vez depende de la cantidad de datos históricos disponibles. También depende de qué tan buena sea nuestra estimación del verdadero estado del sistema. Por este motivo, las varianzas estimadas al entrenar directamente del ensamble no son necesariamente las mismas que las obtenidas al entrenar con datos históricos. En la Fig. 3.5 podemos ver que para las redes que se entrenen con la varianza del ensamble, es esperable que aproximen a $\hat{\sigma}^2$ al ser evaluadas para el estado $\bar{x}$ ya que así fueron entrenadas. Mientras que para el mismo estado, las redes entrenadas para minimizar una función de costo sobre el conjunto de datos históricos se espera que estimen $\tilde{\sigma}^2$.

3.4.1. Estimación basada en el error de pronóstico

Así como está planteada, la figura 3.5 sugiere la posibilidad de entrenar de forma directa usando como objetivo de entrenamiento $\tilde{\sigma}^2$, en vez de usar $\hat{\sigma}^2$. Es decir, entrenar las varianzas usando como objetivo de entrenamiento directamente el error cuadrático. Esta definición es similar a la propuesta en Scher and Messori (2018) en el sentido que se utilizan estimaciones locales del error para inferir la varianza del error, solo que en vez de estimar un escalar para representar la incertidumbre asociada al vector de estado como en Scher and Messori (2018), se podría estimar el vector completo de varianzas asociado al estado.

Definir el error en sistemas dinámicos complejos no es sencillo ya que usualmente se desconoce el estado verdadero del sistema y solo se dispone de una aproximación. Como se discutió en la sección 2.2.1 los estados analizados son la mejor aproximación al estado verdadero que podemos disponer, ya que combinan pronóstico y observación logrando filtrar parte del error sistemático. Supongamos entonces que disponemos para el entrenamiento de los estados analizados medios $\bar{x}^a$ y queremos diseñar una función de costo que nos permita generar un pronóstico probabilístico a partir de un pronóstico determinístico x^d . Llamamos a esta función de costo función de costo basada en el error,

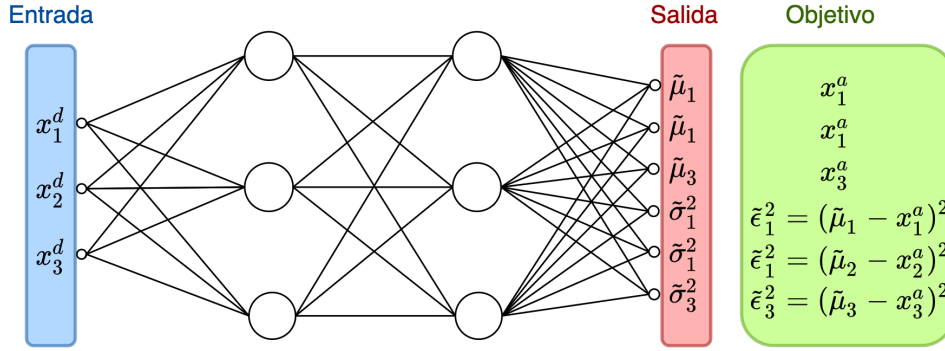


Figura 3.6: Esquema de una RNA donde se muestran los datos de entrenamiento usando la función eMSE. La entrada de la red es el vector de estado pronosticado $\mathbf{x}^d$ (en azul), y la salida son los vectores de media $\tilde{\boldsymbol{\mu}}$ y varianza $\tilde{\boldsymbol{\sigma}}^2$ indicados en color rojo. Cada elemento en la salida de la red es comparado con un valor objetivo (en verde) calculado por la función eMSE a partir del análisis $\mathbf{x}^a$ del conjunto de entrenamiento.

en inglés extended Mean Squared Error (eMSE), ya que utiliza la métrica MSE (función de costo basada en el error cuadrático medio, en inglés Mean Squared Error) en combinación con un objetivo basado en una estimación local del momento estadístico que se busca estimar.

Así, dado un estado $\mathbf{x}^d$ y suponiendo que la distribución de los posibles estados futuros del sistema es gaussiana, la RNA provee una estimación de la PDF de dichos estados a través de los momentos de primer y segundo orden ($\tilde{\boldsymbol{\mu}}; \tilde{\boldsymbol{\sigma}}^2$) como lo indica la figura 3.6. El error cometido por la red puede ser calculado de la siguiente manera:

$$\tilde{\boldsymbol{\epsilon}} = \bar{\mathbf{x}}^a - \tilde{\boldsymbol{\mu}} \quad (3.14)$$

donde $\tilde{\boldsymbol{\epsilon}}$ es el error cometido por la red y lo usaremos para definir el primer término (T_1) de nuestra función de costo eMSE:

$$T_1 = \tilde{\boldsymbol{\epsilon}}^\top \tilde{\boldsymbol{\epsilon}} = \text{MSE}(\mathbf{x}^a, \tilde{\boldsymbol{\mu}}) \quad (3.15)$$

A continuación, el error al cuadrado $\tilde{\boldsymbol{\epsilon}}^2$ se utiliza como estimación local de la varianza (ver Fig. 3.6) y llamaremos a este término T_2 :

$$T_2 = (\tilde{\boldsymbol{\sigma}}^2 - \tilde{\boldsymbol{\epsilon}}^2)^\top (\tilde{\boldsymbol{\sigma}}^2 - \tilde{\boldsymbol{\epsilon}}^2) = \text{MSE}(\tilde{\boldsymbol{\sigma}}^2, \tilde{\boldsymbol{\epsilon}}^2) \quad (3.16)$$

donde los cuadrados denotan productos elemento a elemento o producto de Hadamard, es decir, $\tilde{\epsilon}_l^2 \doteq \tilde{\epsilon}_l \odot \tilde{\epsilon}_l$. La función de costo queda definida entonces por la suma de cada término ponderada por los hiperparámetros α_n :

$$\mathcal{L}_{eMSE} = \alpha_1 T_1 + \alpha_2 T_2. \quad (3.17)$$

Es interesante notar que la elección de los coeficientes α no solo nos permite dar más relevancia a un momento por sobre otro, sino que también nos permite elegir un único momento al hacer alguno de los coeficientes igual a cero. Esta última propiedad será utilizada en el próximo capítulo donde se entrenarán redes separadas, una que estima sólo la media y otra que estima sólo varianza.

3.5. Análisis comparativo

La función de costo LKHD fue muy estudiada y usada en varios contextos como vimos en este capítulo, por eso, la utilizaremos como base de comparación para evaluar la función de costo eMSE propuesta en este trabajo.

3.5.1. Análisis de convergencia

Dos aspectos importantes para estudiar al momento de analizar una función de costo son el valor mínimo de la función, porque indica efectivamente lo que se aprende, y los gradientes en la curva de nivel, porque nos brinda información útil de cómo será la convergencia y las dificultades que podremos encontrar durante el entrenamiento.

Primero analizaremos la función de costo eMSE y, en particular, la sensibilidad a los parámetros α de su formulación (Ec. 3.16) y luego la compararemos con la función LKHD. La figura 3.7 muestra el valor medio (intensidad de color) de la función de costo evaluada sobre una muestra de puntos en función de los parámetros μ y σ^2 . Para realizar este gráfico, se genera una muestra $\mathcal{D}$ de 10000 elementos obtenida a partir de una distribución de probabilidad dada. Se eligió evaluar cuatro muestras de distribuciones de características diferentes. La distribución gaussiana (primer columna Fig. 3.7) se eligió porque las funciones de costo que evaluaremos asumen que la muestra es gaussiana. La función LKHD lo asume de forma explícita en su formulación y la función eMSE, al estimar sólo dos momentos de la distribución, lo asume implícitamente. La segunda columna de la figura 3.7 se construyó en base a una muestra generada a partir de una distribución gamma que, a diferencia de la gaussiana, es asimétrica y la moda es distinta de la media.

La distribución uniforme (tercera columna) se eligió porque es simétrica pero no tiene una moda definida. Y la última columna es una combinación de dos gaussianas que da como resultado una distribución multimodal.

Al interpretar la figura 3.7 en un contexto de minimización de la función de costo, los paneles representan superficies de convergencia hacia el valor mínimo. Lo primero que podemos observar es que no todos los mínimos coinciden con los valores muestrales $\hat{\mu}_D$ y $\hat{\sigma}_D$ de la primera fila. Sobre todo para las distribuciones no gaussianas y en especial para valores de α_2 altos. En particular, para $\alpha_2 = 1$ y las distribuciones gamma y binormal (segunda y cuarta columna respectivamente) el punto mínimo $(\mu_{min}, \sigma_{min})$ de la función de costo es muy diferente de los valores muestrales. Esto coincide con los resultados de [Camporeale and Carè \(2021\)](#) y con los experimentos de [Nix and Weigend \(1994\)](#) y [Bishop and Qazaz \(1997\)](#) que recomendaban no optimizar ambos parámetros (μ y σ^2) en simultáneo. A medida que decrementamos el valor de α_2 , y en consecuencia debilitamos la interrelación entre μ y σ^2 , los valores mínimos coinciden con los muestrales. Para $\alpha_2 = 0,01$ se puede ver que las curvas de nivel muestran un muy alto gradiente en μ (casi independiente de σ^2) y un gradiente muy suave en σ^2 (casi independiente de μ).

En la figura 3.8 se comparan las curvas de nivel de LKHD y de eMSE. Los mínimos alcanzados por ambas funciones de costo se dan, prácticamente, para el mismo par $(\mu_{min}, \sigma_{min}^2)$ en todas las distribuciones evaluadas. Pero la forma de la superficie de convergencia es distinta. La función LKHD (paneles verdes) muestra valores muy altos y con un gradiente muy pronunciado para valores de σ^2 cercanos a cero. Para valores alejados de cero, los gradientes son más suaves y están relativamente balanceados entre las direcciones de μ y σ^2 en comparación con eMSE (paneles rojos). Se puede considerar que valores de σ^2 muy cercanos a cero son conceptualmente un error, y en consecuencia, una función que penalice fuertemente estos valores parece deseable. En este sentido, la función LKHD parece tener una ventaja sobre eMSE que no penaliza valores de σ^2 cercanos o iguales a cero. Pero, por otro lado, la inversión de la varianza presente en LKHD, puede inestabilizar numéricamente el algoritmo de minimización dificultando el entrenamiento sobre todo en casos de sobreentrenamiento [Nix and Weigend \(1994\)](#)

3.5.2. Ejemplo de estimación de la incertidumbre usando redes neuronales

En la sección anterior, se compararon y analizaron algunas propiedades de las funciones LKHD y eMSE. Ahora las utilizaremos para entrenar sendas RNAs que estimen la

probabilidad de y dada por:

$$P(y|x) \sim \mathcal{N}(\mu_y(x), \sigma_y^2(x)) \quad (3.18)$$

con

$$\begin{aligned} \mu_y(x) &= x^2 \\ \sigma_y^2(x) &= \sin(2\pi x^2 + 10^{-3}) \end{aligned} \quad (3.19)$$

Para el entrenamiento se generaron 20.000 muestras de x a partir de una distribución uniforme ($\mathcal{U}(0, 1; 1)$) para construir los pares (x, y) según la ecuación 3.18. Del total de muestras se separaron 500 para validación y el resto se utilizó para entrenar las RNA. La RNAs utilizada es un perceptron multicapa de 1 neurona en la capa de entrada, 2 capas ocultas con 30 neuronas cada una y 2 neuronas en la capa de salida (una para estimar μ y otra para σ^2). Todas las capas tienen una función de activación tanh salvo la capa de salida que usa una función de activación lineal.

Los resultados en la figura 3.9 muestran que ambas metodologías logran estimar muy bien la variabilidad de la incertidumbre en este ejemplo sencillo. También se puede observar una mayor tendencia de la función eMSE a sobrestimar los valores de varianza cuando estos son cercanos a cero. Este es un punto que seguiremos analizando a lo largo de los siguientes capítulos.

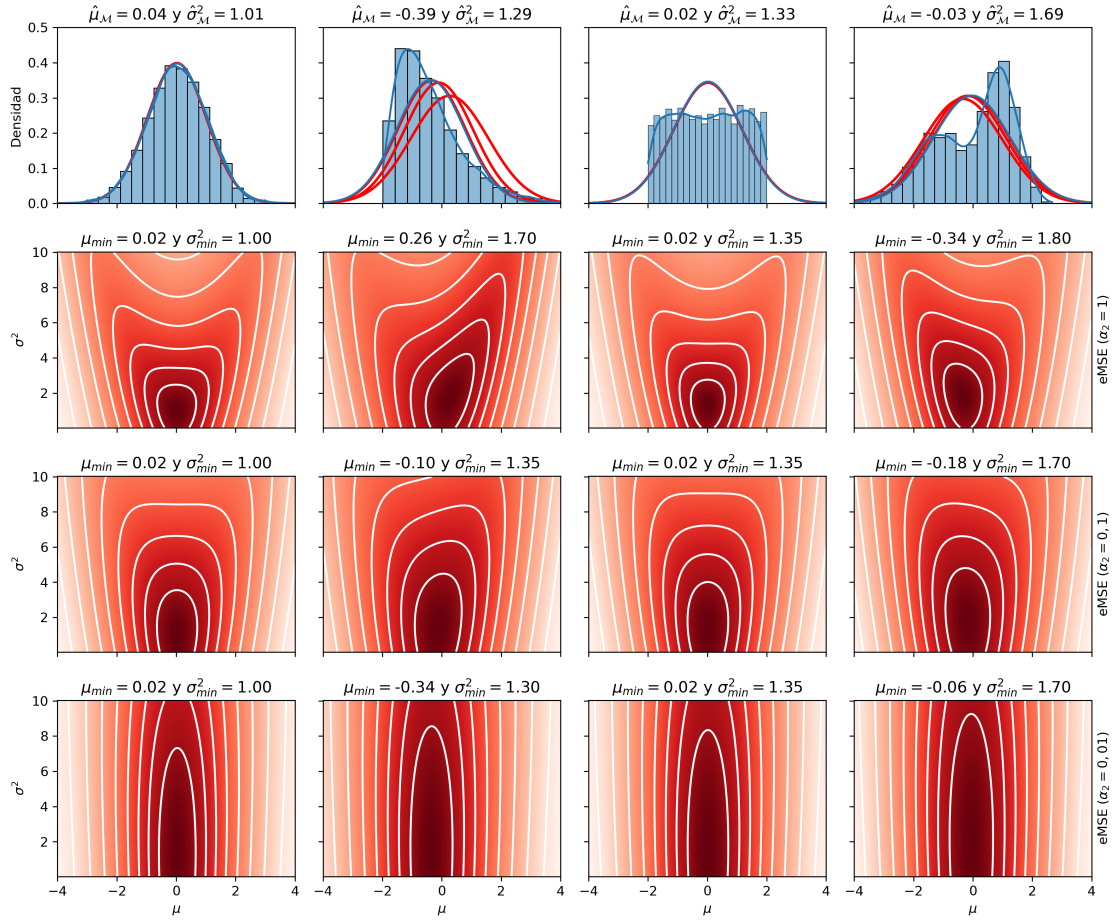


Figura 3.7: Los paneles muestran el valor medio de la función de costo evaluada sobre todos los elementos de cada muestra en función de μ (eje de abscisas) y σ^2 (eje de ordenadas). La fila superior muestra los histogramas (línea y barras azules) correspondientes a las 4 muestras que se usaron para evaluar la función eMSE (paneles de las filas dos a cuatro). Se evaluaron tres valores distintos para el coeficiente α_2 (ver Ec. 3.17) indicados a la derecha de cada fila. Se usó el coeficiente $\alpha_1 = 1$ en todos los casos. Los valores $\hat{\mu}$ y $\hat{\sigma}^2$ en la primer fila corresponden a la media y varianza muestrales respectivamente. Los valores μ_{min} y σ^2_{min} que figuran en el título del resto de los paneles representan los valores para los cuales la función de costo alcanzó el mínimo. En las filas dos a cuatro, la intensidad del color determina el valor de la función de costo, los colores oscuros representan valores bajos y los claros valores altos. Junto con los histogramas (fila superior) están dibujadas la densidad de probabilidad correspondiente a la gaussiana con μ_{min} y σ^2_{min} de las funciones de costo evaluadas (curvas rojas).

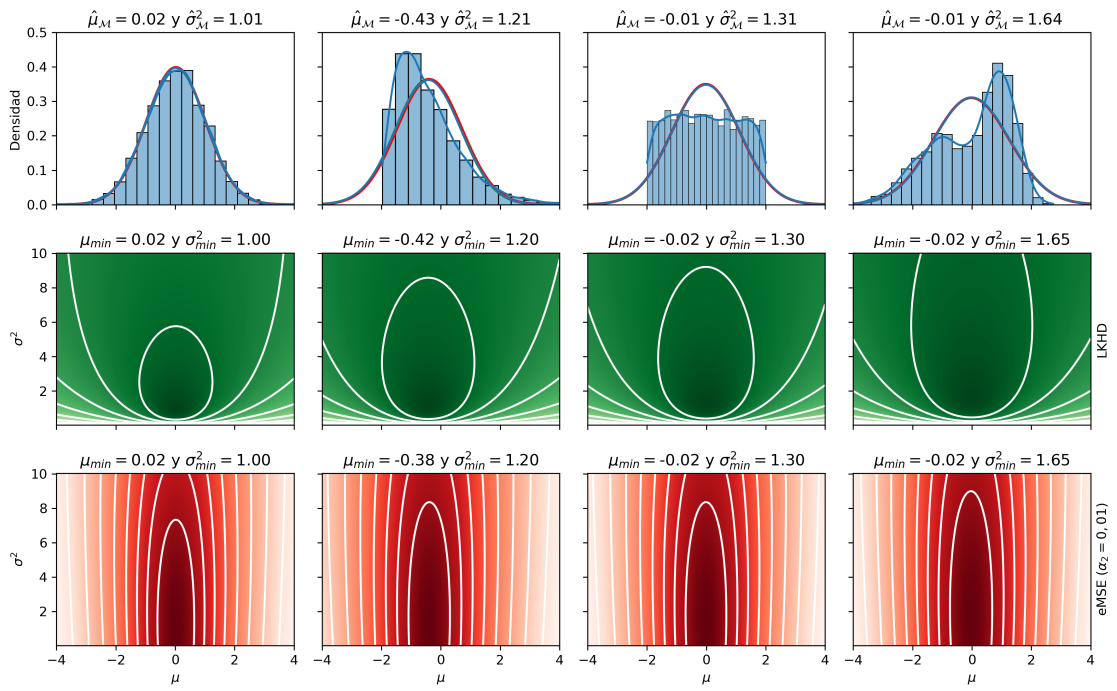


Figura 3.8: Idem figura 3.7, pero en este caso, se comparan las funciones de costo LKHD en verde y eMSE en rojo.

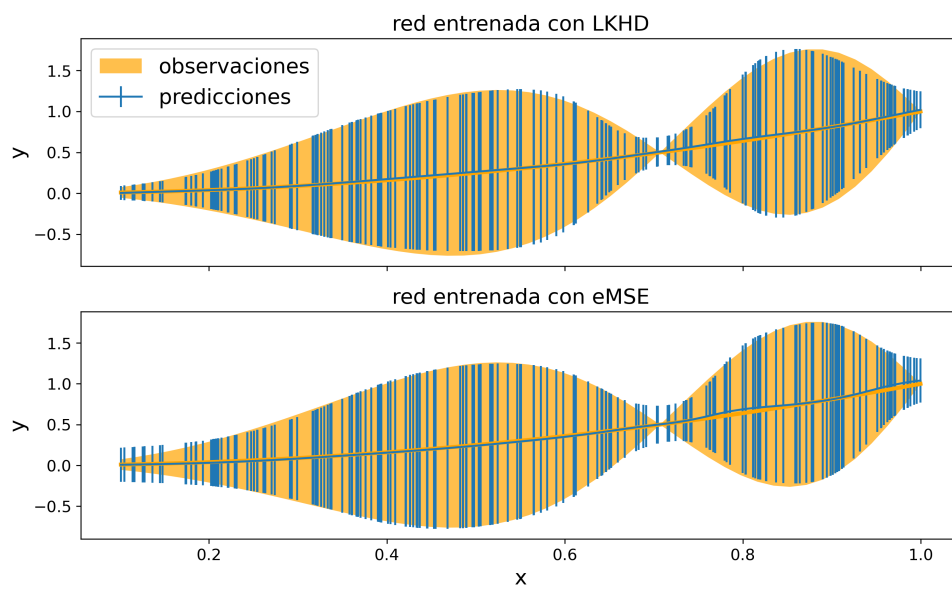


Figura 3.9: Predicción de la probabilidad de y (en azul) dada por RNAs entrenadas con la función de costo LKHD (arriba) y eMSE (abajo). El sombreado naranja representa la varianza real $\sigma_y^2(x)$ del conjunto de datos.

Capítulo 4

Pronósticos Probabilísticos

Los pronósticos por ensambles son, hasta ahora, el enfoque más exitoso para producir pronósticos a diferentes plazos, junto con una estimación de su incertidumbre. Las principales limitaciones del pronóstico por ensamble son el elevado coste computacional y la dificultad para captar y cuantificar las diferentes fuentes de incertidumbre, en particular aquellas asociadas a los errores del modelo.

El objetivo de este capítulo es analizar en qué medida una red neuronal es capaz de identificar las diferentes fuentes de incertidumbre presentes en un pronóstico, en particular las asociadas a las condiciones iniciales y las introducidas por el error del modelo. Para lograr este objetivo, se realizaron experimentos con un modelo dinámico sencillo en un escenario de modelo perfecto, en el que la fuente de incertidumbre sólo es introducida por las condiciones iniciales, y en un escenario de modelo imperfecto en el que el error del modelo contribuye también a la incertidumbre en los pronósticos. Estos experimentos se repitieron para diferentes plazos de pronóstico y se compararon tres técnicas diferentes de entrenamiento para estimar la incertidumbre mediante redes neuronales. Además, se llevaron a cabo experimentos con datos reales como prueba de concepto para evaluar el rendimiento de las metodologías utilizando datos de un modelo numérico meteorológico de última generación.

4.1. Diseño experimental

Se realizarán experimentos utilizando el sistema dinámico caótico simple descrito en [Lorenz \(1995\)](#) para evaluar el rendimiento de las redes neuronales artificiales (RNAs). Se consideran dos escenarios diferentes, un escenario en el que el modelo utilizado para

estimar el estado del sistema y producir pronósticos es una representación perfecta de la dinámica del sistema (es decir, $\eta_k = 0$ en la Ec. 2.1) y otro escenario en el que se usa un modelo aproximado (o subrogado) en donde parte de la dinámica del sistema no está representada y, por tanto, los pronósticos se ven afectados por los errores del modelo.

4.1.1. Escenario de modelo perfecto

En el escenario de modelo perfecto (EMP) se utiliza la versión de una sola escala del sistema dinámico de Lorenz (Lorenz, 1995) dado por la siguiente ecuación

$$\frac{dx_{(i)}}{dt} = -x_{(i-2)}x_{(i-1)} + x_{(i-1)}x_{(i+1)} - x_{(i)} + F, \quad (4.1)$$

donde $x_{(i)}$ es la componente i -ésima del vector de estados $\mathbf{x}$ y F es una constante que representa un forzante externo que determina cuán caótico será el sistema. Se imponen condiciones de contorno cíclicas para que este sistema de ecuaciones se asemeje a un sistema forzado advectivo-disipativo sobre un círculo de latitud.

En primer lugar, se realiza una larga secuencia de integración del modelo, que representa la verdadera evolución del sistema la cual se asume desconocida. Llamaremos a esta secuencia de estados: *estados verdaderos* o *verdad*. El modelo se integra durante 500 unidades de tiempo utilizando el método Runge-Kutta de 4º orden, con un forzante $F = 8$, un paso de tiempo de 0,0125 unidades de tiempo y $N_x = 8$ variables de estado. En segundo lugar, se genera un conjunto de observaciones a partir de los estados verdaderos añadiendo ruido gaussiano no correlacionado en espacio ni en tiempo, con media cero y covarianza $\mathbf{R} = \sigma_R^2 \mathbf{I}$ y con $\sigma_R = 1$. Todas las variables de estado se observan cada 4 pasos de tiempo (0,05 unidades de tiempo).

Las condiciones iniciales para los pronósticos numéricos se generan utilizando un sistema de asimilación de datos basado en ensambles (the Local ensemble Transform Kalman Filter, LETKF; Hunt et al., 2007), que permite obtener una estimación del estado del sistema a intervalos regulares de manera secuencial (es decir, en este caso cada 0,05 unidades de tiempo que es cuando hay observaciones disponibles). Se utiliza un ensamble de 50 miembros para generar los análisis, ya que aumentar la cantidad de miembros no resulta en una mejora considerable en la calidad del análisis. La asimilación de datos se lleva a cabo para toda la longitud de la integración *verdad* dando un total de 13000 estimaciones del estado del sistema y su incertidumbre asociada. Estos análisis serán utilizados para entrenar y verificar las redes neuronales de la siguiente manera: 7000 se utilizan para el conjunto de entrenamiento, 3000 para la validación y 3000 para evaluación. Por último,

se generan dos conjuntos de pronósticos: un pronóstico por ensamble en el que cada miembro se inicializa a partir del análisis del miembro correspondiente y un pronóstico determinístico inicializado a partir del análisis medio, ambos serán usados durante el entrenamiento. Los pronósticos se integran hasta un plazo máximo de 3,5 unidades de tiempo (280 pasos de tiempo) que es el plazo donde el error de pronóstico está cerca de su saturación no lineal.

Para todos los experimentos, las trayectorias reales (*estados verdaderos*) se suponen desconocidos asemejando la información que estaría disponible en experimentos con datos meteorológicos reales.

4.1.2. Escenario de modelo imperfecto

Para la generación de la integración *verdad* en el escenario de modelo imperfecto (EMI), utilizamos el modelo de Lorenz de dos escalas (Lorenz, 1995) que está definido por el siguiente sistema de ecuaciones diferenciales acopladas:

$$\frac{dx_{(i)}}{dt} = -x_{(i-2)}x_{(i-1)} + x_{(i-1)}x_{(i+1)} - x_{(i)} + F - \frac{hc}{b} \sum_{j=J(i-1)+1}^{iJ} y_{(j)} \quad (4.2)$$

$$\frac{dy_{(j)}}{dt} = -cb y_{(j+1)}y_{(j+2)} + cb y_{(j+1)}y_{(j-1)} - c y_{(j)} + \frac{hc}{b} x_{(\text{int}[(j-1)/J]+1)}, \quad (4.3)$$

donde $x_{(i)}$ son las variables de estado asociadas a la dinámica lenta, mientras que $y_{(j)}$ son las variables asociadas a la dinámica más rápida. J es el número de variables y para cada variable x , y h , c , y b son parámetros independientes del tiempo que controlan la fuerza de acoplamiento entre los dos sistemas. Se aplican condiciones de contorno cíclicas a ambos conjuntos de ecuaciones, a saber, $x_{(1)} = x_{(S+1)}$, y $y_{(1)} = y_{(J \cdot S+1)}$. Este sistema de dos escalas es una representación sencilla de sistemas con múltiples escalas espacio-temporales como la atmósfera o el océano. En nuestros experimentos, $J = 32$ y $N_x = 8$ (es decir, el vector y tiene un total de 256 variables) y el forzante F se fija en 20 para obtener un comportamiento caótico. El modelo subrogado utilizado para la asimilación de datos y el pronóstico viene dado por:

$$\frac{dx_{(i)}}{dt} = -x_{(i-1)}(x_{(i-2)} - x_{(i+1)}) - x_{(i)} + F + G_{(i)}, \quad (4.4)$$

donde $G_{(i)}$ es un término de parametrización dependiente del estado que aproxima el efecto de la dinámica no representada (es decir, el efecto de las variables rápidas y). Al igual que en [Pulido et al. \(2016\)](#), se supone que $G_{(i)}$ es una función lineal de la variable de estado $x_{(i)}$

$$G_{(i)} = \alpha x_{(i)} + \beta, \quad (4.5)$$

con $\alpha = 19,16$ y $\beta = -0,81$ parámetros constantes cuyos valores óptimos se toman de [Scheffler et al. \(2019\)](#). El uso de un modelo subrogado que no considera la dinámica a pequeña escala permite considerar los efectos del error del modelo en los pronósticos. La configuración experimental general, incluyendo la frecuencia de observación, el ruido de observación y la duración del experimento de asimilación, es la misma que en el EMP. Los pronósticos por ensamble generados en el escenario de modelo imperfecto no incluyen una representación explícita de los errores del modelo (es decir, $\hat{\delta}_{l,0}^{(n)} = 0$ en la Ec. 2.3).

4.2. Metodología

Para estimar la componente sistemática del error del pronóstico y la varianza del error del pronóstico utilizando técnicas de aprendizaje automático, se considera un modelo que consiste en dos RNAs que proporcionan el primer y el segundo momento de la distribución de probabilidad de los posibles estados del sistema en el plazo l . Dicha distribución de probabilidad se supone gaussiana. La entrada del modelo (de las dos RNAs) consiste en un pronóstico determinístico del estado del sistema en diferentes plazos ($\{\mathbf{x}_0^d, \dots, \mathbf{x}_l^d, \dots, \mathbf{x}_L^d\}$, $0 \leq l \leq L$ ver Fig. 4.1), que da información sobre la evolución dinámica del sistema a lo largo de una ventana temporal en torno al plazo l . Nuestro objetivo es estimar la media de la distribución de estados al tiempo l ($\tilde{\mathbf{x}}_l$) y su varianza ($\tilde{\sigma}^2$), ambos dependientes del estado del sistema. Obsérvese que tanto $\tilde{\mathbf{x}}_l$ como $\tilde{\sigma}^2$ son vectores N_x -dimensionales.

