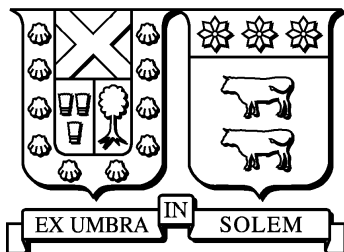


UNIVERSIDAD TÉCNICA FEDERICO SANTA MARÍA

DEPARTAMENTO DE INFORMÁTICA

SANTIAGO – CHILE



“DISEÑO, IMPLEMENTACIÓN Y VALIDACIÓN
DE UN MODELO PREDICTIVO DE LA
CONCENTRACIÓN DE OZONO TROPOSFÉRICO
EN SANTIAGO BASADO EN MÉTODOS DE
APRENDIZAJE AUTOMÁTICO”

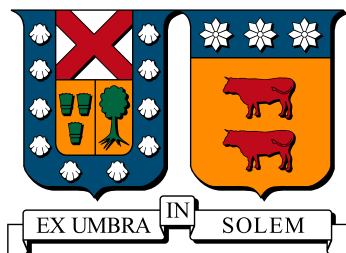
IVÁN EDUARDO GONZÁLEZ LÓPEZ

MEMORIA DE TITULACIÓN PARA OPTAR AL TÍTULO DE
INGENIERO CIVIL INFORMÁTICO

PROFESOR GUÍA: RICARDO ÑANCULEF

NOVIEMBRE 2017

UNIVERSIDAD TÉCNICA FEDERICO SANTA MARÍA
DEPARTAMENTO DE INFORMÁTICA
SANTIAGO – CHILE



**“DISEÑO, IMPLEMENTACIÓN Y VALIDACIÓN
DE UN MODELO PREDICTIVO DE LA
CONCENTRACIÓN DE OZONO
TROPOSFÉRICO EN SANTIAGO BASADO EN
MÉTODOS DE APRENDIZAJE AUTOMÁTICO”**

IVÁN EDUARDO GONZÁLEZ LÓPEZ

**MEMORIA DE TITULACIÓN PARA OPTAR AL TÍTULO DE
INGENIERO CIVIL INFORMÁTICO**

PROFESOR GUÍA: RICARDO ÑANCULEF

PROFESOR CORREFERENTE: MARÍA DOMINGUEZ

NOVIEMBRE 2017

Resumen

A pesar de que el Ozono troposférico se encuentra de forma natural en el aire respirable, en niveles más altos que los que se establecen en las normas nacionales vigentes, origina serios problemas para la salud humana y el entorno natural. La situación se suele agravar en las grandes zonas urbanas, donde debido a la actividad humana, hay una mayor concentración de los precursores de este contaminante. Por lo tanto, se hace necesario contar con herramientas capaces de modelar y predecir la concentración del Ozono troposférico, con el fin de proveer la información necesaria a las autoridades correspondientes y así puedan llevar a cabo de forma oportuna, las acciones preventivas pertinentes.

Este trabajo propone un modelo de predicción para el nivel de Ozono troposférico en la ciudad de Santiago de Chile, basado en Redes Neuronales Recurrentes LSTM, utilizando como predictores diversas variables de tipo químico y/o meteorológico. Los resultados muestran que este tipo de redes alcanzan un rendimiento competitivo respecto a los obtenidos por una Regresión Lineal, Support Vector Machines (SVM), las Redes Neuronales Feed Forward y modelos ARIMA para series de tiempo.

Palabras clave — *Ozono troposférico, Predicción, Redes Neuronales Recurrentes, Long Short Term Memory*

Abstract

Although tropospheric ozone is found naturally in the breathable air, at higher levels than those established in the current national regulations, it causes serious problems for human health and the natural environment. The situation is usually aggravated in large urban areas, where due to human activity, there is a greater concentration of the precursors of this pollutant. Therefore, it is necessary to have tools capable of modeling and predicting the concentration of tropospheric ozone, in order to provide the necessary information to the corresponding authorities so that they can carry out in a timely manner the relevant preventive actions.

This work proposes a prediction model for the tropospheric ozone level in the city of Santiago de Chile, based on Recurrent Neural Networks LSTM, using as predictors various variables of chemical and meteorological type. The results show that this type of networks achieve a competitive performance with respect to those obtained by Linear Regression, SVM, Feed Forward Neural Networks and ARIMA models for time series.

Keywords — *Tropospheric Ozone, Forecasting, Recurrent Neural Networks, Long Short Term Memory*

Índice de Contenidos

Resumen	III
Abstract	IV
Índice de Contenidos	V
Lista de Tablas	IX
Lista de Figuras	XI
Acrónimos	XIII
Glosario	XV
Introducción	1
1. Marco Conceptual	4
1.1. Contexto	5
1.1.1. El Ozono	5
1.1.2. Formación del Ozono troposférico	5
1.1.3. Efectos del Ozono troposférico	7
1.1.4. Normativa y estándares en Chile	8

1.1.5.	Normativas y estándares internacionales	9
1.2.	Modelos predictivos basados en series de tiempo	9
1.2.1.	Procesos Auto regresivos (AR)	10
1.2.2.	Procesos de Media Móvil (MA)	10
1.2.3.	Procesos ARMA	11
1.2.4.	Procesos ARIMA	11
1.3.	Modelos predictivos basados en Aprendizaje Automático	11
1.3.1.	Regresión Lineal	13
1.3.2.	Support Vector Machines	14
1.3.3.	Redes Neuronales Artificiales	16
1.3.4.	Redes Neuronales Recurrentes	19
1.3.5.	Principal Component Analysis	25
1.4.	Métricas de rendimiento	26
2.	Estado del Arte	28
2.1.	Modelos	28
2.1.1.	Redes Neuronales	29
2.1.2.	Support Vector Machines	31
2.1.3.	ARIMA	31
2.2.	Aspectos Metodológicos	32
2.2.1.	Variables de Entrada y Salida	32
2.2.2.	Horizonte de Predicción	33
2.2.3.	Validación I: Manipulación de datos	34
2.2.4.	Validación II: Medidas de Rendimiento	35
2.3.	Resumen estudios analizados	37

3. Metodología y Propuesta	40
3.1. Caso de Estudio	41
3.1.1. Antecedentes	41
3.1.2. Fuentes de Datos	44
3.2. Configuración Experimental	46
3.2.1. Requerimientos de Software	46
3.2.2. Análisis Descriptivo de los Datos	48
3.2.3. Objetivos del Análisis Predictivo	56
3.2.4. Preprocesamiento de datos	57
3.3. Configuración de los modelos	60
3.3.1. Aspectos Generales	60
3.3.2. Datos de Entrada y Salida	62
3.3.3. Modelo Persistente	64
3.3.4. Regresión Lineal	64
3.3.5. Support Vector Machine	64
3.3.6. Red Neuronal Feed Forward	65
3.3.7. Redes Neuronales Recurrentes	67
3.3.8. ARIMA	70
4. Resultados Experimentales	72
4.1. Máximos diarios	73
4.1.1. Resumen general	73
4.1.2. Cantidad de <i>timesteps</i>	75
4.1.3. Redes Neuronales	76
4.1.4. Comparación con otros estudios	77
4.2. Concentración horaria	79

4.2.1. Resumen general	79
4.2.2. Cantidad de <i>timesteps</i>	81
4.2.3. Redes Neuronales	82
4.3. Disminución del número de variables de entrada	84
4.3.1. Máximos diarios	84
4.3.2. Concentración horaria	85
Conclusiones	90
Bibliografía	94

Índice de tablas

2.1. Aspectos arquitecturales y otras configuraciones de las Redes Feed Forward.	30
2.2. Modelos basados en Redes Neuronales Recurrentes.	30
2.3. Modelos basados en SVM.	31
2.4. Medidas de rendimiento empleadas en los distintos estudios analizados. . .	36
2.5. Resumen de los estudios analizados.	37
3.1. Variables de entrada de los modelos.	45
3.2. Vista parcial del dataframe para el conjunto total de datos.	57
3.3. Tamaño de los datasets de entrenamiento, validación y pruebas.	59
3.4. Parámetros optimizados para la SVR.	64
3.5. Parámetros optimizados para el modelo ARIMA.	71
4.1. Resumen máximos diarios	73
4.4. Resumen concentración horaria.	80
4.5. Resumen concentración horaria (solo máximos).	80
4.6. Resumen máximos diarios (variables meteorológicas + Ozono).	85

4.7. Resumen concentración horaria (variables meteorológicas + Ozono).	85
--	----

Índice de figuras

1.1. Distribución del Ozono en la atmósfera terrestre.	6
1.2. Ejemplo de una neurona.	17
1.3. Funciones de activación.	17
1.4. Ejemplo de Red Neuronal Feed Forward	18
1.5. Modelos de procesamiento para Redes neuronales Recurrentes	20
1.6. Red Neuronal Recurrente.	21
1.7. Red Neuronal Recurrente desenrollada a lo largo de tres <i>timesteps</i>	22
1.8. Red Neuronal Recurrente ELMAN.	23
1.9. Célula LSTM	23
3.1. Emisión de precursores químicos del Ozono troposférico en la Región Me- tropolitana	43
3.2. Estaciones de monitoreo de la calidad del aire desplegadas en la Región Me- tropolitana.	44
3.3. Serie de tiempo del Ozono troposférico.	49
3.4. Series de tiempo contaminantes atmosféricos	50
3.5. Series de tiempo variables meteorológicas	51

3.6. Series de tiempo radiación solar ultravioleta	51
3.14. Flujo de ejecución de pruebas para un modelo.	61
3.15. Arreglo 2-dimensional de entrada para un algoritmo de Aprendizaje Au- tomático.	62
3.16. Arreglo de entrada para una Red Neuronal Recurrente.	63
3.19. Arquitectura <i>encoder-decoder</i>	69

Acrónimos

ARIMA Autoregressive Integrated Moving Average Model.

CH₄ Metano.

CO Monóxido de Carbono.

CO₂ Dióxido de Carbono.

COV Compuestos Orgánicos Volátiles.

DV Dirección del viento.

EEA Agencia Europea de Medio Ambiente.

$\mu\text{g}/\text{m}^3$ microgramos por metro cúbico.

HNM Hidrocarburos no metánicos.

HR Humedad relativa del aire.

INE Instituto Nacional de Estadísticas.

LSTM Long Short Term Memory.

MINSEGPRES Ministerio Secretaría General de la Presidencia.

MMA Ministerio del Medio Ambiente de Chile.

MP Material Particulado.

MP10 Material Particulado 10.

NO Monóxido de Nitrógeno.

NO₂ Dióxido de Nitrógeno.

NO_x Óxidos de Nitrógeno.

O₃ Ozono.

OMS Organización Mundial de la Salud.

PA Presión atmosférica.

PCA Principal Component Analysis.

PP Precipitaciones.

RM Región Metropolitana.

RS Radiación Solar.

SINCA Sistema de Información Nacional de Calidad del Aire.

SO₂ Dióxido de Azufre.

SVM Support Vector Machines.

TEMP Temperatura ambiente.

UVA Radiación Ultravioleta A.

UVB Radiación Ultravioleta B.

VV Velocidad del viento.

Glosario

Concentración de 8 Horas Promedio aritmético de los valores de concentración de 1 hora de Ozono, correspondiente a 8 horas sucesivas, promedio móvil (Glosario SINCA).

Contaminante Primario Contaminante producido directamente por la actividad humana o la naturaleza (Glosario SINCA).

Contaminante Secundario Contaminante producido a partir de algún(os) contaminante(s) primario(s) y otras sustancias (Glosario SINCA).

Efecto Invernadero Es un fenómeno que se explica por la presencia en la atmósfera de algunos componentes (principalmente Dióxido de Carbono (CO_2), vapor de agua, y Ozono (O_3)) que absorben una parte de la radiación infrarroja que emite la superficie de la Tierra y al mismo tiempo emiten energía radiativa de vuelta hacia la superficie. Este proceso contribuye a aumentar la temperatura media cerca del suelo, en comparación a la situación que ocurriría si la atmósfera no tuviera estos componentes (Glosario SINCA).

Norma Primaria de Calidad Ambiental Aquella que establece los valores de las concentraciones y períodos, máximos o mínimos permisibles de elementos, compuestos, sustancias, derivados químicos o biológicos, energías, radiaciones, vibraciones, ruidos o combinación de ellos, cuya presencia o carencia en el ambiente pueda constituir un riesgo para la vida o la salud de la población y definen los niveles que originan situaciones de emergencia (Glosario SINCA).

Radiación Ultravioleta A Corresponde a la radiación UV cuya longitud de onda se encuentra en el rango 315–400 nm (Cordero Raúl R, 2014).

Radiación Ultravioleta B Corresponde a la radiación UV cuya longitud de onda se encuentra en el rango 280–315 nm (Cordero Raúl R, 2014).

Zona Saturada Aquella área geográfica en que una o más Normas de Calidad Ambiental se encuentran sobrepasadas (Glosario SINCA).

Introducción

El Ozono es un gas presente en la atmósfera, que tiene la beneficiosa y esencial función de absorber la mayor parte de la radiación solar ultravioleta, posibilitando así la existencia de la vida terrestre. A lo anterior se le conoce como la Capa de Ozono, que está compuesta por la parte mayoritaria de dicho gas, localizada en la estratósfera (Cordero Raúl R, 2014). Sin embargo, el Ozono troposférico, el cual se encuentra al nivel de la superficie terrestre, es un peligroso contaminante atmosférico con efectos adversos para la salud humana y la producción agrícola, además de contribuir negativamente al Efecto Invernadero.

Producido a través de un proceso fotoquímico, que involucra óxidos de nitrógeno y compuestos orgánicos volátiles en la baja atmósfera, el Ozono troposférico se ha convertido en un problema medioambiental importante que ha ido empeorando en la mayoría de las grandes urbes (Organización Mundial de la Salud, 2005), Santiago de Chile dentro de ellas, influenciado por el crecimiento de la población, la industrialización y el aumento del parque automotriz.

Dicho contexto ha originado una fuerte necesidad por contar con herramientas capaces de predecir de forma veraz y oportuna, la concentración futura del Ozono sobre la superficie, dado que facilitaría la implementación de las medidas preventivas necesarias, por parte de los organismos fiscalizadores correspondientes, en busca del cumplimiento de las normas nacionales vigentes, con tal de mitigar los efectos adversos sobre la población y el medio ambiente.

En consecuencia, este trabajo persigue como objetivo general el:

- Diseñar e implementar un modelo predictivo de la concentración de Ozono Troposférico para Santiago.

Y como objetivos específicos se busca:

- Identificar las variables que permiten predecir la concentración de Ozono en la Tropósfera.
- Identificar el horizonte temporal de validez técnica y práctica del modelo predictivo.
- Comparar las predicciones del modelo respecto de los datos de concentración reales otorgados por el Sistema de Información Nacional de Calidad del Aire (SINCA) para una de las estaciones de monitoreo de Santiago.

Para realizar lo anterior se toma en consideración el aumento explosivo, a lo largo de las últimas décadas, de la creación y uso de técnicas provenientes del subcampo de la Inteligencia Artificial, el Aprendizaje de Máquina o *Machine Learning*, que posibilita a los computadores procesar grandes volúmenes de datos, pudiendo identificar de forma automática patrones dentro de ellos y ofreciendo la capacidad de modelar fenómenos altamente complejos para así, por ejemplo, concretar predicciones del comportamiento futuro de dichos fenómenos. Dentro de tales técnicas destacan las Redes Neuronales Artificiales, que basan su funcionamiento en las neuronas del cerebro humano y que cuentan con la capacidad de aprender cualquier relación subyacente entre las variables presentes en los datos observados.

En concreto, se explorará la utilización de las Redes Neuronales Recurrentes, las cuales tienen una gran capacidad para modelar fenómenos que cuentan de forma inherente con dependencias temporales. Específicamente hablando, se utilizarán las Long Short Term Memory (LSTM), las cuales son un tipo especial de Redes Neuronales Recurrentes y que cuentan con la propiedad de tener una *memoria*, pudiendo recordar información del pasado por largos periodos de tiempo. También, para dicha red, se implementará una arquitectura de tipo *encoder-decoder*, concebida inicialmente para procesar secuencias de texto en traducción por ejemplo. El rendimiento obtenido con las LSTM será comparado con otras técnicas de Aprendizaje Automático tales como: Redes Feed Forward (redes neuronales no recurrentes),

Regresión Lineal y Support Vector Machines (SVM). Además, la comparación se realizará con técnicas basadas en series de tiempo, tales como ARIMA.

El presente trabajo se organiza de la siguiente manera: En el Capítulo 1 se presentan los antecedentes generales del problema, lo cual involucra tanto los aspectos químicos como normativos ambientales, además de proveer una visión general de las técnicas utilizadas para modelar el problema. En el Capítulo 2 se presenta el Estado del Arte donde se hace una revisión de la literatura existente respecto de la generación de modelos predictivos para la concentración de ozono troposférico. En el Capítulo 3 se presenta una descripción del caso a utilizar como base de estudio en este trabajo, además de los aspectos metodológicos necesarios para llevar a cabo la solución del problema, el cual involucra la descripción de todas las decisiones respecto de la escala temporal utilizada, horizonte de predicción, tratamiento de datos faltantes, recolección de datos, pre-procesamiento de datos, división del dataset en conjunto de entrenamiento/validación/test, así como también la implementación de los modelos a ser evaluados y las métricas de rendimiento a usar. En el Capítulo 4 se presenta el análisis de los resultados obtenidos, comparando el rendimiento de todos los modelos evaluados. Y por último las Conclusiones, en donde se presentan las reflexiones finales respecto a la ejecución de la presente memoria, así como también de los posibles trabajos futuros por desarrollar.

Capítulo 1

Marco Conceptual

El propósito de este capítulo es describir los elementos conceptuales que jugarán un papel relevante en el desarrollo de este trabajo, tanto en torno al problema a tratar como las técnicas adoptadas para la su resolución.

En la Sección 1.1 se explicará el contexto químico y ambiental del Ozono Troposférico, detallando los aspectos formativos, efectos adversos y las normativas a nivel nacional e internacional relacionadas a mitigar estas últimas. En la Sección 1.2 se entregarán detalles acerca de los modelos predictivos basados en técnicas de series de tiempo, tales como ARI-MA. En la Sección 1.3 se presentarán distintas técnicas basadas en Aprendizaje Automático para realizar predicciones, entre las cuales se encuentran: la Regresión Lineal, Support Vector Machines (SVM) y Redes Neuronales. Finalmente, en la Sección 1.4, se presentarán las diversas métricas para evaluar el rendimiento de los modelos predictivos.

1.1. Contexto

1.1.1. El Ozono

El Ozono es un gas que está presente de manera natural en la atmósfera. Se trata de una molécula de tres átomos de oxígeno que reacciona rápidamente con varios compuestos químicos y cuyo símbolo es O₃ (la O representa el Oxígeno).

Cerca de un 90 % del Ozono se encuentra en la estratósfera, es decir entre los 10 y los 50 km de altitud. El Ozono estratosférico es comúnmente conocido como la Capa de Ozono, la cual tiene como función filtrar la radiación ultravioleta proveniente del Sol y así posibilitar la existencia de vida en la superficie terrestre.

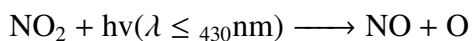
En la tropósfera, es decir a altitudes menores a los 10 km, el Ozono existente (en menor cantidad que en la estratósfera), tiene un efecto sanitizante sobre la atmósfera si es que se le encuentra en concentraciones naturales. No obstante, sobrepasadas estas últimas, hace que dicho gas sea considerado como un contaminante.

1.1.2. Formación del Ozono troposférico

El Ozono no es emitido directamente a la tropósfera, más bien es un contaminante secundario que se genera a través de un complejo proceso fotoquímico (en presencia de luz solar) a partir de precursores químicos tales como los Óxidos de Nitrógeno (NO_x), Compuestos Orgánicos Volátiles y Monóxido de Carbono (Agencia Europea de Medio Ambiente, 2016).

En forma simplificada, la fotoquímica atmosférica que regula la formación de Ozono en la tropósfera puede ser resumida como se muestra a continuación (Organización Mundial de la Salud, 2005).

1. El Dióxido de Nitrógeno (NO₂) absorbe radiación solar (hν), para luego disociarse y formar Monóxido de Nitrógeno (NO) Oxígeno atómico.



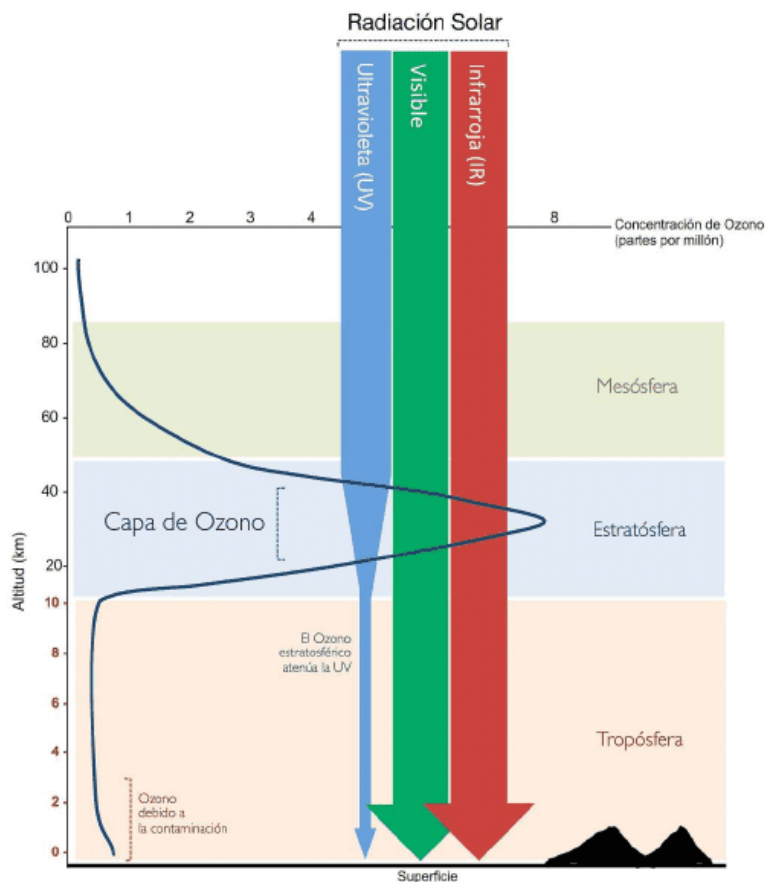
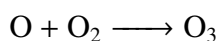
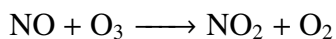


Figura 1.1: Distribución del Ozono en la atmósfera terrestre. Fuente: Cordero Raúl R (2014).

2. El Oxígeno atómico se combina con Oxígeno molecular para formar Ozono.



3. El Ozono es descompuesto mediante la reacción con el Monóxido de Nitrógeno (NO), formando Dióxido de Nitrógeno y Oxígeno molecular.



El mecanismo químico descrito previamente, representa el estado de equilibrio en la atmósfera considerando la ausencia de otras sustancias químicas, situación en que la cantidad de O_3 sería controlada por las cantidades relativas de NO y NO_2 , así como también la intensidad de la radiación solar. Por lo tanto, los altos niveles de Ozono (superiores a los naturales) ocurren cuando tal equilibrio es alterado, ya sea por eventos que consumen NO o favorecen

la producción de NO_2 . Otro factor es la participación conjunta de los COVs y los NO_x en otras reacciones químicas. A través de la acción de radicales hidróxilos (OH) formados por la presencia de radiación solar, los COV son degradados para producir sustancias que reaccionan con el NO para producir NO_2 sin consumir O_3 . El resultado neto de estas reacciones es que más de una molécula de O_3 es formada por cada molécula degradada de COV (Agencia Europea de Medio Ambiente, 2012).

El mayor factor de desequilibrio es la acción humana, la cual es fuente de la mayoría de las emisiones de los precursores del Ozono. En el caso de los Óxidos de Nitrógeno (NO_x) las principales fuentes son los procesos que involucran la quema de combustibles fósiles, tales como: transporte, plantas de energía e industrias. Los Compuestos Orgánicos Volátiles son emitidos desde un gran número de fuentes que incluyen las pinturas o limpiadores, transporte por carretera, refinerías, solventes, actividades agrícolas y combustión de biomasa. Los COV también son emitidos por fuentes naturales como la vegetación, en cantidades dependientes de la temperatura. El Monóxido de Carbono (CO) es un gas emitido como resultado de la quema incompleta de combustibles fósiles y biocombustibles (Amann y cols., 2008).

Ahora bien, los procesos fotoquímicos asociados a la formación del Ozono troposférico no presentan constantes de velocidad fija, sino que presentan variaciones de acuerdo a factores meteorológicos: la intensidad y el ángulo de incidencia de la radiación solar UV; las condiciones de ventilación, que pueden variar las concentraciones de los precursores; la estacionalidad, ya que por ejemplo en Verano, las concentraciones de Ozono suelen aumentar, dada la mayor presencia de radiación solar (Agencia Europea de Medio Ambiente, 2016; Amann y cols., 2008).

1.1.3. Efectos del Ozono troposférico

El Ozono troposférico puede producir los siguientes tipos de efectos adversos (EEA Agencia Europea de Medio Ambiente, 2016):

- Efectos sobre la salud:

- Puede disminuir la función pulmonar; agravar el asma y otras enfermedades pulmonares; puede conducir a una muerte prematura.
- Efectos sobre los ecosistemas:
 - Daña la vegetación, perjudicando la reproducción y el crecimiento. Puede alterar la estructura del ecosistema reduciendo la biodiversidad y disminuyendo la absorción de CO₂ por parte de las plantas (fotosíntesis).
- Efectos sobre el clima:
 - El Ozono es un gas de efecto invernadero, lo cual contribuye negativamente al calentamiento de la atmósfera del planeta.

Dichos efectos adversos tienden a acrecentarse durante los periodos de verano, cuando las niveles de Ozono troposférico son más altos, debido a la presencia de mayor radiación solar.

1.1.4. Normativa y estándares en Chile

En el año 1996, a través del Decreto Supremo N° 131/96 del Ministerio Secretaría General de la Presidencia (MINSEGPRES), se declaró a la Región Metropolitana (RM) como Zona Saturada por Ozono y otros contaminantes. Posteriormente, en el año 2003 entró en vigencia el Decreto de Ley 112 del MINSEGPRES, que establece la Norma Primaria de Calidad de Aire para Ozono (O₃), con tal de proteger la salud de la población de los efectos adversos derivados de la exposición al contaminante. Tal norma establece un nivel máximo para el Ozono en 61 ppb (120 ug/m³N) como concentración de 8 horas. Además establece tres niveles en concentraciones de 1 hora, que originan situaciones de emergencia ambiental:

- Nivel 1 (alerta ambiental): 204–407 ppb (400–799 ug/m³N)
- Nivel 2 (preemergencia ambiental): 408–509 ppb (800–999 ug/m³N)
- Nivel 3 (emergencia ambiental): 510 ppb o superior (1000 ug/m³N o superior)

1.1.5. Normativas y estándares internacionales

A nivel internacional la referencia principal es la Organización Mundial de la Salud (OMS). El límite recomendado por esta organización es de 100 microgramos por metro cúbico como concentración promedio de 8 horas (Organización Mundial de la Salud, 2005). Otras autoridades ambientales importantes son las de Estados Unidos y Europa:

- Agencia de Protección Ambiental de Estados Unidos (2015): $120 \mu\text{g}/\text{m}^3$ concentración promedio de 8 horas.
- Agencia Europea de Medio Ambiente (2016): 70 ppb concentración promedio de 8 horas.

1.2. Modelos predictivos basados en series de tiempo

Una serie de tiempo en un conjunto de observaciones medidas de forma secuencial a lo largo del tiempo (Chatfield, 2000). Existen dos tipos de series de tiempo:

- Series de tiempo continuo: Obtenidas a través de mediciones hechas de forma continua a lo largo del tiempo.
- Series de tiempo discreto: Obtenidas a través de mediciones realizadas a lo largo de un conjunto discreto de puntos de tiempo.

En ambos casos, la variable medida puede ser discreta o continua. Para una serie de tiempo discreto, la dimensión del tiempo es la discreta. Para una serie de tiempo continua, la variable observada es típicamente una variable en $\mathbb{R}$ registrada de forma continua. La manera usual de analizar una serie como tal, es muestrearla en intervalos iguales de tiempo para así discretizarla, perdiéndose poca o ninguna información si es que los intervalos son lo suficientemente pequeños (Chatfield, 2000).

Con motivos de análisis predictivo para series de tiempo univariada, existen diversas herramientas, dentro de las cuales se destacan los modelos Autoregressive Integrated Moving

Average Model (ARIMA), introducida por Box y Jenkins (1970), cuyas partes componentes se detallarán a continuación.

