Universidad Nacional de Cuyo
Facultad de Ciencias Exactas y Naturales

Predicción de Precipitación en Mendoza y Buenos Aires Mediante Técnicas de Aprendizaje de Máquinas

por
Mandrilli, Paula Alejandra

Director: Dr. Santos, Jorge Rubén

Co-Director: Dr. Monge, David A.

Informe del Seminario de Investigación,
presentado como requisito parcial para la obtención del título de
Licenciada en Ciencias Básicas, Orientación Física

Mendoza, Marzo de 2018

Resumen

Estudiamos dos modelos de Aprendizaje de Máquinas (AM) para evaluar sus desempeños en la predicción de ocurrencia de precipitación en la ciudad de Mendoza y en la Ciudad Autónoma de Buenos Aires (CABA) en invierno y en verano.

En primer lugar, analizamos las características espacio-temporal de los datos meteorológicos (precipitación acumulada en tres horas y estado de la atmósfera cada seis horas) correspondientes a un período de quince años (2000 a 2014 inclusive), buscando aquellos con mayor influencia en la formación de la precipitación, facilitando la identificación del patrón que determina la precipitación mediante los modelos Regresión Logística (RL) y Red Neuronal Artificial (RNA). En segundo lugar, aprendimos los modelos mediante técnicas de AM y los evaluamos según métricas de tasa de aciertos y falsos negativos y de tiempos de ejecución al comparar las predicciones con las estimaciones de precipitación mediante información satelital. Luego, contrastamos la capacidad de predicción de dichos modelos con la del modelo Weather Research and Forecasting Model (WRF) aplicados al año 2011.

Llevamos a cabo un análisis de distintas bases de datos incluyendo datos de reanálisis, datos de precipitación y datos del fenómeno El niño. Dichos análisis incluyen un análisis de wavelets y un análisis de correlación entre las variables de reanálisis. Finalmente generamos una lista de 12 atributos que se tomaron como entrada en el aprendizaje de los modelos propuestos.

Por medio de técnicas de balance de clases, selección de atributos, selección de modelos y validación cruzada de k iteraciones descubrimos que para el problema planteado, las técnicas y las configuraciones elegidas, los modelos de AM aplicados a los datos de Mendoza resultaron tener el mejor desempeño y en todos los casos de pares ciudad-estación la ejecución de los modelos aprendidos fue de cinco o cuatro órdenes de magnitud más rápida que la de WRF.

Adicionalmente, debido a que los modelos de AM seleccionados para predecir a partir de los datos de CABA obtuvieron el máximo porcentaje de tasa de falsos negativos, presentaron un bajo rendimiento. Y por otro lado, si bien WRF tuvo buen porcentaje de aciertos en el caso de ser aplicado a CABA, también resultó tener una alta tasa de falsos negativos. Por lo tanto, no consideramos a los modelos de AM seleccionados ni a WRF como buenos predictores para las configuraciones seleccionadas, ya que no podemos utilizar sus resultados para propiciar a la sociedad de una alerta confiable.

De esta manera, demostramos la aplicabilidad operacional de AM y que los modelos aprendidos pueden considerarse mejores que el modelo de simulación más utilizado mundialmente en la actualidad para resolver dicho problema.

Palabras clave: Predicción de Precipitación, Aprendizaje de Máquinas, Regresión Logística, Redes Neuronales Artificiales.

Agradecimientos

Primero que nada, agradezco a mi familia. A mis padres Ángela Esperanza Escudero y Roberto Luis Mandrilli por brindarme todo su amor, esfuerzo y valores. A mis hermanos Alex-Dril, maestros y compañeros incondicionales: Carolina A., Pablo A., David A. y María Alejandra. A mis abuelos, tíos y primos e hijos por el apoyo y sus enseñanzas inolvidables. A los compañeros de mis hermanos. A mis sobrinos Quiquín, Luca, Mathiu, Tomi y Mateito por la pureza, asombro y alegría de niños y el sentimiento inexplicable transmitido. A todos y cada uno de ellos, por hacerme feliz y ser parte esencial de este logro.

A mis amigos del alma Andre C., Barba H., Belu P., Betsi S., Caro R., Celi M., Chanta F., Chasa B., Franklin P., Fran S., Germaioni G., Ju S., Lu S., Michu R., Nanito A., Rochi C., Romi B., Titi V., Vi N., Kili, E.T. y Tachín por compartir momentos imprescindibles para mi felicidad. Y a sus familiares que me acogieron tratándome como putativa. Aportaron durante mi carrera de grado, haciendo compañía con mates de por medio, cuidados, materiales, consejos y muchas risas.

A mis directores Carlos A. Katania (AKA Harpo), David A. Monge y J. Rubén Santos. A Rubén por responder y aceptar de inmediato aplicar a la Beca EVC-CIN 2015, y compartir la idea de realizar un trabajo interdisciplinario. A Carlos y David por sumarse a la idea. A los tres por la propuesta que llevamos a cabo en estos dos años y medio, hacerme partícipe de discusiones y exigencias productivas y otorgarme libertad sobre decisiones. Por la gran dedicación, sus esforzadas explicaciones y la incesante y vasta tolerancia. No sólo me instruyeron en lo que respecta a lo académico, sino que también contribuyeron en mi formación personal. Espero haya sido tan grato para ellos como para mí trabajar juntos.

A la comunidad de la FCEN-UNCuyo. Especialmente a todos mis profesores por su disponibilidad y recibimiento cada vez que se los solicité. Particularmente a mis profesores Adriana Fornés, Bernardo Gonzalez Riga, Carlos Ruestes, Cecilia Pirrone, Eduardo Bringa, Enrique Miranda, Laura Remaggi, Marta Rey Tudela, Pablo Kaluza, Ricardo Leiva y Sebastián Simondi, a Claudia Sara y al Decano Manuel Tovar, por charlas inolvidablemente enriquecedoras. A Eduardo Bringa por su preocupación y ocupación por los alumnos de Física en relación a cualquier incumbencia de éstos; su disponibilidad en cualquier día y horario, su dedicación, su incentivo y sus consejos. A Pablo Cremades y Jesús Giunta por la disposición y ayuda en cuestiones computacionales. A Adriano Muñoz por su atención y ocupación en pos de la comodidad del alumnado. A Adriana Fornés por su inmenso cariño, su hospitalidad, su lección, su explosiva alegría y enorme sonrisa en nuestro último encuentro inesperado, y por su legado lleno de voluntad. A conocidos de la institución. Todos fueron fundamentales para mi permanencia a gusto en esta casa de estudios.

Agradezco a la Dra. Elina Pacini, al Dr. Sebastián Simondi y a la Lic. Yanina González por aceptar ser parte del jurado de la presente tesis.

A la comunidad de la UNCuyo por darme la oportunidad de vivenciar tan lindas y variadas experiencias, y a aquellos que conocí en las mismas.

A todos por hacer de mí la persona que soy, convirtiéndose en causantes de este trabajo.

Índice general

Índice de figuras

Índice de cuadros

Capítulo 1

Introducción

1.1. Motivación

La predicción del estado de variables atmosféricas es un desafío que la humanidad ha venido enfrentando desde sus orígenes. Recién avanzado el siglo XX, a partir del artículo científico 'El problema de la predicción del tiempo desde el punto de vista de la Mecánica y la Física' publicado en 1904 por el Físico noruego Vilhelm F. K. Bjerkne [2], comenzó el modelado meteorológico, el del clima, de la relación de este último con las distintas partes de la superficie de la tierra (mares, lagos, océanos, montañas) y del efecto de la actividad antropológica (contaminación, deforestación, etc.). Esta ciencia en vanguardia, no tardó en convertirse en un problema científico multidisciplinario.

Particularmente, a causa de la compleja interacción entre las diferentes escalas espaciales y temporales propias de la dinámica de flujo energético atmosférico, el problema de predecir dónde y cuándo ocurrirá algún fenómeno meteorológico, así como su magnitud, es actualmente un área de intenso estudio.

Uno de los principales problemas de los modelos meteorológicos es la dificultad en la predicción de la cantidad de precipitación con precisión, cuyas consecuencias afectan directa e indirectamente a toda la sociedad debido a pérdidas de vida y daños económicos causados por intensas precipitaciones. Su resolución posibilitaría la mitigación de los daños producidos en la región afectada. [3]

Para intentar resolver dicho inconveniente el enfoque más frecuente es la utilización de modelos llamados Predicción Numérica del Tiempo (NWP, por sus siglas en inglés de Numerical Weather Prediction) [4], que contemplan las leyes físicas que describen la dinámica de flujo atmosférico y los esquemas de parametrización que representan procesos físicos en la subgrilla de cálculo.

Entre los modelos NWP más utilizados hoy en día por la comunidad científica internacional y centros de estudios meteorológicos de distintas partes del mundo, podemos citar WRF (por sus siglas en inglés de Weather Research and Forecasting [5]), desarrollado por instituciones de investigación de diferentes países. Entre otros, podemos mencionar a GEM [6] (por sus siglas en inglés de Global Environmental Multiscale), creado originariamente para uso en Canadá, al ECMWF [7] (por sus siglas en inglés de European Centre for Medium-Range Weather Forecasts) y al más utilizado por Brasil, llamado ETA [8] (cuyo nombre proviene del sistema de coordenadas verticales del modelo). Todos ellos modelan el estado y la evolución de los procesos atmosféricos, diferenciándose entre sí en el diseño numérico de la parametrización de los procesos que implementan.

El uso operativo de los modelos NWP en los últimos años se ha visto favorecido por el incremento de la capacidad computacional y el conocimiento más detallado de los procesos atmosféricos en distintas escalas. Sin embargo, utilizando una computadora personal de escritorio para el pronóstico regional, su costo computacional sigue siendo elevado; por ejemplo, para pronosticar a 6 días se requieren 6 h de cómputo con un procesador de 1,4 GH con 16 núcleos, en una zona de resolución espacial de 10 km que sólo delimite la parte norte-centro de la provincia de Mendoza. Además, el éxito de los mismos depende no solamente de los procesos físicos que modelan (frentes fríos, tormentas, viento zonda, etc.), sino también de la zona geográfica de estudio (topografía) [3, 9, 10]. Estos aspectos desfavorables constituyen una gran limitación para el funcionamiento adecuado de un sistema de alerta a corto plazo.

Por lo tanto, sin perder de vista la complejidad de los procesos atmosféricos, con el motivo de predecir características particulares del estado atmosférico (presión, temperatura máxima o mínima, humedad, etc.), vale la pena investigar modelos más sencillos. Se requiere predecir no sólo con un modelo más simple (i.e. con menor costo computacional) con el cual se logren tiempos menores de predicción (i.e. con mayor velocidad de cómputo), sino también que su salida se obtenga con suficiente antelación al evento real para tomar medidas pertinentes. Fue así que desde hace algunas décadas comenzó a utilizarse Inteligencia Artificial (IA) en el campo de la Meteorología [11].

Cabe destacar que en 1956 se pronuncia en público por vez primera el término IA, no obstante fue en el año 1943 cuando W. S. McCulloch y W. Pitts presentan el primer trabajo en dicha área: Redes Neuronales Artificiales [12]. Así, a finales de la década del 50, en búsqueda de su desarrollo, nace la rama Aprendizaje de Máquinas (AM). Arthur L. Samuel, considerado el padre del AM, en 1959 lo definió como el «Campo de estudio que da a las computadoras la habilidad de aprender sin estar explícitamente programadas» [13]. Desde entonces, este campo no ha dejado de avanzar y aplicarse a diversas ciencias.

Si bien desde finales de la década del 90 el AM aplicado a la Meteorología primero se usó

para la predicción de distintas variables atmosféricas [14] [15], ahora también es aprovechado para la predicción de fenómenos atmosféricos, como la precipitación. En esta línea de investigación, investigadores de Canadá [16], Gracia [17], India [18], Australia [19] han hecho uso de esta herramienta, algunos por medio de Redes Neuronales Artificiales. Es de especial interés mencionar que en Argentina se ha atacado la variabilidad de precipitaciones a escalas climáticas [20], pero hasta el momento no se evidencian trabajos a nivel nacional ni regional a escalas horarias donde se aplique AM para predecir precipitación.

Así, por el problema planteado y la limitación expuesta, máxime contando con otras herramientas para tratarlos, es que surge la hipótesis del presente trabajo.

1.2. Hipótesis

Para superar la limitación de la predicción espacial y temporal de la ocurrencia de precipitación, proponemos una estrategia basada en la aplicabilidad de técnicas de AM para dicho propósito para la ciudad de Mendoza (Mza) y la Ciudad Autónoma de Buenos Aires (CABA) ubicadas en la zona central de la República Argentina en regiones climatológicamente diferenciadas.

En este contexto, es posible el uso operativo de AM por medio de Regresión Logística (RL) y Red Neuronal Artificial (RNA), de forma autónoma (i.e. sin intervención humana), capaces de pronosticar la ocurrencia de precipitación a partir de datos históricos atmosféricos de Mza y de CABA.

1.3. Objetivos

El objetivo principal del estudio es evaluar el desempeño de los modelos RL y RNA en su capacidad para predecir correctamente la ocurrencia de precipitación en Mza y CABA con condiciones climatológicas claramente diferenciadas, comparándolos con el modelo de simulación atmosférico más utilizado (WRF).

De dicho objetivo se desprende el interés por contribuir con la lista de atributos que más influyan en la iniciación de convecciones profundas que originan precipitación y la búsqueda de una metodología de predicción de precipitación como método novedoso para la zona central argentina logrando menores tiempos de cómputo a los usuales al predecir correctamente ocurrencia de precipitación en las ciudades mencionadas.

1.4. Organización del Trabajo

En el siguiente capítulo, Capítulo 2, nos adentramos en la manera en que predice el modelo de simulación NWP-WRF. Primero contamos de qué consta y de qué se lo provee para el cálculo, y luego restringimos dicha información a nuestro caso.

En el Capítulo 3 explicamos de qué se trata AM y cómo se aplica en este trabajo. En primer lugar, damos a conocer sus aspectos generales. Después planteamos el problema de aprendizaje que nos incumbe. Luego, en la Sección 3.3, damos a entender el proceso de aprendizaje y la técnica que utilizamos en su aplicación. Y en la última sección describimos en detalle los modelos RL y RNA y métodos a aplicar en el aprendizaje de los mismos.

Los dos capítulos posteriores constan de los estudios realizados. En el Capítulo 4, Estudio I, exploramos los datos meteorológicos para extraer aquellos de interés y conveniencia para entrenar y evaluar los modelos aplicados en el Estudio II, Capítulo 5. En el primer capítulo mencionado analizamos los datos teniendo en cuenta la experiencia de expertos y los resultados obtenidos mediante herramientas computacionales que exponemos. En el segundo, mediante técnicas de AM, evaluamos el desempeño de los modelos seleccionados en su capacidad de predicción de ocurrencia de precipitación en invierno y en verano en Mza y CABA.

En el último capítulo manifestamos las conclusiones generales del presente trabajo, limitaciones advertidas en la práctica y posibles líneas de investigación futuras que se desprenden de los estudios.

Capítulo 2

Modelo WRF

La previsión de eventos meteorológicos extremos, tales como huracanes, tornados, tormentas asociadas a grandes cantidades de precipitación, sequías, entre otros, es de vital importancia en la elaboración de estrategias en el proceso de mitigación de los daños que las sociedades deben enfrentar. Los primeros intentos sobre métodos matemáticos manuales que relacionan el medio ambiente con los seres vivos comenzaron en 1920, pero no fue hasta los años 50, con la aparición de las primeras computadoras de propósito general, que comenzó a utilizarse el cálculo numérico para conocer el estado futuro de la atmósfera. Por ejemplo en los 70 y 80 se realizaron los primeros pronósticos de trayectorias de ciclones tropicales y calidad de aire, y actualmente se realiza el pronóstico de diversos procesos atmosféricos de forma global y regional abarcando desde horas (muy corto plazo) hasta días (corto-mediano plazo), semanas (mediano-largo plazo) y meses (escala climática).

Si bien se ha mejorado el pronóstico de eventos meteorológicos tanto a nivel regional como global, persisten numerosos desafíos que los modelos numéricos aún deben mejorar, entre los cuales podemos mencionar el pronóstico de la cantidad de precipitación que caerá en un determinado lugar en un intervalo de tiempo dado. Este problema es conocido como QPF (de sus siglas en inglés de Quantitative Precipitation Forecast), el cual motiva una intensa área de estudio que involucra un trabajo multidisciplinar de la comunidad científica internacional.

Para dicho propósito se utilizan distintos modelos numéricos que incluyen el estado de arte en la representación de los procesos físicos y químicos. Entre los más populares está el modelo WRF (de sus siglas en inglés de Weather Research and Forecasting Model [5]), el cual es de libre acceso y cuenta con una gran colaboración internacional en la mejora continua de la representación de los procesos atmosféricos.

En la presente tesis utilizamos WRF para la previsión de la ocurrencia de precipitación acumulada a tres horas durante todo el año 2011, en Mza y CABA, para la posterior evaluación de los

modelos de Aprendizaje de Máquinas que se detallan en el capítulo siguiente.

En este capítulo entonces, describimos el modelo WRF, su estructura y la aplicación particular al caso de estudio.

2.1. Ecuaciones Fundamentales

El modelo WRF incluye la representación numérica de ecuaciones fundamentales entre las que se incluyen la dinámica y termodinámica de la atmósfera mediante la utilización de ecuaciones fundamentales de estado, masa, energía y momento.

A dichas ecuaciones fundamentales, omitiendo la fuerza de Coriolis por simplicidad de notación, las expresamos a continuación en coordenadas cartesianas.

Ecuación de estado de la parcela de aire:

$$p = \rho R_d T$$

siendo p la presión del aire; ρ la densidad del aire; $R_d = \frac{2}{7} c_p$ la constante de gas de aire seco donde $c_p = 1004.5\ JK^{-1}kg^{-1}$ es el calor específico a presión constante del aire seco; y T la temperatura absoluta del aire.

Conservación de la masa de la parcela de aire:

$$\frac{\partial \rho}{\partial t} + \frac{\partial U}{\partial x} + \frac{\partial V}{\partial y} + \frac{\partial W}{\partial z} = 0$$

siendo $U = \rho u$, $V = \rho v$ y $W = \rho w$ donde u, v, w son las componentes de la velocidad en las direcciones x, y, z respectivamente.

Conservación del momento de la parcela de aire:

$$\frac{\partial U}{\partial t} + c_p \Theta \frac{\partial \pi}{\partial x} = -\frac{\partial (Uu)}{\partial x} - \frac{\partial (Vu)}{\partial y} - \frac{\partial (Wu)}{\partial z} + F_x$$

$$\frac{\partial V}{\partial t} + c_p \Theta \frac{\partial \pi}{\partial y} = -\frac{\partial (Uv)}{\partial x} - \frac{\partial (Vv)}{\partial y} - \frac{\partial (Wv)}{\partial z} + F_y$$

$$\frac{\partial W}{\partial t} + c_p \Theta \frac{\partial \pi}{\partial z} + g\rho = -\frac{\partial (Uw)}{\partial x} - \frac{\partial (Vw)}{\partial y} - \frac{\partial (Ww)}{\partial z} + F_z$$

siendo $\Theta = \rho\theta$ donde θ es la temperatura potencial; Π la función Exner $\left(\frac{p}{p_0}\right)^{\frac{R_d}{cp}}$; F_x, F_y y F_z las componentes de la fuerza de fricción.

Conservación de energía de la parcela de aire:

$$\frac{\partial \Theta}{\partial t} + \frac{\partial (U\theta)}{\partial x} + \frac{\partial (V\theta)}{\partial y} + \frac{\partial (W\theta)}{\partial z} = \rho Q$$

Cabe mencionar que este sistema de ecuaciones debe ser resuelto numéricamente en una determinada grilla de cálculo que en este caso es una Arakawa escalonada tipo C, donde la componente norte-sur (oeste-este) de la velocidad está ubicada en los lados norte y sur (oeste y este) de la red, mientras que la temperatura está ubicada en el centro de red representada por la letra θ, como mostramos en la Figura 2.1.1.

