



UNIVERSIDAD DE LA REPÚBLICA
FACULTAD DE INGENIERÍA



Detección de pérdidas no técnicas de energía utilizando modelos de aprendizaje profundo para series temporales multivariadas

TESIS PRESENTADA A LA FACULTAD DE INGENIERÍA DE LA
UNIVERSIDAD DE LA REPÚBLICA POR

Fernanda Maldonado

EN CUMPLIMIENTO PARCIAL DE LOS REQUERIMIENTOS
PARA LA OBTENCIÓN DEL TÍTULO DE
MAGISTER EN CIENCIA DE DATOS Y APRENDIZAJE AUTOMÁTICO.

DIRECTORES DE TESIS

Dr. Ing. Pablo Massaferró Universidad de la República
Dra. Ing. Lorena Etcheverry Universidad de la República

TRIBUNAL

Dr. Ing. J. Matias Di Martino Duke University
Mg. Ing. Juan Pablo Kosut Universidad de la República
Dr. Ing. Guillermo Moncecchi Universidad de la República

DIRECTOR ACADÉMICO

Dra. Ing. Lorena Etcheverry Universidad de la República

Montevideo
jueves 11 julio, 2024

Detección de pérdidas no técnicas de energía utilizando modelos de aprendizaje profundo para series temporales multivariadas, Fernanda Maldonado.

ISSN 1688-2806

Esta tesis fue preparada en L^AT_EX usando la clase iietesis (v1.1).

Contiene un total de 91 páginas.

Compilada el jueves 11 julio, 2024.

<http://iie.fing.edu.uy/>

Resumen

Las pérdidas no técnicas conocidas como NTL por su sigla en inglés (*non-technical losses*) en el sector energético presentan un desafío significativo tanto para las empresas proveedoras de servicios públicos como para las economías globales. Estas pérdidas, que abarcan desde el robo hasta la manipulación de medidores, resultan en una disminución de los ingresos, distorsiones en los patrones de consumo y comprometen los suministros de energía. Sus repercusiones afectan tanto a las empresas como a los consumidores, ya que la detección efectiva de un consumo irregular requiere inspecciones *in situ* con personal especializado, lo cual genera un costo extra para las empresas que termina afectando al precio de la energía. Para optimizar la selección de instalaciones a inspeccionar, es fundamental contar con algoritmos que evalúen la probabilidad de fraude, minimizando así las inspecciones innecesarias. Este tema adquiere cada vez más relevancia, ya que no solo se centra en la estabilidad financiera de las empresas, sino que también busca garantizar un acceso equitativo, seguro y confiable a los recursos energéticos.

Dado el impacto que tienen las NTL en diferentes ámbitos como financiero y medioambiental, la literatura existente ofrece varios trabajos relacionados con este tema. Sin embargo, son escasos los estudios que abordan el problema mediante el análisis de series temporales multivariadas. Es importante destacar que esta tesis se basa en datos reales, prescindiendo por completo de datos sintéticos.

Mediante la combinación de la infraestructura de medición avanzada, en inglés *Advanced Metering Infrastructure* (AMI), la ciencia de datos y el aprendizaje profundo, se busca abordar este problema con un enfoque innovador. Probando distintos tipos de arquitecturas de redes neuronales tales como convolucionales, recurrentes y residuales. La AMI permite recopilar datos con una resolución quinceminutal, lo que proporciona mediciones más precisas y frecuentes del consumo de energía, mejorando así las capacidades de monitoreo. La fusión de esta rica fuente de datos con redes neuronales permite revelar conocimientos ocultos dentro de las series temporales multivariadas, que son difíciles de capturar mediante métodos tradicionales. Además, se explora la correlación entre el consumo de energía y un factor externo: la temperatura ambiente. Al aprovechar las variaciones en la temperatura ambiente y su relación con el consumo de energía, se analiza cómo estas fluctuaciones afectan la demanda de energía, especialmente en lo que respecta a la calefacción y refrigeración. Por tanto, este estudio se centra en el análisis de series temporales multivariadas que incluyen el consumo de energía activa, energía reactiva y la temperatura ambiente.

En esta tesis se examinan los desafíos y las oportunidades que presenta la integración de AMI y el aprendizaje profundo supervisado, desde el preprocesamiento de datos y el entrenamiento de diferentes modelos hasta la interpretación de resultados, con el fin de abordar de manera proactiva las pérdidas no técnicas y buscando ser más eficientes a la hora de inspeccionar a los clientes. Se busca crear nuevos algoritmos que ayuden a seleccionar qué clientes inspeccionar. Los principales aportes de esta

tesis son: (i) recolección y creación de un conjunto de datos con series temporales de energía activa, reactiva y temperatura ambiente en resolución horaria y con un historial de un año para 50k clientes de UTE, necesarios para el análisis y evaluación de los resultados, (ii) comparación de dos técnicas de normalización de series temporales, (iii) estudio de cómo se correlaciona la temperatura ambiente en el consumo energético, (iv) aplicación de diferentes tipos arquitecturas de aprendizaje profundo (convolucional, recurrente y residual) en series temporales multivariadas.

Los resultados de esta tesis continúan reafirmando que los datos de alta resolución, correspondientes a series temporales horarias, poseen un mayor potencial en comparación con los datos de baja resolución, como los consumos mensuales. Además, muestra que la temperatura ambiente influye en los patrones de consumo de energía, desempeñando así un papel indirecto en las NTL. Este hallazgo no solo abre nuevas oportunidades en la detección de NTL, sino que también conduce al desarrollo de una nueva arquitectura para abordar este tema.

Tabla de contenidos

Resumen	I
1. Introducción	1
1.1. Motivación	1
1.2. Descripción de objetivos y alcance	3
1.2.1. Preguntas de investigación	4
1.2.2. Alcance de la tesis	4
1.3. Estructura de la tesis	4
2. Estado del arte en clasificación de series temporales multivariadas y detección automática de NTL	5
2.1. Introducción	5
2.2. Clasificación de series temporales univariadas y multivariadas	5
2.3. Enfoques para la clasificación de series	7
2.3.1. Modelos generativos	7
2.3.2. Modelos discriminativos	8
2.3.3. Elección de enfoque	9
2.4. Trabajos relacionados a NTL	9
2.5. Detección de NTL usando aprendizaje profundo	12
2.5.1. Redes neuronales convolucionales CNN	12
2.5.2. Redes neuronales recurrentes: RNN y LSTM	14
2.5.3. Redes residuales ResNet	16
2.6. Métricas para la evaluación de los modelos	17
3. Uso de temperatura ambiente en la detección de NTL	19
3.1. Introducción	19
3.2. Técnicas para estudiar el vínculo entre series temporales	20
3.2.1. Coeficiente de correlación de Pearson	20
3.2.2. Correlación cruzada	20
3.2.3. Técnica seleccionada	21
3.3. Datos utilizados	21
3.3.1. Datos de medidores inteligentes	21
3.3.2. Obtención y extracción de los datos de medidores inteligentes	22
3.3.3. Características adicionales	23
3.3.4. Recolección de datos de temperatura	24
3.3.5. Preparación del conjunto de datos	25
3.4. Correlación entre temperatura ambiente y consumo energético	29
3.5. Evaluación de nueva característica de correlación frente a características ya conocidas	31
3.5.1. Preparación del conjunto de datos	32

Tabla de contenidos

3.5.2. Definición del modelo y métricas para su evaluación	33
3.5.3. Evaluación del modelo y conclusiones	34
3.6. Conclusiones	36
4. Detección de NTL en series temporales multivariadas	39
4.1. Introducción	39
4.2. Datos utilizados	39
4.2.1. Organización del conjunto de datos	39
4.2.2. Preparación de los datos	41
4.2.3. Normalización en series temporales	42
4.2.4. Experimento con distintas normalizaciones	43
4.3. Estrategia para la búsqueda de hiperparámetros	46
5. Experimentos con redes neuronales profundas	49
5.1. Introducción	49
5.1.1. Datos utilizados	49
5.2. Experimentos con distintas arquitecturas de redes neuronales usando series de consumo de energía activa, reactiva y temperatura ambiente.	50
5.2.1. Definición de arquitectura basada en redes convolucionales	50
5.2.2. Definición de arquitectura basada en redes residuales	51
5.2.3. Definición de arquitectura basada en redes recurrentes	51
5.2.4. Comparativa entre arquitecturas	52
5.3. Experimentos variando la cantidad de series temporales a utilizar	54
5.4. Experimentos aumentando el largo de las series temporales	56
5.5. Evaluación de tiempos de optimización y entrenamiento de los modelos	58
6. Conclusiones	61
6.1. Conclusiones	61
6.2. Trabajo Futuro	62
Apéndice	64
A. Experimento con Fully Convolutional Neural Network	65
B. Experimento con serie de energía activa en resolución quinceminutal	69
Referencias	73
Índice de tablas	78
Índice de figuras	81

Capítulo 1

Introducción

1.1. Motivación

La electricidad desempeña un papel crucial en la vida moderna, siendo fundamental para una amplia gama de actividades cotidianas. La misma está tomando cada vez más protagonismo en el transporte a través de los vehículos eléctricos. A nivel mundial en 2020, 1 de cada 25 autos vendidos era eléctrico y en 2023 1 de cada 5 [3]. Otro incentivo ha sido el uso extendido de bombas de calor para calefacción y refrigeración [2]. A su vez, la demanda energética se vio desafiada por la crisis energética desencadenada por la invasión de Rusia a Ucrania. La misma ha traído como consecuencia un aumento en el precio de los combustibles que termina impactando en el costo de la generación de electricidad en todo el mundo, presionando a la baja del consumo en muchas regiones. Pese a esta crisis, la demanda mundial se mantuvo resistente, aumentando casi un 2 % en 2022 según el informe de mercado eléctrico publicado por la IEA (International Energy Agency) en Febrero 2023 [7].

Por otro lado, la generación de energía es actualmente la principal fuente de emisiones de dióxido de carbono (CO_2) a nivel mundial. A pesar de esto, el sector energético está liderando la transición hacia un futuro de cero emisiones netas mediante el aumento de la participación de energías renovables, como la eólica y la solar. En Uruguay, si se dan las condiciones de lluvias habituales, casi la totalidad (97 %) de la demanda eléctrica del país se satisface utilizando fuentes renovables de energía, que incluyen energía eólica, biomasa, solar e hidráulica según informe [1] publicado en Noviembre 2023.