Los pronósticos de $\tilde{\mathbf{x}}_l$ son necesarios para estimar el ruido heterocedástico del pronóstico $\tilde{\sigma}_l^2(\mathbf{x}_l)$, y al mismo tiempo el ruido heterocedástico afecta la estimación de $\tilde{\mathbf{x}}_l$ modificando la tasa de aprendizaje (ver Eq. 3.4 Sección 3). Por lo tanto, en general, existe un acoplamiento bidireccional entre las dos cantidades. No obstante, como se discutió en el capítulo 3, los trabajos [Nix and Weigend \(1994\)](#); [Bishop and Qazaz \(1997\)](#); [Camporeale and Carè \(2021\)](#) recomiendan desacoplar el entrenamiento de las redes que estiman la media y la varianza de la distribución. En esta tesis, se entrena primero la red que estima

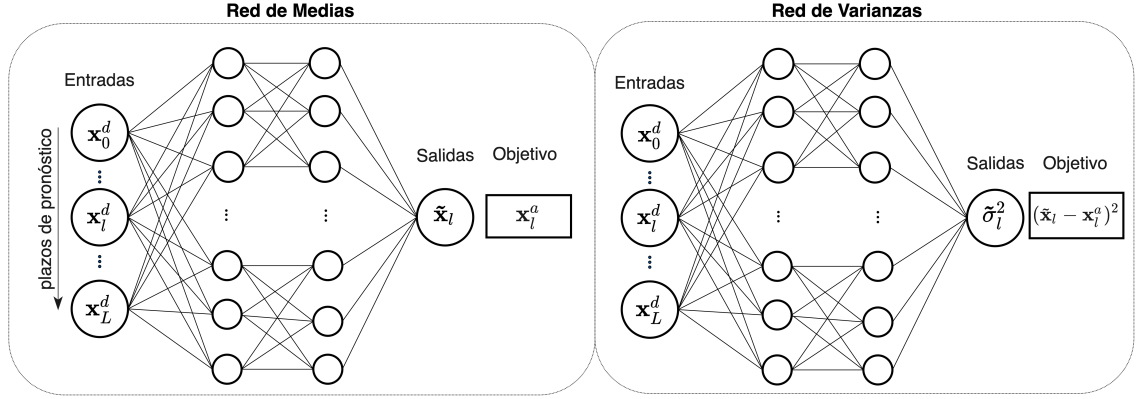


Figura 4.1: Esquema de la arquitectura de la red que estima el estado medio del sistema (izquierda) y de la red que estima la varianza del error de pronóstico (derecha). En primer lugar, la red que proporciona el estado medio se entrena utilizando los análisis como objetivo. A continuación, la red que estima las varianzas se entrena utilizando como objetivo de entrenamiento la diferencia entre el estado medio pronosticado y el análisis, es decir, un proxy del error del pronóstico.

el estado medio, asumiendo ruido homocedástico, es decir, que $\tilde{\sigma}^2$ no afecta a la estimación de $\tilde{\mathbf{x}}_l$. Una vez obtenida la convergencia, se entrena la red que estima $\tilde{\sigma}^2$ usando las estimaciones del estado medio proporcionadas por la primera red. Observemos que la suposición de ruido homocedástico es sólo para la regresión del estado medio, para la regresión de la varianza el ruido se supone heterocedástico.

4.2.1. Entrenamiento de la red de estados medios

En primer lugar se entrena la red de estados medios, como se muestra en el panel izquierdo de la Figura 4.1. Para ello, utilizamos los estados analizados ($\mathbf{x}_l^a$), N_x -dimensionales, válidos al mismo tiempo del pronóstico ($\tilde{\mathbf{x}}_l$) como objetivo de entrenamiento y se usó la función de costo basada en el error cuadrático medio, en inglés Mean Squared Error (MSE).

$$\mathcal{L}_{MSE}^x = (\tilde{\mathbf{x}}_l - \mathbf{x}_l^a)^\top (\tilde{\mathbf{x}}_l - \mathbf{x}_l^a). \quad (4.6)$$

Los análisis se utilizan como objetivo de entrenamiento en lugar de las observaciones disponibles porque proporcionan una estimación completa del vector de estado en la retícula del modelo en tiempos regulares y con un error que es estadísticamente menor

que el error observacional (Carrassi et al., 2018). Este esquema proporciona directamente una estimación del estado medio de forma supervisada.

4.2.2. Entrenamiento de la red de varianza

A continuación, definiremos el entrenamiento de las redes neuronales que estiman la varianza suponiendo que los estados medios $\tilde{x}_l$ ya fueron entrenados y están disponibles. Para el resto de las definiciones, supondremos que los estados medios $\tilde{x}_l$, los análisis $\tilde{x}_l^a$ y las varianzas $\tilde{\sigma}_l^2$ son siempre válidas al mismo tiempo y, en consecuencia, suprimiremos el uso del subíndice l para las siguientes definiciones.

Entrenamiento a partir de un ensemble de pronósticos (RNA-mse)

En esta estrategia, seguimos un enfoque similar al de Scher and Messori (2018) discutido en el Cap. 3, en donde la red que estima $\tilde{\sigma}^2$ se entrena utilizando como objetivo σ^{e2} (es decir, la dispersión de un ensemble de pronósticos válidos para el mismo plazo). Como en el caso de la red que estima el estado medio, se supone que $\tilde{\sigma}^2$ y σ^{e2} son de dimensión N_x .

El entrenamiento de la red se realiza utilizando la función MSE :

$$\mathcal{L}_{MSE}^{\sigma} = (\tilde{\sigma}^2 - \sigma^{e2})^{\top} (\tilde{\sigma}^2 - \sigma^{e2}). \quad (4.7)$$

El principal inconveniente de un entrenamiento directo es que requiere una cuantificación a priori y explícita de la incertidumbre, que en este caso está dada por la dispersión del ensemble σ^{e2} . Por este motivo, necesita una integración completa del ensemble durante el período de entrenamiento de la red. Por otra parte, la red así entrenada, heredará las mismas características del pronóstico por ensambles en términos de la capacidad para cuantificar las diferentes fuentes de errores.

Una diferencia respecto de lo implementado por Scher and Messori (2018) es que la estimación de la varianza se realiza individualmente para cada una de las variables del sistema, y no integrando sobre un conjunto de variables. Esto implica que la red debe ser capaz de reproducir adecuadamente la variabilidad espacial de la varianza.

Entrenamiento basado en la verosimilitud gaussiana (RNA-lik)

Esta técnica de entrenamiento es la propuesta en [Nix and Weigend \(1994\)](#) y utilizada en [Wang et al. \(2018\)](#) también discutida y definida en el capítulo 3 por la ecuación 3.3.

$$\mathcal{L}_{LKHD}^{\sigma} = \sum_i \frac{1}{2} \left[\frac{(x_i^a - \tilde{x}_i)^2}{\tilde{\sigma}_i^2} + N_x \ln \tilde{\sigma}_i^2 \right] \quad (4.8)$$

Al introducir esta función de costo, estamos asumiendo la independencia de los errores en las diferentes variables y tiempos del pronóstico. Además, no se necesita explícitamente la incertidumbre del análisis, como en el caso $\mathcal{L}_{MSE}^{\sigma}$ definida anteriormente, sólo se necesitan los análisis medios.

Entrenamiento basado en el error del pronóstico (RNA-ext)

Esta es nuestra propuesta de entrenamiento introducida y analizada en el capítulo 3, en la sección 3.4.1. En este caso sólo se quiere entrenar la red de varianzas, entonces todos los coeficientes α serán igual a cero, salvo $\alpha_2 = 1$ de la ecuación 3.17. Entonces la función queda definida de la siguiente forma:

$$\mathcal{L}_{eMSE}^{\sigma} = (\tilde{\sigma}^2 - \tilde{\epsilon}^2)^{\top} (\tilde{\sigma}^2 - \tilde{\epsilon}^2) = \text{MSE}(\tilde{\sigma}^2, \tilde{\epsilon}^2) \quad (4.9)$$

donde $\tilde{\epsilon} = x^a - \tilde{x}$.

Al igual que la $\mathcal{L}_{LKHD}^{\sigma}$, esta función de costo permite inferir la incertidumbre indirectamente a partir de un conjunto de estimaciones de los errores del pronóstico para diferentes estados del sistema y, por tanto, no es necesario contar con una estimación a priori de la incertidumbre del pronóstico.

Obsérvese que la función de costo $\mathcal{L}_{MSE}^x$ utilizada para entrenar la red de estados medios puede pensarse como la ecuación 3.17 con $\alpha_1 = 1$ y $\alpha_2 = 0$.

En esta sección, hemos definido diferentes funciones de costo considerando una única muestra del conjunto de datos. Sin embargo, hay que tener en cuenta que la optimización de las RNAs tiene como objetivo minimizar la función de costo de todo el conjunto de datos, que viene dada por la suma sobre todas las muestras del conjunto de datos de entrenamiento.

4.2.3. Arquitectura y entrenamiento de la red

Se eligió una arquitectura totalmente conectada para todos los experimentos de este capítulo porque es una arquitectura genérica, lo que significa que no hace ninguna suposición sobre los datos, y porque la baja dimensionalidad del problema que estamos analizando lo permite. Para problemas de mayor dimensionalidad, se pueden utilizar otras arquitecturas (por ejemplo, redes convolucionales) para disminuir el número de parámetros libres de la red. Para cada experimento, ambas redes (de estado medio y de varianza) tienen dos capas ocultas con 50 neuronas cada una y funciones de activación softplus (Dugas et al., 2001). La red de estado medio debe generar salidas dentro del rango de valores posibles para las variables de estado, por lo que utilizamos una función de activación lineal en la capa de salida. Para la red de varianza, utilizamos una función de activación softplus en la capa de salida, así garantizamos que las varianzas sean positivas.

La arquitectura de la red se definió a partir de experimentos preliminares que consintieron en variar el tamaño de las capas ocultas (de 16 a 200 neuronas) y las funciones de activación de las capas ocultas (sigmoide, ReLU y softplus). La arquitectura elegida fue la que presentaba el menor valor medio de la función de costo en todos los experimentos. Todas las redes se entrenaron con el optimizador Adam (Kingma and Ba, 2017) y utilizando mini-lotes con un tamaño de lote de 50 elementos. La función de costo se evalúa sobre el conjunto de validación cada 20 épocas de entrenamiento. El entrenamiento se detiene cuando la función de costo evaluada sobre el conjunto de validación deja de disminuir o comienza a aumentar (detención temprana del entrenamiento).

En experimentos preliminares se observó una gran sensibilidad del entrenamiento a la tasa de aprendizaje, en inglés Learning Rate (η_{lr}) y decaimiento de pesos, en inglés Weight Decay (WD), por lo que se realizó una exploración por fuerza bruta del espacio de dichos hiperparámetros utilizando $LR=1e^{-2}, 1e^{-3}, 1e^{-4}, 1e^{-5}, 1e^{-6}$ y $WD=0, 5e^{-1}, 1e^{-1}, 1e^{-2}$. El entrenamiento de las RNAs se repitió 5 veces para cada combinación de LR y WD para filtrar el ruido asociado a la inicialización estocástica de los pesos de la red. Los hiperparámetros elegidos son los que produjeron el menor valor de la función de costo evaluado sobre el conjunto de validación y promediado sobre las 5 repeticiones del entrenamiento.

Para la evaluación del rendimiento de las RNAs en los experimentos, se tomaron como representativos tres plazos de pronóstico diferentes de 4, 80 y 160 pasos de tiempo. La figura 4.2 muestra las evoluciones del error del pronóstico determinístico y por ensamble para escenarios EMP y EMI en función del plazo de pronóstico. Los resultados están promediados sobre un gran número de inicializaciones. Se espera que la comparación

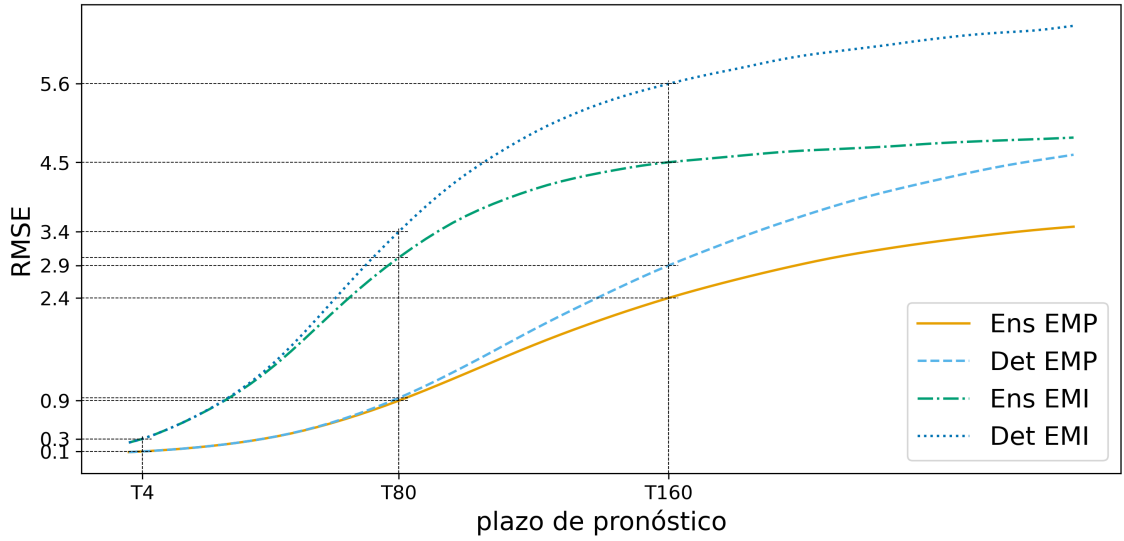


Figura 4.2: RMSE como función del plazo de pronóstico en los EMP y EMI utilizando la evolución de una sola integración del modelo (Det) y la media de un ensemble de 50 integraciones del modelo (Ens).

del rendimiento en diferentes plazos de pronóstico arroje algo de luz sobre el impacto de las incertidumbres de las condiciones iniciales y del error del modelo en la incertidumbre pronosticada por la red neuronal artificial (RNA).

El primer plazo (T4) corresponde a un pronóstico de muy corto plazo (4 pasos de tiempo), como los que se utilizan habitualmente en un ciclo de asimilación de datos. Para este plazo, el error de pronóstico es comparable al error presente en los análisis utilizados como objetivo de entrenamiento para la RNA y, por tanto, resulta interesante evaluar en qué medida la red es capaz de cuantificar adecuadamente la incertidumbre en este contexto. Por otra parte, y como se analizará en los capítulos posteriores, la cuantificación de la incertidumbre en estos plazos es fundamental para la asimilación de datos. El segundo plazo (T80) se encuentra dentro del régimen de crecimiento lineal del error, especialmente en el experimento del modelo perfecto (véase la Fig. 4.2). Para este plazo, la diferencia entre el error del pronóstico determinístico y el del ensemble es pequeña en el EMP, pero es importante en el EMI. A diferencia de T4, el error inicial propagado a través de la dinámica del sistema en T80 ha crecido de forma tal que es significativamente más grande que los errores presentes en la condición inicial. El tercer plazo (T160) está dentro del régimen de crecimiento no lineal del error y también está cerca de la saturación del error en el EMI. En este plazo, el error inicial ha evolucionado a través de la dinámica del sis-

tema y ha crecido lo suficiente como para que los efectos no lineales sean importantes, esto se evidencia en la disminución de la tasa de crecimiento del error. Por lo tanto, el sistema se encuentra en el límite de la predictibilidad, tanto en el escenario EMP como EMI.

Para cada escenario (EMP y EMI) y para cada plazo de tiempo seleccionado (T4, T80, T160), realizamos un pronóstico determinístico, un pronóstico por ensamble (línea de base para la comparación) y tres modelos basados en RNA obtenidos utilizando las tres estrategias de entrenamiento descritas en la sección 4.2 (RNA-mse, RNA-ext y RNA-lik).

4.3. Métricas de verificación

Para comparar diferentes aspectos del rendimiento de los distintos sistemas de pronóstico (determinístico, ensamble y RNA con diferentes estrategias de entrenamiento) utilizamos cuatro métricas: el error cuadrático medio, la probabilidad de cobertura, la correlación entre la desviación estándar estimada y el error del pronóstico y el histograma de la transformación integral de probabilidad (histograma PIT del inglés Probability Integral Transform). Para todas las métricas, se utiliza el estado verdadero ($\mathbf{x}^t$) como estado de referencia.

4.3.1. Error cuadrático medio

El objetivo de esta métrica es evaluar la precisión en la estimación del estado del sistema. Cuanto más se acerque el valor a cero, más precisa será la estimación del pronóstico. El RMSE se calcula como

$$\text{RMSE} = \sqrt{\frac{1}{N_x \cdot M} \sum_{j=1}^M (\tilde{\mathbf{x}}_j - \mathbf{x}_j^t)^\top (\tilde{\mathbf{x}}_j - \mathbf{x}_j^t)}, \quad (4.10)$$

donde $\mathbf{x}^t$ es el estado verdadero del sistema válido para el mismo tiempo del pronóstico, N_x es la dimensión del vector de estado y M es el número total de muestras del conjunto de verificación. El RMSE es una métrica determinística [Murphy \(1993\)](#) y no tiene en cuenta la incertidumbre del pronóstico, por lo que el objetivo principal de esta métrica es medir la precisión de nuestra estimación de la media de la distribución de estados $\tilde{\mathbf{x}}$.

4.3.2. Correlación desviación estándar/error absoluto

Para estudiar la calidad de la estimación de la incertidumbre, podemos comprobar que la magnitud de la varianza estimada covaríe con la magnitud de los errores del pronóstico. Para esto se utiliza el coeficiente de correlación de Pearson entre $\tilde{\sigma}_l$ y el valor absoluto del error de pronóstico $|\epsilon_l| = (|\tilde{\mathbf{x}}_l - \mathbf{x}_l^t|)$ que se obtiene de la siguiente manera:

$$\text{CORR}(\tilde{\sigma}_l, |\epsilon_l|) = \frac{\text{cov}(\tilde{\sigma}_l, |\epsilon_l|)}{\text{std}(\tilde{\sigma}_l)\text{std}(|\epsilon_l|)} \quad (4.11)$$

donde cov y std son la covarianza espacio-temporal y la desviación estándar respectivamente. El propósito de esta métrica es evaluar la correspondencia en la evolución temporal y espacial de la incertidumbre estimada por la red y el error del pronóstico. Una buena estimación de la incertidumbre debe estar correlacionada positivamente con el error de pronóstico. Los valores de correlación que no son significativamente distintos de cero indican que no se está capturando adecuadamente la dependencia de la incertidumbre del pronóstico con el estado del sistema.

4.3.3. Histogramas PIT

Una característica importante para analizar es la calibración de los pronósticos probabilísticos. Según [Gneiting et al. \(2007\)](#) la calibración refleja la consistencia entre la distribución de probabilidad del error estimada y la distribución de probabilidad del error real en un sentido estadístico. La calibración se puede cuantificar utilizando los histogramas PIT. Sea F_t la estimación, a un tiempo t , de la función de densidad acumulada (CDF) del error del pronóstico para la i -ésima variable de estado ($\epsilon_{(i)}$). El histograma PIT se construye como el histograma de $F_t(\epsilon_{(i)}^t)$ (es decir, la probabilidad estimada de que el error sea menor al error que efectivamente ocurrió) calculado para diferentes tiempos y variables de estado. Un histograma plano es una indicación de consistencia entre las distribuciones de probabilidad de error estimadas y reales. Sin embargo, tal y como se discute en [Hamill et al. \(2001\)](#), un histograma PIT plano no es suficiente para confirmar que el pronóstico esté calibrado o que sea confiable, ya que un histograma PIT plano puede ser el resultado de la compensación de sesgos condicionales. En [Wilks \(2019\)](#) miden la *uniformidad* del histograma PIT de la siguiente manera:

$$\chi^2 = \frac{N_b}{M \cdot N_x} \sum_{b=1}^{N_b} \left(f_b - \frac{N_b}{M \cdot N_x} \right)^2, \quad (4.12)$$

donde f_b es la frecuencia del b -ésimo intervalo, N_b es el número de intervalos y M es el tamaño de la muestra. Un χ^2 igual a 0 corresponde a un histograma uniforme, mientras que los valores positivos indican un apartamiento de la uniformidad.

4.3.4. Probabilidad de cobertura

Otra forma de evaluar la calibración de un pronóstico probabilístico es utilizar la probabilidad de cobertura. Esta consiste en calcular el número de veces que un intervalo de confianza estimado contiene al valor verdadero.

$$CP = \frac{\sum_{j=0}^M \mathbb{1}(\ell_j < x_j^a < u_j)}{M \cdot N_x}, \quad (4.13)$$

donde ℓ_l/u_l es el límite inferior/superior del intervalo de confianza de la estimación con distribución normal $\mathcal{N}(\tilde{x}_l, \tilde{\sigma}_l)$, $\mathbb{1}$ es la función indicatriz (toma el valor de 1 si la condición es verdadera y 0 en caso contrario) y M el tamaño del conjunto de datos. El valor perfecto para CP es igual al nivel de confianza utilizado para construir el intervalo de confianza, en nuestro caso 90 % (0,9). Un valor de CP superior/inferior al nivel de confianza indicaría una sobre/subestimación de la incertidumbre. La probabilidad de cobertura es una medida global de la confiabilidad o calibración de la incertidumbre cuantificada.

4.4. Resultados

4.4.1. Sensibilidad a los datos de entrada

Nuestro objetivo general es lograr entrenar RNAs capaces de estimar un pronóstico probabilístico correspondiente a un plazo de pronóstico dado. En esta sección buscamos entender si información adicional acerca de la evolución dinámica del sistema puede ayudar a la RNA a mejorar su estimación. En particular, buscamos determinar el impacto sobre la estimación de la varianza al incluir plazos anteriores o posteriores al plazo de pronóstico estimado, y de la separación de dichos plazos con respecto al plazo de pronóstico estimado.

Se realizaron 17 experimentos para evaluar la precisión del pronóstico corregido $\tilde{x}$ variando la cantidad de estados en la entrada de la red (entre 1 y 3 plazos de pronóstico) y considerando como posibles entradas a los pronósticos con plazos entre 4 y 60 pasos de tiempo alrededor del plazo que se desea corregir.

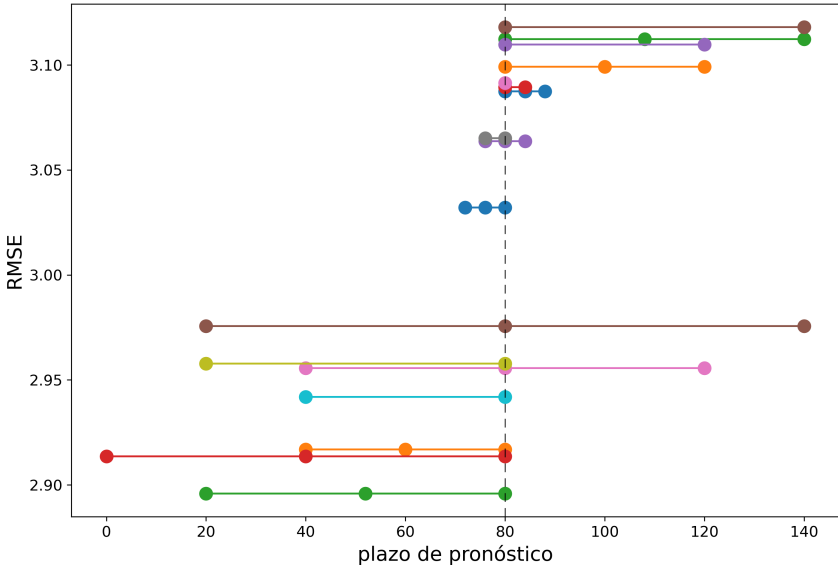


Figura 4.3: RMSE de la media estimada ($\tilde{x}_l$) para las redes con diferentes entradas. El RMSE se indica en el eje y , mientras que el eje x indica el plazo de pronóstico. Cada experimento está representado por una línea horizontal con un color diferente y los círculos de colores indican los plazos del pronóstico determinístico considerados para la estimación del error medio. Por ejemplo, la línea verde oscura de la parte inferior de la figura tiene puntos en los valores 20, 50 y 80, lo que significa que la entrada de la red es el conjunto de pronósticos determinísticos correspondientes a estos tres plazos. El RMSE alcanzado en el conjunto de prueba por esta red es de 2,9.

En estos experimentos, la salida de la red es el estado medio estimado para el plazo T80 (es decir, $l = 80$). Estos experimentos se realizaron únicamente para el EMI y en todos se utilizaron los mismos hiperparámetros y la misma arquitectura de red (sólo se varió la entrada). Para cada experimento, repetimos el entrenamiento de la RNA diez veces utilizando diferentes pesos iniciales para reducir el impacto de la inicialización de los pesos en el rendimiento resultante de las redes.

La figura 4.3 muestra el RMSE del estado estimado $\tilde{x}_l$ obtenido con las redes que utilizan diferentes datos de entrada. En general, todas las entradas producen buenos resultados, con valores de RMSE que están por debajo del valor correspondiente al pronóstico determinístico sin corregir (x_l). Sin embargo, las que incluyen información de los plazos previos al plazo de la salida l , funcionan mejor que las que incluyen plazos posteriores a l . Esto puede estar relacionado con el hecho de que los errores de pronóstico, y en parti-

cular los errores del modelo, evolucionan de acuerdo con la dinámica del sistema (véase la ecuación 2.2), por lo que incluir información sobre los estados pronosticados previos puede contribuir a describir mejor el error medio de pronóstico en el plazo l . El rango de plazos que abarca la entrada también tiene un impacto significativo en el rendimiento de la RNA. Las redes que abarcan un rango de plazos más amplio (y que utilizan información de plazos anteriores a l) producen estimaciones con menor RMSE. Además, el rendimiento de la RNA mejora cuanto más plazos se incorporen en la entrada, es decir que 3 plazos en la entrada produce menores valores de RMSE respecto a las que utilizan 2 o 1. El RMSE más bajo se obtuvo con la RNA de tres plazos en la entrada en los tiempos $\{\mathbf{x}_{20}^d, \mathbf{x}_{50}^d, \mathbf{x}_{80}^d\}$.

En base a estos resultados, en el resto de este capítulo se utilizan las siguientes entradas: para T4 la entrada consiste en $\{\mathbf{x}_0^d, \mathbf{x}_4^d\}$ (sólo dos tiempos de entrada dado que en este caso el plazo de pronóstico es muy corto), para T80 utilizamos $\{\mathbf{x}_0^d, \mathbf{x}_{40}^d, \mathbf{x}_{80}^d\}$; y para T160, utilizamos $\{\mathbf{x}_0^d, \mathbf{x}_{80}^d, \mathbf{x}_{160}^d\}$.

4.4.2. Experimentos con modelo perfecto

En los experimentos de modelo perfecto, los errores de pronóstico son el resultado de errores en la especificación de las condiciones iniciales. Este escenario particular nos permite evaluar hasta qué punto las RNAs capturan el crecimiento de los errores debido al comportamiento caótico. Además, estos experimentos están diseñados para evaluar si las RNAs son capaces de filtrar los modos más inciertos presentes en el estado pronosticado, como en el caso de la media del ensemble (Kalnay, 2003). En el pronóstico por ensambles, la media del ensemble $\bar{\mathbf{x}}_l^e$ ejerce un filtrado del error asociado a la pérdida de predictibilidad a medida que l aumenta. Además, la dispersión del ensemble (σ_l^e) en el EMP es una estimación fiable de la desviación estándar del error del pronóstico que depende del estado. Esto es especialmente cierto cuando las perturbaciones iniciales del ensemble se generan utilizando un filtro de Kalman por ensambles (Candille et al., 2007). La figura 4.4a muestra la evolución temporal de la media del ensemble de pronósticos en T80 para la primera variable de estado $x_{(1)}^d$. La figura 4.4b muestra la predicción de la RNA obtenida con el enfoque RNA-lik. A primera vista, tanto el ensemble como las redes proporcionan buenas estimaciones del estado del sistema. Dado que nos encontramos en el régimen de error lineal, en general, no se encuentran diferencias importantes entre el desempeño de un pronóstico determinístico inicializado con la media del ensemble, la media del ensemble de pronósticos y el estado medio estimado a partir de la red. En cuanto a la incertidumbre del pronóstico, en general, ambos sistemas proporcionan

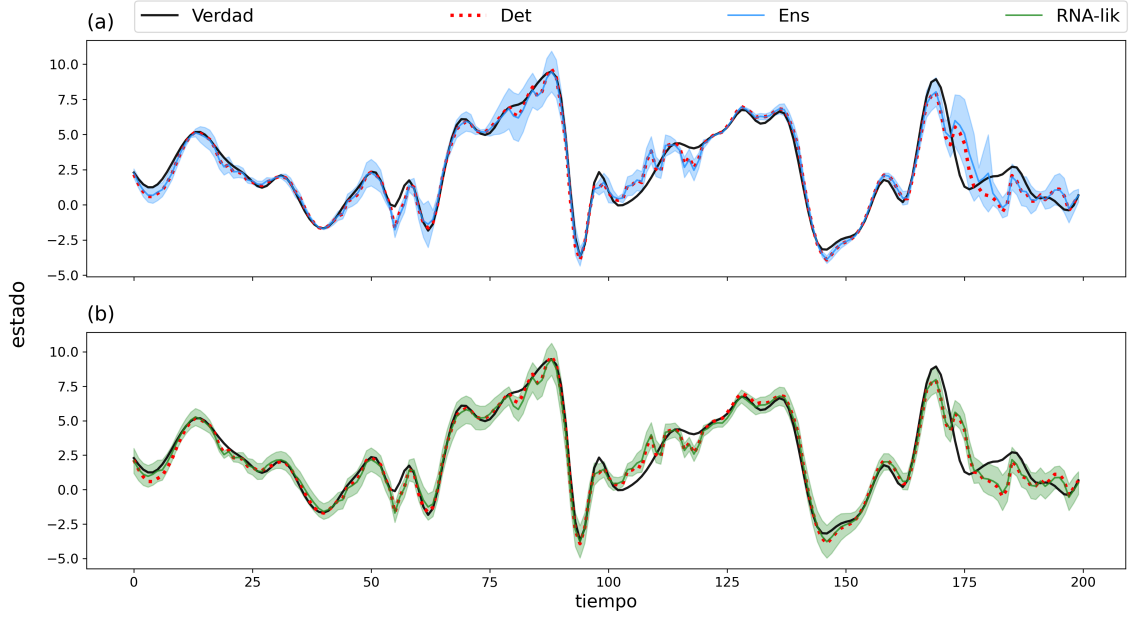


Figura 4.4: Evolución temporal del pronóstico a 80 pasos de tiempo (T80) de la primera variable de estado y su incertidumbre asociada para 200 pasos temporales consecutivos tomados del conjunto de pruebas. (a) Media de los pronósticos por ensamble ($\bar{x}_{(1),l}^e$, línea azul sólida) y dispersión del ensamble ($\sigma_{(1),l}^e$, sombreada). (b) Media estimada por la red RNA-lik ($\tilde{x}_{(1),l}$, línea verde sólida) y la dispersión predicha ($\tilde{\sigma}_{(1),l}$, sombreada). En ambos paneles se muestra el pronóstico del modelo determinístico ($x_{(1),l}^d$, líneas rojas punteadas) y la verdad de base (líneas negras sólidas).

una buena estimación dependiente del estado del sistema (es decir, la curva negra está dentro del área sombreada la mayor parte del tiempo). La incertidumbre estimada a partir de la RNA muestra una menor variabilidad temporal que la del ensamble. La relación entre el pronóstico y su incertidumbre no es trivial. Valores similares del pronóstico determinístico para la primera variable de estado están asociados a valores diferentes para la varianza del error estimada.