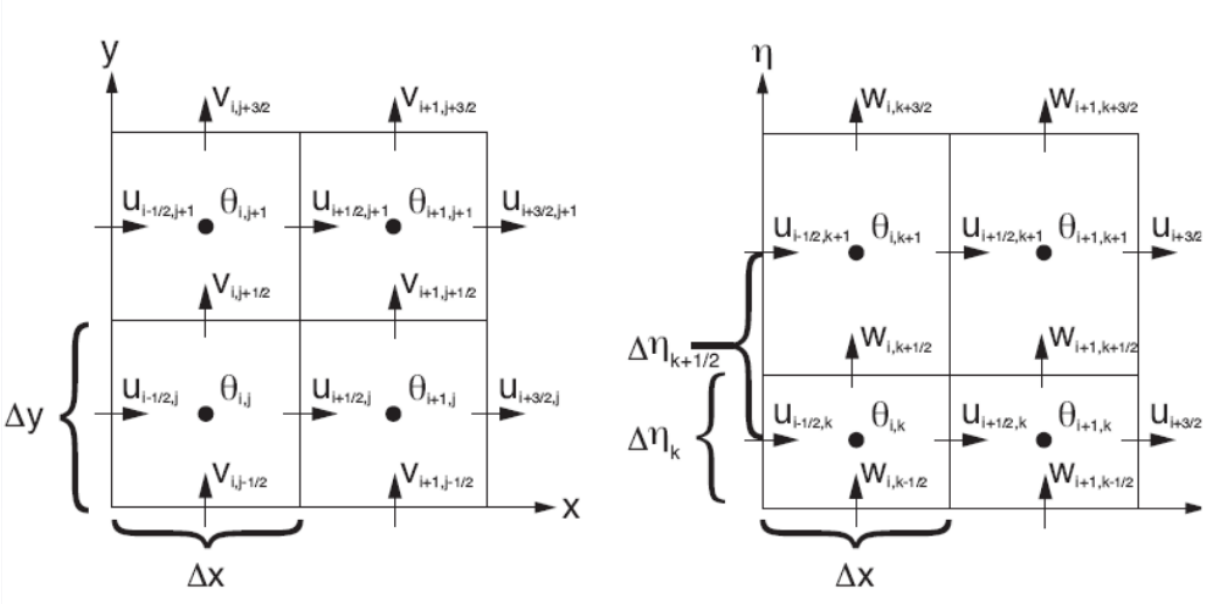


Figura 2.1.1: Estructura de la grilla de cálculo horizontal y vertical de Arakawa Escalonada tipo C.

La estructura vertical de la grilla es resuelta mediante el uso de la coordenada de masa vertical η, la cual se caracteriza por seguir la estructura de la topografía. Dicha estructura se encuentra representada por la Figura 2.1.2. Esta coordenada está definida como

$$\eta = \frac{p_h - p_{ht}}{\mu}$$

siendo $\mu = p_{hs} - p_{ht}$ donde p_h, p_{ht} y p_{hs} representan las componentes de la presión hidrostática, a lo largo de la superficie y en límites superiores respectivamente. Así $\mu(x,y)$ representa la masa por unidad de área en la columna del dominio del modelo en (x,y), y η la coordenada vertical hidrostática con seguimiento del terreno.

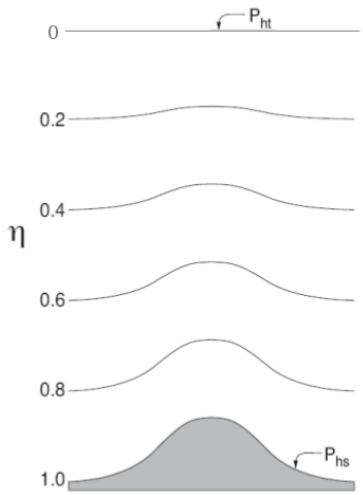


Figura 2.1.2: Interpretación de la coordenada η. Notar que p_{ht} es constante.

Como podemos visualizar, el espaciamiento elegido entre cada punto de la grilla de cálculo dependerá de la estructura del fenómeno atmosférico que estemos estudiando. No obstante habrán otros procesos en una menor escala, los cuales no estarán "resueltos" por la grilla, pero la presencia de los mismos debe ser tenida en cuenta para la correcta representación numérica de los intercambios de energía y momento entre las distintas escalas. El efecto de estos procesos en una subgrilla se encuentra incluido en las llamadas parametrizaciones, la cuales detallamos en la siguiente sección.

2.1.1. Parametrizaciones

Los procesos físicos que son parametrizados en la subgrilla de cálculo los especificamos en la Figura 2.1.3 . Y en forma conjunta se identifican las distintas interacciones entre los mismos. La microfísica corresponde a la representación de los distintos hidrometeoros presentes en una nube, tales como gotas de nube, lluvia, nieve y granizo. Estos esquemas no sólo proveen la cantidad por unidad de volumen de los hidrometeoros, sino también la distribución por tamaño. La parametrización de cúmulus representa las nubes y/o tormentas que ocurren en un tamaño menor que el de la grilla de cálculo, los cuales no son representados o resueltos directamente, pero su efecto de transporte de momento, masa y energía debe ser tenido en cuenta. Los procesos radiativos incluyen los ondas largas (OL) y cortas (OC) que resultan de la interacción con las ondas electromagnéticas de distintas longitudes de ondas como resultado de la interacción de la atmósfera con el Sol, las nubes y la superficie del planeta.

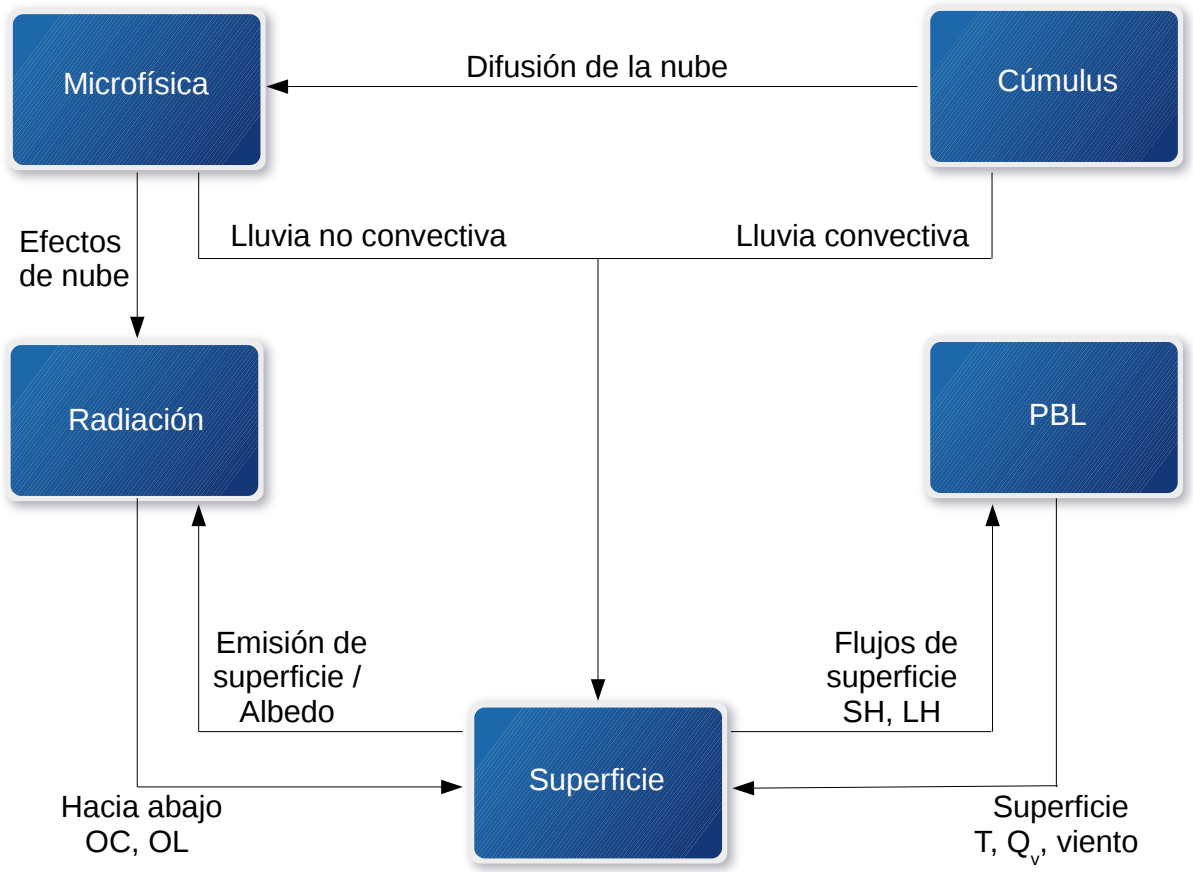


Figura 2.1.3: Esquema de los procesos físicos que son parametrizados en la subgrilla de cálculo, y sus interacciones representadas por flechas (adaptación de la Presentación de Laura Bianco en el marco del curso [1]). PBL hace referencia a la capa límite, OC y OL a las ondas corta y larga respectivamente, SH y LH al calor sensible y al calor latente respectivamente, T a la temperatura y Q_v al calor de vapor.

Los procesos ocurridos en la capa límite (Planetary Boundary Layer -PBL-) tienen en cuenta el efecto del transporte turbulento del calor sensible en la superficie (Surface Turbulent Sensible Heat -SH-) y el transporte de momento y energía entre la superficies de la tierra y las capas mas bajas de la atmósfera (Surface Layer -SL-).

Los procesos que ocurren en la superficie, tienen en cuenta el estado y la evolución del mismo en lo que respecta a la humedad y las característica propias de suelo.

2.1.2. Condiciones Iniciales y de Contorno

Para el diagnóstico y pronóstico del estado de la atmósfera se deben proveer las condiciones iniciales que corresponden al estado de la atmósfera (vientos, humedad, temperatura y presión) al momento de comenzar la integración del modelo y las condiciones de contorno correspondientes, por tratarse de un modelo de área limitada. Estas condiciones iniciales y de contorno provienen generalmente de un modelo global (GFS o ECMWF) o de los archivos NCEP-FNL

(Final Operational Global Tropospheric Analysis) generados por el NCEP (National Center for Environmental Prediction) [21].

2.2. Dominio de Cálculo

Dado que el modelo WRF es un modelo multiescala, el cual tiene la capacidad de simular distintas escalas espacio-temporales de la atmósfera, se debe recurrir a grillas de cálculo con distinto espaciamiento. Para ello se utiliza el proceso de anidado, donde se configuran dominios a distinto espaciamiento de grilla, siendo el dominio mayor el dominio padre mientras el inmediato inferior el hijo, como representamos en la Figura 2.2.1. La superposición de grillas no está permitida y ninguna grilla puede tener más de un dominio padre.

Además, existen procesos de retroalimentación de un solo sentido o de doble sentido.

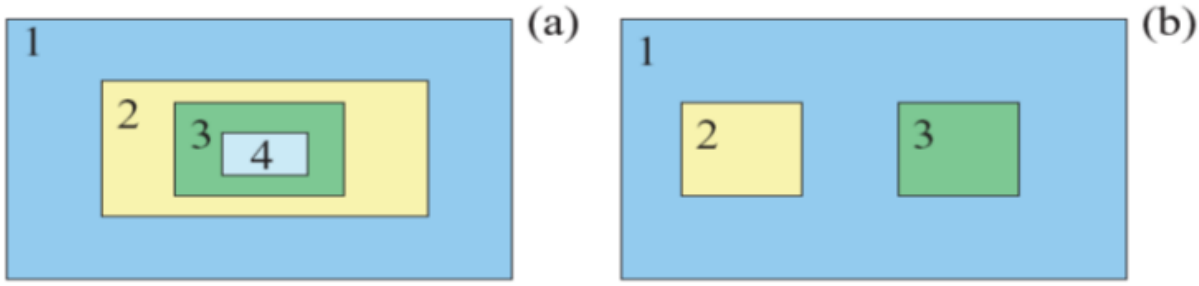


Figura 2.2.1: Ejemplificación de tipos de anidados. A la derecha (a), los llamados Nidos Telescópicos y a la izquierda (b) nidos al mismo nivel respecto a la grilla padre.

2.3. Aplicación Particular

Una vez que explicamos las características generales del modelo WRF, estamos en condiciones de presentar el diseño de dominio de cálculo y las parametrizaciones particulares que utilizamos para aplicar dicho modelo como parámetro de comparación al momento de la evaluación de los modelos de AM en el Estudio II (Capítulo 5).

2.3.1. Consideraciones

La región de cálculo considerada para el modelo WRF fueron dos dominios anidados como se representa en la Figura 2.3.1. El dominio padre $d01$ de 30 km incluye la zona sur de América del Sur y el dominio $d02$ de 10 km incluye la zona central de Argentina, donde se especifican las ciudades que serán de interés para nuestra evaluación, Mza y CABA. En cada dominio se utilizaron 40 niveles verticales.

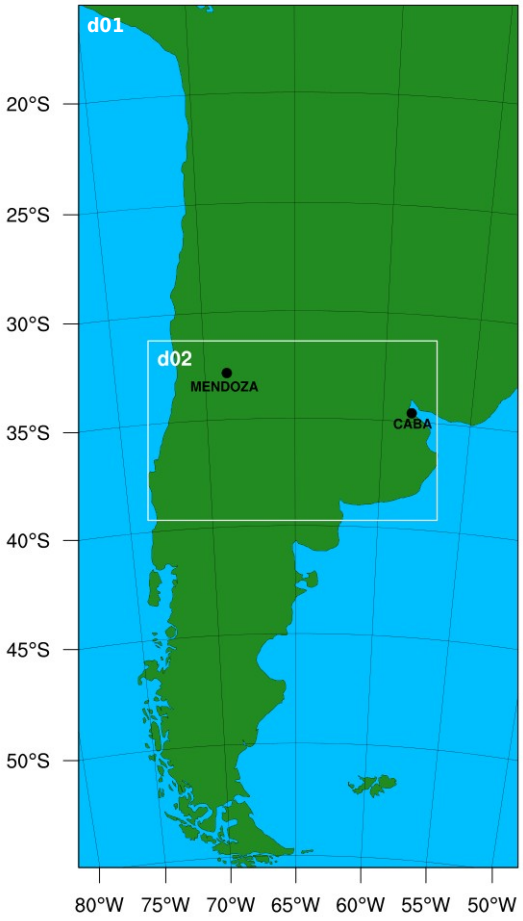


Figura 2.3.1: Diseño del dominio de cálculo considerado para WRF. Dominio $d01$ de 30 km anidado con el dominio $d02$ de 10 km indicado en blanco.

Para más detalles, en la Cuadro 2.1 especificamos las parametrizaciones consideradas en cada dominio en el procesos de integración del modelo de simulación utilizado en nuestro trabajo.

Cuadro 2.1: Esquemas físicos del proceso de integración considerado al utilizar WRF.

Esquema físico	Dominios	
	*d*01	*d*02
Microfísica	WRF – SingleMoment-6	WRF – SingleMoment-6
Capa límite planetaria	Yonsei University PBL	Yonsei University PBL
Procesos de radiación de onda larga	Modelo Onda Larga de Transferencia de Radiación Rápida	Modelo Onda Larga de Transferencia de Radiación Rápida
Procesos de radiación de onda corta	MM5 Dudhia Onda Corta	MM5 Dudhia Onda Corta
Modelo de superficie	NOAH	NOAH
Procesos de difusión térmica	Esquema Monin-Obukov	Esquema Monin-Obukov
Parametrización de cúmulo	Esquema NewGrell	Esquema NewGrell

Hay que tener en cuenta que las condiciones iniciales y de contorno provienen de los archivos NCEP-FNL, con datos cada 6 horas con espaciamiento de grilla de 1° x 1°.

Es importante mencionar que lo integramos durante el año 2011 a fin de obtener la precipitación acumulada horaria durante los 365 días con inicializaciones de 24 horas de tiempo de spin-up, el cual se entiende como tiempo necesario para que el modelo se estabilice debido a las incertezas de las condiciones iniciales. Y luego lo comparamos con estimaciones de precipitación (calculadas mediante el método Climate Prediction Center morphing method–CMORPH– [22]) extraídas del Centro de Predicción Climática (CPC, por sus siglas en inglés) (`http://www.cpc.ncep.noaa.gov`).

Respecto a estos datos, profundizaremos sobre sus archivos en el Estudio I (Capítulo 4).

Los resultados de las simulaciones se presentan y discuten en la Sección 5.3.

Capítulo 3

Aprendizaje de Máquinas

En el capítulo anterior exploramos el rasgo meteorológico del problema tratado durante la investigación. Dimos a conocer la descripción del modelo WRF y la configuración utilizada en nuestro trabajo. En cambio, en el presente capítulo exploraremos los modelos que serán evaluados en el Estudio II según este modelo tomado como parámetro.

Teniendo en cuenta que la ocurrencia de patrones atmosféricos similares conducen a eventos meteorológicos también similares, en este capítulo veremos cómo poder reconocer patrones mediante el Aprendizaje de Máquinas (AM). A diferencia de los modelos NWP, esta tecnología cuenta con la ventaja de aprender modelos de manera automática sin tener que codificarlos en función de un conjunto de datos provistos, lo cual deriva en la posibilidad de reducir considerablemente el gasto computacional al utilizarla.

El camino elegido para que una máquina aprenda algo en particular tiene sus fundamentos en su propia aplicación, cobrando sentido la información implícita en los datos que se poseen [23]. Sin embrago, antes de introducirnos en la aplicación, es necesario comprender las características generales de AM y el funcionamiento de los métodos utilizados.

En este capítulo tratamos los fundamentos teóricos de AM. En la Sección 3.1 damos a conocer su definición y sus cualidades, y el tipo de aprendizaje que imponemos en los experimentos. En la Sección 3.2 especificamos el problema que queremos resolver en este trabajo definido en el marco de AM. En la Sección 3.3, explicamos el proceso de aprendizaje que atraviesa la máquina. Y en la Sección 3.4, describimos los dos modelos de AM que estudiamos.

3.1. Concepto

AM es la rama de las Ciencias de la Computación donde las máquinas aprenden sin intervención humana, es decir, el «Campo de estudio que da a las computadoras la habilidad de

aprender sin estar explícitamente programadas» [13].

Una definición más precisa es la que propuso Tom M. Mitchell y que describe el proceso de aprendizaje de la siguiente forma: «Se dice que un programa computacional aprende de una experiencia E respecto a alguna tarea T y alguna medida de desempeño P, si su desempeño en T, medido por P, mejora con la experiencia E» [24].

En otras palabras podemos decir que un programa aprende si su desempeño en la realización de una tarea, mejora con la experiencia en dicha tarea. Lo interesante de este enfoque es que permite aprender o derivar programas o (como en el caso de esta tesis) *modelos predictivos* de forma automática sin necesidad de que una persona deba especificar explícitamente el comportamiento requerido, sino que ese comportamiento deseado se obtiene a partir de un conjunto de ejemplos[1].

Por dicha característica, el AM se ha convertido en una tecnología en pleno auge y se está extendiendo a diversas disciplinas de la ciencia [25, 26, 27], desarrollando además varias técnicas de predicción. Una forma de predecir es reconocer un patrón a partir de datos (entradas) y explicitarlo de manera tal que pueda dar respuestas (salidas) ante nuevos datos. En otras palabras, el AM hace posible estudiar y construir algoritmos[2] capaces de aprender modelos y programas desde conjuntos de datos utilizados para entrenar y hacer predicciones sobre datos antes no vistos. Este mismo enfoque es el abordado por esta tesis.

En términos formales, el objetivo del AM es la construcción de una función tal que dado un vector de atributos, llamado también vector de entrada, proporcione una estimación lo suficientemente buena de lo que se quiera hallar (objetivo), llamado valor de salida, que esté relacionado con dicho vector. Esto se hace de forma progresiva, ajustando la función que obtiene momentánea mediante otra función asignada para tal fin, al disminuir errores.

Cuando los algoritmos aprenden a partir de un conjunto de ejemplos dados etiquetados por expertos con la respuesta verdadera (i.e. con el resultado real) para asignar correctamente la salida a toda entrada de la misma naturaleza, el aprendizaje se denomina Aprendizaje Supervisado. Dicho de otra manera, la máquina aprende desde pares de objetos donde una componente del par son variables y la otra, los resultados reales. Este es el tipo de aprendizaje utilizado en el presente trabajo, ya que contamos con datos meteorológicos de ciertos lugares en ciertos momentos (entrada), y estimaciones de lluvia o no lluvia momentos posteriores en los mismos lugares (por observaciones reales).

[1] Dicho conjunto corresponde a la experiencia E mencionada por T. Mitchell.

[2] Un grupo finito de operaciones organizadas de manera lógica y ordenada que permite solucionar un determinado problema.

3.2. Un Problema de Clasificación

Como ya hemos descrito anteriormente, la idea principal del trabajo es hacer un estudio para que la máquina aprenda modelos de manera automática capaces de predecir la ocurrencia de precipitación. Para lograrlo, le entregamos a la máquina valores de variables meteorológicas (atributos) previos al evento lluvia en un lugar y un tiempo determinados y el dato de ocurrencia o no ocurrencia de lluvia en dicho lugar en un momento posterior; así, al repetir esta acción varias veces, la máquina logra reconocer un patrón causa-consecuencia y lo representa mediante una función, mientras minimiza el error que se le solicite. De esta manera, cuando se le dan nuevos valores de las variables, la función predecirá si lloverá o no. Entonces definimos el problema de predicción de precipitación como un problema de clasificación, donde lo que nos interesa es determinar si ante una configuración de variables meteorológicas establecida se producirá o no precipitación.