1.2.1. Procesos Auto regresivos (AR)

Una serie de tiempo $\{X_t\}$ se dice que es un proceso auto regresivo de orden p ($AR(p)$) si es la combinación lineal de los p valores pasados más una variable de ruido aleatoria, de tal forma que:

$$X_t = \phi_1 X_{t-1} + \phi_2 X_{t-2} + \dots + \phi_p X_{t-p} + Z_t \quad (1.1)$$

Donde $\{Z_t\}$ es un proceso aleatorio con media igual a cero y varianza σ_z^2 . Usando el operador de desplazamiento hacia atrás B , tal que $BX_t = X_{t-1}$, el modelo $AR(p)$ puede ser reescrito de la siguiente forma:

$$\phi(B)X_t = Z_t \quad (1.2)$$

Donde $\phi(B) = 1 - \phi_1 B - \phi_2 B^2 - \dots - \phi_p B^p$ es un polinomio en B de orden p .

1.2.2. Procesos de Media Móvil (MA)

Una serie de tiempo $\{X_t\}$ se dice que es un proceso de media móvil de orden q ($MA(q)$) si es una combinación lineal de los q últimos ruidos aleatorios, de tal forma que:

$$X_t = Z_t + \theta_1 Z_{t-1} + \dots + \theta_q Z_{t-q} \quad (1.3)$$

Donde $\{Z_t\}$ denota un proceso puramente aleatorio con media igual a cero y varianza constante σ_z^2 . (1.3) puede ser reescrita de la siguiente forma:

$$X_t = \theta(B)Z_t \quad (1.4)$$

Donde $\theta(B) = 1 + \theta_1 B + \dots + \theta_q B^q$ es un polinomio en B de orden q .

1.2.3. Procesos ARMA

Un modelo mixto de media móvil auto regresiva con p términos auto regresivos y q términos de media móvil es abreviado como $ARMA(p, q)$, el cual puede ser escrito como:

$$\phi(B)X_t = \theta(B)Z_t \quad (1.5)$$

Donde $\phi(B)$, $\theta(B)$ son polinomios en B de órdenes finitos p, q respectivamente. Esto es una combinación de las ecuaciones (1.1) y (1.3).

1.2.4. Procesos ARIMA

En la práctica, muchas series de tiempo no son estacionarias, es decir, sus propiedades estadísticas como la media y varianza cambian a través del tiempo, por lo que no se pueden aplicar directamente los modelos AR, MA y ARMA. Una forma de lidiar con la no estacionariedad de una serie es aplicar un proceso de *diferenciación*. Una diferenciación de primer orden, expresada como $(X_t - X_{t-1}) = (1 - B)X_t$ pueden ser a su vez, diferenciada para obtener una diferenciación de mayor orden. La d -ésima diferenciación puede ser escrita como $(1 - B)^d X_t$. Si la serie de los datos originales es diferenciada d veces antes de ajustar un proceso $ARMA(p, q)$, el modelo para la serie no diferenciada original se dice que es un proceso $ARIMA(p, d, q)$ donde la letra I denota *integrated* (integrado) y d indica el número de diferenciaciones hechas (Chatfield, 2000). Formalmente, un proceso ARIMA se describe como:

$$\phi(B)(1 - B)^d X_t = \theta(B)Z_t \quad (1.6)$$

Donde el operador combinado AR es ahora $\phi(B)(1 - B)^d$.

1.3. Modelos predictivos basados en Aprendizaje Automático

Este tipo de algoritmos permiten identificar patrones en los datos observados, construir modelos que explican ciertos fenómenos y predecir cosas sin la necesidad de contar con reglas

y modelos explícitos pre-programados. En otras palabras, son algoritmos que son capaces de aprender de los datos. Mitchell (1997) entrega la siguiente definición: “*Un programa de computador se dice que aprende de una experiencia E con respecto a una clase o tipo de tarea T y una medida de rendimiento P , si su rendimiento en T medido por P , mejora con la experiencia E .*”

Esta experiencia se manifiesta como un *dataset* o conjunto de datos. Un dataset es una colección de muchos ejemplos, los cuales a su vez son una colección de características que han sido cuantitativamente medidas a partir de un objeto o suceso y que el algoritmo de Aprendizaje Automático se quiere que procese (Goodfellow, Bengio, y Courville, 2016). Dicho dataset se suele describir como un conjunto de m vectores numéricos n -dimensionales $x^{(i)}$, o más formalmente como $\{x^{(i)}; i = 1, \dots, m; x^{(i)} \in \mathbb{R}^n\}$.

A grandes rasgos, los algoritmos de Aprendizaje Automático pueden ser clasificados en dos tipos: Aprendizaje Supervisado y no Supervisado; los cuales se diferencian por el tipo de experiencia a la cual son expuestos durante el proceso de aprendizaje.

- **Aprendizaje Supervisado:** En general, esta categoría de algoritmos involucra la construcción de un modelo para predecir o estimar una salida, basada en una o más entradas (James, Witten, Hastie, y Tibshirani, 2013). Para ello, los algoritmos experimentan o aprenden un dataset en el cual cada ejemplo está asociado a una etiqueta o variable objetivo.

Formalmente, se utiliza $x^{(i)}$ para denotar las características de entrada e $y^{(i)}$ para denotar la salida o variable objetivo que se está tratando de predecir. Un par $(x^{(i)}, y^{(i)})$ es llamado un ejemplo de entrenamiento y el conjunto de datos que se usará para aprender consta de una lista de m ejemplos $\{(x^{(i)}, y^{(i)}); i = 1, \dots, m\}$ el cual es llamado conjunto de entrenamiento. Se usará $\mathcal{X}$ para denotar el espacio de las variables de entrada, e $\mathcal{Y}$ el espacio de los valores de salida. El objetivo en un problema de aprendizaje supervisado, dado un conjunto de datos de entrenamiento, es aprender una función $h : \mathcal{X} \rightarrow \mathcal{Y}$, la cual mapea las características de entradas con las de salida.

Cuando la variable objetivo y que se está tratando de predecir es continua, se llamará al problema de aprendizaje como uno de regresión. Cuando y solo puede tomar un

número pequeño de valores discretos, este problema será uno de clasificación.

- **Aprendizaje no Supervisado:** Este tipo de algoritmos experimenta un conjunto de datos con variables entradas $x^{(i)}$ pero sin las salidas u objetivos de supervisión $y^{(i)}$. Aquí, el interés principal no es la predicción, sino más bien extraer información de la estructura y relaciones subyacentes del conjunto de datos (James y cols., 2013). Con este tipo de algoritmos, uno de los problemas más importantes es el de agrupamiento (*clustering*).

1.3.1. Regresión Lineal

Dado el conjunto de datos de entrenamiento, compuesto por m ejemplos $\{(x^{(i)}, y^{(i)}), i = 1, \dots, m\}$, se define $\mathcal{X} = \mathbb{R}^n$ como el espacio de los valores de entrada e $\mathcal{Y} = \mathbb{R}$ los de salida. Se define el modelo lineal $h : \mathcal{X} \rightarrow \mathcal{Y}$ como:

$$h(x) = \sum_{j=1}^n \theta_j x_j = \theta^T x \quad (1.7)$$

Donde los θ_j son los pesos que parametrizan el espacio de funciones lineales desde $\mathcal{X}$ a $\mathcal{Y}$. En el extremo derecho de (1.7), θ y x son vectores, mientras que n representa el número de variables de entrada.

Se define la función de costo (1.8), la cual mide para cada valor de los θ , cuán cerca las predicciones $h(x^{(i)})$ están de los correspondientes $y^{(i)}$.

$$J(\theta) = \frac{1}{2} \sum_{i=1}^m (h_{\theta}(x^{(i)}) - y^{(i)})^2 \quad (1.8)$$

El objetivo es escoger los θ que minimizan (1.8). Para hacer lo anterior, comúnmente se utiliza el método del **Gradiente Descendente**, el cual a partir de un θ inicial aplica el siguiente proceso iterativo:

$$\theta_j = \theta_j - \alpha \frac{\partial}{\partial \theta_j} J(\theta) \quad (1.9)$$

Donde α es el llamado *learning rate* (tasa de aprendizaje). Luego de derivar $\frac{\partial}{\partial \theta_j} J(\theta)$, la regla de actualización para un sólo ejemplo de entrenamiento se presenta como:

$$\theta_j = \theta_j + \alpha(y^{(i)} - h_\theta(x^{(i)}))x_j^{(i)} \quad (1.10)$$

En términos matriciales, sea X una matriz de $m \times n$ donde cada fila corresponde a los $x^{(i)}$. Sea además $\vec{y}$ un vector m -dimensional que contiene todos los valores objetivo del conjunto de entrenamiento. Se define entonces $J(\theta)$ como:

$$J(\theta) = \frac{1}{2}(X\theta - \vec{y})^T(X\theta - \vec{y}) \quad (1.11)$$

Por lo que los valores de θ que minimizan (1.11) son:

$$\theta = (X^T X)^{-1} X^T \vec{y} \quad (1.12)$$

1.3.2. Support Vector Machines

Las Support Vector Machines (SVM) son una técnica de aprendizaje supervisado que se utiliza tanto para problemas de clasificación como de regresión. Para estos últimos, denotada como Epsilon Support Vector Regression (ϵ -SVR) (Smola y Schölkopf, 2004), considérese el conjunto de entrenamiento compuesto por m ejemplos $\{(x^{(i)}, y^{(i)}), i = 1, \dots, m\}$ con $X \subset \mathbb{R}^n$ denotando el espacio de las entradas e $Y \subset \mathbb{R}$ las salidas. Para el caso de funciones lineales, el objetivo principal es encontrar una función $f(x)$:

$$f(x) = \langle w, x \rangle + b \text{ con } w \in X, b \in \mathbb{R} \quad (1.13)$$

Donde $\langle \rangle$ denota el producto punto en X . Para lo anterior, es necesario resolver el siguiente problema de optimización:

$$\begin{array}{ll} \text{minimizar} & \frac{1}{2} \|w\|^2 \\ \text{sujeto a} & \begin{cases} y^{(i)} - \langle w, x^{(i)} \rangle - b \leq \epsilon \\ \langle w, x^{(i)} \rangle + b - y^{(i)} \leq \epsilon \end{cases} \end{array} \quad (1.14)$$

Es decir, $f(x)$ debe ser una función tal que tenga un error de a lo más ϵ (margen duro), para cada uno de los valores verdaderos $y^{(i)}$. Sin embargo, puede que tal problema de optimización

no sea factible, es decir, no se pueda encontrar una función $f(x)$ que cumpla las restricciones de (1.14). En ese caso, con tal de encontrar una solución factible, se permiten algunos errores introduciendo las variables de holgura ξ_i, ξ_i^* a (1.14), redefiniéndose el problema de optimización de la siguiente forma:

$$\begin{aligned} & \text{minimizar} && \frac{1}{2} \|w\|^2 + C \sum_{i=1}^m (\xi_i + \xi_i^*) \\ & \text{sujeto a} && \begin{cases} y^{(i)} - \langle w, x^{(i)} \rangle - b & \leq \epsilon + \xi_i \\ \langle w, x^{(i)} \rangle + b - y^{(i)} & \leq \epsilon + \xi_i^* \\ \xi_i, \xi_i^* & \geq 0 \end{cases} \end{aligned} \quad (1.15)$$

En (1.15) la constante $C > 0$, establece el *trade-off* entre la complejidad y exactitud de $f(x)$.

La función objetivo del problema de optimización (1.15) puede ser reescrita como una función de Lagrange, introduciendo los multiplicadores de Lagrange α_i, α_i^* , tal como se muestra a continuación.

$$\begin{aligned} & \text{maximizar} && \begin{cases} -\frac{1}{2} \sum_{i,j=1}^m (\alpha_i - \alpha_i^*)(\alpha_j - \alpha_j^*) \langle x^{(i)}, x^{(j)} \rangle \\ -\epsilon \sum_{i=1}^m (\alpha_i + \alpha_i^*) + \sum_{i=1}^m y_i (\alpha_i - \alpha_i^*) \end{cases} \\ & \text{sujeto a} && \sum_{i=1}^m (\alpha_i - \alpha_i^*) = 0 \quad \alpha_i, \alpha_i^* \in [0, C] \end{aligned} \quad (1.16)$$

Además, w y $f(x)$ también pueden ser reescritas como funciones de α_i, α_i^* .

$$w = \sum_{i=1}^m (\alpha_i - \alpha_i^*) x^{(i)}; \quad f(x) = \sum_{i=1}^m (\alpha_i - \alpha_i^*) \langle x^{(i)}, x \rangle + b \quad (1.17)$$

Las definiciones anteriores estaban orientadas a resolver problemas de carácter lineal. En el caso de los no lineales, se realiza un mapeo de los datos de entrenamiento $x^{(i)}$ a un espacio de características de mayor dimensionalidad a través de la transformación $\Phi : X \rightarrow \mathcal{F}$. Dado que (1.16) y (1.17) dependen solo de productos internos entre patrones de $x^{(i)}$, simplemente

basta con conocer una función k (*kernel*) que cumpla con $k(x, x') = \langle \Phi(x), \Phi(x') \rangle$, para reescribir (1.16) como:

$$\begin{aligned} & \text{maximizar} \quad \begin{cases} -\frac{1}{2} \sum_{i,j=1}^m (\alpha_i - \alpha_i^*)(\alpha_j - \alpha_j^*) k(x^{(i)}, x^{(j)}) \\ -\epsilon \sum_{i=1}^m (\alpha_i + \alpha_i^*) + \sum_{i=1}^m y_i (\alpha_i - \alpha_i^*) \end{cases} \\ & \text{sujeto a} \quad \sum_{i=1}^m (\alpha_i - \alpha_i^*) = 0 \quad \alpha_i, \alpha_i^* \in [0, C] \end{aligned} \quad (1.18)$$

$$w = \sum_{i=1}^m (\alpha_i - \alpha_i^*) \Phi(x^{(i)}); \quad f(x) = \sum_{i=1}^m (\alpha_i - \alpha_i^*) k(x^{(i)}, x) + b \quad (1.19)$$

Un ejemplo clásico de k es el *kernel* gaussiano *radial basis function* (rbf):

$$k(x, x') = \exp\left(-\frac{1}{2\sigma^2} \|x - x'\|^2\right) \quad (1.20)$$

1.3.3. Redes Neuronales Artificiales

Las redes neuronales artificiales son sistemas de computación biológicamente inspirados, compuestas por unidades de procesamiento llamadas neuronas artificiales, comúnmente referidas como nodos o unidades, las cuales se encuentran interconectadas mediante arcos que representan las conexiones sinápticas de una red neuronal biológica.

Una neurona es una unidad computacional, representada por la figura 1.2, la cual recibe n entradas $x_1, x_2, \dots, x_n$ produciendo como salida:

$$a = \varphi\left(\sum_{i=1}^n w_i x_i + b\right) \quad (1.21)$$

Donde los pesos $w_1, w_2, \dots, w_n$, son números reales que expresan la importancia de las entradas respectivas en relación a la salida y b es un escalar de sesgo. $\varphi(\cdot)$ es una función no lineal llamada función de activación. Distintos tipos de estas funciones pueden ser utilizadas (ver figura 1.3).

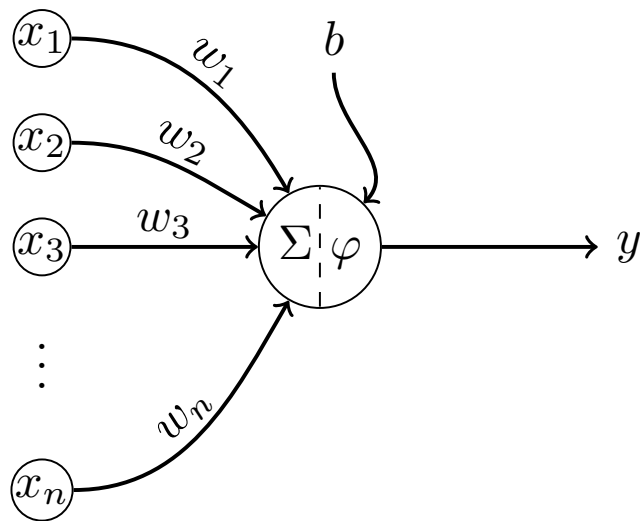


Figura 1.2: Ejemplo de una neurona.

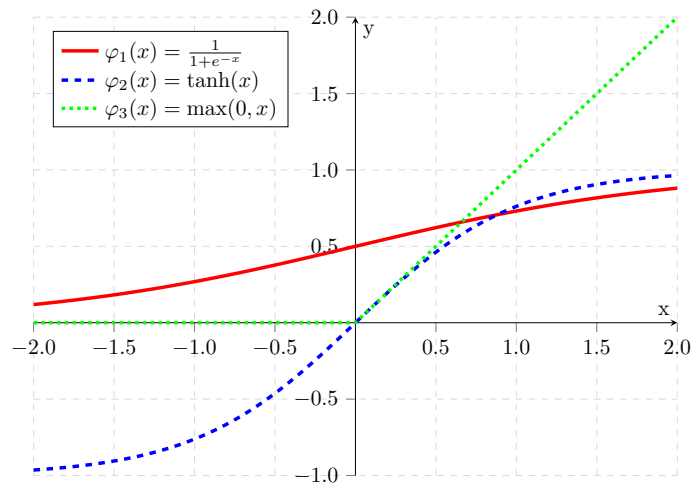


Figura 1.3: Funciones de activación.

El tipo más tradicional de redes neuronales artificiales son las Feed Forward, las cuales son modeladas como colecciones de neuronas que están conectadas en un grafo acíclico, donde las salidas de algunas neuronas pueden convertirse en la entrada de otras. Los ciclos no son permitidos ya que implicaría la existencia de bucles infinitos, los cuales impedirían el flujo de la información siempre hacia adelante dentro de la red. Las redes neuronales son a menudo organizadas en distintas capas de neuronas. La forma más tradicional de hacerlo es como se muestra en la figura 1.4, es decir, con una capa de entrada (más a la izquierda), una capa

oculta (la del centro), y la capa de salida (más a la derecha).

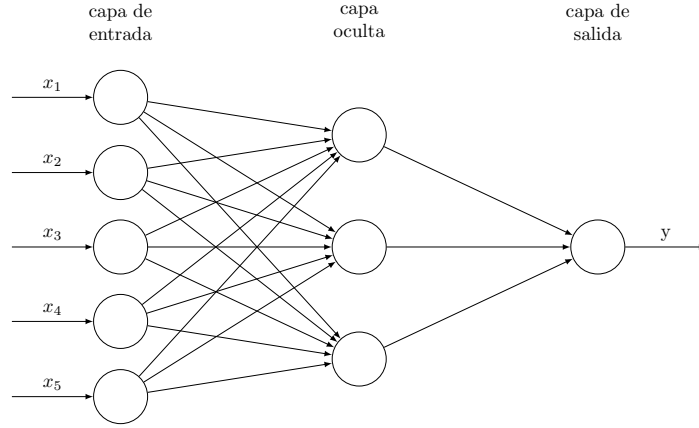


Figura 1.4: Red Neuronal Feed Forward. En esta red las capas están totalmente conectadas. Esto significa que una neurona en cualquier capa está conectada con todas las neuronas de la capa previa.

En términos notacionales, sea n_l el número de capas en la red; por lo que $n_l = 3$ en la figura 1.4. La capa i se denotará como L_i , por lo que L_1 es la capa de entrada y L_{n_l} es la capa de salida. Los parámetros de la red son (W, b) . Se denotará por $a_i^{(l)}$ la activación o valor de salida para la neurona i en la capa l .

Entrenamiento

El entrenamiento de las redes es logrado a través de un proceso iterativo llamado *Backpropagation*, el cual usa la regla de la cadena para calcular la derivada de la función de pérdida J con respecto a cada parámetro en la red. Los pesos son luego modificados mediante el método del Gradiente Descendente. Dado que la función de pérdida no es convexa, no se puede asegurar que el proceso de *Backpropagation* alcance un mínimo global.

Supóngase que se tiene el conjunto de entrenamiento $\{(x^{(1)}, y^{(1)}), \dots, (x^{(m)}, y^{(m)})\}$ de m ejemplos. Para un solo ejemplo de entrenamiento (x, y) , se define la función de costo con respecto a ese ejemplo como J . El objetivo es minimizar J como función de W y b . Una iteración del

gradiente descendente actualiza los parámetros W, b como sigue:

$$w_{ij}^{(l)} = w_{ij}^{(l)} - \alpha \frac{\partial}{\partial w_{ij}^{(l)}} J(w, b) \quad (1.22)$$

$$b_i^{(l)} = b_i^{(l)} - \alpha \frac{\partial}{\partial b_i^{(l)}} J(w, b) \quad (1.23)$$

Donde α es el *learning rate* (tasa de aprendizaje).

Dado un ejemplo de entrenamiento (x, y) , primero se ejecutará un paso hacia adelante para calcular todas las activaciones a través de la red, incluyendo el valor de la salida $\hat{y}$. Luego, para cada neurona i en la capa l , se calculará $\delta_i^{(l)}$ que mide en cuánto esa neurona es responsable por cualquier error en la salida. Para cada neurona de salida, se puede calcular directamente la diferencia entre la activación de la red y el valor objetivo verdadero, redefiniéndose como $\delta_i^{(n_l)}$ basado en un promedio ponderado de los errores de las neuronas que usan $a_i^{(l)}$ como entrada. La secuencia de pasos que componen el entrenamiento, se describe a continuación:

1. Realizar una paso hacia adelante, calculando las salidas de las funciones de activación de las capas $L_2, \dots, L_{n_l}$.
2. Para la capa de salida (n_l), establecer:

$$\delta^{n_l} = -(y - a^{(n_l)})f'(z^{(n_l)}) \quad (1.24)$$

3. Para $l = n_l - 1, n_l - 2, \dots, 2$ hacer:

$$\delta^{(l)} = ((W^{(l+1)})^T \delta^{(l+1)})f'(z^{(l)}) \quad (1.25)$$

4. Calcular las derivadas parciales:

$$\nabla_{W^{(l)}} J(W, b; x, y) = \delta^{(l+1)}(a^{(l)})^T \quad (1.26)$$

$$\nabla_{b^{(l)}} J(W, b; x, y) = \delta^{(l+1)} \quad (1.27)$$

1.3.4. Redes Neuronales Recurrentes

A pesar de su poder, las redes neuronales Feed Forward tradicionales tienen limitaciones. Principalmente, confían en la suposición de independencia entre los ejemplos de entrenamiento y pruebas. Después de que cada ejemplo es procesado, el estado completo de la red

se pierde. Si cada ejemplo es generado independientemente, esto no representa un problema. Pero si los datos están relacionados en el tiempo y espacio, esto es inaceptable. Aquí es donde entran en juego las Redes Neuronales Recurrentes, las cuales pueden capturar las dinámicas de secuencias de datos mediante la existencia de ciclos en la red de nodos. A diferencia de las redes Feed Forward, las Redes Recurrentes pueden retener un estado, el cual puede representar información desde una ventana de contexto arbitrariamente larga (Lipton, Berkowitz, y Elkan, 2015).

Diversos modelos de procesamiento de secuencias pueden implementar las redes recurrentes (figura 1.5), ya sea para modelar entradas y salidas secuenciales, así como asignaciones entre puntos de datos individuales y secuencias (en ambas direcciones).

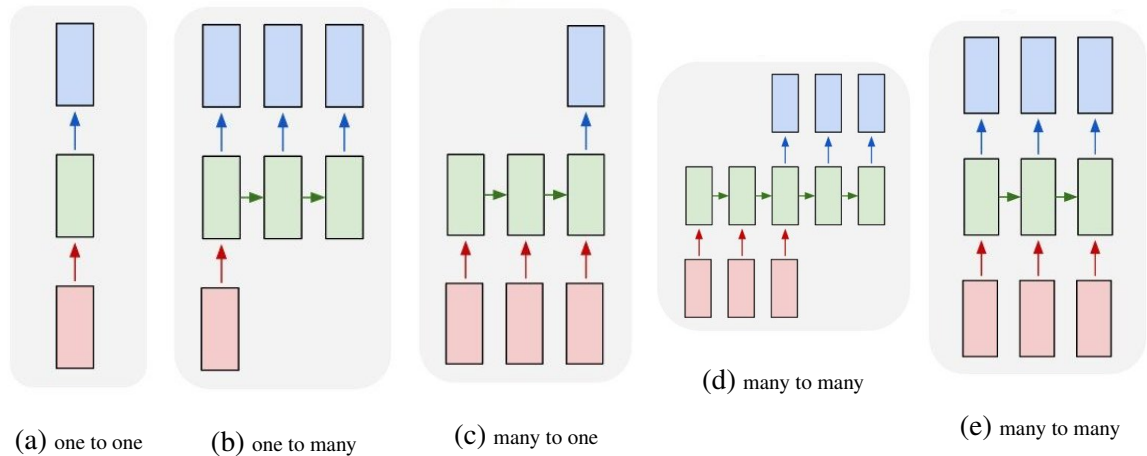


Figura 1.5: En cada figura, los rectángulos rojos, verdes y azules corresponden a las entradas, estados ocultos y salidas de la red respectivamente. (a) Este caso representa a una red Feed Forward Tradicional. (b) Subtitulado de imágenes, donde una imagen es una entrada de tipo no secuencial y la salida es la secuencia de palabras que conforman el subtítulo. (c) Clasificación de texto y video son tareas en que una secuencia es mapeada a un vector de largo fijo. (d) Esta arquitectura ha sido usada para traducción de lenguaje natural, una tarea de tipo secuencia a secuencia, las cuales pueden tener largos distintos y variables. (e) Esta arquitectura ha sido usada para aprender modelos generativos de texto, prediciendo en cada paso el siguiente caracter. Fuente: Karpathy (2015)

En términos más formales, una red neuronal recurrente es una red que está especializada en

procesar una secuencia de valores $x^{(1)}, \dots, x^{(T)}$, indexados por el tiempo $t = 1, \dots, T$. La mayoría de las redes recurrentes pueden también procesar secuencias de largo variable (Goodfellow y cols., 2016). En la figura 1.6 se grafica la representación clásica de una red neuronal recurrente.

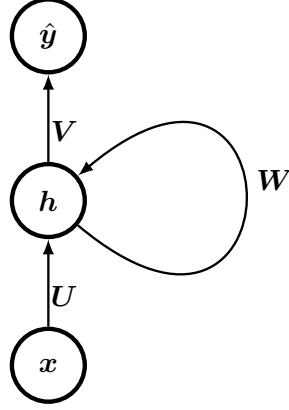


Figura 1.6: Red Neuronal Recurrente.

En cada *timestep*, la salida del paso previo junto con el vector de entrada $x^{(t)}$, son entradas a la capa oculta para producir la salida $\hat{y}^{(t)}$ y el estado oculto $h^{(t)}$ (ecuaciones (1.28) y (1.29)).

$$h^{(t)} = \varphi_h(W h^{(t-1)} + U x^{(t)}) \quad (1.28)$$

$$\hat{y}^{(t)} = \varphi_o(V h^{(t)}) \quad (1.29)$$

Donde W es la matriz de pesos recurrentes, U es la matriz de pesos entrada a estados ocultos y V es la matriz de pesos entre estados ocultos y salida. φ_h y φ_o son funciones de activación, tales como la tangente hiperbólica o sigmoideal.

Las dinámicas de la red representadas en la figura 1.6, pueden ser visualizadas desenrollándola tal como se ve en la figura 1.7. Ahí, dicha red puede ser interpretada ya no como una estructura cíclica, sino más bien como una red profunda con una capa por cada *timestep*, compartiendo los pesos a lo largo del tiempo. Esta red puede ser entrenada a través de muchos *timesteps* usando Backpropagation. En este caso, el algoritmo es llamado *Backpropagation through time* (BPTT) (Lipton y cols., 2015). El cálculo del gradiente involucra el realizar un

paso de propagación hacia adelante, de izquierda a derecha a través de la figura 1.7, seguida por un paso de propagación hacia atrás, de derecha a izquierda (Goodfellow y cols., 2016).

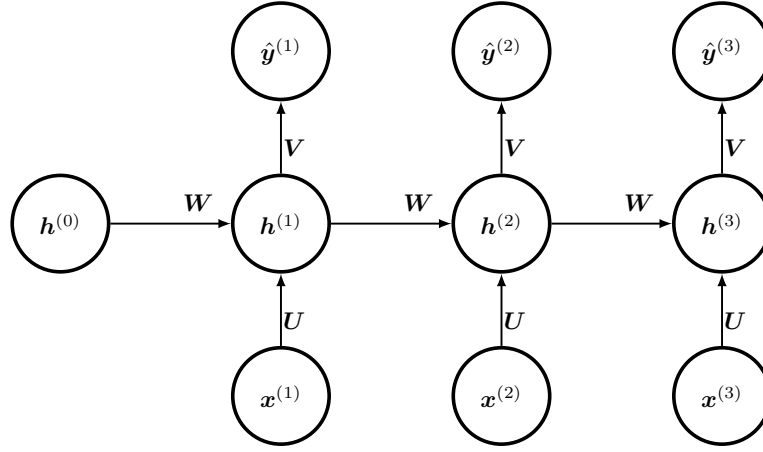


Figura 1.7: Red Neuronal Recurrente desenrollada a lo largo de tres *timesteps*.