La figura 4.5 muestra la evolución espacio-temporal del error absoluto $|\epsilon|$ (paneles de la izquierda) y el desvío estándar del pronóstico σ (paneles de la derecha). El desvío estándar para el pronóstico determinístico es la desviación estándar de la diferencia entre el pronóstico y el análisis ($\sigma^c = (\sum_{i=1}^{N_x} \sum_{j=1}^M (\mathbf{x}_{(i),j,l}^d - \mathbf{x}_{(i),j+l}^a)^2)^{\frac{1}{2}}$) sobre la muestra de entrenamiento, y por tanto es independiente del estado (Figura 4.5f), a esta estimación la denominamos estimación constante o climatológica. Los errores absolutos de los tres

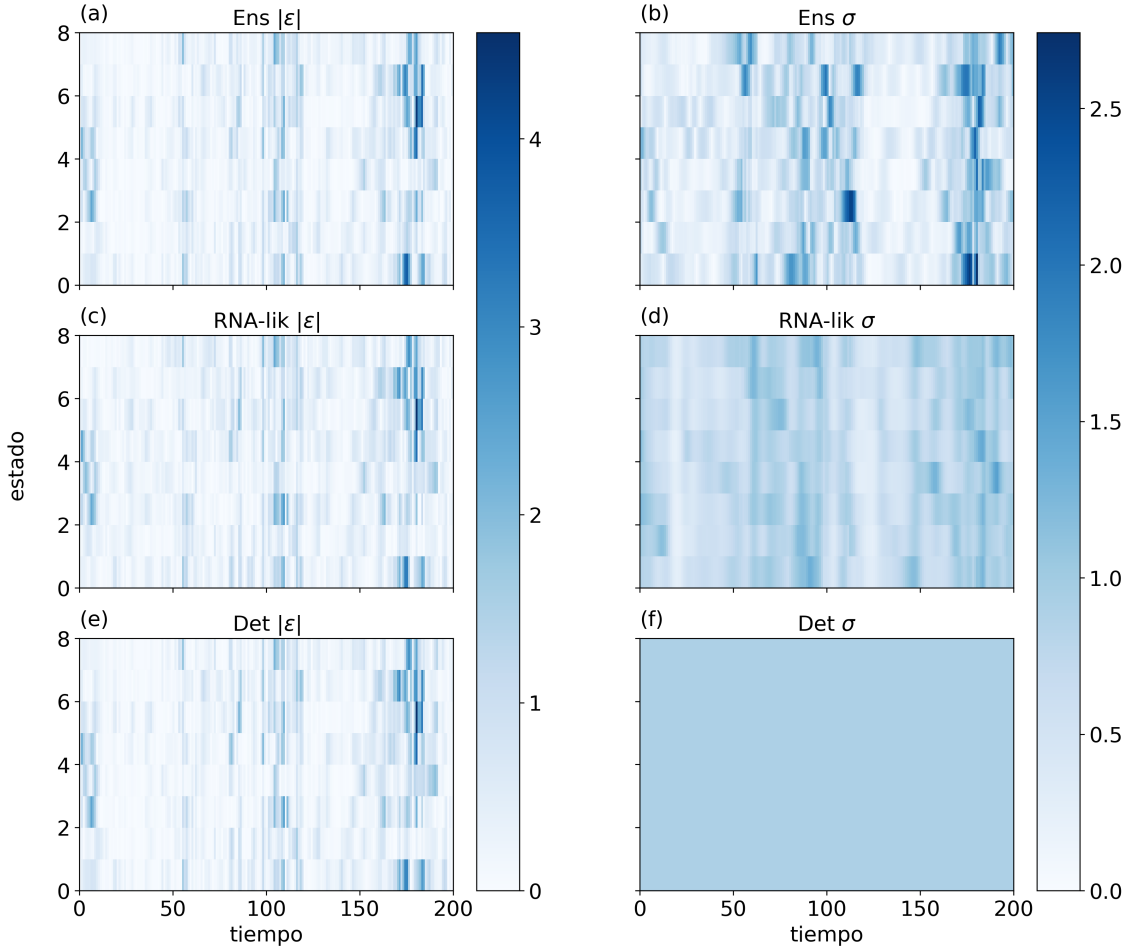


Figura 4.5: Paneles de la izquierda (a,c,e): error absoluto ($|\epsilon|$) de cada variable de estado (eje y) como función del tiempo (eje x) para el plazo de pronóstico T80 y para (a), la media del ensemble, (c) $\tilde{x}_l$ obtenido con el enfoque RNA-lik y (e) el pronóstico determinístico. Paneles de la derecha (b,d,f): la desviación estándar del error de pronóstico (σ) para cada variable del modelo (eje y) y como función del tiempo (eje x) y obtenida a partir de: (b) el pronóstico por ensambles, (d) el enfoque RNA-lik y (f) la estimación estática. El rango temporal corresponde a los primeros 200 tiempos del conjunto de prueba

experimentos muestran una variabilidad relativamente grande y son bastante similares en términos de magnitud, evolución espacio-temporal y localización de los valores extremos. Esto confirma la fuerte dependencia de los errores de pronóstico con el estado del sistema.

En el panel 4.5b, las áreas con máxima dispersión del ensamble están en buena concordancia con la ubicación de los mayores errores de pronóstico (panel 4.5a). Es importante señalar que no se espera una coincidencia perfecta entre estas dos magnitudes, ya que el error de pronóstico es una realización de la función de densidad de probabilidad, en inglés Probability Density Function (PDF) estimada a partir del ensamble de pronósticos. Las RNAs también proporcionan una estimación del desvío estándar que coincide, a grandes rasgos, con la evolución del error. La figura 4.5 muestra los resultados correspondientes al entrenamiento RNA-lik, pero el obtenido por RNA-ext y RNA-mse es muy similar y, en consecuencia, no se muestran. Se observa también que las RNAs muestran una variabilidad espacial y temporal menor que la estimada a partir del ensamble.

Estos resultados son bastante alentadores, ya que muestran que las metodologías basadas en RNA son capaces de estimar la incertidumbre del pronóstico en función del estado del sistema. Además, esta estimación se consigue utilizando como entrada un pronóstico determinístico cuyo coste computacional es mucho más bajo que la integración de un pronóstico por ensambles.

Hasta ahora hemos analizado cualitativamente la habilidad de la RNAs para cuantificar la incertidumbre del pronóstico a partir de un pronóstico determinístico en el plazo T80. La figura 4.6 resume los resultados obtenidos para los diferentes tiempos de salida considerados (T4, T80 y T160) y para las diferentes métricas descritas en la sección 4.2. Los intervalos de confianza para las diferentes métricas se calculan mediante la técnica de bootstrap en la que se obtienen 500 muestras de la muestra original utilizando una selección aleatoria con sustitución. Para el cálculo de las métricas se seleccionan pronósticos en tiempos válidos que están separados por más de 20 pasos de tiempo para aumentar la independencia entre los diferentes elementos de la muestra.

En lo que respecta al RMSE (primera fila de la Fig. 4.6), todos los sistemas de pronóstico muestran un rendimiento similar para T4 y T80, siendo la media del ensamble ligeramente mejor en T80. Todas las RNAs tienen el mismo RMSE ya que el entrenamiento de la red que produce el pronóstico corregido (la red de medias) es el mismo para todas ellas.

Para el tiempo T160 (Fig. 4.6c) el crecimiento del error se ve afectado por los efectos no lineales, el ensamble es capaz de filtrar adecuadamente algunos de los modos impredecibles, dando como resultado un RMSE menor con respecto a los pronósticos determinísticos. Un resultado importante es que las RNAs superan al pronóstico determinístico para este plazo de pronóstico. Esto sugiere que las RNAs son capaces de filtrar al menos parte de los modos imprevisibles, aunque no son tan eficientes como la media del ensamble

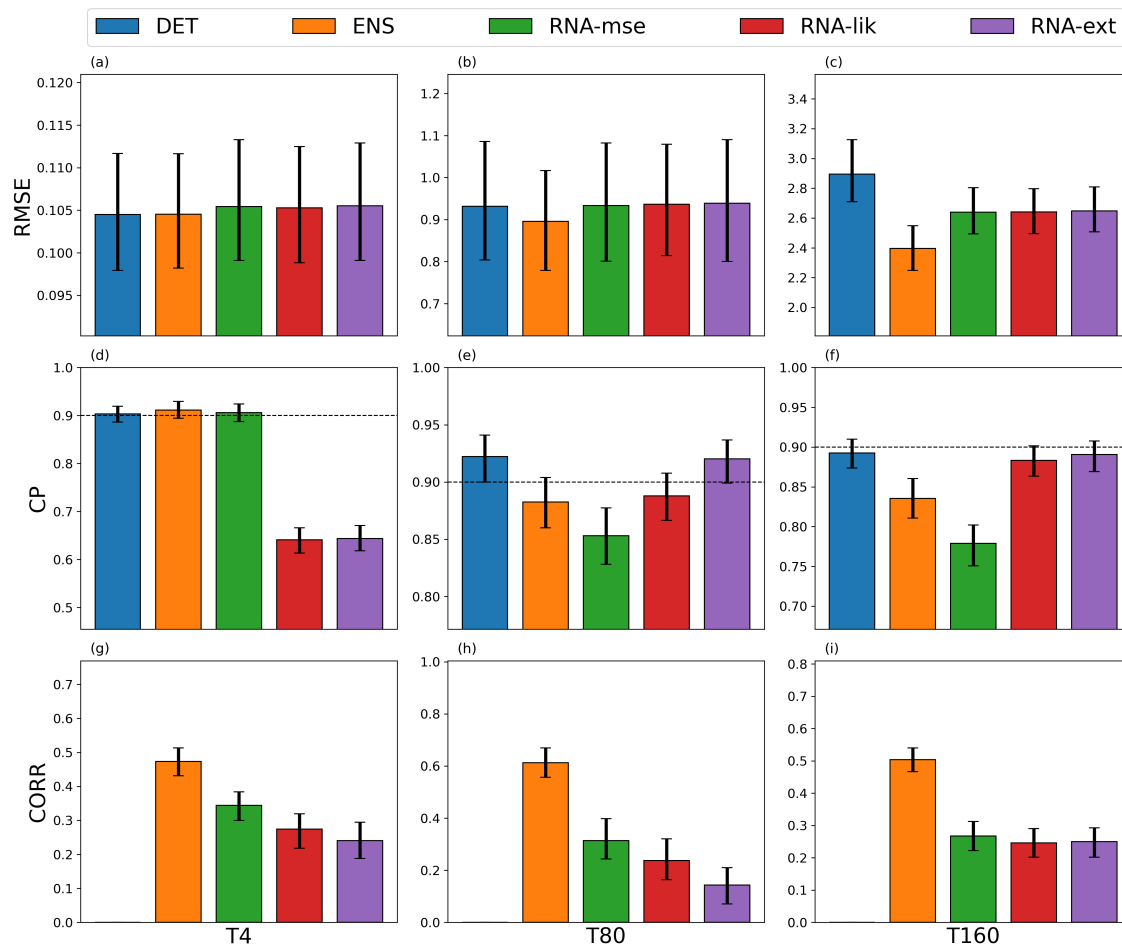


Figura 4.6: métricas correspondientes al pronóstico determinístico (barra azul), al pronóstico por ensamble (barra amarilla) y a los enfoques RNA-mse (barra verde), RNA-lik (barra roja) y RNA-ext (barras moradas) en el escenario de modelo perfecto. La primera fila corresponde al RMSE, la segunda a la probabilidad de cobertura del 90 % y la tercera al coeficiente de correlación desviación estándar/error absoluto. La primera, segunda y tercera columnas corresponden a los plazos T4, T80 y T160 respectivamente. La línea de error en la parte superior de las barras indica el intervalo de confianza del 95 % calculado mediante la técnica de bootstrap. La línea horizontal de los paneles de la segunda fila indica el valor perfecto asociado a la probabilidad de cobertura del 90 %.

para hacerlo (Notar que el ensamble es de 50 miembros y el sistema dinámico de 8 variables, en casos de alta dimensionalidad esta relación se invierte degradando el desempeño de la metodología). La segunda fila de la Fig. 4.6 muestra la probabilidad de cobertura

(CP). La estimación de los enfoques RNA-ext y RNA-lik para los tiempos de pronóstico T4 presenta una fuerte subestimación de la incertidumbre. En el plazo de pronóstico T4, la magnitud del error de pronóstico en el EMP está muy correlacionada con el error de análisis, es decir, el proxy de error de pronóstico utilizado para el entrenamiento está subestimando la magnitud del error de pronóstico. Esto afecta a los enfoques RNA-ext y RNA-lik, que se basan en este proxy para la estimación de la incertidumbre del pronóstico, pero no al enfoque NN-mse, que se entrena en función de la incertidumbre estimada por el ensamble. Para verificar esta conjetura, realizamos experimentos adicionales entrenando una red con el enfoque RNA-lik pero utilizando el estado real para calcular el error del pronóstico y construir así el objetivo para el entrenamiento. Como resultado, se encuentra una mejora en términos de RMSE y CP. En particular, la subestimación de la incertidumbre del pronóstico en T4 no está presente en estos experimentos. Los valores de CP pasan de 0,63 cuando se entrena con el análisis a 0,87 cuando se entrena con la verdad, y los valores de RMSE varían de 0,12 a 0,10 respectivamente.

En los plazos más largos (T80 y T160), el ensamble y la red RNA-mse subestiman la desviación estándar del pronóstico. La razón de esto no está clara, pero puede ser el resultado de una estimación subóptima de la incertidumbre del análisis en el sistema de asimilación de datos basado en el ensamble. En cualquier caso, este comportamiento muestra una posible limitación del enfoque RNA-mse que puede heredar sesgos en la cuantificación de la incertidumbre del pronóstico derivados de una formulación subóptima del ensamble utilizado para el entrenamiento de la red.

La tercera fila de la Fig. 4.6 muestra la correlación espacio-temporal entre la desviación estándar estimada del pronóstico y el error absoluto del mismo. Una baja correlación entre el error y el desvío estándar indica que el sistema es incapaz de captar la naturaleza dependiente del estado de la incertidumbre del pronóstico. Las figuras 4.6g, h e i muestran correlaciones positivas para el pronóstico por ensamble y las RNAs. Como era de esperar, la desviación estándar independiente del tiempo del error del pronóstico determinístico no está correlacionada con el error de pronóstico real. El pronóstico por ensamble supera a las RNAs en todos los plazos considerados. Se trata de un resultado esperado, ya que el pronóstico por ensamble se basa en el muestreo directo de la función de densidad de probabilidad del estado del sistema, especialmente en estos experimentos de modelo perfecto. Por otra parte, al utilizar en estos experimentos un ensamble de 50 miembros, que es relativamente grande dada la dimensión del estado, el efecto de los errores de muestreo (que suele afectar el desempeño de los pronósticos por ensambles) es relativamente pequeño. La red RNA-mse que se entrena directamente a partir de un

ensamble de pronósticos tiene un rendimiento ligeramente mejor que las redes entrenadas indirectamente en T4 y T80, pero en T160 todas las RNAs muestran un rendimiento similar. Se consideraron otras métricas de rendimiento, como el CRPS, del inglés Continuous Ranked Probability Score, pero no se muestran dado que no aporta información adicional a la proporcionada por las métricas mencionadas anteriormente.

4.4.3. Experimentos con modelo imperfecto

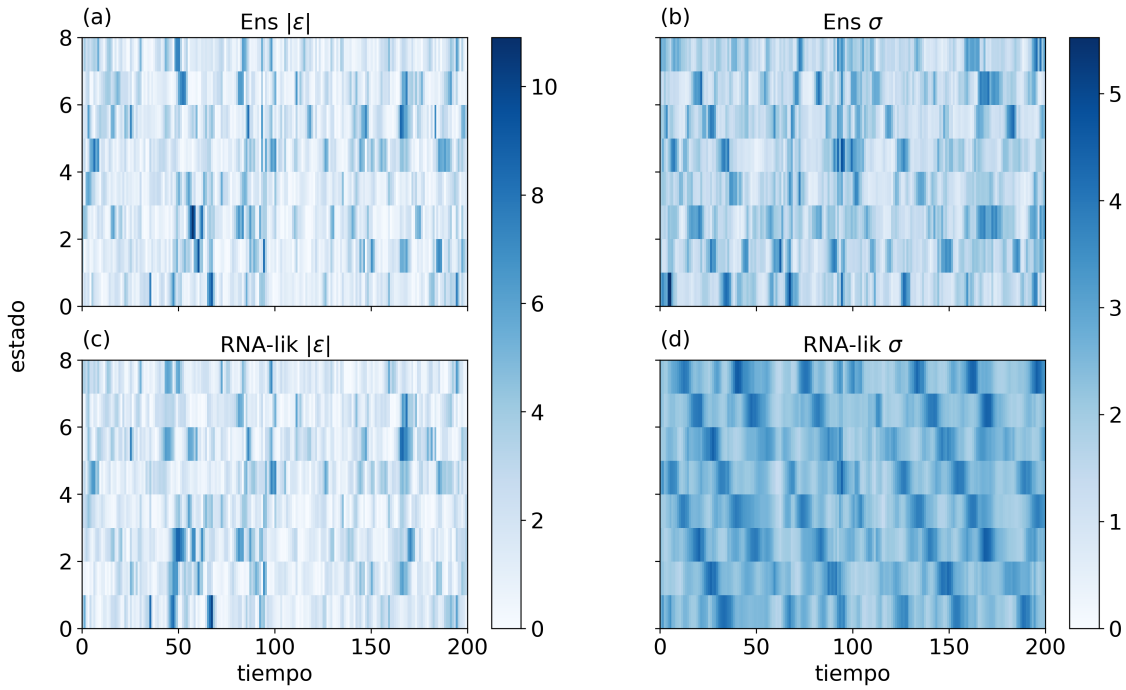


Figura 4.7: Paneles de la izquierda (a,c): evolución temporal del error absoluto de cada variable de estado (eje y) del modelo dinámico en el plazo T80 para la media del ensamble (a) y el enfoque RNA-lik (c). Paneles de la derecha (b,d): la desviación estándar del error de pronóstico estimada a partir del ensamble (b) y del RNA-lik (d).

En el escenario de modelo imperfecto, los errores de pronóstico son mayores debido a la contribución de los errores del modelo. Además, la media del error de pronóstico suele ser distinta de cero, es decir, existe un componente de error sistemático.

En este escenario, el error del pronóstico crece más rápidamente que en el EMP (véase la figura 4.2). En EMI, el plazo T80 está al principio del régimen no lineal y T160 está cerca

de la saturación del error, lo que pone de manifiesto el rol de los errores de modelo en acelerar la pérdida de predictibilidad.

La figura 4.7 muestra la evolución espacio-temporal del error absoluto y su desviación estándar estimados por el ensamble y el enfoque RNA-lik. Aunque los patrones de error no son tan parecidos como los observados en el EMP (Fig. 4.5), existe una gran similitud en la forma y la intensidad de los mismos (Fig. 4.7a y 4.7c). Los paneles (b) y (d) muestran el desvío estándar del error estimados a partir del ensamble y de la red neuronal. Tanto la red como el ensamble son capaces de representar la incertidumbre del pronóstico dependiente del estado como en el escenario EMP. La desviación estándar estimada es mayor para el enfoque RNA-lik que para el ensamble. Esto indica una subestimación de la desviación estándar del error en el enfoque de pronóstico por ensamble que puede atribuirse en parte a la falta de representación del error del modelo en la construcción del ensamble.

En la Fig. 4.8 se muestra un resumen del rendimiento de los distintos sistemas de pronóstico en el escenario EMI para los plazos T4, T80 y T160. En términos de RMSE, las RNAs obtienen una ventaja notable en la precisión de la estimación del estado en todos los plazos. Las pronósticos corregidos por las RNAs son incluso mejores que el pronóstico medio del ensamble. Esto sugiere que la contribución de los errores del modelo es significativa y comparable o incluso mayor que el error asociado al conocimiento imperfecto de las condiciones iniciales. Las RNAs son capaces de captar parcialmente los errores del modelo que dependen del estado, mientras que el pronóstico determinístico y el pronóstico por ensamble no lo hacen. (véase la Fig. 4.10).

El CP (segunda fila de la Fig. 4.8) muestra una clara ventaja de las RNAs sobre los pronósticos determinísticos y por ensamble, particularmente en T80 y T160. En T4 se observa un comportamiento similar al del EMP, con RNA-ext y RNA-lik con peores resultados que RNA-mse. Esto es, de nuevo, una consecuencia directa de la utilización del análisis como objetivo de aprendizaje para la estimación de la incertidumbre del pronóstico. Una vez que el error de pronóstico es mayor (por ejemplo, T80), este comportamiento se invierte: las redes que aprenden la incertidumbre a partir de los datos se comportan mejor que el ensamble en términos de CP, como se muestra en la Fig. 4.7. Es interesante notar que la técnica de aprendizaje RNA-mse, que aprende directamente del ensamble, se desempeña mejor que el propio ensamble en términos de CP. Esto se debe en parte a que la red mejora el estado del pronóstico al eliminar el componente de error de pronóstico sistemático dependiente del estado. Así, la desviación estándar del pronóstico estimada es más coherente con los errores de pronóstico que en el caso del pronóstico por ensam-

ble. Este resultado coincide con los obtenidos en Grönquist et al. (2021) y analizados en la sección 3.1, esto sugiere que incluso un ensamble con una representación subóptima de los errores del modelo puede ser un objetivo razonable para el entrenamiento de una RNA. El ensamble muestra el peor rendimiento en términos de CP para T80 y T160. Esto se debe a que la media del ensamble se ve afectada por el error de modelo que no se tienen en cuenta en la formulación del ensamble, lo que conduce a una subestimación de la desviación estándar del pronóstico.

La correlación entre el error absoluto y la desviación estándar del pronóstico (tercera fila de la Fig. 4.8) muestra una correlación similar entre RNA y ensamble.

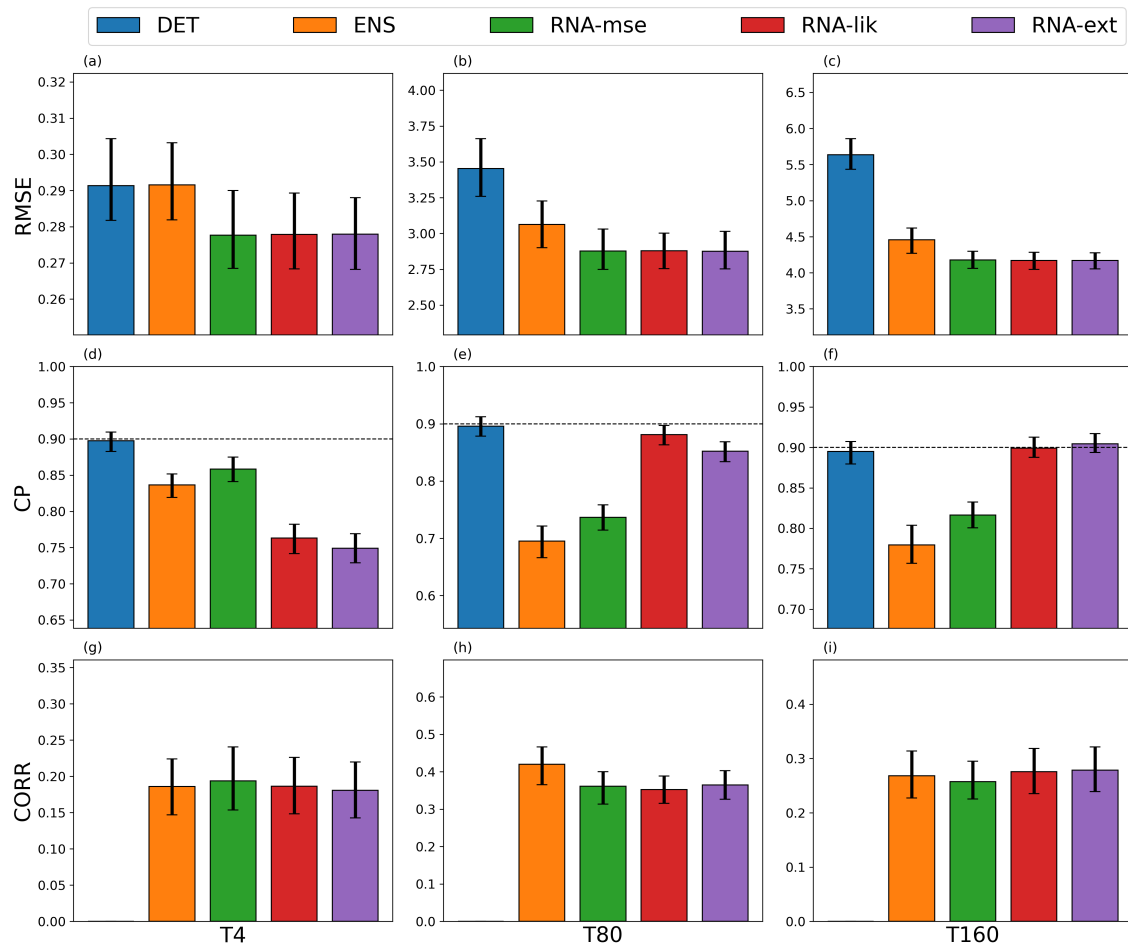


Figura 4.8: Igual que la Fig. 6 pero para el escenario de modelo imperfecto.

La figura 4.9 muestra los histogramas PIT para los pronósticos en T80 generados por el

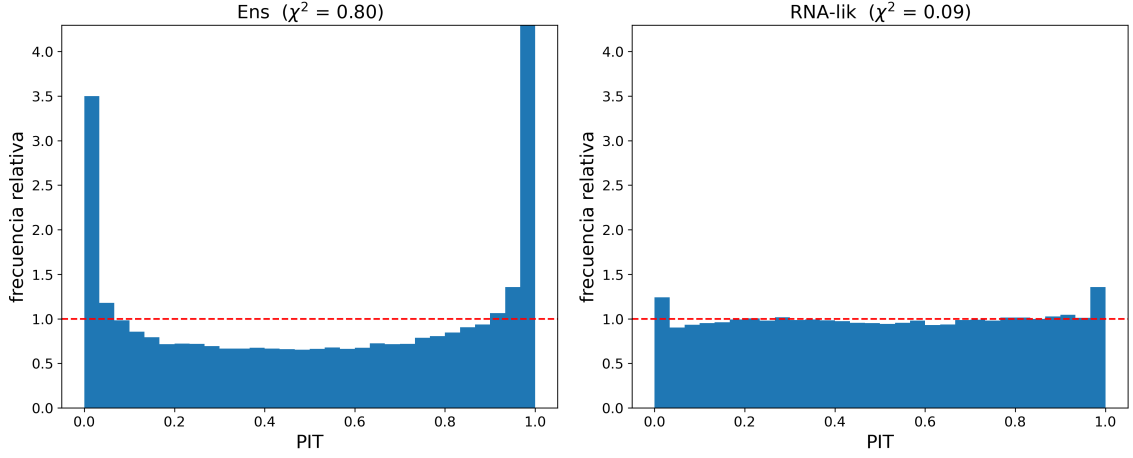


Figura 4.9: Histograma de la transformación integral de probabilidad para el pronóstico por ensemble (panel izquierdo) y para el pronóstico RNA-lik. Ambos casos corresponden al plazo de pronóstico T80. En la parte superior de cada panel se indica el parámetro de uniformidad χ^2 correspondiente.

ensemble y RNA-lik. Los otros enfoques de entrenamiento RNA tienen un rendimiento similar y no se incluyen en la figura. El pronóstico por ensemble muestra un histograma PIT en forma de U, lo que indica una subestimación de la incertidumbre del pronóstico [Hamill et al. \(2001\)](#). Esta falta de confiabilidad del ensemble está asociada a una representación subóptima de los errores del modelo que introduce un sesgo sistemático en la media del ensemble y contribuye a incrementar la componente estocástica de los errores del pronóstico. Ambos aspectos son parcialmente tenidos en cuenta por las RNAs, lo que explica la mejora de la confiabilidad del pronóstico (nótese que el valor de χ^2 es un orden de magnitud menor para las redes que para el ensemble).

4.4.4. Análisis del rendimiento de las estimaciones

Para comprender mejor cómo dependen los errores de pronóstico de las variables de estado y hasta qué punto estas dependencias están representadas por las RNAs y el ensemble, examinamos algunos aspectos de la relación entre las variables de estado y el error de pronóstico. Nos centramos en los experimentos EMI comentados en la sección anterior. Los paneles de la izquierda de la Fig. 4.10 muestran histogramas 2D de la distribución del error de pronóstico para cada variable en particular ($\epsilon_{(i)}$) en función del de la media del ensemble de pronósticos para dicha variable ($\bar{x}_{(i)}^e$, Fig. 4.10a) y la RNA-ext ($\tilde{x}_{(i)}$, Fig.