El objetivo de un Problema de Clasificación es determinar la clase a la que pertenece un objeto a partir de sus características (atributos). Para poder solucionarlo, hay supuestos fundamentales: los atributos determinan la clase, y el conjunto de ejemplos etiquetados provisto es lo suficientemente grande como para soportar la asociación atributo-clase. Pero usualmente no se sabe exactamente cuáles son las características que determinan la clase, tampoco cuáles son las causas que han generado los datos ni cómo esas causas se manifiestan en características observadas. También pueden faltar ejemplos para encontrar regularidades en la relación entre características y clase, y pueden faltar atributos relevantes, o más aún, puede haber un exceso de características (por ser algunas irrelevantes) (lo cual, finalmente, se traducirá en un bajo rendimiento práctico).

Luego, para que el modelo pueda predecir específicamente los ejemplos, se ajusta éste con los datos y se generaliza para que pueda predecir en nuevas instancias la clase adecuada.

Este procedimiento da por resultado un clasificador y se usa para asignar la información deseada a nuevos ejemplos no analizados por expertos. Así, el clasificador infiere cómo la información dada se asocia a la información deseada, es decir, cómo las características determinan la clase. Y se supone que existe algún modelo apropiado para capturar esta asociación, como se explica en el segundo capítulo de [28].

A continuación, describimos los detalles del proceso de aprendizaje.

3.3. Proceso de Aprendizaje

Para comprender mejor el proceso de aprendizaje de un modelo comenzaremos dando un ejemplo. Supongamos que contamos con algunos casos de cuando hubo o no precipitación y que

además contamos con datos sobre la temperatura y humedad que hubo en cada uno de esos casos (como muestra la Figura 3.3.1).

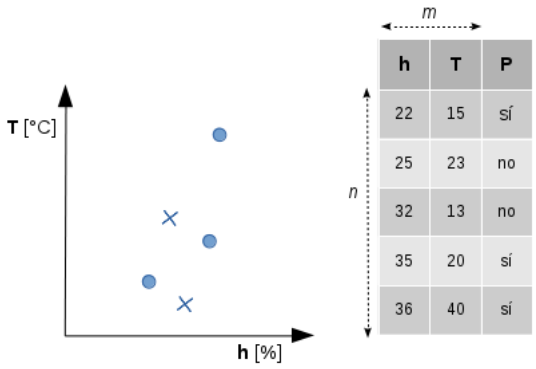


Figura 3.3.1: Ejemplo hipotético del problema de predicción de ocurrencia de precipitación (P) para dos variables, i.e. temperatura (T) y humedad (h). Del lado izquierdo se presenta un gráfico en el cual los círculos representan casos donde hubo precipitación (sí) y las cruces casos donde no hubo precipitación (no). Del lado derecho se presenta una tabla con algunos valores tomados por las variables y la correspondiente ocurrencia de precipitación.

La nomenclatura que utilizaremos en lo que resta del documento es la siguiente:

- n: número de casos o ejemplos. En nuestro ejemplo $n = 5$.
- m: número de variables de entrada. En nuestro ejemplo $m = 2$.
- $x \in \mathbb{R}^{m+1}$: representa un vector de las variables de entrada al cual extendemos con una primer componente $x_0 = 1$ para simplificar la notación. En el ejemplo dicho vector tiene dos componentes correspondientes a la temperatura y la humedad, es decir, $x \in \mathbb{R}^3$.
- y: es la variable de salida u objetivo que en nuestro caso puede tomar los valores 0 o 1 indicando la no ocurrencia de precipitación y la ocurrencia de precipitación respectivamente.
- (x, y): es el par ordenado que representa un ejemplo.
- $\left\{(x^{(i)}, y^{(i)})\right\}$: es el conjunto de entrenamiento, siendo $(x^{(i)}, y^{(i)})$ el i-ésimo ejemplo de entrenamiento, $x^{(i)} \in \mathbb{R}^{m+1}$ el vector de entrada (cuyas componentes son los atributos) del i-ésimo ejemplo de entrenamiento e $y^{(i)} \in \{0,1\}$ el objetivo del i-ésimo ejemplo de entrenamiento. En nuestro ejemplo $(x^{(4)}, y^{(4)}) = (1,35,20;1)$.
- $x_j^{(i)}$ con $j = 0,...,m$: es el j-ésimo valor (atributo) del vector de entrada del i-ésimo ejemplo de entrenamiento. Así $x_0^{(i)} = 1$, y en nuestro ejemplo $x_3^{(4)} = 20$.
- $y \in \{0,1\}^n$ el vector objetivo del conjunto de entrenamiento.

Es importante tener en cuenta que en nuestro caso elegimos el valor 0 como clase negativa *no-llueve* y el valor 1 como clase positiva *llueve*.

Como se mencionó anteriormente el objetivo último del AM es la obtención de un modelo de manera automática a partir de un conjunto de ejemplos. Ese modelo es una función que mapea el espacio de vectores de entrada con los valores de la variable objetivo. En la Figura 3.3.2 esto puede visualizarse para nuestro caso. Formalmente, el modelo es una función $h_\theta(x)\colon \mathbb{R}^{m+1} \to \{0,1\}^n$, siendo $\theta \in \mathbb{R}^{m+1}$ un vector de parámetros de dicha función, el cual representa interacciones entre las variables de entrada x y el objetivo y. Así por ejemplo $h_\theta(x^{(i)})$ es el valor obtenido con el vector de entrada del i-ésimo ejemplo. Existen numerosos tipos de modelos que pueden ser utilizados. En la Sección 3.4 se presentan los dos modelos utilizados en esta tesis, i.e. Regresión Logística y Redes Neuronales Artificiales.

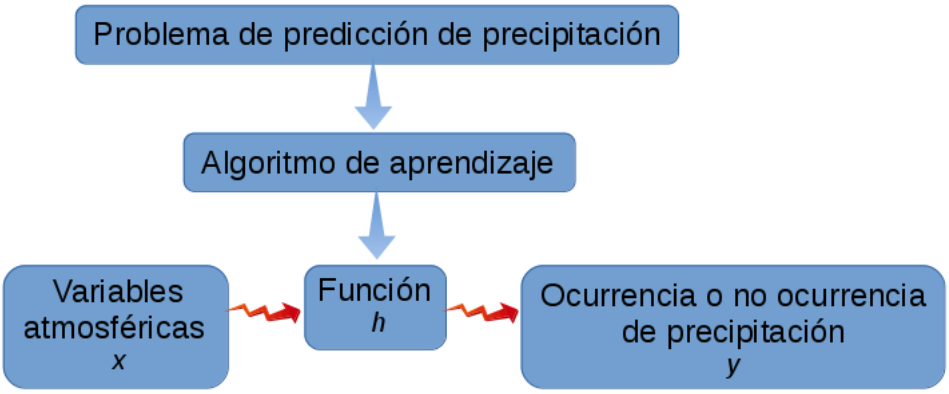


Figura 3.3.2: Esquema del proceso de aprendizaje para el problema de predicción de precipitación.

Aprender un modelo es entonces encontrar los valores de θ de forma tal que para cada ejemplo el valor dado por $h_\theta(x)$, se aproxime al valor objetivo y. Es decir, el objetivo del proceso de aprendizaje es minimizar la diferencia entre lo obtenido por $h_\theta(x^{(i)})$ y la respuesta verdadera $y^{(i)}$ $\forall i$. O sea que al elegir θ, se minimice una función que represente dicha diferencia y que se conoce como función de costo, notada $\text{costo}\,(h_\theta(x), y) = J(\theta)$; obteniendo así la función denotada $\min_\theta J(\theta)$ y el vector de parámetros θ adecuado para hacer una predicción dada una nueva entrada x.

Es significativo notar que $h_\theta(x)$ es una función que depende directamente de x (para valores fijos de θ), mientras $J(\theta)$ es función de θ (aunque éste dependerá indirectamente del conjunto de entrenamiento dado a la máquina).

3.3.1. Gradiente Descendente

Para lograr el objetivo del proceso de aprendizaje, utilizamos la técnica Gradiente Descendente [29], un método para problemas de optimización convexa.

El Gradiente Descendente radica en seguir la dirección del gradiente de una función error hasta llegar al mínimo. Comienza con un vector de parámetros inicial arbitrario que luego modifica según la dirección del mínimo más pronunciado de la función error, calculada mediante el gradiente. Este proceso continúa hasta que se alcanza el error mínimo global.

Por lo tanto, consiste en encontrar $\min_{\theta} J(\theta)$ asignando[3] $\theta_j \leftarrow \theta_j - \alpha \frac{\partial J(\theta)}{\partial \theta_j}$ con $\alpha \in \mathbb{R}$, computándolo simultáneamente $\forall j$, comenzando con un θ arbitrario, y repitiéndolo hasta converger. En Algoritmo 3.1 se presenta el pseudocódigo del método del Gradiente Descendente.

Algoritmo 3.1: Pseudocódigo del método de gradiente descendente.

repetir hasta lograr la convergencia {

$$\theta_j \leftarrow \theta_j - \alpha \frac{\partial J(\theta)}{\partial \theta_j}$$

}

La constante α se llama tasa de aprendizaje, ya que controla el tamaño de los pasos entre los cálculos de cada θ temporario, lo cual determina la rapidez de aprendizaje. Por ello es importante seleccionarla apropiadamente (ver Figura 3.3.3).

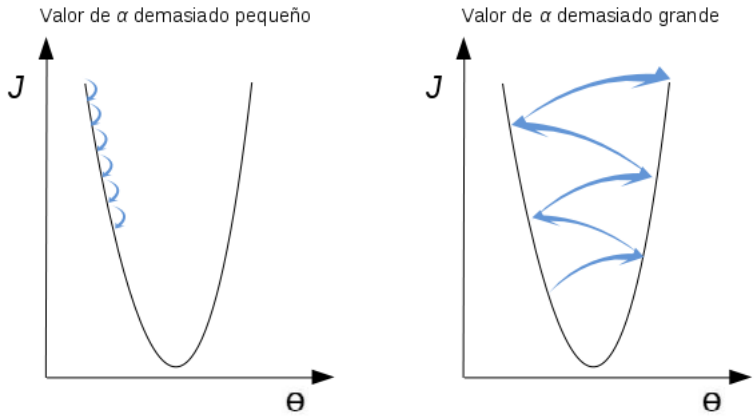


Figura 3.3.3: Representación de los pasos (flechas celestes) entre los cálculos de cada θ temporario según el valor de la tasa de aprendizaje α. Como sucede en el gráfico de la izquierda, cuando se le asigna a α un valor demasiado pequeño, el aprendizaje demora mucho tiempo porque los pasos son demasiado pequeños. En cambio, como sucede en el de la derecha, cuando se le asigna un valor muy grande, el proceso del Gradiente Descendente puede volverse inestable y no lograr converger a un mínimo.

En la sección siguiente describimos los modelos aplicados en esta tesis.

[3] En programación, una instrucción de asignación (o simplemente asignación) consiste en asignar el resultado de la evaluación de una expresión a una variable. Cambia el valor de la variable que está a la izquierda por un literal o el resultado de la expresión que se encuentra a la derecha (i.e. tipo de operador que sirve para almacenar un valor en una variable). Suele notarse $:=$ o $\leftarrow$. No debe confundirse con los términos matemáticos *definición* ni *igualdad*, notados $:=$ y $=$ respectivamente.

3.4. Modelos

Como se mencionó anteriormente, para el proceso de aprendizaje de la máquina, necesitamos un modelo y estimar sus parámetros en base a un conjunto de datos.

La elección del modelo que se le exige aprender a la máquina depende del problema que se quiera resolver y de la precisión, la calidad y la cantidad de datos [23]. Y según el modelo elegido, es la manera en que la máquina predecirá. El foco está puesto en poder medir la capacidad predictiva de los modelos, no en descubrir los factores causales directos que rigen sólo bajo los ejemplos dados.

El modelo puede capturar relaciones lineales o no lineales entre las distintas variables de análisis. Existe gran cantidad de modelos lineales que pueden utilizarse para problemas de clasificación. Dichos modelos representan un hiperplano en el espacio de atributos del problema[4] y discrimina a los ejemplos en una clase o la otra. Al momento de predecir, los modelos lineales utilizan dicho hiperplano para separar los ejemplos determinando si pertenecen a una u otra clase. Por dicho motivo, se hace referencia a este hiperplano como límite de decisión. Si bien los clasificadores lineales son modelos sencillos, estos se consideran una buena primera aproximación al problema en cuestión. Además, sirven de base para la comparación con modelos más complejos.

No obstante, en la práctica las clases pueden no ser linealmente separables por el límite de decisión, como se muestra en la Figura 3.4.1, imposibilitando la aplicación de modelos lineales. Además, si bien cuando se usa un modelo lineal éste es más sencillo de interpretar, uno no lineal es más representativo cuando la dinámica del problema es compleja.

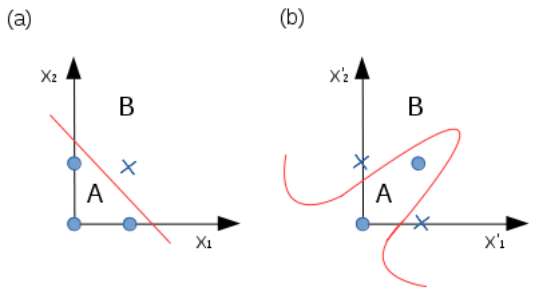


Figura 3.4.1: Gráficos en el que puede visualizarse la diferencia entre: (a) un caso en el que las clases (A y B) (correspondientes a vectores de entrada de variables x_1 y x_2) son linealmente separables por el límite de decisión (curva roja); y (b) un caso en el que no (cuyos vectores de entrada tienen variables $x_1^{'}$ y $x_2^{'}$).

En función de los discutido en los párrafos anteriores se describen los fundamentos teóricos de un modelo lineal ampliamente utilizado en el AM, i.e. Regresión Logística (RL). Dicho modelo

[4]En este trabajo el espacio de atributos se encuentra definido por el conjunto de variables meteorológicas utilizadas como entrada.

se discute en la siguiente sección y se incorpora a este estudio como una primera aproximación al problema y para lograr un mayor entendimiento de las características del mismo. Luego en la Sección 3.4.2 se presenta el modelo de Redes Neuronales Artificiales (RNAs), las cuales constituyen modelos no lineales capaces de capturar patrones complejos en los datos y por ende capaces de superar las limitaciones de representación de los modelos lineales.

3.4.1. Regresión Logística

En el caso de problemas de clasificación biclase se utiliza el modelo lineal RL. Su denominación se debe a que está basada en una función logística. Esta función presenta la ventaja de dar como resultados (salida) valores entre 0 y 1 (no-llueve, llueve), como mostramos más adelante. Luego, la salida se interpreta como la probabilidad de que la entrada pertenezca a la clase 1 (en nuestro caso, llueve).

Inmediatamente explicamos cómo aplicamos RL a nuestro caso.

Para asignarle una de las dos clases, *no-llueve/llueve*, a la entrada x, tomamos el producto escalar (combinación lineal) entre el vector de entrada $x^{(i)}$ del i-ésimo ejemplo y el vector de parámetros θ, y utilizamos la representación binaria del valor objetivo $y^{(i)} = t$: $t_1 = 0$ representa la clase *no-llueve* y $t_2 = 1$ la clase *llueve*. Así, el valor de $p(t)$ puede interpretarse como la probabilidad de que se asigne alguna de las clases. Estas condiciones naturalmente motivaron el uso de RL.

Así, la probabilidad de que el modelo prediga la clase t_2 si tiene como dato de entrada el vector de atributos x, es:

$$p(t_2|x) = \frac{p(x|t_2)p(t_2)}{p(x|t_2)p(t_2) + p(x|t_1)p(t_1)}$$

Siendo $z = \ln \frac{p(x|t_2)p(t_2)}{p(x|t_1)p(t_1)}$ se obtiene la función logística:

$$g(z) = \frac{1}{1 + \exp(-z)}$$

y podemos denotar

$$z = \theta^T x = \sum_{j=0}^{m} \theta_j x_j$$

Por lo tanto, la función hipótesis es

$$h_\theta(x) = \frac{1}{1 + exp(-\theta^T x)}$$

De esta forma, la probabilidad de obtener la clase t_2 como una función logística actuando sobre una función lineal z del vector de atributos x parametrizado por θ queda determinada por la función hipótesis:

$$p(t_2|x;\theta) = h_\theta(x)$$

Y la probabilidad de la clase t_1 por:

$$p(t_1|x;\theta) = 1 - p(t_2|x;\theta)$$

Entonces, para nuestro espacio de atributos $(m+1)$-dimensional, este modelo tiene $(m+1)$ parámetros ajustables. Y el límite de decisión, definido por los parámetros, se encuentra donde la probabilidad de predicción es 0.5 o mayor:

$$h_\theta(x) \geq 0.5$$

O sea, y toma el valor de 1 si $g(z) \geq 0.5$ lo cual sucede cuando $z = \theta^T x \geq 0$, y 0 si no (ver Figura 3.4.2).

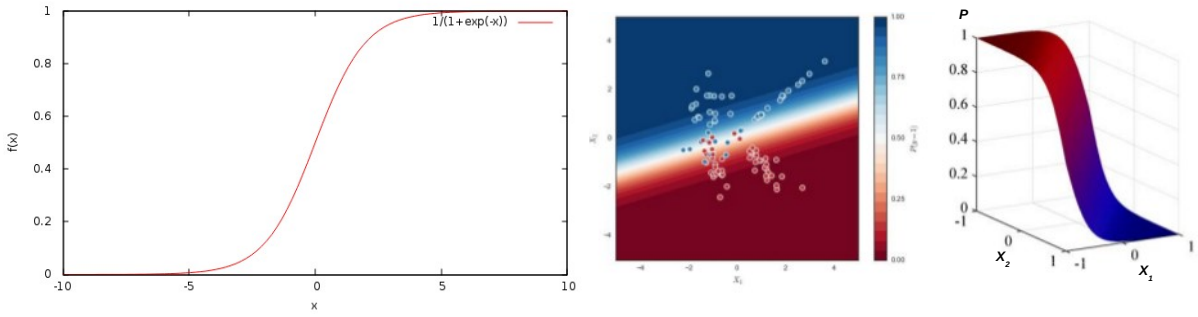


Figura 3.4.2: Se gráfica la función logística (también llamada función sigmoidea) y se representa a su derecha un clasificador lineal basado en RL para un problema hipotético de dos variables, X_1 y X_2. En color azul se observa la región para la cual el modelo predice que habrá lluvia y en color rojo, la región para la cual el modelo predice que no lloverá. El valor de salida del modelo se interpreta como la probabilidad de que el modelo prediga la ocurrencia de lluvia, la cual se referencia con colores y puede visualizarse como la altura de la gráfica en tres dimensiones (P).

3.4.1.1. Función de Costo

En este apartado describimos la función de costo que es optimizada mediante el método Gradiente Descendente (GD) para aprender modelos de RL. En problemas de clasificación en donde existen dos clases posibles, es común utilizar la función de costo denominada Entropía Cruzada.

Para comprender la expresión de la misma, antes de presentarla, cabe mencionar que cuando la función hipótesis es una función logística $h_\theta(x) = \frac{1}{1+\exp(-\theta^T x)}$, la función error cuadrática media $\frac{1}{2}\left(h_\theta(x^{(i)}) - y\right)^2$ (utilizada usualmente como función costo para modelos más simples) se vuelve una función no convexa[5]. Por lo que no podemos garantizar converger en un mínimo global.

Una forma simple de solucionarlo es pensarlo como una probabilidad logarítmica negativa para el dato $y^{(i)}$, bajo el modelo h. Lo que deseamos entonces disminuir en cada ejemplo es:

$$\text{costo}\left(h_\theta(x^{(i)}), y\right) = \begin{cases} -\log\left(h_\theta(x^{(i)})\right) & \text{si } y = 1 \\ -\log\left(1 - h_\theta(x^{(i)})\right) & \text{si } y = 0 \end{cases}$$

De forma compacta:

$$\text{costo}\left(h_\theta(x^{(i)}), y\right) = -y \log\left(h_\theta(x^{(i)})\right) - (1-y)\log\left(1 - h_\theta(x^{(i)})\right)$$

Y así, ponderando, obtenemos la Entropía Cruzada:

$$\text{costo}\left(h_\theta(x), y\right) = \frac{1}{n}\sum_{i=1}^{n}\left[y^{(i)} \log h_\theta(x^{(i)}) + \left(1 - y^{(i)}\right)\log\left(1 - h_\theta(x^{(i)})\right)\right] \tag{3.4.1}$$

Es importante resaltar que esta función hace que el problema que tiene que resolver GD sea de optimización convexa lo cual asegura que el método converja a la solución óptima.

[5]Una función $f(x)$ es convexa en un intervalo $[a,b]$ si para dos puntos x_1 y x_2 en $[a,b]$ y cualquier λ donde $0 < \lambda < 1$, $f[\lambda x_1 + (1-\lambda)x_2] \leq \lambda f(x_1) + (1-\lambda) f(x_2)$ (Rudin 1976, p. 101; cf. Gradshteyn and Ryzhik 2000, p. 1132) (i.e., es una función continua cuyo valor en el punto medio de cada intervalo en su dominio no supera la media aritmética de sus valores en los extremos del intervalo).