Sin embargo, la aplicación de forma repetitiva de la fase de propagación hacia atrás durante una larga secuencia temporal, hace que la contribución de los valores del gradiente vaya paulatinamente desvaneciéndose¹ (haciéndose iguales a cero) a medida que se propaga a tiempos anteriores. A este fenómeno se le es conocido como el problema del gradiente desvaneciente (Goodfellow y cols., 2016).

A continuación se describirán algunas arquitecturas de redes neuronales recurrentes.

Redes ELMAN

En Elman (1990), se introduce una arquitectura basada en una red Feed Forward de tres capas (una de entrada, una oculta y una de salida), pero con el adicional de que cada unidad en la capa oculta tiene asociada una unidad de contexto (ver figura 1.8). Cada una de estas unidades de contexto j' toma como entrada el estado de la correspondiente unidad oculta j en *timestep* anterior, a lo largo de un arco de peso fijo $w_{jj'} = 1$. Este valor es pasado de vuelta al mismo nodo oculto j a través de un arco corriente (Lipton y cols., 2015).

¹O por el contrario, tales valores pueden “explotar” haciéndose demasiado grandes.

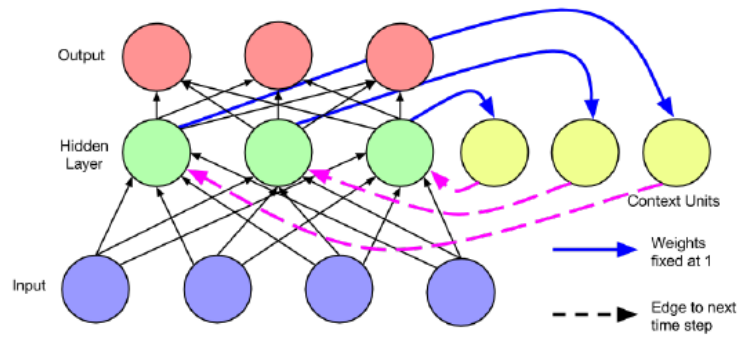


Figura 1.8: Red Neuronal Recurrente ELMAN. Fuente: Lipton y cols. (2015)

Long Short Term Memory (LSTM)

En Hochreiter y Schmidhuber (1997) se introdujo la arquitectura LSTM con el fin de superar el problema de los gradientes desvanecientes. Este modelo se parece a una red neuronal recurrente tradicional con una capa oculta, pero cada nodo (figura 1.6) es reemplazada por una célula de memoria (figura 1.9).

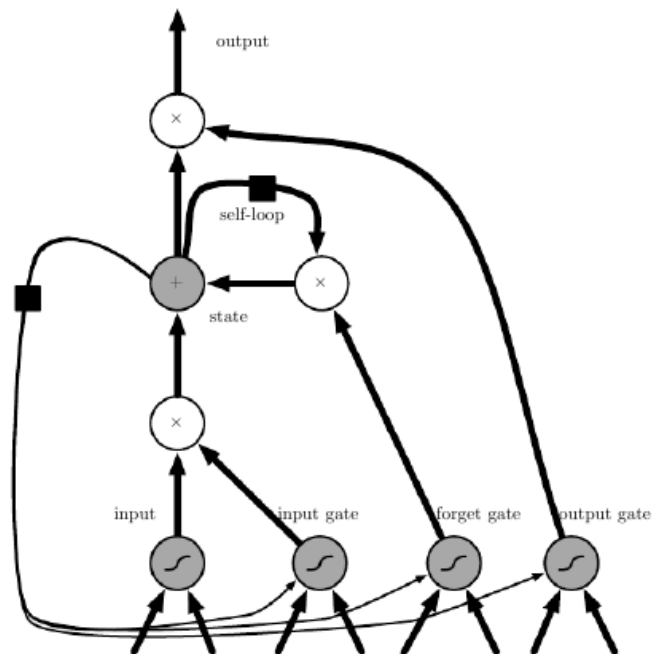


Figura 1.9: Célula LSTM. Fuente: Goodfellow y cols. (2016).

Este tipo de células tienen una recurrencia interna (*self loop*) adicional a la externa de una

red recurrente tradicional. Cada célula tiene las mismas entradas y salidas, tal como una red recurrente ordinaria, pero con más parámetros y un sistema de compuertas que controlan el flujo de la información. Estas compuertas se basan en unidades sigmoideas, de modo que si su valor es cero, el flujo al otro nodo es interrumpido. El caso contrario ocurre si su valor es igual a uno.

El componente más importante es la unidad de estado $s_i^{(t)}$ que tiene una recurrencia interna lineal. Sin embargo, el peso o ponderación asociado a ese *loop*, es controlado por una compuerta del olvido $f_i^{(t)}$ (para el *timestep* t en la célula i), que establece este peso a un valor entre 0 y 1 a través de una unidad sigmoideal (ecuación (1.30)).

$$f_i^{(t)} = \sigma \left(b_i^f + \sum_j U_{i,j}^f x_j^{(t)} + \sum_j W_{i,j}^f h_j^{(t-1)} \right) \quad (1.30)$$

En (1.30), $x^{(t)}$ y $h^{(t)}$ son los vectores actuales de la entrada y la capa oculta respectivamente, conteniendo las salidas de todas las células LSTM, y b^f, U^f, W^f son los sesgos, pesos de entrada y pesos recurrentes para las compuertas de olvido. Luego, el estado interno de la célula LSTM es actualizado como se muestra en (1.31), pero con un peso condicional *self loop* $f_i^{(t)}$.

$$s_i^{(t)} = f_i^{(t)} s_i^{(t-1)} g_i^{(t)} \sigma \left(b_i + \sum_j U_{i,j} x_j^{(t)} + \sum_j W_{i,j} h_j^{(t-1)} \right) \quad (1.31)$$

Donde b, U, W son respectivamente los sesgos, pesos de entrada y pesos recurrentes de dentro de la célula LSTM. La compuerta de entrada externa $g_i^{(t)}$ (ecuación (1.32)) es obtenida de manera similar a la compuerta del olvido, pero con sus propios sesgos y pesos.

$$g_i^{(t)} = \sigma \left(b_i^g + \sum_j U_{i,j}^g x_j^{(t)} + \sum_j W_{i,j}^g h_j^{(t-1)} \right) \quad (1.32)$$

La salida $h_i^{(t)}$ (ecuación (1.34)) de la célula LSTM puede ser filtrada a través de la compuerta de salida $q_i^{(t)}$ (ecuación (1.33)), cuyos parámetros b^o, U^o, W^o son los sesgos, pesos de entrada

y pesos recurrentes respectivamente.

$$q_i^{(t)} = \sigma \left(b_i^o + \sum_j U_{i,j}^o x_j^{(t)} + \sum_j W_{i,j}^o h_j^{(t-1)} \right) \quad (1.33)$$

$$h_i^{(t)} = \tanh(s_i^t) q_i^{(t)} \quad (1.34)$$

1.3.5. Principal Component Analysis

Principal Component Analysis (PCA) es una técnica de aprendizaje no supervisado que tiene como objetivo aprender una nueva representación de un conjunto de datos (más compacta o de menor dimensionalidad), manteniendo la mayor cantidad de información relevante posible, dejando fuera el ruido y revelando una estructura oculta.

Para lo anterior, considérese un conjunto de datos representado por la matriz X de orden $m \times n$ (m es el número de ejemplos de dimensionalidad $\mathbb{R}^n$). PCA encuentra una matriz ortonormal P , que proyecta X a una representación Z de máxima varianza a través de la transformación $Z = PX$, de tal forma que $C_Z = \frac{1}{m}ZZ^T$ es una matriz diagonal. Las filas de P son llamadas las *componentes principales* de X (Shlens, 2014). La matriz C_Z se puede reescribir de la siguiente forma:

$$C_Z = \frac{1}{m}ZZ^T \quad (1.35)$$

$$= \frac{1}{m}(PX)(PX)^T \quad (1.36)$$

$$= \frac{1}{m}PXX^TP^T \quad (1.37)$$

$$= P\left(\frac{1}{n}XX^T\right)P^T \quad (1.38)$$

$$C_Z = PC_XP^T \quad (1.39)$$

Donde C_X , matriz de orden $n \times n$ y simétrica ($C_X = C_X^T$), es la matriz de covarianza de X . Considérese el siguiente teorema para una matriz simétrica A : $A = EDE^T$ donde D es diagonal² y E es una matriz de vectores propios de A organizados como columnas. Con esto último, la matriz de transformación P se selecciona de tal forma en que cada una de sus filas

²Matriz cuadrada cuyos elementos son nulos, excepto los pertenecientes a la diagonal principal.

p_i sea un vector propio de C_X . En consecuencia, las *componentes principales* de X son los vectores propios de $C_X = \frac{1}{m}XX^T$ y el i -ésimo valor de la diagonal de C_Z es la varianza de X a lo largo de p_i .

En definitiva, si se desea proyectar la matriz de datos X a un subespacio de dimensión k con $k < n$, se deben escoger los primeros k vectores propios $p_1, \dots, p_k$ de C_X . Luego, para representar $x^{(i)}$ en la nueva base, se necesita calcular el siguiente vector:

$$z^{(i)} = \begin{bmatrix} p_1 x^{(i)} \\ \vdots \\ p_k x^{(i)} \end{bmatrix} \in \mathbb{R}^k$$

1.4. Métricas de rendimiento

Cuando se habla del rendimiento de los algoritmos basados en problemas de Aprendizaje Supervisado, se está interesado en su capacidad de *generalización*, es decir, en el rendimiento que puedan concretar en torno a un conjunto de datos al cual no han sido expuestos durante el proceso de entrenamiento y así simular un escenario de comportamiento más cercano al mundo real. Es por eso que, para evaluar ese rendimiento, se utiliza un llamado *conjunto de pruebas*.

En función de lo anterior, y con el objetivo de evaluar las habilidades de un algoritmo en cuestión, se debe diseñar una medida cuantitativa o métrica de su rendimiento. Usualmente, esta medida es específica para el problema que está siendo solucionado por el algoritmo. Para un problema de regresión se pueden encontrar las siguientes:

- *Mean square error (MSE)*.

$$MSE = \frac{1}{m} \sum_{i=1}^m (y^{(i)} - \hat{y}^{(i)})^2 \quad (1.40)$$

- *Root mean square error (RMSE)*: Raíz cuadrada del MSE.

- *Coefficiente de determinación (R^2).*

$$R^2 = 1 - \frac{\sum_{i=1}^m (y^{(i)} - \hat{y}^{(i)})^2}{\sum_{i=1}^m (y^{(i)} - \bar{y})^2} \quad (1.41)$$

- *Index of agreement (IOA).*

$$IOA = 1 - \frac{\sum_{i=1}^m (y^{(i)} - \hat{y}^{(i)})^2}{\sum_{i=1}^m (|\hat{y}^{(i)} - \bar{y}| + |y^{(i)} - \bar{y}|)^2} \quad (1.42)$$

- *Mean Absolute Error (MAE).*

$$MAE = \frac{1}{m} \sum_{i=1}^m |y^{(i)} - \hat{y}^{(i)}| \quad (1.43)$$

- *Mean Bias Error (MBE).*

$$MBE = \frac{1}{m} \sum_{i=1}^m (y^{(i)} - \hat{y}^{(i)}) \quad (1.44)$$

- *Fractional Bias (FB).*

$$FB = 2 \frac{\sum_{i=1}^m (y^{(i)} - \hat{y}^{(i)})}{\sum_{i=1}^m (y^{(i)} + \hat{y}^{(i)})} \quad (1.45)$$

- *Normalized Mean Square Error (NMSE).*

$$NMSE = m \frac{\sum_{i=1}^m (y^{(i)} - \hat{y}^{(i)})^2}{\sum_{i=1}^m y^{(i)} \sum_{i=1}^m \hat{y}^{(i)}} \quad (1.46)$$

En las ecuaciones de arriba, los $y^{(i)}$ son los valores observados, $\bar{y}$ es la media de los valores observados e $\hat{y}^{(i)}$ son las predicciones realizadas por el modelo.

Capítulo 2

Estado del Arte

Este capítulo tiene como propósito el análisis y discusión de la literatura existente, desarrollada en torno al objetivo de predecir la concentración de Ozono troposférico.

En la Sección 2.1 se analizan los distintos modelos implementados para la predicción de los niveles de Ozono, así como también los detalles de su implementación. En la Sección 2.2 se presentan ciertos aspectos metodológicos identificados a lo largo del estado del arte, lo cual involucra por ejemplo: el horizonte de predicción, la selección de variables de entrada/salida y el manejo en general de los datos entre otros. Finalmente, en la Sección 2.3 se presenta un resumen con los estudios analizados.

2.1. Modelos

Diversos tipos de modelos se han estudiado e implementado para predecir los niveles de Ozono troposférico. En el caso de la Regresión Lineal, se le ha utilizado fundamentalmente como un *baseline* o modelo de referencia (Agirre-Basurko, Ibarra-Berastegi, y Madariaga, 2006; Biancofiore y cols., 2015; Salazar-Ruiz, Ordieres, Vergara, y Capuz-Rizo, 2008; Sousa, Martins, Alvim-Ferraz, y Pereira, 2007) con el cual, se comparan otros tipos de modelos más sofisticados y que han demostrado sobrellevar de mejor manera la complejidad no lineal de la formación del ozono troposférico (Lin, Trainer, y Liu, 1988).

2.1.1. Redes Neuronales

En función de lo anterior, los modelos predictivos basados en Redes Neuronales Artificiales son los que más han proliferado durante las últimas dos décadas. La gran mayoría de los casos se basan en Redes Feed Forward, haciéndolo mediante configuraciones bastante corrientes, es decir, a través de una arquitectura compuesta por tres capas: una de entrada, una capa oculta y una de salida.

Se observa además un bajo o nulo nivel de optimización de hiperparámetros¹ que pudiesen entregar un mejor rendimiento de la red. A lo más, solo se varía el número de neuronas ocultas dentro de un rango bastante bajo (ver tabla 2.1).

Redes Neuronales Recurrentes

En torno a las Redes Neuronales Recurrentes, estas no han sido exploradas de forma extensiva (ver tabla 2.2). Y cuando se ha hecho, no se aprovechan sus propiedades para procesar secuencias más largas de entrada, es decir, la historia o contexto del pasado para un determinado tiempo $t + 1$ no se extiende más allá de un sólo *timestep* o momento hacia atrás. En otras palabras, las arquitecturas exploradas son siempre *one-to-one*. Además, se repite lo que acontece con las Redes Feed Forward, en que existe poca optimización de hiperparámetros.

El tipo de Red Neuronal Recurrente más frecuente es la ELMAN, la cual es comparada con redes Feed Forward. En Salazar-Ruiz y cols. (2008), se obtiene un rendimiento menor que la red Feed Forward explorada en ese trabajo. Caso contrario ocurre en Biancofiore y cols. (2015). En tanto que en Gorai y Mitra (2017), no se reportan grandes diferencias de rendimiento, considerando las diversas topologías de red exploradas.

Caso excepcional y más interesante es el que se presenta en Ribeiro y Alquézar (2002), donde

¹Los hiperparámetros son configuraciones que no son aprendidas directamente por un algoritmo de aprendizaje automático, pero que controlan ciertos aspectos de su comportamiento, tales como el costo en tiempo y memoria requeridos para su ejecución. Algunos de estos hiperparámetros también afectan la calidad del modelo resultante del proceso de entrenamiento así como también su habilidad de predecir resultados correctos sobre datos nuevos (Goodfellow y cols., 2016). En el caso de las redes neuronales, como hiperparámetros se tienen por ejemplo: el número de capas ocultas, la cantidad de neuronas por las cuales están compuestas estas últimas, o el *learning rate*.

Tabla 2.1: Aspectos arquitecturales y otras configuraciones de las Redes Feed Forward.

Estudio	Capas Ocultas	Neuronas ocultas	Función de activación
Gorai y Mitra (2017)	1	2, ..., 12	
Biancofiore y cols. (2015)	1	4	
Luna y cols. (2014)	1	20	
Hájek y Olej (2012)	1	3, ..., 8	
Özbay y cols. (2011)	2,3	15	Tanh
Ortiz-García y cols. (2010)	1		
Coman y cols. (2008)	1	20	Tanh
Salazar-Ruiz y cols. (2008)	1	1, ..., 29	
Al-Alawi y cols. (2008)			
Sousa y cols. (2007)	1	8	Sigmoidal
Dutot y cols. (2007)	1	1, ..., 8	Tanh
Agirre-Basurko y cols. (2006)	1		Tanh
Wang y cols. (2003)	1		Gaussian
Abdul-Wahab y Al-Alawi (2002)	1		
Gardner y Dorling (2000)	2		Tanh
Jorquera y cols. (1998)	1		Sigmoidal
Yi y Prybutok (1996)	1	4	Sigmoidal

hay más conciencia respecto de las propiedades de las redes recurrentes. En ese trabajo, se utilizan redes recurrentes LSTM a través de una arquitectura tipo *many-to-one*, donde las secuencias de entrada están compuestas por múltiples *timesteps* del pasado de las distintas variables predictoras (Ozono y variables meteorológicas). El rendimiento reportado de las redes LSTM es superior al de las redes ELMAN y Feed Forward implementadas en el mismo trabajo.

Tabla 2.2: Modelos basados en Redes Neuronales Recurrentes.

Estudio	Tipo	Neuronas ocultas
Gorai y Mitra (2017)	ELMAN	2, ..., 12
Biancofiore y cols. (2015)	ELMAN	4
Salazar-Ruiz y cols. (2008)	ELMAN	1, ..., 29
Ribeiro y Alquézar (2002)	LSTM	1

2.1.2. Support Vector Machines

Aunque en menor medida que las Redes Neuronales, las Support Vector Machines (SVM) también han sido utilizadas para predecir la concentración de Ozono Troposférico (ver tabla 2.3). En dos estudios se ha reportado un mejor rendimiento al compararse con el alcanzado mediante Redes Feed Forward (Hájek y Olej, 2012; Ortiz-García y cols., 2010), optimizando diversos hiperparámetros según el kernel utilizado. En Hájek y Olej (2012) el kernel polinomial es el que logra los mejores resultados. En el resto de los estudios (Luna y cols., 2014; Salazar-Ruiz y cols., 2008) no se entregan detalles acerca de la configuración de las SVM implementadas. En Salazar-Ruiz y cols. (2008) el rendimiento de la SVM es menor que el de las redes neuronales implementadas en el mismo estudio. En tanto que en Luna y cols. (2014), no se presentan diferencias significativas entre un tipo de modelo u otro.

Tabla 2.3: Modelos basados en SVM.

Estudio	Kernel	Parámetros sintonizados
Luna y cols. (2014)	-	-
Hájek y Olej (2012)	lineal, rbf, polinomial	$C, \gamma, \epsilon, \beta$
Ortiz-García y cols. (2010)	rbf	C, γ, ϵ
Salazar-Ruiz y cols. (2008)	-	-

2.1.3. ARIMA

Varias investigaciones han utilizado técnicas de series de tiempo para predecir la concentración de Ozono troposférico, especialmente en las etapas más tempranas del desarrollo de las soluciones para este problema, teniendo como objetivo principal el predecir el máximo diario de los niveles de Ozono (Simpson y Layton, 1983). Sin embargo, los resultados obtenidos no parecen ser satisfactorios y su utilización ha sido progresivamente opacada por otras técnicas de mayor complejidad, que no dependen solamente de los valores pasados del Ozono y que pueden incorporar el efecto de otras variables de carácter exógeno.

En Robeson y Steyn (1990), se desarrollan tres modelos: uno determinista/estocástico univariado, ARIMA y una regresión lineal bivariada usando el Ozono y la temperatura. Este último

es el que obtiene los mejores resultados, mientras que el modelo ARIMA (con parámetros $p = 1, d = 1, q = 0$) es apenas mejor que un modelo persistente². El modelo determinista/estocástico univariado es el de peor desempeño. En Yi y Prybutok (1996) se compara el rendimiento de una red neuronal Feed Forward, el de una regresión lineal y el de un modelo ARIMA. En este caso, la red neuronal es la que obtiene los mejores resultados. De manera similar en Chaloulakou, Assimacopoulos, y Lekkas (1999), el modelo ARIMA(0, 1, 2) es inferior en rendimiento a los tres basados en regresión lineal e incluso, inferior a un modelo persistente. Otros estudio notables son los de K. Kumar, Yadav, Singh, Hassan, y Jain (2004); U. Kumar y Jain (2010) desarrollados en la India, aunque los modelos ARIMA desarrollados no son comparados con otros de distinto tipo. En K. Kumar y cols. (2004) el mejor modelo es el ARIMA(1, 0, 1) y en U. Kumar y Jain (2010) es el ARIMA(0, 0, 1).

2.2. Aspectos Metodológicos

2.2.1. Variables de Entrada y Salida

Dado que existe un marco teórico bastante desarrollado en torno a los factores que participan en la formación del Ozono Troposférico, los conjuntos de variables de entrada que se suelen utilizar, son relativamente homogéneos a lo largo de los estudios aquí evaluados.

Lo anterior eso si, está sujeto a la existencia de registros históricos de las mediciones correspondientes, así como también de las condiciones geográficas y climatológicas propias del lugar donde se pretende realizar el estudio, ya que esto determina la presencia o ausencia de ciertas variables, o porque para el investigador le es de interés explorar la relación con ciertos atributos que son característicos del lugar en estudio.

En la tabla 2.5 se muestran las variables entrada y salida presentes en cada estudio aquí analizado.

²Un modelo persistente es aquel en donde la predicción $\hat{y}$ de una variable objetivo para el tiempo $t + 1$ se obtiene como el valor observado de la variable en el tiempo t , es decir, $\hat{y}^{(t+1)} = y^{(t)}$

Variables de Entrada

A grandes rasgos, las variables de entrada que se utilizan se clasifican principalmente en dos grandes grupos:

- Variables químicas: Tienen relación con la concentración de otros gases presentes en la atmósfera, y que actúan como precursores químicos en la generación (directa o indirectamente) del Ozono. Entre ellos se destacan los óxidos de nitrógeno (NO, NO₂, NO_x), Compuestos Orgánicos Volátiles (COV), Monóxido de Carbono (CO), Metano (CH₄), Dióxido de Azufre (SO₂), además de la concentración misma de Ozono (correspondiente a registros de periodos de tiempo anteriores al cual se quiere realizar la predicción).
- Variables meteorológicas: Estas actúan como catalizadores o inhibidores de la formación de Ozono. La principal de ellas es la Radiación Solar (RS). También están la Temperatura ambiente (TEMP) del aire, la Humedad relativa del aire (HR) del aire, la velocidad y dirección del viento, y la presión atmosférica.

Variables de Salida

Respecto de la salida, distintos enfoques se han implementado. Uno de ellos, es la predicción de la concentración máxima diaria, ya sea en promedios de 1 hora (la más frecuente) o de 8 horas, acorde a las regulaciones recomendadas por organismos internacionales e implementadas en los diversos países. El otro enfoque, es el de predecir la concentración horaria del día.

2.2.2. Horizonte de Predicción

Por un asunto de requerimientos prácticos (que las autoridades correspondientes den aviso público de las alertas por episodios críticos para la mitigación efectos adversos, con cierta anticipación), las predicciones se deben realizar antes de medianoche para el día siguiente,

es decir, el horizonte de predicción mínimo es de 1 día o en términos más concretos, para predecir la concentración de Ozono del día $t + 1$, se deben utilizar los datos del día t como mínimo.

Por lo general, en los estudios aquí analizados, lo anterior es cierto. Sin embargo, hay otros en que el horizonte de predicción es menor a un día o de pocas horas (Agirre-Basurko y cols., 2006; Ortiz-García y cols., 2010) o al menos lo toman como alternativa (Biancofiore y cols., 2015). También se da el caso en el que para ciertas variables predictoras, se utilizan los valores del tiempo $t + 1$ para predecir el Ozono en el instante $t + 1$, ya sea utilizando registros medidos y concretos o mediante valores basados en predicciones (Dutot y cols., 2007; Jorquera y cols., 1998; Ribeiro y Alquézar, 2002).

Otros estudios simplemente se limitan a modelar la relación existente entre el Ozono y las demás variables, y no a predecir hacia adelante temporalmente en el futuro, es decir, el Ozono como objetivo y las variables de entrada corresponden al mismo instante t de tiempo (Abdul-Wahab y Al-Alawi, 2002; Gardner y Dorling, 2000; Luna y cols., 2014; Wang y cols., 2003).

2.2.3. Validación I: Manipulación de datos

La regularización es cualquier modificación hecha a un algoritmo de aprendizaje automático, con tal de reducir su error de generalización pero no el de entrenamiento. Este error se entiende como el valor esperado del error que comete el algoritmo de Aprendizaje Automático al evaluar datos a los cuales no ha sido expuesto durante el proceso de entrenamiento, es decir, el conjunto de pruebas (Goodfellow y cols., 2016).

Una de los métodos de regularización para redes neuronales, más populares y fáciles de implementar, es el *Early Stopping*. Utilizado por (Agirre-Basurko y cols., 2006; Coman y cols., 2008; Dutot y cols., 2007; Gardner y Dorling, 2000; Salazar-Ruiz y cols., 2008), este método puede describirse de la siguiente forma:

1. Dividir los datos de entrenamiento en dos conjuntos: uno de entrenamiento y otro de validación, por ejemplo en una proporción de 2 a 1.

2. Entrenar el algoritmo sobre el conjunto de entrenamiento y evaluar el error sobre el conjunto de validación cada cierto tiempo.
3. Detener el entrenamiento tan pronto el error en el conjunto de validación sea mayor que el obtenido en la última evaluación. Aunque esto tiende a relajarse estableciendo una cierta “paciencia”, es decir, una determinada cantidad de iteraciones del proceso de entrenamiento en la cual se tolera el no encontrar una disminución del error de validación.
4. Usar los pesos que tenía la red en el paso anterior como resultado del proceso de entrenamiento.

Este enfoque usa el conjunto de validación para anticipar el comportamiento del algoritmo en un caso de uso real (o conjunto de pruebas), asumiendo que el error en ambos será similar. El error de validación es usado como una estimación del error de generalización (Prechelt, 1998).

También, la presencia o ausencia de una variable depende del nivel de mejora que induce sobre la calidad de las predicciones realizadas por el modelo. Estudios como (Al-Alawi y cols., 2008; Özbay y cols., 2011; Sousa y cols., 2007), emplean Principal Component Analysis (PCA) para disminuir el número de características de entrada y así simplificar los modelos finales. Análogamente, se suelen utilizar técnicas de selección de variables basadas en filtros, como la correlación con la variable objetivo (Hájek y Olej, 2012; Wang y cols., 2003) o basados en la minimización del *MSE* (Dutot y cols., 2007).

2.2.4. Validación II: Medidas de Rendimiento

En la tabla 2.4, se muestran las medidas de rendimiento utilizadas para evaluar los distintos modelos implementados por los estudios analizados.

Tabla 2.4: Medidas de rendimiento empleadas en los distintos estudios analizados.

Estudio	Medida de rendimiento							
	<i>RMSE</i>	<i>MSE</i>	<i>NMSE</i>	<i>IOA</i>	<i>FB</i>	<i>MAE</i>	<i>MBE</i>	<i>R</i> ²
Gorai y Mitra (2017)	•			•	•			•
Biancofiore y cols. (2015)			•		•			
Luna y cols. (2014)	•							
Hájek y Olej (2012)	•							
Özbay y cols. (2011)	•							•
Ortiz-García y cols. (2010)		•						•
U. Kumar y Jain (2010)			•		•	•		
Coman y cols. (2008)	•			•		•	•	•
Salazar-Ruiz y cols. (2008)	•			•	•	•	•	
Al-Alawi y cols. (2008)	•							
Sousa y cols. (2007)	•			•		•	•	
Dutot y cols. (2007)	•			•		•	•	
Agirre-Basurko y cols. (2006)			•		•			
K. Kumar y cols. (2004)	•		•		•	•		
Wang y cols. (2003)						•		
Gómez y cols. (2003)	•							
Abdul-Wahab y Al-Alawi (2002)						•	•	•
Gardner y Dorling (2000)	•			•		•	•	•
Chaloulakou y cols. (1999)	•			•		•		
Jorquera y cols. (1998)	•			•				
Yi y Prybutok (1996)								
Robeson y Steyn (1990)	•			•		•		

2.3. Resumen estudios analizados

Tabla 2.5: Resumen de los estudios analizados.