Por lo tanto, ya sea por un tema de medioambiental o de crisis energética, la electricidad ha pasado a ocupar un lugar destacado en la agenda política de muchos países. Debido a lo valioso que es este recurso y a que el consumo mundial va en crecimiento, se vuelve cada vez más importante hacer un uso eficiente y promover un uso responsable. Es fundamental que su transmisión, distribución y uso se haga de forma eficaz. Se sabe que en toda empresa eléctrica, existen pérdidas las cuales se pueden clasificar entre técnicas y no técnicas. Por un lado, las pérdidas técnicas, son inherentes al transporte y distribución de la electricidad dentro de la red (consecuencia del calentamiento natural de transformadores y conductores que la transportan). Si bien las pérdidas técnicas no pueden ser eliminadas por completo, es posible reducirlas mediante mejoras en la red.

Y por otra parte, se tienen las pérdidas no técnicas, que corresponden a la energía consumida que no se factura. Aunque diversas problemáticas contribuyen a este tipo de pérdidas, una proporción considerable está relacionada con el hurto/fraude, que

Capítulo 1. Introducción

incluye conexiones ilegales a la red eléctrica y manipulación de medidores para reducir el consumo de energía y pagar menos. En general, los niveles de pérdidas eléctricas, son reflejo del nivel de eficiencia de la infraestructura en transmisión y distribución. Así como también son un indicador del desempeño operacional de las empresas. Según estadísticas de pérdidas de electricidad de la base de datos de la Organización Latinoamericana de Energía (OLADE) [12], el índice de pérdidas en los sistemas de transmisión y distribución esta tendiendo a la baja. Luego de permanecer estable en un 15 % durante 6 años (desde 2014 hasta 2019), en el 2020 fue de 14 % y para 2021 fue de 13 % en la región de América Latina y el Caribe. En Uruguay, a Diciembre de 2020 el indicador de pérdidas de energía a nivel de distribución y comercialización era de 18.3 %. Este indicador se encuentra disminuyendo año a año, obteniendo en Diciembre 2021 17.3 %, en Diciembre 2022 16.3 % y en Diciembre 2023 14.9 %, según informes de balances energéticos de la Administración Nacional de Usinas y Transmisiones Eléctricas (UTE). El objetivo es que para Diciembre de 2027 este indicador baje a 14 %.

Para combatir y controlar las pérdidas, es necesario realizar inspecciones en las instalaciones de los clientes. Las inspecciones deben ser realizadas por técnicos especializados en el área, por lo que exige una organización y planificación, las cuales generan consigo costos operativos para la empresa. Por lo tanto, definir a qué clientes inspeccionar es el gran desafío, y para ello, se busca seguir construyendo algoritmos que detecten automáticamente fraudes y ayuden a focalizar las inspecciones en aquellos clientes donde se tiene una sospecha de que con alta probabilidad el cliente esta cometiendo fraude. Cada cliente posee un medidor en la acometida que permite registrar el consumo de su hogar/comercio y esa energía es facturada de forma mensual. Tradicionalmente, la información contenida en los consumos ha sido la clave para determinar a qué clientes se les debe llevar a cabo una inspección. Estos métodos consistían en detectar si los consumos estaban por debajo de cierto umbral o si tenían muy poca varianza. En 2017, UTE comenzó su transición hacia AMI, la cual lleva varios años y planea tener el 100 % de sus clientes con medidores inteligentes. A Febrero de 2024, 1.323.000 de clientes poseen medidor inteligente, sobre un total de 1.500.000 de clientes. Un medidor inteligente, a diferencia de uno tradicional (electrónico o electromecánico), tiene la capacidad de registrar mediciones quinceminutales. El disponer de una medición más precisa y frecuente del consumo de energía, mejora las capacidades de monitoreo y permite detectar patrones que posiblemente antes pasaban inadvertidos. En contrapartida, el volumen de datos a manejar introduce una nueva complejidad a la hora de analizar, manejar y preparar los datos, así como también exige la utilización de algoritmos más potentes como las redes neuronales profundas. Estas redes son capaces de descubrir patrones y relaciones en los datos que son difíciles de capturar mediante métodos convencionales.

Desde el año 2000, la detección automática de NTL ha sido un campo con una gran actividad académica [40, 54]. En la literatura se encuentra que entre 2008 y 2010, se empieza a investigar la aplicación de aprendizaje automático para el análisis de los datos de clientes e identificación de posibles fraudes [41, 44, 45, 51]. Durante varios años, el Instituto de Ingeniería Eléctrica de la Universidad de la República (IIE) y el Departamento de Recuperación de Energía (DRE) de la empresa UTE han establecido un sólido vínculo institucional, trabajando en conjunto para abordar el problema de la detección automática de fraudes. Se comenzó utilizando un enfoque clásico de reconocimiento de patrones sobre consumos mensuales [13, 14, 31], y con la incorporación de AMI, evolucionaron a utilizar algoritmos de aprendizaje profundo que se alimentan de consumos quinceminutales [38]. En conjunto diseñaron una novedosa arquitectura multiresolución capaz de entrenar y predecir curvas de consumo de entrada en resoluciones de un mes o quinceminutal. Este enfoque logra sacar la máxima ventaja de los

1.2. Descripción de objetivos y alcance

datos de medidores inteligentes y de los consumos históricos tomados por medidores convencionales (electrónico o electromecánico). Los experimentos previos a esta tesis demuestran que la integración de datos disponibles en diferentes resoluciones es beneficiosa, lo cual resulta altamente relevante en el proceso de transición de medidores convencionales a inteligentes. Además entre sus conclusiones, se destaca que tres meses de consumos quinceminutales tienen mayor potencial de predicción que tres años de consumos mensuales. Por otro lado, también se crearon arquitecturas que consideran las alarmas que registran los medidores (tales como: corte y restablecimiento de energía, inicio y fin de corriente diferencial), lo cual sigue mejorando el desempeño de la detección de irregularidades [4].

Los resultados positivos obtenidos al trabajar con series de consumo en alta resolución motivan e impulsan a continuar explorando esta área. Además, una de las variables cruciales que se incorpora en este estudio es la temperatura ambiente. La inclusión de esta serie temporal es de particular interés debido al vínculo intrínseco entre la temperatura ambiente y el consumo energético.

Por lo tanto, en esta tesis se busca seguir mejorando la detección automática de fraude a través de arquitecturas que soporten series temporales multivariadas. Las series temporales son un conjunto de observaciones de determinado fenómeno durante cierto tiempo. En las series de tiempo univariadas, los datos se recopilan de una fuente, por ende, cada observación es un escalar único. Mientras que en las series de tiempo multivariadas, las observaciones se registran desde múltiples fuentes simultáneamente y cada punto de datos es un vector multidimensional. Éstas últimas han recibido mucha atención en los últimos años y esto se debe a que cuando se estudia un fenómeno, en general no solo depende de sí mismo, sino que de varios factores. Por lo tanto, en la clasificación de series multivariadas, existen varios desafíos a resolver tales como: identificar cuál es el subconjunto de variables que permiten realizar la clasificación; identificar aquellos intervalos de tiempo en los cuales las variables ayudan a discriminar entre las clases a clasificar; determinar cómo tratar las series cuando poseen distintos largos/resoluciones. En este ámbito, el análisis del comportamiento de las series de consumo de energía activa, energía reactiva y temperatura ambiente puede ser de utilidad para mejorar en la detección de NTL.

La motivación detrás del uso de modelos de aprendizaje profundo reside en su capacidad para aprender automáticamente a partir de los datos, prescindiendo de la necesidad de conocimiento previo del dominio, así como en su habilidad para procesar grandes volúmenes de datos. Por cada cliente se dispone de decenas de miles de datos que ofrecen información sobre su energía activa, energía reactiva y temperatura ambiente, los cuales se representan como series temporales. Aunque existen métodos más tradicionales para estudiar series temporales y extraer características propias (como por ejemplo: el promedio, la desviación, coeficientes de Fourier, entre otros), la extracción resulta sumamente compleja y exige mucho esfuerzo porque las características intrínsecas de los datos de series temporales son difíciles de capturar. En estudio previo realizado bajo el convenio entre IIE y DRE [37], se puede comprobar que da mejores resultados trabajar con series de consumos mensuales crudos, en vez de características construidas a partir de las series.

1.2. Descripción de objetivos y alcance

El objetivo general de este proyecto es evaluar diferentes arquitecturas de aprendizaje profundo y aplicarlas a los datos de UTE, con el fin de seguir mejorando el proceso de detección automática de fraude. Estas arquitecturas de aprendizaje profundo están diseñadas para el tratamiento de series temporales multivariadas. Para la realización de esta tesis se dispone de un gran conjunto de datos etiquetados que contiene 50k

Capítulo 1. Introducción

clientes inspeccionados en campo por UTE. A su vez, se dispone de series temporales de energía activa, energía reactiva y temperatura para cada cliente.

1.2.1. Preguntas de investigación

1. ¿Qué tipos de arquitecturas de redes neuronales profundas son viables para abordar este problema? ¿Existen diferencias significativas en su rendimiento?
2. Se dispone de mediciones de energías desde el momento en que queda instalado el medidor inteligente hasta la actualidad, ¿qué historial de mediciones es suficiente para clasificar un cliente como normal o fraudulento? ¿tres meses o doce meses?
3. ¿Qué sucede cuándo se incorpora una nueva serie de energía reactiva para la clasificación? ¿Aporta información y ayuda en la clasificación?
4. Dado que también se dispone de una base de temperaturas ambiente en resolución horaria para cada uno de los departamentos del país, surgen varias interrogantes: ¿Los clientes cambian su comportamiento de consumo energético junto con los cambios en la temperatura?, ¿El comportamiento es el mismo en los meses de invierno y de verano?, ¿Este comportamiento es el mismo en aquellos clientes que cometen fraude? ¿aporta información y ayuda en la clasificación el hecho de incluir la temperatura ambiente?

