Predicción de Sistemas Dinámicos con Redes Neuronales Profundas

Daniel G. Maino, Lucas C. Uzal, and Pablo M. Granitto

Grupo de Aprendizaje Automatizado y Aplicaciones
CIFASIS (CONICET-UNR-UPCAM)
Centro Internacional Franco-Argentino de Ciencias de la Información y de Sistemas

Resumen En este trabajo se aborda el problema de predicción de series temporales obtenidas de sistemas dinámicos no lineales deterministicos. Se presenta una técnica basada en redes neuronales profundas y se evalúa su rendimiento frente a las redes neuronales convencionales. Se considera en particular el problema predicción para múltiples horizontes utilizando dos estrategias: el uso de redes de salida-múltiple frente a redes convencionales de salida-simple. Los resultados sobre las series temporales consideradas muestran un mejor desempeño de las arquitecturas profundas de salida simple.

1. Introducción

La predicción de series temporales es un campo de interés creciente que juega un papel importante en casi todos los campos de la ciencia y de la ingeniería, tales como la economía, finanzas, meteorología y telecomunicaciones.

La predicción de valores futuros de la serie temporal se realiza basándose en valores previos y el valor actual de la serie temporal, tales valores son usados como entrada en el modelo de predicción $\hat{x}_{t+h} = f_h(x_t, ..., x_{t-t_w}) + \epsilon_h$, donde $\hat{x}_{t+h}$ es la predicción h pasos hacia adelante con respecto al tiempo t, f_h es la función que modela las dependencias entre las observaciones pasadas y futuras, $t_w + 1 = m$ es el tamaño de entrada de la función f, y ϵ_h representa el error del modelado.

Si se dispone de una reconstrucción del espacio de fases [19] utilizando coordanadas de retraso $\bar{x}(t) = [x(t), x(t-\tau), x(t-2\tau), ..., x(t-(m-1)\tau)]$ es posible construir un modelo $F_h : \mathbb{R}^n \to \mathbb{R}$, de predicción ($h$ pasos hacia adelante) sobre este espacio de fases reconstruido. El valor $\hat{x}(t+h) = F_h(\bar{x}(t))$ obtenido corresponde a una única observación futura de la serie temporal.

La reconstrucción de coordenadas de retraso requiere fijar dos parámetros libres: el tiempo de retraso τ y la dimensión de *embedding* m. Estos determinan la ventana temporal $t_w = (m-1)\tau$ correspondiente a cada estado $\bar{x}(t)$ de la reconstrucción.

Uno de los desafíos en la predicción de series temporales es la predicción a *largo plazo*, la cual es en general más compleja en comparación a la predicción a *corto plazo*. Para el caso especial de las series de carácter caótico consideradas

en este trabajo existe una divergencia exponencial en la evolución de estados vecinos y por ende una complejización de la ley que determina su estado para horizontes H crecientes.

1.1. Predicción iterativa vs. predicción directa

Existen dos métodos para la predicción en un horizonte H lejano: el *iterativo* (o recursivo) y el *directo*. En el primero, una predicción con horizonte H es llevada a cabo por iterar H veces un predictor de *paso simple* (one-step-ahead). Una vez que el predictor ha estimado el valor $\hat{x}_{N+1}$, éste es reinyectado como parte de la entrada de tamaño m, para obtener la siguiente predicción, así hasta realizar H iteraciones de un *paso simple*. Por otro lado, el método *directo* consiste en predecir a un horizonte H ajustando directamente un modelo para predecir este valor. Esta estrategia toma como entrada solamente valores de la serie temporal, sin recurrir a iteraciones. De esta forma, los errores que se cometen en realizar las próximas predicciones no son acumulables porque solo se usan como entrada datos de la serie temporal. Cuando se desean predecir todos los valores de $\hat{x}_{N+1}$ a $\hat{x}_{N+H}$, se deben estimar H modelos diferentes. Dado el carácter caótico de las series temporales, el modelo requiere mayor complejidad que el método *iterativo*.

Según el estudio teórico y numérico desarrollado en [6] los métodos directos (en particular, modelos polinómicos locales) presentan mayor error que sus equivalentes métodos iterativos.