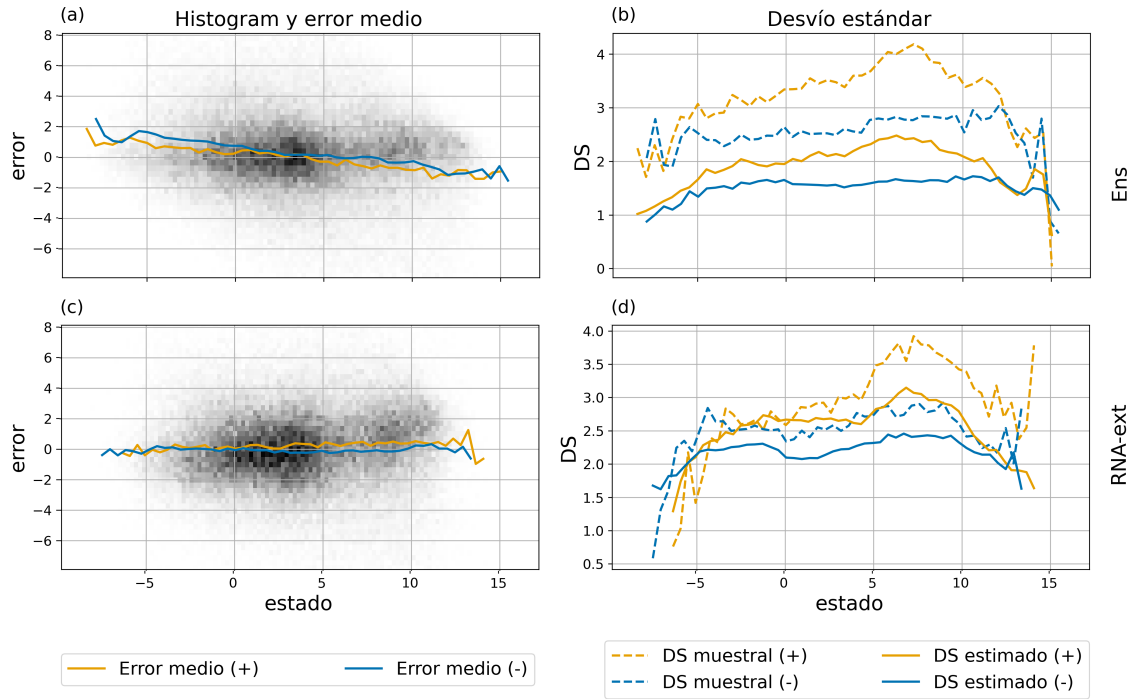


Figura 4.10: Los paneles (a) y (c) muestran la densidad 2D del error de pronóstico y los valores de estado pronosticados (tonos grises). Las líneas sólidas amarillas y azules indican el error medio del pronóstico en función del valor del estado pronosticado para las submuestras de tendencia positiva y tendencia negativa, respectivamente (véase el texto para más detalles). Los paneles (b) y (d) muestran la desviación estándar del error en función de la variable de estado pronosticada para las submuestras de tendencia positiva y de tendencia negativa (línea amarilla discontinua y línea azul discontinua, respectivamente). Las líneas amarillas y azules sólidas representan la desviación estándar estimada para las submuestras de tendencia positiva y de tendencia negativa, respectivamente. Los paneles (a) y (b) corresponden al pronóstico por ensamble y (c) y (d) al pronóstico RNA-ext. Todos los paneles corresponden al escenario de modelo imperfecto.

4.10c). Dado que para este modelo simple todas las variables de estado (subíndice (i)) comparten las mismas estadísticas de error, los resultados de las diferentes variables de estado se agregaron para construir una única muestra. La figura muestra además la relación entre la media del error condicionada al estado pronosticado bajo dos regímenes diferentes: uno en el que la tendencia del estado es positiva (es decir, $\frac{dx_{(i)}}{dt} > 0$, línea amarilla) y otro en el que la tendencia es negativa (es decir, $\frac{dx_{(i)}}{dt} < 0$, línea celeste). Esta distinción está motivada por el trabajo de [Pulido et al. \(2016\)](#), que muestra en un entorno

experimental similar que la magnitud de los errores del modelo tiene un comportamiento no markoviano ya que depende de la derivada temporal de las variables de estado. Para construir estas curvas, el rango de valores de estado se divide en 50 intervalos y, a continuación, se calcula la media de los errores de pronóstico correspondientes a cada intervalo. Este procedimiento se realiza de forma independiente para las submuestras de tendencia positiva y de tendencia negativa.

Las figuras 4.10b y 4.10d muestran la desviación estándar del error condicionada al valor del estado calculada a partir de los datos (líneas discontinuas). Estas curvas se obtienen de forma independiente para la submuestra de tendencia positiva (línea discontinua amarilla) y para la submuestra de tendencia negativa (línea discontinua azul) siguiendo un procedimiento similar al de las curvas de error medio. Las estimaciones de la desviación estándar del error de la RNA y el ensamble se muestran como líneas sólidas en las Figs. 4.10b y 4.10d. En este caso, calculamos la media de la desviación estándar del error estimada en cada uno de los intervalos que cubren el rango de la variable de estado pronosticada. Este procedimiento también se aplica de forma independiente a la tendencia positiva (línea sólida amarilla) y a la tendencia negativa (línea sólida azul).

El error medio del ensamble depende fuertemente de la variable de estado (pendiente en el error medio), aunque el error promediado globalmente es cercano a 0 (Fig. 4.10a). Esta pendiente está probablemente asociada al siguiente mecanismo: cuando se pronostica un valor alto de la variable de estado, es más probable que el error sea negativo que positivo (relacionado con una subestimación del valor). Lo contrario ocurre con los valores bajos de $x(i)$. Esto produce una asimetría en la forma de la distribución de la probabilidad de error que se hace más evidente para los valores extremos del estado. Por el contrario, el error medio de RNA-ext (Fig. 4.10c) es casi independiente del valor de estado pronosticado. Este comportamiento se produce a expensas de una menor frecuencia de valores extremos en los pronósticos corregidos por la RNA. Nótese que los histogramas de la Fig. 4.10c abarcan un rango menor de valores de estado que en la Fig. 4.10a. En la Fig. 4.10b se encuentra una clara diferencia entre las submuestras de tendencia positiva y negativa para la desviación estándar del ensamble, que también depende del valor pronosticado. Este resultado es coherente con las conclusiones de Pulido et al. (2016). La dispersión del ensamble (Fig. 4.10b) subestima la desviación estándar del error de pronóstico pero representa adecuadamente su dependencia con el valor pronosticado.

La figura 4.10d muestra que la RNA-ext proporciona una estimación más fiable de la desviación estándar del pronóstico en comparación con el ensamble (las líneas sólidas y discontinuas están más próximas) y también captura la relación entre la desviación

estándar del pronóstico y el valor del estado pronosticado. Estos resultados sugieren que la dependencia de la incertidumbre del estado del sistema es compleja y no lineal, y que la cuantificación de la incertidumbre basada en RNA es capaz de captar esta complejidad de forma similar a la de un ensamble de pronósticos. Los experimentos realizados con el enfoque RNA-lik dieron resultados similares a RNA-ext y, en consecuencia, no se muestran en la figura.

4.5. Aplicación a modelos de pronóstico numérico de la atmósfera

En esta sección comprobaremos si los resultados obtenidos en las secciones anteriores usando un modelo dinámico sencillo (el modelo de Lorenz) se mantienen al ser aplicados a un sistema de pronóstico meteorológico del estado del arte. En este caso, solo se podrá simular un entorno de modelo imperfecto.

4.5.1. Conjunto de datos

Se utilizarán pronósticos de la presión a nivel del mar del NCEP (National Centers for Environmental Prediction) disponibles desde el 1 de enero de 2000 hasta el 31 de diciembre de 2019 (Guan et al., 2022). Estos son pronósticos retrospectivos realizados con la versión estable (versión 2) del modelo numérico de predicción meteorológica que se aproxima a la versión más reciente utilizada en forma operativa. Este procedimiento elimina los cambios en los componentes de error sistemático asociados a la evolución de las versiones del modelo. Este conjunto de datos permite una mejor caracterización de los errores sistemáticos del modelo y su dependencia con el estado del sistema, lo que los hace especialmente atractivos para el tipo de análisis que se realizará.

Este conjunto de datos consiste en un ensamble global de 5 miembros (un pronóstico de control y 4 pronósticos perturbados). Los pronósticos se inicializan una vez al día a las 00 UTC, y se almacenan con una resolución espacial de 0,25 grados y una resolución temporal de 3 horas. Realizamos los experimentos en un dominio situado sobre el Atlántico Sur (55° W a 5° W y 60° S a 40° S). Esta zona se encuentra dentro del flujo de los oestes en latitudes medias y es bastante activa en términos de ondas baroclínicas, por lo que se espera una importante variabilidad en la incertidumbre del pronóstico en esta zona. Para reducir las dimensiones del problema, se degradó la resolución del pronóstico a 2°, lo que da lugar a un dominio de 10×10 puntos de retícula. En este experimento se

entrenaron dos RNAs que proporcionaron un pronóstico mejorado y su incertidumbre asociada a un plazo de 72 horas. Este plazo se encuentra en la fase lineal del crecimiento del error. La entrada a las redes es una secuencia de pronósticos determinísticos (es decir, el pronóstico de control) a 0, 36 y 72 horas de plazo. Las condiciones iniciales del pronóstico se utilizan como objetivo de entrenamiento.

El conjunto de datos consta de 7302 pronósticos de los cuales utilizamos 5096 para el entrenamiento, 1104 para la validación y 1102 para pruebas. Para reducir el impacto de la estacionalidad en la variabilidad de la presión, estandarizamos los valores de presión con respecto a su media climatológica y su desvío estándar. Para filtrar el ciclo estacional, calculamos la media y la desviación estándar sobre una ventana temporal de 30 días centrada en cada día del año.

4.5.2. Red neuronal

Se utilizó la misma arquitectura que en el resto del capítulo: dos redes totalmente conectadas con dos capas ocultas, una red para la estimación de la media del pronóstico y otra para la estimación del desvío estándar. El tamaño de la entrada y la salida viene determinado por el tamaño de la retícula y consiste en $3 \times 10 \times 10$ neuronas y 10×10 respectivamente para ambas redes. Exploramos la sensibilidad del rendimiento de la red al tamaño de las capas ocultas en el rango de 100 a 2000 neuronas encontrando pocas mejoras más allá de 500 neuronas. Para el entrenamiento de las redes, se siguió la misma metodología explicada en la sección 4.2.

4.5.3. Resultados

En la figura 4.11 muestra los estados analizados para tres eventos distintos (un evento por fila) correspondientes a ciclones extra-tropicales intensos. Para la selección de los casos mostrados, se buscaron los ciclones más intensos en donde el ciclón estuviera relativamente cerca del centro de la retícula.

Tanto el ensamble como las redes sitúan la máxima desviación estándar del error en y al este del ciclón en el caso de la primera y última fila. Para el caso de la segunda fila, los máximos de incertidumbre coinciden con el máximo de error en el ensamble y en las redes, pero el ensamble no tiene asociado el máximo de error al mínimo de presión como sí lo hacen las RNAs. La distribución espacial de la incertidumbre estimada por las RNAs es espacialmente más suave que la obtenida a partir del ensamble. Para las RNAs parece haber una dependencia de la incertidumbre con el valor de la presión en superficie,

con mayores incertidumbres asociadas a los sistemas de baja presión. Esto es de esperar, ya que los sistemas de baja presión durante su fase de amplificación están asociados a regiones con inestabilidad baroclínica, que es uno de los principales mecanismos de amplificación de los errores en la escala sinóptica. Asociado a la intensificación de estos ciclones también pueden presentarse fenómenos de mesoescala, como la convección húmeda profunda, que promueven el rápido crecimiento del error en escalas más pequeñas. La convección húmeda profunda no puede ser representada con precisión en los modelos numéricos globales de última generación, lo que da lugar a una mayor contribución de los errores del modelo al error total del pronóstico.

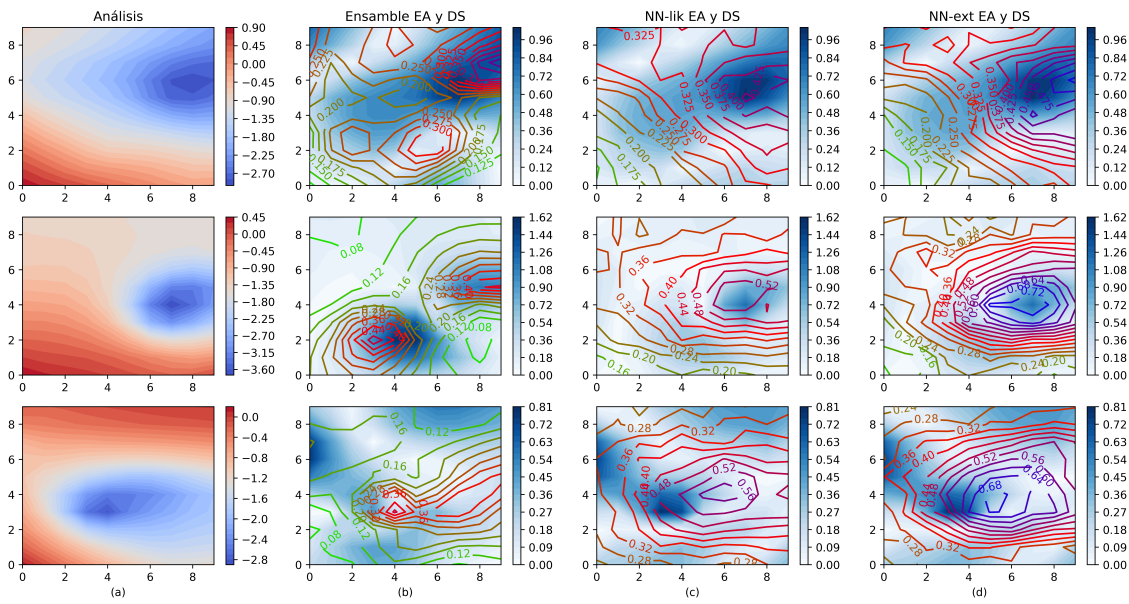


Figura 4.11: Distribución espacial del análisis de la presión a nivel del mar (columna a). Error del pronóstico de 72 hs (sombreado) y la estimación de su desviación estándar (contornos) para las técnicas de ensambles (columna b), RNA-lik (columna c) y RNA-ext (columna d) para los ciclones extratropicales ocurridos en la región de estudio el 20 de junio de 2019 (primera fila), el 3 de enero de 2019 (segunda fila) y el 3 de julio de 2018 (tercera fila).

La figura 4.12 muestra el RMSE, la probabilidad de cobertura y la correlación entre la desviación estándar del error pronosticado y el error absoluto para el ensamble y para el RNA-lik y RNA-ext. Estas métricas muestran un comportamiento similar al de los experimentos sintéticos con el modelo imperfecto en T80. A diferencia de los experimentos idealizados, en este caso no tenemos acceso al estado real, por lo que utilizamos el

análisis como una estimación del estado del sistema para evaluar el rendimiento de las redes. Las redes muestran una mejora en el error cuadrático medio (RMSE) y una mayor confiabilidad medida por la probabilidad de cobertura (CP).

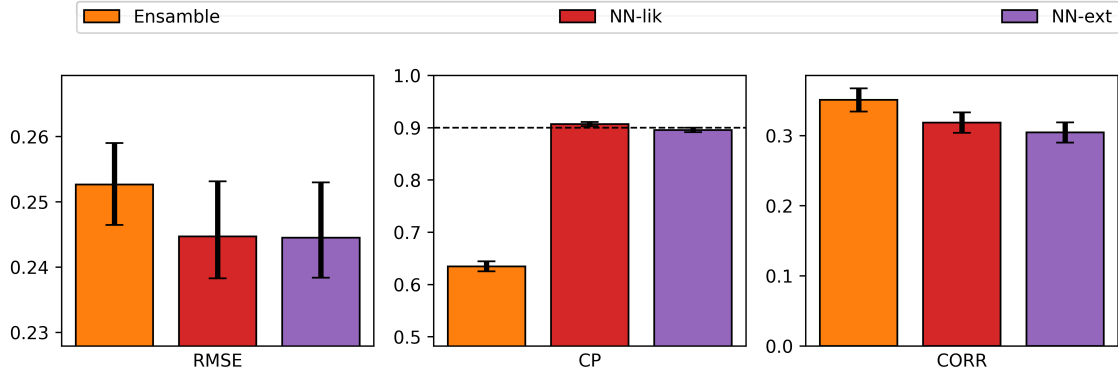


Figura 4.12: Métricas correspondientes al pronóstico por ensamble (barra naranja), y al RNA-lik (barra roja) y al RNA-ext (barra morada) para el pronóstico de presión a nivel del mar para el error cuadrático medio (panel derecho), la probabilidad de cobertura del 90 % (panel central) y el coeficiente de correlación entre la desviación estándar del error y el error absoluto (panel izquierdo). Los segmentos negros en la parte superior de las barras indican el intervalo de confianza del 95 % para las diferentes métricas, calculados mediante la técnica de bootstrap. La línea horizontal del panel central indica el valor perfecto asociado a la probabilidad de cobertura del 90 %.

Las RNAs también muestran un histograma PIT (ver Fig. 4.13) mucho más plano con valores de χ^2 de 0,15 tanto para la RNA-lik como para la RNA-ext frente a 1,03 para el pronóstico por ensamble. Si bien el ensamble tiene un coeficiente de correlación mas alto que las redes, los tres valores son muy similares. También se observa que la red RNA-lik da resultados ligeramente mejores que la RNA-ext, como en los casos sintéticos.

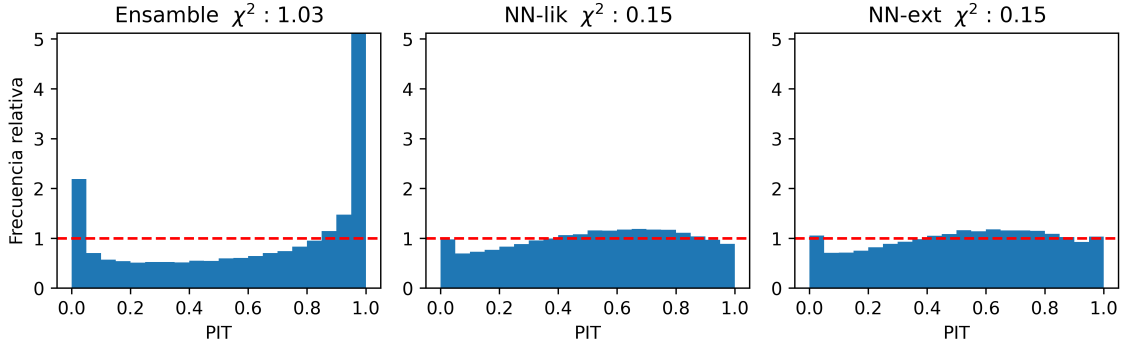


Figura 4.13: Histogramas PIT para el pronóstico por ensambles (panel izquierdo), para el pronóstico RNA-lik (panel central) y RNA-ext (panel derecho). En la parte superior de cada panel se indica el parámetro de planitud χ^2 correspondiente.

4.6. Conclusiones

En general, los resultados obtenidos en este capítulo en un entorno experimental sencillo de prueba de concepto son bastante alentadores en cuanto a la posibilidad de aumentar la precisión de un pronóstico determinístico y proporcionar una estimación fiable de su incertidumbre con un costo computacional mucho menor que el requerido por un pronóstico por ensamble. En particular, mostramos que las RNAs son capaces de filtrar parte de la incertidumbre asociada a los modos imprevisibles, especialmente en el régimen de crecimiento no lineal del error. En este aspecto, las RNA son capaces de emular el efecto de filtrado de la media del ensamble, aunque con menor eficacia. Las RNAs también son capaces de representar con precisión la desviación típica del error de pronóstico dependiente del estado y resultante de la amplificación caótica de los errores de las condiciones iniciales. Se encuentra una subestimación sistemática de la desviación estándar del pronóstico en tiempos cortos en las RNAs debido a nuestra estrategia de entrenamiento en la que el análisis se utiliza como un proxy del verdadero estado del sistema.

Cuando el modelo es imperfecto, las RNAs son capaces de estimar el componente de error sistemático dependiente del estado asociado al efecto de los errores del modelo. Este resultado es especialmente alentador, ya que la representación adecuada de los errores del modelo y su impacto en el pronóstico sigue siendo un importante reto científico. La red entrenada directamente con un conjunto de pronósticos obtiene mejores resultados en el escenario de modelo perfecto. Sin embargo, los enfoques de entrenamiento indirecto (RNA-ext y RNA-lik) producen mejores resultados en el escenario de modelo imperfecto.

Esto se debe principalmente a que, en el entrenamiento directo, la red hereda la estimación de la incertidumbre lograda por el ensamble y, dado que los errores del modelo suelen estar subrepresentados, esto puede conducir a peores resultados en comparación con una red que se entrena indirectamente y es capaz de aprender el efecto de los errores del modelo a partir de los datos.

En los experimentos con un sistema de pronósticos meteorológicos por ensambles del estado del arte, los resultados muestran un rendimiento similar al obtenido en los experimentos con modelos simples, lo que sugiere que los principales aspectos de la metodología pueden extenderse a sistemas más complejos con algunas precauciones. Un aspecto importante relacionado con la extensión de estas metodologías a sistemas de última generación es el aumento de la dimensión de las entradas y las salidas. En nuestros experimentos utilizamos arquitecturas de red simples que no escalan bien con el tamaño de la entrada-salida de la red y de las capas ocultas. Para llevar a cabo una cuantificación de la incertidumbre basada en RNA con modelos de mayor resolución en dominios más amplios, deberían considerarse otras arquitecturas, como las redes convolucionales profundas (por ejemplo [Grönquist et al. \(2021\)](#); [Chapman et al. \(2022\)](#)).

Capítulo 5

Estimación de la matriz de covarianza

Conocer las varianzas asociadas a los pronósticos de las variables de estado de un sistema dinámico es de gran ayuda a la hora de entender e interpretar correctamente un pronóstico y tomar decisiones en consecuencia. Pero otras aplicaciones, como la asimilación de datos, en inglés Data Assimilation (DA), requieren conocer más en profundidad la estructura de la incertidumbre, en particular, cómo las variables de estado se correlacionan entre sí. Esta interdependencia, como ocurría con la varianza, es dependiente del estado y en consecuencia no es trivial estimarla, sobre todo en problemas de muy alta dimensionalidad. También, al igual que la varianza, los ensambles de modelos numéricos son la metodología más robusta y más usada para estimar una matriz de covarianza del error de pronóstico (MCEP). En este capítulo analizaremos distintos métodos para estimar una MCEP dependiente del estado, especialmente usando redes neuronales artificiales (RNAs). También presentaremos una metodología para extender la función de costo basada en el error, en inglés extended Mean Squared Error (eMSE) propuesta en el capítulo 3 para estimar MCEP completas.

Además de los ensambles, hemos comprobado en los capítulos anteriores que las RNAs son una alternativa válida para estimar vectores de varianza del error del pronóstico. Entendiendo al vector de varianza como la diagonal principal de una MCEP, se podría pensar que el problema de estimar una MCEP ya está en parte resuelto, o al menos, que es posible extender las metodologías ya analizadas a MCEP. En la literatura se propusieron algunos métodos para estimar la covarianza de los errores y su dependencia con el estado utilizando técnicas de machine learning. [Ouala et al. \(2018\)](#) utilizaron una red neuronal artificial (RNA) y la función de costo basada en la verosimilitud gaussiana, del inglés Likelihood (LKHD) para estimar una covarianza diagonal del error en un subes-

pacio definido por las componentes principales (EOF del inglés Empirical Orthogonal Function) del error en las variables de estado. Asumiendo que la matriz de covarianza es siempre diagonal en este espacio, este método permite estimar la covarianza completa del error de los pronósticos en el espacio de fase. Ouala et al. (2018) aplicaron esta metodología al pronóstico de la temperatura del océano asimilando observaciones satelitales. Para lidiar con el problema de la alta dimensionalidad, dividen el dominio espacial en subdominios que son procesados individualmente. Las EOF del error son calculadas de forma independiente sobre cada subdominio. Es decir, el estado del sistema en cada subdominio $\mathbf{x}_t^{\text{P}_s}$ (donde s es un identificador del subdominio y t es el tiempo) es proyectado al espacio de las EOF $\mathbf{z}_t^{\text{P}_s} = \mathbf{x}_t^{\text{P}_s} \mathbf{T}^\top$ usando la matriz de transformación $\mathbf{T}$. Proyectar los estados en el espacio de las componentes principales del error tiene dos ventajas fundamentales. La primera es realizar una nueva reducción de dimensionalidad obteniendo un nuevo estado proyectado $\mathbf{z}_t^{\text{P}_s}$ de menor dimensión que $\mathbf{x}_t^{\text{P}_s}$. La segunda ventaja es que, en el espacio transformando, los errores en principio no están correlacionados y la MCEP es una matriz diagonal donde sólo los elementos correspondientes a las varianzas son distintos de cero. Los autores entrenan una red neuronal en el espacio transformado para estimar el vector de varianzas $\tilde{\sigma}^2(\mathbf{z}_t^{\text{P}_s})$ usando la función LKHD (Ec. 3.3). Finalmente recuperan la MCEP en el espacio de estado anti-transformando el vector de varianzas $\tilde{\Sigma}(\mathbf{x}_t) = \mathbf{T}^\top \tilde{\sigma}^2(\mathbf{z}_t^{\text{P}_s}) \mathbf{T}$

Los resultados de su experimentación muestran una mejora en la interpolación de la estructura de pequeña escala comparado con otros métodos evaluados como la técnica de análogos o la interpolación óptima, en inglés Optimal Interpolation (OI). Esta última no tiene en cuenta la variabilidad de la covarianza con el estado y, en cambio, utiliza una covarianza climatológica.

Ouala et al. observan que es importante mejorar la formulación y el entrenamiento del modelo de covarianza, ya que consideran que podría no escalar adecuadamente en un contexto de asimilación secuencial cuando se dispone de observaciones con un muestreo temporal muy irregular. Y concluyen que la optimización de modelos de covarianza en función del muestreo espacio-temporal de observaciones parece ser una vía interesante para continuar investigando.

La metodología planteada en Ouala et al. (2018) es muy interesante porque logra estimar una MCEP utilizando la misma metodología que estuvimos analizando en los capítulos anteriores, más una transformación de espacios. Pero esto significa que el grado de libertad efectivo (cantidad de varianzas en el espacio transformado) es mucho menor que la cantidad de elementos que se deben estimar para obtener una MCEP completa en el

espacio de estado, y esto necesariamente impone restricciones a las posibles soluciones que la metodología puede estimar.

5.1. Estimación de la covarianza usando verosimilitud

La estimación de una MCEP completa a partir de datos se ha investigado en otros contextos. Williams (1996) utilizó una red neuronal para estimar los parámetros de una distribución gaussiana multivariada. Hu and Kantor (2015) presentaron una predicción paramétrica de la covarianza para ruido heterocedástico y Liu et al. (2018) implementaron un modelo de aprendizaje profundo para la inferencia de la matriz de covarianza de los errores de observación y lo utilizaron en la estimación de la posición para aplicaciones de navegación. En estos casos, se estima una descomposición de Cholesky de la matriz de covarianza.

Williams (1996) extiende la propuesta de Nix and Weigend (1994) (ver Cap. 3) de entrenar una RNA usando la verosimilitud gaussiana, pero esta vez considerando una distribución multivariada del error del pronóstico. Asumiendo que el error es gaussiano, propone entrenar una RNA que estime, en función del estado, la media $\tilde{\mu}(x)$ y la covarianza $\tilde{\Sigma}(x)$ de la distribución dada por:

$$P(y|x, \tilde{\mu}, \tilde{\Sigma}) = (2\pi)^{-\frac{n}{2}} |\tilde{\Sigma}|^{-\frac{1}{2}} \exp \left[-\frac{(y - \tilde{\mu})^\top \tilde{\Sigma}^{-1} (y - \tilde{\mu})}{2} \right] \quad (5.1)$$

Para el caso de un vector de estados $x \in \mathbb{R}^n$ tendremos las salidas $\tilde{\mu} \in \mathbb{R}^n$ y $\tilde{\Sigma} \in \mathbb{R}^{n \times n}$. La ecuación 5.1 es similar a la ecuación 3.2, esta última usaba un vector varianza ($\tilde{\sigma}^2$) que ahora es una MCEP ($\tilde{\Sigma}$). Siguiendo el procedimiento del Cap. 3 y aplicando el logaritmo negativo a la verosimilitud y descartando constantes, la función de costo a minimizar resulta en:

$$\mathcal{L} = \log(|\tilde{\Sigma}|) + (y - \tilde{\mu})^\top \tilde{\Sigma}^{-1} (y - \tilde{\mu}) \quad (5.2)$$

La covarianza ($\tilde{\Sigma}$) debe cumplir una condición de simetría que se puede imponer estimando los $n(n+1)/2$ coeficientes de la parte triangular superior (o inferior) de la matriz. También se debe garantizar que sea invertible y definida positiva.

Para resolver el problema de la invertibilidad y la positividad, Williams (1996) propone utilizar la factorización de Cholesky de una matriz simétrica definida positiva donde

$\tilde{\Sigma} = \mathbf{A}^\top \mathbf{A}$ con $\mathbf{A}$ una matriz triangular superior y con elementos estrictamente positivos en la diagonal. Recíprocamente, si $\mathbf{A}$ es una matriz triangular superior con entradas diagonales estrictamente positivas, entonces $\mathbf{A}^\top \mathbf{A}$ es simétrica definida positiva. Además, la inversa de una matriz definida positiva es definida positiva. Entonces, para evitar invertir la MCEP durante el entrenamiento, propone que la red estime la matriz de precisión $\mathbf{B} = \tilde{\Sigma}^{-1}$ y en particular, $\mathbf{B} = \tilde{\Sigma}^{-1} = \mathbf{A}^\top \mathbf{A}$. Además, sabiendo que el determinante de $\mathbf{A}^\top \mathbf{A}$ es el producto de los elementos de la diagonal de $\mathbf{A}$, podemos reescribir la Ec. 5.2 como:

$$\mathcal{L} = - \left(\sum_{i=0}^N \log A_{ii} \right) + (\mathbf{y} - \tilde{\boldsymbol{\mu}})^\top \mathbf{A}^\top \mathbf{A} (\mathbf{y} - \tilde{\boldsymbol{\mu}}) \quad (5.3)$$

De forma similar, en [Hu and Kantor \(2015\)](#) utilizan la descomposición *modificada de Cholesky* que descompone una matriz definida positiva en el producto de una matriz triangular inferior $\mathbf{L}$ restringida a tener sólo unos en la diagonal y una matriz diagonal $\mathbf{D}$ restringida a tener sólo valores positivos, entonces $\mathbf{B} = \tilde{\Sigma}^{-1} = \mathbf{A}^\top \mathbf{A} = \mathbf{L} \mathbf{D} \mathbf{L}^\top$.