1.2.2. Alcance de la tesis

Esta tesis se enfoca en el uso de datos que son exclusivos de medidores inteligentes. Se utilizan datos con resolución horaria de energías activa y reactiva. Para estudiar si la temperatura aporta información valiosa, se utilizan datos de consumos mensuales de energía activa y otras características específicas del cliente tales como: potencia contratada, ubicación geográfica, entre otras. Éstas fueron utilizadas en estudios previos y permiten evaluar qué tan útil puede ser la temperatura frente a características ya conocidas. Queda por fuera del alcance de esta tesis el uso de datos referentes a microbalances de subestaciones.

1.3. Estructura de la tesis

El resto de este documento se estructura de la siguiente forma:

- Capítulo 2: Se realiza una exploración del estado del arte en clasificación de series temporales univariadas y multivariadas, y en arquitecturas de aprendizaje profundo para la detección de NTL. Se hace un análisis sobre métricas para evaluar el desempeño de los modelos en conjunto de datos desbalanceados.
- Capítulo 3: Se estudia el impacto que tiene la temperatura ambiente en el consumo eléctrico y en la detección de NTL.
- Capítulo 4: Se detallan las características de los conjuntos de datos y su preparación. Se describen técnicas de normalización de series temporales. Se define la estrategia de búsqueda de hiperparámetros a utilizar en los experimentos con modelos de aprendizaje profundo.
- Capítulo 5: Se describen los experimentos realizados sobre los distintos tipos de arquitecturas y se analizan los resultados obtenidos.
- Capítulo 6: Se resumen las principales conclusiones de la tesis y se presentan posibles líneas de trabajo a futuro.

Capítulo 2

Estado del arte en clasificación de series temporales multivariadas y detección automática de NTL

2.1. Introducción

Con el objetivo de seguir mejorando el proceso de detección de NTL y aprovechando el aumento en la cantidad de datos que generan las infraestructuras avanzadas de medición, se abre paso a explorar nuevas alternativas para tratar este problema. En este contexto, se plantean las siguientes interrogantes: ¿cómo se puede aplicar enfoques de aprendizaje profundo para la clasificación de series temporales de manera efectiva? ¿cuáles han sido los últimos avances en estas áreas?

Este capítulo busca definir conceptos básicos que ayuden a comprender y describir formalmente el problema de clasificación de series temporales haciendo uso de arquitecturas de aprendizaje profundo. Se examinan los distintos enfoques para atacar el problema. Adicionalmente, se incluye una revisión de trabajos previos relacionados con la detección de NTL para comprender el estado actual de la investigación en este campo. Se describen los tipos de redes neuronales más utilizados en este tipo de problemas, detallando sus características y particularidades. Por último, se describen las métricas más adecuadas para la evaluación de los modelos en conjuntos de datos desbalanceados.

2.2. Clasificación de series temporales univariadas y multivariadas

Las series temporales son un conjunto de observaciones de determinado fenómeno durante cierto tiempo. En las series de tiempo univariadas, los datos provienen de una sola fuente, lo que significa que cada observación consiste en un solo valor. Por otro lado, en las series de tiempo multivariadas, las observaciones se recopilan de múltiples fuentes simultáneamente, lo que resulta en cada punto de datos representado como un vector multidimensional. En los últimos años, las series temporales multivariadas han ganado considerable atención. Esto se debe a que, al estudiar un fenómeno, generalmente está influenciado por múltiples factores, no solo por sí mismo. La naturaleza

Capítulo 2. Estado del arte en clasificación de series temporales multivariadas y detección automática de NTL

secuencial de los datos temporales presenta desafíos únicos que requieren enfoques especializados para comprender y extraer patrones significativos.

La clasificación de series temporales univariadas y multivariadas implica la tarea de asignar una etiqueta a una secuencia de datos en función de sus características y propiedades. En el caso univariado, el análisis se enfoca en una única serie temporal, mientras que en el enfoque multivariado se aborda la interacción y dependencia entre múltiples series temporales.

Hassan et al. [27] resume los avances que han habido con respecto a la clasificación de series temporales, ya sean univariadas o multivariadas usando modelos de aprendizaje profundo. A su vez se encargan de probar una gran variedad de arquitecturas sobre distintos tipos de conjuntos de datos obtenidos del archivo universitario perteneciente a la Universidad de California, Riverside [11]. Si bien el artículo fue presentado en el 2019, las arquitecturas que describe siguen siendo vigentes al día de hoy.

Formalmente, el problema se puede describir a través de estas definiciones:

- **Definición 1:** Una serie temporal univariada $X = [x_1, x_2, \dots, x_T]$ es un conjunto ordenado de valores reales de largo T .
- **Definición 2:** Una serie temporal multivariada $X = [X^1, X^2, \dots, X^M]$ es un conjunto compuesto de M series univariadas diferentes donde cada $X^i \in \mathbb{R}^T$
- **Definición 3:** Un conjunto de datos $D = (X_1, Y_1), (X_2, Y_2), \dots, (X_N, Y_N)$ es una colección de pares (X_i, Y_i) donde X_i podría ser una serie temporal univariada o multivariada, con Y_i como vector de etiqueta correspondiente en formato *one-hot*. Para un conjunto de datos que tiene K clases, el vector *one-hot* Y_i tiene largo K y cada elemento $j \in [1, K]$ es igual a 1 si la clase de X_i es j y 0 en caso contrario.

Dadas estas definiciones la clasificación de series temporales consiste en entrenar un clasificador en un conjunto de datos D con el objetivo de que aprenda ese mapeo entre X e Y . En la Figura 2.1, se puede observar un *framework* unificado de aprendizaje profundo para la clasificación de series temporales. La red neuronal lleva a cabo la clasificación mediante la composición de funciones.

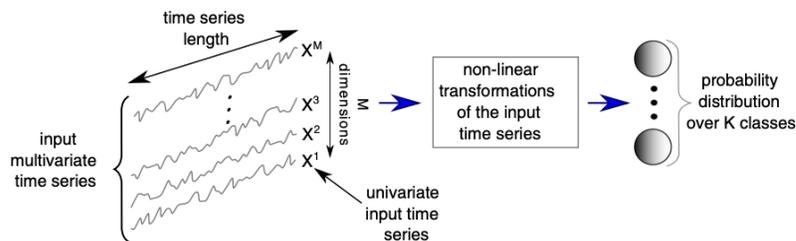


Figura 2.1: *Framework* unificado de aprendizaje profundo para la clasificación de series temporales - Imagen tomada del artículo citado en [27].

Por lo tanto, dada una entrada x , una red neuronal realiza los siguientes cálculos para predecir la clase (ver Fórmula 2.1):

$$f_L(W_L, x) = f_{L-1}(W_{L-1}, f_{L-2}(W_{L-2}, \dots, f_1(W_1, x))) \quad (2.1)$$

donde f_i corresponde a la función no lineal aplicada en la capa i y W_i corresponde a los pesos vinculados a la capa i .

2.3. Enfoques para la clasificación de series

Durante el entrenamiento, a la red se la alimenta de un conjunto etiquetado D . Primero, los pesos de las capas son inicializados (por ejemplo de forma aleatoria). Después de la inicialización de los pesos, la primera capa de la red neuronal toma esa entrada y aplica una transformación lineal entre la entrada y los pesos de su capa (hace una suma ponderada). Luego esa suma es la entrada de una función de activación y esta transformación no es lineal. La salida de esta función se pasa a la siguiente capa. Las siguientes capas harán lo mismo (esto es lo que se conoce como *feed-forward propagation*), hasta llegar a la última capa, la cual dará una distribución de probabilidad sobre K clases a clasificar y aquella clase para la cual la probabilidad sea máxima, será la clase que le asigne el clasificador a la instancia x de entrada.

Para que el modelo aprenda durante el entrenamiento es necesario definir una función de *loss*. La misma sirve para medir que tan bien se está ajustando el modelo a los datos de entrenamiento y en base a esta función y su gradiente es que se aprenden los valores de los pesos que debe tener la red para que el *loss* sea mínimo. El descenso por gradiente estocástico es un algoritmo de optimización que permite minimizar la función de *loss* respecto a un conjunto de entrenamiento. El algoritmo de *backpropagation* permite calcular de forma eficiente las derivadas de una red neuronal, para poder ajustar el error de cada uno de los pesos en las diferentes capas. Es importante que las funciones de activación elegidas sean diferenciables y continuas sobre el dominio. Esto es porque típicamente se utiliza *backpropagation* para aprender los pesos y este algoritmo hace uso de las derivadas. Durante la fase de test, el clasificador es probado en datos que nunca había visto antes y será la referencia para medir el desempeño del modelo y su capacidad de generalización. Para hacer esta medición, es importante elegir una métrica adecuada.

La gran ventaja de usar modelos de aprendizaje profundo en estos escenarios es que los mismos son capaces de aprender automáticamente a partir de los datos, sin conocimiento previo del dominio. En contraste, existen otros métodos más tradicionales que buscan estudiar más las series y sacar características propias (como por ejemplo: el promedio, la desviación, coeficientes de Fourier, entre otros). Pero, esta extracción es muy compleja, exige mucho esfuerzo y conocimiento experto del dominio.

2.3. Enfoques para la clasificación de series

Los enfoques de aprendizaje profundo para la clasificación de series temporales se pueden dividir en dos grandes categorías: los modelos generativos y discriminativos. En el artículo [27] se define una taxonomía que resume los distintos enfoques y se muestran en la Figura 2.2.

2.3.1. Modelos generativos

En el universo de los modelos generativos, el objetivo es aprender una representación de la serie más que la clasificación en sí y en esta etapa las redes se entrenan de forma no supervisada. Luego, con esa representación, se alimenta otro modelo encargado de predecir la clase bajo entrenamiento supervisado. Además, este grupo se subdivide en dos: *auto encoders* y *echo state networks* (ESNs).

Dentro del grupo de *auto encoders* se encuentra el trabajo de Rajan et al. [50] donde en una primera etapa utilizan redes recurrentes para aprender la representación latente de series temporales vinculadas a electrocardiogramas y luego en la segunda etapa utilizan un ensamblado de árboles de decisión (*random forest*) como clasificador para hacer efectivamente la predicción de la clase.

Por otro lado, en el grupo de las ESNs, las mismas fueron creadas por Jaeger y Haas [28] para la predicción de series temporales en canales de comunicación inalámbricos.