1.2. Redes Neuronales Profundas

Si bien las redes neuronales de tan solo una capa oculta resultan aproximadores universales de funciones continuas [5], al momento de modelar funciones de alta variabilidad estas requieren de un número exponencialmente mayor de neuronas que una arquitectura de mayor profundidad [3]. En consecuencia, surge la necesidad de implementar las *arquitecturas profundas*, las cuales poseen muchas capas de componentes adaptativos *no-lineales*, permitiendo la representación de una amplia familia de funciones de manera más compacta que las arquitecturas poco profundas utilizadas habitualmente.

El método estándar de aprendizaje consiste básicamente en inicializar los pesos de la red neuronal con valores aleatorios y luego ajustarlos minimizando una función de error utilizando un algoritmo de descenso por gradiente. Se sabe que de esta forma se obtienen soluciones pobres para redes neuronales profundas debido a que el descenso de gradiente fácilmente puede quedar atrapado en un mínimo local. Este problema se resolvió a partir de 2007 donde se propuso un pre-entrenamiento de cada capa en forma secuencial, el cual consiste en un entrenamiento *no supervisado* mediante RBMs (*Máquinas de Boltzmann Restringidas*) [7]. Esto permite inicializar los parámetros de la red profunda en una región cerca del óptimo buscado, para luego aplicar algoritmos de descenso por el gradiente realizando así un *ajuste fino* (*fine-tune*). Las arquitecturas profundas fueron aplicadas desde entonces en numerosas áreas: problemas de

reconocimiento de imágenes [8], reconocimiento de voz [16], modelado acústico [15], secuencia de video [4], entre otros.

Recientemente ha quedado en evidencia que resulta posible entrenar ciertas arquitecturas profundas sin necesidad un pre-entrenamiento, si se dispone una cantidad suficientemente grande de datos supervisados (etiquetados) [9]. Entre los elementos clave para que esto sea posible [10] se encuentra el reemplazo de la función de activación sigmoidea utilizada convencionalmente, por unidades de activación *ReLUs* (Rectified Linear Units) cuya función de activación es $f(x) = max(0, x)$ ó, en su versión diferenciable $f(x) = \log(1 + \exp(x))$. Con este tipo de unidad de activación, la cantidad de épocas de *backpropagation*, es aproximadamente 6 veces menor que al usar unidades sigmoideas. Además, el tipo de saturación de estas unidades reduce el estancamiento en mínimos locales del descenso por gradiente. Estos resultados motivaron a utilizar las unidades *ReLUs* en este trabajo, sin pre-entrenamiento dado que se dispone de gran cantidad de datos para realizar un entrenamiento supervisado. Pruebas preliminares considerando un pre-entrenamiento no supervisado con RBMs no mostraron mejoras respecto del entrenamiento adoptado en este trabajo.

1.3. Salida-Múltiple vs. Salida-Simple

Por lo general, cuando una aplicación práctica requiere una predicción de largo plazo de una serie temporal, suele ser necesario predecir, además de x_{t+H}, toda la secuencia de valores entre x_t e x_{t+H}. La opción más simple es entrenar H modelos independientes. Las redes de *salida-múltiple* permiten predecir simultáneamente múltiples horizontes con una única red. Cada una de las neuronas de salida estará especializada en predecir un horizonte distinto. Recientemente se han publicado trabajos en los que se encuentra evidencia del beneficio de construir un modelo con *salida-múltiple*, de manera que éste aprenda y preserve las dependencias entre los valores de la predicción [2]. Los resultados mencionados corresponden al modelado de series temporales estocásticas. En este trabajo compararemos las dos estrategias, *salida-múltiple* y *salida-simple*, para las series temporales caóticas (deterministas) en estudio. Se consideraron redes con m salidas con una separación temporal τ. Se quiere evaluar si el entrenamiento minimizando el error sobre las múltiples salidas tenga un efecto regularizador que pueda otorgarle a la red un mejor error de generalización sobre el horizonte más lejano respecto de las redes de salida simple.