El tratamiento de [Hu and Kantor \(2015\)](#) permite interpretar, en el contexto de las covarianzas, a $\mathbf{D}$ como una MCEP transformada por $\mathbf{L}$. En este sentido, se asemeja a la idea de [Ouala et al.](#) que estima una matriz diagonal que es transformada del espacio EOF por la matriz $\mathbf{T}$ fija. Pero en este caso, la matriz de transformación $\mathbf{L}$ es dependiente del estado. Esta descomposición también nos permite identificar fácilmente las salidas de la red correspondientes a los elementos de la diagonal ($\mathbf{D}$) y forzarlos a que éstas sean siempre positivas. Utilizando la descomposición de Cholesky modificada, se puede reescribir la función de costo de la siguiente manera:

$$\mathcal{L} = - \left(\sum_i \log D_{ii} \right) + (\mathbf{y} - \tilde{\boldsymbol{\mu}})^\top \mathbf{L} \mathbf{D} \mathbf{L}^\top (\mathbf{y} - \tilde{\boldsymbol{\mu}}) \quad (5.4)$$

que es la función de costo que utilizan en [Hu and Kantor \(2015\)](#) y [Liu et al. \(2018\)](#) para estimar matrices de covarianza de error en odometría (sensores de rango y sensores visuales respectivamente). En ambos trabajos se realizan experimentos con sensores simulados (datos sintéticos) y con datos reales y, en todos los casos, la estimación de la MCEPs mejora la representación del ruido en los sensores (observaciones), resultando esto en una mejor precisión de la localización del vehículo.

En sus conclusiones, [Hu and Kantor \(2015\)](#) advierten que extender sus resultados a altas dimensiones requiere de más investigación, en particular para la representación y manipulación de grandes matrices de covarianzas o matrices de covarianzas esparzas.

5.2. Estimación de la covarianza usando análogos

Hasta ahora se discutieron técnicas basadas en la verosimilitud gaussiana para entrenar redes neuronales capaces de estimar la MCEP. Otra forma de estimar la incertidumbre es mediante la técnica de análogos ([Lguensat et al., 2017](#)). En esta estrategia se utiliza un conjunto de estados *similares* del sistema dinámico para emular la dinámica del sistema. Para esto es necesario contar con un repositorio o catálogo de estados históricos del sistema, ya sean observaciones o el resultado de integrar un modelo numérico durante un período prolongado de tiempo. El catálogo de referencia está conformado por pares de vectores de estado consecutivos, separados todos por el mismo plazo temporal dt .

Dado un catálogo $\mathcal{C}$, se define el operador de pronóstico por análogos $\mathcal{A}$ como un emulador estadístico del modelo no lineal, basado en ejemplares del estado $\mathbf{x}$ al tiempo t y los estados correspondientes al tiempo $t + dt$ (sucesores). Es decir que, para cualquier estado $\mathbf{x}_t$, emulamos el siguiente estado en el tiempo $t + dt$ basándonos en los sucesores de sus vecinos más próximos en el catálogo $\mathcal{C}$ al tiempo t . Esta misma idea puede ser extendida a la construcción de un pronóstico por ensambles a partir del cual se puede obtener la MCEP calculando la covarianza empírica de igual forma que se realiza en los ensambles basados en múltiples integraciones de un modelo numérico. Este método es puramente basado en datos en el sentido que no se necesita integrar un modelo numérico para realizar el pronóstico del estado y de la covarianza.

[Lguensat et al.](#) proponen el método de asimilación de datos por análogos, en inglés Analog Data Assimilation (AnDA) y evalúan su desempeño comparando distintas opciones de operadores de pronóstico $\mathcal{A}$ y usando como línea base de comparación otras técnicas de DA en un entorno de prueba basado en simulaciones del modelo Lorenz'63 y Lorenz'96.

El pilar fundamental del enfoque de pronósticos basados en análogos es el catálogo. Como tal, el método AnDA se relaciona profundamente con la calidad y representatividad del catálogo. Así, los autores explican que la aplicación de AnDA a sistemas de alta dimensión es otro desafío futuro. Como se detalla en [van den Dool \(1994\)](#), el número de elementos de un catálogo deberá crecer exponencialmente con la dimensión intrínseca del estado para garantizar la recuperación de análogos con una precisión determinada.

Esto hace poco realista la aplicación directa de estrategias por análogos en un espacio de estados con una dimensionalidad intrínseca superior a 10. En consecuencia, los $\mathcal{A}$ globales (vector de estado completo) son probablemente inadecuados para los sistemas de alta dimensionalidad. Por el contrario, $\mathcal{A}$ locales permiten descomponer el pronóstico del estado de alta dimensionalidad en una serie de operaciones locales y de baja dimensionalidad de forma similar a la propuesta de Ouala et al. de dividir el dominio en subdominios más pequeños.

En Grooms (2021) se plantea una metodología alternativa al uso de un catálogo para la obtención de estados análogos. En este trabajo se propone construir los análogos usando autoencoders variacionales, del inglés Variational Auto Encoder (VAE) (Kingma and Welling, 2019). Grooms se basa en que un conjunto de análogos proceden de una región pequeña en el atractor del sistema dinámico que está cerca o incluye al verdadero estado del sistema, y por lo tanto, la variabilidad entre los análogos representa las varianzas y covarianzas correspondientes a ese sector del atractor (es decir la matriz de covarianza del estado correspondiente a un entorno de un estado determinado del sistema). En consecuencia, se espera que proporcionen una representación aproximada de la dependencia de la matriz de covarianza de los errores con el estado del sistema. Este planteo remite a la figura 3.5 que compara el conjunto de entrenamiento de las RNAs con un ensamble de modelos dinámicos. En este sentido, la técnica de análogos al utilizar datos históricos para realizar estimaciones es más afín a las RNAs que a los ensambles. La diferencia es que la cantidad, selección y combinación de análogos se realiza de forma explícita por el operador $\mathcal{A}$ mientras que en las RNAs esto se determina de forma implícita a través de los pesos de la red aprendidos durante el entrenamiento.

Grooms explora una nueva forma de obtener estados análogos utilizando VAE, que a diferencia del autoencoder estándar, impone una estructura particular en el espacio latente.

Supongamos que $\mathbf{z}_i = \text{ENC}(\mathbf{x}_i)$ es una representación del estado $\mathbf{x}_i$ en el espacio latente y que $\text{DEC}(\mathbf{z}_i + \boldsymbol{\varepsilon})$ es la decodificación de ese mismo estado más una pequeña perturbación ($\boldsymbol{\varepsilon}$ pequeño) en ese espacio latente. En un autoencoder estándar, $\text{DEC}(\mathbf{z}_i + \boldsymbol{\varepsilon})$ y $\mathbf{x}_i$ no tienen por qué ser similares y, en consecuencia, no siempre son útiles como modelos generativos.

Los VAE agregan una restricción adicional que implica que se eligen los pesos de las redes ENC y DEC para que la distribución de los datos en el espacio latente sea aproximadamente gaussiana. Para lograr esto, se diseña la arquitectura de la red de manera que el espacio latente se divide en dos, de modo que $\text{ENC}(\mathbf{x}) = (\tilde{\boldsymbol{\mu}}, \tilde{\boldsymbol{\sigma}})$ con $\tilde{\boldsymbol{\mu}}, \tilde{\boldsymbol{\sigma}} \in \mathbb{R}^l$, y l es

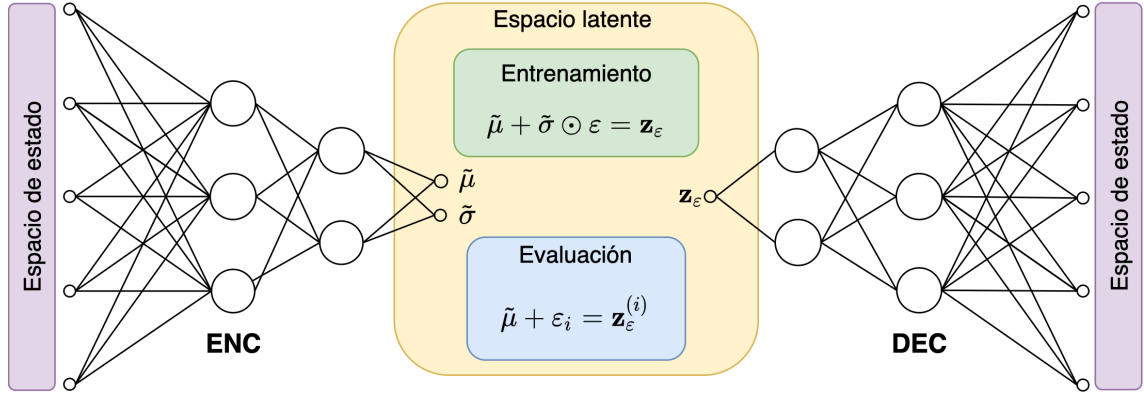


Figura 5.1: Representación de un VAE entrenado para generar análogos a partir de un estado determinístico en la entrada. Las redes **ENC** y **DEC** transforman el vector de estado hacia y desde el espacio latente respectivamente. Durante el entrenamiento se utiliza $\varepsilon \sim \mathcal{N}(0, 1)$ para garantizar una estructura gaussiana en el espacio latente. Los estados análogos se construyen decodificando estados perturbados ($\varepsilon_i \sim \mathcal{N}(0, 1)$) en el espacio latente.

la dimensión del estado en el espacio latente, como muestra la Fig. 5.1. Luego, durante el entrenamiento, se construye un vector $\mathbf{z}_\varepsilon = \tilde{\boldsymbol{\mu}} + \tilde{\boldsymbol{\sigma}} \odot \boldsymbol{\varepsilon}$, donde $\boldsymbol{\varepsilon}$ es un vector aleatorio con distribución normal estándar y $\odot$ denota el producto elemento a elemento o de Hadamard. Así, $\mathbf{z}_\varepsilon$ se usa dentro de la función de costo como un término adicional de regularización para penalizar las desviaciones de la distribución estimada en el espacio latente de una distribución normal estándar. Finalmente, cuando el VAE está entrenado, los análogos se construyen mediante perturbaciones en el estado latente. Es decir, se codifica un estado latente $(\tilde{\boldsymbol{\mu}} + \tilde{\boldsymbol{\sigma}})$ a partir de un pronóstico $\mathbf{x}$, y cada miembro del ensamble $\mathbf{x}^{(i)}$ se construye decodificando perturbaciones $\tilde{\mathbf{z}}^{(i)} = \tilde{\boldsymbol{\mu}} + \boldsymbol{\varepsilon}_i$ con $i = 1, \dots, N_e$ y N_e es el tamaño del ensamble y donde $\boldsymbol{\varepsilon}_i$ son muestras independientes de una distribución normal estándar. Notar que el vector $\tilde{\boldsymbol{\sigma}}$ producido por el codificador no se utiliza durante este proceso.

Grooms reconoce que entrenar un VAE para reproducir un estado completo de un modelo geofísico es probablemente muy difícil y puede ser prácticamente imposible debido a la alta dimensionalidad del problema. Pero sugiere utilizar una metodología local en la línea del filtro de Kalman local por ensambles (Brusdal et al., 2003; Evensen, 2003; Ott et al., 2004) (LEnKF). Este enfoque analiza, para cada punto de retícula del modelo, un ensamble y las observaciones cercanas a dicho punto. **Grooms** sugiere que su método

podría utilizarse fácilmente en este contexto generando, para cada punto de retícula, un ensamble de análogos. La ventaja de este enfoque local es que el VAE sólo se entrena para generar ensambles de una representación parcial (local) del estado del sistema.

5.3. Estimación de la covarianza usando el error

En esta sección, extenderemos la función eMSE (sección 3.4.1) para estimar MCEP completas. Para eso partiremos nuevamente del error del pronóstico ϵ (Eq. 3.14) y nos concentraremos solamente en el término T_2 correspondiente a la estimación del segundo momento de la función de densidad de probabilidad, en inglés Probability Density Function (PDF) en la función eMSE (Eq. 3.16).

Al igual que para la Ec. 3.14, supondremos que se dispone de los estados analizados medios para aproximar el error de pronóstico ($\hat{\epsilon} \sim \epsilon$). En la sección 6.2.4 se analizan otras opciones para $\hat{\epsilon}$ que serán evaluadas en forma experimental.

Así, $\hat{\epsilon}$ puede utilizarse para construir una MCEP puntual como:

$$\hat{\Sigma}_k = \hat{\epsilon}_k \hat{\epsilon}_k^\top = (\bar{\mathbf{x}}_k^a - \tilde{\boldsymbol{\mu}}_k)(\bar{\mathbf{x}}_k^a - \tilde{\boldsymbol{\mu}}_k)^\top. \quad (5.5)$$

donde $\hat{\Sigma}_k$ es la estimación de la MCEP basada únicamente en la muestra k . Llamaremos $\tilde{\Sigma}$ a las salidas de la RNA que estimarán la covarianza de la PDF del error de manera análoga a la varianza en la figura 3.6

Entonces, el término de covarianza T_{cov} queda definida como el cuadrado de la norma de Frobenius entre la salida de la RNA ($\tilde{\Sigma}_k$), y la aproximación de la MCEP ($\hat{\Sigma}_k$),

$$T_{cov} = \left\| \tilde{\Sigma}_k - \hat{\epsilon}_k(\hat{\epsilon}_k)^\top \right\|_F^2 = \sum_{i,j=0}^{N_x, N_x} (\tilde{\Sigma}_k^{(i,j)} - [\hat{\epsilon}_k(\hat{\epsilon}_k)^\top]^{(i,j)})^2 = \text{MSE}(\tilde{\Sigma}_k, \hat{\Sigma}_k) \quad (5.6)$$

y finalmente la función de costo resulta:

$$\mathcal{L}_{eMSE_{cov}} = \alpha_1 T_1 + \alpha_2 T_{cov} \quad (5.7)$$

Una ventaja fundamental de esta definición es que nos permita estimar la MCEP directamente de la salida de la RNA (a diferencia de la función LKHD_{cov} que estimaba la matriz de precisión), y no involucra la inversión de una matriz en su formulación. Esta característica tiene la ventaja adicional de fácilmente entrenar sólo un subconjunto de los elementos de la MCEP, por ejemplo las n primeras diagonales, a este proceso se lo

denomina localización y se analizará en el siguiente capítulo.

5.4. Análisis comparativo

Se realizará una prueba sencilla para evaluar la habilidad de las RNAs para estimar la MCEP. En particular se evaluará el desempeño de la función eMSE_{cov} (Eq. 5.7) usando como línea de base de comparación la función LKHD_{cov} (Eq. 5.3).

5.4.1. Experimentación

Evaluaremos la estimación de covarianza en un escenario de modelo perfecto (EMP) similar al descrito en la Sec. 4.4.2 usando el modelo dinámico de Lorenz'63 de tres variables que está definido por el siguiente sistema de ecuaciones diferenciales ordinarias, del inglés Ordinary Differential Equations (ODE):

$$\begin{aligned}\frac{\partial x}{\partial t} &= \sigma(y - x) \\ \frac{\partial y}{\partial t} &= x(\rho - z) - y \\ \frac{\partial z}{\partial t} &= xy - \beta z.\end{aligned}\tag{5.8}$$

Para nuestro experimento, se utilizaron los siguientes valores de los parámetros: $\{\beta = \frac{8}{3}; \rho = 28; \sigma = 10\}$. Entonces integraremos una simulación *verdad* con un paso temporal $dt = 0,01$ durante $N = 50,000$ pasos de tiempo. Cada uno de los vectores de estado $(\mathbf{x}_k^t, k \in \{1, \dots, N\})$ es observado con un error de observación gaussiano tal que $x_k^o = \mathcal{N}(x_k^t, \frac{1}{2})$. Así, se genera un ensamble de condiciones iniciales que son utilizadas para inicializar un pronóstico de 25 pasos de tiempo de plazo.

Finalmente, para estimar la MCEP se entrenará una RNA con dos capas ocultas completamente conectadas de 75 neuronas cada una. La capa de entrada tiene 3 neuronas correspondientes a las 3 variables de estado del sistema (x, y, z) y, en este caso, como sólo nos interesa estimar la MCEP, la capa de salida consiste en 6 neuronas (3 varianzas y 3 covarianzas). Para el entrenamiento se utilizó un conjunto de 30.000 muestras y para la verificación se usaron 5.000 muestras. Para evaluar los resultados se utilizó un conjunto de prueba de 10.000 muestras. Los tres conjuntos (entrenamiento, verificación y pruebas) consisten en muestras disjuntas para garantizar la independencia de los resultados.

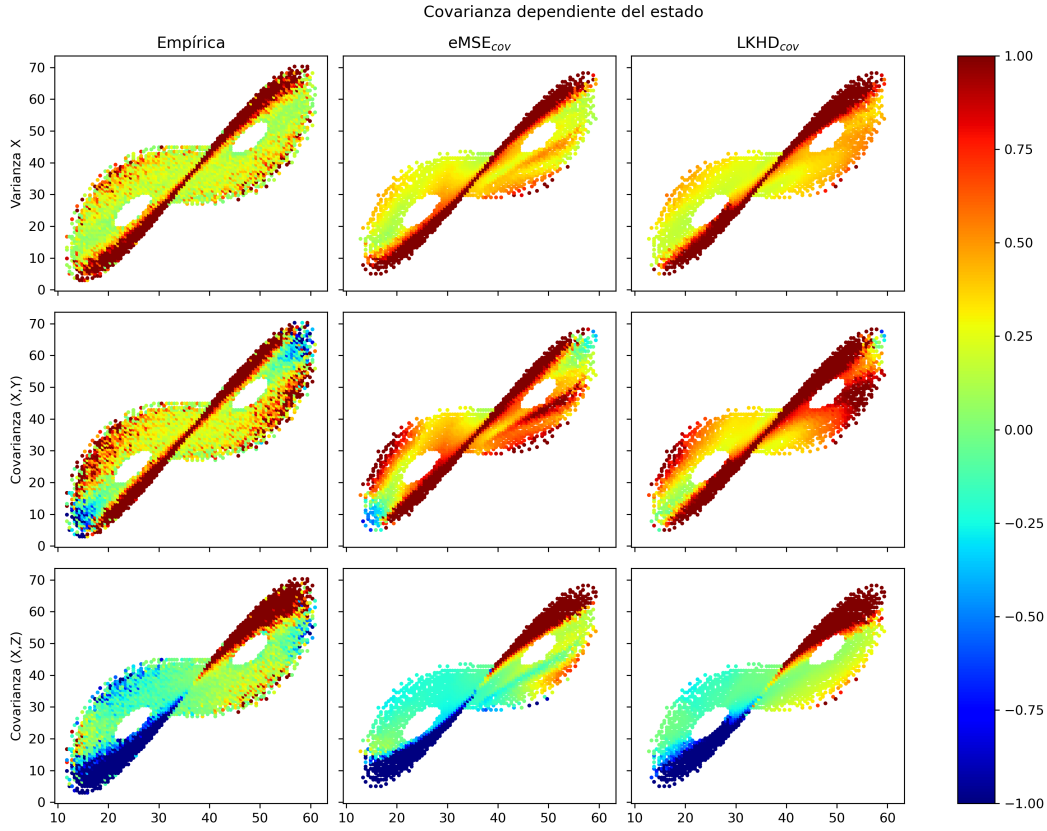


Figura 5.2: Comparación de los elementos de las MCEPs en función del estado proyectado sobre el plano xy estimada de forma empírica (columna izquierda), usando una RNA entrenada con la función de costo $eMSE_{cov}$ basada en los errores (columna central) y usando una RNA entrenada con la función de costo $LKHD_{cov}$ basada en la verosimilitud (columna derecha).

5.4.2. Resultados

La Fig. 5.2 muestra en cada fila los valores de covarianzas asociados al estado estimados de manera empírica (paneles en columna izquierda), la función $eMSE_{cov}$ (paneles en columna central) y $LKHD_{cov}$ en los paneles de la columna derecha. Para generar cada panel, se discretizó el rango de valores de estado de las variables x e y en 100×100 celdas. En el caso de las redes (paneles en la segunda y tercera columna), para cada celda se calculó el valor medio de la covarianza estimada por la red en todas las trayectorias que atraviesan dicha celda. En el caso de la estimación empírica (paneles de la primera

columna) se calcula directamente los elementos de la matriz de covarianza utilizando los errores de todos los pronósticos correspondientes a las trayectorias que atraviesan cada una de las celdas.

Los paneles muestran una clara dependencia de la MCEP con el estado del sistema para las tres técnicas de estimación. La estructura general de esta dependencia está bien representada en las tres técnicas, aunque las RNAs muestran una variabilidad en la covarianza más suave comparado con la estimación empírica. Esto podría deberse en parte a errores de muestreo sobre todo en regiones donde no hay tanta densidad de trayectoria. Las RNAs también parecen ser menos sensibles a los valores extremos comparado con la estimación empírica. Por ejemplo, para la covarianza (xy) cerca de los extremos del atractor, alrededor de los puntos $(15,10)$ y $(60,60)$, se ven valores extremos negativos (azules) en la estimación empírica que no están reflejados con la misma intensidad en las RNAs. Lo mismo ocurre para la covarianza (xz) alrededor del punto $(25,35)$

5.4.3. Conclusiones

Pudimos verificar que ambas funciones de costo ($LKHD_{cov}$ y $eMSE_{cov}$) pueden estimar una MCEP conservando su dependencia con el estado y manteniendo una estructura similar a la obtenida de forma empírica. Se mostró cómo, a partir de la descomposición de Cholesky, podemos garantizar que la MCEP estimada por la red es definida positiva. Esto agrega tiempo de cómputo, principalmente durante el entrenamiento, puesto que la MCEP debe ser reconstruida a partir de la salida de la red. Las RNAs entrenadas con la función de costo $LKHD_{cov}$, como predicen la matriz de precisión, exigen un paso adicional de post procesamiento que implica invertir la matriz de precisión para obtener la de covarianza. En muy alta dimensionalidad, creemos que esta inversión de matriz puede acarrear problemas de inestabilidad numérica. Esto pareciera una clara ventaja para la función de costo $eMSE_{cov}$ que estima directamente la covarianza y permite, además, entrenar solo los elementos de la covarianza que se desean estimar reduciendo la cantidad de parámetros entrenables de la red y, al mismo, tiempo localizando la MCEP desde su construcción. Esto no es posible de realizar con la función de costo $LKHD_{cov}$ al menos de forma directa ya que, aunque solo se estime una matriz banda para reducir la cantidad de parámetros, la inversa de una matriz banda no es una matriz banda y, por lo tanto, no es tan directo localizar la matriz.

Capítulo 6

Asimilación de datos

Un aspecto clave para la asimilación de datos, en inglés Data Assimilation (DA), es la cuantificación de la incertidumbre del pronóstico. Los sistemas de DA basados en ensambles utilizan un conjunto de simulaciones para estimar la incertidumbre en los pronósticos. Sin embargo, este enfoque es exigente en términos de cálculo, diseño y desarrollo.

En el capítulo anterior se analizaron métodos alternativos al ensamble de modelos para estimar la matriz de covarianza del error de pronóstico (MCEP) utilizando una única integración del modelo. En particular se estudió el uso de redes neuronales artificiales (RNAs) entrenadas mediante una función de costo que tiene en cuenta el hecho de que los errores del pronóstico son heterocedásticos. En este capítulo presentaremos un método híbrido de asimilación de datos basado en el filtro de Kalman y usando redes neuronales (UnnKF) que combina una actualización de análisis basado en el filtro de Kalman, del inglés Kalman Filter (KF), y una estimación de la MCEP basada en una red neuronal artificial (RNA). Para evaluar esta metodología, se realizarán experimentos de simulación de sistemas de observación utilizando el modelo de Lorenz'96.

Este capítulo se estructura de la siguiente manera. En la sección 4.2 se describe la metodología híbrida de asimilación de datos. El diseño de los experimentos para evaluar el rendimiento de la metodología de asimilación se describe en la Sección 4.1. En la Sección 6.3 se analizan los resultados obtenidos. Y finalmente, en la sección 6.4 se comentan las conclusiones obtenidas.

6.1. Método híbrido de asimilación de datos (UnnKF)

En la sección 2.2.1 se introdujo la metodología secuencial de DA. En esta sección, definiremos un método híbrido que utiliza la MCEP estimada con RNA a partir de un pronóstico determinístico realizado con un modelo físico (Cap. 5).

Las ecuaciones 2.8 se utilizan para calcular un análisis combinando pronósticos y observaciones. Nosotros proponemos aproximar el pronóstico $\mathbf{x}_k^f$ de la ecuación 2.8a por el pronóstico corregido $\tilde{\boldsymbol{\mu}}_k^f$ y la incertidumbre asociada $\mathbf{P}_k^f$ de la ecuación 2.8b por la MCEP estimada por una RNA ($\mathbf{P}_k^f \approx \tilde{\boldsymbol{\Sigma}}_k^f$). Entonces, las ecuaciones del método híbrido para calcular un análisis son:

$$\mathbf{x}_k^a = \tilde{\boldsymbol{\mu}}_k^f + \mathbf{K}(\mathbf{y}_k - \mathcal{H}(\tilde{\boldsymbol{\mu}}_k^f)), \quad (6.1a)$$

$$\mathbf{K} = \tilde{\boldsymbol{\Sigma}}_k^f \mathbf{H}^\top (\mathbf{H} \tilde{\boldsymbol{\Sigma}}_k^f \mathbf{H}^\top + \mathbf{R}_k)^{-1}. \quad (6.1b)$$

La figura 6.1 muestra una representación esquemática de UnnKF. Cada ciclo comienza con la integración del modelo dinámico inicializado con el análisis del ciclo anterior $\mathbf{x}_{k-1}^a$ para generar un pronóstico determinístico $\mathbf{x}_k^f = \mathcal{M}(\mathbf{x}_{k-1}^a)$. El pronóstico y su condición inicial (el análisis) se utilizan como entradas a la red neuronal para obtener una corrección del pronóstico y una estimación de la MCEP $(\tilde{\boldsymbol{\mu}}_k^f, \tilde{\boldsymbol{\Sigma}}_k^f) = \mathcal{F}_{RNA}(\mathbf{x}_{k-1}^a, \mathbf{x}_k^f)$. Entonces la salida de la RNA se utiliza junto con los datos de observaciones ($\mathbf{y}_k$ y $\mathbf{R}_k$) para generar un nuevo análisis mediante las ecuaciones 6.1. El nuevo análisis se utiliza como condición inicial para producir un nuevo pronóstico cerrando el ciclo de asimilación secuencial.

Este enfoque usa una única integración del modelo dinámico para evolucionar el estado del sistema del tiempo $k - 1$ al tiempo k como en el método de interpolación óptima, en inglés Optimal Interpolation (OI), o el método variacional tridimensional (3Dvar). No obstante, a diferencia de estos últimos, utiliza una estimación de la MCEP dependiente del estado como en el filtro de Kalman por ensambles, en inglés Ensemble Kalman Filter (EnKF), pero sin necesidad de integrar un ensamble de pronósticos.

El análisis obtenido a través de la Ec. 6.1 es muy sensible a la calidad de la MCEP. Por ejemplo, si los términos diagonales están sobreestimados, el análisis tiende a sobreajustarse a las observaciones. Por otra parte, si los elementos fuera de la diagonal no están bien estimados, la información de las observaciones no se propaga adecuadamente a las variables del sistema que no están siendo directamente observadas. En los enfoques OI o

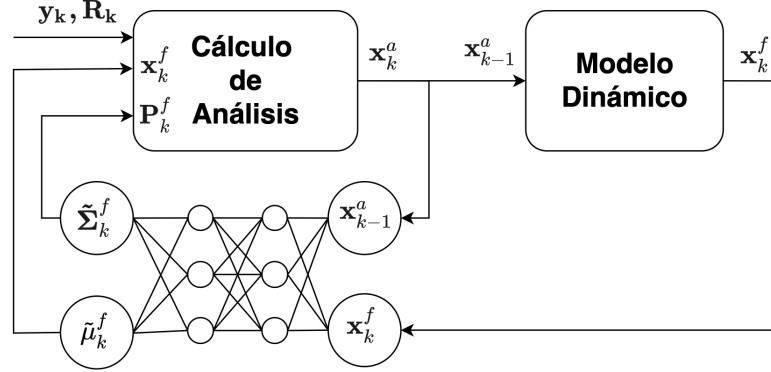


Figura 6.1: Representación esquemática de UnnKF (véase el texto para más detalles).

3Dvar, se supone que la covarianza del pronóstico no depende del estado del sistema. En el EnKF, se tiene en cuenta la dependencia del estado, pero los errores de muestreo debidos a un tamaño de ensamble reducido pueden afectar su precisión (Hamill et al., 2001). En el caso de UnnKF, la calidad de la covarianza vendrá determinada por la capacidad de la RNA para aprender la relación entre el estado del sistema y el error asociado.

En las metodologías por ensambles que estiman la MCEP empíricamente (por ejemplo EnKF), suele utilizarse algún método de localización de la MCEP para paliar el error de muestreo, sobre todo cuando se utilizan ensambles de pocos miembros. La idea es filtrar covarianzas espurias entre variables de estado que están distantes entre sí espacialmente. Esto se logra aplicando una matriz de localización $\mathbf{C}$ a la MCEP para forzar el decaimiento de las covarianzas con el aumento de la distancia en el espacio físico. En general, la estructura de la matriz de localización es una decisión de diseño basada en el conocimiento de la dinámica del problema o en resultados empíricos. El concepto de localización también puede aplicarse a las MCEP estimadas usando RNA directamente sobre la estimación o, durante el entrenamiento, modificando la función de costo. Por ejemplo, podemos implementar la localización de la covarianza, modificando la función de costo definida en el capítulo anterior (Ec. 5.6) de la siguiente manera:

$$T_{cov} = \|\tilde{\Sigma}_k - \mathbf{C} \odot [\epsilon_k^f (\epsilon_k^f)^\top]\|_F \quad (6.2)$$

donde la matriz $\mathbf{C}$ se supone conocida a priori a partir del conocimiento de las interacciones dinámicas y $\odot$ es el producto elemento a elemento. Una consecuencia directa de aplicar la matriz de localización es que todos los elementos de la matriz de salida ($\tilde{\Sigma}_k$) co-

respondientes a elementos con valor igual a 0 en C pueden no estimarse, lo cual reduce el tamaño de la capa de salida de la red. De este modo, reducimos también el número de parámetros de entrenamiento y limitamos el cálculo de las covarianzas a sólo las sub-diagonales seleccionadas, es decir, $\tilde{\Sigma}_k$ puede representarse mediante una matriz banda (véase la Sec. 6.2.2).