Estudio	Objetivo	Predictores	Lugar
Gorai y Mitra (2017)	Máximo diario ^b	HR, TEMP, VV, DV, NO ₂ , O ₃	Kolkata, India.
Biancofiore y cols. (2015)	Concentración hora- ria.	TEMP, HR, PA, VV, DV, NO ₂ , O ₃ , UVA	Pescara, Italia.
Luna y cols. (2014)	Concentración hora- ria.	TEMP, HR, VV, RS, NO ₂ , NO, NO _x , CO	Río de Janeiro, Bra- sil.
Hájek y Olej (2012)	Promedio diario.	HR, RS, variable <i>dummy</i> mes del año, NO ₂ , NO, NO _x , O ₃	Pardubice, República Checa.
Özbay y cols. (2011)	Concentración hora- ria.	TEMP, HR, VV, DV, PP, PA, RS, MP10, SO ₂ , O ₃ , NO ₂ , NO, NO _x , CO, HNM	Dilovasi, Turquía.
Ortiz-García y cols. (2010)	Concentración hora- ria.	O ₃ , TEMP, RS	Madrid, España.
U. Kumar y Jain (2010)	Promedio diario.	O ₃	Delhi, India.
Coman y cols. (2008)	Concentración hora- ria.	TEMP, RS, HR, duración luz solar, VV, DV, O ₃ , NO ₂	París, Francia.
Salazar-Ruiz y cols. (2008)	Máximo diario ^a	TEMP, RS, PA, VV, DV, O ₃ , NO ₂ , NO, NO _x , CO	Mexicali, Calexico (Frontera México- EEUU).
Al-Alawi y cols. (2008)	Concentración hora- ria.	TEMP, HR, RS, VV, DV, CH ₄ , HNM, NO ₂ , NO, SO ₂ , CO ₂ , CO	Ciudad de Kuwait, Kuwait.

Continúa en la siguiente página

Tabla 2.5 – continuación desde la página anterior

Estudio	Objetivo	Predictores	Lugar
Sousa y cols. (2007)	Concentración hora- ria.	TEMP, HR, VV, O ₃ , NO ₂ , NO	Oporto, Portugal.
Dutot y cols. (2007)	Máximo diario: ^a	TEMP, VV, DV, Nubosi- dad, O ₃	Centro de Francia.
Agirre-Basurko y cols. (2006)	Concentración hora- ria.	VV, DV, TEMP, HR, PA, RS, gradiente térmico (°C), O ₃ , NO ₂ , variables relativas al tráfico vehicular	Bilbao, España.
K. Kumar y cols. (2004)	Máximo diario: ^a	O ₃	Brunei Darussalam, India.
Wang y cols. (2003)	Máximo diario: ^a	O ₃ , NO ₂ , NO _x , RS, TEMP, VV	Hong Kong.
Gómez y cols. (2003)	Máximo diario.	TEMP, O ₃ , VV, Nubosidad	Este de Austria
Abdul-Wahab y Al- Alawi (2002)	Máximo diario: ^a	TEMP, HR, RS, VV, DV, CH ₄ , HNM, NO ₂ , NO, SO ₂ , CO ₂ , CO	Ciudad de Kuwait, Kuwait
Gardner y Dorling (2000)	Concentración hora- ria	VV, DV, fecha juliana ³ , nu- bosidad, visibilidad, TEMP, hora del día	(Bristol, Edinburgh, Eskdalemuir Leeds, Southampton), Reino Unido
Chaloulakou y cols. (1999)	Máximo diario: ^a	O ₃ , TEMP, NO ₂ , CO, VV, DV	Atenas, Grecia.
Jorquera y cols. (1998)	Máximo diario: ^a	O ₃ , TEMP	Santiago, Chile.

Continúa en la siguiente página

³La fecha juliana, día juliano o DJ (JD, por sus siglas en inglés) es el número de días y fracción transcurridos desde el mediodía del 1.º de enero del año 4713 a. C

Tabla 2.5 – continuación desde la página anterior

Estudio	Objetivo	Predictores	Lugar
Yi y Prybutok (1996)	Máximo diario ^a	O ₃ , NO, NO ₂ , NO _x , CO ₂ , VV, DV, variable <i>dummy</i> si el día es festivo o hábil	Dallas, Texas, EE.UU.
Robeson y Steyn (1990)	Máximo diario ^a	O ₃ , TEMP	British Columbia, Canadá.

Compuestos Orgánicos Volátiles (COV), Óxidos de Nitrógeno (NO_x), Temperatura ambiente (TEMP), Humedad relativa del aire (HR), Presión atmosférica (PA), Velocidad del viento (VV), Dirección del viento (DV), Radiación Solar (RS), Precipitaciones (PP), Monóxido de Nitrógeno (NO), Dióxido de Nitrógeno (NO₂), Metano (CH₄), Monóxido de Carbono (CO), Dióxido de Carbono (CO₂), Dióxido de Azufre (SO₂), Material Particulado (MP), Hidrocarburos no metánicos (HNM).

^a Concentración de 1 hora.

^b Concentración de 8 horas.

Capítulo 3

Metodología y Propuesta

En este capítulo, se presentan las características del caso de estudio que atañen a este trabajo, es decir, la predicción del Ozono troposférico en la ciudad de Santiago, así como también la estrategia de resolución del problema asociado.

En la Sección 3.1 se contextualiza la situación relativa a la contaminación del aire en la ciudad de Santiago, presentando además al Sistema de Información Nacional de Calidad del Aire (SINCA) como la fuente principal de los datos a utilizar, el cual pone a disposición pública los registros generados por las estaciones de monitoreo de la calidad del aire en Santiago. En la Sección 3.2 se detallan las herramientas de software a ser utilizadas, ya sea para la manipulación de los datos o confección de los modelos. También se realiza un análisis descriptivo de los datos, para luego explicitar los objetivos del análisis predictivo de los experimentos. Posteriormente, se describe el flujo de preparación y generación de los conjuntos de datos, tanto para el entrenamiento de los modelos como también para la validación de los mismos. Y finalmente, en la Sección 3.3 se presentan los detalles de implementación de los modelos, con su estructura y parámetros a optimizar, además del modo de evaluación del rendimiento de estos últimos.

3.1. Caso de Estudio

3.1.1. Antecedentes

La ciudad de Santiago se localiza en la Región Metropolitana (RM), zona central de Chile, en los 33°26' de latitud Sur y 70°41' de longitud Oeste, a 520 metros de altitud sobre el nivel del mar. Se caracteriza por ser una de las ciudades más contaminadas de Sudamérica, fenómeno derivado principalmente de las características geográficas y meteorológicas de la región en la cual se ubica, las cuales propician la formación y acumulación de contaminantes en la atmósfera.

Características geográficas

La ciudad se ubica en una depresión o cuenca, caracterizada por estar rodeada de un anillo de relieves montañosos en casi todo su entorno.

Por el oriente de la ciudad, la Cordillera de los Andes con cerros que superan los 3200 m., perturba la circulación general de la atmósfera de la zona central, produciendo frecuentemente el desarrollo de una baja presión, que genera condiciones de aumento de la temperatura y una disminución de la humedad relativa en la cuenca, fortaleciendo los fenómenos de inversión térmica, que en otras palabras genera una masa de aire caliente que tapa a la ciudad desde una altura que varía entre los 200 y 1000 m. (invierno y verano respectivamente), dificultando la elevación y dispersión de los contaminantes (Ministerio del Medio Ambiente Chile, 2015).

Y por el poniente de la ciudad se encuentra la Cordillera de la Costa, que alcanza alturas por sobre los 2000 m., la cual aleja y aísla a la ciudad del clima oceánico, dotándola de su característica mediterraneidad (Ministerio del Medio Ambiente Chile, 2015).

Al final, dicho anillo montañoso impone fuertes restricciones a la circulación de vientos y, por ende, a la renovación del aire al interior de la cuenca. Por ello, en épocas de estabilidad atmosférica, los contaminantes quedan atrapados dentro de la cuenca que alberga a la ciudad

de Santiago (Ministerio del Medio Ambiente Chile, 2015).

Características meteorológicas

El clima se caracteriza por ser del tipo mediterráneo, cuya principal cualidad es la presencia de una estación seca prolongada y un invierno que concentra la mayoría de las precipitaciones. La temperatura media anual es de 13,9°C, en tanto que el mes más cálido corresponde al mes de Enero, alcanzando una temperatura media de 22,1°C (con máximas superiores a los 30°C durante el día) y altos índices de radiación UV. El mes más frío corresponde al mes de Julio con 7,7°C en promedio. Durante el invierno, las lluvias son relativamente escasas, con un promedio anual de 356,2 mm y concentradas durante las temporadas invernales, principalmente entre los meses de mayo a septiembre, no contribuyendo de manera significativa a la limpieza de la atmósfera (Biblioteca del Congreso Nacional de Chile, s.f.).

Desarrollo urbano

Santiago concentra la mayoría de la población de la RM, siendo la ciudad más grande y poblada de Chile, con un explosivo crecimiento iniciado a mediados del siglo XX, producido por la migración campo-ciudad, dadas las características físicas, socio económicas, de oportunidad y urbanas que presenta. Según los resultados preliminares del censo del año 2017, la RM cuenta con una población de 7.036.792 habitantes (Instituto Nacional de Estadísticas, 2017). Considerando que posee una superficie de 15.403,20 km^2 , equivalentes al 2,0 %, del territorio nacional, convierte a la RM en una zona de alta densidad demográfica, con 456,84 habitantes por km^2 .

Sin embargo, este crecimiento lleva consigo una serie de externalidades negativas, que derivan del comportamiento humano dentro de la ciudad y que inciden sobre la contaminación del aire. En el gráfico 3.1, se observa el aporte según tipo de fuente de emisión o sector de actividad humana, a la producción de los gases contaminantes Óxidos de Nitrógeno (NO_x), Compuestos Orgánicos Volátiles (COV) y Monóxido de Carbono (CO), precursores del Ozono troposférico. Se observa que:

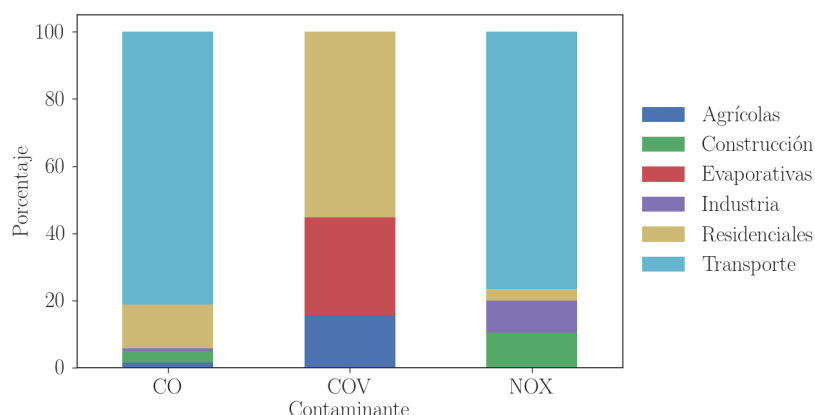


Figura 3.1: En base a los datos del Informe *Actualización y sistematización del inventario de emisiones de contaminantes atmosféricos en la Región Metropolitana* elaborado por Universidad de Santiago de Chile (2014)

- El transporte es responsable por más del 75 % de las emisiones de los Óxidos de Nitrógeno. Según la Encuesta Anual de Parque Vehicular 2016 del INE¹, un total de 4.960.945 vehículos circularon en Chile durante 2016, con la Región Metropolitana teniendo el mayor parque automotriz, equivalente a 1.968.954 vehículos (39,7 % del total).
- Para los COV, el tipo de fuente que más aporta es el residencial (55,2 %), el cual involucra la calefacción domiciliaria (por leña, gas licuado, parafina) y solventes y pinturas. Luego siguen las fuentes de tipo evaporativas (29,2 %), que incluyen lavasecos, pintado de vehículos, y fugas comerciales entre otros. Finalmente, las actividades agrícolas (15,5 %), que involucran quemas agrícolas, incendios forestales, crianza de animales, rellenos sanitarios y plantas de tratamientos de agua.
- En el caso del Monóxido de Carbono, el transporte aporta con un 81,2 % de las emisiones, seguido por el sector residencial con un 13,1 %.

¹<http://www.ine.cl/prensa/detalle-prensa/2017/05/22/m%C3%A1s-de-4-9-millones-de-veh%C3%ADculos-circularon-en-el-pa%C3%ADs-durante-2016>

3.1.2. Fuentes de Datos

El Ministerio del Medio Ambiente de Chile (MMA) cuenta con una red de 11 estaciones de monitoreo de la calidad del aire en la Región Metropolitana, las cuales miden y registran de forma continua, los niveles de diversas variables contaminantes y meteorológicas (ver figura 3.2). Tales registros, son dispuestos públicamente través del Sistema de Información Nacional de Calidad del Aire (SINCA)², plataforma web en línea.

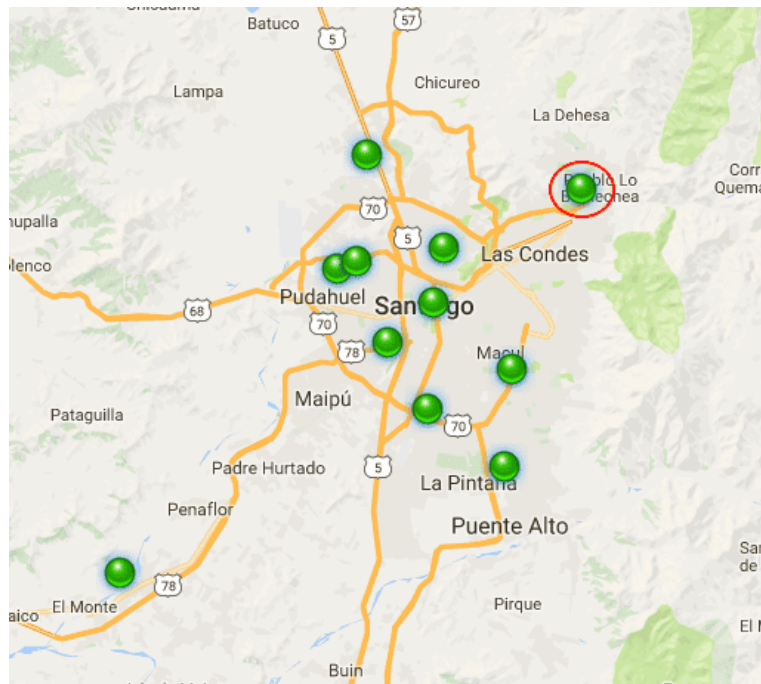


Figura 3.2: Estaciones de monitoreo de la calidad del aire desplegadas en la Región Metropolitana. Fuente: SINCA.

Para la realización del estudio, se tomó como caso base a la estación de monitoreo de la comuna de Las Condes, localizada en el sector oriente de la ciudad de Santiago (rodeada por un círculo rojo en la figura 3.2). Esta estación cuenta con representatividad poblacional, es decir, está calificada para verificar la norma de calidad del aire del Ozono en la Región Metropolitana, a partir de las mediciones que se realicen en ella (Ministerio de Salud, 2006).

²<http://sinca.mma.gob.cl/index.php/region/index/id/M>

Recolección de Datos

Los registros asociados a las mencionadas variables, pueden ser descargados³ en archivos de texto en formato csv, los cuales son presentados como promedios aritméticos de 1 hora y la inclusión de nuevos datos es llevada a cabo cada 1 hora.

Ahora bien, dado que las estaciones de monitoreo del SINCA no hacen registro de la radiación solar ultravioleta (UV), se utilizaron los datos reportados por la estación meteorológica del Departamento de Física de la Universidad de Santiago de Chile (USACH), localizada en la comuna de Estación Central, en el centro de la ciudad de Santiago. Allí, desde el 2001 hasta la actualidad, se realiza un monitoreo continuo del Índice UV, la Radiación Ultravioleta A y la Radiación Ultravioleta B, cuyos registros son actualizados cada 3 minutos. Tales datos se encuentran organizados en archivos de texto, uno por cada día, y disponibles en las siguientes urls:

- Desde el año 2001 hasta el 2008: http://ambiente.usach.cl/datos_historicos/
- Desde el año 2009 hasta el presente: http://ambiente.usach.cl/Datos_horarios/

En la tabla 3.1 se muestra un resumen de las características de las variables recolectadas.

Tabla 3.1: Variables de entrada.

Parámetro	Unidad	Fecha Primer Registro	Fecha Último Registro
Ozono (O ₃)	ppb	1997-04-02	2017-09-16
Monóxido de Nitrógeno (NO)	ppb	2000-01-01	2017-09-15
Dióxido de Nitrógeno (NO ₂)	ppb	2005-03-17	2017-09-16
Óxidos de Nitrógeno (NO _x)	ppb	2000-01-01	2017-09-15
Monóxido de Carbono (CO)	ppm	1997-04-05	2017-09-16
Radiación Ultravioleta A (UVA)	mW/cm ²	2001-05-11	2017-09-16
Radiación Ultravioleta B (UVB)	uW/cm ²	2001-05-11	2017-09-16
Temperatura ambiente (TEMP)	°C	2003-12-14	2017-09-16
Humedad relativa del aire (HR)	%	2003-12-14	2017-09-16
Velocidad del viento (VV)	m/s	2003-12-14	2016-06-19
Dirección del viento (DV)	°	2003-12-14	2016-06-19

³<http://sinca.mma.gob.cl/index.php/estacion/index/id/233>

3.2. Configuración Experimental

3.2.1. Requerimientos de Software

El desarrollo de este trabajo se realizó bajo un ambiente Linux, específicamente, utilizando como distribución Arch Linux⁴ x86_64.

Lenguajes de Programación

Python⁵ es un lenguaje de programación interpretado, de código abierto, con una sintaxis simple e intuitiva, con la flexibilidad de utilizar variados paradigmas de programación: procedural, orientado a objetos, scripting. Dispone de un gran abanico de librerías, ya sea de manera estándar como también creadas por terceras partes, que permiten hacer mucho con poco esfuerzo. Posee un excelente manejo de estructuras de datos, tales como listas, diccionarios (hashes) o tuplas. En los últimos años ha ido ganando terreno en el área de la ciencia de los datos, dada la riqueza de las herramientas disponibles.

La mayoría de las distribuciones de Linux, cuentan desde un inicio con una versión de Python ya instalada. Arch Linux proviene con las versiones 3.x (versión por defecto) y 2.x. Se utiliza además `virtualenv`⁶, la cual es una herramienta para crear ambientes de desarrollo aislados de Python, posibilitando la instalación de paquetes sin tener que interferir con otros sistemas o aplicaciones.

La versión de Python utilizada en este trabajo es la 3.6.2. Los principales paquetes de Python utilizados son los siguientes:

- **Numpy**⁷: Es la librería central para la computación científica en Python. Provee objetos tipo arreglos multidimensionales de datos de alto rendimiento, además de herramientas asociadas a su manejo.

⁴<http://www.archlinux.org/>

⁵<http://www.python.org/>

⁶<https://virtualenv.pypa.io/en/stable/>

⁷<http://www.numpy.org/>

- **Pandas**⁸: Librería diseñada para la manipulación y análisis de datos. En concreto, ofrece estructuras de datos para manipular tablas numéricas y series de tiempo.
- **Scikit Learn**⁹: Librería para Aprendizaje Automático, ofreciendo modelos de clasificación, regresión, clustering y Support Vector Machines (SVM) entre otros.
- **TensorFlow**¹⁰: Es una librería desarrollada por Google para trabajar en Aprendizaje Automático. Caracterizada principalmente por su capacidad de construir y entrenar redes neuronales de variados tipos.
- **Keras**¹¹: Es una librería que provee de una API de alto nivel para trabajar con redes neuronales. Capaz de funcionar sobre TensorFlow, permite crear modelos de redes neuronales de manera sencilla e intuitiva.
- **Matplotlib**¹²: Librería para trazado de gráficos.
- **Seaborn**¹³: Es una librería de visualización estadística basada en Matplotlib.

R¹⁴ es un lenguaje de programación y un ambiente de trabajo para realizar computación estadística y gráfica. De código abierto, provee de una gran variedad de herramientas estadísticas (modelado lineal y no lineal, pruebas estadísticas clásicas, análisis serie de tiempo, clasificación, etc.). Usado de forma extensa en el área de la ciencia de los datos. Funcionalmente extensible mediante paquetes creados por terceras partes. Entre los paquetes de R a utilizar se encuentran:

- **Forecast**¹⁵: Ofrece métodos y herramientas para realizar, presentar y analizar predicciones de series tiempo univariadas, como por ejemplo, modelamiento ARIMA.

⁸<http://pandas.pydata.org/>

⁹<http://scikit-learn.org/>

¹⁰<http://www.tensorflow.org/>

¹¹<http://keras.io/>

¹²<http://matplotlib.org/>

¹³<http://seaborn.pydata.org/>

¹⁴<https://www.r-project.org/>

¹⁵<https://cran.r-project.org/web/packages/forecast/index.html>

3.2.2. Análisis Descriptivo de los Datos

Antes de que los datos recabados por las estaciones de monitoreo, sean aceptados en una base de datos final, los datos erróneos deben ser filtrados o extraídos. Este proceso de filtrado recibe el nombre de validación, y corresponde a la verificación de la exactitud, integridad y consistencia de la información generada, tomando en consideración criterios cuantitativos y cualitativos¹⁶.

Los datos mostrados en SINCA pueden pertenecer a una de las siguientes categorías:

- Registros no validados: Corresponde a la información recibida en línea, de forma automática, desde las estaciones de monitoreo que se encuentren conectadas al sistema.
- Registros preliminares: Corresponden a datos validados en una primera instancia, efectuada por los operadores de la estación donde se corrigen los datos desfasados o se eliminan aquellos que corresponden a fallas o mantenciones (como por ejemplo la calibración de los instrumentos de medición).
- Registros oficiales: Corresponden a aquellos registros previamente validados por los operadores de las estaciones (registros preliminares), que pasan por un segundo proceso de validación realizado por algún organismo estatal que tenga competencias sobre los datos.

En lo que respecta a los datos descargados, estos corresponden al periodo que va desde el 1 de Enero del 2006 hasta el 31 de Diciembre del 2016. En las gráficas de las series de tiempo de las distintas variables que se mostrarán a continuación, en color azul se presentan los registros que pertenecen a la categoría de registros oficiales, en color naranja los que pertenecen a la categoría de preliminares y en color verde, los registros no validados. Por motivos prácticos y de simplicidad, se grafican los promedios diarios de cada variable.

En la figura 3.3 se puede observar la serie de tiempo para el Ozono troposférico, la cual presenta un notorio comportamiento cíclico, en donde la concentración del gas oscila entre periodos de altos y bajos niveles a lo largo de un año. En general, la mayoría de los registros

¹⁶<http://sinca.mma.gob.cl/index.php/pagina/index/id/validacion>

se encuentran validados, a excepción de un pequeño periodo de dos meses (Julio-Agosto 2015) en donde simplemente no hay registros de tipo alguno.

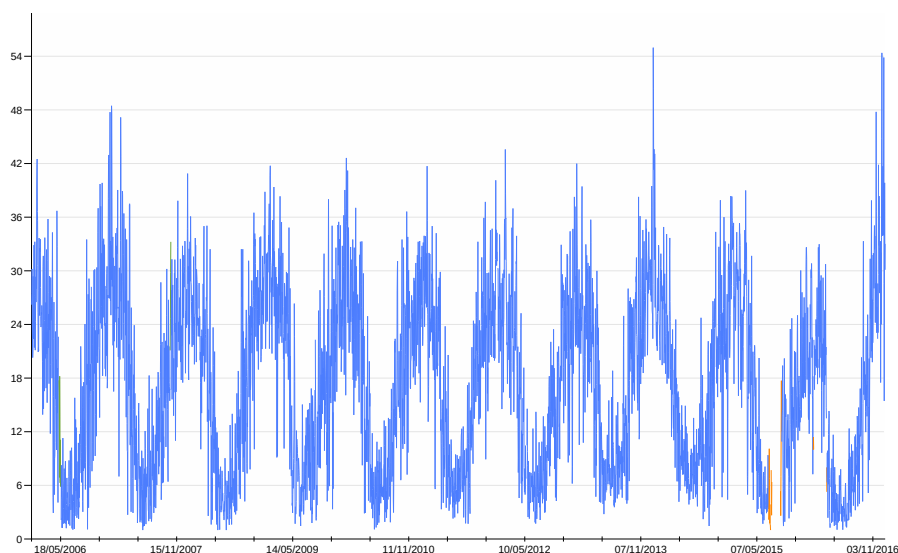


Figura 3.3: Serie de tiempo del Ozono troposférico.

Similar comportamiento cíclico se puede observar en las otras variables contaminantes (Monóxido de Nitrógeno, Dióxido de Nitrógeno (NO_2), Óxidos de Nitrógeno (NO_x) y Monóxido de Carbono (CO)), cuyas gráficas pueden verse en la figura 3.4. Eso sí, se observa que una importante cantidad de datos correspondientes a los óxidos de nitrógeno en general, solo se encuentran en estado preliminar (periodo cercano a los tres años, entre los años 2012 y 2015) y que dentro del año 2012, entre los meses de Julio y Octubre, los datos no están validados. En menor medida se aprecia los mismo para el Monóxido de Carbono.

En tanto que las variables de tipo meteorológico, cuyas series de tiempo se pueden ver en la figura 3.5, también presentan cierto comportamiento cíclico.

En lo que respecta a los datos de la radiación solar ultravioleta, las series de tiempo correspondientes se pueden observar en la figura 3.6, cuyas gráficas son resultado de elaboración propia. En ambas variables (UVA y UVB), también se puede observar un comportamiento cíclico como el descrito previamente con las otras variables. En torno a la Radiación Ultravioleta A (UVA), se observa que desde el 2012 en adelante parece haber un comportamiento anómalo o inusual si es que se le compara con lo ocurre desde el 2011 hacia atrás, donde se observa un patrón oscilatorio evidente. Esto último podría deberse a un mal funcionamiento

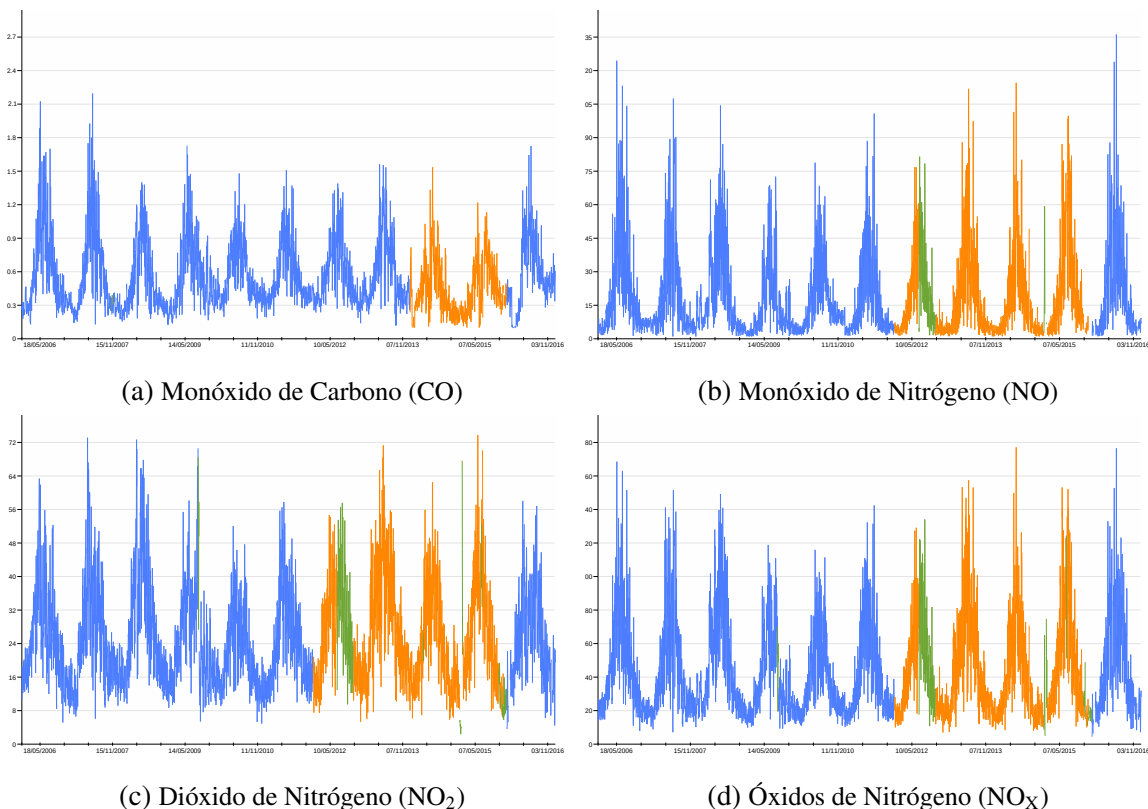


Figura 3.4: Series de tiempo contaminantes atmosféricos.

de los instrumentos de medición, lo que da pie al tener que recordar que estos registros son obtenidos desde la estación de monitoreo en la USACH y que, al menos en lo que se deja explícito¹⁷, no son parte de proceso de revisión o validación alguno. En lo relativo a la Radiación Ultravioleta B (UVB), no se evidencian anomalías importantes, a excepción de un solo *outlier*.

Ahora bien, en la figura 3.7 se grafica la serie de tiempo del Ozono troposférico, destacándose los periodos de mayor concentración del contaminante (a través de dos líneas verticales rojas consecutivas). Cada periodo se extiende a lo largo de cinco meses desde el 2 de Noviembre (a las 1 hrs.) hasta el 31 de Marzo (a las 23 hrs.) del año siguiente, lo cual coincide aproximadamente con la medianía de la Primavera y la totalidad del consecuente Verano.