2. Experimentos

Preparación de los Datos Como primer paso se busca obtener una reconstrucción del espacio de fases a partir de considerar coordenadas de retraso. Los correspondientes parámetros τ y m se determinaron con la metodología propuesta en [20]. En todos los casos se utilizaron los primeros 8.000 datos para el conjunto de *entrenamiento*, 2.000 datos para *validación*, y los últimos 10.000 datos se reservaron para estimar el error de cada método (conjunto de *test*).

Determinación de hiperparámetros óptimos Utilizando como referencia el error sobre el conjunto de validación, en el caso de las redes *no-profundas*, se encontró un valor óptimo de cantidad de neuronas alrededor de 200. Este valor resultó apropiado para todas las series temporales, por lo tanto se adoptó ese valor para realizar los experimentos. Por otro lado se consideraron redes neuronales *profundas* de tres capas ocultas y se exploraron arquitecturas con la misma cantidad de neuronas en cada capa. Como punto de partida se eligió la cantidad de neuronas de modo tal que la red tenga la misma cantidad de pesos (parámetros) que la red no-profunda, obteniendo 20 neuronas por capa. Esta configuración presentó menor error sobre el conjunto de validación que configuraciones con más neuronas ocultas por capa con lo cuál se adoptó esta arquitectura para los experimentos. En cuanto a la cantidad de épocas de descenso por gradiente, se utilizó un criterio de *detención temprana* utilizando como referencia el mínimo del error sobre el conjunto de validación.

3. Resultados

Los distintos métodos fueron comparados utilizando el error cuadrático medio (normalizado) sobre el conjunto de test para un rango de horizontes que llega hasta el límite de predecibilidad de cada serie. En cada gráfica de error, además de los resultados con redes neuronales, se presenta como referencia el error obtenido con modelos locales lineales [18]. Se consideraron series sintéticas y una serie real (circuito de Chua, Sec. 3.2).

3.1. Series sintéticas

En el primer caso de estudio se considera la serie de Mackey-Glass [12] integrada numéricamente a partir de la ecuación $\dot{x} = [ax(t-\overline{\tau})]/[1+x^c(t-\overline{\tau})] - bx$ con parámetros $a = 0{,}2$; $b = 0{,}1$; $c = 10$ y $\overline{\tau} = 17$. Para esta serie, el tamaño de ventana óptimo obtenido es $t_w = 30$ y dimensión de embedding $m = 4$ (tiempo de retraso $\tau = 10$) [21].

El resultado más sobresaliente de la Fig. 1 es que la red profunda de *salida-simple* (4-20-20-20-1) es la red con mejor comportamiento en un balance sobre el rango completo horizontes: presenta un error menor o comparable respecto al resto de los métodos evaluados.

En las predicciones a corto plazo las mayores diferencias entre métodos ocurre entre las estrategias *directa* vs. *iterativa* y entre *salida-simple* vs *salida-múltiple*, imponiéndose las primeras sobre las segundas respectivamente. La profundidad de la red no parece ser un elemento clave en la predicción a *corto plazo*. Esto se debe a que la predicción a corto plazo no requiere gran no-linealidad para esta serie.

En el caso *iterativo* se observa un resultado desfavorable para las predicciones a *corto plazo* frente al método *directo*. Esta relación se invierte en un pequeño rango dentro los horizontes de *largo plazo*.

Para el caso de *salida-múltiple* se observa sistemáticamente un peor desempeño independientemente de la profundidad de la red o estrategia *iterativa* o *directa*. Esto sugiere que entrenar estas redes minimizando el error sobre las cuatro neuronas de salida deteriora el desempeño sobre el conjunto de test sobre la neurona correspondiente al horizonte más lejano (que es el que se evalúa en la figura para comparar con los métodos de *salida-simple*).