6.2. Diseño experimental

En esta sección se presentan los experimentos orientados a evaluar la precisión y eficiencia de la estimación de la covarianza en el contexto de un sistema de asimilación de datos secuencial. Aunque la corrección del error sistemático del pronóstico podría mejorar el rendimiento de la asimilación, no la incluimos como parte de los experimentos ya que podría ocultar la sensibilidad al error del análisis en la estimación de la covarianza. De esta manera, la variabilidad en la calidad del análisis puede atribuirse únicamente a la calidad de la estimación de la MCEP por la RNA. El método de asimilación utilizado es el UnnKF presentado en la sección anterior, pero en este caso, sólo se estima la MCEP basada en un pronóstico determinístico $\mathbf{x}_k^f$ como se muestra en la Fig. 6.2.

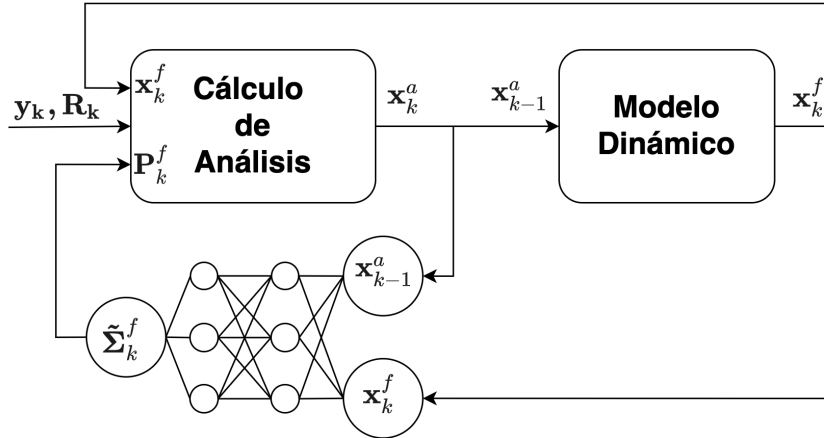


Figura 6.2: Representación esquemática de un ciclo de asimilación UnnKF donde se utiliza una RNA sólo para estimar la MCEP.

6.2.1. Modelo dinámico y conjunto de entrenamiento

La metodología UnnKF se evalúa en un escenario de modelo imperfecto (EMI) similar al que se usó en el capítulo 4 pero con otra configuración de parámetros del modelo y usando un sistema de observación parcial del estado. Es decir, se usó el modelo de Lorenz'96 de dos escalas con $N_x = 100$ variables lentas y $J = 32$ variables rápidas por cada variable lenta, los parámetros de acoplamiento entre escalas son $h = 1$, $c = 10$ y $b = 10$ y se usó un forzante $F = 26$ para obtener un comportamiento caótico.

Las observaciones se generan a partir de la integración de la *verdad* cada 8 pasos temporales añadiendo un error gaussiano de media cero y varianza igual a 0,2 y están disponibles sólo para los puntos impares de retícula, es decir, sólo se observa el 50 % de las variables de estado del sistema.

Dado el conjunto de observaciones y el modelo de pronóstico, se utilizó la metodología EnKF (Sec. 2.2.2) para generar un conjunto de análisis $\mathbf{x}_k^a$ que asumiremos como la mejor aproximación al estado real del sistema. Se eligió el EnKF estocástico, ya que el uso de filtros determinísticos como el LETKF (Hunt et al., 2007) afectó negativamente al entrenamiento de los modelos de RNA utilizados, debido a que en estos filtros los miembros del ensamble no son equiprobables y algunos miembros se asocian persistentemente con mayores desviaciones respecto de la media del ensamble. Este efecto ya ha sido reportado por Amezcua et al. (2012) y encontrado en un experimento con datos reales por Kondo and Miyoshi (2019).

Para entrenar las RNAs y como línea de base para evaluar la metodología UnnKF se generaron dos conjuntos de análisis: uno basado en un ensamble de 100 miembros y el otro en un ensamble de 5 miembros. Para reducir el impacto de los errores de muestreo, en ambos casos se utilizó la función de localización sugerida en Gaspari and Cohn (1999) con una escala de localización de 7 puntos de retícula que fue el valor que minimizó la raíz cuadrada del error cuadrático medio, en inglés Root Mean Squared Error (RMSE), de los análisis. De la misma manera, el factor de inflación se ajustó al valor que minimiza el RMSE, dando un factor de inflación óptimo de 1,15 para el conjunto de 100 miembros y de 1,35 para el experimento del conjunto de 5 miembros.

Para armar el conjunto de entrenamiento se utilizaron los primeros 10.000 análisis y para el conjunto de validación los 5.000 análisis siguientes. El tamaño del conjunto de entrenamiento en unidades de tiempo del modelo Lorenz equivale a 10 años de datos en tiempos atmosféricos. El conjunto de pruebas consta de 15.000 análisis completamente independientes del conjuntos de entrenamiento y validación.

capa	canales entrada	canales salida	kernel	activación
entrada	2	32	3	Softplus
oculta	32	32	3	Softplus
salida	32	n_d	3	Softplus(0)+linear(1 : d)

Tabla 6.1: Descripción de la arquitectura de la red neuronal convolucional utilizada en la experimentación. El número de canales de la capa de salida (n_d) es el número de sub-diagonales de la MCEP que serán estimados.

6.2.2. Arquitectura de las RNA

En la experimentación de los capítulos previos se utilizaron RNAs completamente conectadas, pero al aumentar considerablemente el espacio de estados ($N_x = 8$ en el capítulo 4 y $N_x = 100$ en este capítulo) comprobamos que esta arquitectura no escalaba adecuadamente respecto de la cantidad de parámetros entrenables. Por ese motivo, en este capítulo, se utilizan redes convolucionales porque nos permiten reducir la cantidad de conexiones (parámetros entrenables) y porque asumen invariancia de traslación, lo que concuerda con la isotropía estadística de la dinámica de Lorenz'96. Las características de la red (cantidad de capas, cantidad de filtros convolucionales en cada capa, etc.) se optimizaron mediante experimentos de sensibilidad minimizando el valor de la función de costo para el conjunto de validación. La tabla 6.1 describe la arquitectura de red utilizada para la mayoría de los experimentos en este capítulo. La extensión espacial de los filtros que dieron mejores resultados es de 3 variables. Este resultado es coherente con el comportamiento de las variables de Lorenz que presentan interacciones localizadas, es decir, dos variables alejadas entre sí tienen covarianzas cercanas a cero. Dado que la condición de contorno del modelo de Lorenz es circular, se aplicó el mismo criterio a las capas convolucionales (padding circular) que además nos permite mantener sin cambios las dimensiones de la representación espacial a través de la red. Para las dos primeras capas convolucionales, se utilizó la función de activación Softplus porque obtuvo, durante los experimentos de optimización preliminares, una convergencia ligeramente superior a otras funciones de activación consideradas (logística y ReLU).

En la capa de salida se utilizaron dos funciones de activación distintas. La función Softplus se utilizó para los elementos de salida correspondientes a la diagonal principal (n_0) para garantizar que las varianzas estimadas sean positivas. Para los elementos correspondientes a las covarianzas (elementos subdiagonales de la MCEP $n_1 \dots n_d$) se usó la función de activación lineal. En experimentos preliminares, se observó que el uso de activaciones lineales o ReLU en la diagonal principal (n_0) puede conducir a la estimación

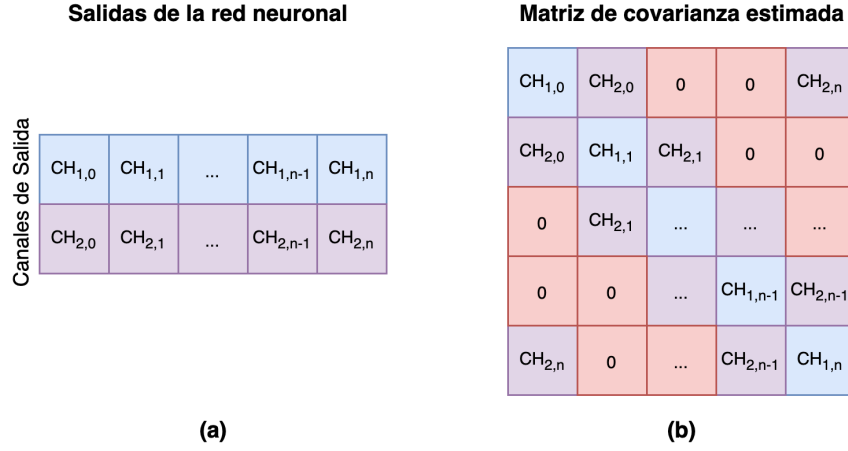


Figura 6.3: La salida de la red se muestra en el panel (a), el elemento i -ésimo del canal 0 es la varianza correspondiente a la variable de estado i -ésimo. El canal d -ésimo corresponde a la covarianza entre las variables de estado separadas por una distancia d como se muestra en el panel (b). Se supone que los valores de covarianza no representados con la salida de la RNA son cero.

de varianzas negativas o iguales a cero, respectivamente. El tamaño de la capa de salida depende directamente de la elección de la matriz de localización C de la Ec. 6.2. En esta instancia, se decidió utilizar una función del tipo Heaviside para localizar los elementos de la covarianza objetivo a una distancia d determinada. Es decir, si la distancia entre dos variables de Lorenz es menor que d puntos de retícula, dejamos la covarianza correspondiente sin cambios, y si es mayor que d , ponemos la covarianza correspondiente a 0. Esto equivale a mantener sólo las primeras n_d subdiagonales de la covarianza objetivo. Entonces diseñamos la RNA para estimar sólo las primeras n_d subdiagonales de la MCEP, ya que todas las demás subdiagonales son 0 (Figura 6.3). En la sección 6.3.2 se analizan resultados de sensibilidad al parámetro d .

6.2.3. Entrenamiento de las RNAs

La elección de una arquitectura de red y los hiperparámetros de entrenamiento óptimos resultó una tarea ardua. Si bien la convergencia durante el entrenamiento a valores bajos de la función de costo fue relativamente fácil de lograr, la principal dificultad fue encontrar un conjunto de hiperparámetros tal que la MCEP estimada por la red nos permita generar largas secuencias de análisis usando el método UnnKF.

En capítulos previos se discutió y se analizó la gran sensibilidad al sobreajuste que pre-

sentan las redes entrenadas para estimar incertidumbre en comparación con la estimación del error medio. Para controlar el sobreajuste se utilizó, además de la técnica de parada temprana (detener el entrenamiento cuando el valor de la función de costo sobre el conjunto de validación deje de disminuir), la regularización L_2 o decaimiento de pesos, en inglés Weight Decay (WD). En particular se observó que, para valores mayores a 1 del parámetro λ que ajusta el decaimiento de los pesos, la variabilidad de la MCEP estimada se suaviza y disminuye hasta volverse prácticamente independiente del estado asemejándose a una MCEP climatológica, pero que permitía generar largas secuencias de análisis con UnnKF.

Esto nos permitió, de forma iterativa, ir alternando pasos de ajuste de arquitectura y disminución del valor de λ , hasta lograr optimizar los hiperparámetros de entrenamiento y la capacidad de la RNA para valores de λ iguales a cero. Finalmente, para entrenar las RNA de todos los experimentos se utilizó el optimizador AdamW (Loshchilov and Hutter, 2017) con un valor de tasa de aprendizaje, en inglés Learning Rate (η_{lr}), de 0,001 y un tamaño de lote de 50 muestras.

6.2.4. Estimación del error de pronóstico

En la sección 5.3 se evaluó una metodología para entrenar RNAs que estiman la MCEP usando el error del pronóstico como objetivo de entrenamiento. Pero conocer el error real del pronóstico implica conocer el estado *verdadero* del sistema, y esto no es posible para sistemas como la atmósfera y el océano de los cuales sólo se dispone de aproximaciones como los pronósticos, las observaciones o los estados analizados. La figura 6.4 esquematiza este problema y muestra algunas aproximaciones del error que pueden considerarse en función de los datos disponibles en sistemas reales. A continuación, se listan posibles aproximaciones del error de pronóstico a corto plazo (ϵ_k^f) que serán evaluadas en términos de cómo expresan la variabilidad espacio-temporal del mismo:

- **Inicialización Media - Media del Análisis (MMA):** En este caso definimos $\epsilon_k^f = \mathbf{x}_k^f - \bar{\mathbf{x}}_k^a$ donde $\mathbf{x}_k^f$ es un pronóstico determinístico inicializado con el análisis medio del ensamble ($\mathbf{x}_k^f = \mathcal{M}(\bar{\mathbf{x}}_{k-1}^a)$). En este caso, estamos tomando la diferencia entre el estado más probable del sistema dadas todas las observaciones hasta el tiempo $k - 1$ y el estado más probable del sistema dadas todas las observaciones hasta el tiempo k .
- **Inicialización Media - Random Análisis (MRA):** Definimos el error como $\epsilon_k^f = \mathbf{x}_k^f - \mathbf{x}_k^{a, (n_r)}$ donde $\mathbf{x}_k^{a, (n_r)}$ es un miembro aleatorio del ensamble de análisis. Esta

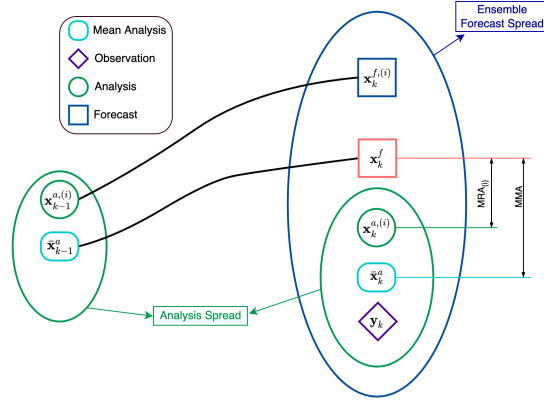


Figura 6.4: Esquema de un ciclo de asimilación por ensemble en el que se muestran las aproximaciones del error del pronóstico MRA y MMA utilizando sólo el miembro i -ésimo del ensemble.

aproximación supone que los miembros del ensemble de análisis representan realizaciones igualmente probables del estado verdadero. Es importante señalar que para la construcción del conjunto de entrenamiento, la selección aleatoria se realiza una sola vez, es decir, el conjunto de datos es el mismo en todas las épocas de entrenamiento. Con esta formulación sería posible aumentar el conjunto de entrenamiento utilizando todos los miembros disponibles del ensemble. De hecho, ampliar el conjunto de entrenamiento de esta forma mejora la calidad de la MCEP, y en consecuencia, disminuye el RMSE de los análisis. Sin embargo, a efectos de la comparación, se optó por utilizar sólo un miembro seleccionado aleatoriamente de modo que el tamaño del conjunto de entrenamiento sea el mismo que en el resto de los métodos de aproximación elegidos.

- **Inicialización Media - Estado Verdadero (MNT):** Dado que estamos realizando experimentos idealizados, tenemos acceso al estado verdadero del sistema y podemos calcular el error real del pronóstico como $\epsilon_k^f = \mathbf{x}_k^f - \mathbf{x}_k^t$, donde $\mathbf{x}_k^t$ es el estado verdadero del sistema. Como esta representación del error no puede obtenerse en aplicaciones reales, sólo se utiliza como referencia y a efectos comparativos.

También se evaluaron otras aproximaciones del error, por ejemplo, $\epsilon_k^f = \mathbf{x}_k^{f,(n_r)} - \bar{\mathbf{x}}_k^a$ donde $\mathbf{x}_k^{f,(n_r)} = \mathcal{M}(\mathbf{x}_{k-1}^{a,(n_r)})$ es un miembro aleatorio del ensemble de pronósticos o $\epsilon_k^f = \mathbf{x}_k^{f,(n_r)} - \mathbf{x}_k^{a,(n_r)}$, que es la diferencia entre un miembro aleatorio del ensemble de pronósticos y el miembro correspondiente en el ensemble de análisis, pero ninguno de estos mejoró los resultados de **MRA**.

Es importante notar que a cada aproximación del error le corresponde una varianza distinta. Por lo tanto, las trazas de las MCEP estimadas por una RNA entrenada usando diferentes aproximaciones del error pueden ser significativamente distintas. Para reducir el impacto de este efecto en el ciclo de asimilación y poder comparar los distintos enfoques de una forma más consistente, se aplica un factor de inflación multiplicativo en los experimentos de DA de la misma manera que se hace en EnKF. El factor de inflación multiplicativo se optimiza independientemente para cada aproximación del error buscando exhaustivamente el valor que minimiza el RMSE de los análisis.

6.3. Resultados

El objetivo principal de la experimentación realizada fue, por un lado, evaluar la calidad de la MCEP y el rendimiento de la metodología UnnKF usando una RNA entrenada con la función de costo basada en el error, en inglés extended Mean Squared Error (eMSE) (Sec. 5.3). Por otro lado, estudiar la solidez de la solución mediante experimentos de sensibilidad y usando la metodología EnKF como base de comparación.

Identificamos a cada experimento UnnKF con un nombre compuesto por dos partes, la primera hace referencia a la aproximación del error (Sec. 6.2.4) utilizada durante el entrenamiento de la RNA y la segunda es el número de diagonales de la MCEP que estima la RNA (Sec. 6.2.2). De manera similar, los experimentos EnKF usados como referencia se denominan *ENS* seguido del número de miembros usados en el ensamble, por ejemplo *ENS5* refiere al experimento de DA con el método EnKF usando un ensamble de 5 miembros.

Se comenzó comparando la evolución temporal de las MCEP estimadas con una RNA entrenada con la aproximación MRA y $n_d = 6$ (MRA6), y con ensambles de modelos de 5 (ENS5) y 100 miembros (ENS100) en un entorno EMI (ver Sec. 6.2.1). Los resultados para algunos de los elementos de la MCEP se muestran en la figura 6.5. Se distingue entre variables de estado observadas y no observadas durante el proceso de asimilación, ya que la variabilidad del error puede ser notoriamente diferente y, en consecuencia, también la variabilidad de los elementos de la MCEP. Para tener una idea general de similitud entre las MCEP estimadas, cada panel indica el coeficiente de correlación temporal de MRA6 y ENS5 con respecto a ENS100 para todo el conjunto de pruebas. Para todos los casos, aun los que no se muestran en la figura 6.5, el coeficiente de correlación de la estimación MRA6 es superior a la correlación de ENS5. Estos valores coinciden con las evoluciones temporales donde se puede ver que ENS5 produce covarianzas con una mayor variabi-

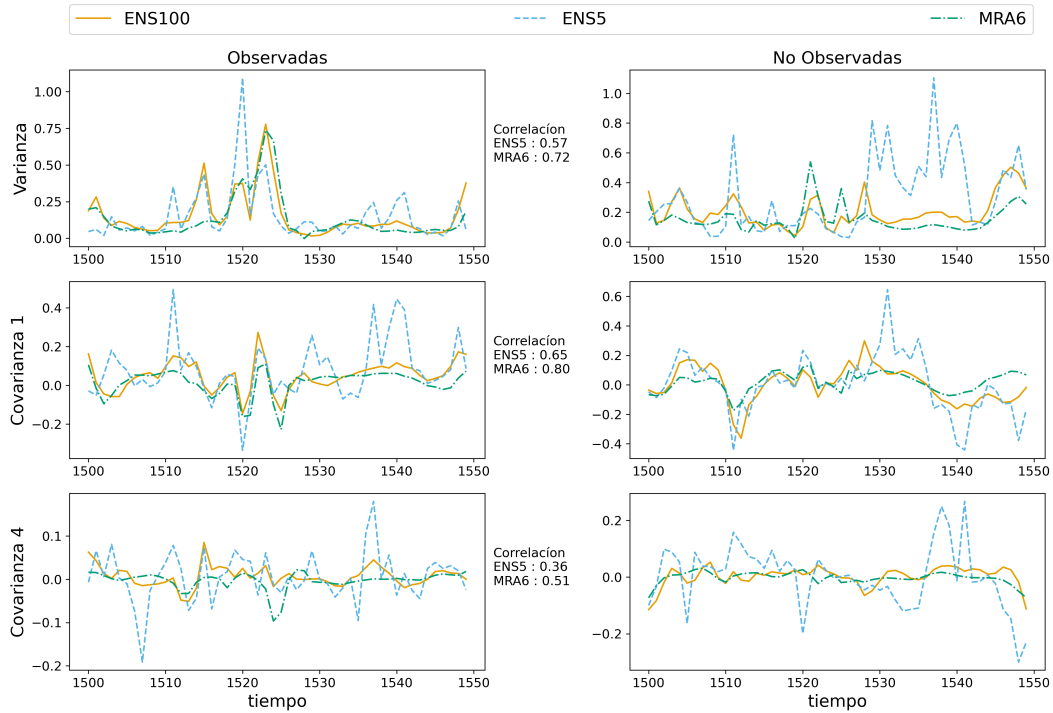


Figura 6.5: Evolución temporal de los elementos seleccionados de la MCEP a lo largo de 100 pasos temporales consecutivos a partir del ciclo 1500 del conjunto de pruebas. El panel izquierdo de la primera fila muestra la varianza de la primera variable observada ($MCEP_{(0,0)}$) y a la derecha la primera variable no observada ($MCEP_{(1,1)}$). La segunda y tercera fila muestran la covarianza de la variable de interés con su primera y cuarta vecina respectivamente ($MCEP_{(i,i+1)}$ y $MCEP_{(i,i+4)}$). A la derecha de cada panel se indica el coeficiente de correlación calculado sobre 15.000 ciclos de asimilación, entre ENS5 y MRA6 con respecto a ENS100.

alidad temporal en comparación con ENS100 debido al efecto del error de muestreo. En cambio, MRA6 sigue de cerca la variabilidad de ENS100 tanto para las variables observadas como para las no observadas, aunque MRA6 tiene una variabilidad más suave que ENS100 y a veces parece omitir algunos extremos (por ejemplo, el tiempo 1515 para las variables observadas en todas las covarianzas). Pero también es capaz de reproducir con bastante precisión otros extremos presentes en ENS100 (por ejemplo, la varianza y la covarianza en el momento 1525 para las variables observadas).

También se puede evaluar la calidad global de la estructura espacio-temporal de las MCEP comparando el RMSE de los análisis generados en un contexto de DA. La Fi-

gura 6.6 compara el RMSE de secuencias de análisis generados con UnnKF (MRA6) y con EnKF (ENS5 y ENS100). La línea negra en la parte superior de cada barra representa el intervalo de confianza del 95 % calculado mediante un enfoque bootstrap utilizando 500 submuestras obtenidas del conjunto de datos de prueba mediante selección aleatoria con reemplazo y seleccionando muestras que están separadas más de 20 pasos temporales entre sí para aumentar la independencia entre los diferentes elementos de la muestra. Los resultados de la figura 6.6 indican que, tanto para las variables observadas como pa-

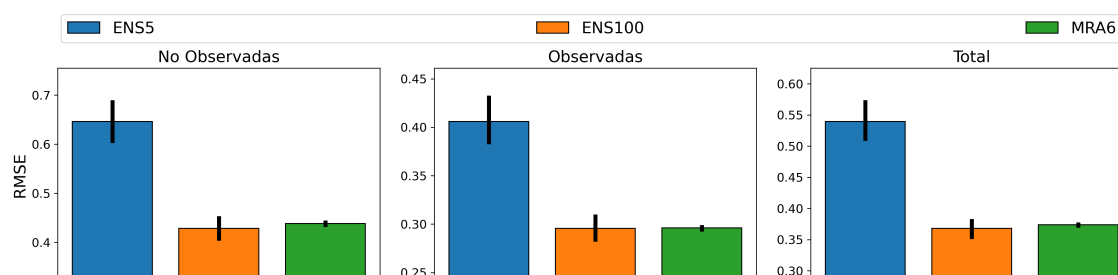


Figura 6.6: RMSE de los análisis generados por un ensemble de 5 miembros (ENS5), un ensemble de 100 miembros (ENS100) utilizando el método EnKF y el método UnnKF (MRA6) para el modelo de Lorenz'96 de 100 variables durante 15.000 ciclos consecutivos de asimilación basados en datos del conjunto de prueba. Se muestran los RMSE de las variables no observadas (izquierda), de las variables observadas (centro) y el RMSE total (derecha). Obsérvese que los rangos de los ejes RMSE son diferentes en cada panel, esto es para resaltar la diferencia entre los experimentos. La línea negra de la parte superior de cada barra indica los límites de un intervalo de confianza del 95 % obtenido mediante un método bootstrap.

ra las no observadas, el RMSE del experimento MRA6 está mucho más cerca de ENS100 que de ENS5. Esto concuerda con el análisis de la evolución temporal de los elementos de la MCEP y muestra que la metodología propuesta es capaz de generar una estimación de la MCEP dependiente del estado lo suficientemente robusta como para generar largas secuencias de ciclos de asimilación utilizando únicamente un pronóstico determinista como entrada.

6.3.1. Sensibilidad a la aproximación del error del pronóstico

En la sección 6.2.4 se discutió el problema de aproximar el error del pronóstico y se identificaron posibles candidatos para entrenar la RNA. La Figura 6.7 muestra el RMSE

de secuencias de análisis generados con UnnKF usando la RNA entrenada con el error real del pronóstico (MNT) y aquellas entrenadas con las aproximaciones MRA y MMA. En todos los casos se estiman $n_d = 6$ diagonales de la MCEP.

En línea general, todas las aproximaciones resultaron estables y precisas en el sentido de lograr producir largos ciclos de asimilación con bajos valores de RMSE, mostrando una variabilidad del 10 % entre la mejor aproximación (MRA) y la peor (MMA).

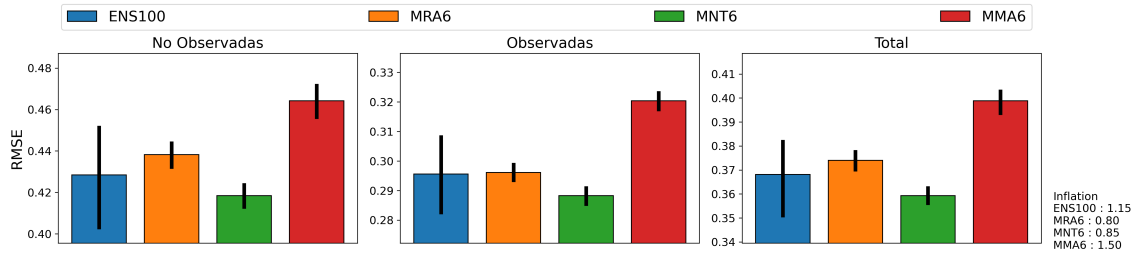


Figura 6.7: Se muestra el RMSE de las variables no observadas del sistema (panel izquierdo), las variables observadas (panel central) y el RMSE total (panel derecho) de los análisis generados utilizando un conjunto de datos de prueba independiente para las redes entrenadas con los conjuntos de datos MNT, MRA y MMA. La línea negra de la parte superior de cada barra indica los límites de un intervalo de confianza del 95 % obtenido mediante un enfoque bootstrap.

El uso del análisis medio como objetivo de entrenamiento da como resultado un RMSE significativamente mayor en comparación con el uso de la aproximación MRA. Además, el UnnKF utilizando MMA requiere un factor de inflación multiplicativo mayor para lograr un rendimiento óptimo, lo que indica que esta aproximación subestima la amplitud de los errores. Esto podría deberse al hecho de que un pronóstico determinístico inicializado con un análisis medio para un plazo de pronóstico muy corto es similar a la media del ensamble de pronósticos, y como estamos aproximando el error como la diferencia entre el pronóstico determinístico y la media del ensamble de análisis, no se está teniendo en cuenta que tanto el pronóstico como el estado verdadero del sistema son realizaciones aleatorias de estas distribuciones. En cambio, en el enfoque MRA se contempla que los miembros del ensamble de análisis son realizaciones igualmente probables del estado del sistema y se elige una de las posibles realizaciones para el cálculo del error. El valor de inflación óptimo de esta aproximación resultó ser menor que uno, lo que implica una sobrestimación de la varianza de los errores.

El experimento MRA obtuvo valores de RMSE comparables con ENS100, especialmente

para las variables observadas, sin embargo, para las variables no observadas aparecen diferencias significativas. Parte de estas diferencias pueden explicarse por el efecto del error de modelo presente tanto en el pronóstico como en el análisis de nuestros experimentos. Aunque la asimilación de las observaciones reduce en parte el impacto de los errores en el análisis, otra parte permanece, especialmente en las variables que no están observadas, lo que degrada la representación del error.

A pesar de las diferencias, todas las aproximaciones resultaron ser buenas aproximaciones si consideramos que el RMSE es 8,35 % mayor que el RMSE de ENS100 para el experimento MMA, mientras que para ENS5 es un 46 % mayor.

El RMSE más bajo lo tiene el experimento MNT6, siendo incluso mejor que ENS100. Este resultado es muy interesante y nos indica que al usar un objetivo adecuado de entrenamiento, es posible entrenar una RNA capaz de proporcionar, a partir de una única integración de un modelo físico, una estimación de la incertidumbre del pronóstico dependiente del estado que es más precisa (en términos de minimización del RMSE del análisis) que la proporcionada por un ensamble grande de pronósticos a un costo computacional mucho menor. El valor de RMSE alcanzado por MNT6 se logra con una inflación multiplicativa de 0,85 y aplicando la función de localización de Gaspari-Cohn ([Gaspari and Cohn, 1999](#)) con distancia de 4 puntos de retícula a la estimación de la RNA. Si no se usara la función de localización, la inflación óptima es de 0,8 y el RMSE aumenta un 1,7 %. Esto podría indicar una ligera sobreestimación de las covarianzas distantes para el experimento MNT. El valor mínimo de RMSE para los experimentos MMA y MRA se obtiene sin aplicar ninguna función de localización adicional a la matriz de covarianza.

6.3.2. Sensibilidad a la localización

Como se analizó en la sección 6.1, para la metodología UnnKF la localización se puede aplicar en dos momentos: durante el entrenamiento de la RNA simplificando el entrenamiento y reduciendo la cantidad de parámetros entrenables de la RNA, o a posteriori aplicando una función de localización a la estimación de la RNA de la misma manera que se localiza la MCEP estimada con ensambles de pronósticos. En esta sección se analiza la sensibilidad a la primera forma de localización, es decir, sólo se restringe la cantidad de diagonales de la MCEP que serán estimadas, pero no se escalan sus elementos. Entonces, se busca estudiar cómo la RNA modela las covarianzas entre variables distantes y hasta qué punto la inclusión de covarianzas entre variables lejanas mejora la precisión del análisis. Como referencia, en la figura 6.9 se muestra cómo la MCEP propaga información de variables observadas a no observadas.