¹⁷<http://ambiente.usach.cl/uv/datos.html>

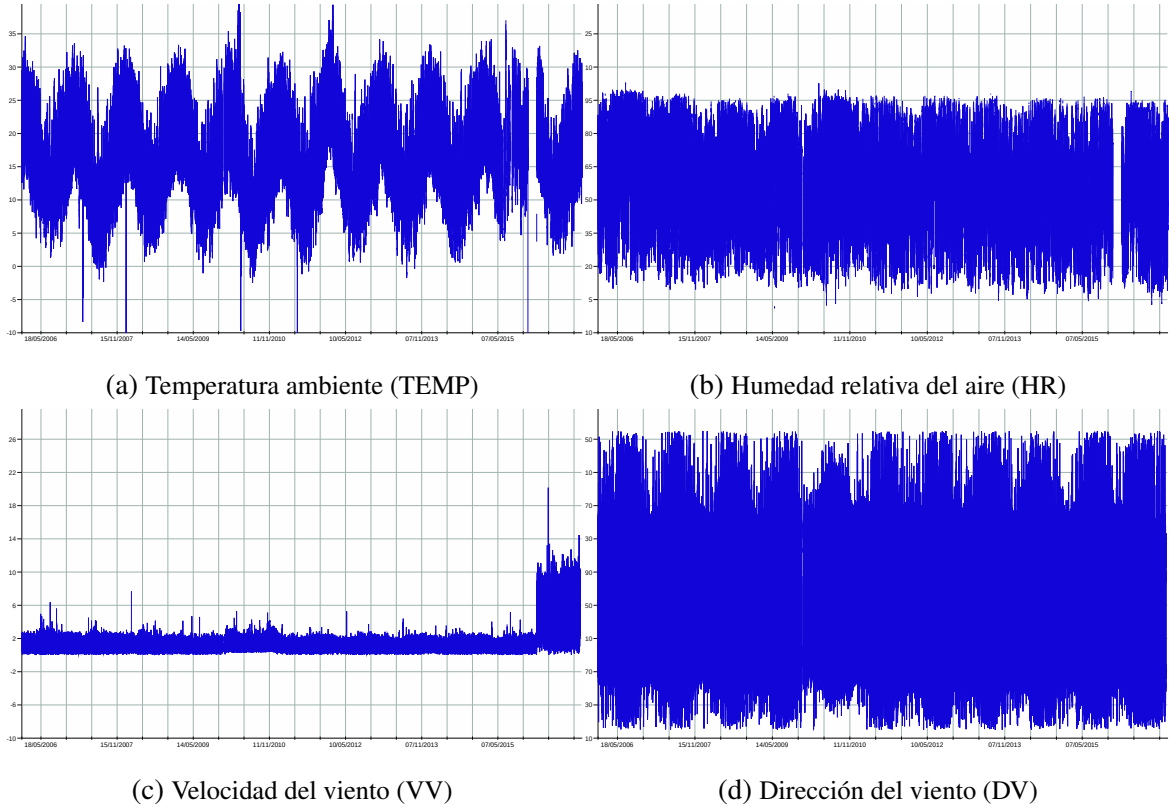


Figura 3.5: Series de tiempo variables meteorológicas.

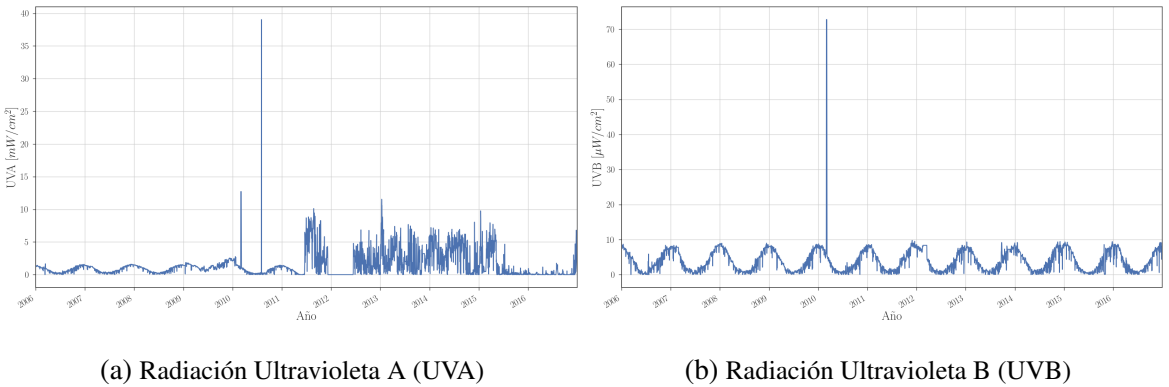


Figura 3.6: Series de tiempo radiación solar ultravioleta.

Lo anterior tiene sentido, tomando en cuenta que durante las estaciones de Primavera y Verano es cuando la radiación solar alcanza las mayores intensidades dentro del año, recordando que tal variable es uno de los factores principales en la formación de Ozono troposférico.

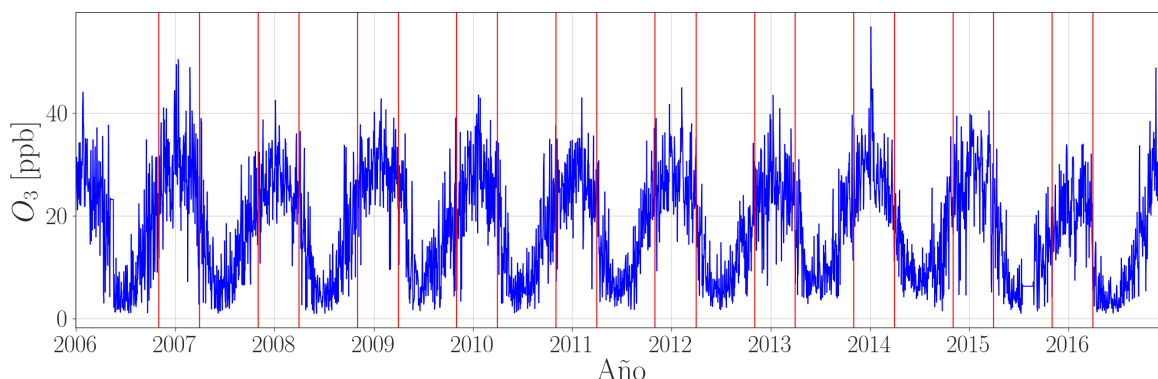


Figura 3.7: Serie de tiempo del Ozono troposférico, con los periodos de alta concentración destacados.

En función de lo anterior, este trabajo se enfoca solo en los datos que pertenecen a los anteriormente descritos periodos de alta concentración de Ozono (referidos de ahora en adelante como “veranos”), con el fin de producir y evaluar modelos predictivos. Se consideran los periodos de entre los años 2007 y 2015, usando como referencia el mes de Marzo de cada año. Se descarta el del 2016 por ser particularmente accidentado, en el sentido de la cantidad de datos ausentes.

Acorde al marco de análisis establecido previamente, en la figura 3.8 se puede observar el comportamiento promedio que tiene la concentración de Ozono troposférico durante el transcurso de un día. Ahí se ve como el nivel de Ozono aumenta durante la mañana alcanzando luego su *peak* a las 13 hrs., coincidiendo aproximadamente con el momento en que el Sol llega a su máxima altitud. Posteriormente, el Ozono disminuye a medida que comienza a atardecer y dar paso a la noche.

En la figura 3.9 se considera un rango de observación de una semana cualquiera, con tal de apreciar como lo anteriormente descrito es un proceso periódico que se da diariamente.

Finalmente, en las figuras 3.10 y 3.11 se grafican las series de tiempo para cada uno de los Veranos considerados, ya sea tomando en cuenta la totalidad de las horas de un día o sólo los peaks diarios respectivamente. Vale la pena notar que en relación a los peaks de concentración, las series de tiempo asociadas son bastante caóticas y aleatorias, presentando ningún patrón aparente.

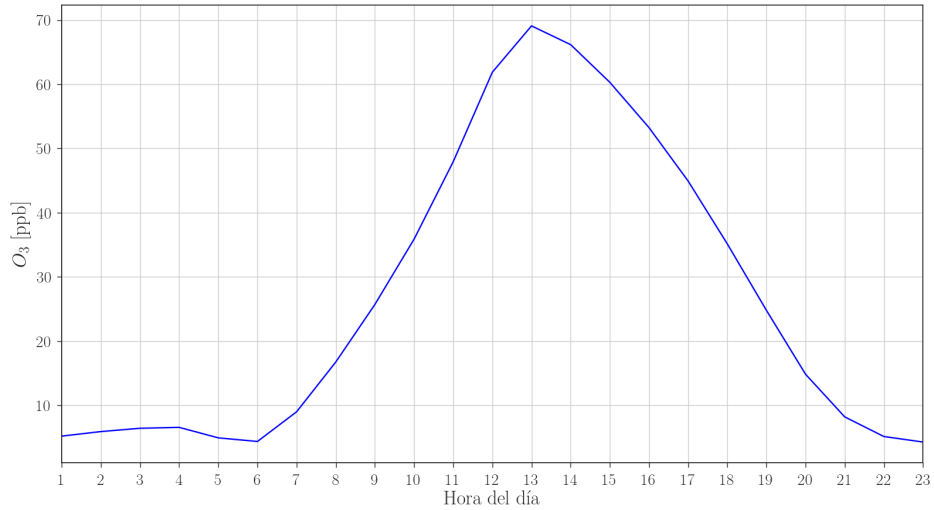


Figura 3.8: Serie de tiempo horaria del Ozono troposférico para los días del año de alta concentración.

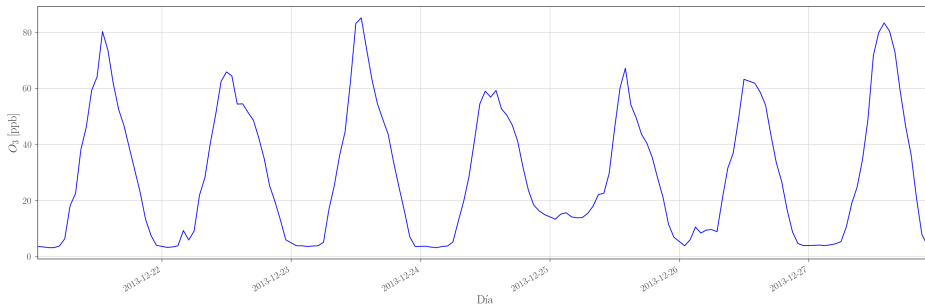


Figura 3.9: Serie de tiempo del Ozono para una semana de ejemplo.

En tanto que en la figura 3.12 se muestra de manera gráfica la correlación lineal entre todas las variables, cada una de ellas en el mismo instante t de tiempo. Ahí se ve que existe una alta correlación entre el Ozono y las variables de carácter meteorológico, principalmente con la Temperatura ambiente (TEMP). Según Robeson y Steyn (1990), lo anterior se explica por dos razones: primero, las altas temperaturas del aire son un excelente indicador de las condiciones ambientales que conducen la producción y acumulación de O_3 (i.e. condiciones anticiclónicas con cielos despejados y vientos ligeros asociados); y segundo, las constantes de velocidad de las reacciones fotoquímicas son altamente dependientes de la temperatura.

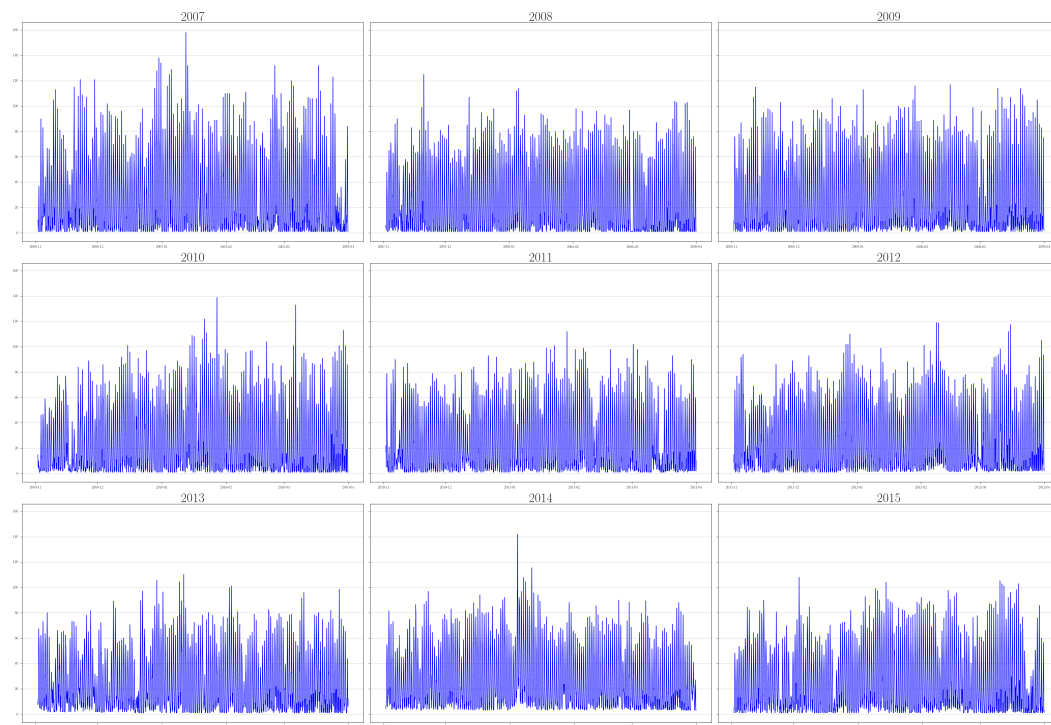


Figura 3.10: Series de tiempo del Ozono para cada Verano.

Respecto de la radiación solar ultravioleta, la correlación con el Ozono es menor, a pesar de que es un precursor de este contaminante. Sin embargo cabe señalar que, como se describió con anterioridad, los registros relativos a la radiación solar son “sucios”, particularmente los de la Radiación Ultravioleta A¹⁸. Otro aspecto que pudiese incidir es que, como se describió previamente, estas variables son monitoreadas desde una locación distinta que la del Ozono.

En el caso de las variables del grupo de los contaminantes, los Óxidos de Nitrógeno (NO_x) y el Monóxido de Carbono (CO), se observa que presentan una correlación negativa con el Ozono. Esto se explica por el hecho de que tales variables se van “consumiendo” a medida que este último se produce. Cabe destacar también la alta correlación existente entre las variables de dicho grupo, haciendo suponer que existe cierta redundancia de información respecto a lo que explican tales variables, en relación a la producción del Ozono.

¹⁸Considerando sólo hasta el año 2011, la correlación de la Radiación Ultravioleta A con el Ozono, aumenta desde 0,33 hasta 0,44.



Figura 3.11: Series de tiempo para los peaks del Ozono en cada Verano.

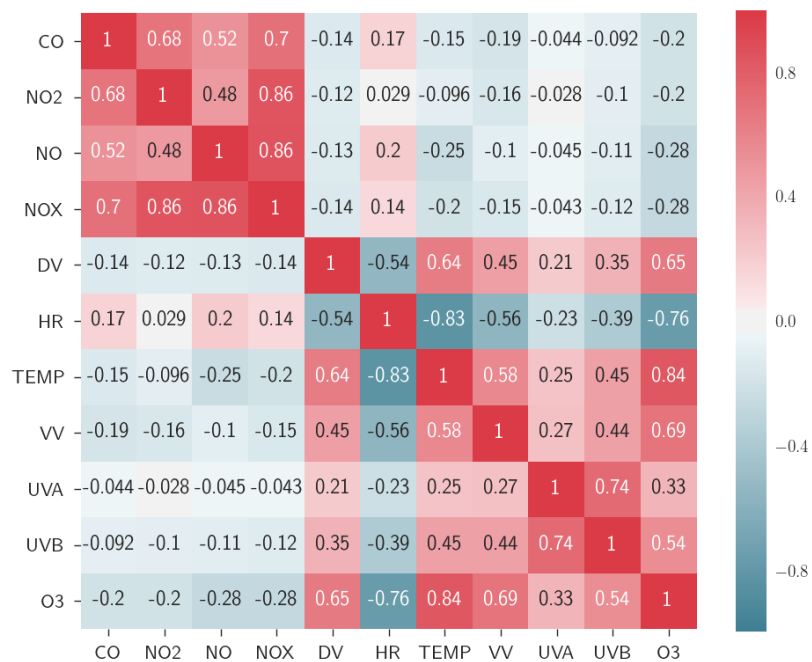
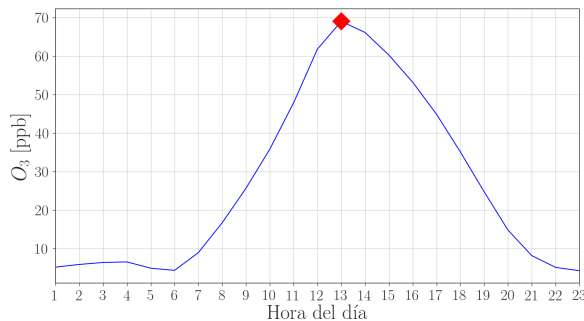


Figura 3.12: Matriz de correlación entre el Ozono y el resto de las variables

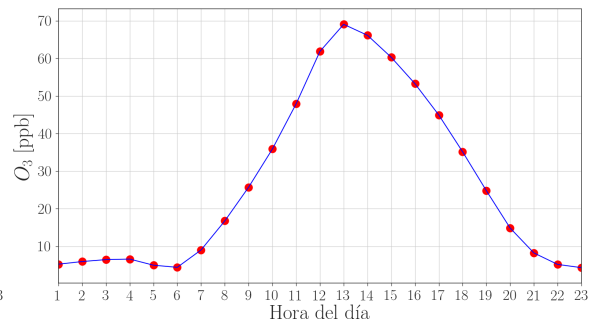
3.2.3. Objetivos del Análisis Predictivo

Los modelos predictivos a desarrollar, tendrán como variable objetivo la concentración de Ozono troposférico considerando dos modalidades:

- Predicción del máximo diario (en concentración de 1 hora) del Ozono troposférico (figura 3.13a).
- Predicción de la concentración horaria de Ozono (figura 3.13b), es decir, una extensión de la modalidad anterior al resto de las horas del día.



(a) Mximo diario de Ozono.



(b) Concentracin horaria de Ozono.

Figura 3.13: Objetivos de prediccin para la concentracin de Ozono troposfrico.

En ambos casos, el horizonte de prediccin ser de un da. Para lo anterior, los modelos podrn hacer uso de los valores pasados del Ozono (O_3) as como tambin del resto de las variables contaminantes y meteorolgicas:

- Variables contaminantes: Monxido de Nitrgeno (NO), Dixido de Nitrgeno (NO_2), xidos de Nitrgeno (NO_x), Monxido de Carbono (CO).
- Variables meteorolgicas: Radiacin Ultravioleta A (UVA), Radiacin Ultravioleta B (UVB), Temperatura ambiente (TEMP), Humedad relativa del aire (HR), Velocidad del viento (VV), Direccin del viento (DV).

Adems, se analizar el efecto de disminuir la cantidad de variables de entrada de los modelos, dejando de lado las de tipo contaminante manteniendo solo las de tipo meteorolgico

junto al Ozono. Esto último tiene como base lo observado en la matriz de correlación de la figura 3.12 del punto 3.2.2, donde las variables de caracter contaminante son las que tienen (en valor absoluto) la correlación más baja con el Ozono.

3.2.4. Preprocesamiento de datos

Luego de descargar los datos desde el SINCA y la estación de la USACH, las series de tiempo correspondientes a cada parámetro de la tabla 3.1 se encuentran separadas, cada una en su propio archivo csv. Por lo tanto, para facilitar el manejo futuro de los datos, se es necesario aglutinar las series de tiempo en un solo archivo unificado. Para ello, se utiliza la librería Pandas y la función `concat`¹⁹ para crear un `dataframe` unificado que está indexado por la fecha y hora de registro, con cada columna representando la serie de tiempo de una variable (ver tabla 3.2). Previamente eso sí y recordando que los datos de la radiación solar UVA y UVB se actualizan cada 3 minutos, las series de tiempo correspondientes a estos dos parámetros se deben agregar en promedios de 1 hora, con tal de tener la misma frecuencia de registro que el resto de los parámetros.

Tabla 3.2: Vista parcial del `dataframe` para el conjunto total de datos.

	CO	NO2	NO	NOX	WD	RH	TEMP	WS	UVA	UVB	O3
registered_on											
2014-12-19 01:00:00	0.1596	18.4642	1.0000	19.4642	72.8683	59.2500	14.1000	0.7985	0.0000	0.0000	2.6087
2014-12-19 02:00:00	0.1797	17.5200	1.0000	18.5200	95.7548	65.0833	13.1083	0.6424	0.0000	0.0000	1.8595
2014-12-19 03:00:00	0.1296	13.2855	1.0000	14.2855	66.7485	65.6666	12.7167	0.8736	0.0000	0.0000	3.0249
2014-12-19 04:00:00	0.1296	11.6833	1.0000	12.6833	91.1517	69.6666	11.8500	0.6756	0.0000	0.0000	2.3590
2014-12-19 05:00:00	0.1296	20.2367	5.5791	25.8158	57.0273	67.3333	12.1000	1.3938	0.0000	0.0000	1.4432
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
2014-12-19 19:00:00	0.1296	16.7475	1.0000	17.7475	210.1580	50.1667	18.8333	1.3128	0.0000	0.5362	14.2637
2014-12-19 20:00:00	0.1296	20.6958	1.0000	21.6958	190.9140	56.5833	16.7833	1.1161	0.0000	0.0000	7.1874
2014-12-19 21:00:00	0.1296	18.9792	1.0000	19.9792	206.6120	61.7500	15.1917	1.0230	0.0000	0.0000	5.1894
2014-12-19 22:00:00	0.1296	19.9233	1.0000	20.9233	194.2150	66.8333	14.0167	1.0777	0.0000	0.0000	2.1092
2014-12-19 23:00:00	0.1296	19.3225	1.0000	20.3225	201.4370	72.0833	12.9667	1.0848	0.0000	0.0000	1.2767
2014-12-20 01:00:00	0.1997	18.2442	5.5116	23.7558	219.7770	80.5833	11.5833	0.8001	0.0000	0.0000	1.0022
2014-12-20 02:00:00	0.1296	16.6378	2.2255	18.8633	227.3680	85.0833	10.5833	0.6166	0.0000	0.0000	1.0000
2014-12-20 03:00:00	0.1519	14.3078	1.9233	16.2311	53.6857	88.3333	9.9916	0.2126	0.0000	0.0000	1.0000

Luego, el `dataframe` es guardado en un archivo binario en formato `hdf5`²⁰, tecnología que ofrece un alto rendimiento en el manejo de grandes colecciones de datos, para luego ser

¹⁹<https://pandas.pydata.org/pandas-docs/stable/merging.html>

²⁰<https://support.hdfgroup.org/HDF5/whatishdf5.html>

importado fácilmente como un `dataframe` en Pandas.

En función de lo expuesto en el punto 3.2.2, se establece la creación de tres conjuntos de datos o *datasets*: uno de entrenamiento, con tal de permitir a los modelos aprender la relación entre el Ozono y las otras variables; uno de validación, usado para la optimización de hiperparámetros; y uno de prueba, con tal de cuantificar el error de generalización. Dichos *datasets* estarán compuestos de la siguiente forma:

- El dataset de entrenamiento estará compuesto por los datos de los siete veranos que van desde el 2007 hasta el 2013.
- El dataset de validación estará compuesto por los datos del verano del 2014.
- El dataset de pruebas estará compuesto por los datos del verano del 2015.

La generación de estos *datasets*, a partir del dataset global de datos, se realizará a través de la aplicación sucesiva de los siguientes pasos de preprocesamiento:

1. Abrir el dataset global de datos mediante un `dataframe` de Pandas.
2. Eliminar las columnas de las variables predictoras no deseadas (opcional).
3. Eliminar las filas del `dataframe` cuyo registro haya sido a las 00:00 hrs. Lo anterior es porque las series de tiempo de los datos del SINCA corresponden al rango horario 1–23 hrs, por lo que los registros correspondientes a la medianoche son nulos. Solo los datos de la Radiación Solar poseen registros de tal instante de tiempo.
4. Manejo de los datos ausentes. Este tipo de datos son nulos o no se encuentran presentes por cualquiera sea la razón. Según la categorización de SINCA, un registro no validado será considerado como dato ausente. La estrategia para rellenar los datos ausentes es propagar el último registro disponible del pasado de la serie de tiempo, pero que además sea de la misma hora del día. Lo anterior aplica para cada columna del `dataframe`.
5. Con tal de tener años uniformes, se eliminan los registros con fecha igual al 29 de Febrero de los años bisiestos.

6. División del dataset en tres partes: entrenamiento, validación y pruebas. El tamaño de estos se muestran en la tabla 3.3.

Tabla 3.3: Tamaño de los datasets de entrenamiento, validación y pruebas.

Dataset	Días	Número de Registros
Entrenamiento	1050	24150
Validación	150	3450
Pruebas	150	3450

7. Como paso opcional, se extrae el registro donde se reporta la máxima concentración de Ozono troposférico para cada día en el dataset. Con esto, se construyen datasets diarios, en el sentido de que se tiene un solo registro por día, dando pie a la modalidad de predicción presentada en la figura 3.13a. En caso contrario, los datasets permanecen siendo horarios, es decir, se tienen 23 registros por cada día dando pie a la modalidad de predicción presentada en la figura 3.13b.
8. Normalización de los datos. Este proceso se realiza mediante la resta de la media y luego dividiendo por la desviación estándar correspondientes a cada una de las variables. Para ello, proveniente del módulo `sklearn.preprocessing`, se utiliza la clase `StandardScaler`²¹, la cual se aplica en primera instancia sobre el conjunto de entrenamiento con tal de obtener un *ajuste*, el cual está compuesto por las medias y desviaciones estándar de cada variable, ambas estadísticas usadas para la normalización. Luego, este *ajuste* es reutilizado para la transformación de los datasets de validación y pruebas. Con lo anterior, las variables tanto de entrada como de salida son normalizadas para tener las propiedades de la distribución normal estándar:

$$\mu = 0$$

$$\sigma = 1$$

Donde μ es la media y σ es la desviación estándar.

²¹<http://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.StandardScaler.html>

9. Finalmente, se generan los arreglos multidimensionales de entrada y salida, para cada uno de los datasets (entrenamiento, validación y pruebas). Las características de dichos arreglos se muestran en la sección 3.3.

3.3. Configuración de los modelos

3.3.1. Aspectos Generales

Con tal de obtener el mejor rendimiento de predicción posible con cada modelo, se realizará un proceso de optimización de hiperparámetros a través de una búsqueda por malla (*grid search*). En esta última, para cada hiperparámetro se selecciona un conjunto finito de valores a explorar. Luego el modelo se entrena utilizando todas las combinaciones posibles de hiperparámetros, seleccionando la que arroja el menor error en el dataset de validación.

Como hiperparámetro común a todos los modelos a evaluar es el número p de *timesteps* o cantidad de valores pasados de las series de tiempo, de las variables utilizadas para predecir la concentración futura de Ozono. El conjunto de posibles *timesteps* es $\{1, 2, 3, 5, 10, 15\}$. Además, cada algoritmo en particular posee sus propios hiperparámetros, los cuales serán detallados más adelante.

En el diagrama de la figura 3.14, se pueden observar el conjunto de pasos que conforman el flujo de ejecución de pruebas que se aplicará para cada configuración de hiperparámetros, dado un algoritmo en específico.

A grandes rasgos, cada paso se describe como sigue a continuación:

1. El primer paso, es la preparación de datos, el cual implica la construcción de los arreglos multidimensionales de entrada y salida para los datasets de entrenamiento, validación y pruebas. Este paso es dependiente de la cantidad de *timesteps*. En el caso de los modelos basados en series de tiempo (ARIMA) este paso es omitido, ya que su implementación es completamente distinta.

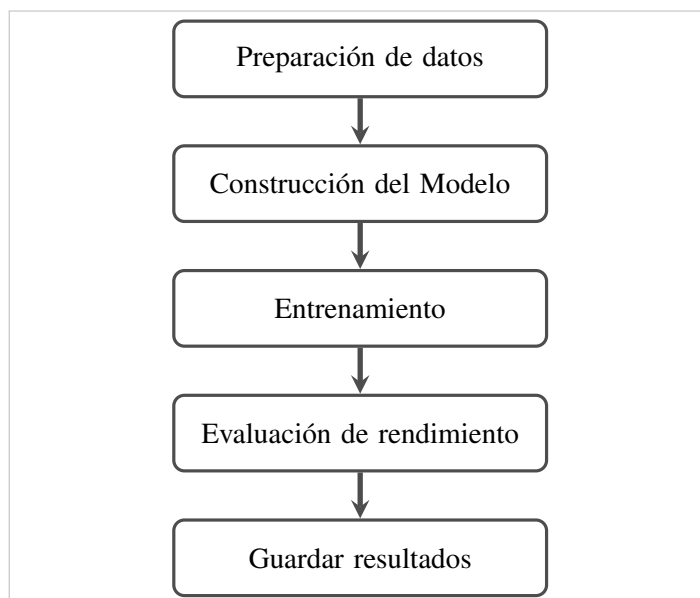


Figura 3.14: Flujo de ejecución de pruebas para un modelo.