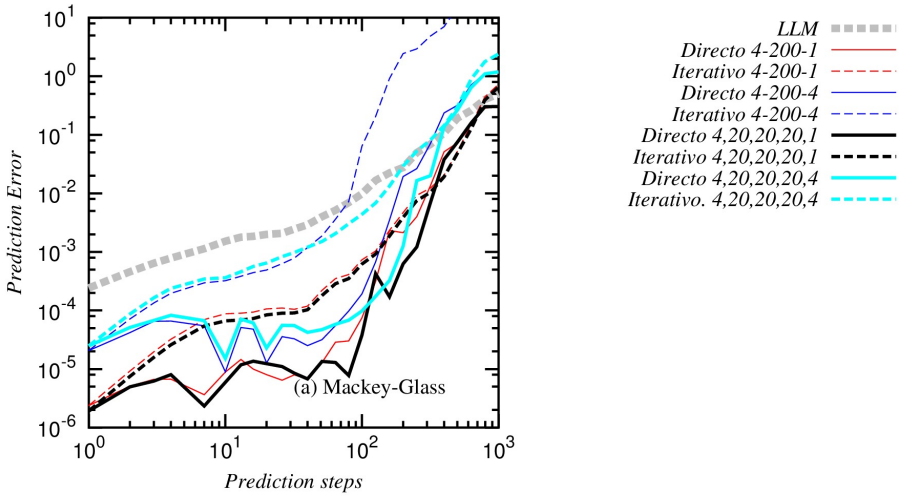

Figura 1. Serie temporal de Mackey-Glass. Error de predicción en función del horizonte H (ambos en escala logarítmica), para los métodos *iterativo* y *directo*, tanto en RN *profundas* como *no-profundas*, tomando como referencia los modelos locales lineales (LLM).

Una diferencia sistemática entre las variantes de los métodos, se encuentra entre los métodos *iterativos* y *directos* (Sec. 1.1), donde se puede observar que, en general, los métodos *directos* dan mejores resultados para esta esta serie. Pero para este método, si se quiere la predicción completa hasta un horizonte H usando como entrada solamente datos de entrenamiento, se deben entrenar H predictores (RN) (Sec. 1.1). Lo cual implica consumir más recursos que con el método *iterativo* que sólo necesita entrenar una única RN para $H=1$.

Se realizaron experimentos equivalentes utilizando las series de Rössler [17] y de Lorenz [11] obteniéndose resultados cualitativamente equivalentes: las redes neuronales profundas de salida simple con el método directo presentan el mejor resultado sobre el rango de horizontes [13].

3.2. Circuito de Chua

En esta sección se muestra el uso de la metodología propuesta para la serie temporal del circuito de Chua. Esta serie temporal corresponde a mediciones de la corriente en el inductor realizadas por Aguirre *et al.* [14], cuyos datos están disponibles en [1]. Para esta serie, el tamaño de ventana óptimo obtenido es $t_w = 81$ y su dimensión $m = 3$ [21], resultando $\tau = 27$ el tiempo de retraso. En la Fig. 2 se muestra la serie temporal del circuito de Chua luego de ser suavizada con con un filtro de Savitzky-Golay.

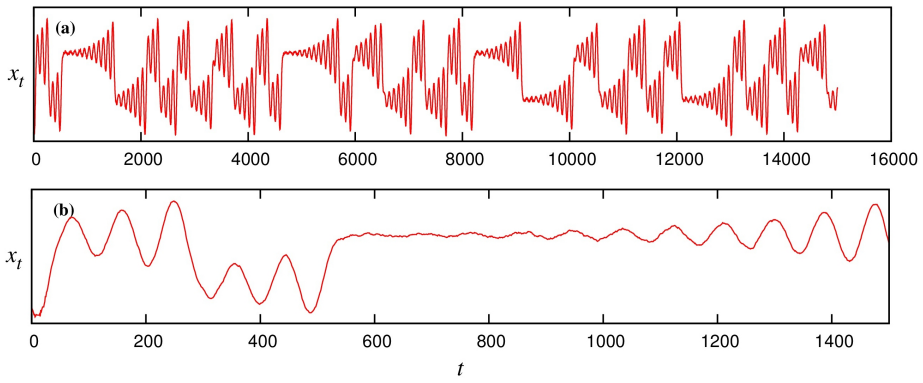

Figura 2. Serie temporal del circuito de Chua. (a) Serie temporal completa. (b) Primeros 1500 datos de la serie temporal, ambas suavizadas con un filtro de Savitzky-Golay.