Capítulo 2. Estado del arte en clasificación de series temporales multivariadas y detección automática de NTL

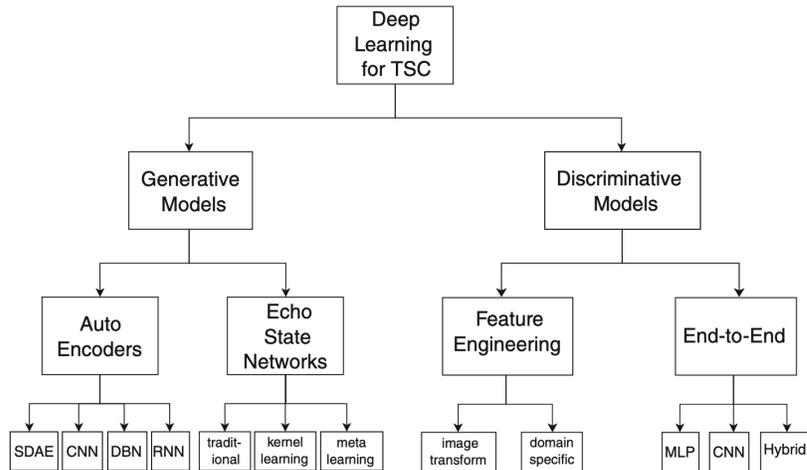


Figura 2.2: Taxonomía con los diferentes enfoques para la clasificación de series temporales - Imagen tomada del artículo citado en [27].

Las ESNs son un tipo de redes recurrentes que se conforman de tres componentes: una capa de entrada, un depósito llamado *reservoir* y una capa de salida. Los pesos vinculados al depósito tienen la particularidad de que no son entrenables (se inicializan de forma aleatoria y quedan fijos). Una aplicación de esta arquitectura en clasificación de series temporales se encuentra en el trabajo [9].

2.3.2. Modelos discriminativos

En el universo de los modelos discriminativos, los modelos buscan aprender el mapeo de la serie temporal y la distribución de probabilidad de pertenecer a una clase. A su vez, la categoría se divide en dos grupos: ingeniería de características (*feature engineering*) y de extremo a extremo (*end-to-end*).

Dentro del grupo de *feature engineering*, la técnica más común es la de codificar series temporales en imágenes, usando métodos como Gramian Angular Field (GAF) [59]. GAF es un método de transformación de datos que convierte series temporales de una dimensión en dos dimensiones, preservando las características originales de los datos y agregando correlaciones temporales. Pero también se encuentran otros métodos de extracción de características que son manuales y requieren de expertos en el dominio. Estas características son las que luego serán la entrada de un clasificador. Por ejemplo, en el trabajo de Uemura et al. [56], se extrajeron varias características (como la velocidad) de los datos del sensor colocado en la mano de un cirujano para determinar el nivel de habilidad durante el entrenamiento quirúrgico. Estas características primero se diseñan manualmente utilizando algunos conocimientos del dominio y luego se envían a un clasificador discriminativo.

Por otra parte, en el grupo de *end-to-end*, el foco está en que el modelo sea capaz de aprender automáticamente del dominio a partir de datos crudos, sin necesidad de crear características especiales. Existe un estudio [19], que hace un gran repaso de los avances en cuanto a clasificación de series temporales vinculadas a señales bioeléctricas (tales como: electroencefalograma, electrocardiograma, entre otros) y en este campo, la interpretación humana requiere de años de formación para adquirir conocimientos especializados. Incluso con conocimiento experto, la interpretación manual sufre de

2.4. Trabajos relacionados a NTL

variabilidad entre distintos operadores. Es un proceso bastante tedioso en el que los errores humanos pueden ser causados por la fatiga. En cambio, aplicando técnicas de aprendizaje profundo, siendo las redes convolucionales las más populares en esta área, estos tipos de sesgos se eliminan y la clasificación puede llevarse a cabo de manera más rápida y económica con respecto a métodos tradicionales.

2.3.3. Elección de enfoque

La elección entre un enfoque generativo y discriminativo depende de diversos factores, como la complejidad de los datos, el tamaño del conjunto de datos y los recursos disponibles. Los enfoques generativos pueden ser útiles cuando se necesita comprender la estructura subyacente de los datos y generar muestras sintéticas, mientras que los enfoques discriminativos son eficaces cuando el objetivo principal es clasificar directamente las instancias en clases específicas. Atacar un problema con un enfoque generativo es más costoso ya que implica el entrenamiento de dos modelos.

En esta tesis se trabaja con un enfoque discriminativo debido a la disponibilidad de un amplio conjunto de datos, lo que hace innecesaria la generación de muestras sintéticas. De lo contrario, evaluar la calidad de las muestras generadas por un modelo generativo sería una tarea muy difícil debido a la complejidad de los datos de este escenario.

2.4. Trabajos relacionados a NTL

En la literatura se encuentran numerosas propuestas para abordar el problema de NTL. Varias de ellas surgen de un trabajo conjunto entre UTE y la Facultad de Ingeniería que durante 10 años han trabajado en equipo, generando publicaciones de artículos científicos en conferencias y revistas reconocidas internacionalmente en el área. Estas investigaciones han contribuido a abordar el problema de detección de NTL en diferentes etapas, desde antes de la migración hacia AMI hasta durante la implementación. En esta revisión solamente se presentan algunos de estos trabajos.

En primer lugar, se destaca la investigación [37] que analiza dos enfoques diferentes para la detección de NTL: construcción manual de características frente a arquitecturas de aprendizaje profundo. El estudio se realizó sobre una base de datos de 311k clientes de UTE, siendo una de las bases con datos reales (no sintéticos) más grandes etiquetadas hasta el momento. A partir de esta gran base, se crean dos conjuntos de datos: uno que contiene los últimos tres años de consumos mensuales de energía (llamados datos crudos); y otro que contiene características construidas manualmente a partir de esos consumos mensuales. Los datos se encuentran etiquetados por técnicos de UTE, donde un 11.8% del total de la base corresponde a casos de fraude. Se prueban una gran variedad de arquitecturas que son populares en este campo, algunas de ellas de aprendizaje automático (ejemplos: Random Forest, Extreme Gradient Boosting) y otras de aprendizaje profundo (ejemplos: CNN, LSTM). Del estudio se desprenden varios descubrimientos interesantes:

- Tienen más potencial los datos crudos que aquellas características construidas manualmente.
- Cuando se complementan los datos crudos con características contractuales (como potencia contratada, cantidad de días de mora, cantidad de irregularidades previas, entre otros) y de ubicación geográfica, contribuye a una mejora en la clasificación y aumenta el desempeño del modelo.
- Cuando el tamaño del conjunto de entrenamiento aumenta por encima de 80k, la ganancia a nivel de desempeño del modelo se vuelve despreciable.

Capítulo 2. Estado del arte en clasificación de series temporales multivariadas y detección automática de NTL

- Muestran cómo a través de la técnica de ataques adversarios se puede comprender cuáles son los patrones que tienen las curvas de consumo que caracterizan tanto a los clientes fraudulentos como a los normales. Los clientes fraudulentos suelen tener una caída del consumo o una variabilidad de consumo atípica, mientras que los clientes normales suelen tener un consumo que oscila y una suave variabilidad estacional.

En segundo lugar, se quiere resaltar el trabajo [38] el cual ataca el problema de NTL durante el despliegue masivo de medidores. El mismo presentó un gran desafío y fue un trabajo innovador, dado que en la literatura no existían trabajos previos que presenten cómo atacar el problema en un escenario donde conviven datos con diferentes tipos de resoluciones (clientes con medidores convencionales reportan mediciones mensuales, mientras que clientes con medidores inteligentes reportan mediciones quinceminutales). Desde el momento en que se instala el primer medidor inteligente en una red, comienzan a convivir datos de diferente resolución temporal. Los datos de medición inteligente comienzan en fechas diferentes para cada cliente, generando bases de datos de series temporales con largos variables. Por esta razón, el trabajo propone una arquitectura de aprendizaje profundo capaz de soportar múltiples resoluciones para sobrellevar la transición hacia la medición inteligente, el cual es un proceso que lleva varios años. El estudio utilizó dos conjuntos de datos, uno con fraude real y otro con fraude simulado. El conjunto de datos real posee 10596 clientes con medidor inteligente instalado. Los mismos fueron inspeccionados por técnicos de UTE, por lo que disponen de una etiqueta indicando si son fraudulentos. Para cada cliente se poseen 90 días de medidas quinceminutales previas a la inspección y el histórico de consumo mensual en los tres años previos a la inspección. En total se poseen 772 casos de fraude que representan un 7.3% del total. Y por otro lado, se creó una base con fraudes típicos simulados sobre consumo reales publicados por la Comisión Reguladora de Energía de Irlanda [46] que corresponden a 6435 clientes durante un periodo de 17 meses.

Como resultado, se demuestra que la arquitectura propuesta para múltiple resolución tiene un desempeño superior que aquellas arquitecturas que solamente soportan una sola resolución, tanto en la base de datos real como en la sintética. En la Figura 2.3 se puede observar cómo es la arquitectura diseñada e implementada. La misma se compone de convoluciones en 1D y 2D.

Por otra parte, merece mención el trabajo de [24], que presenta una arquitectura innovadora para abordar las NTL mediante el análisis de múltiples series temporales, aplicada con éxito a datos reales de China. El estudio se realiza sobre una base con 311k clientes, de los cuales solo el 1.49% está etiquetado como fraudulento. En este trabajo, se modelan los comportamientos de los clientes en base a tres niveles diferenciados de información:

- Registros de consumo diario de energía (nivel micro) vinculado a 311k clientes. Para cada cliente se tiene tres registros: uno que representa el total de energía consumida en el día, otro que indica el consumo de energía en horas pico del día y por último otro que marca el consumo de energía fuera de horas pico del día.
- Registros de NTL (nivel meso) para 3.9k áreas las cuales cubren todos los clientes de la base.
- Registros de temperatura (nivel macro) para las 11 ciudades involucradas con los clientes. Para cada ciudad, se tiene el mínimo y máximo de temperatura por día.