2. El siguiente paso es el de construcción del modelo, el cual involucra instanciar la clase mediante la cual se implementa el algoritmo en cuestión, considerando los hiperparámetros que le son propios a su tipo.
3. El tercer paso es el de entrenamiento, mediante el cual el algoritmo aprende del dataset de entrenamiento con tal de ajustar su configuración interna.
4. El paso de evaluación de rendimiento implica en realizar las predicciones para los tres datasets (entrenamiento, validación y pruebas) y cuantificar los errores resultantes. Para esto último, se utilizará la métrica del *Root Mean Square Error (RMSE)*, presentada en la sección 1.4. Como detalle, dado que las pruebas se realizan con los datos normalizados (tanto de entrada como de salida), la medición del rendimiento de los modelos se realiza con los valores de salida del Ozono en su escala original (en ppb). Para invertir la normalización se utiliza la función `inverse.transform` de la clase `StandardScaler`.
5. Por último, el RMSE obtenido en cada dataset es guardado en un archivo csv de salida, junto con el detalle de la configuración de hiperparámetros recién evaluada.

Finalmente, luego de evaluar todas las combinaciones de hiperparámetros, se obtiene la configuración con menor RMSE de validación junto con el RMSE de pruebas asociado, ya que a partir de este último se establece el error de generalización correspondiente.

3.3.2. Datos de Entrada y Salida

Las estructura de los datos de entrada y salida difieren en función de la forma de procesamiento del algoritmo en cuestión.

En el caso de los modelos de Aprendizaje de Máquina más clásicos, i.e. Regresión Lineal, SVM y Redes Feed Forward, tal estructura se fundamenta según lo expresado en la sección 1.3. En términos prácticos, para los datos de entrada se tiene un arreglo 2-dimensional, tal como se muestra en la figura 3.15. Usando la notación en forma de tupla de Numpy para el *shape* dimensional, un arreglo como tal tiene un *shape* igual a (m, n) , donde m es el número de ejemplos del dataset y n el de variables predictoras. En el caso de la salida, es simplemente un arreglo unidimensional de *shape* $(m,)$. En esta situación, un $x^{(i)}$ perteneciente a un tiempo t tiene como respuesta a $y^{(i)}$ en el tiempo $t + 1$.

$$\left[\begin{array}{cccc} \left[\begin{array}{cccc} 0,1390 & 2,2447 & \dots & -0,9799 \end{array} \right] & \rightarrow & x^{(1)} \\ \left[\begin{array}{cccc} -0,1984 & -0,4050 & \dots & -1,0142 \end{array} \right] & \rightarrow & x^{(2)} \\ & & \vdots & \vdots \\ \left[\begin{array}{cccc} 0,1390 & 2,2447 & \dots & -0,9799 \end{array} \right] & \rightarrow & x^{(m)} \end{array} \right]$$

Figura 3.15: Arreglo 2-dimensional de entrada para un algoritmo de Aprendizaje Automático.

Sin embargo, la estructura anterior no permite procesar datos secuenciales, pudiéndose usar solo un *timestep*. Para incorporar $p > 1$ *timesteps* es necesario agregarlos explícitamente, concatenados en cada $x^{(i)}$. Es decir, el *shape* resultante del arreglo de la figura 3.15 es

$(m, n \times p)$. No obstante, esto tiene como consecuencia el aumento de la complejidad del problema a resolver por el modelo, dado que ahora hay una mayor cantidad de dimensiones que considerar.

Para el caso de las Redes Neuronales Recurrentes, estas cuentan con la capacidad de procesar datos de tipo secuencial, pudiendo utilizar más de un *timestep* sin ninguna dificultad. En la figura 3.16 se muestra el arreglo 3-dimensional de entrada con *shape* (m, p, n) , donde el *batch* es un conjunto compuesto por m secuencias, cada una de las cuales compuesta por p *timesteps* de dimensión n . En esta situación cada secuencia puede tener asociada, como variable objetivo, un escalar u otra secuencia de salida. En el primer caso, el *shape* del arreglo de salida es $(m, 1)$, mientras que en el segundo caso es $(m, n_y, 1)$ donde n_y es largo de la secuencia de salida.

$$\text{Batch} \left\{ \begin{array}{l} \text{Secuencia} \left\{ \begin{array}{l} \text{timestep} \left\{ \begin{bmatrix} 0,4366 & \cdots & -1,0143 \end{bmatrix} \\ \begin{bmatrix} -0,1984 & \cdots & -1,0142 \end{bmatrix} \end{array} \right\} \\ \begin{bmatrix} 1,0715 & \cdots & -0,9387 \end{bmatrix} \\ \begin{bmatrix} -0,1984 & \cdots & -0,7497 \end{bmatrix} \\ \vdots \\ \begin{bmatrix} -0,4583 & \cdots & -0,5173 \end{bmatrix} \\ \begin{bmatrix} -0,1293 & \cdots & 0,3737 \end{bmatrix} \end{array} \right\} \end{array} \right\}$$

Figura 3.16: Arreglo de entrada para una Red Neuronal Recurrente. En este ejemplo, cada secuencia está compuesta de dos *timesteps*.

Por último, para el caso del modelo basado en ARIMA, el tratamiento de los datos es totalmente distinto y se verá en el punto 3.3.8.

3.3.3. Modelo Persistente

Este es el modelo más simple ya que no requiere ningún tipo de configuración especial en particular. Para la concentración del Ozono (O_3) en el tiempo $t + 1$ utiliza como predicción los valores observados del pasado en el tiempo t :

$$O_3^{(t+1)} = O_3^{(t)} \quad (3.1)$$

Este modelo se implementará por motivos de comparación y solo para la modalidad de máximos diarios.

3.3.4. Regresión Lineal

Para implementar el modelo de regresión lineal, se utiliza la clase `LinearRegression`²² del módulo `sklearn.linear_model` de Scikit Learn. No se requieren configuraciones adicionales de parámetros en particular.

3.3.5. Support Vector Machine

El modelo basado en SVM se crea utilizando la clase `SVR`²³ del módulo `sklearn.svm` de Scikit Learn, la cual es una implementación de la ϵ -SVR vista en la Sección 1.3. Los parámetros optimizados, con los conjuntos de valores posibles, se muestran a continuación en la tabla 3.4:

Tabla 3.4: Parámetros optimizados para la SVR.

Parámetro	Conjunto de valores posibles
C	$\{10^{-3}, 10^{-2}, 10^{-1}, 10^0, 10^1, 10^2\}$
γ	$\{10^{-2}, 10^{-1}, 10^0, 10^1, 10^2\}$

²²http://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LinearRegression.html

²³<http://scikit-learn.org/stable/modules/generated/sklearn.svm.SVR.html>

3.3.6. Red Neuronal Feed Forward

El modelo de Red Neuronal Feed Forward se implementa a través de la librería Keras, utilizando la clase `Sequential`²⁴, la cual permite apilar un conjunto *layers* (capas) de forma secuencial. En la figura 3.17 se puede observar un diagrama del modelo de la red, compuesto por una arquitectura tradicional de tres capas totalmente conectadas: una de entrada, una oculta y la de salida. Para la capa oculta, implementada mediante la clase `Dense`²⁵ del módulo `keras.layers`, la cantidad de nodos (*hn*) es optimizada dentro del conjunto de valores {5, 10, 15, 20, 50, 100}. La función de activación de estos últimos es la función *Rectified Linear Unit (ReLU)*. En tanto que la capa de salida, compuesta por un único nodo, también se implementa mediante la capa `Dense` con función de activación lineal.

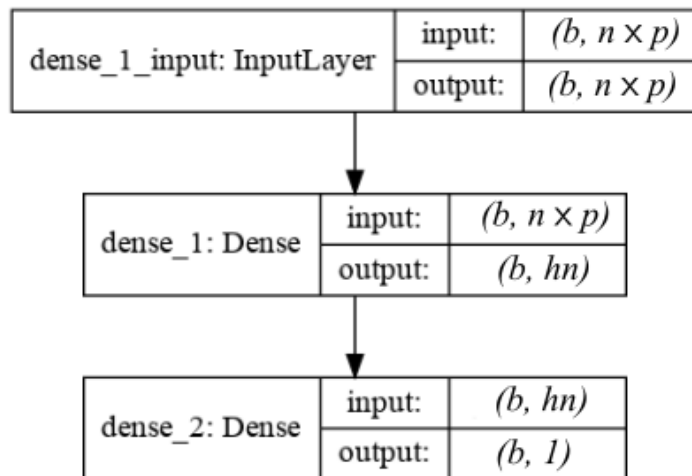


Figura 3.17: Modelo de la Red Neuronal Feed Forward. En el diagrama, b es el tamaño del *batch* de entrenamiento, n es el número de variables de entrada, p es el número de *timesteps* y hn la cantidad de nodos ocultos.

²⁴<https://keras.io/getting-started/sequential-model-guide/>

²⁵<https://keras.io/layers/core/#dense>

Entrenamiento

El relación al proceso de entrenamiento de la red, se tienen las siguientes características:

- Tamaño del *batch* (b) de entrenamiento: 50, valor que debe ser divisor exacto del tamaño de los datasets (tabla 3.3).
- Algoritmo de optimización del gradiente descendente: *Adam* (Kingma y Ba, 2014).
- Función de costo: *Mean Square Error* (*MSE*).
- El número máximo de *epochs*²⁶ es de 500.
- Dado que los pesos de la red se inicializan con valores aleatorios, considerando la misma configuración de hiperparámetros, el proceso de entrenamiento y evaluación se realizará al menos 10 veces. El rendimiento final será considerado como la media de esas 10 ejecuciones.

Regularización

Como método de regularización se utiliza *Early Stopping*, mediante la clase `EarlyStopping`²⁷ del módulo `keras.callbacks`. Como parámetros relevantes se tiene:

- `patience`: Cantidad de *epochs* sin mejoras en la función de costo después de la cual el entrenamiento se detendrá. El valor establecido es igual a 10.
- `min_delta`: Mínimo cambio en la función de costo que se considerará como una mejora. El valor establecido es igual 0.001.

²⁶Número de iteraciones de entrenamiento, es decir, el número de veces en que todos los ejemplos de entrenamiento son propagados hacia adelante y luego hacia atrás a través de la red, como parte del proceso de *Backpropagation*.

²⁷<https://keras.io/callbacks/#earlystopping>

3.3.7. Redes Neuronales Recurrentes

Se crean dos modelos para este tipo de redes. Ambos, fundamentados en la utilización de células LSTM, se diferencian entre sí por el modo de procesamiento de las secuencias de entrada y las de salida. Por un lado, se tiene un modelo de red *many to one* (figura 1.5c), donde se tiene como entrada una secuencia de largo variable y como salida un escalar o valor real. En el segundo caso, se tiene un modelo de red *many to many* (figura 1.5d), el cual procesa una secuencia de entrada de largo variable produciendo como salida otra secuencia de largo fijo o variable.

En torno a las características del proceso de entrenamiento y regularización, estas serán iguales a las presentes para las Redes Neuronales Feed Forward.

En relación a la implementación de ambos modelos, esta se hace usando la clase `Sequential` (al igual que la red Feed Forward). Las células LSTM, se implementan en base a la capa LSTM²⁸ del módulo `keras.layers`. Algunos aspectos importantes son:

- Función de activación: Tangente Hiperbólica (*tanh*).
- Parámetro `stateful` en `False`. Esto implica que el estado de la célula se reinicia al finalizar el procesamiento de un *batch* en la etapa de entrenamiento.

Y para la capa de salida, se utiliza una función de activación lineal.

Modelo *many to one*

En la figura 3.18 se puede observar la composición de la arquitectura de este modelo. Al igual que en una Red Feed Forward, se tienen tres capas, pero con la oculta siendo una capa LSTM. La cantidad de nodos en esta última es optimizada dentro del conjunto de valores {5, 10, 15, 20, 50, 100}. El nodo de salida tiene una función de activación lineal.

²⁸<https://keras.io/layers/recurrent/#lstm>

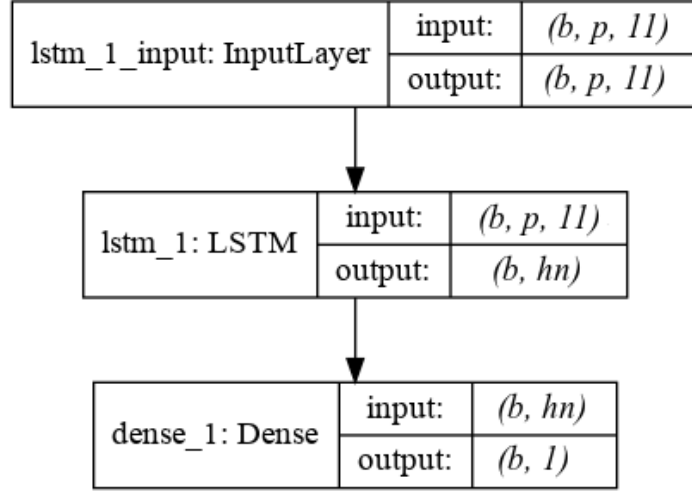


Figura 3.18: Modelo de Red Neuronal Recurrente con células LSTM y procesamiento *many to one*. En el diagrama, b es el tamaño del *batch* de entrenamiento, p es el número de *time-steps* y hn es la cantidad de nodos ocultos.

Modelo *many to many*

A través de una secuencia de entrada compuesta por las $p \times 23$ horas previas, este modelo predice una secuencia de salida de las siguientes 23 horas de la concentración de Ozono²⁹. En términos visuales, las secuencias de entrada y de salida se pueden ejemplificar mediante las series de tiempo vistas en el punto 3.2.2, en que la figura 3.8 representa una secuencia de salida, mientras que la serie de tiempo semanal de la figura 3.9 representa una secuencia de entrada de largo $p \times 23$ horas con $p = 7$.

La estructura de este modelo se basa en el trabajo presentado por (Cho y cols., 2014), donde se propone una arquitectura que aprende a codificar una secuencia de largo variable en una representación vectorial de largo fijo, para luego decodificar esta última en una secuencia de largo variable. En la figura 3.19 se muestra el cómo se estructura dicha arquitectura, la que en términos simples, procesa una secuencia de entrada $(x^{(1)}, x^{(2)}, \dots, x^{(n_x)})$ de largo n_x , codificándola en una especie de *resumen* C y así, a partir de este último, generar la secuencia de salida $(y^{(1)}, y^{(2)}, \dots, y^{(n_y)})$ de largo n_y . Ambas secuencias, entrada y salida, pueden no ser del

²⁹Esta definición, describe un escenario en que se predice la concentración horaria del Ozono, según el punto 3.2.3. Sin embargo, con este mismo modelo, se puede implementar una modalidad *many to one*, donde se tiene una secuencia de entrada de largo p para predecir solo el máximo diario de Ozono.

mismo largo. Esta arquitectura fue inicialmente concebida para la traducción de frases entre idiomas. Otros trabajos, como en Sutskever, Vinyals, y Le (2014), proponen una arquitectura similar para el mismo fin.

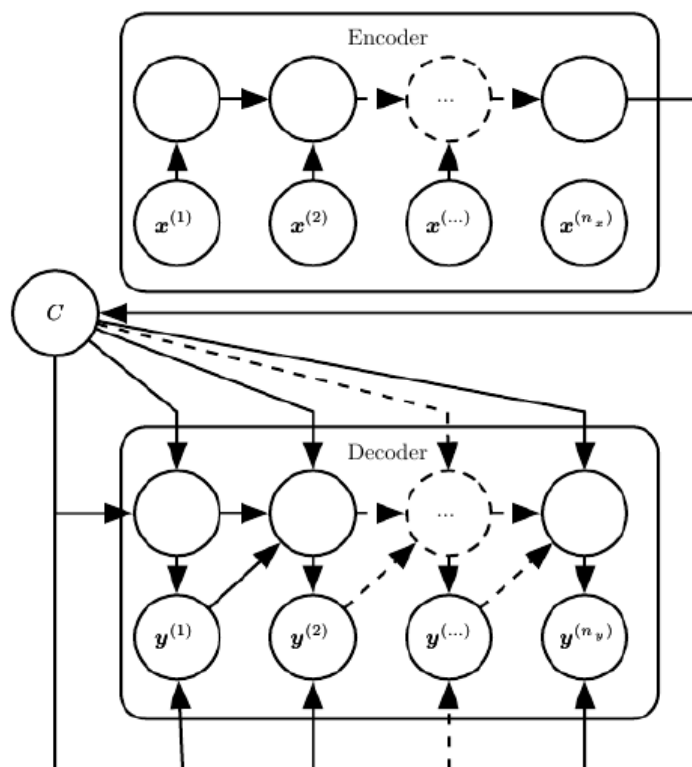


Figura 3.19: Arquitectura *encoder-decoder*. Fuente: Goodfellow y cols. (2016).

La implementación de la arquitectura descrita en la figura 3.19, se realiza mediante el modelo de la figura 3.20. Ahí, el *encoder* y *decoder*, se implementan como la capa LSTM (`lstm_1` y `lstm_2` respectivamente). La codificación de salida del *encoder* genera un vector temporal a partir de la secuencia de entrada. Luego, tal codificación es repetida $n_y = 23$ veces a través de la capa `RepeatVector`³⁰ para posteriormente ser decodificada por el *decoder*.

Ambos, el *encoder* y el *decoder*, comparten la misma cantidad de células LSTM, cantidad que es optimizada dentro del conjunto de valores $\{5, 10, 15, 20, 50, 100\}$.

³⁰<https://keras.io/layers/core/#repeatvector>

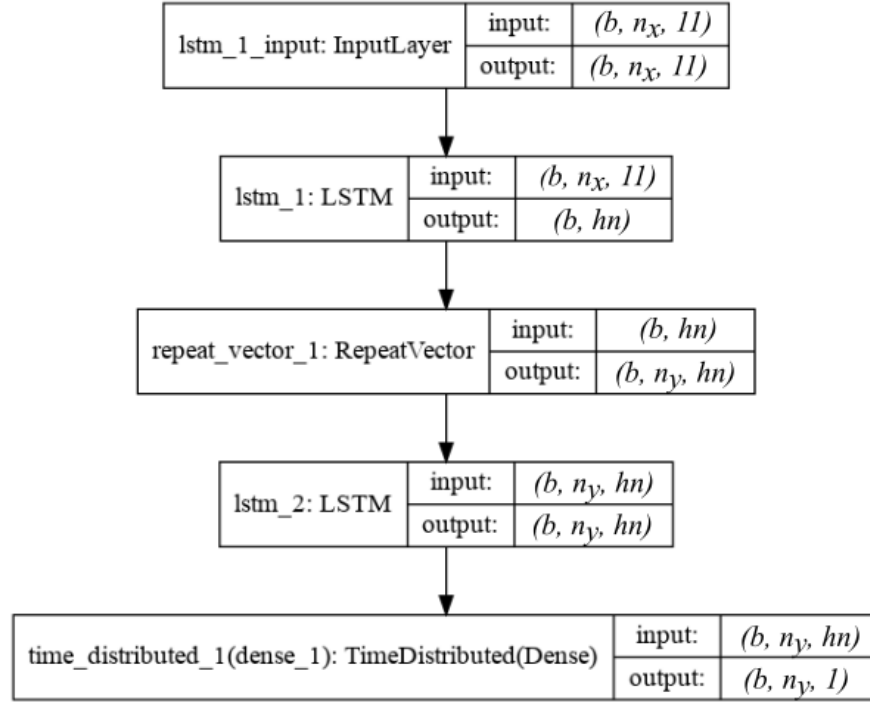


Figura 3.20: Modelo de Red Neuronal Recurrente con células LSTM y procesamiento *many to many*. En el diagrama, b es el tamaño del *batch* de entrenamiento, n_x es el largo de la secuencia de entrada, hn es la cantidad de nodos ocultos y n_y es el largo de la secuencia de salida.

3.3.8. ARIMA

A diferencia de los modelos basados en Aprendizaje Automático, ARIMA será implementado a través de R, utilizando el paquete `forecast`. Además, será aplicado solo para la modalidad de máximos diarios, por: su sola relevancia para los máximos diarios -de acuerdo al Estado del Arte en el Capítulo 2- y por la imposibilidad de programar la modalidad de concentración horaria, en que se considere como *timesteps* los registros que pertenecen a los días anteriores y no a las horas previas dentro del mismo día al cual pertenece el valor objetivo a predecir.

En relación a la composición de los datasets de entrenamiento, validación y pruebas, estos difieren en que solo consideran la serie de tiempo del Ozono, prescindiendo del resto de las variables contaminantes y meteorológicas. Además, tales datasets se almacenan en formato

csv.

En el caso de ARIMA, el objetivo es encontrar la mejor combinación de los parámetros p, q, d vistos en la sección 1.2. El conjunto de valores posibles a explorar para cada parámetro, se muestran en la tabla 3.5

Tabla 3.5: Parámetros optimizados para el modelo ARIMA.

Parámetro	Conjunto de valores posibles
p	{1, 2, 3, 5, 10, 15}
d	{1, 2, 3}
q	{1, 3, 3}

En el caso de los valores posibles para el parámetro p , estos coinciden con lo detallado en el punto 3.3.1 para los *timesteps*.

Capítulo 4

Resultados Experimentales

En este capítulo, se presentarán los resultados derivados de los experimentos para predecir la concentración de Ozono troposférico, a partir de los algoritmos basados en Aprendizaje Automático y series de tiempo. Tales resultados permitirán establecer una visión general acerca del desempeño de los algoritmos, comparándolos entre sí en torno a los distintos objetivos de predicción presentados en el capítulo anterior.

El presente capítulo se organiza de la siguiente forma: en la Sección 4.1 se presentan los resultados relativos a la predicción de los máximos diarios de Ozono, comparando el rendimiento de los distintos algoritmos, analizando además aspectos generales y específicos de estos últimos; se extiende este análisis de manera análoga en la Sección 4.2 para la concentración horaria de Ozono; y finalmente, en la Sección 4.3, se analiza el efecto de reducir la cantidad de variables predictoras en el rendimiento de los algoritmos basados en Aprendizaje Automático, basado en el análisis descriptivo de los datos realizado en el capítulo anterior.

4.1. Máximos diarios

4.1.1. Resumen general

En la tabla 4.1 se presenta el RMSE alcanzado (en el dataset de pruebas) por cada algoritmo en la predicción de los máximos diarios de Ozono troposférico, junto con la configuración de hiperparámetros correspondiente, obtenida como resultado del proceso de *grid search*. Ahí se puede observar que el mejor desempeño lo alcanzó la Red Feed Forward (FFN) con un RMSE de 15,240. Sin embargo, este resultado no se destaca en gran medida respecto del obtenido por el resto de los algoritmos, donde por ejemplo la SVR, que es la que obtiene el peor rendimiento, exhibe un RMSE mayor en apenas un 2,6 % respecto del de la red Feed Forward. Modelos más simples como ARIMA, que solo utilizan la serie de tiempo del Ozono, alcanza un desempeño similar al de la Red Feed Forward.

Tabla 4.1: Resumen máximos diarios

	LR (p^a)	SVR ($p; C; \gamma$)	FFN ($p; hn^b$)	LSTM m2o ^c ($p; hn$)	LSTM m2m ^d ($p; hn$)	ARIMA ($p; q; d$)
Hiperparámetros	(10)	(5; 1,0; 0,01)	(1; 100)	(1; 100)	(1; 100)	(15; 2; 3)
RMSE	15,582	15,632	15,240	15,329	15,310	15,250

^a # *timesteps*

^b Nodos ocultos

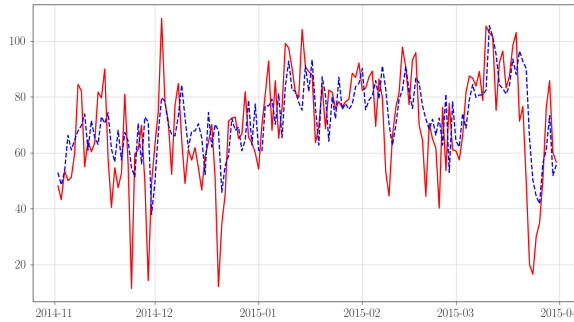
^c *many to one*

^d *many to many*

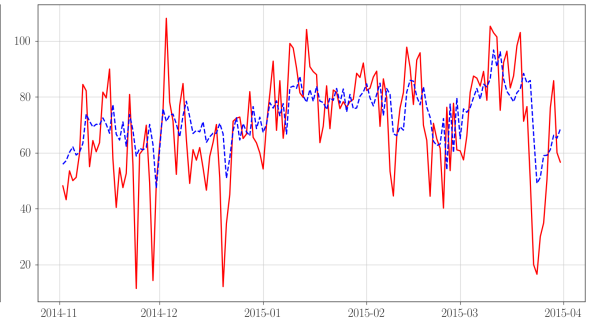
En tanto que con el modelo persistente se obtiene un RMSE igual a 16,432, por lo que la utilización de todos los anteriores modelos supone una mejora de los resultados. Cuantitativamente hablando, el modelo basado en una red neuronal Feed Forward genera una disminución de 7,254 % respecto del persistente.

Visualmente hablando, en la figura 4.1 se muestran las gráficas de las predicciones hechas por cada algoritmo, junto con la serie de tiempo del valor real observado del Ozono troposférico, que cómo se mencionó en el capítulo anterior, es parte del dataset de pruebas. Ahí se observa que entre los distintos algoritmos, las predicciones son bastante similares, tomando en cuenta

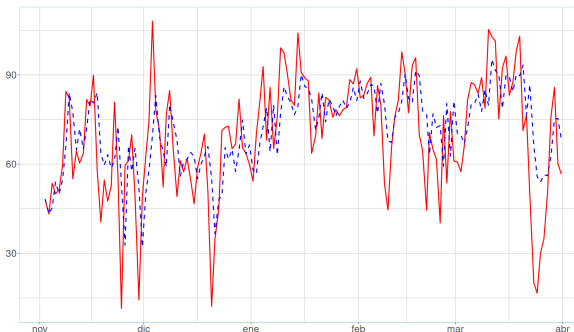
que tienden a subestimar el valor del Ozono en momentos que se dan máximos locales dentro de la serie de tiempo. Análogamente, cuando se presentan mínimos locales dentro de la serie de tiempo, los algoritmos tienden a sobreestimar la concentración de Ozono.



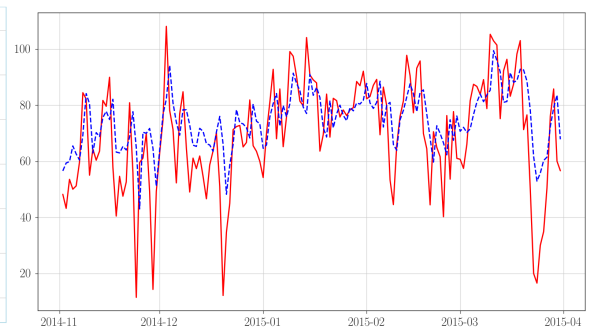
(a) Regresión Lineal.



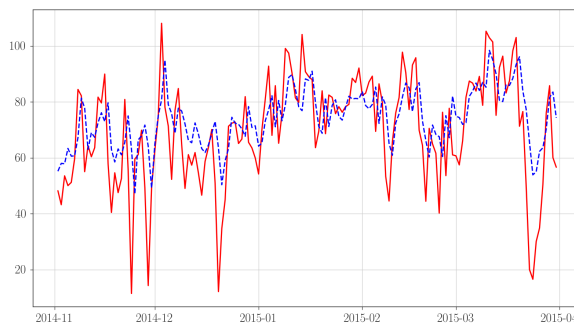
(b) Support Vector Regression.



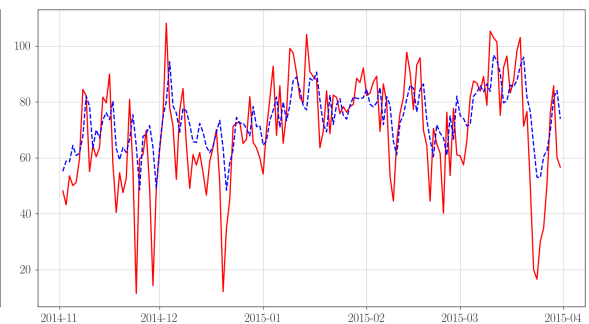
(c) ARIMA.



(d) Feed Forward Neural Network.



(e) LSTM *many to one*.



(f) LSTM *many to many*.

Figura 4.1: Predicciones para el dataset de prueba. En color rojo se grafican los valores reales u observados para la serie de tiempo del Ozono, mientras que con un estilo punteado en azul, se grafican las predicciones realizadas por el modelo respectivo.