En la Fig. 3, tenemos las curvas de error vs. horizonte de predicción H. Para esta serie, a diferencia de las anteriores, el mejor desempeño se obtiene con un método *iterativo*. Sin embargo sigue siendo una red profunda la que presenta menor error en todo el rango de horizontes H. Las redes neuronales estudiadas se comportan de manera similar hasta $H = 10$, luego las curvas de error se separan y finalmente alcanzan valores similares a partir de $H = 400$. De los distintos comportamientos vemos que las RN (3-200-N) directas son las primeras en aumentar su error con el horizonte (en $H \approx 40$). En $H \approx 10^2$ diverge el error de las RN (3-20-20-20-N) directas. Las redes de *salida-simple* iteradas son las que alcanzan mayores horizontes manteniendo un bajo error, logrando la red profunda 3-20-20-20-1 iterada un error significativamente menor que el caso no-profundo en este rango. Finalmente las redes iteradas de *salida-múltiple* son las que peor desempeño presentan en las predicciones a largo plazo.

4. Conclusiones

En este trabajo se evaluaron redes neuronales profundas para la predicción de series temporales caóticas. Se comparó su desempeño con el de las redes neu-

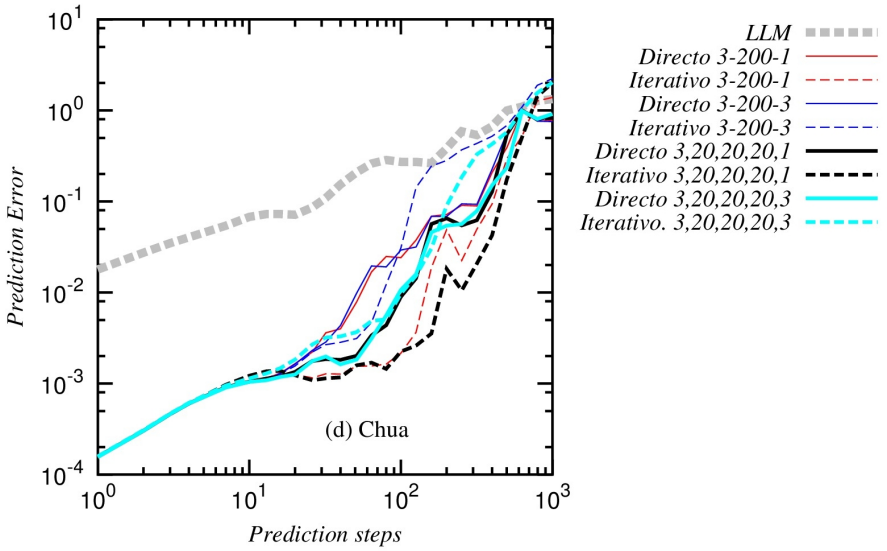

Figura 3. Serie temporal de Circuito de Chua. Error de predicción en función del horizonte H, ambos en escala logarítmica.

ronales convencionales (de una única capa oculta) para realizar predicción en distintos horizontes hasta el límite de predecibilidad de cada serie. Los resultados sobre series temporales sintéticas y reales sugieren que las arquitecturas profundas superan a las redes neuronales convencionales para todo el rango de horizontes.

En cuanto al tiempo de ejecución, al considerar redes profundas con igual cantidad de parámetros que las redes no-profundas, no representó una diferencia considerable, por lo tanto vale considerar el uso de estas arquitecturas para la tarea de predicción. Con respecto a la *salida-múltiple*, no se encontró beneficio frente a un modelo con *salida-simple*.

Finalmente, respecto al desempeño de métodos iterativos vs. directos (discusión inicialmente abordada en [6]), no se pueden sacar mayores conclusiones de los resultados obtenidos ya que para las series sintéticas los métodos directos presentaron menor error y lo contrario ocurrió para la serie real del circuito de Chua.