En la Figura 2.4 se muestra el modelo propuesto donde en la primera capa se tiene tres redes LSTM, una para cada serie temporal de entrada. En el primer nivel, el objetivo es modelar la influencia de factores de nivel macro y meso en el comportamiento

2.4. Trabajos relacionados a NTL

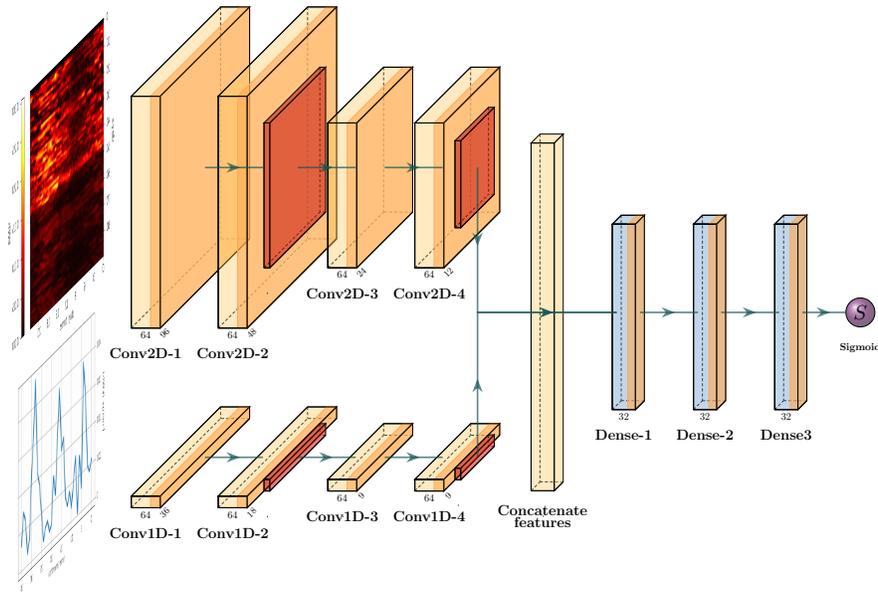


Figura 2.3: Arquitectura de multiresolución para detección de NTL. La parte superior es la red vinculada a los datos de alta resolución (medidas quinceminutales), la misma posee cuatro capas de convolución 2D de 64 filtros. En color naranja se resalta la aplicación de la función de activación ReLU y en rojo las capas de pooling. La parte inferior de la red es la vinculada a los datos de baja resolución (medidas mensuales), la misma posee cuatro capas de convolución 1D de 64 filtros. Las salidas de ambas redes de convolución (de 1D y 2D) se concatenan y son la entrada a una serie de capas de redes neuronales totalmente conectadas con activación ReLU. La salida queda determinada por una función sigmoide. Imagen obtenida del artículo *Fraud detection on power grids while transitioning to smart meters by leveraging multi-resolution consumption data*.

de los usuarios. En el segundo nivel, se fusionan las salidas de las redes de temperatura y clientes, y las de clientes y áreas, y se tienen dos redes LSTM para procesar cada una de las nuevas series. La salida de ambas redes se vuelve a fusionar y es la entrada a otra red LSTM, seguida de una capa densa. En la capa de salida se calcula la probabilidad de que sea un cliente fraudulento a través de la función *softmax*.

En su análisis, descubren que es probable que los ladrones de electricidad consuman mucha más energía eléctrica que los usuarios normales, especialmente bajo temperaturas extremadamente altas o bajas. A pesar de que el comportamiento de robo reduciría el valor registrado del consumo de energía eléctrica, la explicación encontrada a este fenómeno proviene de la idea de que las personas suelen adoptar comportamientos riesgosos sólo cuando esperan un beneficio mayor. En cuanto a este escenario, se espera que los usuarios cuyo consumo de electricidad es alto tengan una motivación mucho más fuerte para robar energía, ya que reducirían sus costos significativamente al involucrarse en un comportamiento de robo. Por el contrario, si un usuario tiene un consumo muy bajo de electricidad, no es necesario que realice tales operaciones ilegales, ya que este comportamiento en caso de ser descubierto implicaría una sanción económica elevada.

Este trabajo marcó un hito para la empresa distribuidora ya que con algoritmos previos no habían tenido éxito y no resultaban eficientes en la práctica. Como ejemplo,

Capítulo 2. Estado del arte en clasificación de series temporales multivariadas y detección automática de NTL

se pone que en 2018, ninguno de los modelos que tenía fue capaz de detectar fraude. Sin embargo, cuando se puso en práctica el nuevo modelo descrito logró una precisión del 15%.

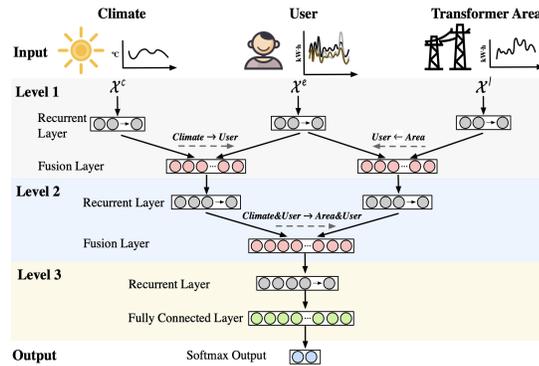


Figura 2.4: Arquitectura multiresolución para detección de NTL. La arquitectura recibe como entrada datos de diferentes fuentes: clima, pérdida en el área y consumos de clientes. La misma se organiza en tres niveles, en cada uno de ellos se ingresan diferentes secuencias en diferentes capas recurrentes, cuyas representaciones latentes se fusionan en pares. Las líneas discontinuas indican la dirección de la fusión. La capa de salida calcula la probabilidad de fraude a través de la función softmax. Imagen obtenida del artículo *Understanding electricity-theft behavior via multi-source data*.

2.5. Detección de NTL usando aprendizaje profundo

2.5.1. Redes neuronales convolucionales CNN

Las redes neuronales convolucionales (CNN) tienen una rica historia que abarca varias décadas, con desarrollos significativos en su arquitectura y aplicaciones. Los conceptos fundamentales que luego conduciría al éxito a las CNN surgieron en la década de 1960 con el trabajo de Hubel y Wiesel sobre el sistema visual de los gatos [25]. Fue en ese entonces cuando la idea del aprendizaje de características jerárquicas comenzó a tomar forma. Descubrieron que ciertas neuronas en la corteza visual responden a patrones y orientaciones específicas. En la década de 1980, surge la primera arquitectura de red neuronal, creada por Kunihiko Fukushima llamada Neocognitron [20], utilizada para el reconocimiento de caracteres escritos a mano. Esto sentó las bases para las redes convolucionales. Yann LeCun y otros ampliaron estas ideas con su arquitectura LeNet-5 a principios de la década de 1990, que se utilizó para el reconocimiento de dígitos [33]. Durante la década del 2000, el éxito fue limitado. A pesar de los grandes avances que se había tenido, las CNN enfrentaron limitaciones vinculadas a las restricciones computacionales y la falta de grandes conjuntos de datos etiquetados, por lo que su uso práctico fue reducido durante ese periodo. Las CNN resurgieron alrededor de 2012 con *ImageNet Large Scale Visual Recognition Challenge* (ILSVRC) [53]. AlexNet [32], desarrollado por Alex Krizhevsky, Ilya Sutskever y Geoffrey Hinton, ganó la competencia con su arquitectura profunda y aceleración de GPU, lo que avanzó significativamente en el campo del aprendizaje profundo. Esto marcó el comienzo de que las CNN se volvieran ampliamente útiles en varias tareas de visión por computadora, tales como: clasificación de imágenes, la detección de objetos y la segmentación de imágenes.

2.5. Detección de NTL usando aprendizaje profundo

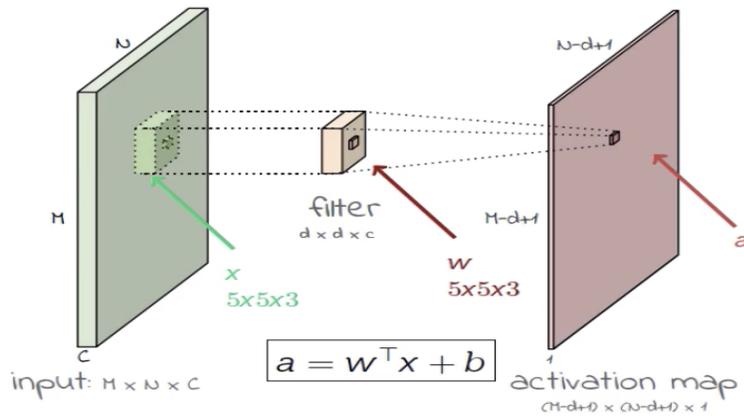


Figura 2.5: Capa de convolución, imagen tomada de las notas del curso *Stanford CS class CS231n Convolutional Neural Networks for Visual Recognition*.

Este tipo de redes fueron diseñadas principalmente para procesar y analizar datos en grillas. La capa de convolución es el bloque de construcción central de una red convolucional, encargada de realizar la mayor parte del trabajo pesado computacional, ver Figura 2.5. En esta capa, un conjunto de pequeños filtros (también llamados núcleos) se deslizan sobre los datos de entrada. Los filtros aplican operaciones de multiplicación y suma de elementos, capturando efectivamente patrones o características locales. Estos patrones pueden incluir bordes, esquinas, texturas y estructuras más complejas.

Se caracterizan por su estructura jerárquica donde las primeras capas extraen información de bajo nivel como bordes y a medida que se va profundizando en la red se van ensamblando estas características simples en patrones u objetos más complejos de alto nivel. Otra gran característica que poseen es el uso compartido de parámetros, lo cual baja el costo de entrenamiento e impone invarianza. Cada filtro en una capa convolucional se aplica a toda la entrada, independientemente de su posición. Esto brinda invarianza ya que permite que la red generalice características en diferentes ubicaciones en el espacio de entrada. Entre la entrada y el filtro se realiza una serie de productos internos, generando como salida un mapa de activación. Cada uno de los filtros representan patrones y sus salidas son mapas de activación con respecto a esos patrones, ver Figura 2.6. Al observar los mapas de activación, es posible identificar qué aspectos está aprendiendo la red y qué características están provocando su activación. Las funciones de activación introducen la no linealidad en la arquitectura de CNN, y esto le permite aprender relaciones complejas entre características, por ejemplo, *rectified linear unit* (ReLU) es una opción popular de función de activación. La misma reemplaza los valores negativos con cero y mantiene los valores positivos sin cambios.