Así, una función de valor real definida en un intervalo n-dimensional se llama convexa (o convexa hacia abajo o cóncava hacia arriba) si el segmento de línea entre dos puntos cualesquiera en el gráfico de la función se encuentra por encima de o sobre el gráfico, en un espacio euclidiano (o más general, un espacio vectorial) de al menos dos dimensiones.

Además destacaremos dos extremos en la interpretación del costo computacional que puede ocurrir por las características de la función.

En un extremo, esta función puede tender a infinito ($-\log 0 \to \infty$) en dos situaciones. Una sucede cuando el modelo predice que alguna clase no tiene probabilidad de ocurrencia ($h = 0$) y, sin embargo, la clase en realidad aparece ($y = 1$); lo cual es físicamente incoherente. La otra, cuando el modelo predice que alguna clase tiene probabilidad de ocurrencia ($h = 1$) y, sin embargo, la clase en realidad no aparece ($y = 0$); lo cual convierte en incorrecta a la predicción. En ambas situaciones, significa que el costo por el aprendizaje del algoritmo es muy alto.

Para evitar este problema, hay que asegurarse de que el modelo no suponga un imposible mientras pueda suceder. Para ello, por lo general, se usan funciones de "máxima suavidad" como modelos de hipótesis, que dejan al menos alguna posibilidad para cada opción. Y si se utiliza algún otro modelo de hipótesis, corresponde regularizarlo ("suavizarlo") para que no hipotetice los ceros donde no debería. Por ello es que para la investigación informada por medio del presente, utilizamos la función logística.

En el otro extremo, si se predice que un evento ocurre y realmente ocurre ($y = 1$, $h = 1$) o si se predice que no ocurre y realmente no ocurre ($y = 0$, $h = 0$), no hay costo en el aprendizaje (en ambos casos da como resultado $\log 1 = 0$).

3.4.2. Redes Neuronales Artificiales

Como ya anticipamos, debido a que el problema a tratar en este trabajo involucra la compleja dinámica del flujo energético de la atmósfera, nos interesa estudiar también modelos no lineales para resolverlo.

Un modelo que en principio es capaz de capturar las complejas interacciones de las variables son las Redes Neuronales Artificiales (RNAs). Dichos modelos están inspirados en los sistemas nerviosos biológicos [30].

Precisamente esa inspiración fue el motivo por el cual las RNAs cuentan con importantes capacidades propias de los sistemas nerviosos de los seres vivos: procesar, almacenar y comunicar información de forma paralela; reconocer patrones complejos; aprender y generalizar lo aprendido. Además, y no menos importante, al igual que los sistemas biológicos, poseen una gran especialización funcional de sus unidades de cómputo [31]. Estos aspectos esenciales las convierten en poderosas herramientas para el estudio en nuestro caso.

Procedemos a describir entonces la configuración general y el modo de operar de dichas redes. Luego especificamos la red sometida al proceso de aprendizaje de nuestros experimentos al aplicar el modelo no lineal RNA.

3.4.2.1. Descripción

El desarrollo de las RNAs estuvo incentivado particularmente en el hecho de que las neuronas en el cerebro están masivamente interconectadas, lo que permite que un problema se descomponga en subproblemas que pueden resolverse de manera más simple. Según el Psicólogo canadiense Donald Olding Hebb, quien trató de comprender cómo la función de las neuronas contribuía a procesos psicológicos como el aprendizaje, los cambios adaptativos y el almacenamiento de información involucrado en al aprendizaje están asociados a la plasticidad de las fuerzas sinápticas [32].

Las RNAs constan de unidades de cómputo, llamadas neuronas artificiales o nodos, interconectadas dispuestas en capas que procesan la información dada y vínculos dirigidos, llamados pesos, que transmiten la información en base a una topología definida, como muestra la Figura 3.4.3. En dicha figura puede visualizarse que los nodos que no son objetivo de ninguna conexión constituyen la capa de entrada, los nodos que no son fuente de ninguna conexión componen la de salida, y el resto de las unidades se encuentra en las llamadas capas ocultas. Luego, la arquitectura particular que se diseña (por ejemplo, la cantidad de nodos en cada capa) responde a un tipo de problema concreto que se quiere resolver.

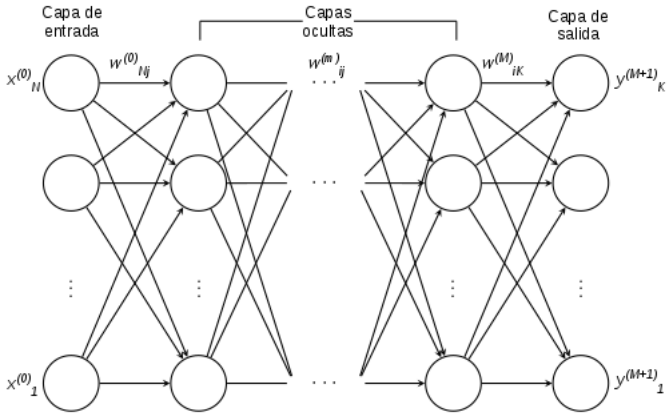


Figura 3.4.3: Diagrama de la estructura de una RNA simple con N nodos de entrada, M capas ocultas, K nodos de salida y dirección de propagación de información sólo hacia adelante. Se representan los nodos por círculos conectados por flechas y se indican los pesos de neurona a neurona como $w_{ij}^{(m)}$ con $0 \leq m \leq M$, $i = 1, ..., p$ y $j = 1, ..., q$, siendo p la neurona de la que "proviene" cada uno y q a la que "llega" (en el caso de los pesos que llegan a la primer capa oculta, p toma el valor N como máximo; y en el caso en que los pesos salen de la última, q toma como máximo el valor K).

El objetivo principal de estas redes es alcanzar un alto rendimiento de cómputo para cierta tarea mediante la densa interconexión de sus nodos [33]. Una red primero recibe una señal de entrada mediante un conjunto de datos ejemplos, luego su módulo de razonamiento se basa en la propagación de conocimiento (datos y pesos como parámetros libres) por las distintas capas y finalmente produce una señal de salida.

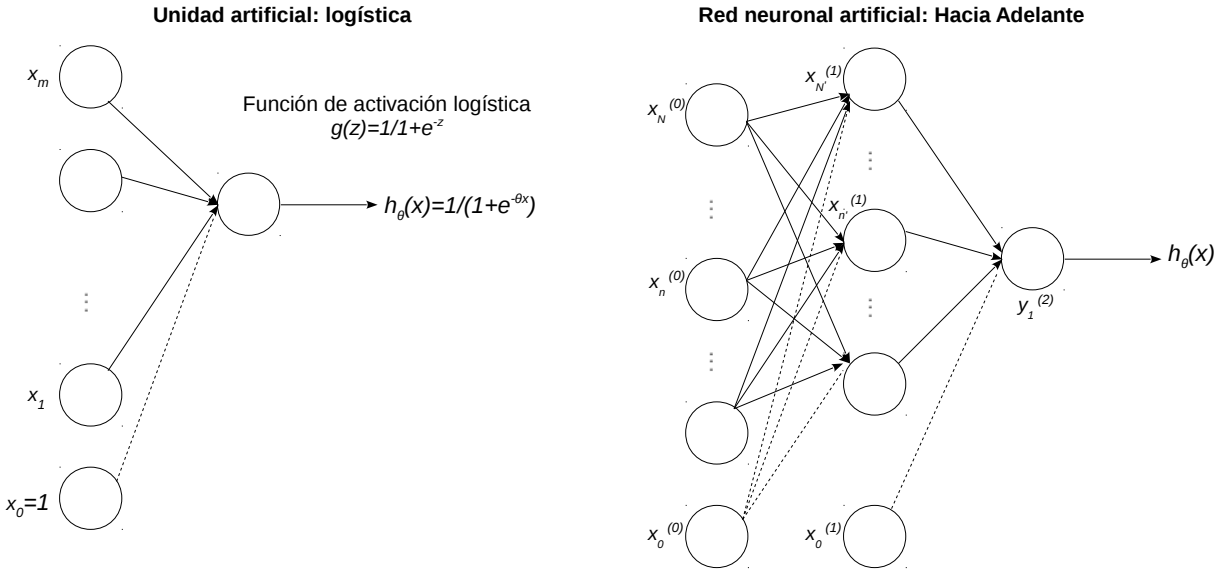


Figura 3.4.4: Diseño de una neurona artificial logística con unidad adicional (izquierda) y del modelo neuronal Red Neuronal Artificial Hacia Adelante (i.e. procesamiento de información unidireccional) con una capa oculta y un solo nodo de salida (derecha).

La funcionalidad de las redes viene dada tanto por los pesos como por la función concreta de activación de cada neurona (ver Figura 3.4.4). Para la presente tesis dicha función es logística (la cual presentamos en la Subsección 3.4.1), y la denotamos g. Entonces, denotando $x_{n'}^{(m)}$ a la función de activación de la unidad n' con $1 \leq n' \leq N'$ en la capa m y $\Theta^{(m)}$ a la matriz de elementos pesos que controla el mapeo de la capa m a la capa $m+1$:

$$x_{n'}^{(1)} = g\left(\Theta_{n'0}^{(0)} x_0^{(0)} + \Theta_{n'1}^{(0)} x_1^{(0)} + ... + \Theta_{n'N}^{(0)} x_N^{(0)}\right) = g\left(\sum_{i=0}^{N} \Theta_{n'i}^{(0)} x_i^{(0)}\right)$$

Por lo tanto, la función hipótesis para un ejemplo es

$$h_\Theta(x) = g\left(\Theta_{10}^{(1)} x_0^{(1)} + \Theta_{11}^{(1)} x_1^{(1)} + ... + \Theta_{1N'}^{(1)} x_{N'}^{(1)}\right) = g\left(\sum_{i=0}^{N'} \Theta_{1i}^{(1)} x_i^{(1)}\right)$$

Luego

$$\begin{aligned} h_\Theta(x) &= g\left\{\sum_{i=0}^{N'}\Theta_{1i}^{(1)}\left[g'\left(\sum_{j=0}^{N}\Theta_{n'j}^{(0)}x_j^{(0)}\right)\right]\right\} \\ &= \frac{1}{1+exp\left\{-\sum_{i=0}^{N'}\Theta_{1i}^{(1)}\left[g'\left(\sum_{j=0}^{N}\Theta_{n'j}^{(0)}x_j^{(0)}\right)\right]\right\}} \\ &= \frac{1}{1+exp\left(-\Theta^T x\right)} \end{aligned}$$

Respecto a índices, es importante no confundir el supraíndice entre paréntesis correspondiente al número de capa con el correspondiente al número de ejemplo dado en el proceso de aprendizaje; ni confundir el subíndice correspondiente al número de nodo con el número de componente de un vector en el dicho proceso.

Cabe mencionar que la función de costo, al aplicar el modelo RNA, es la misma utilizada para el aprendizaje de modelos de RL, es decir, Entropía Cruzada (Ecuación 3.4.1). Al considerar la matriz Θ y la función hipótesis $h_\Theta(x)$, obtenemos:

$$J(\Theta) = -\frac{1}{n}\sum_{i=1}^{n}\left[y^{(i)}logh_\Theta(x) + \left(1-y^{(i)}\right)log\left(1-h_\Theta(x)\right)\right]$$

Finalmente, dado que los pesos que conectan ambas capas dependen unos de otros, calcular el gradiente de $J(\Theta)$ no es trivial, y por ello se utiliza el método de Retropropagación.

3.4.2.2. Método de Retropropagación

Cuando en la década del 70 se requirió aplicar métodos de aprendizaje más complejos que el conocido hasta el momento (para entrenar la red denominada Perceptron [34]), se necesitaban tantas iteraciones que llevó a los expertos a explorar varios algoritmos [35]. Así fue que se llegó a pensar en actualizar el vector de parámetros involucrando dos decisiones: eligiendo una dirección en la que se modifique dicho vector y eligiendo una distancia para moverse en dicha dirección.

Un método que logra estas decisiones fundamentales de manera simple en las RNAs es la Retropropagación, publicado en 1886 por Rumelhart, Hinton y Williams [36]. En la actualidad sigue siendo utilizado por su efectividad, aunque adaptado según el tipo de problema [37, 38].

Para involucrar ambas decisiones en dicho método, la dirección se elige tomando el negativo del gradiente (i.e. computando errores hacia atrás para luego actualizarlos; he aquí la razón

de la denominación del método), y la distancia se determina por una constante denominada tasa de aprendizaje (que controla el tamaño de los pasos entre los cálculos de cada vector de parámetros temporario; de forma análoga a la tasa de aprendizaje de la Figura 3.3.3).

Para un mejor entendimiento, resumimos como sigue:

1. Se tiene $J(\Theta) = -\frac{1}{n}\sum_{i=1}^{n}\left[y^{(i)}\log h_\Theta(x) + \left(1-y^{(i)}\right)\log\left(1-h_\Theta(x)\right)\right]$
2. Se computa $-\frac{\partial J(\Theta)}{\partial \Theta_{ij}^{(m)}} = 0$ actualizando los parámetros $\Theta_{ij}^{(m)}$
3. Se obtiene $\min_\theta J(\theta)$

Recordamos que la red debe realizar una propagación hacia adelante. Una vez dado el vector de entrada $x^{(0)}$, la secuencia de cálculo es:

1. $\Theta^{(0)}x^{(0)}$
2. $x^{(1)} = g\left(\Theta^{(0)}x^{(0)}\right)$
3. $\Theta^{(1)}x^{(1)}$
4. $x^{(2)} = y_1^{(2)} = h_\Theta(x) = g\left(\Theta^{(1)}x^{(1)}\right)$

Además, antes de presentar el algoritmo de retropropagación, debemos mencionar que notaremos $\delta_{n'}^{(m)}$ al error del nodo n' en la capa m; por lo que $\delta_1^{(2)}$ será el error asociado a la unidad de salida. Esto nos permite simplificar notación al escribir los correspondientes vectores error:

- $\delta^{(2)} = h_\Theta(x) - y$
- $\delta^{(1)} = x^{(1)}\left(1-x^{(1)}\right)\left(\Theta^{(1)}\right)^T\delta^{(2)}$

También cabe aclarar que simbolizaremos λ a la tasa de aprendizaje y $\Delta_{ij}^{(m)}$ a $\frac{\partial J(\Theta)}{\partial \Theta_{ij}^{(m)}}$.

Proseguimos con el algoritmo de la Retropropagación para un conjunto de n ejemplos:

- Tener el conjunto de ejemplos $\left\{\left(x^{(i)}, y^{(i)}\right); i = 1,...,n\right\}$
- Iniciar los pesos de la red con valores cercanos a cero.
- Coleccionar $\Delta_{ij}^{(m)} = 0\ \forall i; j = 1,...,N'; m = 1,2$
- $\forall i$

  realizar la propagación hacia adelante para computar $x^{(m)}$

  computar $\delta^{(2)} = x^{(2)} - y^{(i)}$

  realizar la retropropagación para computar $\delta^{(1)}$

  $\Delta^{(m)} \leftarrow \Delta^{(m)} + \delta^{(m+1)}\left(x^{(m)}\right)^T$

- $\Delta_{ij}^{(m)} \leftarrow \frac{1}{n}\Delta_{ij}^{(m)}$
- Computar $-\lambda\Delta_{ij}^{(m)}$ para algún λ dado

Así, mediante el proceso explicado, minimizamos la función costo para mejorar el desempeño de la generalización en el proceso de aprendizaje del modelo RNA.

Capítulo 4

Estudio I: Análisis de Datos

Una vez expuestas en los capítulos anteriores las consideraciones generales necesarias para contextualizar y comprender nuestra investigación, damos a conocer a través del presente capítulo el primer estudio concluido, el cual determina las variables meteorológicas necesarias para el posterior estudio, donde se aplican los modelos.

En este capítulo analizamos los datos meteorológicos utilizados en el proceso de aprendizaje supervisado de los modelos. En la primer sección damos a conocer las bases de datos utilizadas y sus características de extensión espacial y temporal. A continuación exhibimos los análisis realizados con dichos datos, mediante dos eficaces herramientas computacionales. Por último, enunciamos los atributos de entrada del aprendizaje.

4.1. Datos

A fin de aprender los modelos de predicción de precipitación, se tomaron datos provenientes de distintas fuentes. Como queremos resolver un problema relacionado con la dinámica atmosférica y sus posibles impactos locales o regionales, utilizamos bases de datos climatológicos que contienen información de las propiedades estadísticas de los diferentes fenómenos registrados y bases de datos que contienen información acerca del estado global de la atmósfera.

Para contar con la cantidad y regularidad espacial y temporal de datos deseados, en pos de optimizar el desempeño de los modelos, consideramos el período comprendido entre los años 2000 y 2014 inclusive. Además, dado que estudiamos la ocurrencia de un fenómeno de características mesoescalares (i.e. cuya extensión espacial se encuentra en el orden de los cientos de kilómetros), acotamos el rango de datos de entrada a las regiones geográficas cercanas a las localizaciones bajo estudio, cubriendo las zonas de las ciudades de Mendoza y de Buenos Aires.

A continuación presentamos dichos datos.

Datos de Precipitación Estos datos corresponden a estimaciones de las precipitaciones a partir de datos de emisión de radiación electromagnética en el rango del infrarrojo y de microondas provenientes de topes nubosos (Climate Prediction Center morphing method–CMORPH–) [22]. Para el presente trabajo, utilizamos datos cuya resolución espacial es 0.25°x0.25° con tasa de precipitación promedio acumulada cada 3 horas, proporcionados por el Centro de Predicción Climática (CPC, por sus siglas en inglés) (`http://www.cpc.ncep.noaa.gov`).

Para cada ciudad bajo estudio, utilizamos los datos correspondientes a una grilla de 4x4 puntos geográficos, que luego fueron interpolados bilinealmente al punto en estudio: Aeropuerto Internacional Gobernador Francisco Gabrielli (32°49´54´´S 68°47´34´´O) en el caso de Mza y Aeroparque Metropolitano Jorge Newbery (34°33´32´´S 58°24´59´´O) en el de CABA. Representamos dichas grillas en la Figura 4.1.1.

Datos de Reanálisis Tales datos son variables meteorológicas sinópticas (Final Operational Global Tropospheric Analysis–FNL–) del Centro Nacional de Predicción Ambiental (NCEP, por sus siglas en inglés) [21]. Las mismas son una adaptación de las mediciones reales (estaciones de superficie, perfilado atmosférico, etc.) a una red regular de cobertura global, a 26 niveles verticales distribuidos entre 1000 mbar y 10 mbar con una resolución horizontal de 1°x1° cada 6 horas.

En la presente tesis utilizamos los datos correspondientes a una grilla de 3x3 puntos geográficos, las cuales se representan en la Figura 4.1.1, sobre las que se centraron las grillas correspondientes a los datos de precipitación.

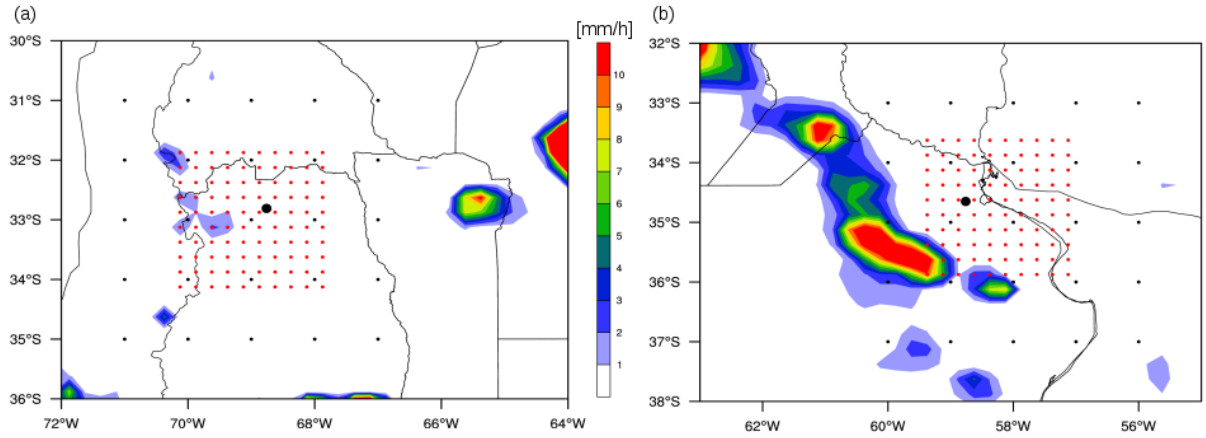


Figura 4.1.1: A modo ilustrativo se grafican sectores de las grillas FNL (puntos negros) y CMORPH (puntos rosa), en las que pueden observarse las grillas consideradas para nuestro estudio en Mendoza (a) y CABA (b). Ambas se encuentran globalmente y regularmente espaciadas en latitud y longitud en el globo terráqueo. Además, también como ejemplo, se indican intensidades de precipitación por la red regular de CMORPH según la escala de colores de referencia (en el centro). Los puntos mayores negros corresponden a la ubicación de las estaciones meteorológicas en El Plumerillo (a) y Aeroparque (b).