Como trabajo futuro resta verificar si la diferencia de comportamiento entre las series sintéticas y la serie real se debe a la presencia de ruido en esta última. Para identificar esto, una posibilidad es repetir los experimentos agregando ruido a las series sintéticas. Esto permitiría verificar si el método iterativo en presencia de ruido supera al método directo. Esto implicaría que predecir a un horizonte de largo plazo en forma directa es resolver un problema no-lineal de mayor dificultad y mayor error que el que se genera al amplificar el ruido de la misma serie

temporal al iterar predicciones de corto plazo. Esta conclusión fue la obtenida por [6] utilizando modelos locales polinómicos.

Referencias

[1] MACSIN. Research Group. Chua's circuit meassured data

[2] Ben Taieb, S., Bontempi, G., Atiya, A.F., Sorjamaa, A.: A review and comparison of strategies for multi-step ahead time series forecasting based on the nn5 forecasting competition. Expert Syst. Appl. 39(8), 7067–7083 (Jun 2012)

[3] Bengio, Y.: Learning deep architectures for ai. Foundations and Trends in Machine Learning 2(1), 1–127 (2009)

[4] Cook, J., Sutskever, I., Mnih, A., Hinton, G.E.: Visualizing similarity data with a mixture of maps. JMLR - Proceedings Track 2, 67–74 (2007)

[5] Cybenko, G.: Approximation by superpositions of a sigmoidal function. Mathematics of Control, Signals, and Systems 2, 303–314 (1989)

[6] Farmer, J.D., Sidorowich, J.J.: Exploiting chaos to predict the future and reduce noise. Evolution, learning, and cognition p. 277 (1988)

[7] Hinton, G.E., Osindero, S., Teh, Y.W.: A fast learning algorithm for deep belief nets. Neural Computation 18(7), 1527–1554 (2006)

[8] Hinton, G.E., Osindero, S., Teh, Y.W.: A fast learning algorithm for deep belief nets. Neural Computation 18(7), 1527–1554 (2006)

[9] Hinton, G.E.e.a.: Improving neural networks by preventing co-adaptation of feature detectors arxiv preprint arxiv:1207.0580 (2012) (2012)

[10] Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. pp. 1106–1114 (2012)

[11] Lorenz, E. N.: Deterministic nonperiodic flow. J. Atmos. Sci. 20, 130–141 (1963)

[12] Mackey, M., Glass, L.: Oscillation and chaos in physiological control systems. Science 197(4300), 287–289 (1977)

[13] Maino, D.G.: Tesina de Licenciatura en Ciencias de la Computación, Predicción de Sistemas Dinámicos con Redes Neuronales Profundas (2013)

[14] Mendes, L., Aguirre, G., Rodrigues, E.: Nonlinear identification and cluster analysis of chaotic attractors from a real implementation of Chua's circuit. Int J. Bifurcat. Chaos 7(6), 1411–1423 (1997)

[15] Rahman Mohamed, A., Hinton, G.E., Penn, G.: Understanding how deep belief networks perform acoustic modelling. In: ICASSP. pp. 4273–4276. IEEE (2012)

[16] Rahman Mohamed, A., Sainath, T.N., Dahl, G.E., Ramabhadran, B., Hinton, G.E., Picheny, M.A.: Deep belief networks using discriminative features for phone recognition. In: ICASSP. pp. 5060–5063. IEEE (2011)

[17] Rössler, O.: An equation for continuous chaos. Phys. Lett. A 57(5), 397–398 (1976)

[18] Sauer, T.: Time series prediction by using delay coordinate embedding. In: Weigend, A.S., Gershenfeld, N.A. (eds.) Time Series Prediction: forecasting the future and understanding the past, pp. 175–193. Addison Wesley, Harlow, UK (1993)

[19] Takens, F.: Detecting strange attractors in turbulence. In: Rand, D., Young, L.S. (eds.) Dynamical Systems and Turbulence, Warwick 1980, Lecture Notes in Mathematics, vol. 898, pp. 366–381. Springer Berlin / Heidelberg (1981)

[20] Uzal, L.C., Grinblat, G.L., Verdes, P.F.: Optimal reconstruction of dynamical systems: A noise amplification approach. Physical Review E - Statistical, Nonlinear, and Soft Matter Physics 84(1) (2011)

[21] Uzal, L.C.: Tesis de doctorado en Física, Técnicas de reconstrucción de sistemas dinámicos (2012)