Las capas de convolución van acompañadas de capas de *pooling* (van intercaladas). Estas capas realizan una operación de submuestreo que reduce la dimensión espacial de la información, disminuyendo la cantidad de información a procesar y buscando preservar la información importante. Opera en cada mapa de activación por separado y también impone invarianza (por ejemplo en *max pooling* el máximo es invariable a la posición donde esté).

Otra capa muy utilizada es la llamada *dropout*, la cual consiste en apagar aleatoriamente un porcentaje de las neuronas en una capa durante el entrenamiento. Este procedimiento impide que la red neuronal dependa demasiado de ciertas característi-

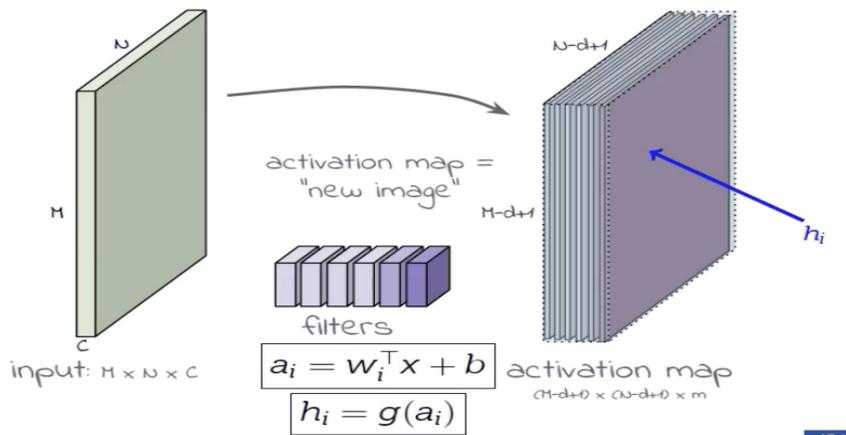


Figura 2.6: Capa de convolución con múltiples filtros y mapas de activación, imagen tomada de las notas del curso *Stanford CS class CS231n Convolutional Neural Networks for Visual Recognition*.

cas o conexiones particulares y en su lugar busca que tenga redundancia. Esto ayuda a mejorar la generalización del modelo, ya que obliga a que cada neurona contribuya de manera más robusta e independiente a la predicción.

Si bien las CNN se desarrollaron originalmente para el procesamiento de imágenes, su arquitectura se ha adaptado a otros dominios, como el texto y los datos de series temporales. La clave está en tratar los intervalos de tiempo como dimensiones espaciales. Por sus características y por su uso en trabajos anteriores vinculados a NTL, resulta atractivo utilizar esta arquitectura con series temporales multivariadas.

2.5.2. Redes neuronales recurrentes: RNN y LSTM

En 1986, Rumelhart et al. [52] introducen las redes recurrentes (RNN) al mundo de las redes neuronales. Estas redes fueron diseñadas para ser flexibles al ser capaces de procesar secuencias de diferentes largos y para mantener la memoria de entradas pasadas. Fue una salida al problema de las redes neuronales normales que tienen dificultades para lidiar con entradas y salidas de tamaños variables (como lo son escenarios de: traducción automática, reconocimiento de voz, clasificación de texto, etc.). La gran diferencia de este tipo de redes con respecto a las tradicionales es que tienen bucles de retroalimentación, que buscan incorporar información del pasado en la predicción de la red. Sin embargo, las primeras RNN enfrentaron dificultades para aprender dependencias a largo plazo debido al famoso problema del desvanecimiento del gradiente, conocido como *Vanilla RNN*. Este problema se manifiesta durante el proceso de *back-propagation*, cuando los gradientes tienden a cero a medida que se propagan hacia atrás. Como resultado, las actualizaciones de los pesos en las capas más profundas de la red se vuelven mínimas, lo que conduce a que estas capas reciban poca o ninguna información sobre cómo ajustarse para mejorar el rendimiento de la red.

Luego, surgen las redes *Long Short-Term Memory* (LSTM) [23] introducidas por Sepp Hochreiter y Jürgen Schmidhuber en 1997, que son el tipo de RNN más popular. Las mismas han sido diseñadas para abordar el problema del desvanecimiento del gradiente y capturar dependencias de largo plazo en datos secuenciales.

El foco principal de las RNN es lograr recordar la mayor cantidad de información importante posible del pasado, para poder pronosticar el futuro con mayor precisión.

2.5. Detección de NTL usando aprendizaje profundo

La unidad básica de una arquitectura LSTM tiene como componentes claves:

- Memoria a largo plazo representada por la celda *cell state*.
- Memoria a corto plazo representada por la celda *hidden state*.
- Funciones de activación sigmoide y tangente hiperbólica que van a determinar qué porcentaje de información considerar en las distintas etapas de la red.

La interacción entre las memorias de corto y largo plazo se puede apreciar en la Figura 2.7.

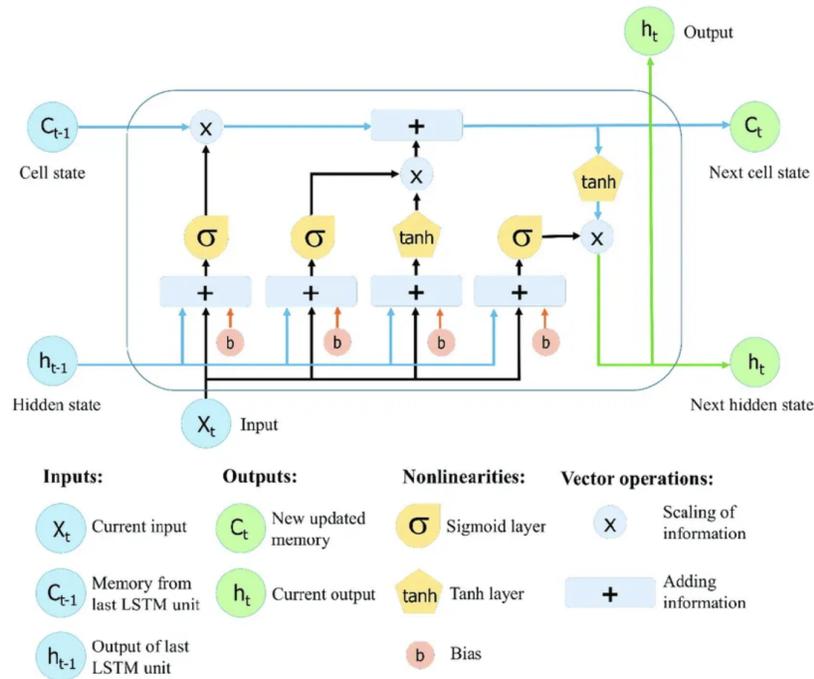


Figura 2.7: Unidad básica de una arquitectura LSTM, imagen obtenida del artículo *Application of Long Short-Term Memory (LSTM) Neural Network for Flood Forecasting*.

Para poder entender cómo funciona, las operaciones se dividen en tres etapas:

1. Compuerta de olvido: la celda de entrada en combinación con la celda de estado oculto determinan qué porcentaje de la memoria a largo plazo se desea recordar.
2. Compuerta de entrada: se divide en dos grandes operaciones que combinan la celda de estado oculto con la entrada pero para diferentes fines. Aquella que tiene la función de activación hiperbólica busca crear una potencial memoria a largo plazo y la que tiene la función sigmoide busca definir qué porcentaje de esa memoria potencial agregar a la memoria a largo plazo. Su función en conjunto es actualizar la memoria a largo plazo.
3. Compuerta de salida: es la etapa final, se actualiza la memoria a corto plazo y se divide en dos cálculos. Aquella que tiene la función de activación hiperbólica busca crear una nueva memoria a corto plazo y la que tiene la función sigmoide busca definir qué porcentaje de esa nueva memoria agregar a la memoria a corto plazo.

Capítulo 2. Estado del arte en clasificación de series temporales multivariadas y detección automática de NTL

El motivo de probar esta arquitectura con series temporales multivariadas es porque los datos de series temporales se caracterizan por su naturaleza secuencial, lo que los hace perfectos para las capacidades de las redes LSTM. Al aprender el comportamiento normal de una secuencia, esta arquitectura es capaz de identificar desviaciones, señalando posibles anomalías o valores atípicos.

2.5.3. Redes residuales ResNet

Las redes residuales (ResNet) fueron presentadas en 2015 por Kaiming He, Xiangyu Zhang, Shaoqing Ren y Jian Sun en [22]. Esta arquitectura de red neuronal convolucional (CNN) fue diseñada específicamente para manejar cientos o incluso miles de capas convolucionales. Las mismas representaron un avance fundamental en las redes neuronales profundas. Antes de que éstas existieran, las redes neuronales profundas enfrentaban desafíos a medida que se hacían más profundas y su rendimiento quedaba limitado. Cuanto más profunda era la red, más aumentaba el error de entrenamiento y en consecuencia también aumentaba el error de testeo. El problema detrás era el desvanecimiento o explotación del gradiente, donde los gradientes disminuyen o aumentan exponencialmente a medida que se propagan hacia atrás a través de numerosas capas.

La principal innovación fue la introducción de bloques residuales (ver Figura 2.8). Para ello, se utiliza la técnica de saltar conexiones, en inglés *skip connection* o *shortcut connection*, lo cual hace que las redes aprendan los cambios incrementales en los datos. En este caso, la *shortcut connection* simplemente consiste en repetir la entrada y ese valor se suma con la salida de la capa anterior, para luego aplicarle la función de activación ReLU. Estas conexiones no agregan ningún parámetro extra ni complejidad computacional. Los creadores de esta red descubrieron que apilando estos bloques residuales conjuntos era la clave para entrenar redes neuronales más profundas sin sufrir el problema del desvanecimiento del gradiente.

En 2015, ResNet mostró resultados notables ILSVRC. A pesar de tener 152 capas, ResNet logró una mayor precisión que las arquitecturas menos profundas. Esto demostró que profundizar en las redes no solo era factible, sino también beneficioso. ResNet trascendió rápidamente el reconocimiento de imágenes y se aplicó a varios dominios, incluida la detección de objetos, la segmentación de imágenes y el procesamiento del lenguaje natural (NLP).