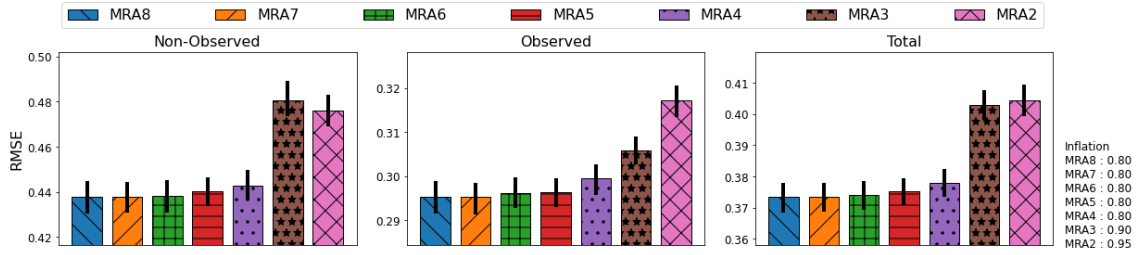


Figura 6.8: Se muestra el RMSE para matrices de covarianza con diferente número de diagonales no nulas: $\{2, 3, 4, 5, 6, 7, 8\}$, para las variables no observadas del sistema (panel izquierdo), las variables observadas (panel central) y el RMSE total (panel derecho) de los análisis generados con UnnKF utilizando un conjunto de datos de prueba independiente. Los valores óptimos de inflación para cada experimento se indican a la derecha de la figura. Los límites de un intervalo de confianza del 95 % obtenido mediante un enfoque bootstrap se indican con una línea negra en la parte superior de cada barra.

Los resultados de la figura 6.8 muestran que el RMSE de los análisis disminuye cuando aumenta n_d a excepción del experimento con $n_d = 3$. Para este caso, en las variables no observadas se obtiene un aumento del error del análisis con respecto al caso $n_d = 2$. Esto puede deberse a que, en el experimento con $n_d = 3$, el número de observaciones utilizadas para corregir las variables no observadas del análisis es el mismo que en el experimento con $n_d = 2$. Sin embargo, el rendimiento en las variables observadas mejora cuando n_d se aumenta de 2 a 3, ya que el número de observaciones asimiladas en las variables observadas aumenta de 1 a 3. Este efecto se esquematiza en la Figura 6.9 mostrando que las subdiagonales pares propagan la información de las observaciones a las variables no observadas, mientras que las subdiagonales impares propagan la información de las observaciones variables observadas. Este efecto parece imperceptible a partir de la 4ª subdiagonal, probablemente debido a la pequeña amplitud de las covarianzas estimadas.

6.3.3. Sensibilidad a la dimensión del vector de estados

En esta sección investigamos cómo la dimensión del estado y el número de diagonales estimadas de la MCEP se relacionan con la capacidad requerida por la RNA, en particular el número de filtros convolucionales en la capa oculta que produce los análisis más precisos.

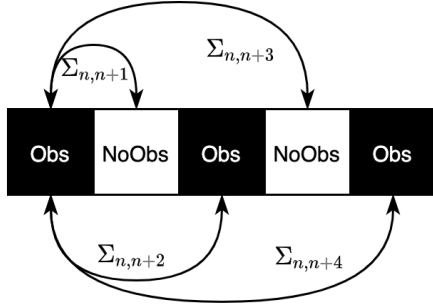


Figura 6.9: Esquema de propagación de la información provista por las observaciones hacia las variables de estado observadas (**Obs**) y no observadas (**NoObs**) durante el proceso de asimilación de datos. Se muestra cómo cada covarianza ($\Sigma_{n,i}$) propaga la información de una variable (n) a otra (i).

Evaluamos el rendimiento del UnnKF para diferentes dimensiones del estado (N_x) realizando tres experimentos con $N_x \in \{8, 40, 100\}$. Para el primer experimento $N_x = 8$, estimamos la MCEP completa ($n_d = 4$). Para $N_x = 40$ y $N_x = 100$ se consideró $n_d = 6$. Estos experimentos se comparan con un ensemble pequeño de 5 miembros y un ensemble grande de 50 miembros para $N_x = 8$ y 100 miembros para $N_x = 40$ y $N_x = 100$. En todos los casos, el ensemble grande presenta un tamaño tal que no se obtienen mejoras significativas al incrementar la cantidad de miembros más allá de este tamaño.

En los resultados que se muestran en la Figura 6.10, el UnnKF tiene un RMSE cercano al del ensemble grande en todos los experimentos. Esto indica que el rendimiento de la estimación de la covarianza tiene poca sensibilidad al tamaño del vector de estado en estos experimentos. Y para todos los casos, el rendimiento del UnnKF es significativamente mejor que el alcanzado con el ensemble pequeño, aun cuando éste está debidamente optimizado en términos de la inflación multiplicativa y de la localización.

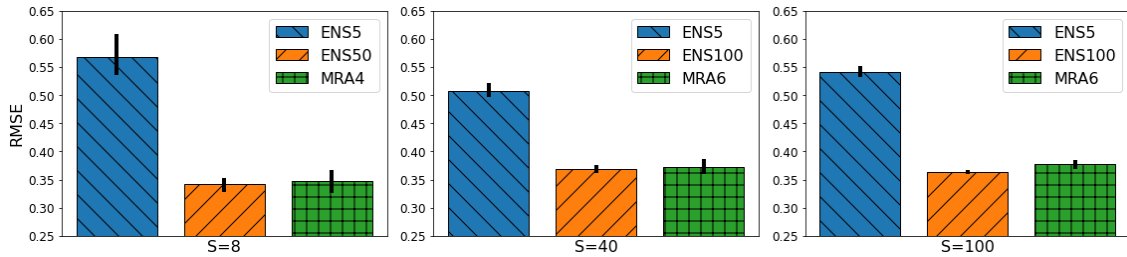


Figura 6.10: RMSE de los análisis generados para el modelo Lorenz'96 de 8, 40 y 100 variables de estado (paneles derecho, central e izquierdo, respectivamente). Cada panel muestra el RMSE alcanzado para la metodología EnKF con un ensemble pequeño de 5 miembros (ENS5), un ensemble grande de 50 miembros para $N_x = 8$ (ENS50) y 100 miembros para $N_x = 40$ y $N_x = 100$ (ENS100) y para el experimento MRA6.

También se realizaron experimentos para evaluar la relación entre la cantidad de pará-

número de canales en la capa oculta	$n_d = 2$		$n_d = 8$	
	func. costo	RMSE	func. costo	RMSE
10	0,11816	0,4085	0,068481	0,39275
20	0,11758	0,4050	0,068203	0,37823
32	0,11748	0,4042	0,068105	0,37336
40	0,11755	0,4053	0,068130	0,37331
50	0,11766	0,4044	0,068115	0,37335

Tabla 6.2: Función de costo y RMSE de los análisis calculado sobre el conjunto de datos de prueba para distintas arquitecturas de RNA. La primera columna muestra el número de núcleos convolucionales utilizados en la capa oculta. Los resultados se presentan para el caso en que se estimaron 2 diagonales ($n_d = 2$) y 8 diagonales ($n_d = 8$) de la MCEP.

metros entrenables y el número de diagonales estimadas variando la cantidad de canales de la capa oculta de la RNA y verificando la convergencia (valor alcanzado por la función de costo) y el RMSE del análisis resultante.

Los resultados de la Tabla 6.2 muestran que la capacidad óptima de la red es prácticamente insensible al número de diagonales estimadas. Para $n_d = 2$ (200 variables de salida) el número óptimo de filtros es 32 y para $n_d = 8$ (800 variables de salida) el óptimo es 40. Se observó que usar redes con más capacidad degrada ligeramente el RMSE. Este resultado puede estar vinculado a la dinámica de Lorenz'96 donde las variables distantes espacialmente prácticamente no interactúan entre sí y, en consecuencia, los valores de covarianzas son muy pequeños.

6.3.4. Sensibilidad a la calidad del objetivo de entrenamiento

Hasta este momento, independientemente de la aproximación del error que se haya utilizado, todas las RNA se entrenaron usando análisis generados con la metodología EnKF y un ensamble de 100 miembros. Sin embargo, en escenarios reales de sistemas meteorológicos, los análisis históricos disponibles suelen estar generados con ensambles mucho más pequeños respecto de la dimensión del estado. Para investigar cómo impacta la calidad de los análisis en el entrenamiento de las RNAs, realizamos un experimento adicional denominando MRA6_E5 que usa como objetivo de entrenamiento los análisis generados con un EnKF de sólo 5 miembros. El resultado se muestra en la figura 6.11 junto con los resultados anteriores (MRA6 y MNT6) usados como base de comparación.

Un resultado interesante es que el experimento MRA6_5 genera análisis con un RMSE mucho menor que los análisis de ENS5 utilizados como objetivo de entrenamiento.

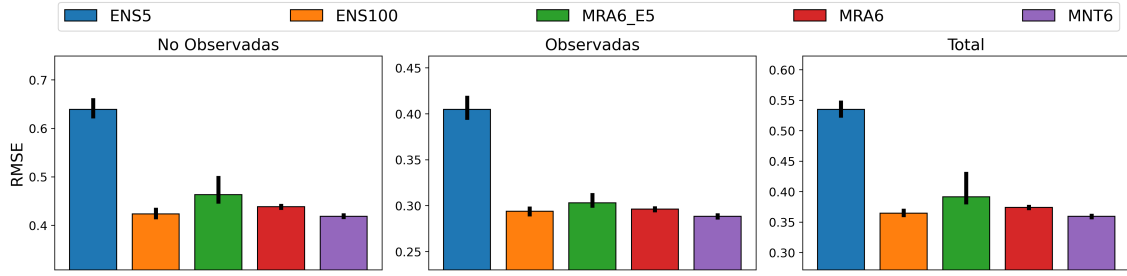


Figura 6.11: RMSE de diferentes experimentos de asimilación donde ENS100 y ENS5 corresponden al rendimiento del EnKF para un ensamble de 100 y 5 respectivamente. MNT6 corresponde al rendimiento de una red neuronal entrenada con el estado verdadero, MRA6_E5 es una red entrenada con el conjunto de datos de 5 miembros y MRA6 es la red entrenada con un conjunto de datos de 100 miembros.

Este resultado puede explicarse por el hecho de que el ensamble pequeño tiene pocos miembros para estimar todos los elementos de la covarianza, por lo que se espera que los errores de muestreo sean grandes. Sin embargo, durante el entrenamiento y debido al uso de múltiples instancias en diferentes momentos (ver Fig. 3.5) la RNA logra suavizar las muestras, lo que conduce a una estimación más fiable de la MCEP.

En cambio, los análisis generados con ensambles de muchos miembros tienen un error de muestreo muy pequeño, y es probable que el error del modelo sea el error dominante. Entonces los análisis tienen un sesgo, por lo que una parte del error del pronóstico no está bien representada durante el entrenamiento y, en consecuencia, el RMSE resultante es bastante similar, con una ligera ventaja de ENS100 frente a MRA6. Esto no ocurre con el experimento MNT6 que supera a ENS100.

Hasta el momento se evaluó la metodología UnnKF sin considerar la corrección de la media de la distribución (ver Fig. 6.2) y utilizando el pronóstico determinístico. Con el fin de explorar toda la capacidad del método UnnKF, se realizó un último experimento donde se estima media y covarianza como se indica en la figura 6.1. En este caso, el error de entrenamiento no se calcula en base al pronóstico determinístico, sino sobre la estimación de la salida de medias ($\tilde{\mu}_k^f$). Para comparar las metodologías EnKF y UnnKF se intentó equiparar el desempeño de las RNAs con la cantidad de miembros necesarios para obtener el mismo RMSE con la técnica EnKF.

La figura 6.12 muestra el RMSE de los análisis generados por cada experimento sobre la curva que relaciona el RMSE del método EnKF según la cantidad de miembros utilizados en el ensamble. La discrepancia observada para el experimento ENS100 de 100 miembros

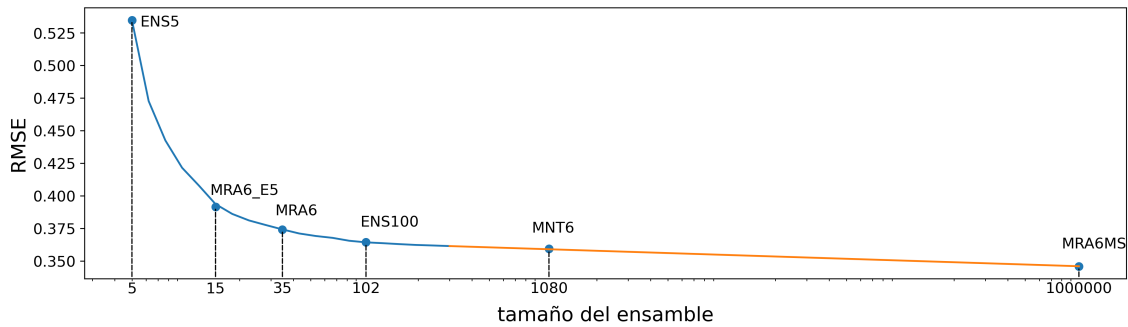


Figura 6.12: La curva azul muestra el RMSE de los análisis generados con EnKF en función del tamaño del ensemble, la recta naranja es una extrapolación lineal. El RMSE alcanzado por cada experimento se representa sobre la curva para relacionar el rendimiento de UnnKF con el tamaño de ensemble equivalente para EnKF.

se debe a que, para generar la curva, se evaluó el RMSE de 5.000 análisis del conjunto de pruebas mientras que el RMSE de los experimentos está calculado con 15.000 muestras. Si bien esta equivalencia no es rigurosa, sirve para poner en perspectiva el ahorro en términos de cálculos y recursos de cómputo necesarios que supone el uso de UnnKF respecto de EnKF.

Dos resultados son notorios: uno es el experimento MRA6_E5 que logra mejorar la precisión de los análisis de forma equivalente a un ensemble 3 veces más grande que el utilizado para entrenar el experimento. Y el otro es el experimento MRA6MS, que integra los resultados del capítulo 3 con los de este capítulo, alcanzando un desempeño comparable a la técnica EnKF con un ensemble de más de 1.000.000 miembros (por supuesto considerando el hecho de que la zona naranja de la curva de error es una extrapolación lineal).

6.4. Conclusiones

En este capítulo se utilizó una metodología de estimación de la covarianza de los errores del pronóstico de un sistema dinámico caótico y su dependencia con el estado para diseñar un método híbrido de asimilación de datos que denominamos UnnKF. El método se evaluó en términos de estabilidad numérica y escalabilidad con la dimensión del estado. Los resultados mostraron que la técnica es estable, permitiendo realizar largas secuencias de análisis de más de 15.000 ciclos de asimilación, y con un desempeño comparable a utilizar ensambles de gran tamaño con el método EnKF.

También se mostró que el método UnnKF logra disminuir considerablemente el RMSE de los análisis generados con EnKF y ensambles pequeños utilizados para entrenar la RNA con la función de costo eMSE. La capacidad de UnnKF para superar el rendimiento del conjunto de entrenamiento sugiere la posibilidad de entrenar de forma sucesiva con análisis de producción propia generados en un entrenamiento anterior. Este proceso podría repetirse eventualmente hasta que se alcance la convergencia y el RMSE se estabilice en el valor más bajo posible.

Además, se observó que la capacidad de la red en términos de parámetros entrenables no mostró una gran sensibilidad a la dimensión del espacio de estados ni a la cantidad de elementos estimados de la MCEP. Este resultado es especialmente alentador para explorar el uso de la metodología propuesta en aplicaciones más realistas con espacios de estado de muy alta dimensionalidad como la atmósfera y los océanos, aunque ciertamente se requiere más investigación para confirmarlo, así como también el abordaje de otras problemáticas como la imposibilidad de representar explícitamente la matriz de covarianza de los errores del pronóstico en sistemas de muy alta dimensión.

Capítulo 7

Conclusiones

A lo largo de esta tesis se ha estudiado en profundidad la problemática de la estimación de la incertidumbre en estados pronosticados de sistemas dinámicos caóticos, en especial mediante el uso de redes neuronales artificiales (RNAs).

Se analizaron y discutieron metodologías del estado del arte para estimar la media y varianza de la distribución de probabilidad de un pronóstico y su dependencia con el estado del sistema. En base a este análisis, se propuso un método de entrenamiento de RNAs basado en una nueva función de costo que denominamos eMSE. Nuestra metodología y otras estrategias fueron evaluadas para la estimación conjunta del error sistemático del pronóstico dependiente del estado y su incertidumbre asociada. Evaluamos el rendimiento de las RNAs en experimentos realizados con un modelo dinámico caótico simple, y las comparamos con un pronóstico determinístico y un ensamble de pronósticos en un escenario de modelo perfecto (EMP) y un escenario de modelo imperfecto (EMI) para diferentes plazos de pronóstico. En EMP los resultados mostraron que las redes lograban filtrar parte del error asociado a la incertidumbre de las condiciones iniciales, pero no tan bien como un ensamble grande de pronósticos, sobre todo para plazos cortos de pronóstico. En EMI, las RNAs logran corregir parte del error de modelo mejorando el RMSE y también una estimación más precisa y mejor calibrada de la incertidumbre del pronóstico comparado con el ensamble en todos los plazos de pronóstico evaluados. Tras los buenos resultados obtenidos en la estimación de pronósticos probabilísticos usando RNAs, se estudió la posibilidad de extender la metodología a la estimación de distribuciones multivariadas, en particular a la estimación de la matriz de covarianza del error de pronóstico (MCEP). Se propuso una extensión de eMSE para la inferencia de MCEP y se comparó su desempeño con la técnica de entrenamiento basada en la verosimilitud

gaussiana LKHD, resultando ambos estimadores igualmente buenos y comparables con el método de ensambles. Sin embargo, nuestra propuesta tiene las siguientes ventajas por sobre LKHD, que la vuelve más atractiva para casos de muy alta dimensionalidad:

- eMSE es más simple en su formulación y, por ende, más rápida y económica en términos de cómputo, puesto que no hay que reconstruir la MCEP durante el entrenamiento de las RNAs.
- eMSE es más estable numéricamente al momento de pronosticar la MCEP, puesto que no hay que invertir la matriz de precisión.
- eMSE estima directamente la MCEP, y esto permite efectuar la localización de la covarianza directamente a la salida de la red neuronal artificial (RNA) y reducir el número de parámetros entrenables y, en consecuencia, acelerar y simplificar el entrenamiento.

Por otro lado, existen puntos importantes a tener en cuenta en el uso de eMSE:

- Se necesita definir y ajustar correctamente los parámetros α si se desea estimar más de un momento a la vez.
- eMSE no penaliza las varianzas cercanas, iguales o incluso menores que cero, y esto se debe garantizar desde la arquitectura de la red.
- La formulación original de eMSE no garantiza obtener matrices definidas positivas, esto se puede subsanar estimando la descomposición de Cholesky de la MCEP a costa de aumentar la complejidad computacional durante el entrenamiento.

Una de las aplicaciones donde se requirió una estimación de la estructura de la MCEP es en la asimilación de datos, en inglés Data Assimilation (DA). Para esto se diseñó un método de DA que combina la técnica del filtro de Kalman con la estimación de MCEP usando RNAs que denominamos UnnKF. Esta metodología nos permitió generar largas secuencias de ciclos de DA (más de 15.000 ciclos consecutivos) usando observaciones parciales del estado del sistema. Y, además, evaluar de forma integral los métodos propuestos en esta tesis en una aplicación concreta, logrando resultados prometedores incluso superando en experimentos los resultados de técnicas robustas como los ensambles de pronósticos.

Reflexiones y trabajos futuros

Los ensambles son una herramienta confiable y muy poderosa al momento de generar pronósticos probabilísticos. En nuestra experimentación, las redes no fueron capaces de mejorar su desempeño en un EMP que utiliza un gran número de miembros. Aunque sí mostraron mejores resultados en comparación con ensambles pequeños en donde el efecto de los errores de muestreo es significativo. Esto indica que las redes, con un conjunto de entrenamiento suficientemente grande, podrían ser menos sensibles al error de muestreo, lo cual motivaría su aplicación en problemas de detección y reducción de los efectos de dichos errores (e.g. localización de la matriz de covarianza). En un escenario de modelo imperfecto, donde el modelo no logra representar correctamente parte de la dinámica del sistema, un ensamble monomodelo hereda esta deficiencia sin lograr compensar el sesgo introducido y sin poder representar el aporte de los errores de modelo a la componente estocástica del error y cómo depende dicho aporte con el estado del sistema. Para paliar esta situación se debe implementar ensambles que incorporen un tratamiento explícito de las imperfecciones del modelo (p.e. esquemas de corrección del sesgo, ensambles multimodelo, esquemas de inflación, entre otros). Por supuesto, diseñar esta clase de ensambles no es fácil y difícilmente se logre capturar toda la complejidad de los errores de modelo. Todos los experimentos de ensambles utilizados como línea de base de comparación en esta tesis fueron ensambles monomodelos, en trabajos futuros se podría evaluar el uso de ensambles multimodelos y/o técnicas de corrección de sesgo que podrían mejorar su desempeño quizás acercándose o aun mejorando el de las RNAs. Pero aun así, esto no invalida la capacidad de las RNAs de corregir el sesgo y, en todo caso, siguen siendo más simples de diseñar (en comparación con un ensamble multimodelo bien ajustado) y más rápidas y económicas en términos de cómputo.

Creemos que el camino de investigación que se inició con esta tesis no está cerrado, por el contrario, varias preguntas quedaron planteadas que necesitan un estudio más profundo para poder responderlas. Pero entendemos que una continuación directa es implementar las metodologías propuestas en problemas y datos reales, especialmente en el ámbito de las geociencias y en particular en la meteorología, que es la ciencia que motivó este trabajo doctoral. Algunos de los desafíos clave que siguen de este trabajo se describen a continuación.

Alta dimensionalidad

Extender los resultados de esta tesis a estados de muy alta dimensión requiere aún de mucho trabajo y existen varias líneas que pueden adoptarse. Es necesario seguir investigando cómo calcular de forma más eficiente la actualización del análisis. En este trabajo, realizamos una estimación explícita de la actualización del análisis basada en la ecuación de Kalman. Sin embargo, este enfoque no es factible en dimensiones muy altas. Si bien nuestra metodología permite estimar MCEP localizadas logrando reducir cuantiosamente su dimensión, se puede ir aún más allá. Por ejemplo, adaptar UnnKF para trabajar en dominios locales como se implementa en OI (Carrassi et al., 2018) o mediante la implementación de parches como en (Ouala et al., 2018). Otra posibilidad es desarrollar métodos que permitan generar muestras aleatorias consistentes con las covarianzas estimadas como en Grooms (2021).

Estimación del error de pronóstico

Un aspecto fundamental del entrenamiento de RNAs para la estimación de la incertidumbre es, directa o indirectamente, obtener un conjunto de entrenamiento donde el error del pronóstico esté bien representado. Durante la experimentación, se detectó cierta sensibilidad en el desempeño de las redes a la hora de estimar las varianzas y covarianzas del error al método de asimilación utilizado para generar los análisis que se usan como objetivo de entrenamiento. En particular, se observó que entrenar usando un ensamble de análisis dio mejores resultados que entrenar con el análisis medio. Pero el hecho de que miembros del ensamble pueden agruparse y persistentemente desviarse de la media, afecta negativamente al entrenamiento de los modelos aun cuando los análisis medios generados reportan el mismo rendimiento en términos del error cuadrático medio. El filtro de Kalman estocástico no sufre este problema debido al muestreo aleatorio en la actualización de cada miembro del ensamble, no obstante, creemos que este punto debería ser estudiado con más profundidad.

Redes de observación

En los experimentos de DA realizados y analizados en esta tesis, se utilizó una red de observaciones regulares y constantes en el tiempo. Es decir, en cada paso de asimilación está disponible la misma cantidad de observaciones y en el mismo punto de retícula. Esta hipótesis no sólo no es realista, sino que impone una estructura determinada en la incertidumbre de las condiciones iniciales, generado que las RNAs aprendan implícita-

mente la estructura de la red de observación subyacente. Si bien esto se podría subsanar incorporando a la entrada de la red la incertidumbre asociada a la condición inicial del pronóstico, es necesario continuar investigando la flexibilidad de la metodología en escenarios en los que las redes de observación cambian en el tiempo.

Eventos extremos

La representación y pronóstico de eventos extremos es un problema que ocupa activamente a la comunidad científica en geociencias. Los modelos estadísticos basados en datos históricos suelen tener problemas para representar adecuadamente situaciones extremas debido a la baja frecuencia de estos eventos en las bases de datos. Por otra parte, los modelos dinámicos suelen tener limitaciones a la hora de representar eventos extremos asociados a escalas pequeñas debido a los errores de truncado y a la parametrización de los procesos de pequeña escala. También en pronósticos probabilísticos, cuando se aproximan distribuciones desconocidas mediante gaussianas suele subestimarse la probabilidad real en las colas de la distribución. Si bien en este trabajo de tesis se analizó la calibración del pronóstico y el rango de respuesta de las variables de estado, consideramos que un análisis más profundo debe llevarse a cabo para el caso particular de los eventos extremos. En este sentido, el método de entrenamiento eMSE se podría extender para estimar cualquier momento centrado de la distribución del error que uno desee agregando a la función de costo el término enésimo correspondiente $T_n = (\tilde{\sigma}_l^n - \tilde{\epsilon}_l^n)^\top (\tilde{\sigma}_l^n - \tilde{\epsilon}_l^n)$ siendo $\tilde{\sigma}_l^n$ la salida de la red. Esto abre la posibilidad de poder estimar distribuciones unimodales no gaussianas. También, como trabajo futuro, se puede investigar la posibilidad de estimar mezclas de gaussianas para representar la forma de la función de distribución de probabilidad. De esta manera, se buscaría representar mejor la frecuencia de eventos extremos y, al mismo tiempo, considerar la evolución no-gaussiana de la función de distribución de probabilidad que suele ocurrir en plazos de pronóstico más largos, cuando la dinámica del error entra en el régimen no lineal.

Arquitecturas de RNA

En los experimentos de esta tesis se utilizaron arquitecturas de red, del inglés multi-layer Perceptron (MLP) y red neuronal convolucional, en inglés Convolutional Neural Network (CNN), relativamente sencillas. Esto fue suficiente para representar la dinámica del error en el modelo de Lorenz'96 a dos escalas. Sin embargo, consideramos que se debe continuar la investigación evaluando arquitecturas más complejas y específicas

que representen adecuadamente la información y el conocimiento de la dinámica de la atmósfera y el océano. Por ejemplo, en este trabajo se constató que los MLP mejoran su desempeño al incorporar en la entrada información de la evolución temporal del sistema dinámico. Esto sugiere que arquitecturas recurrentes, como las LSTM del inglés Long-Short Term Memory (Srivastava et al., 2015), podrían mejorar aún más el desempeño como se muestra en Amemiya et al. (2023). También los campos meteorológicos podrían ser mejor representados mediante arquitecturas como las GN del inglés Graph Network (Battaglia et al., 2018) o GAT del inglés Graph Attention Network (Velicković et al., 2018) especializadas en representar conocimiento estructurado en retículas, o las arquitecturas IRN Implicit Neural Representations que, por el contrario, aprenden una función continua dado un conjunto de puntos como las redes WIRE del inglés Wavelet Implicit Neural Representations (Saragadam et al., 2023) o las SIREN del inglés Implicit Neural Representations with Periodic Activation Functions (Sitzmann et al., 2020)

Tamaño del conjunto de datos

El tamaño del conjunto de datos disponibles para entrenar las RNAs en los experimentos de esta tesis se mantuvo constante, buscando mantener una equivalencia con la longitud de las bases de datos de reanálisis atmosféricos disponibles que suele ser de alrededor de 30 años. Esta equivalencia se estableció en base a los tiempos de crecimiento de los errores en la escala sinóptica en la atmósfera y en el modelo de Lorenz. Si se tiene en cuenta que la gran mayoría de los antecedentes analizados plantean el problema de los conjuntos de entrenamientos pequeños y el sobreajuste para estas metodologías, consideramos que se debería llevar a cabo un análisis más profundo del impacto y sensibilidad al tamaño del conjunto de entrenamiento, especialmente para experimentos de muy alta dimensionalidad.

Perspectivas en el pronóstico

Aunque el uso de técnicas de aprendizaje automático, en inglés Machine Learning (ML), para la generación de pronósticos en sistemas dinámicos caóticos no fue el tema central de esta tesis, es un área de investigación muy activa como se pudo comprobar en los antecedentes analizados para este trabajo. Uno de los principales atractivos es que los modelos basados en ML son mucho menos costosos en términos de cómputo, con lo cual correr un ensamble de miles de miembros podría ser equivalente a integrar un modelo físico (o quizás aún más barato). Pero los errores de modelo y su representación

van a seguir siendo un tema relevante en el contexto de modelos basados en ML, por lo que tener técnicas que puedan corregir sesgos, tanto en la estimación de la media como en la estimación de la incertidumbre, tienen un lugar importante en posibles sistemas de pronóstico operativos basados en ML. En este sentido, entendemos que las conclusiones y resultados de este trabajo siguen siendo aplicables e interesantes en este contexto donde, por ejemplo, grandes ensambles generados con dichos modelos pueden ser combinados con nuestras técnicas para mejorar aún más la cuantificación de su incertidumbre asociada.

Epílogo

En esta tesis se mostró que nuestro método de entrenamiento de redes neuronales puede corregir sesgos en los pronósticos, tanto en la estimación de la media como en la estimación de la incertidumbre, utilizando estados analizados históricos. Asimismo, puede proporcionar un pronóstico probabilístico fiable a partir de un único pronóstico determinístico, independientemente si éste proviene de la integración de un modelo físico o de otra técnica de pronóstico, como los análogos o modelos basados en aprendizaje automático.

Además, la metodología propuesta de asimilación de datos secuenciales permite generar análisis con error cuadrático medio comparables al uso de grandes ensambles, pero con el costo computacional de un único pronóstico.

Por último, confiamos que esta tesis sienta un precedente sólido para continuar la investigación y el desarrollo de metodologías híbridas de ML y DA, extendiendo así el alcance de nuestros resultados a sistemas de muy alta dimensionalidad como la atmósfera y el océano. Y abriendo la posibilidad de generar sistemas operativos de asimilación de datos y predicción meteorológica a centros meteorológicos donde la capacidad de cálculo es limitada, como en los países en desarrollo donde metodologías como 4Dvar o EnKF requieren de un gran esfuerzo en términos del costo computacional asociadas a las mismas.