Para la elección de variables de la escala sinóptica con más influencia en el ciclo de vida de los fenómenos altamente influyentes para que ocurra la precipitación, como los procesos convectivos profundos, se eligen para este estudio aquellas variables que resultan más significativas según otros estudios existentes [39, 15]. Teniendo en cuenta esto y un posterior análisis detallado en la siguiente Sección 4.2, trabajamos con 9 de ellas como atributos de los modelos, las cuales se enuncian en la Sección 4.3.

Datos del Fenómeno El Niño Debido a que el Fenómeno El Niño es un factor de modulación en la distribución de precipitación a escala planetaria, se decidió incorporar datos referentes a éste en el proceso del aprendizaje de modelos. El Índice del Niño Oceánico (ONI, por sus siglas en inglés) corresponde también al CPC. Es un coeficiente en función de las anomalías de la temperatura promedio mensual a nivel del mar (TSM ERSST.v3b en la región Niño 3.4 [40]) (http://ggweather.com/enso/oni.htm) el cual se calcula tomando como base las temperaturas de dicho mes, el anterior y el posterior desde el año 1971 a 2015.

4.2. Análisis de Datos

Los datos disponibles que posteriormente serán utilizados en AM muchas veces deben ser recolectados previamente ensamblados, integrados, reducidos y preparados [41], de manera adecuada y según corresponda en cada caso.

Para la elección del diseño de los experimentos que se explicarán en el próximo capítulo, es fundamental un análisis previo de los datos disponibles para garantizar un buen desempeño

en el reconocimiento de patrones y la predicción de ocurrencia de precipitación por parte de los modelos.

Si bien es necesario contar con una cantidad suficiente de datos que se correspondan con el problema a resolver, la misma no se debe exceder; de manera tal que el modelo sea más simple que el conjunto de datos que representa. Los tamaños de las bases de datos con las que contamos son adecuados para nuestro estudio, lo cual nos permite realizar un análisis de los mismos y optar por los que consideremos más convenientes para realizar el Estudio II, el cual consiste en la aplicación de los modelos.

De esta forma, al reducir el número de datos que serán entradas en el proceso de aprendizaje de los modelos, no sólo nos servirá para mejorar el desempeño de éstos, sino también para mejorar el rendimiento en nuestra aplicación.

Sin embargo, además, la exploración de datos puede resultar decisiva en la consideración de otros datos para tratar determinado problema de clasificación, que anteriormente no se pensaban como relevantes, como ocurrió en este primer estudio.

Para lograr estos propósitos que resultarán beneficiosos a la hora de aplicar los modelos, hicimos dos análisis por separado, descritos en esta sección: de precipitaciones y de correlación entre variables.

4.2.1. Análisis de Precipitaciones

Sabiendo que pueden presentarse fenómenos meteorológicos con cierta regularidad temporal, es conveniente realizar algún análisis con el cual pueda reconocerse dicha frecuencia. Para el caso de precipitaciones en cierto tiempo y lugar, se usa el Análisis de Wavelets [42].

El Análisis de Wavelets se origina a causa de las limitaciones del Análisis de Fourier. Mientras en el segundo la resolución temporal y la resolución en frecuencias de una señal están acopladas (lo cual no permite ver ambas a la vez), el primer análisis mencionado sí ofrece información simultánea temporal y de frecuencia. Por ello, la representación de la señal al realizar un Análisis de Wavelets, es en términos de versiones trasladadas y dilatadas de una onda.

Este análisis, nos da información sobre el espectro de frecuencias en función del tiempo mediante una transformada. Las funciones utilizadas para hacer la transformada son varias, por lo que conviene usar aquella cuya forma se adecúe mejor al tipo de señal con el que se trabaja.

El hecho de que el análisis sea local, es lo que le da a la transformada de Wavelets sus interesantes propiedades. Así, este análisis presenta ciertas virtudes, como por ejemplo:

- estar especialmente indicado para señales con pulsos o intermitencias; sucesos que ocurren de manera no periódica.

- ser estable frente a señales de tipo intermitentes: si se añade un impulso localizado en el tiempo a una señal, sólo algunos coeficientes se modificarán.
- ser útil cuando los modos de vibración no son modos propios del sistema, ya que no mezcla información de los modos de oscilación al descomponer en modos propios mediante expansión local.

Por ello, para distinguir la frecuencia de máxima intensidad de las estimaciones de precipitaciones de cada zona, y determinar sus posibles causas, nosotros realizamos dicho análisis. Y para la transformada optamos por la función MORLET, entre las conocidas, para obtener gráficas con picos más pronunciados. La Figura 4.2.1 muestra los resultados para las dos regiones de estudio.

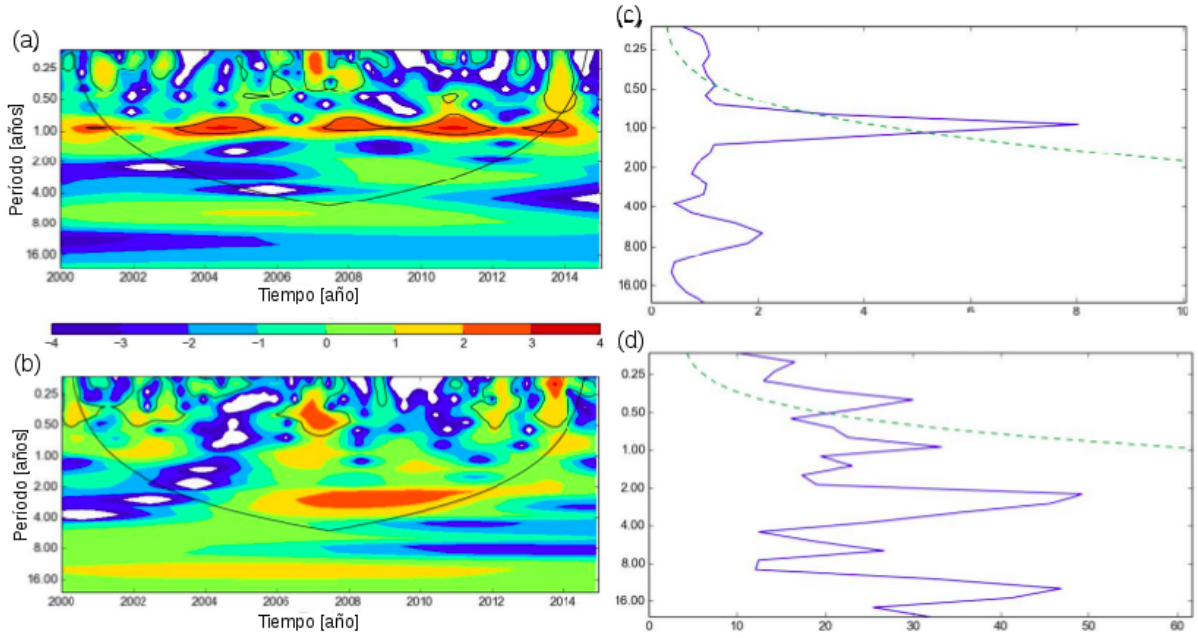


Figura 4.2.1: A la izquierda, espectro espacial y temporal de potencia de Wavelets para Mendoza (a) y CABA (b), donde los contornos de colores son las varianzas normalizadas, que indican la importancia de frecuencias particulares durante el período de estudio (2000-2014 inclusive). A la derecha, espectro de potencia global para Mendoza (c) y CABA (d), que indican la contribución total de las distintas frecuencias. La línea negra en (a) y (b) y la línea entrecortada en (c) y (d) representan el límite de la zona de confianza, siendo la región superior el 95%.

Como puede observarse en la figura, aunque con picos fuera de las zonas de confianza, los espectros de potencia indican un período anual claramente marcado en el caso de Mendoza y otro período entre 3 y 4 años en el caso de CABA.

Basados en estas observaciones y en la existencia de fuentes que aseguran que el fenómeno El Niño tiene un período de oscilación entre 2 y 10 años y que modula la intensidad mundial de precipitación [43], advertimos la periodicidad anual de lluvia y la posibilidad de que El Niño

aporte información nueva para explicar el fenómeno estudiado. Por lo tanto, aún sin poder asegurar esto último, decidimos agregar el ONI como característica de entrada para los modelos y trabajar sólo con datos correspondientes a las estaciones verano (al considerar diciembre, enero y febrero) e invierno (junio, julio y agosto). Por ello, trabajamos con dos entradas más: El Niño anual y El Niño mensual.

Correspondiéndose con las imágenes de la figura anterior, la cantidad de instancias de ocurrencia y de no ocurrencia de lluvia en cada par ciudad-estación, puede observarse rápidamente en la Figura 4.2.2, notándose un desbalanceo de clases.

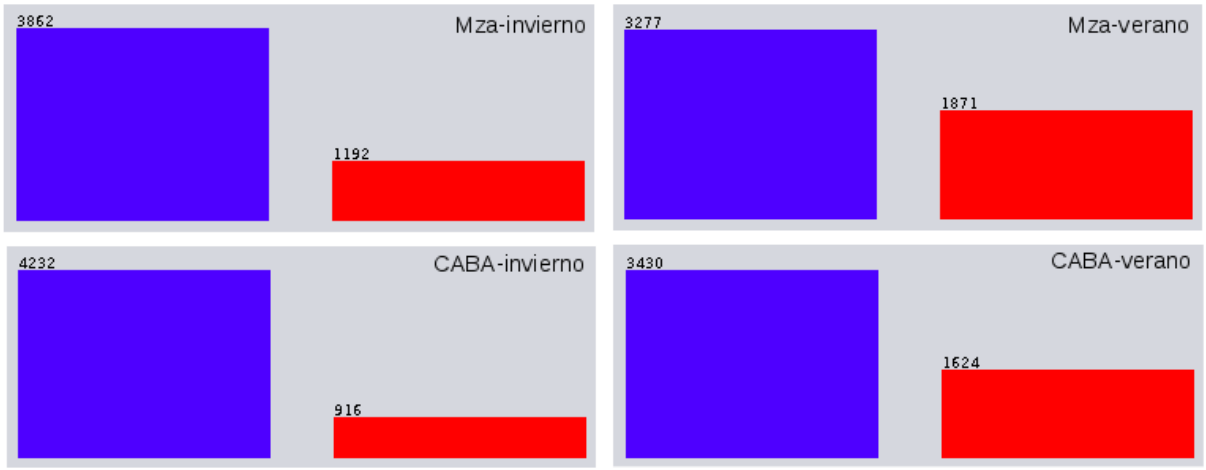


Figura 4.2.2: Cantidad de instancias de precipitación (rojo) y no precipitación (azul) de los datos considerados, correspondientes a Mza-verano, Mza-invierno, CABA-verano y CABA-invierno.

4.2.2. Análisis de Correlaciones entre Variables

A fin de mejorar el rendimiento práctico de la máquina, se considera una posible reducción de variables de entrada del proceso de aprendizaje, mediante grado de interacción de las mismas.

Para analizar las interacciones entre las variables meteorológicas llevamos a cabo un análisis de las correlaciones lineales existentes. Para ello computamos las matrices de correlación entre los atributos considerados. La Figura 4.2.3 presenta los mapas de calor de dichas correlaciones para las dos regiones bajo estudio [44]. Los colores fríos representan correlaciones positivas altas entre pares de variables, y los cálidos las negativas altas.

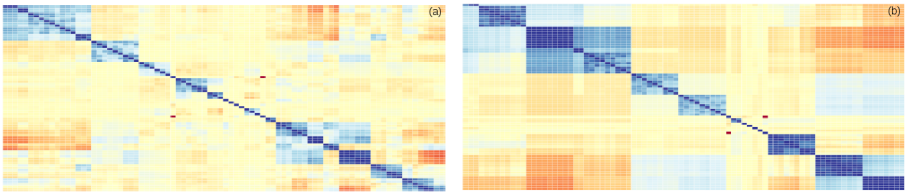


Figura 4.2.3: Mapas de Calor anuales de las matrices de correlación de todos los atributos de cada punto considerado, para Mendoza (a) y CABA (b). En colores fríos se miden las correlaciones positivas altas entre pares de variables. En colores cálidos se miden las correlaciones negativas altas entre pares de variables.

En la figura puede observarse que existen bloques distinguibles por similitud de color. Esto indica mayor correlación entre los pares de variables fila-columna (de la matriz) del bloque, en comparación a la correlación entre alguna de ellas y otra no perteneciente al par.

Lo que observamos a partir de los mapas de calor es que en el caso de CABA existe una mayor correlación zonal. En términos meteorológicos esto podría llegar a adjudicarse a la diferencia topográfica entre las regiones; en la provincia de Buenos Aires, caracterizada por ser una zona llana en comparación a Mendoza, donde existe un marcado gradiente topográfico.

4.3. Atributos para el Aprendizaje

En la Cuadro 4.1, se enumeran los atributos de los que constará el vector de entrada del proceso de aprendizaje de los modelos aplicados a la predicción de ocurrencia de precipitación en CABA y Mza. Para una mejor interpretación de lo que realizamos, llamaremos *evento* al momento en el que se observa si hay o no precipitación como dato objetivo.

Es importante mencionar que los datos de Reanálisis considerados para el aprendizaje de los modelos, corresponden a los valores medidos 6 horas antes del *evento*.

Además, cabe recordar, que la variable de salida es 0 o 1 indicando no ocurrencia de precipitación u ocurrencia de precipitación respectivamente.

Cuadro 4.1: Atributos considerados como entrada para el aprendizaje de modelos predictivos.

Fuente de datos	Variable	Cant.	Nivel	Descripción
Precipitaciones	Hora	x1	–	Hora del evento
	Día	x1	-	Día del evento
	Mes	x1	-	Mes del evento
	Año	x1	-	Año del evento
Reanálisis	Temperatura [°C]	x9	500 hPa	Temperatura
	Humedad relativa [%]	x9	850 hPa	Indica contenido del vapor de agua respecto a la saturación
	Vorticidad absoluta [s^{-1}]	x9	500 hPa	Indica el grado de rotación planetaria y local del fluido.
	Divergencia [s^{-1}]	x9	200 hPa	Indica el grado de convergencia del flujo atmosférico.
	Viento zonal (u) [m/s]	x9	850 hPa	Componente oeste-este del viento
	Viento meridional (v) [m/s]	x9	850 hPa	Componente norte-sur del viento
	Presión	x9	a nivel del mar	Presión
	CAPE [J/kg]	x9	-	Energía potencial disponible para el inicio de convección
	CIN [J/kg]	x9	-	Barrera de potencial energética a vencer para iniciar la convección
El Niño	Anual	x1	a nivel del mar	Anomalía de la temperatura media de tres meses consecutivos.
	Mensual	x1	a nivel del mar	Se categorizan en débil, moderado, fuerte y muy fuerte.

4.4. Conclusión

En este capítulo dimos a conocer la estructura de los datos y los atributos meteorológicos considerados para la elección de aquellos que se tomarán como entrada en el aprendizaje de los modelos propuestos en los capítulos anteriores. Primero ubicamos dichos datos en tiempo y

espacio, y en la última sección, en base a los resultados de los análisis exhibidos, proveímos una lista de los 12 atributos seleccionados.

En el siguiente capítulo se aprenden dichos modelos predictivos, a través de diferentes técnicas de AM, y se los compara con el modelo WRF.

Capítulo 5

Estudio II: Aprendizaje de Modelos

Si bien hoy en día, mundialmente, se utilizan modelos de simulación NWP para predecir fenómenos climatológicos, éstos son computacionalmente costosos y aún presentan limitaciones (ver Capítulo 2). Por ello, en el presente capítulo estudiamos la aplicabilidad de modelos alternativos para dicho fin.

En este segundo estudio verificamos la aplicabilidad operacional de AM de forma automática mediante RL y RNA, modelos lineal y no lineal respectivamente (presentados en el Capítulo 3), para predecir ocurrencia de precipitación en Mza y en CABA en invierno y en verano, a partir de datos históricos.

Para asegurarnos de obtener resultados confiables, en función de lo encontrado en el estudio anterior (ver Sección 4.2), realizamos preprocesaminento de los datos en el período explicitados en el capítulo anterior. Además, luego del proceso de aprendizaje, validamos y evaluamos el desempeño de los modelos con configuraciones particulares, para ambas ciudades y estaciones. Y finalmente convalidamos los resultados de los modelos aprendidos contrastando sus desempeños con el modelo de simulación más utilizado en el mundo, WRF.

Las diferentes técnicas de AM utilizadas se llevaron a cabo mediante la plataforma Weka [45]. La misma provee las implementaciones de los algoritmos de AM y los modelos utilizados en este trabajo, y además permite la visualización y realización de análisis estadísticos.

El presente capítulo estará organizado de la siguiente manera. En la Sección 5.1 explicamos las técnicas utilizadas para realizar el preproceso de datos. La segunda sección consta de las configuraciones seleccionadas de los modelos, la manera en que validamos el proceso de aprendizaje y los resultados de dicha validación expresados por distintas métricas. En la tercer sección mostramos la evaluación de los modelos con mejor desempeño en el aprendizaje y la comparación con el desempeño del modelo de simulación WRF. En la última sección, terminamos el capítulo con las conclusiones del mismo.

5.1. Preproceso de Datos

Para que los algoritmos tengan un mejor funcionamiento, es necesario preprocesar los datos de manera que estos estén mejor preparados y que se facilite el proceso de aprendizaje. En función de las características de los datos obtenidas como parte del análisis de datos presentado en el capítulo anterior, creemos apropiado realizar dos tipos de operaciones de preproceso: balance de clases y selección de atributos. Ambos tipos de preproceso se discuten en las Subsecciones 5.1.1 y 5.1.2.

5.1.1. Balance de Clases

Conociendo que en la climatología regional estudiada ocurren pocas precipitaciones [46] y evidenciando en la Subsección 4.2.1 un desbalance de clases *llueve* y *no-llueve* en el período estudiado en cada ciudad y estación, decidimos experimentar tanto con las clases desbalanceadas como con las clases previamente balanceadas sobre el conjunto de ejemplos sometido al aprendizaje.

Para balancear las clases utilizamos la técnica de remuestreo (a la que haremos referencia con R). Esta técnica produce una submuestra al azar de conjunto de datos, siendo capaz de igualar las cantidades de datos objetivo de cada clase al crear datos objetivo de la clase que se le especifique. En este estudio especificamos que las dos clases tuviesen igual cantidad de instancias al aumentar datos sobre la clase minoritaria (clase *llueve*), quedando cada conjunto de datos con las cantidades de instancias en ambas clases reportadas en la Cuadro 5.1.

Cuadro 5.1: Cantidad de instancias al utilizar remuestreo.

Ciudad	Estación	Cantidad de Instancias		
		Original		Remuestreo
		No precipitación	Precipitación	
Mza	invierno	3862	1192	3534
	verano	3277	1871	4114
CABA	invierno	4232	916	4489
	verano	3430	1624	3682

5.1.2. Selección de Atributos

Como se concluyó en la Subsección 4.2.2, existe mayor correlación entre variables meteorológicas en algunos puntos de las grillas geográficas consideradas. Esto significa que algunas de las variables contienen información redundante y por lo tanto pueden ser excluidas.

Por ello, exploramos el uso de dos métodos para hacer una selección automática de atributos a modo de reducir la cantidad de variables excluyendo aquellas que pudieran ser redundantes. A continuación se presentan ambos métodos.

5.1.2.1. Selección de Atributos Basado en Correlaciones de Subconjuntos

Esta técnica (en inglés Correlation-based feature Subset Selection –CFS–) [47] selecciona un subconjunto basado en la correlación de atributos. Lo realiza de la siguiente manera: evalúa el mérito de un subconjunto de atributos mediante la consideración de la capacidad predictiva individual de cada uno junto al grado de redundancia entre ellos. Hay que tener en cuenta que tiene preferencia por los subconjuntos de características que guardan una relación directa con la clase aunque tengan menor intercorrelación.

5.1.2.2. Evaluación de Consistencia de Subconjuntos

Esta otra técnica (en inglés Consistency Subset Eval –CON–) [48] evalúa el mérito de un subconjunto de atributos mediante el nivel de consistencia en los valores de la clase cuando las instancias de entrenamiento son proyectadas dentro del subconjunto de atributos. Es decir, para seleccionar se basa en la consistencia de los valores de los atributos respecto a las clases. La consistencia de cualquier subconjunto nunca puede ser menor que la de todo el conjunto de atributos, por lo tanto, la práctica habitual es utilizar este subconjunto evaluador junto con una investigación aleatoria o exhaustiva que busca el menor subconjunto con una consistencia igual a la del conjunto de todos los atributos.

5.2. Selección de Modelos

Una vez balanceadas las clases y seleccionados los atributos, se aplican los modelos. En este trabajo proponemos la aplicación de los modelos RL y RNA, presentados en las subsecciones 3.4.1 y 3.4.2. La red artificial utilizada cada vez que apliquemos RNA es la Red Neuronal Artificial Hacia Adelante (i.e. con procesamiento de información unidireccional) con una capa oculta, un nodo de salida y función de activación logística (tanto en los nodos ocultos como en el de salida).