En resumen, este tipo de arquitectura marcó un hito transformador en la historia del aprendizaje profundo. Su introducción abordó los desafíos en el entrenamiento de

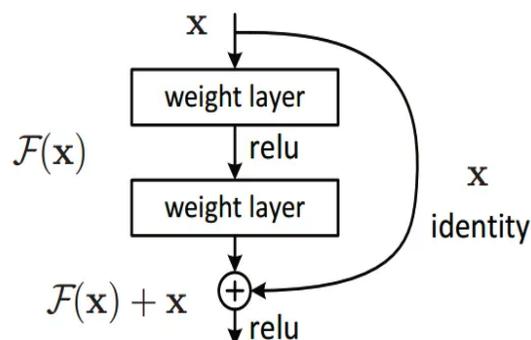


Figura 2.8: Bloque residual. Imagen obtenida del artículo *Deep Residual Learning for Image Recognition*.

2.6. Métricas para la evaluación de los modelos

redes profundas, lo que permitió la creación de arquitecturas significativamente más profundas que logran un rendimiento de vanguardia en varios dominios. El desarrollo de ResNet allanó el camino para avances que se extienden más allá del reconocimiento de imágenes. En lo que refiere a clasificación de series temporales, ha demostrado un destacado desempeño frente a otros tipos de arquitecturas en diversos conjuntos de datos [27] y por esa razón es elegida para abordar el problema de detección de NTL.

2.6. Métricas para la evaluación de los modelos

Elegir una métrica de evaluación adecuada es un aspecto fundamental para evaluar el rendimiento y la eficacia de un modelo. La elección de la métrica no solo influye en qué tan bien se entiende el rendimiento del modelo, sino también en las decisiones que se toman en función de sus resultados. Una buena métrica debe alinearse con los objetivos del problema en cuestión. Diferentes tareas requieren diferentes métricas. Por ejemplo, la exactitud (*accuracy*) puede ser adecuada para la clasificación de imágenes, pero para conjuntos de datos desbalanceados, puede ser una métrica traicionera, ya que el modelo podría lograr una alta exactitud simplemente prediciendo la clase mayoritaria.

A su vez, una métrica adecuada habilita la comparación justa y objetiva entre distintos modelos o configuraciones de hiperparámetros. Por ejemplo, mientras se entrena un modelo los ajustes se realizan en base a la métrica seleccionada, ya que el proceso de optimización estará enfocado en hacer mejorar esa métrica. Finalizado el entrenamiento, el mejor modelo también es seleccionado en base a la métrica. Por lo tanto su elección, es un paso crítico en el proceso de desarrollo de un modelo el cual asegura que los resultados sean útiles y relevantes para la aplicación en cuestión. Es por ello, que resulta esencial elegir cuidadosamente qué métrica utilizar en función del contexto y los requisitos del problema.

Dado que en este trabajo la importancia es detectar fraude, simplemente una métrica como la exactitud no sería suficiente. En este problema el modelo debe ser capaz de clasificar las series temporales en dos categorías: normal (negativo) y fraude (positivo). El conjunto de datos se encuentra altamente desbalanceado (sólo un 7% corresponde a la clase positiva). Dentro de este contexto, el objetivo es que el modelo detecte correctamente la clase positiva para decidir si se inspecciona o no a un cliente, pero sin descuidar la detección correcta de la clase negativa, ya que las inspecciones tienen su costo. Por lo tanto, en esta tarea de clasificación pueden suceder cuatro escenarios posibles:

- Inspeccionar a un cliente clasificado como fraudulento y que realmente sea fraudulento (verdadero positivo)
- Inspeccionar a un cliente clasificado como fraudulento y que en realidad sea normal (falso positivo)
- No inspeccionar a un cliente clasificado como normal cuando en realidad era fraudulento (falso negativo)
- No inspeccionar a un cliente clasificado como normal y que realmente sea normal (verdadero negativo)

Para conjuntos de datos desbalanceados, dos métricas muy utilizadas en la literatura son *precision* y *recall*. *Precision* el porcentaje de acierto para la clase positiva, por lo que es crucial cuando el costo de los falsos positivos es alto como lo es en este escenario ya que el costo de una actividad de inspección requiere: técnicos especializados, combustible, tiempo, materiales. *Recall* es el porcentaje de la clase positiva que se está detectando, su foco está en hacer hincapié en capturar todas las instancias

Capítulo 2. Estado del arte en clasificación de series temporales multivariadas y detección automática de NTL

positivas, por lo que es importante cuando el costo de los falsos negativos es alto como también lo es en este marco porque implica no detectar a un cliente que está robando energía. La relación entre *precision* y *recall* es inversamente proporcional, por lo que es importante encontrar un equilibrio entre ambas. Según los objetivos del problema, puede convenir priorizar una frente a otra o que sean igualmente importantes.

Una forma de encontrar ese equilibrio, es utilizando la medida F_β , que vincula *precision* y *recall* a través de la Fórmula 2.2:

$$F_\beta = (1 + \beta^2) \frac{\textit{precision} \times \textit{recall}}{\beta^2 \textit{precision} + \textit{recall}} \quad (2.2)$$

No es una métrica intuitiva al ver su fórmula pero ayuda a encontrar un balance entre *precision* y *recall* que sea aceptable para el problema. Si el valor de β es menor a 1 significa que se le dará más importancia a *precision*, si es mayor a 1 se le dará más importancia a *recall* y en caso de ser igual a 1, tanto la *precision* como *recall* son igualmente importantes. Para cualquier parámetro beta, el mejor valor posible para F_β es 1 y el peor es 0.

Una forma de ver este compromiso de forma gráfica es a través de la métrica *Precision Recall Area Under Curve* (PR AUC), que permite visualizar cómo varían estas métricas a medida que cambia el umbral de decisión. Es muy utilizado en situaciones donde el conjunto de datos está desequilibrado y la cantidad de casos negativos es ampliamente mayor que la cantidad de casos positivos. En este contexto, una realidad es que por un tema de capacidad operativa, no es posible inspeccionar a todos los clientes sospechosos, lo que hace necesario tener una forma de evaluar cómo sería el rendimiento si se define como objetivo detectar cierto porcentaje de los clientes fraudulentos.

También se tiene la métrica *Receiver Operating Characteristic Area Under the Curve* (ROC AUC). La misma representa la curva entre *recall* y la tasa de verdaderos positivos. Es una métrica popular porque proporciona una visión más amplia de la capacidad de un modelo para discriminar entre clases a través de diferentes valores de umbral.

Por último, se encuentra la métrica *Precision@k* que es popular cuando un modelo produce más datos de los que se pueden procesar. Esto también va de la mano con este escenario, donde idealmente sería bueno poder inspeccionar a todos los clientes pero no es posible por un tema de costos, tiempo y personal capacitado. Esta métrica lo que reporta es la precisión obtenida al clasificar como positivas un porcentaje k de la base de test. Dicho de otra forma, permite evaluar cómo sería la precisión del modelo si se define como objetivo realizar k% de inspecciones.

En resumen, estas métricas colectivamente brindan información sobre varios aspectos del rendimiento de un modelo de clasificación. En general es común considerar múltiples métricas para poder obtener una visión integral de las fortalezas y debilidades de un modelo, pero solamente se elige una sola para el entrenamiento y búsqueda de hiperparámetros. Por lo tanto, para evaluar y seleccionar modelos de manera efectiva, resulta fundamental que la métrica sea apropiada para el contexto. Una métrica bien seleccionada asegura que los resultados sean de utilidad y relevantes para el problema en cuestión. En este caso la elegida es PR AUC debido a su uso en trabajos anteriores vinculados a NTL [36–38] y porque se adecuaba muy bien a conjuntos de datos desbalanceados.

Capítulo 3

Uso de temperatura ambiente en la detección de NTL

3.1. Introducción

En el transcurso de las rutinas diarias, los cambios en la temperatura exterior suelen afectar la manera en que los ambientes cerrados se ajustan para garantizar un nivel de comodidad adecuado. Esto se manifiesta, por ejemplo, en la activación de sistemas de calefacción durante el invierno o la utilización de aires acondicionados en períodos de altas temperaturas en verano. En este contexto, la relación entre la temperatura y el consumo de energía de las personas es innegable. Estas acciones, aparentemente rutinarias, son de hecho momentos cruciales que influyen directamente en los patrones de consumo de electricidad. La energía que se usa para calentar, enfriar y mantener ambientes interiores confortables constituye una parte sustancial de nuestro consumo total de energía [8,30,35,58]. Si bien es intuitivo suponer que las temperaturas más frías conducen a mayores demandas de calefacción y las temperaturas más altas aumentan la necesidad de refrigeración, las complejidades de esta relación también depende de otros factores como nivel económico, tipo de electrodomésticos y la inercia térmica que posea la construcción del hogar, entre otros. En la literatura se encontró un estudio [30], donde usan datos de medidores inteligentes para examinar cómo impactan en el consumo energético diferentes variables tales como: el clima, las características del edificio y electrodomésticos. El estudio concluye que el consumo mínimo diario está más influenciado por el clima, ubicación geográfica y las características físicas del edificio. El consumo máximo diario está más influenciado por los electrodomésticos que consumen mucha energía y no funcionan constantemente, como por ejemplo, un termotanque. También existe otra investigación en la literatura [29], que al estudiar el efecto del clima en el consumo de energía en Irlanda, llegó a la conclusión de que la temperatura afecta indirectamente el consumo energético porque los días más agradables, las personas tienden a hacer más actividades al aire libre y por ende, pasan menos tiempo en sus hogares consumiendo energía.