4.1.2. Cantidad de *timesteps*

Considerando que en el capítulo anterior, se estableció que la cantidad de *timesteps* sería un hiperparámetro común a optimizar para todos los algoritmos, se analiza la influencia que tiene dicho hiperparámetro sobre el RMSE conseguido en el dataset de validación. Para ello, los resultados son agregados y promediados en torno a cada valor posible para la cantidad de *timesteps*. Lo anterior da como resultado la gráfica de la figura 4.2 donde se muestra el RMSE en función del número de *timesteps*. Ahí se aprecia que para los algoritmos basados en Aprendizaje Automático, el utilizar una mayor cantidad de valores pasados de la serie de tiempo, no redundaba en una mejora del rendimiento de tales algoritmos, generando incluso un empeoramiento significativo en la Red Neuronal Feed Forward. En relación a los algoritmos que sí tienen la capacidad de manejar datos secuenciales i.e. Redes Neuronales Recurrentes, se observa que generan un rendimiento relativamente estable, aunque con un empeoramiento paulatino a medida que se aumenta el número de *timesteps*. Lo anterior hace suponer que para la predicción del Ozono troposférico, no se es necesario considerar un rango de tiempo demasiado extenso hacia el pasado de la serie. Sin embargo, en la vereda contraria, el modelo para series de tiempo ARIMA muestra una mejora consistente del rendimiento al aumentar la cantidad *timesteps* utilizados.

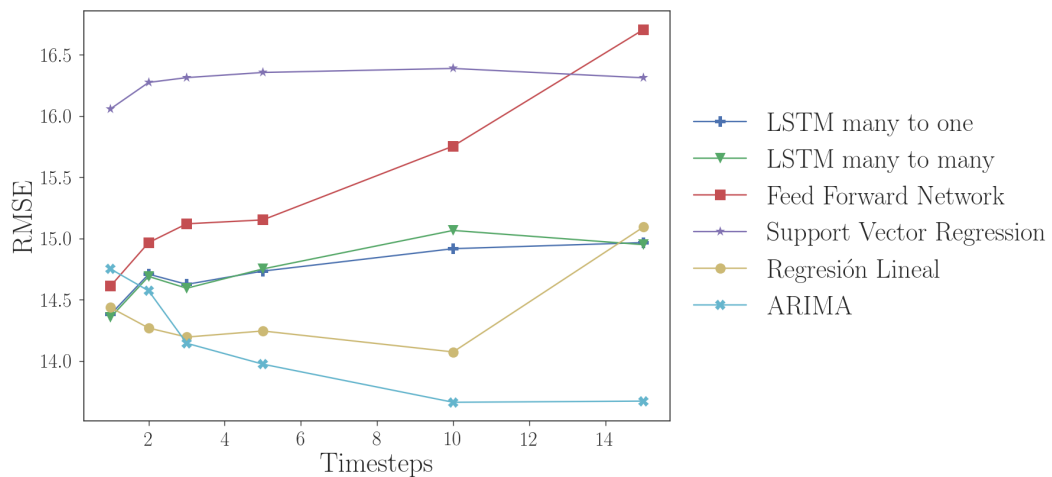


Figura 4.2: RMSE en función de la cantidad (*timesteps*) utilizados.

4.1.3. Redes Neuronales

Dado que se implementaron tres modelos basados en Redes Neuronales, se analizan ciertos aspectos comunes que son influenciados por el número de neuronas que componen las capas ocultas de las redes, el cual era un hiperparámetro a optimizar.

En la figura 4.3 se grafica el RMSE de validación en función del número de nodos ocultos. Se observa que en promedio, un aumento en la cantidad de dicho hiperparámetro no origina una mejora de rendimiento. Para los tres modelos de redes neuronales, el mejor rendimiento se alcanza con un número bajo de neuronas ocultas, menor a 20 específicamente. Esto muestra que una red más compleja, en términos del tamaño de su o sus capas ocultas, no es mejor que una red más simple. Ahora bien, de la misma gráfica se desprende también que las Redes Recurrentes LSTM tienen un rendimiento superior a la Red Feed Forward.

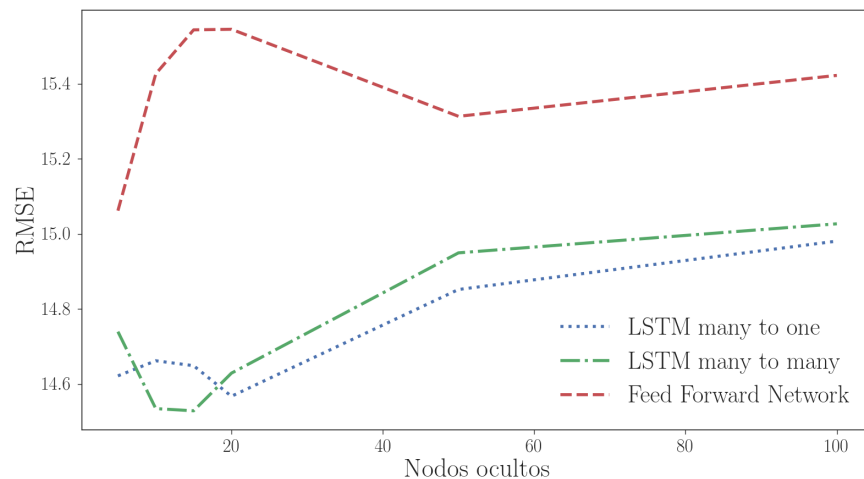
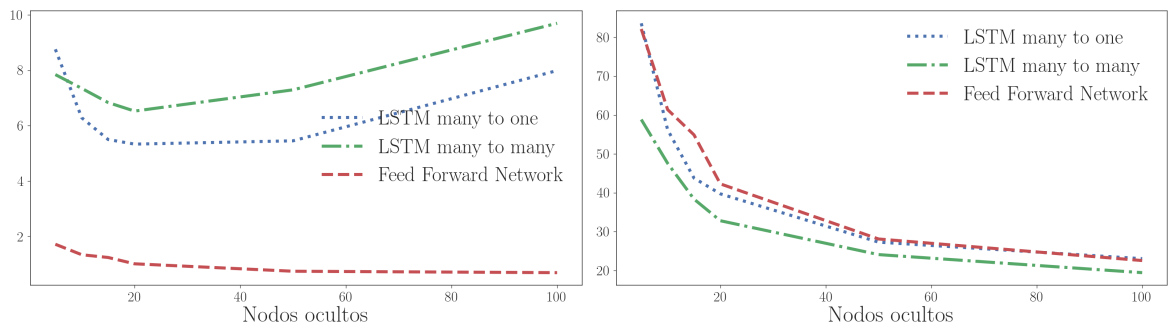


Figura 4.3: RMSE en función de la cantidad de nodos ocultos.

En tanto que en las gráficas de la figura 4.4, se muestran el tiempo (en segundos) y el número de *epochs* requeridos para entrenar las redes según la cantidad de neuronas ocultas. Primero, en relación al tiempo de entrenamiento (figura 4.4a), las redes recurrentes LSTM requieren más de este último si se compara con la red Feed Forward, hecho que tiene sentido si se considera que las redes recurrentes son modelos que requieren ajustar más parámetros que una red neuronal tradicional, dada la existencia de ciclos o conexiones recurrentes en las primeras. Se observa además que la que más tarda en aprender es el modelo recurrente *many*

to many, ya que tal como se presentó en el capítulo pasado, es un modelo más complejo por el hecho de poseer dos capas ocultas a diferencia de los otros que solo tienen una. Segundo, en la figura 4.4b se muestra que los tres modelos, de manera similar, requieren de una menor cantidad de *epochs* a medida que el número de nodos ocultos aumenta. Tal número de *epochs* es determinado por el proceso de regularización de *Early Stopping*, implementado para evitar que los modelos se vuelvan demasiado complejos perjudicando luego su capacidad de generalización. Pues bien, los resultados avalan tal afirmación, al detener el entrenamiento de los modelos más complejos (de más nodos ocultos) con mayor antelación.



(a) Tiempo de entrenamiento.

(b) Número de *epochs*.

Figura 4.4: Tiempo de entrenamiento (4.4a) y número de *epochs* (4.4b) en función del número de nodos ocultos.

4.1.4. Comparación con otros estudios

De cierta forma es complicado generar una comparación de los presentes resultados con los provistos por otros estudios, dado que principalmente en estos últimos:

- Se utilizan conjuntos de datos distintos
- Cuyos registros pueden pertenecer a un distinto periodo de tiempo
- Con variables de predicción distintas, las cuales responden a las características particulares del lugar donde se lleva a cabo el estudio

- Y con la implementación de técnicas o modelos distintos, mediante configuraciones diferentes

Sin embargo, el estudio realizado por Jorquera y cols. (1998) presenta ciertos aspectos que permiten realizar una comparación. A grandes rasgos, en aquel estudio se construyeron modelos predictivos basados en redes neuronales artificiales, modelado difuso y técnicas para series de tiempo. El objetivo era predecir el máximo diario de la concentración de Ozono, utilizando los datos registrados por las estaciones de monitoreo localizadas en las comunas de Las Condes e Independencia, entre los años 1990 y 1994. Las variables predictoras utilizadas son el Ozono y la temperatura del aire que actúa como variable exógena.

Considerando la estación de Las Condes (la cual compete al presente trabajo), se utilizaron los datos que se dividieron en múltiples subconjuntos y que se muestran en la tabla 4.2. Los registros correspondientes a los subconjuntos E2, E3 y E5 fueron concatenados en un conjunto más grande con tal de ser utilizados para el ajuste de los modelos implementados, mientras que los subconjuntos E6 y E7 fueron utilizados para evaluar y determinar el rendimiento de tales modelos.

Tabla 4.2: Subconjuntos de datos correspondientes a los registros monitoreados por la estación Las Condes, utilizados en Jorquera y cols. (1998).

Subconjunto de datos	Periodo	Días
E1	1992/02/17–1992/03/31	43
E2	1992/10/30–1992/12/18	50
E3	1992/12/21–1993/03/05	75
E4	1993/03/07–1993/05/15	70
E5	1994/01/28–1994/03/16	48
E6	1994/03/22–1994/11/11	235
E7	1994/11/16–1995/02/16	93

Tomando en consideración el subconjunto E7, el cual pertenece a un rango de tiempo similar al de los veranos utilizados en el presente trabajo, en Jorquera y cols. (1998) se alcanzaron los rendimientos que se muestran en la tabla 4.3. Ahí se observa que el mejor modelo es

el basado en técnicas de series de tiempo, mientras que el basado en redes neuronales y en modelado difuso son apenas mejores que un modelo persistente.

Tabla 4.3: RMSE obtenido por los modelos implementados por Jorquera y cols. (1998) evaluando sobre el subconjunto de datos E7.

	Modelo Persistente	Series de Tiempo	Red Neuronal	Modelo Difuso
RMSE	34,3	29,81	34,0	33,3

Comparando con los resultados que se muestran en la tabla 4.1 (los cuales son mejores), la diferencia con los obtenidos por Jorquera y cols. (1998) se podría explicar por las siguientes razones:

- Los conjuntos de datos utilizados en el presente trabajo son de mayor tamaño, en especial para los datos de entrenamiento.
- En Jorquera y cols. (1998), no se utilizan como variables predictoras las concentraciones de los precursores del Ozono: Óxidos de Nitrógeno (NO_x), Compuestos Orgánicos Volátiles (COV) y Radiación Solar (RS).

4.2. Concentración horaria

4.2.1. Resumen general

En la tabla 4.4 se muestra el RMSE o error de generalización producido en el dataset de pruebas, junto con la configuración de hiperparámetros asociada y obtenida del proceso de búsqueda *grid search*. Ahí se observa que la Red Long Short Term Memory (LSTM) con arquitectura *many to one* es la que obtiene el mejor rendimiento, aunque no de manera significativa en relación al resto de los modelos¹.

¹En este caso, ARIMA no se implementó para la modalidad de concentración horaria por: su sola relevancia para los máximos diarios -de acuerdo al Estado del Arte en el capítulo 2- y por la imposibilidad de programar la modalidad de concentración horaria, en que se considere como *timesteps* los registros que pertenecen a los días anteriores y no a las horas previas dentro del mismo día al cual pertenece el valor objetivo a predecir.

Tabla 4.4: Resumen concentración horaria.

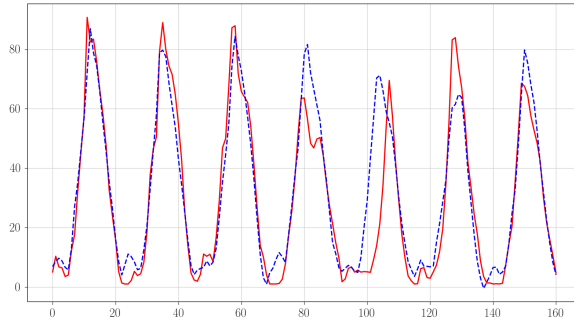
	LR (p)	SVR ($p; C; \gamma$)	FFN ($p; hn$)	LSTM m2o ($p; hn$)	LSTM m2m ($p; hn$)
Hiperparámetros	(15)	(3; 1,0; 0,01)	(15; 5)	(10; 5)	(5; 10)
RMSE	9,750	9,838	9,815	9,496	10,144

En las gráficas de la figura 4.5 se muestran las predicciones hechas por los modelos, tomando como referencia una semana de ejemplo para una mejor visualización.

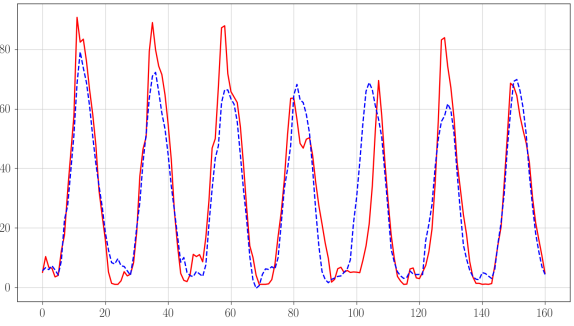
En general, se observa que los valores intermedios de la concentración de Ozono son mejor predichos que los máximos. Esto último podría deberse a que los primeros responden a un comportamiento que exhibe un patrón periódico más claro y estable a los largo del tiempo, a diferencia de los máximos, los cuales presentan un comportamiento más caótico y aleatorio, tal como se observó en el análisis descriptivo (punto 3.2.2) del capítulo anterior. En relación a esto, si se comparan los resultados de la tabla 4.4 con los de la tabla 4.1, se observa que los valores del RMSE de la primera son bastante más bajos que los de la segunda, probablemente empujados por el hecho de que al tratar con la serie horaria completa se procesan datos más fáciles de predecir, dado ese patrón periódico ya mencionado. Dicho sea de paso, si se realiza la selección de la mejor configuración de hiperparámetros en función del RMSE calculado sobre sólo los máximos diarios, se obtienen resultados más equiparables a los de la sección 4.1 (aunque peores en rendimiento), tal como se muestra en la tabla 4.5.

Tabla 4.5: Resumen concentración horaria (solo máximos).

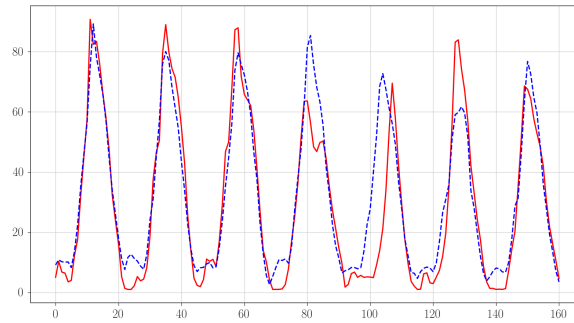
	LR (p)	SVR ($p; C; \gamma$)	FFN ($p; hn$)	LSTM m2o ($p; hn$)	LSTM m2m ($p; hn$)
Hiperparámetros	(15)	(3; 1,0; 0,01)	(10; 10)	(10; 15)	(3; 5)
RMSE	16,700	16,458	15,686	15,398	16,412



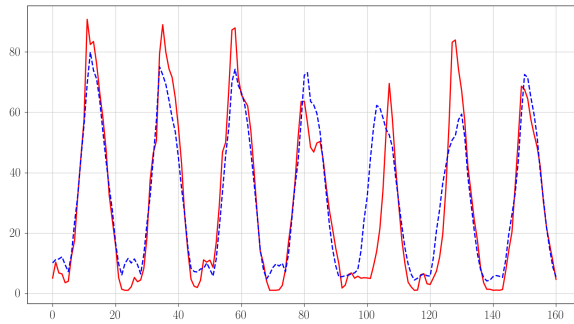
(a) Regresión Lineal.



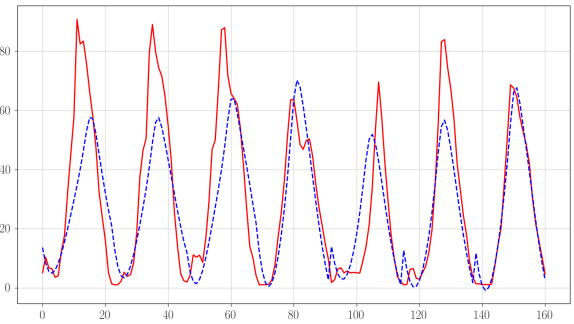
(b) Support Vector Regression.



(c) Feed Forward Neural Network.



(d) LSTM *many to one*.



(e) LSTM *many to many*.

Figura 4.5: Predicciones para el dataset de prueba. En color rojo se grafican los valores reales u observados para la serie de tiempo del Ozono, mientras que con un estilo punteado en azul, se grafican las predicciones realizadas por el modelo respectivo.

4.2.2. Cantidad de *timesteps*

En la figura 4.6 se muestra la variación en el rendimiento de los algoritmos, al aumentar la cantidad de *timesteps* utilizados para predecir el Ozono. Se observa que, a excepción de la

SVR, la utilización de más valores del pasado de la serie de tiempo no incide en el rendimiento de los algoritmos, mostrando un comportamiento parejo en torno a un RMSE que está dentro del rango 9–10. En relación a la SVR, hay que recordar que este no es un algoritmo que posea la capacidad nativa de manejar datos de tipo secuencial. Para posibilitar lo anterior, los valores pasados de todas las variables se deben procesar como entradas explícitas, lo que tiene como consecuencia un aumento de la dimensionalidad del problema y por consiguiente un aumento en su complejidad. Intentar ajustar una función bajo esas condiciones es más complicado, redundando en un decremento de la capacidad de predicción del algoritmo sobre el dataset de validación, hecho que, en este caso, afectó más a la SVR que al modelo de Regresión Lineal o la Red Neuronal Feed Forward cuyo entrenamiento va de la mano con un método de regularización que puede contrarrestar el fenómeno anterior.

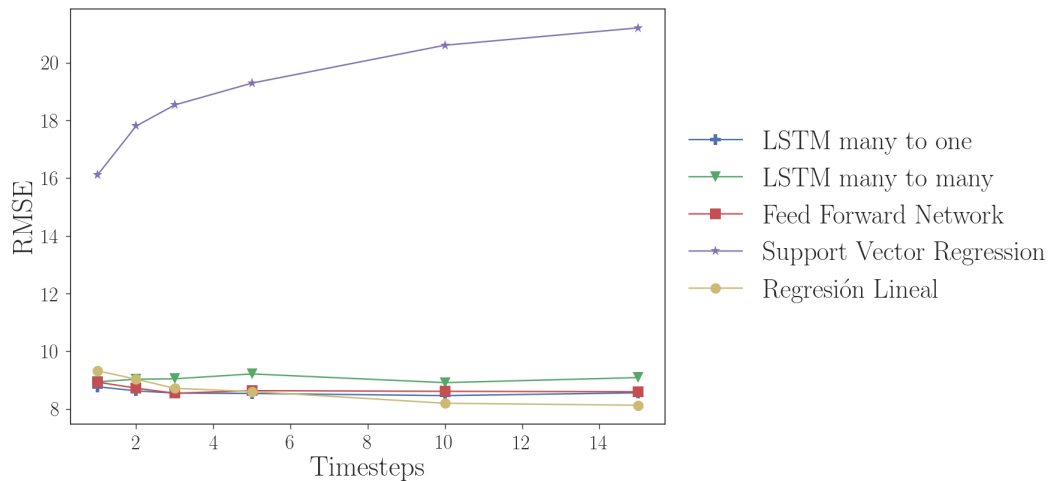


Figura 4.6: RMSE en función del número de valores pasados (*timesteps*).

4.2.3. Redes Neuronales

En la figura 4.7 se grafica la incidencia de la variación del número de neuronas ocultas como hiperparámetro, sobre el rendimiento de los modelos basados en Redes Neuronales. Se observa que, al igual que en la sección 4.1, las redes neuronales otorgan un mejor rendimiento con un número bajo de nodos ocultos, a pesar de que las diferencias en el RMSE no son demasiado significativas entre distintas configuraciones.

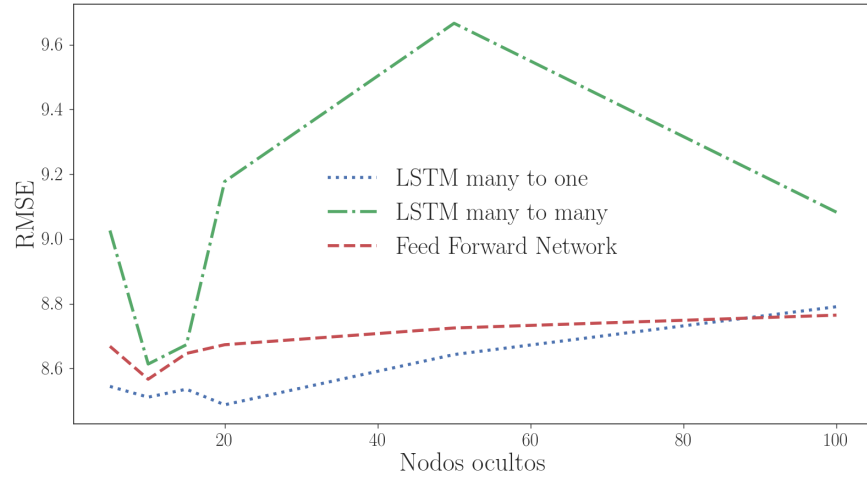
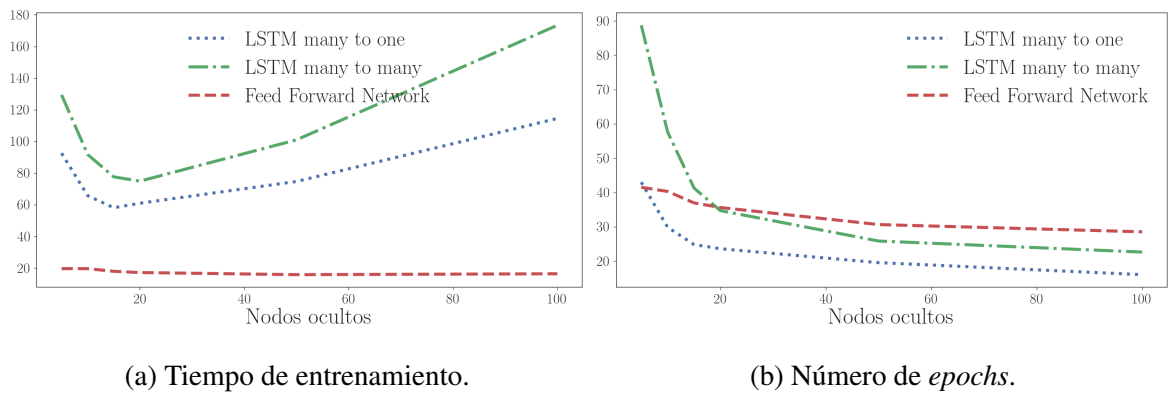


Figura 4.7: RMSE en función de la cantidad de nodos ocultos.

Donde sí incide con mayor notoriedad utilizar más nodos ocultos, es en el tiempo de entrenamiento (figura 4.8a), el cual afecta en mayor medida a las Redes Neuronales Recurrentes. En general, se observan tiempos de entrenamiento que son 10 o más veces los vistos en la sección 4.1, hecho que se explica por la utilización de una mayor cantidad de datos (23 veces más, dado que se predice la serie horaria de todo un día). En relación a los *epochs* (figura 4.8b), se observa una disminución de estos al momento de aumentar la cantidad de nodos ocultos, hecho que se explica de forma análoga a lo acontecido en el punto 4.1.3.



(a) Tiempo de entrenamiento.

(b) Número de *epochs*.

Figura 4.8: Tiempo de entrenamiento (4.4a) y número de *epochs* (4.4b) en función del número de nodos ocultos.

4.3. Disminución del número de variables de entrada

4.3.1. Máximos diarios

En la tabla 4.6, se muestra el RMSE alcanzado en el dataset de pruebas por cada algoritmo, junto con la configuración optimizada de hiperparámetros correspondiente. Además se presenta la variación absoluta y porcentual del RMSE en relación a lo obtenido en la sección 4.1. Notar que la configuración de hiperparámetros, derivada del *grid search*, no coinciden necesariamente.

Se observa que en general, los rendimientos son mejorados respecto de los resultados que se muestran en la sección 4.1 (con excepción de la red Feed Forward). La mayor disminución del RMSE en términos absolutos la logra la Regresión Lineal. Sin embargo, el mejor rendimiento es generado por la Red Neuronal Recurrente LSTM (arquitectura *many to many*), la cual logra disminuir el RMSE en un 5,1 % respecto del valor alcanzado por el mismo algoritmo y en un 4,7 % respecto de lo alcanzado por la Red Neuronal Feed Forward (el con mejor rendimiento), ambos de la sección 4.1.

Respecto del modelo persistente, la Red Neuronal Recurrente LSTM (arquitectura *many to many*) genera una disminución del 12,190 % en el RMSE con el conjunto de pruebas.

El porqué de esta mejora en los rendimientos de los modelos, al disminuir la cantidad de variables de entrada quitando los NO_x y el CO, podría deberse a que la información que aportan tales variables ya se encuentra presente en la serie del Ozono (recordando que esta variable también se está utilizando como predictor), por lo que el aporte de los NO_x y el CO sería redundante. Ahora bien, esto no implica que estas variables sean intrínsecamente irrelevantes en la formación del Ozono, considerando además lo que se vió en el Capítulo 1 en relación a los aspectos teóricos que sustentan la formación del Ozono troposférico.

En las figuras 4.9 y 4.10 se grafica el RMSE de validación en función del número de *time-steps* y el de nodos ocultos respectivamente. Se observa que en ambos aspectos, disminuir la cantidad de variables de predicción redundante en una mejora del rendimiento de cada algoritmo.

Tabla 4.6: Resumen máximos diarios (variables meteorológicas + Ozono).

	LR	SVR	FFN	LSTM m2o	LSTM m2m
	(p)	($p; C; \gamma$)	($p; hn$)	($p; hn$)	($p; hn$)
Hiperparámetros	(10)	(10; 1,0; 0,01)	(5; 5)	(2; 15)	(2; 15)
RMSE	14,789	15,495	15,425	14,617	14,529
Δ RMSE	-0,793	-0,137	0,185	-0,712	-0,781
Δ RMSE %	-5,089	-0,876	1,214	-4,645	-5,101

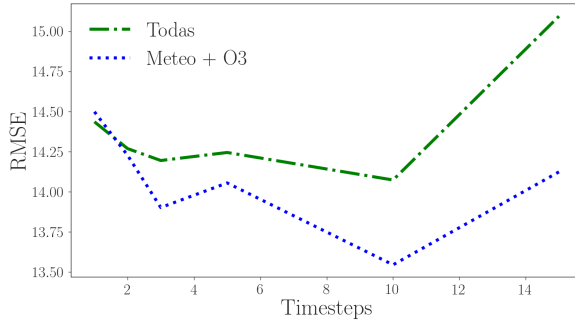
4.3.2. Concentración horaria

En la tabla 4.7, se muestra el RMSE alcanzado en el dataset de pruebas por cada algoritmo, junto con la configuración optimizada de hiperparámetros correspondiente. Además se presenta la variación absoluta y porcentual del RMSE en relación a lo obtenido en la sección 4.2. Notar que la configuración de hiperparámetros, derivada del *grid search*, no coinciden necesariamente.

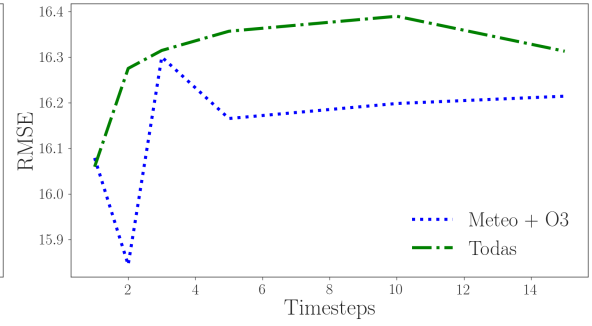
Se observa que existe una mejora general de rendimiento (excepto la Regresión Lineal), respecto de los resultados de la sección 4.2. Sin embargo, esa mejora es menor que la que se da con los máximos diarios. En este caso nuevamente la Red Recurrente LSTM (arquitectura *many to many*) es la que obtiene el mejor desempeño, además de ser la que más disminuyó el RMSE, ya sea en términos absolutos como porcentuales. Comparando este modelo con el que generó el mejor resultado de la sección 4.2, la Red Recurrente LSTM (arquitectura *many to one*), se obtiene una disminución porcentual del RMSE igual al 0,979 % respecto de ésta última.