Dado que existen numerosas configuraciones posibles en cuanto a la conformación de los modelos así como parámetros de su aprendizaje, es necesario determinar qué modelo es el que mejor desempeño tiene para poder ser evaluado y/o comparado con otros modelos alternativos. En el área de AM este proceso se conoce como *selección de modelos*. Para hacer la selección

de los mejores modelos tanto de RL como de RNA, consideramos los siguientes parámetros en lo que se refiere al preproceso de datos:

- balance de clases: *no*, *sí*; y
- selección de atributos: *no*, *CFS*, *CON*.

En el caso de las RNAs además evaluamos las configuraciones que surgen de variar:

- cantidad de neuronas ocultas (N'): $\{10, 30, 100\}$, y
- tiempo de entrenamiento (t) (i.e. cantidad de veces que se le presentan los datos a la red para actualizar los pesos): $\{500, 1000, 1500\}$.

Las posibles combinaciones resultan en un total de 6 configuraciones para el modelo lineal RL (ver Cuadro 5.2) y de 54 para el modelo no lineal RNA (ver Cuadro 5.3).

Cuadro 5.2: Configuraciones exploradas para los modelos RL.

Parámetro	Valores posibles	Descripción
Balance de clases	{sin R, con R}	si se aplica o no remuestreo para balancear clases
Selección de atributos	{no, CFS, CON}	sin selección, selección mediante CFS o selección mediante CON

Cuadro 5.3: Configuraciones exploradas para los modelos RNA.

Parámetro	Valores posibles	Descripción
Balance de clases	{sin R, con R}	si se aplica o no remuestreo para balancear clases
Selección de atributos	{no, CFS, CON}	sin selección, selección mediante CFS o selección mediante CON
Neuronas ocultas	$\{10, 30, 100\}$	cantidad de nodos ocultos
Tiempo de entrenamiento	$\{500, 1000, 1500\}$	cantidad de veces que se le presentan los datos a la red para actualizar los pesos

Es importante tener en cuenta que como conjunto de entrada utilizamos la totalidad de los datos disponibles a excepción de los datos correspondientes al año 2011. Dicho subconjunto de datos se reserva a los fines de la evaluación de los modelos, lo cual será discutido en la Sección 5.3.

Cabe mencionar que cualquier opción de cambio en Weka que no haya sido especificada en esta sección, se tomó por defecto (e.g. el valor de la tasa de aprendizaje α, presentado en la Subsección 3.3.1, fue aquel que figura por defecto: 0.3).

Además, para realizar la selección de modelos optamos por llevar a cabo una validación cruzada de k iteraciones para el proceso de aprendizaje y luego tomamos como métrica de comparación de desempeño la precisión, es decir, porcentaje de aciertos de cada modelo.

5.2.1. Validación Cruzada de k Iteraciones

La validación cruzada es una técnica para evaluar los resultados del proceso de aprendizaje de algún modelo garantizando que sean independientes de los datos ejemplo dados; i.e. garantiza que la máquina es capaz de generalizar cuando se le presenten nuevos datos. Es usualmente utilizada en contextos en los que se quiere estimar la precisión de modelos predictivos, como ocurre en nuestro caso.

El método de validación cruzada de k iteraciones divide el conjunto de ejemplos de entrada en k subconjuntos. Uno de los subconjuntos se utiliza para la validación del entrenamiento de los otros $(k-1)$ subconjuntos. Este proceso se repite k veces, con cada uno de los posibles k subconjuntos, como se ejemplifica en la Figura 5.2.1. Luego se calcula la media aritmética de los resultados de cada iteración para obtener un único resultado. De esta forma, evaluando sobre las k combinaciones de subconjuntos de entrenamiento y validación, se reduce el sesgo de selección y la varianza en el poder de predicción. La elección del número de iteraciones depende de la medida del conjunto de ejemplos.

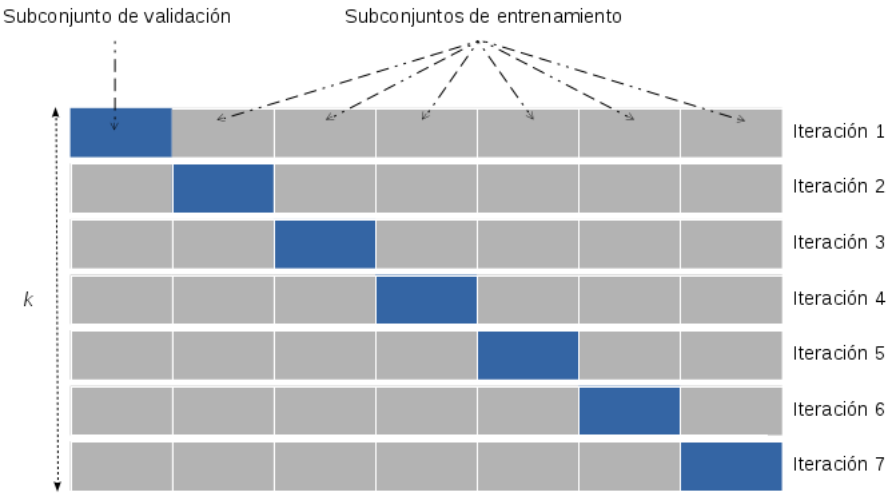


Figura 5.2.1: Representación de los subconjuntos del método de validación cruzada de 7 iteraciones. Los rectángulos grises hacen referencia a los de entrenamiento, y los azules a los de validación. Con ellos se entrena un modelo con seis subconjuntos y se valida con uno. Así, se realiza el proceso de aprendizaje con diferentes subconjuntos de validación.

Para tener una medida robusta del desempeño en este estudio utilizamos 10 iteraciones. Utilizar dicha cantidad es una práctica común en el área de AM, y en el artículo [49] pueden encontrarse justificaciones de ello.

5.2.2. Resultados

Una vez que realizamos la validación cruzada de 10 iteraciones de todos los parámetros y todas las configuraciones de los modelos aplicados a cada ciudad y estación en estudio, en esta sección exponemos en gráficas los mejores valores obtenidos de cuatro métricas que describimos a continuación, de las cuales dos resultan de cálculos con valores de la Matriz de Confusión representada en la Figura 5.2.2.

Matriz de confusión		*Valor objetivo*	
		1	***0***
Valor predicho	***1***	**VP**	**FP**
	0	**FN**	**VN**

Figura 5.2.2: Matriz de confusión, cuyos valores (VP, FP, FN, VN) en las columnas corresponden a datos objetivo y los de las filas a los predichos por el modelo aplicado; siendo el valor 0 la clase negativa *no-llueve* y el valor 1 la clase positiva *llueve*. VP son los verdaderos positivos, FP los falsos positivos, FN los falsos negativos y VN los verdaderos negativos.

Porcentaje de aciertos Proporción de predicciones correctas. En nuestro caso, son las veces que el modelo *predijo que iba a llover* y la salida era *llueve*, sumado a las veces que *predijo que no iba a llover* y la salida era *no-llueve*, dividido el total de predicciones.

Correspondiéndose a la matriz de confusión de la figura de arriba:

$$Aciertos = \frac{VP + VN}{VP + FP + FN + VN}$$

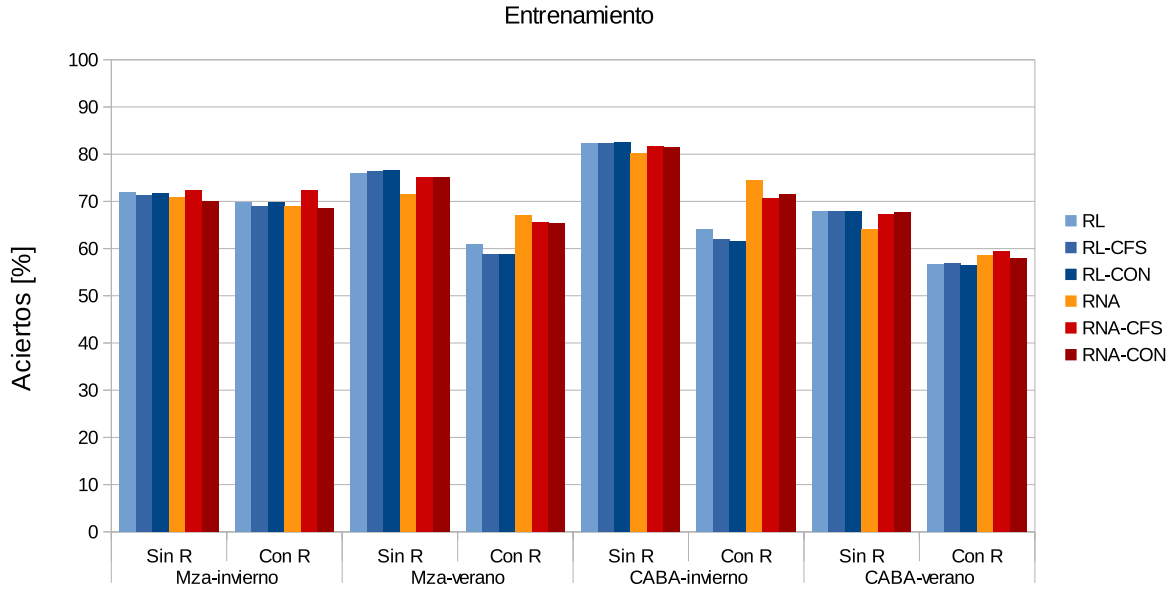
Tasa de Falsos Negativos (TFN) Proporción de casos en que la máquina *predijo que no iba a llover*, pero eran casos de salida clase *llueve*.

Correspondiéndose a la matriz de confusión de la figura de arriba:

$$TFN = \frac{FN}{VP + FN}$$

Tiempo de Entrenamiento Demora de la unidad central de procesamiento de la máquina en entrenar el modelo.

Tiempo de Validación Demora de la unidad central de procesamiento de la máquina en validar el modelo.



Aciertos		RL	RL-CFS	RL-CON	RNA	RNA-CFS	RNA-CON
Mza-invierno	Sin R	71.89	71.19	71.79	70.88	72.27	70.03
	Con R	69.69	69.04	69.83	68.92	72.43	68.55
Mza-verano	Sin R	75.98	76.46	76.63	71.53	75.02	75.01
	Con R	60.87	58.73	58.88	66.97	65.63	65.31
CABA-invierno	Sin R	82.39	82.19	**82.43**	80.21	**81.67**	81.35
	Con R	64.16	61.88	61.6	74.42	70.71	71.48
CABA-verano	Sin R	67.85	67.93	67.81	64.11	67.35	67.76
	Con R	56.73	56.78	**56.38**	58.63	59.4	**57.92**

Figura 5.2.3: Gráfica de barras de los mayores porcentajes de aciertos de los modelos aplicados en datos de cada ciudad y estación sin remuestreo (sin R) o con remuestreo (con R) y sin selección de atributos, con CFS o con CON. Debajo, los valores en una tabla, donde se exponen en negrita los mayores y menores entre las distintas configuraciones de RL y entre las de RNA.

En esta primer gráfica de resultados del Estudio II, Figura 5.2.3, visualizamos que los mayores porcentajes de aciertos de los 60 modelos aprendidos con validación cruzada están entre 56% y 83%. Y de estos 48 modelos los mayores porcentajes, entre 67% y 83%, son aquellos correspondientes a los 8 modelos (4 con CFS, 3 con CON y 1 sin selección de atributos) cuyas configuraciones resumimos en el Cuadro 5.2 y el Cuadro 5.3. Puede verse también, tanto en la figura como en los cuadros, que el máximo se obtiene con RL aplicado a los datos de CABA-invierno.

Además, si bien no por mucha diferencia de valores de porcentajes (sólo en el caso de CABA-invierno con remuestreo entre RL y RNA llega al 10% la diferencia), al comparar los modelos RL con RNA, RL-CFS con RNA-CFS y RL-CON con RNA-CON, existe una diferencia de 2 casos favorables para configuraciones correspondientes a RL (el total es 13 para RL y 11 para RNA).

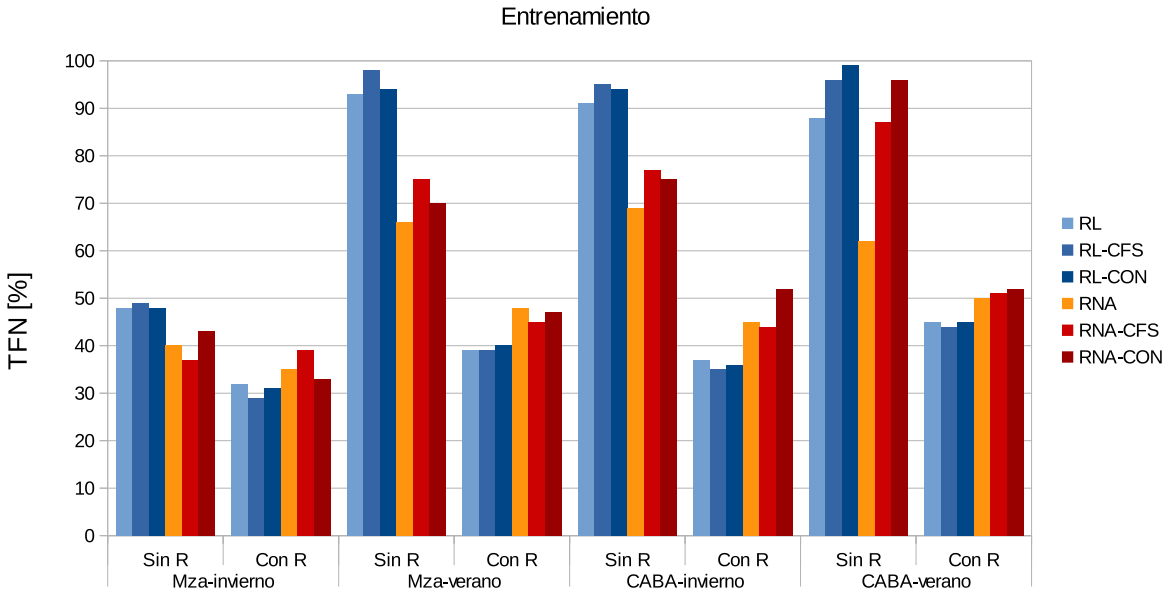
El peor caso de aciertos corresponde al modelo RL-CON con remuestreo aplicado a los datos correspondientes a CABA-verano.

Cuadro 5.4: Configuraciones con mayores porcentajes de aciertos para los modelos de RL.

Ciudad	Estación	Balance de clases	Selección atributos	Aciertos [%]
Mza	invierno	no	no	71.89
	verano	no	CON	76.63
CABA	invierno	no	CON	82.43
	verano	no	CFS	67.93

Cuadro 5.5: Configuraciones con mayores porcentajes de aciertos para los modelos de RNA.

Ciudad	Estación	Balance de clases	Selección atributos	N'	t	Aciertos [%]
Mza	invierno	sí	CFS	10	500	72.43
	verano	no	CFS	10	500	75.02
CABA	invierno	no	CFS	10	500	81.67
	verano	no	CON	10	500	67.76



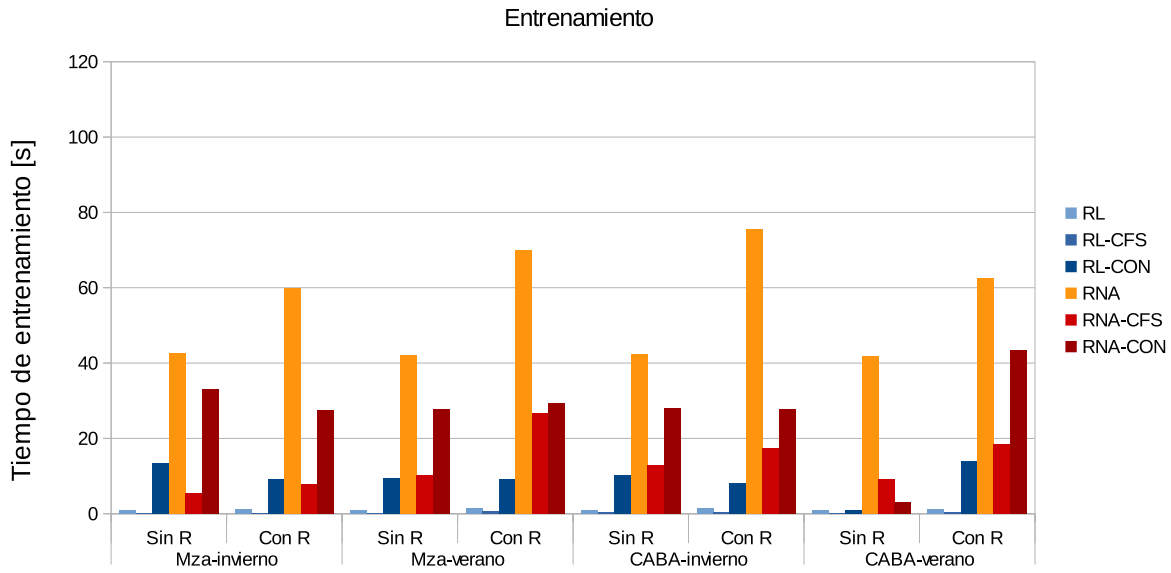
TFN		RL	RL-CFS	RL-CON	RNA	RNA-CFS	RNA-CON
Mza-invierno	Sin R	48	49	48	40	37	43
	Con R	32	**29**	31	35	39	**33**
Mza-verano	Sin R	93	98	94	66	75	70
	Con R	39	39	40	48	45	47
CABA-invierno	Sin R	91	95	94	69	77	75
	Con R	37	35	36	45	44	52
CABA-verano	Sin R	88	96	**99**	62	87	**96**
	Con R	45	44	45	50	51	52

Figura 5.2.4: Gráfica de barras de los menores porcentajes de tasas de falsos negativos de los modelos aplicados en datos de cada ciudad y estación sin remuestreo (sin R) o con remuestreo (con R) y sin selección de atributos, con CFS o con CON. Debajo, los valores en una tabla, donde se exponen en negrita los mayores y menores entre las distintas configuraciones de RL y entre las de RNA.

En esta segunda gráfica de barras, Figura 5.2.4, notamos que los menores porcentajes de TFN de los 60 modelos aprendidos con validación cruzada están entre 29 % y 99 %.

También observamos que hay una gran diferencia entre los porcentajes obtenidos al balancear y no balancear las clases de modelos RL; llegando a un 60 % menos en el caso de haber utilizado remuestreo (CABA-invierno RL-CFS). En los casos en que se aplica RNA la diferencia no es tan grande (llegando en un solo caso al 44 %; la mayoría está por debajo del 24 %).

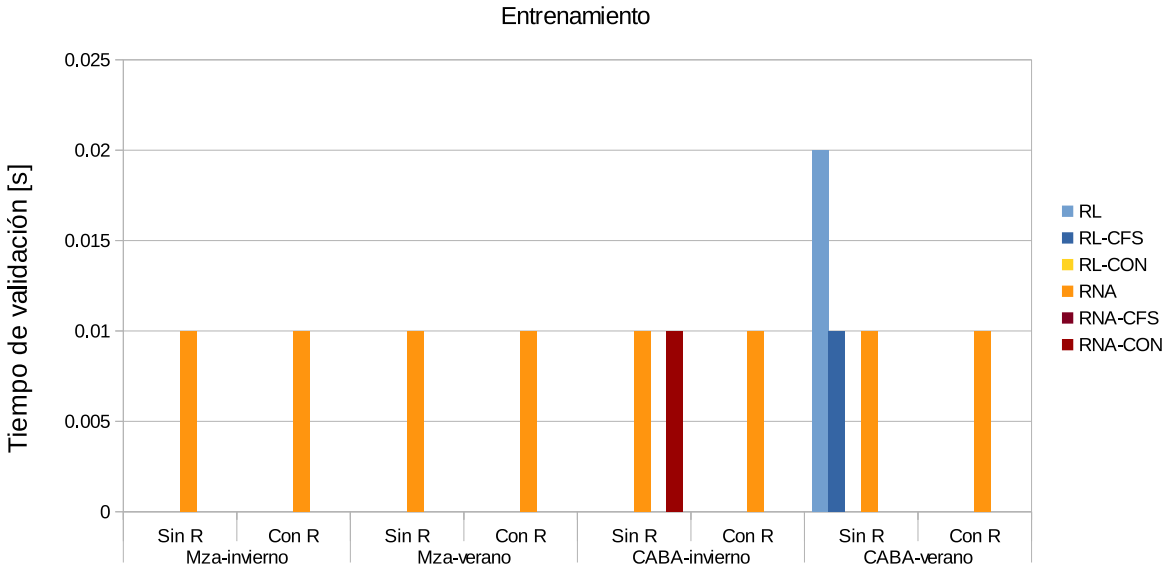
El peor caso de TFN corresponde al modelo RL-CON sin remuestreo aplicado a los datos correspondientes a CABA-verano.