Bibliografía

- Amemiya, A., M. Shlok, and T. Miyoshi, 2023: Application of recurrent neural networks to model bias correction: Idealized experiments with the lorenz-96 model. *Journal of Advances in Modeling Earth Systems*, **15** (2), e2022MS003164, doi:<https://doi.org/10.1029/2022MS003164>, URL <https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/2022MS003164>, e2022MS003164 2022MS003164, <https://agupubs.onlinelibrary.wiley.com/doi/pdf/10.1029/2022MS003164>.
- Amezcuá, J., K. Ide, C. Bishop, and E. Kalnay, 2012: Ensemble clustering in deterministic ensemble kalman filters. *Tellus A*, **64**, doi:10.3402/tellusa.v64i0.18039.
- Andrew Ng, MOOC: Improving deep neural networks: Hyperparameter tuning, regularization and optimization. DeepLearning.AI.
- Bannister, R. N., 2017: A review of operational methods of variational and ensemble-variational data assimilation. *Quarterly Journal of the Royal Meteorological Society*, **143** (703), 607–633, doi:<https://doi.org/10.1002/qj.2982>, URL <https://rmets.onlinelibrary.wiley.com/doi/abs/10.1002/qj.2982>, <https://rmets.onlinelibrary.wiley.com/doi/pdf/10.1002/qj.2982>.
- Barnes, E., and R. Barnes, 2021: Controlled abstention neural networks for identifying skillful predictions for regression problems. *Journal of Advances in Modeling Earth Systems*, **13**, doi:10.1029/2021MS002575.
- Battaglia, P. W., and Coauthors, 2018: Relational inductive biases, deep learning, and graph networks. [1806.01261](https://arxiv.org/abs/1806.01261).
- Bishop, C., 2006: *Pattern Recognition and Machine Learning*. Information Science and Statistics, Springer, URL <https://books.google.com.ar/books?id=qWPwnQEACAAJ>.

- Bishop, C., and C. S. Qazaz, 1997: Regression with input-dependent noise: A bayesian treatment. *Advances in Neural Information Processing Systems*, MIT Press, Vol. 9, 347–353, URL <https://www.microsoft.com/en-us/research/publication/regression-with-input-dependent-noise-a-bayesian-treatment/>.
- Bocquet, M., J. Brajard, A. Carrassi, and L. Bertino, 2019: Data assimilation as a learning tool to infer ordinary differential equation representations of dynamical models. *Nonlinear Processes in Geophysics*, **26** (3), 143–162, doi:10.5194/npg-26-143-2019, URL <https://npg.copernicus.org/articles/26/143/2019/>.
- Bonavita, M., and P. Laloyaux, 2020: Machine learning for model error inference and correction. *Journal of Advances in Modeling Earth Systems*, **12** (12), e2020MS002232, doi:<https://doi.org/10.1029/2020MS002232>, URL <https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/2020MS002232>, e2020MS002232 10.1029/2020MS002232, <https://agupubs.onlinelibrary.wiley.com/doi/pdf/10.1029/2020MS002232>.
- Brajard, J., A. Carrassi, M. Bocquet, and L. Bertino, 2020: Combining data assimilation and machine learning to emulate a dynamical model from sparse and noisy observations: A case study with the lorenz 96 model. *Journal of Computational Science*, **44**, 101 171, doi:<https://doi.org/10.1016/j.jocs.2020.101171>, URL <https://www.sciencedirect.com/science/article/pii/S1877750320304725>.
- Brajard, J., A. Carrassi, M. Bocquet, and L. Bertino, 2021: Combining data assimilation and machine learning to infer unresolved scale parametrization. *Philos Trans A Math Phys Eng Sci*, **379** (2194), 20200 086, doi:10.1098/rsta.2020.0086.
- Bremnes, J. B., 2020: Ensemble postprocessing using quantile function regression based on neural networks and bernstein polynomials. *Monthly Weather Review*, **148** (1), 403 – 414, doi:10.1175/MWR-D-19-0227.1, URL <https://journals.ametsoc.org/view/journals/mwre/148/1/mwr-d-19-0227.1.xml>.
- Brusdal, K., J. Brankart, G. Halberstadt, G. Evensen, P. Brasseur, P. van Leeuwen, E. Dombrowsky, and J. Verron, 2003: A demonstration of ensemble-based assimilation methods with a layered ogcm from the perspective of operational ocean forecasting systems. *Journal of Marine Systems*, **40–41**, 253–289, doi:[https://doi.org/10.1016/S0924-7963\(03\)00021-6](https://doi.org/10.1016/S0924-7963(03)00021-6), URL <https://www.sciencedirect.com/science/article/pii/S0924796303000216>, the Use of Data Assimilation in Coupled Hydrodynamic, Ecological and Bio-geo-chemical Models of the Ocean. Selected papers from the 33rd Inter-

national Liege Colloquium on Ocean Dynamics, held in Liege, Belgium on May 7-11th, 2001.

- Buizza, C., and Coauthors, 2022: Data learning: Integrating data assimilation and machine learning. *Journal of Computational Science*, **58**, 101525, doi:<https://doi.org/10.1016/j.jocs.2021.101525>, URL <https://www.sciencedirect.com/science/article/pii/S1877750321001861>.
- Burgers, G., P. J. van Leeuwen, and G. Evensen, 1998: Analysis scheme in the ensemble kalman filter. *Monthly Weather Review*, **126** (6), 1719 – 1724, doi:[https://doi.org/10.1175/1520-0493\(1998\)126<1719:ASITEK>2.0.CO;2](https://doi.org/10.1175/1520-0493(1998)126<1719:ASITEK>2.0.CO;2), URL https://journals.ametsoc.org/view/journals/mwre/126/6/1520-0493_1998_126_1719_asitek_2.0.co_2.xml.
- Camporeale, E., and A. Carè, 2021: Accrue: Accurate and reliable uncertainty estimate in deterministic models. *International Journal for Uncertainty Quantification*, **11** (4), 81–94.
- Camporeale, E., X. Chu, O. V. Agapitov, and J. Bortnik, 2019: On the generation of probabilistic forecasts from deterministic models. *Space Weather*, doi:10.1029/2018SW002026.
- Candille, G., C. Côté, P. Houtekamer, and G. Pellerin, 2007: Verification of an ensemble prediction system against observation. *Monthly Weather Review*, **135**, doi:10.1175/MWR3414.1.
- Cannon, A. J., 2018: Non-crossing nonlinear regression quantiles by monotone composite quantile regression neural network, with application to rainfall extremes. *Stochastic Environ. Res. Risk Assess.*, **32**, 3207–3225.
- Carrassi, A., M. Bocquet, L. Bertino, and G. Evensen, 2018: Data assimilation in the geosciences: An overview of methods, issues, and perspectives. *WIREs Climate Change*, **9** (5), e535, doi:<https://doi.org/10.1002/wcc.535>.
- Chapman, W. E., L. D. Monache, S. Alessandrini, A. C. Subramanian, F. M. Ralph, S.-P. Xie, S. Lerch, and N. Hayatbini, 2022: Probabilistic predictions from deterministic atmospheric river forecasts with deep learning. *Monthly Weather Review*, **150** (1), 215 – 234, doi:10.1175/MWR-D-21-0106.1, URL <https://journals.ametsoc.org/view/journals/mwre/150/1/MWR-D-21-0106.1.xml>.

- Cheng, S., and Coauthors, 2023: Machine learning with data assimilation and uncertainty quantification for dynamical systems: a review. *arXiv preprint arXiv:2303.10462*.
- Christensen, H. M., I. M. Moroz, and T. N. Palmer, 2015: Stochastic and perturbed parameter representations of model uncertainty in convection parameterization. *Journal of the Atmospheric Sciences*, **72** (6), 2525 – 2544, doi:<https://doi.org/10.1175/JAS-D-14-0250.1>, URL <https://journals.ametsoc.org/view/journals/atsc/72/6/jas-d-14-0250.1.xml>.
- Clare, M., O. Jamil, and C. Morcrette, 2021: A computationally efficient neural network for predicting weather forecast probabilities. [2103.14430](https://arxiv.org/abs/2103.14430).
- CS231n, MOOC: Convolutional neural networks for visual recognition. <https://cs231n.github.io/neural-networks-3/>.
- Dugas, C., Y. Bengio, F. Elisle, and C. Nadeau, 2001: Incorporating second-order functional knowledge for better option pricing. *Cirano Working Papers*.
- D’Isanto, A., and K. L. Polsterer, 2018: Photometric redshift estimation via deep learning. *Astronomy & Astrophysics*, **609**, A111, doi:10.1051/0004-6361/201731326.
- Evensen, G., 2003: The ensemble kalman filter: theoretical formulation and practical implementation. *Ocean Dynamics*, doi:10.1007/s10236-003-0036-9, URL <https://doi.org/10.1007/s10236-003-0036-9>.
- Faranda, D., G. Messori, and P. Yiou, 2017: Dynamical proxies of North Atlantic predictability and extremes. *Scientific Reports*, doi:10.1038/srep41278, URL <https://hal.science/hal-01340301>.
- Farchi, A., M. Bocquet, P. Laloyaux, M. Bonavita, and Q. Malartic, 2021a: A comparison of combined data assimilation and machine learning methods for offline and online model error correction. *Journal of Computational Science*, **55**, 101 468, doi:<https://doi.org/10.1016/j.jocs.2021.101468>, URL <https://www.sciencedirect.com/science/article/pii/S1877750321001435>.
- Farchi, A., M. Chrut, M. Bocquet, P. Laloyaux, and M. Bonavita, 2022: Online model error correction with neural networks in the incremental 4d-var framework. [2210.13817](https://arxiv.org/abs/2210.13817).

-
- Farchi, A., P. Laloyaux, M. Bonavita, and M. Bocquet, 2021b: Using machine learning to correct model error in data assimilation and forecast applications. *Quarterly Journal of the Royal Meteorological Society*, **147**, doi:10.1002/qj.4116.
- Ferranti, L., S. Corti, and M. Janousek, 2015: Flow-dependent verification of the ecmwf ensemble over the euro-atlantic sector. *Quarterly Journal of the Royal Meteorological Society*, **141** (688), 916–924, doi:<https://doi.org/10.1002/qj.2411>, URL <https://rmets.onlinelibrary.wiley.com/doi/abs/10.1002/qj.2411>, <https://rmets.onlinelibrary.wiley.com/doi/pdf/10.1002/qj.2411>.
- Gandin, L., 1965: Objective analysis of meteorological fields: Gidrometeorologicheskoe izdatel'stvo (gimiz), leningrad. *Translated by Israel Program for Scientific Translations, Jerusalem*.
- Gaspari, G., and S. E. Cohn, 1999: Construction of correlation functions in two and three dimensions. *Quarterly Journal of the Royal Meteorological Society*, **125** (554), 723–757, doi:<https://doi.org/10.1002/qj.49712555417>, URL <https://rmets.onlinelibrary.wiley.com/doi/abs/10.1002/qj.49712555417>, <https://rmets.onlinelibrary.wiley.com/doi/pdf/10.1002/qj.49712555417>.
- Gasthaus, J., K. Benidis, Y. Wang, S. S. Rangapuram, D. Salinas, V. Flunkert, and T. Januschowski, 2019: Probabilistic forecasting with spline quantile function rnns. *Proceedings of the Twenty-Second International Conference on Artificial Intelligence and Statistics*, K. Chaudhuri, and M. Sugiyama, Eds., PMLR, Proceedings of Machine Learning Research, Vol. 89, 1901–1910, URL <https://proceedings.mlr.press/v89/gasthaus19a.html>.
- Gneiting, T., F. Balabdaoui, and A. E. Raftery, 2007: Probabilistic forecasts, calibration and sharpness. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, **69** (2), 243–268, doi:<https://doi.org/10.1111/j.1467-9868.2007.00587.x>, URL <https://rss.onlinelibrary.wiley.com/doi/abs/10.1111/j.1467-9868.2007.00587.x>, <https://rss.onlinelibrary.wiley.com/doi/pdf/10.1111/j.1467-9868.2007.00587.x>.
- Goodfellow, I., Y. Bengio, and A. Courville, 2016: *Deep Learning*. Adaptive Computation and Machine Learning series, MIT Press, URL <https://books.google.com.ar/books?id=Np9SDQAAQBAJ>.
- Grooms, I., 2021: Analog ensemble data assimilation and a method for constructing analogs with variational autoencoders. *Quarterly Journal of the Royal Meteorological Society*, **147** (734), 139–149, doi:<https://doi.org/10.1002/qj.3910>.

- Grönquist, P., C. Yao, T. Ben-Nun, N. Dryden, P. Dueben, S. Li, and T. Hoefler, 2021: Deep learning for post-processing ensemble weather forecasts. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, **379** (2194), 20200 092, doi:10.1098/rsta.2020.0092.
- Guan, H., and Coauthors, 2022: Gefsv12 reforecast dataset for supporting subseasonal and hydrometeorological applications. *Monthly Weather Review*, **150** (3), 647 – 665, doi:10.1175/MWR-D-21-0245.1, URL <https://journals.ametsoc.org/view/journals/mwre/150/3/MWR-D-21-0245.1.xml>.
- Hagedorn, R., R. Buizza, T. M. Hamill, M. Leutbecher, and T. N. Palmer, 2012: Comparing tigge multimodel forecasts with reforecast-calibrated ecmwf ensemble forecasts. *Quarterly Journal of the Royal Meteorological Society*, **138** (668), 1814–1827, doi: <https://doi.org/10.1002/qj.1895>.
- Hamill, T., and J. Whitaker, 2006: Probabilistic quantitative precipitation forecasts based on reforecast analogs: Theory and application. *Monthly Weather Review*, **134**, doi:10.1175/MWR3237.1.
- Hamill, T. M., G. T. Bates, J. S. Whitaker, D. R. Murray, M. Fiorino, T. J. Galarneau, Y. Zhu, and W. Lapenta, 2013: Noaa’s second-generation global medium-range ensemble reforecast dataset. *Bulletin of the American Meteorological Society*, **94** (10), 1553 – 1565, doi: <https://doi.org/10.1175/BAMS-D-12-00014.1>, URL <https://journals.ametsoc.org/view/journals/bams/94/10/bams-d-12-00014.1.xml>.
- Hamill, T. M., J. S. Whitaker, and C. Snyder, 2001: Distance-dependent filtering of background error covariance estimates in an ensemble kalman filter. *Monthly Weather Review*, **129** (11), 2776 – 2790, doi:[https://doi.org/10.1175/1520-0493\(2001\)129<2776:DDFOBE>2.0.CO;2](https://doi.org/10.1175/1520-0493(2001)129<2776:DDFOBE>2.0.CO;2), URL https://journals.ametsoc.org/view/journals/mwre/129/11/1520-0493_2001_129_2776_ddfobe_2.0.co_2.xml.
- Härter, F. P., and H. F. de Campos Velho, 2008: New approach to applying neural network in nonlinear dynamic model. *Applied Mathematical Modelling*, **32** (12), 2621–2633, doi:<https://doi.org/10.1016/j.apm.2007.09.006>, URL <https://www.sciencedirect.com/science/article/pii/S0307904X07002296>.
- Haupt, S. E., W. Chapman, S. V. Adams, C. Kirkwood, J. S. Hosking, N. H. Robinson, S. Lerch, and A. C. Subramanian, 2021: Towards implementing artificial intelligence post-processing in weather and climate:proposed actions from the oxford 2019

-
- workshop. *Philosophical transactions. Series A, Mathematical, physical, and engineering sciences*, **379** (2194), doi:10.1098/rsta.2020.0091.
- Herman, G. R., and R. S. Schumacher, 2018: Money Doesn't Grow on Trees, but Forecasts Do: Forecasting Extreme Precipitation with Random Forests. *Monthly Weather Review*, **146** (5), 1571–1600, doi:10.1175/MWR-D-17-0250.1.
- Hou, D., E. Kalnay, and K. Droegemeier, 2001: Objective verification of the samex '98 ensemble forecasts. *Monthly Weather Review - MON WEATHER REV*, **129**, doi:10.1175/1520-0493(2001)129<0073:OVOTSE>2.0.CO;2.
- Houtekamer, P. L., and F. Zhang, 2016: Review of the ensemble kalman filter for atmospheric data assimilation. *Monthly Weather Review*, **144** (12), 4489 – 4532, doi:10.1175/MWR-D-15-0440.1, URL <https://journals.ametsoc.org/view/journals/mwre/144/12/mwr-d-15-0440.1.xml>.
- Hsieh, W. W., and B. Tang, 1998: Applying neural network models to prediction and data analysis in meteorology and oceanography. *Bulletin of the American Meteorological Society*, **79** (9), 1855 – 1870, doi:https://doi.org/10.1175/1520-0477(1998)079<1855:ANNMTP>2.0.CO;2, URL https://journals.ametsoc.org/view/journals/bams/79/9/1520-0477_1998_079_1855_annmtp_2_0_co_2.xml.
- Hu, H., and G. Kantor, 2015: Parametric covariance prediction for heteroscedastic noise. *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 3052–3057, doi:10.1109/IROS.2015.7353798.
- Hunt, B. R., E. J. Kostelich, and I. Szunyogh, 2007: Efficient data assimilation for spatiotemporal chaos: A local ensemble transform kalman filter. *Physica D: Nonlinear Phenomena*, **230** (1), 112 – 126, doi:https://doi.org/10.1016/j.physd.2006.11.008, data Assimilation.
- James Le, MOOC: Neural networks 101. <https://jameskle.com/writes/neural-networks-101>.
- Kalnay, E., 2003: *Atmospheric Modeling, Data Assimilation and Predictability*. Cambridge University Press, doi:10.1017/CBO9780511802270.
- Kingma, D. P., and J. Ba, 2017: Adam: A method for stochastic optimization. [1412.6980](#).

- Kingma, D. P., and M. Welling, 2019: An Introduction to Variational Autoencoders. *arXiv e-prints*, arXiv:1906.02691, doi:10.48550/arXiv.1906.02691, [1906.02691](https://arxiv.org/abs/1906.02691).
- Kondo, K., and T. Miyoshi, 2019: Non-gaussian statistics in global atmospheric dynamics: a study with a 10 240-member ensemble kalman filter using an intermediate atmospheric general circulation model. *Nonlinear Processes in Geophysics*, **26** (3), 211–225, doi: 10.5194/npg-26-211-2019, URL <https://npg.copernicus.org/articles/26/211/2019/>.
- Lecun, Y., 1989: *Generalization and network design strategies*. Elsevier.
- Lguensat, R., P. Tandeo, P. Ailliot, M. Pulido, and R. Fablet, 2017: The analog data assimilation. *Monthly Weather Review*, **145** (10), 4093 – 4107, doi:<https://doi.org/10.1175/MWR-D-16-0441.1>, URL <https://journals.ametsoc.org/view/journals/mwre/145/10/mwr-d-16-0441.1.xml>.
- Liu, K., K. Ok, W. Vega-Brown, and N. Roy, 2018: Deep inference for covariance estimation: Learning gaussian noise models for state estimation. *2018 IEEE International Conference on Robotics and Automation (ICRA)*, 1436–1443, doi:10.1109/ICRA.2018.8461047.
- Lorenz, E., 1995: Predictability: a problem partly solved. ECMWF, Shinfield Park, Reading, 1-18 pp.
- Loshchilov, I., and F. Hutter, 2017: Decoupled weight decay regularization. arXiv, URL <https://arxiv.org/abs/1711.05101>, doi:10.48550/ARXIV.1711.05101.
- Lu, H., Y. Ou, C. Qin, and L. Jin, 2021: A fuzzy neural network bagging ensemble forecasting model for 72-h forecast of low-temperature chilling injury. *Natural Hazards: Journal of the International Society for the Prevention and Mitigation of Natural Hazards*, **105** (2), 2147–2160, doi:10.1007/s11069-020-04393-.
- MacKay, D. J., 1992: Bayesian methods for adaptive models. Ph.D. thesis, California Institute of Technology., doi:10.7907/H3A1-WM07., URL <https://resolver.caltech.edu/CaltechETD:etd-01042007-131447>.
- Marzban, C., 2003: Neural networks for postprocessing model output: Arps. *Monthly Weather Review*, **131** (6), 1103 – 1111, doi:10.1175/1520-0493(2003)131<1103:NNFPMO>2.0.CO;2, URL https://journals.ametsoc.org/view/journals/mwre/131/6/1520-0493_2003_131_1103_nnfpmo_2.0.co_2.xml.

-
- McCulloch, W. S., and W. Pitts, 1943: A logical calculus of the ideas immanent in nervous activity. *The bulletin of mathematical biophysics*, doi:10.1007/BF02478259, URL <https://doi.org/10.1007/BF02478259>.
- Murphy, A. H., 1993: What is a good forecast? an essay on the nature of goodness in weather forecasting. *Weather and Forecasting*, **8** (2), 281–293, doi:10.1175/1520-0434(1993)008<0281:WIAGFA>2.0.CO;2.
- Nix, D., and A. Weigend, 1994: Estimating the mean and variance of the target probability distribution. *Proceedings of 1994 IEEE International Conference on Neural Networks (ICNN'94)*, Vol. 1, 55–60 vol.1, doi:10.1109/ICNN.1994.374138.
- Ott, E., and Coauthors, 2004: A local ensemble kalman filter for atmospheric data assimilation. *Tellus A: Dynamic Meteorology and Oceanography*, **56** (5), 415–428, doi:10.3402/tellusa.v56i5.14462, URL <https://doi.org/10.3402/tellusa.v56i5.14462>, <https://doi.org/10.3402/tellusa.v56i5.14462>.
- Ouala, S., R. Fablet, C. Herzet, B. Chapron, A. Pascual, F. Collard, and L. Gaultier, 2018: Neural network based kalman filters for the spatio-temporal interpolation of satellite-derived sea surface temperature. *Remote Sensing*, **10** (12), doi:10.3390/rs10121864, URL <https://www.mdpi.com/2072-4292/10/12/1864>.
- Parrish, D. F., and J. C. Derber, 1992: The national meteorological center's spectral statistical-interpolation system. 1747–1763 pp.
- Pulido, M., G. Scheffler, J. J. Ruiz, M. M. Lucini, and P. Tandeo, 2016: Estimation of the functional form of subgrid-scale parametrizations using ensemble-based data assimilation: a simple model experiment. *Quarterly Journal of the Royal Meteorological Society*, **142** (701), 2974–2984, doi:<https://doi.org/10.1002/qj.2879>.
- Rabier, F., H. Järvinen, E. Klinker, J.-F. Mahfouf, and A. Simmons, 2000: The ecmwf operational implementation of four-dimensional variational assimilation. i: Experimental results with simplified physics. *Quarterly Journal of the Royal Meteorological Society*, **126** (564), 1143–1170, doi:<https://doi.org/10.1002/qj.49712656415>, URL <https://rmets.onlinelibrary.wiley.com/doi/abs/10.1002/qj.49712656415>, <https://rmets.onlinelibrary.wiley.com/doi/pdf/10.1002/qj.49712656415>.
- Rasp, S., and S. Lerch, 2018: Neural networks for postprocessing ensemble weather forecasts. *Monthly Weather Review*, **146** (11), 3885–3900, doi:10.1175/mwr-d-18-0187.1.

- Rosenblatt, F., 1962: *Principles of Neurodynamics: Perceptrons and the Theory of Brain Mechanisms*. Cornell Aeronautical Laboratory. Report no. VG-1196-G-8, Spartan Books, URL <https://books.google.com.ar/books?id=7FhRAAAAMAAJ>.
- Rumelhart, D. E., G. E. Hinton, and R. J. Williams, 1986: Learning representations by back-propagating errors. *Nature*, 533 – 536, doi:10.1038/323533a0, URL <https://doi.org/10.1038/323533a0>.
- Saragadam, V., D. LeJeune, J. Tan, G. Balakrishnan, A. Veeraraghavan, and R. Baraniuk, 2023: Wire: Wavelet implicit neural representations.
- Scheffler, G., J. Ruiz, and M. Pulido, 2019: Inference of stochastic parametrizations for model error treatment using nested ensemble kalman filters. *Quarterly Journal of the Royal Meteorological Society*, **145** (722), 2028–2045, doi:<https://doi.org/10.1002/qj.3542>.
- Scher, S., and G. Messori, 2018: Predicting weather forecast uncertainty with machine learning. *Quarterly Journal of the Royal Meteorological Society*, doi:10.1002/qj.3410.
- Scheuerer, M., M. B. Switanek, R. P. Worsnop, and T. M. Hamill, 2020: Using artificial neural networks for generating probabilistic subseasonal precipitation forecasts over california. *Monthly Weather Review*, **148** (8), 3489 – 3506, doi:10.1175/MWR-D-20-0096.1, URL <https://journals.ametsoc.org/view/journals/mwre/148/8/mwrD200096.xml>.
- Schlosser, L., T. Hothorn, R. Stauffer, and A. Zeileis, 2019: Distributional regression forests for probabilistic precipitation forecasting in complex terrain. *The Annals of Applied Statistics*, **13** (3), 1564 – 1589, doi:10.1214/19-AOAS1247, URL <https://doi.org/10.1214/19-AOAS1247>.
- Sitzmann, V., J. N. P. Martel, A. W. Bergman, D. B. Lindell, and G. Wetzstein, 2020: Implicit neural representations with periodic activation functions. [2006.09661](https://arxiv.org/abs/2006.09661).
- Srivastava, N., E. Mansimov, and R. Salakhutdinov, 2015: Unsupervised learning of video representations using lstms. *Proceedings of the 32nd International Conference on International Conference on Machine Learning - Volume 37*, JMLR.org, 843–852, ICML’15.
- Taillardat, M., O. Mestre, M. Zamo, and P. Naveau, 2016: Calibrated ensemble forecasts using quantile regression forests and ensemble model output statistics. *Monthly Weather Review*, **144** (6), 2375 – 2393, doi:10.1175/MWR-D-15-0260.1, URL <https://journals.ametsoc.org/view/journals/mwre/144/6/mwr-d-15-0260.1.xml>.

-
- Tandeo, P., P. Ailliot, M. Bocquet, A. Carrassi, T. Miyoshi, M. Pulido, and Y. Zhen, 2020: A review of innovation-based methods to jointly estimate model and observation error covariance matrices in ensemble data assimilation. *Monthly Weather Review*, **148** (10), 3973 – 3994, doi:<https://doi.org/10.1175/MWR-D-19-0240.1>, URL <https://journals.ametsoc.org/view/journals/mwre/148/10/mwrD190240.xml>.
- van den Dool, H. M., 1994: Searching for analogues, how long must we wait? *Tellus A: Dynamic Meteorology and Oceanography*, doi:10.3402/tellusa.v46i3.15481.
- van Leeuwen, P. J., H. R. Künsch, L. Nerger, R. Potthast, and S. Reich, 2019: Particle filters for high-dimensional geoscience applications: A review. *Quarterly Journal of the Royal Meteorological Society*, **145** (723), 2335–2365, doi:<https://doi.org/10.1002/qj.3551>, URL <https://rmets.onlinelibrary.wiley.com/doi/abs/10.1002/qj.3551>, <https://rmets.onlinelibrary.wiley.com/doi/pdf/10.1002/qj.3551>.
- Vannitsem, S., and Coauthors, 2021: Statistical postprocessing for weather forecasts: Review, challenges, and avenues in a big data world. *Bulletin of the American Meteorological Society*, **102** (3), E681 – E699, doi:10.1175/BAMS-D-19-0308.1, URL <https://journals.ametsoc.org/view/journals/bams/102/3/BAMS-D-19-0308.1.xml>.
- Veldkamp, S., K. Whan, S. Dirksen, and M. Schmeits, 2021: Statistical postprocessing of wind speed forecasts using convolutional neural networks. *Monthly Weather Review*, **149** (4), 1141 – 1152, doi:10.1175/MWR-D-20-0219.1, URL <https://journals.ametsoc.org/view/journals/mwre/149/4/MWR-D-20-0219.1.xml>.
- Velicković, P., G. Cucurull, A. Casanova, A. Romero, P. Liò, and Y. Bengio, 2018: Graph attention networks. [1710.10903](https://arxiv.org/abs/1710.10903).
- Wang, B., J. Lu, Z. Yan, H. Luo, T. Li, Y. Zheng, and G. Zhang, 2018: Deep uncertainty quantification: A machine learning approach for weather forecasting. [1812.09467](https://arxiv.org/abs/1812.09467).
- Widrow, B., and M. E. Hoff, 1960: Adaptive switching circuits. *Neurocomputing, Volume 1: Foundations of Research*, 960 IRE WESCON Convention Record[i], doi:10.7551/mitpress/4943.003.0012, URL <https://doi.org/10.7551/mitpress/4943.003.0012>, https://direct.mit.edu/book/chapter-pdf/2049887/c010053_9780262267137.pdf.
- Wilks, D. S., 2019: Indices of rank histogram flatness and their sampling properties. *Monthly Weather Review*, **147** (2), 763 – 769, doi:10.1175/MWR-D-18-0369.1, URL <https://journals.ametsoc.org/view/journals/mwre/147/2/mwr-d-18-0369.1.xml>.

- Williams, P. M., 1996: Using neural networks to model conditional multivariate densities. *Neural Computation*, **8** (4), 843–854, doi:10.1162/neco.1996.8.4.843.
- Yang, L. M., and I. Grooms, 2021: Machine learning techniques to construct patched analog ensembles for data assimilation. *Journal of Computational Physics*, **443**, 110 532, doi:<https://doi.org/10.1016/j.jcp.2021.110532>, URL <https://www.sciencedirect.com/science/article/pii/S0021999121004277>.



UNIVERSIDAD DE BUENOS AIRES
Facultad de Ciencias Exactas y Naturales
Departamento de Ciencias de la Atmósfera y los Océanos

Cuantificación de la incertidumbre en los pronósticos usando redes neuronales: aplicación en asimilación de datos

Tesis presentada para optar por el título de Doctor de la Universidad de Buenos Aires
en el área Ciencias de la Atmósfera y los Océanos

Lic. Maximiliano Antonio Sacco

Director de Tesis: *Dr. Juan Jose Ruiz*

Co-Director de Tesis: *Dr. Manuel Pulido*

Consejero de estudios: *Dr. Enrique Carlos Segura*

Lugar de Trabajo: Centro de Investigaciones del Mar y de la Atmósfera. CONICET-UBA
Servicio Meteorológico Nacional Argentino.

Buenos Aires, Agosto 2023