Tabla 4.7: Resumen concentración horaria (variables meteorológicas + Ozono).

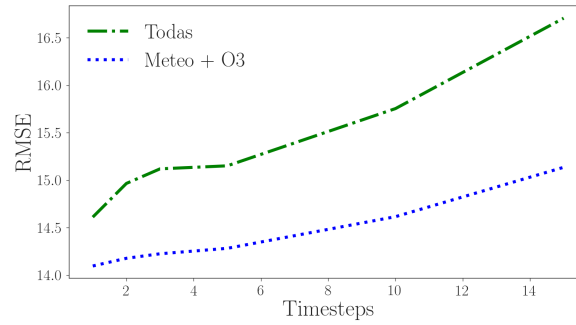
	LR	SVR	FFN	LSTM m2o	LSTM m2m
	(p)	($p; C; \gamma$)	($p; hn$)	($p; hn$)	($p; hn$)
Hiperparámetros	(15)	(3; 1,0; 0,01)	(10; 5)	(10; 5)	(10; 10)
RMSE	9,754	9,589	9,722	9,425	9,403
Δ RMSE	0,004	-0,249	-0,093	-0,071	-0,741
Δ RMSE %	0,041	-2,531	-0,948	-0,748	-7,305



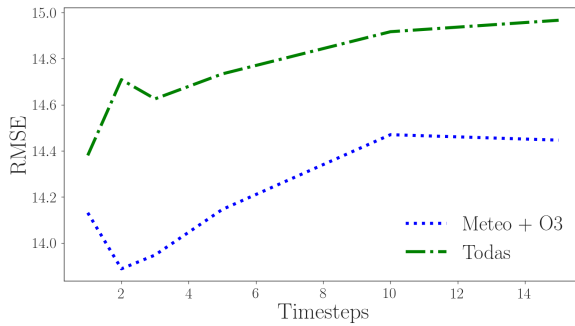
(a) Regresión Lineal.



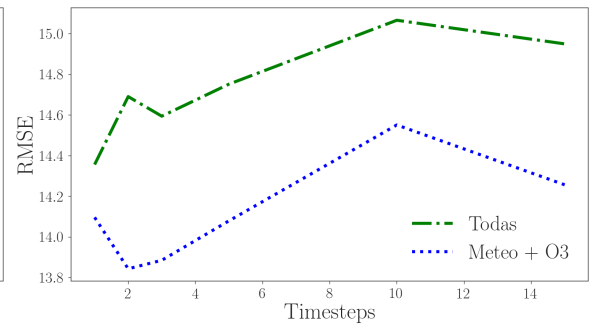
(b) Support Vector Regression.



(c) Feed Forward Neural Network.



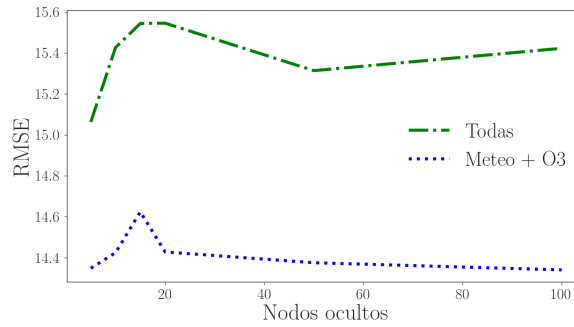
(d) LSTM *many to one*.



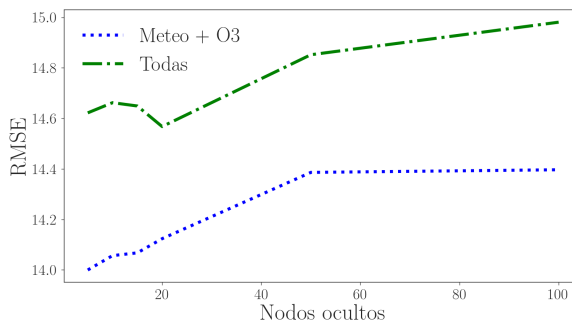
(e) LSTM *many to many*.

Figura 4.9: RMSE en función del número de *timesteps*, considerando los resultados obtenidos al utilizar todas las variables de predicción y los resultados de usar sólo las variables meteorológicas más el Ozono.

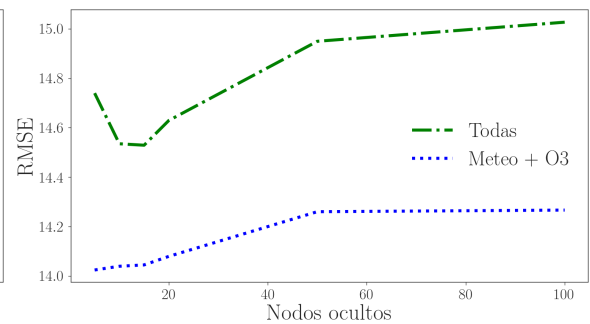
En relación a la variación del RMSE en función de la cantidad de *timesteps*, esta se puede ver en las gráficas de la figura 4.11. En general, al disminuir la cantidad de variables de entrada se mejora el rendimiento de los modelos, a excepción de la Regresión Lineal (figura 4.11a). En el caso de la SVR (figura 4.11b), en ambos casos el rendimiento empeora a medida que



(a) Feed Forward Neural Network.



(b) LSTM *many to one*.

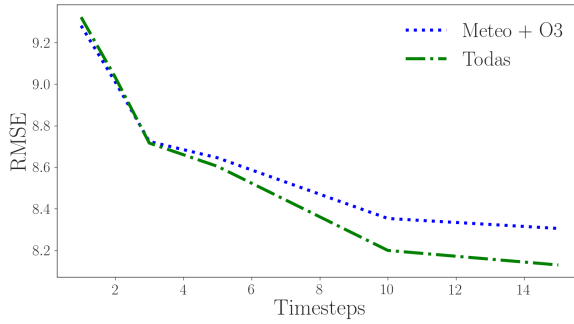


(c) LSTM *many to many*.

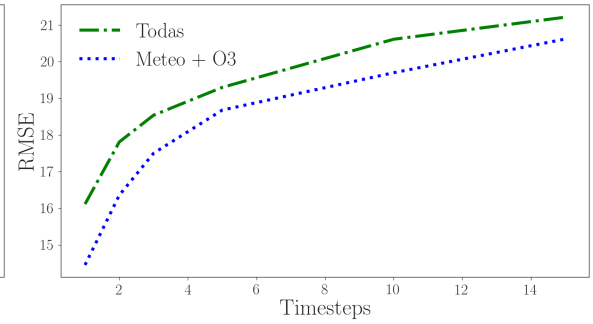
Figura 4.10: RMSE en función del número de nodos ocultos de los modelos basados en redes neuronales, considerando los resultados obtenidos al utilizar todas las variables de predicción y los resultados de utilizar sólo las variables meteorológicas más el Ozono.

se utiliza una mayor cantidad de valores pasados de las variables. En torno a la red Feed Forward, es donde se aprecia la menor diferencia. Y por otro lado, las redes recurrentes son las que presentan las diferencias más visibles a favor del uso de menos variables de entrada.

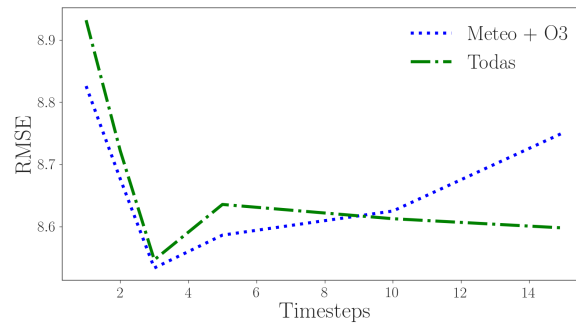
Finalmente, respecto de la cantidad de nodos ocultos en los modelos basados en redes neuronales, la variación del RMSE en cada caso se puede observar en las gráficas de la figura 4.12. Ahí se observa que en los tres casos, la utilización de una menor cantidad de variables de entrada (Ozono + variables meteorológicas), redundará en un RMSE menor.



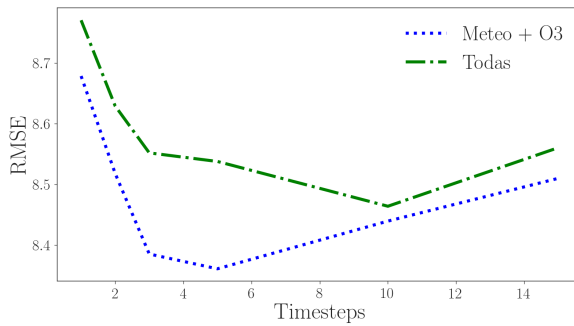
(a) Regresión Lineal.



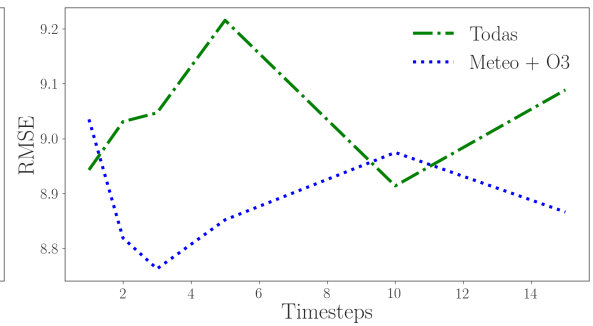
(b) Support Vector Regression.



(c) Feed Forward Neural Network.

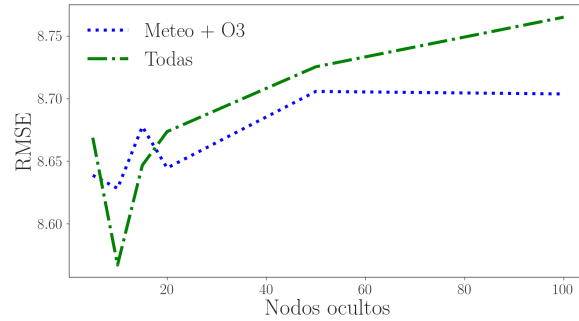


(d) LSTM *many to one*.

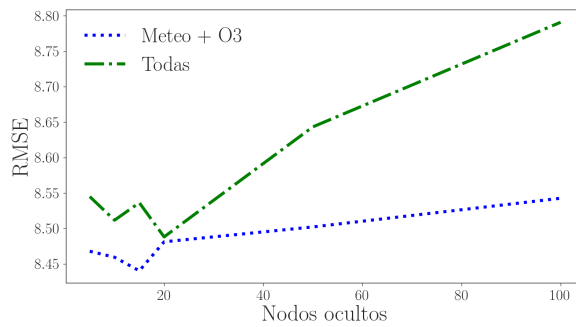


(e) LSTM *many to many*.

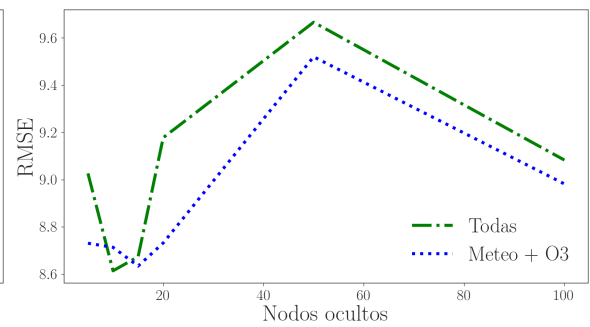
Figura 4.11: RMSE en función del número de *timesteps* de los modelos basados en redes neuronales, considerando los resultados obtenidos al utilizar todas las variables de predicción y los resultados de usar sólo las variables meteorológicas más el Ozono.



(a) Feed Forward Neural Network.



(b) LSTM *many to one*.



(c) LSTM *many to many*.

Figura 4.12: RMSE en función del número de nodos ocultos, considerando los resultados obtenidos al utilizar todas las variables de predicción y los resultados de usar sólo las variables meteorológicas más el Ozono.

Conclusiones

En este trabajo, se han desarrollado e implementado diversos modelos predictivos para la concentración de Ozono troposférico en la ciudad de Santiago, en base a los datos provenientes de entre otras fuentes, el Sistema de Información Nacional de Calidad del Aire (SINCA), poniendo el foco en los periodos del año con los niveles más altos del contaminante. Basándose la mayoría de tales modelos en técnicas de Aprendizaje Automático, los resultados muestran rendimientos bastante competitivos entre unos y otros, no evidenciándose una clara supremacía de alguno por sobre el resto. Se consideraron dos modalidades u objetivos de predicción: (1) el máximo diario de la concentración de Ozono y (2) la concentración horaria del Ozono, que en términos prácticos es una extensión de la primera modalidad al resto de cada hora del día.

Para la modalidad del máximo diario, un modelo basado en Redes Neuronales Feed Forward es el que obtiene los mejores resultados. No muy lejos, un modelo ARIMA para series de tiempo es el que lo secunda con cierta sorpresa, dado que es un tipo de modelo que utiliza solo los valores del pasado del Ozono obviando el resto de las variables de entrada. En tanto que para la modalidad de concentración horaria, un modelo basado en redes neuronales recurrentes LSTM es el que obtuvo los mejores resultados.

En otros estudios, específicamente en el de Jorquera y cols. (1998), se desarrollan modelos - basados en redes neuronales, modelado difuso y series de tiempo- para predecir los máximos diarios de la concentración de Ozono, los cuales se validan utilizando un conjunto de datos con características similares a las usadas en el presente trabajo: provenientes de la estación de monitoreo ubicada en Las Condes y pertenecientes a un periodo del año similar (Noviembre-Febrero). Pues bien, los resultados en torno al rendimiento de los modelos -cuantificado por

el RMSE- en el presente estudio son aproximadamente un 50 % mejores que los obtenidos por Jorquera y cols. (1998). Este hecho probablemente se deba a que principalmente, en el presente trabajo, se cuenta con una mayor cantidad de variables de entrada y/o se utiliza un conjunto de entrenamiento de mayores dimensiones, haciendo poco presumible que sea por el tipo de modelos implementados.

Abordando ciertos puntos específicos analizados en el presente trabajo, los resultados muestran que tanto para la modalidad de máximos diarios como de la concentración horaria, los algoritmos basados en Aprendizaje Automático no mejoran de forma evidente su desempeño, al utilizar una mayor cantidad de valores pasados de las variables de entrada e incluso en algunos casos, empeorando los resultados. Esto hace suponer que en general, solo basta con la información del pasado más próximo para predecir las concentraciones de Ozono del día siguiente. En términos prácticos, lo anterior implica el poder formular modelos más simples que deriven en un menor tiempo de entrenamiento.

A pesar de que con la modalidad de predicción de la concentración horaria de Ozono, se obtiene un rendimiento superior a la modalidad de máximos diarios -reflejado por la obtención de un menor RMSE-, es preferible utilizar esta última si es que el énfasis del análisis predictivo son los *peak* diarios de Ozono, teniendo como ventaja el tener que manejar una menor cantidad de datos y por ende, entrenar modelos en una menor cantidad de tiempo. Dichos resultados se vieron empujados por el hecho de que en la modalidad de concentración horaria, se incluyen objetivos que son más fáciles de predecir dado que responden a un comportamiento más homogéneamente observable en el tiempo.

También, a través del análisis descriptivo de los datos, se observó que la temperatura del aire es la variable de entrada con mayor importancia respecto de la concentración de Ozono, si es que se considera la correlación entre tales variables como el indicador decisivo, hecho que se replica en otros estudios previos. En esa misma línea, son las variables de carácter meteorológico las que explican en mayor magnitud los niveles de Ozono en la tropósfera. Lo anterior trajo como consecuencia la generación y evaluación de modelos predictivos que prescinden de las variables de entrada del tipo contaminante (Óxidos de Nitrógeno (NO_x))

y Monóxido de Carbono (CO)), en que cuyos resultados muestran una mejora del rendimiento si es que se le compara con los modelos que utilizan todas las variables de entrada. Bajo esta lógica, los modelos basados en Redes Neuronales Recurrentes LSTM son los que mejoraron en mayor medida su desempeño, especialmente el que utiliza la arquitectura *encoder-decoder*.

Como conclusión final, el desarrollo del presente trabajo de memoria ha permitido legar una base de código fuente mediante el cual, se ejecutaron una serie de experimentos cuyos resultados proveen de una base de conocimientos que servirá de punto de partida para la realización de un estudio más detallado del problema, con miras a una tesis de postgrado. En ese sentido, respecto del trabajo futuro potencial a realizar, y teniendo como objetivo principal mejorar los rendimientos aquí registrados, este se podría dividir en los siguientes aspectos:

- Realizar un análisis descriptivo más profundo en torno a la producción del Ozono y las distintas variables involucradas.
 - Si bien se encontró que el obviar la inclusión de las variables contaminantes incrementa el rendimiento de los modelos, esto puede redundar en la pérdida de información valiosa que podría ser aprovechada con una mejor selección de las variables de entrada. Esto último pasaría por utilizar otras estrategias para cuantificar la importancia de cada variable de entrada, respecto del nivel registrado de Ozono troposférico, como por ejemplo la Información Mutua (*Mutual Information*), la cual es una medida de la dependencia mutua entre dos variables.
 - Realizar un análisis de correlación del Ozono, registrado en el tiempo t , con los valores pasados (en los tiempos $t - 1, t - 2, \dots, t - p$) del resto de las variables contaminantes y meteorológicas.
- Reformular la configuración de los modelos basados en Aprendizaje Automático, especialmente el de las Redes Neuronales Recurrentes.
 - Explorar de manera más extensa otras configuraciones de hiperparámetros, los cuales pueden modificar en forma relevante el comportamiento de los modelos aquí desarrollados. Las Redes Neuronales se caracterizan por poseer una gran

cantidad de hiperparámetros candidatos a modificar y que pudiesen alterar los resultados obtenidos: número de capas ocultas, cantidad de nodos por capa oculta, funciones de activación, tamaño del *batch* de entrenamiento, algoritmos de optimización de gradiente descendente, entre otros.

- Estudiar en mayor detalle la modalidad *stateful* de las Redes Neuronales Recurrentes LSTM, de acuerdo a la librería *keras* de Python.
- Extender el análisis y evaluación de los modelos a otros períodos del año.
- Ampliar las fuentes de datos considerando otras estaciones de monitoreo en la ciudad de Santiago.
- Modelar el problema de predicción de la concentración del Ozono como uno de clasificación, determinando la existencia o no de un episodio crítico de emergencia ambiental, de acuerdo a las normas primarias de calidad del aire para Ozono vistas en la sección 1.1.

Bibliografía

- Abdul-Wahab, S. A., y Al-Alawi, S. M. (2002). Assessment and prediction of tropospheric ozone concentration levels using artificial neural networks. *Environmental Modelling & Software*, 17(3), 219–228.
- Agencia de Protección Ambiental de Estados Unidos. (2015). *2015 national ambient air quality standards (naaqs) for ozone*. <https://www.epa.gov/ozone-pollution/2015-national-ambient-air-quality-standards-naaqs-ozone>. (Accedido el 2017-06-01)
- Agencia Europea de Medio Ambiente. (2012). *Air quality in europe — 2012 report*. <https://www.eea.europa.eu/publications/air-quality-in-europe-2012>. (Accedido el 2017-06-01)
- Agencia Europea de Medio Ambiente. (2016). *Air quality in europe — 2016 report*. <https://www.eea.europa.eu/publications/air-quality-in-europe-2016>. (Accedido el 2017-06-01)
- Agirre-Basurko, E., Ibarra-Berastegi, G., y Madariaga, I. (2006). Regression and multilayer perceptron-based models to forecast hourly o₃ and no₂ levels in the bilbao area. *Environmental Modelling & Software*, 21(4), 430–446.
- Al-Alawi, S. M., Abdul-Wahab, S. A., y Bakheit, C. S. (2008). Combining principal component regression and artificial neural networks for more accurate predictions of ground-level ozone. *Environmental Modelling & Software*, 23(4), 396–403.
- Amann, M., Derwent, D., Forsberg, B., Hänninen, O., Hurley, F., Krzyzanowski, M., . . . Simpson, D. (2008). *Health risks of ozone from long-range transboundary air pollution*. World Health Organization.
- Biancofiore, F., Verdecchia, M., Di Carlo, P., Tomassetti, B., Aruffo, E., Busilacchio, M., . . .

- Colangeli, C. (2015). Analysis of surface ozone using a recurrent neural network. *Science of the Total Environment*, 514, 379–387.
- Biblioteca del Congreso Nacional de Chile. (s.f.). *Clima y vegetación región metropolitana de santiago*. <http://www.bcn.cl/siit/nuestropais/region13/clima.htm>. (Accedido el 2017-07-03)
- Chaloulakou, A., Assimacopoulos, D., y Lekkas, T. (1999). Forecasting daily maximum ozone concentrations in the athens basin. *Environmental Monitoring and Assessment*, 56(1), 97–112.
- Chatfield, C. (2000). *Time-series forecasting*. CRC Press.
- Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., y Bengio, Y. (2014). Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*.
- Coman, A., Ionescu, A., y Candau, Y. (2008). Hourly ozone prediction for a 24-h horizon using neural networks. *Environmental Modelling & Software*, 23(12), 1407–1421.
- Cordero Raúl R, D. A. (2014). *Ozono y radiación uv respuestas a las preguntas claves*. <http://antarctica.cl/wp/wp-content/uploads/2017/06/OzonoyRadiacionUV.compressed3.pdf>. (Accedido el 2016-10-07)
- Dutot, A.-L., Rynkiewicz, J., Steiner, F. E., y Rude, J. (2007). A 24-h forecast of ozone peaks and exceedance levels using neural classifiers and weather predictions. *Environmental Modelling & Software*, 22(9), 1261–1269.
- Elman, J. L. (1990). Finding structure in time. *Cognitive science*, 14(2), 179–211.
- Gardner, M., y Dorling, S. (2000). Statistical surface ozone models: an improved methodology to account for non-linear behaviour. *Atmospheric Environment*, 34(1), 21–34.
- Gómez, P., Nebot, A., Ribeiro, S., Alquézar, R., Mugica, F., y Wotawa, F. (2003). Local maximum ozone concentration prediction using soft computing methodologies. *Systems analysis modelling simulation*, 43(8), 1011–1031.
- Goodfellow, I., Bengio, Y., y Courville, A. (2016). *Deep learning*. MIT Press. (<http://www.deeplearningbook.org>)
- Gorai, A., y Mitra, G. (2017). A comparative study of the feed forward back propagation (ffbp) and layer recurrent (lr) neural network model for forecasting ground level ozone concentration. *Air Quality, Atmosphere & Health*, 10(2), 213–223.

- Hájek, P., y Olej, V. (2012). Ozone prediction on the basis of neural networks, support vector regression and methods with uncertainty. *Ecological Informatics*, 12, 31–42.
- Hochreiter, S., y Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, 9(8), 1735–1780.
- Instituto Nacional de Estadísticas. (2017). *Entrega de resultados preliminares*. <http://www.censo2017.cl/wp-content/uploads/2017/08/Proceso-Censal-Resultados-preliminares-31-08-2017.pdf>. (Accedido el 20107-10-10)
- James, G., Witten, D., Hastie, T., y Tibshirani, R. (2013). *An introduction to statistical learning* (Vol. 112). Springer.
- Jorquera, H., Pérez, R., Cipriano, A., Espejo, A., Letelier, M. V., y Acuña, G. (1998). Forecasting ozone daily maximum levels at santiago, chile. *Atmospheric Environment*, 32(20), 3415–3424.
- Karpathy, A. (2015). *The unreasonable effectiveness of recurrent neural networks*. <https://karpathy.github.io/2015/05/21/rnn-effectiveness/>. (Accedido el 2017-05-03)
- Kingma, D., y Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.
- Kumar, K., Yadav, A., Singh, M., Hassan, H., y Jain, V. (2004). Forecasting daily maximum surface ozone concentrations in brunei darussalam—an arima modeling approach. *Journal of the Air & Waste Management Association*, 54(7), 809–814.
- Kumar, U., y Jain, V. (2010). Arima forecasting of ambient air pollutants (o₃, no, no₂ and co). *Stochastic Environmental Research and Risk Assessment*, 24(5), 751–760.
- Lin, X., Trainer, M., y Liu, S. (1988). On the nonlinearity of the tropospheric ozone production. *Journal of Geophysical Research: Atmospheres*, 93(D12), 15879–15888.
- Lipton, Z. C., Berkowitz, J., y Elkan, C. (2015). A critical review of recurrent neural networks for sequence learning. *arXiv preprint arXiv:1506.00019*.
- Luna, A., Paredes, M., de Oliveira, G., y Corrêa, S. (2014). Prediction of ozone concentration in tropospheric levels using artificial neural networks and support vector machine at rio de janeiro, brazil. *Atmospheric Environment*, 98, 98–104.

- Ministerio de Salud. (2006). *Clasifica como estaciones de monitoreo de calidad de aire con representacion poblacional para gases de monoxido de carbono (co), ozono (o3) y dióxido de azufre (so2)*. <https://www.leychile.cl/Navegar?idNorma=245880&idParte=&idVersion=2006-01-04>. (Accedido el 2017-06-01)
- Ministerio del Medio Ambiente Chile. (s.f.). *Glosario de términos*. <http://sinca.mma.gob.cl/index.php/pagina/index/id/glosario>. (Accedido el 2016-10-01)
- Ministerio del Medio Ambiente Chile. (2015). *Anteproyecto del plan de prevención y descontaminación atmosférica para la región metropolitana de santiago*. http://santiagoorepira.gob.cl/pdf/Anteproyecto_del_Plan_de_Preencion_y_Descontaminacion_atmosferica_para_la_Region_Metropolitana_de_Santiago.pdf. (Accedido el 2017-07-05)
- Ministerio Secretaría General de la Presidencia. (s.f.-a). *Decreto de ley 112: establece norma primaria de calidad de aire para ozono (o3)*. <https://www.leychile.cl/Navegar?idNorma=208198>. (Accedido el 2016-10-01)
- Ministerio Secretaría General de la Presidencia. (s.f.-b). *Decreto supremo 131: Declaración zona saturada por ozono y otros contaminantes*. <https://www.leychile.cl/Navegar?idNorma=9768>. (Accedido el 2016-10-04)
- Mitchell, T. (1997). *Machine learning*. McGraw Hill.
- Organización Mundial de la Salud. (2005). *Air quality guidelines. global update 2005. particulate matter, ozone, nitrogen dioxide and sulfur dioxide*. http://whqlibdoc.who.int/hq/2006/WHO_SDE_PHE_OEH_06.02_spa.pdf. (Accedido el 2017-06-01)
- Ortiz-García, E., Salcedo-Sanz, S., Pérez-Bellido, Á., Portilla-Figueras, J., y Prieto, L. (2010). Prediction of hourly o3 concentrations using support vector regression algorithms. *Atmospheric Environment*, 44(35), 4481–4488.
- Özbay, B., Keskin, G. A., Doğruparmak, Ş. Ç., y Ayberk, S. (2011). Predicting tropospheric ozone concentrations in different temporal scales by using multilayer perceptron models. *Ecological Informatics*, 6(3), 242–247.
- Prechelt, L. (1998). Early stopping-but when? *Neural Networks: Tricks of the trade*, 553–553.
- Ribeiro, S., y Alquézar, R. (2002). Local maximum ozone concentration prediction using lstm recurrent neural networks.

- Robeson, S., y Steyn, D. (1990). Evaluation and comparison of statistical forecast models for daily maximum ozone concentrations. *Atmospheric Environment. Part B. Urban Atmosphere*, 24(2), 303–312.
- Salazar-Ruiz, E., Ordieres, J., Vergara, E., y Capuz-Rizo, S. (2008). Development and comparative analysis of tropospheric ozone prediction models using linear and artificial intelligence-based models in mexicali, baja california (mexico) and calexico, california (us). *Environmental Modelling & Software*, 23(8), 1056–1069.
- Shlens, J. (2014). A tutorial on principal component analysis. *CoRR*, abs/1404.1100. Descargado de <http://arxiv.org/abs/1404.1100>
- Simpson, R., y Layton, A. (1983). Forecasting peak ozone levels. *Atmospheric Environment (1967)*, 17(9), 1649–1654.
- Smola, A. J., y Schölkopf, B. (2004). A tutorial on support vector regression. *Statistics and computing*, 14(3), 199–222.
- Sousa, S., Martins, F., Alvim-Ferraz, M., y Pereira, M. C. (2007). Multiple linear regression and artificial neural networks based on principal components to predict ozone concentrations. *Environmental Modelling & Software*, 22(1), 97–103.
- Sutskever, I., Vinyals, O., y Le, Q. V. (2014). Sequence to sequence learning with neural networks. En *Advances in neural information processing systems* (pp. 3104–3112).
- Universidad de Santiago de Chile. (2014). *Actualización y sistematización del inventario de emisiones de contaminantes atmosféricos en la Región Metropolitana*.
- Wang, W., Lu, W., Wang, X., y Leung, A. Y. (2003). Prediction of maximum daily ozone level using combined neural network and statistical characteristics. *Environment international*, 29(5), 555–562.
- Yi, J., y Prybutok, V. R. (1996). A neural network model forecasting for prediction of daily maximum ozone concentration in an industrialized urban area. *Environmental pollution*, 92(3), 349–357.