Tiempo de entrenam.		RL	RL-CFS	RL-CON	RNA	RNA-CFS	RNA-CON
Mza-invierno	Sin R	0.85	**0.14**	13.48	42.71	5.45	32.95
	Con R	1.21	0.19	9.08	59.83	7.8	27.48
Mza-verano	Sin R	0.82	0.23	9.4	42.02	10.16	27.85
	Con R	1.4	0.62	9.15	69.91	26.72	29.21
CABA-invierno	Sin R	0.86	0.29	10.33	42.45	12.86	28.09
	Con R	1.47	0.44	8.09	**75.6**	17.34	27.85
CABA-verano	Sin R	0.87	0.24	0.97	41.89	9.25	**2.93**
	Con R	1.28	0.41	**13.94**	62.52	18.46	43.41

Figura 5.2.5: Gráfica de barras de los menores valores de tiempos de entrenamiento de los modelos aplicados en datos de cada ciudad y estación sin remuestreo (sin R) o con remuestreo (con R) y sin selección de atributos, con CFS o con CON. Debajo, los valores en una tabla, donde se exponen en negrita los mayores y menores entre las distintas configuraciones de RL y entre las de RNA.

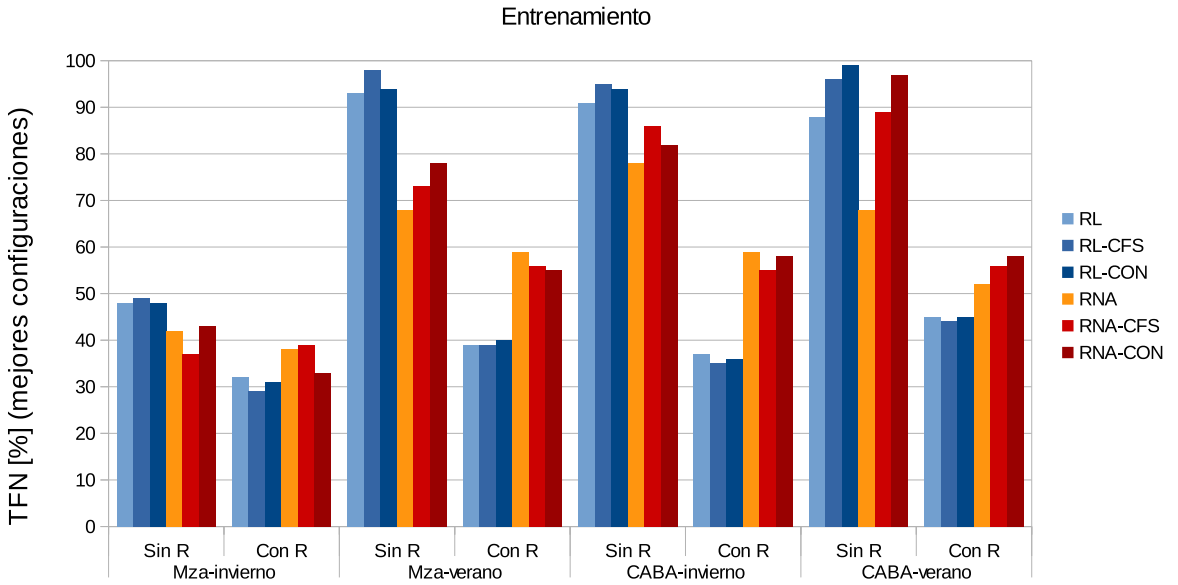
En la Figura 5.2.5 se destacan los segundos que pasaron durante el entrenamiento de las RNA, aunque no superan los 76 s y la mayoría se encuentra por debajo de los 18 s. El mejor tiempo de entrenamiento se logró sin remuestreo con RL-CFS sobre los datos de Mza-invierno.



Tiempo de validación		RL	RL-CFS	RL-CON	RNA	RNA-CFS	RNA-CON
Mza-invierno	Sin R	0	0	0	0.01	0	0
	Con R	0	0	0	0.01	0	0
Mza-verano	Sin R	0	0	0	0.01	0	0
	Con R	0	0	0	0.01	0	0
CABA-invierno	Sin R	0	0	0	0.01	0	0.01
	Con R	0	0	0	0.01	0	0
CABA-verano	Sin R	0.02	0.01	0	0.01	0	0
	Con R	0	0	0	0.01	0	0

Figura 5.2.6: Gráfica de barras de los menores valores de tiempos de validación de los modelos aplicados en datos de cada ciudad y estación sin remuestreo (sin R) o con remuestreo (con R) y sin selección de atributos, con CFS o con CON. Debajo, los valores en una tabla, donde se exponen en negrita los mayores y menores entre las distintas configuraciones de RL y entre las de RNA.

Observando la Figura 5.2.6, de métricas obtenidas en los entrenamientos de todos los modelos sobre todos los conjuntos de datos, exceptuando los correspondientes al año 2011, resalta el escaso tiempo utilizado por la máquina en el momento de validar; demorando entre 0.02 s y 0.00 s.



TFN (mejores)		RL	RL-CFS	RL-CON	RNA	RNA-CFS	RNA-CON
Mza-invierno	Sin R	48	49	48	42	37	43
	Con R	32	**29**	31	38	39	**33**
Mza-verano	Sin R	93	98	94	68	73	78
	Con R	39	39	40	59	56	55
CABA-invierno	Sin R	91	95	94	78	86	82
	Con R	37	35	36	59	55	58
CABA-verano	Sin R	88	96	**99**	68	89	**97**
	Con R	45	44	45	54	56	58

Figura 5.2.7: Gráfica de barras de los porcentajes de tasas de falsos negativos de los modelos con las configuraciones que resultaron con mayor tasa de aciertos aplicados en datos de cada ciudad y estación sin remuestreo (sin R) o con remuestreo (con R) y sin selección de atributos, con CFS o con CON. Debajo, los valores en una tabla, donde se exponen en negrita los mayores y menores entre las distintas configuraciones de RL y entre las de RNA.

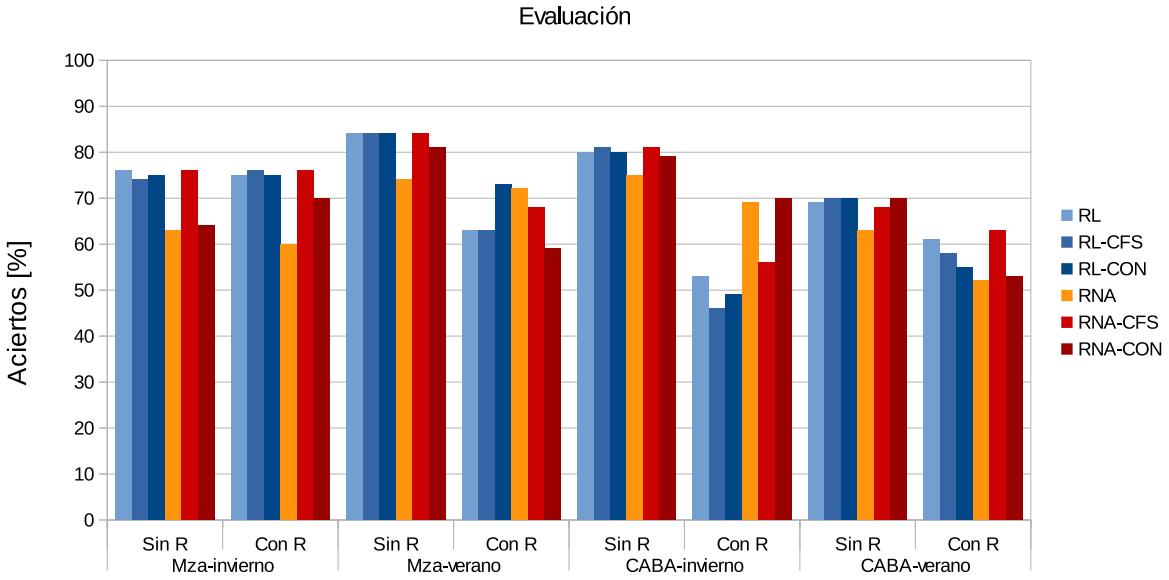
En esta última gráfica de la subsección, Figura 5.2.7, donde se grafican los porcentajes de TFN de los 8 modelos con las configuraciones que resultaron con mayor tasa de aciertos, percibimos que sus porcentajes están entre 29 % y 99 % al igual que los mejores porcentajes de TFN entre los 60 modelos (Figura 5.2.4), ya que las configuraciones de los modelos RL son las mismas y las de las RNA no superaron dichos extremos (llegando en un solo caso al 39 %; la mayoría está por debajo del 24 %).

Así mismo notamos que hay una gran diferencia entre los porcentajes obtenidos al balancear y no balancear las clases de modelos RL; llegando también al 60 % en el mismo caso (utilizando remuestreo sobre CABA-invierno con RL-CFS). Y de la misma manera, en los casos en que se aplica RNA la diferencia no es tan grande.

Por lo tanto, al igual que en la Figura 5.2.4, el peor caso se observa sobre el modelo RL-CON sin remuestreo aplicado a los datos correspondientes a CABA-verano.

5.3. Evaluación de los Modelos

En la presente sección, entrenamos nuevamente los 48 modelos con las parametrizaciones que resultaron con el mayor porcentaje de aciertos de los 60 modelos explorados (Sección 5.2), utilizando todos los datos excepto los correspondientes al año 2011. Y medimos el desempeño de éstos sobre el conjunto de prueba, es decir, sobre las precipitaciones correspondientes al año 2011 proporcionadas por CMORPH. A continuación, en la Figura 5.3.1 y la Figura 5.3.2, mostramos los resultados de las métricas (porcentaje de aciertos y TFN) utilizadas para tal fin.



Evaluación							
Aciertos		RL	RL-CFS	RL-CON	RNA	RNA-CFS	RNA-CON
Mza-invierno	Sin R	76.09	73.64	75.27	63.04	75.54	63.59
	Con R	74.73	75.54	75.44	60.32	75.54	69.57
Mza-verano	Sin R	**84.17**	83.89	83.89	74.44	**83.89**	81.11
	Con R	62.78	62.5	72.5	71.67	68.06	58.61
CABA-invierno	Sin R	80.16	80.71	80.43	75	80.71	79.08
	Con R	52.72	**45.92**	49.18	68.75	55.98	59.84
CABA-verano	Sin R	69.44	69.72	69.72	63.05	67.5	69.72
	Con R	61.11	57.5	54.72	**52.22**	63.33	53.33

Figura 5.3.1: Gráfica de barras de los mayores porcentajes de aciertos de los modelos *evaluados* en datos de cada ciudad y estación sin remuestreo (sin R) o con remuestreo (con R) y sin selección de atributos, con CFS o con CON. Debajo, los valores en una tabla, donde se exponen en negrita los mayores y menores entre las distintas configuraciones de RL y entre las de RNA.

En esta figura vemos que los mayores porcentajes de aciertos de los 48 modelos evaluados están entre 52% y 85%. Y los mayores porcentajes, entre 70% y 84%, son aquellos correspondientes a los 10 modelos (6 con CFS, 2 con CON y 2 sin selección de atributos) cuyas configuraciones resumimos en la Cuadro 5.6 y la Cuadro 5.7. Podemos ver también, tanto en la figura como en los cuadros, que el máximo se obtiene con RL aplicado a los datos de Mza-verano.

Cuadro 5.6: Configuraciones con mayores porcentajes de aciertos para los modelos RL evaluados.

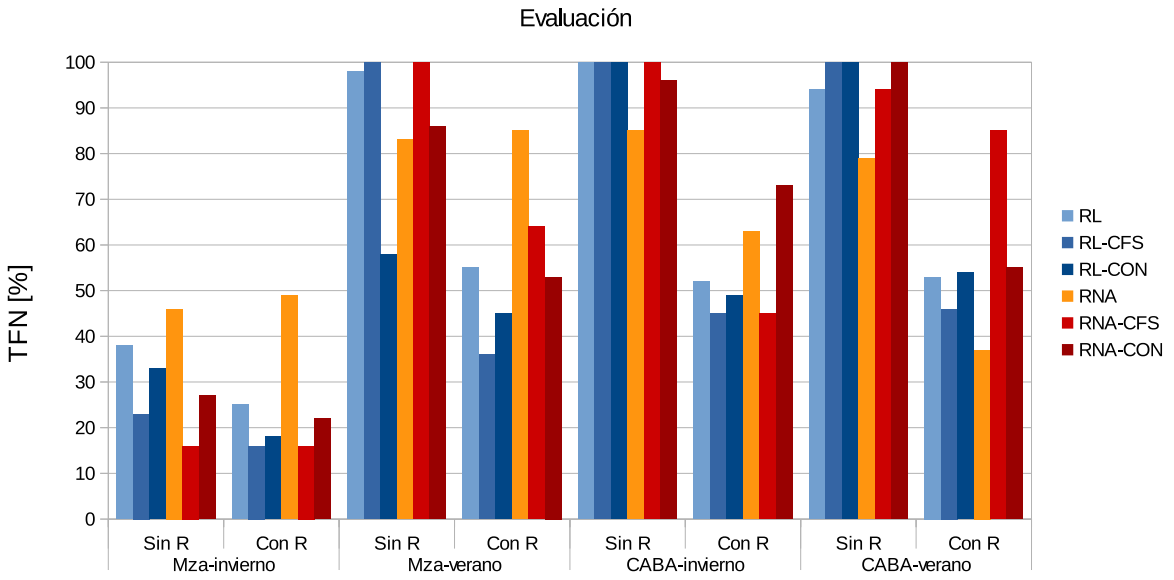
Ciudad	Estación	Balance de clases	Selección atributos	Aciertos [%]
Mza	invierno	no	no	76.02
	verano	no	no	84.17
CABA	invierno	no	CFS	81.71
	verano	no	CFS	69.72
		no	CON	

Cuadro 5.7: Configuraciones con mayores porcentajes de aciertos para los modelos RNA evaluados.

Ciudad	Estación	Balance de clases	Selección atributos	N'	t	Aciertos [%]
Mza	invierno	sí	CFS	10	500	75.54
		no	CFS	30	1000	
	verano	no	CFS	10	500	83.89
CABA	invierno	no	CFS	10	500	80.71
	verano	no	CON	10	500	69.72

Además, si bien no por mucha diferencia de valores de porcentajes (sólo en el caso de CABA-invierno con remuestreo entre RL y RNA llega al 16% la diferencia), al comparar los modelos RL con RNA, RL-CFS con RNA-CFS y RL-CON con RNA-CON, existe una diferencia de 8 casos favorables para configuraciones correspondientes a RL (el total es 18 para RL y 10 para RNA; resultando el mismo porcentaje en 4 casos).

El peor caso de aciertos corresponde al modelo RL-CFS con remuestreo aplicado a los datos correspondientes a CABA-invierno.



Evaluación							
TFN		RL	RL-CFS	RL-CON	RNA	RNA-CFS	RNA-CON
Mza-invierno	Sin R	38	23	33	46	**16**	27
	Con R	25	**16**	18	49	**16**	22
Mza-verano	Sin R	98	**100**	58	83	**100**	86
	Con R	55	36	45	85	64	53
CABA-invierno	Sin R	**100**	**100**	**100**	85	**100**	96
	Con R	52	45	49	63	45	73
CABA-verano	Sin R	94	**100**	**100**	79	94	**100**
	Con R	53	46	54	37	85	55

Figura 5.3.2: Gráfica de barras de los menores porcentajes de TFN de los modelos *evaluados* en datos de cada ciudad y estación sin remuestreo (sin R) o con remuestreo (con R) y sin selección de atributos, con CFS o con CON. Debajo, los valores en una tabla, donde se exponen en negrita los mayores y menores entre las distintas configuraciones de RL y entre las de RNA.

En esta otra figura de arriba notamos que los menores porcentajes de TFN de los modelos evaluados son del 16 %. También que para los datos de Mza-invierno todos se encuentran debajo del 50 %. En el caso de los otros tres pares ciudad-estación, se llega al 100 % de TFN.

Por otro lado hicimos las simulaciones utilizando el modelo WRF para todo el año 2011 con las configuraciones especificadas en el Capítulo 2 interpolando bilinealmente los resultados a los mismos puntos geográficos de RL y RNA donde se observan las estimaciones de CMORPH. Luego contrastamos sus predicciones con los datos de precipitación de CMORPH para tener una base justa de comparación entre WRF (Sección 2.3) y los 8 mejores modelos aprendidos (ver cuadros 5.4 y 5.5).

En el Cuadro 5.8 presentamos los porcentajes de aciertos de dichos modelos evaluados y del modelo WRF, en el Cuadro 5.9 sus porcentajes de TFN y en el Cuadro 5.10 las demoras de ejecución de los mismos. Los mejores valores son resaltados en negrita.

Cuadro 5.8: Porcentajes de aciertos para los modelos aprendidos con las parametrizaciones que resultaron con el mayor porcentaje de aciertos y para WRF.

Ciudad	Estación	RL	RNA	WRF
Mza	invierno	**76.09**	75.54	68.83
	verano	**83.89**	**83.89**	68.14
CABA	invierno	80.43	80.71	**82.84**
	verano	69.72	69.72	**84.49**

Si bien en el cuadro de aciertos se nota una cercanía de altos porcentajes (con una diferencia máxima de 15.75% para Mza-verano) en los tres modelos para cada ciudad-estación, los modelos evaluados RL y RNA son mejores para Mza, mientras WRF lo es para CABA.

Cuadro 5.9: Porcentajes de TFN para los modelos aprendidos con las parametrizaciones que resultaron con el mayor porcentaje de aciertos y para WRF.

Ciudad	Estación	RL	RNA	WRF
Mza	invierno	38	**16**	83.33
	verano	**58**	100	81.25
CABA	invierno	100	100	**58.33**
	verano	100	100	**62.79**

En este cuadro inmediatamente por encima, Cuadro 5.9, existe una gran diferencia entre los porcentajes de TFN obtenidos por los modelos con mayores aciertos aprendidos y los obtenidos con WRF.

Cuadro 5.10: Demoras de la máquina en aprender y predecir al aplicar los modelos de AM con las parametrizaciones que resultaron con el mayor porcentaje de aciertos y en predecir aplicando WRF.

Estación	Ciudad	RL			RNA			WRF
		Apr. [s]	Pred. [s]	Total [s]	Apr. [s]	Pred. [s]	Total [s]	[días]
invierno	Mza	0.85	1.12	**1.97**	7.8	35.82	43.62	4
	CABA	10.33	17.37	**27.7**	12.86	16.93	29.79	
verano	Mza	9.4	12.29	21.69	10.16	10.92	**21.08**	3
	CABA	0.25	0.79	**1.04**	2.93	0.39	3.32	

En este último cuadro puede apreciarse la gran diferencia sustancial respecto a tiempos al predecir con modelos de AM, en lugar de hacerlo con WRF. Los modelos de AM predicen en segundos (menos de medio minuto), mientras que WRF lo hace en días.

Concluyendo, al tener en cuenta todo lo analizado a partir de los resultados expuestos, los modelos aprendidos tienen mejor desempeño que WRF al aplicarse sobre datos de Mza. Y

para datos correspondientes a dicha ciudad en los meses considerados invierno en este estudio (junio, julio, agosto), es mejor hacer previamente una selección de atributos mediante la técnica CFS, antes que realizarla con la técnica CON o no hacerla (ya sea aplicando RL o RNA, con o sin remuestreo); ya que si bien no se obtuvo con dicha configuración el mejor porcentaje de aciertos (76.09 %), se obtuvo uno muy próximo (75.54 %) y el mínimo porcentaje de TFN (16 %). En cambio, para datos correspondientes a los meses considerados verano en este estudio (diciembre, enero, febrero), es mejor aplicar RL y hacer previamente una selección de atributos mediante la técnica CON, antes que las otras posibilidades experimentadas.

Es muy importante destacar, que los modelos aprendidos, tienen porcentajes de aciertos superiores al 75 % y porcentaje de TFN de 16 % en el caso de RL con CFS y con datos cuyas clases fueron balanceadas mediante remuestreo, y en el caso de las RNA con CFS (tanto con datos balanceados mediante remuestreo, como con los datos desbalanceados) con 10 nodos escondidos y tiempo de entrenamiento 500, y también con 30 nodos escondidos y tiempo de entrenamiento 1000.

En el párrafo anterior, se nombran dos estructuras distintas de RNA. Pero como la segunda mencionada tiene mayor cantidad de nodos y mayor tiempo de entrenamiento, la primera tendrá mejor rendimiento, ya que si bien obtienen la misma cantidad de aciertos, con la primera la máquina se demorará menos en computar.

También es muy importante resaltar, que, en cambio, los porcentajes de TFN que resultaron de aplicar el modelo NWP de simulación más utilizado hoy en día, WRF, son muy elevados en ambos estaciones de CABA en los que los porcentajes de aciertos son mejores que los dos modelos aprendidos. Por lo tanto, si bien obtuvimos mejores porcentajes de aciertos con WRF sobre datos de CABA, el desempeño no es bueno, ya que los porcentajes de TFN son muy altos para considerarlos como un buen modelo para predecir precipitación porque la probabilidad de evitar daños socio-económicos es muy baja (pensando en el costo que podría llegar a significar). Una situación concreta en la que se producirían dichos daños, sería cuando se anuncie que no habrá precipitación en CABA y sin embrago luego se produjera una tormenta que inundase gran parte de la ciudad ocasionando pérdidas de vida e infraestructura (resultando un gran costo socio-económico tanto para la provincia como para el país).

Además es fundamental notar que en la evaluación, resultan considerablemente menores los valores de TFN obtenidos al hacer previamente remuestreo.

5.4. Conclusiones

La eficacia para predecir correctamente la ocurrencia de precipitación en Mza y CABA con condiciones climatológicas claramente diferenciadas al aplicar distintas configuraciones diseñadas

mediante técnicas de AM y selección de modelos RL y RNA fue demostrada en este capítulo, donde la limitación de predicción espacial y temporal con la que se contaba fue superada en parte.

Dicha eficacia se vio reflejada en la aplicabilidad sobre datos históricos de Mza de los meses considerados como invierno (junio, julio y agosto), no así en el caso de CABA. Si bien en el caso de esta última ciudad mencionada la tasa de aciertos en la predicción fue alta, la tasa de falsos negativos fue tan alta que su aplicación es inaceptable como recurso de alerta, ya que puede llegar a ocasionar una catástrofe ante la falta de alarma.

Logramos además en este segundo estudio reportado tiempos de ejecución mucho menores que los tiempos obtenidos por WRF para resolver el problema planteado, probando que sólo la etapa en la que se aprende el modelo a partir de datos meteorológicos es computacionalmente costosa aunque muy baja.

Capítulo 6

Conclusiones

6.1. Conclusiones

Con el presente trabajo demostramos experimentalmente la aplicabilidad operacional de AM automáticamente mediante RL y RNA para predecir ocurrencia de precipitación en Mendoza y CABA a partir de datos históricos de 14 años comprendiendo desde el año 2000 al año 2014, brindando un método operacional novedoso para predicción de ocurrencia de precipitación en la zona central argentina sin antecedentes en el país.

Con el primer estudio realizado (Estudio I) en el Capítulo 4 proveímos la estructura específica de los datos históricos considerados y de los atributos de entrada en el proceso de aprendizaje de los modelos RL y RNA, resultando ser estos últimos: hora, día, mes y año del evento de precipitación (ocurrencia o no), temperatura a 500 hPa, humedad relativa a 850 hPa, vorticidad absoluta a 500 hPa, divergencia a 200 hPa, viento zonal a 850 hPa, viento meridional a 850 hPa, presión, CAPE, CIN, Niño anual y Niño mensual.

En el segundo estudio (Estudio II) en el Capítulo 5 demostramos la eficacia de configuraciones diseñadas mediante selección de modelos RL y RNA y técnicas de AM, donde se aprendieron y validaron los modelos con todos los datos a excepción de los correspondientes al año 2011, y luego se evaluaron los mismos con los datos correspondientes a dicho año con las configuraciones de mejor desempeño según porcentaje de aciertos y de TFN en el aprendizaje, comparándolos con los resultados del modelo de simulación NWP-WRF aplicado a los mismos datos históricos. De esta manera, logramos evaluar el desempeño de los modelos en su capacidad para predecir correctamente la ocurrencia de precipitación en Mza y CABA con condiciones climatológicas claramente diferenciadas.

Particularmente, según los valores de las métricas aciertos y tasa de falsos negativos, el desempeño es muy bueno en el caso de aplicar los modelos aprendidos RL y RNA sobre los datos de

Mza-invierno considerados en este estudio. Demostramos que los modelos con mejor desempeño (predictivo) y rendimiento (respecto a costo computacional) entre los 6 de RL, los 54 de RNA y WRF, son: RL con selección de atributos por la técnica CFS y previo balance de calses mediante remuestreo, tardando 1.97 s (tasa de aciertos: 76.09 %; tasa de falsos negativos: 38 %); y RNA de 10 nodos escondidos y tiempo de entrenamiento 500, con selección de atributos CFS y también realizando previamente remuestreo, tardando 43.62 s (tasa de aciertos: 75.54 %; tasa de falsos negativos: 16 %).

Vale recordar que RL es un modelo simple en comparación a RNA, y a su vez la RNA de 10 nodos y tiempo de entrenamiento 500 es la de arquitectura más simple entre las configuraciones de diseño de redes consideradas en este trabajo. Por lo que se logra un fundamento aún más sólido para la preferencia de los modelos de AM: al aplicar los modelos aprendidos de mejor desempeño, no sólo logramos la mejor precisión, sino también los menores tiempos de ejecución.

Así, logramos además en esta investigación reportada tiempos de ejecución mucho menores que los tiempos obtenidos por WRF, probando a la vez que sólo la etapa en la que se aprende el modelo a partir de datos meteorológicos es computacionalmente costosa aunque muy baja; evidenciando que al aplicar WRF, cada predicción realizada exigiría un gasto computacional, convirtiéndose en un gasto considerablemente mucho mayor a aquel obtenido al aplicar uno de los algoritmos aprendidos.

Respecto al desempeño de los modelos utilizando y sin utilizar remuestreo, no hubo notable diferencia en los porcentajes de aciertos, pero sí la hubo en los de TFN. Esto y el hecho de que ambos modelos que resultaron tener el mejor desempeño y rendimiento fueron aplicados sobre los datos balanceados previamente con remuestreo, muestra que es mejor realizar remuestreo. Como se esperaba, al estar las clases *llueve* y *no-llueve* muy desbalanceadas (en los cuatro pares ciudad-estación) resulta difícil para los modelos aprender los patrones que explican los casos de la clase minoritaria (clase *llueve*, en los cuatro pares).

También tuvo sentido haber hecho selección automática de atributos, ya que el método CFS resultó ser el más exitoso entre los tres parámetros considerados.

En lo que nos concierne de la aplicabilidad de WRF, éste resultó con mayores porcentajes de aciertos que los modelos de AM estudiados al predecir precipitación en CABA en el período investigado, pero los porcentajes de TFN que se obtienen con dicho modelo de simulación son muy altos para poder considerarlo como un buen modelo para predecir precipitación. Sin embargo, en el caso de los modelos RL y RNA sobre datos de esta ciudad, los porcentajes de TFN fueron definitivamente peores: 100 %. Por lo tanto, con tal desenlace, no podemos juzgar aún alguno de los tres modelos como buen predictor de precipitaciones en CABA, ni en verano ni en invierno.

Concluimos entonces que para superar y mejorar el desempeño y rendimiento de los modelos de AM estudiados deben aplicarse aquellos modelos diseñados que obtuvieron porcentaje de acierto menor, aunque sólo por pocas unidades (ya que todos los porcentajes de aciertos fueron próximos entre sí), pero también mucho menor porcentaje de TFN. Para predecir ocurrencia de precipitación en Mza-verano sería el modelo RL con CFS con remuestreo; en CABA-invierno el mismo o RNA con CFS con remuestreo y en CABA-verano también RL con CFS con remuestreo o RNA sin selección de atributos y con remuestreo. De esta manera mejoraría en todos los casos el desempeño en sus capacidades predictivas y disminuiría el riesgo de predecir que no lloverá cuando dicha situación podría ocurrir y desencadenar alguna catástrofe. Lo cual es de gran interés para evitar daños socioeconómicos severos en el país.

Cabe mencionar también que sobre los porcentajes de TFN hubo una diferencia notable entre los obtenidos en el entrenamiento y aquellos obtenidos en la evaluación, siendo los del entrenamiento menores. Esto puede atribuirse a un sobreajuste en el caso de CABA, i.e. a que los modelos se ajustaron demasiado a los conjuntos ejemplo (en el aprendizaje) y no son capaces de predecir correctamente cuando se aplican a nuevos datos (en la evaluación).

6.2. Limitaciones

Desde el punto de vista meteorológico, la principal limitación es la falta de datos brindados por las estaciones meteorológicas del país y la falta de estaciones meteorológicas en el país que recopilen regular y homogéneamente datos necesarios (e.g ocurrencia de precipitación, cantidad de precipitación, variables meteorológicas).

Además, dicha limitación experimentada puede ser la causante del sobreajuste atribuido a los altos porcentajes de TFN del desempeño de los modelos de AM en puntos geográficos y períodos determinados.

6.3. Trabajo futuro

La variabilidad geográfica y temporal en estudio se anticipa a la posibilidad de diseñar un esquema para que la aplicación del modelo propuesto pueda trasladarse a zonas climatológicamente similares a las ciudades mencionadas (dentro y fuera de Argentina), abarcando así áreas con condiciones extremas: valle árido con altos cordones montañosos y llano húmedo limítrofe al mar.

Para ello, es importante averiguar la razón de por qué en Mza y no en CABA funcionaron mejor los modelos aprendidos; si realmente es debido a la topografía, o influyen más las condiciones meteorológicas.

Por otro lado, desde un enfoque más específico, el trabajo inmediato a llevar a cabo es respecto al diseño de las RNA. Aquellas de una capa oculta son más simples que muchas otras, al igual que la función logística lo es de otras funciones. Existen RNA con más de una capa oculta y funciones de activación de cómputo más complejas (e.g. tangente hiperbólica), que logran identificar mejor el patrón causa-consecuencia entre entradas y salidas. Por lo tanto, su aplicación podría superar el desempeño y/o rendimiento que logramos en el presente trabajo. Pero no agregaríamos nodos ocultos ni aumentaríamos el tiempo de entrenamiento, ya que esto no mejoró el desempeño, pero sí disminuiría el rendimiento debido a que la ejecución demoraría más.

Con el nuevo diseño, sería importante investigar la causa de TFN elevados de los modelos aprendidos al aplicarlos en Mza-invierno (ya que fueron los mejores modelos entre todos los estudiados con porcentajes cercanos al 15%).

Para ello, en lugar de CMORPH podrían usarse datos de estaciones meteorológicas ya que CMORPH es simplemente una estimación de precipitaciones reales, por lo que puede ser una fuente mayor de errores.

Además, una tarea conveniente sería realizar alguna técnica de filtrado de ruido sobre el Análisis de Wavelets obtenido y después restar dichas señales filtradas a las ondas de señales no filtradas, para mejorar la precisión del análisis, y de esta forma, por las conclusiones de éste, el desempeño de los algoritmos aprendidos.

Otra tarea conveniente sería eliminar variables consideradas como entrada en el caso de CABA según la correlación observada en los mapas de calor, ya que quizá sean redundantes y su eliminación favorecería el rendimiento de los modelos, tanto de AM como de WRF. En otras palabras, quizá no tengan tanta relevancia y se esperaría que disminuyera el gasto computacional sin aumentar predicciones incorrectas.

También, nos preguntaríamos si en el proceso de aprendizaje de los modelos se produjo un sobreajuste. Esto se responde analizando la evolución de errores de entrenamiento y de evaluación, y experimentando según resultados.

Por último mencionaremos que este trabajo y el futuro serían trabajos preliminares para una exploración de predicción cuantitativa de precipitación. Primero se estudiaría con tres clases, según mm/h acumulados de agua; por ejemplo *no severo*, *moderado* y *severo*. Y después se intentaría en dirección a la predicción de la cantidad de precipitación. Predecir la cantidad de precipitación es además un problema más complejo y podría requerir más o mejores datos, e incluso mayor capacidad de cómputo para aprender modelos precisos.

Bibliografía

[1] Roger A. Pielke Sr. Atoc 7500: Mesoscale meteorological modeling, Spring 2008. 8, 21

[2] V. BJERKNES. Das problem der wettervorhers-age, betrachtet vom standpunkte der mechanik und der physik. *Meteor. Z.*, 21:1–7, 1904. 13

[3] Jorge Rubén Santos, Federico Norte, Stella Moreiras, Diego Araneo, and Silvia Simonelli. Prediccion de episodios de precipitacion que ocasionan aludes en el area montañosa del noroeste de la provincia de mendoza, argentina. *Geoacta*, 40(1):65–75, 2015. 13, 14

[4] Frederick G Shuman. History of numerical weather prediction at the national meteorological center. *Weather and Forecasting*, 4(3):286–296, 1989. 13

[5] W. C. Skamarock, J. B. Klemp, J. Dudhia, D. O. Gill, D. M. Barker, M. G Duda, X.-Y. Huang, W. Wang, and J. G. Powers. A description of the advanced research wrf version 3. 2008. NCAR Tech. Note NCAR/TN-475+STR, 113 pp. 14, 17

[6] Kao-San Yeh, Jean Côté, Sylvie Gravel, André Méthot, Alaine Patoine, Michel Roch, and Andrew Staniforth. The cmc–mrb global environmental multiscale (gem) model. part iii: Nonhydrostatic formulation. *Monthly Weather Review*, 130(2):339–356, 2002. 14

[7] A.J. Simmons and D. Dent. The ecmwf multi-tasking weather prediction model. *Computer Physics Reports*, 11(1):165 – 194, 1989. 14

[8] Fedor Mesinger, Sin Chan Chou, Jorge L. Gomes, Dusan Jovic, Paulo Bastos, Josiane F. Bustamante, Lazar Lazic, André A. Lyra, Sandra Morelli, Ivan Ristic, and Katarina Veljovic. An upgraded version of the eta model. *Meteorology and Atmospheric Physics*, 116(3):63–79, May 2012. 14

[9] Yue Zheng, Kiran Alapaty, Jerold A Herwehe, Anthony D Del Genio, and Dev Niyogi. Improving high-resolution weather forecasts using the weather research and forecasting

(wrf) model with an updated kain–fritsch scheme. *Monthly Weather Review*, 144(3):833–860, 2016. 14

[10] Joon-Bum Jee and Sangil Kim. Sensitivity study on high-resolution wrf precipitation forecast for a heavy rainfall event. *Atmosphere*, 8(6):96, 2017. 14

[11] J.M. Gutiérrez, R. Cano, A.S. Cofi no, and C. Sordo. *Redes Probabilísticas y Neuronales en las Ciencias Atmosféricas*. Instituto Nacional de Meteorología, Ministerio de Medio Ambiente, Madrid (2004), 2004. 14

[12] Warren S McCulloch and Walter Pitts. A logical calculus of the ideas immanent in nervous activity. *The bulletin of mathematical biophysics*, 5(4):115–133, 1943. 14

[13] Arthur L. Samuel. Some studies in machine learning using the game of checkers. *IBM JOURNAL OF RESEARCH AND DEVELOPMENT*, pages 71–105, 1959. 14, 26

[14] William W Hsieh and Benyang Tang. Applying neural network models to prediction and data analysis in meteorology and oceanography. *Bulletin of the American Meteorological Society*, 79(9):1855–1870, 1998. 15

[15] Benyang Tang, William W Hsieh, Adam H Monahan, and Fredolin T Tangang. Skill comparisons between neural networks and canonical correlation analysis in predicting the equatorial pacific sea surface temperatures. *Journal of Climate*, 13(1):287–293, 2000. 15, 43

[16] F. Mekanik, M. A. Imteaz, S. Gato-Trinidad, and A. Elmahdi. Multiple regression and Artificial Neural Network for long-term rainfall forecasting using large scale climate modes. *Journal of Hydrology*, 503:11–21, 2013. 15

[17] P. T. Nastos, K. P. Moustris, I. K. Larissi, and A. G. Paliatsos. Rain intensity forecast using Artificial Neural Networks in Athens, Greece. *Atmospheric Research*, 119:153–160, 2013. 15

[18] Nachiketa Acharya, Surajit Chattopadhyay, Makarand Kulkarni, and Uma Mohanty. A neurocomputing approach to predict monsoon rainfall in monthly scale using sst anomaly as a predictor. *Acta Geophysica*, 60(1):260–279, 2012. 15

[19] John Abbot and Jennifer Marohasy. Input selection and optimisation for monthly rainfall forecasting in queensland, australia, using artificial neural networks. *Atmospheric Research*, 138:166–178, 2014. 15

[20] Diana Analía Domínguez and Marcela Hebe González. Variabilidad de la precipitación en el centro oeste de argentina y un modelo de predicción estadística. *Meteorologica*, 38(2):105–120, 2013. 15

[21] National Centers for Environmental Prediction, National Weather Service, NOAA, and U.S. Department of Commerce. Ncep fnl operational model global tropospheric analyses, continuing from july 1999, 2000. 22, 42

[22] Robert J. Joyce, John E. Janowiak, Phillip A. Arkin, and Pingping Xie. CMORPH: A method that produces global precipitation estimates from passive microwave and infrared data at high spatial and temporal resolution. *Journal of Hydrometeorology*, 5(3):487–503, 2004. 24, 42

[23] Peter Flach. *Machine learning: the art and science of algorithms that make sense of data*. Cambridge University Press, 2012. 25, 31

[24] Tom M Mitchell. *Machine Learning*. McGraw-Hill Science/Engineering/Math, March 1997. page 2. 26

[25] Juan Carrasquilla and Roger G. Melko. Machine learning phases of matter. *Nature Physics*, 13(431), January 2017. 26

[26] Johan Bollen, Huina Mao, and Xiaojun Zeng. Twitter mood predicts the stock market. *Journal of Computational Science*, 2(1):1 – 8, 2011. 26

[27] John Laird and Michael VanLent. Human-level ai's killer application: Interactive computer games. *AI magazine*, 22(2):15, 2001. 26

[28] Tom M Mitchell. *Machine Learning*. McGraw-Hill Science/Engineering/Math, March 1997. 27

[29] Svetlana S. Petrova and Alexander D. Solov'ev. The origin of the method of steepest descent. *Historia Mathematica*, 24(4):361 – 375, 1997. 29

[30] James A Freeman and David M Skapura. *Algorithms, Applications, and Programming Techniques*. Addison-Wesley Publishing Company, USA, 1991. 35

[31] Raúl Rojas. Neural networks: a systematic introduction, 2013. 35

[32] Donald Olding Hebb. The organization of behavior; a neuropsycholocigal theory. *A Wiley Book in Clinical Psychology.*, pages 62–78, 1949. 36

[33] Richard Lippmann. An introduction to computing with neural nets. *IEEE Assp magazine*, 4(2):4–22, 1987. 36

[34] Frank Rosenblatt. The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological review*, 65(6):386, 1958. 38

[35] Christopher M Bishop. *Neural networks for pattern recognition*. Oxford university press, 1995. 38

[36] David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams. Learning representations by back-propagating errors. *nature*, 323(6088):533, 1986. 38

[37] Jose Miguel Hernandez-Lobato and Ryan Adams. Probabilistic backpropagation for scalable learning of bayesian neural networks. In Francis Bach and David Blei, editors, *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pages 1861–1869, Lille, France, 07–09 Jul 2015. PMLR. 38

[38] Yaroslav Ganin and Victor Lempitsky. Unsupervised domain adaptation by backpropagation. In Francis Bach and David Blei, editors, *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pages 1180–1189, Lille, France, 07–09 Jul 2015. PMLR. 38

[39] Yonas B Dibike and Paulin Coulibaly. Temporal neural networks for downscaling climate variability and extremes. *Neural Networks*, 19(2):135–144, 2006. 43

[40] Kevin & National Center for Atmospheric Research Staff Trenberth. The climate data guide: Nino sst indices (nino 1+2, 3, 3.4, 4; oni and tni). Retrieved from https://climatedataguide.ucar.edu/climate-data/nino-sst-indices-nino-12-3-34-4-oni-and-tni. Last modified 02 Feb 2016. 43

[41] Ian H. Witten, Eibe Frank, and Mark A. Hall. *Data Mining: Practical Machine Learning Tools and Techniques*. Morgan Kaufman, 3rd edition, January 2011. pages 39-60. 43

[42] Christopher Torrence and Gilbert P Compo. A practical guide to wavelet analisys. *Bulletin of the American Meteorological Society*, 1998. 44

[43] Christopher Torrence and Peter J Webster. The annual cycle of persistence in the el nño/-southern oscillation. *Quarterly Journal of the Royal Meteorological Society*, 124(550):1985–2004, 1998. 45

[44] Mandrilli P. A., Monge D. A., Catania C. A., and Santos J. R. Técnicas de aprendizaje de máquinas para predicción de precipitación. Tucumán, Argentina., Octubre 2016. 101Â° Reunión de la Asociación Física Argentina., Asociación Física Argentina. 46

[45] Mark Hall, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann, and Ian H. Witten. The weka data mining software: An update. *SIGKDD Explor. Newsl.*, 11(1):10–18, November 2009. 50

[46] A Rubi Bianchi and Silvia Ana Carla Cravero. Atlas climático digital de la república argentina. *INTA Ediciones*, 2010. 51

[47] M. A. Hall. *Correlation-based Feature Subset Selection for Machine Learning*. PhD thesis, University of Waikato, Hamilton, New Zealand, 1998. 52

[48] H. Liu and R. Setiono. A probabilistic approach to feature selection - a filter solution. In *13th International Conference on Machine Learning*, pages 319–327, 1996. 52

[49] Ron Kohavi et al. A study of cross-validation and bootstrap for accuracy estimation and model selection. In *Ijcai*, volume 14, pages 1137–1145. Montreal, Canada, 1995. 55