Según IEA, el gas natural es actualmente la mayor fuente de energía destinada a calefacción en los edificios a nivel mundial y representa el 42% de la demanda de energía de calefacción en 2022 [5]. Pero el objetivo es ir hacia fuentes de energía que sean de bajas emisiones, como las bombas de calor. En cuanto a refrigeración, cerca de dos mil millones de unidades de aire acondicionado están operativas en todo el mundo en la actualidad [6], convirtiendo la refrigeración de espacios en uno de los

Capítulo 3. Uso de temperatura ambiente en la detección de NTL

principales factores que contribuyen tanto al aumento de la demanda de electricidad en edificaciones como al aumento de capacidad de generación para satisfacer la demanda pico de energía. Desde 2015 hasta 2022, la temperatura media anual ha comenzado a ascender, siendo estos ocho años los más cálidos registrados según todos los conjuntos de datos compilados por la Organización Meteorológica Mundial (OMM) [43]. Con varios eventos de calor extremo y temperaturas récord en todo el mundo, el 2023 fue el año más cálido desde que hay registros por un amplio margen [42]. Por lo tanto, resulta interesante analizar cómo se comporta este fenómeno en Uruguay.

En este capítulo, se explora el nexo temperatura ambiente-consumo de electricidad utilizando los datos de consumo horario de 50 mil clientes de UTE. Se crean modelos para evaluar si la relación entre temperatura ambiente-consumo energético puede aportar información relevante para la detección de NTL. Se responden las preguntas definidas en 1.2.1.4.

3.2. Técnicas para estudiar el vínculo entre series temporales

El problema puede pensarse como la correlación entre temperatura ambiente y consumo energético. Para poder analizar y comprender más este vínculo, existen diversas técnicas, entre las que se encuentran el coeficiente de correlación de Pearson y la correlación cruzada, las cuales se reseñan a continuación.

3.2.1. Coeficiente de correlación de Pearson

El coeficiente de correlación de Pearson es una medida estadística que cuantifica la fuerza y la dirección de una relación lineal entre dos variables aleatorias. Es adecuado para identificar si existe o no una relación directamente/inversamente proporcional entre las variables. El coeficiente varía entre -1 y 1. Un valor de correlación negativo, indica que cuando una aumenta, la otra tiende a disminuir y un valor de correlación positivo, indica que cuando una serie aumenta, la otra también tiende a aumentar. Un valor cercano a cero indica baja correlación, es decir que las dos series son independientes entre sí. Si bien esta técnica es muy sencilla y simple, no tiene en cuenta la estructura temporal de las series temporales. Es decir, si se tienen dos series temporales x_1, x_2, \dots, x_n e y_1, y_2, \dots, y_n , asume que cada muestra x_i con $i = 1..n$ son independientes entre sí (análogo para y_i), lo cual es una limitante al querer estudiar cómo es la sincronía entre dos series a lo largo del tiempo.

3.2.2. Correlación cruzada

La correlación cruzada es una técnica estadística utilizada para cuantificar la similitud entre dos series temporales. Esta técnica va midiendo cómo varían ambas series a lo largo del tiempo. Es muy útil cuando se desea estudiar si existe una relación líder-seguidor en la que la serie líder tiene un comportamiento que repercute sobre la serie seguidor y permite saber en qué medida se están afectando y con qué retraso. La correlación cruzada da como resultado una nueva serie que se obtiene a través de la Fórmula 3.1, obtenida del libro [49].

$$(x \otimes y)[d] = \sum_t \overline{x[t]}y[t+d] \quad (3.1)$$

siendo x e y series temporales, t representa el tiempo sobre el cual están definidas ambas series, \overline{x} indica el conjugado de x y d es un valor entero que indica el

desplazamiento.

3.2.3. Técnica seleccionada

Para aplicar la correlación cruzada, se debe definir un conjunto de desfases posibles $d = \{d_1, d_2, \dots, d_n\}$, sobre los cuales interesa estudiar cómo se comportan las series. Esto da como resultado una nueva serie z para estudiar. Por definición, se sabe que cada componente z_i está indicando cómo es la correlación entre x e y si se aplica un desfase d_i en y .

En esta tesis, se utiliza la técnica de correlación cruzada, tomando como entrada las series temporales de temperatura ambiente y consumo energético. El motivo de usar esta técnica es porque la correlación cruzada tiene en cuenta la posibilidad de que los cambios en una serie de tiempo, no afecten inmediatamente a la otra. Esto es sumamente importante dado que el impacto de la temperatura ambiente en el consumo de electricidad está condicionado por la inercia térmica de las construcciones, lo cual hace que el efecto no sea necesariamente instantáneo.

Dado este escenario, al aplicar la Fórmula 3.1 se obtiene una nueva serie z para la cual se plantea utilizar solamente el máximo de z porque sería el momento en el cual x e y se encuentran más correlacionadas, es decir, más sincronizadas. Tomando en cuenta que x representa el consumo energético e y representa la temperatura ambiente, encontrar el punto donde se encuentren más sincronizadas equivale a encontrar cómo es que impacta la temperatura ambiente en el consumo energético.

3.3. Datos utilizados

En esta sección se describen los datos utilizados, su significado, obtención, extracción y preparación.

3.3.1. Datos de medidores inteligentes

Un medidor inteligente es un dispositivo eléctrico con capacidad de registrar hasta 45 días de mediciones que suceden en un predio donde UTE suministra servicio eléctrico.

Una medida es consumo integrado cuarto-horario (quinceminutal) para una magnitud. Dentro de las magnitudes disponibles se encuentran: Energía Activa Entrante, Energía Activa Saliente, Energía Reactiva Cuadrante 1, 2, 3 y 4. En esta tesis se trabaja con Energía Activa Entrante (se resume su nombre a Energía Activa) y Energía Reactiva Cuadrante 1 (se resume su nombre a Energía Reactiva). La energía activa se mide en kWh (kilo Watts hora) y es aquella que consumen electrodomésticos tales como: termotanques, hornos eléctricos, radiadores eléctricos, entre otros. Por otro lado, la energía reactiva se mide en kVARh (kilo Voltio-Amperio reactivo hora) y es aquella necesaria para el funcionamiento de determinados equipos como por ejemplo: heladeras, lavarropas, microondas, etc. En la Figura 3.1 se puede visualizar para cuatro clientes mediciones de energías activa y reactiva durante siete días. En estos casos se puede apreciar como la energía activa y reactiva están relacionadas pero no necesariamente siempre van juntas en todo el periodo. El consumo de ambas va a depender del equipamiento del hogar. Resulta de interés analizar ambas energías porque las discrepancias entre ellas registradas pueden indicar manipulaciones en el sistema eléctrico, como la instalación de dispositivos ilegales para alterar la medición del consumo.

Un registro representa la lectura que indica un medidor para una magnitud en un determinado momento. Existe 3 tipos de registros: diarios (a una fecha y hora determinada), parcial (al momento de un reinicio manual) y cierre (al momento de

Capítulo 3. Uso de temperatura ambiente en la detección de NTL

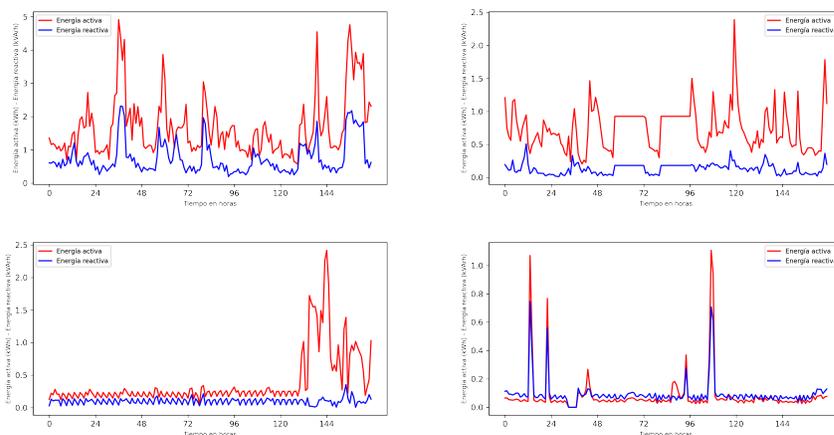


Figura 3.1: Mediciones durante 7 días de energía activa (color rojo) y energía reactiva (color azul) para diferentes clientes.

un reinicio automático). Los registros de tipo cierre son los más usados debido a que permiten saber cómo fue el consumo energético de un cliente a lo largo de un mes y son utilizados para la facturación.

3.3.2. Obtención y extracción de los datos de medidores inteligentes

Para la recolección y almacenamiento a diario de los datos provenientes de los medidores inteligentes UTE posee dos grandes sistemas. Por un lado se tiene *HeadEnd System* (HES), que es el sistema encargado de gestionar la recolección de los datos de los medidores, verificar su conectividad en la red y ofrecer un mecanismo de acceso seguro a los medidores para realizar actualizaciones de configuración y firmware. A diario, el HES ejecuta distintas tareas de recolección en base al tipo de dato que se quiere recolectar. Los datos enviados por los medidores, se almacenan en una base relacional Oracle. Por otro lado, se encuentra el sistema *Meter Data Management* (MDM), que es el responsable de centralizar y gestionar todo el histórico de datos enviados por los medidores inteligentes. Se compone de diferentes módulos encargados de realizar: extracción y carga de datos, cálculos, estimaciones, y validaciones sobre los datos. MDM recibe a diario cientos de millones de medidas y para poder procesar esos grandes volúmenes de datos utiliza tecnologías como Spark y Hadoop MapReduce. Los datos quedan almacenados en una base de datos distribuida HBase.

Para dimensionar las cantidades, se tiene:

- Se reciben a diario alrededor de 840.000.000 mediciones.
- Se tiene actualmente alrededor de 1.323.000 de clientes con medidor inteligente.
- Desde que se comenzó con la instalación de medidores inteligentes hasta ahora, se tiene un total de 46.5 T ocupados en almacenar medidas.

Un detalle a tener en cuenta es que la medida enviada por el medidor corresponde al consumo integrado quinceminutal, pero luego MDM en su módulo de cálculos realiza agregaciones para también disponer de un consumo integrado a nivel horario y a nivel diario. Además, cuando cierto medidor no envía una medida, ya sea por problemas

3.3. Datos utilizados

del medidor, problemas de la red o por problemas vinculados al HES, MDM realiza estimaciones para completar esos datos faltantes. Las estimaciones siempre se hacen sobre la resolución horaria, son validadas y ajustadas cuando el medidor envía el registro de cierre. A partir del registro de cierre es que se hacen ajustes para que las medidas estimadas sean coherentes con el consumo energético mensual registrado para el cliente.

Para el armado del conjunto de datos utilizado en esta tesis, fue necesario el diseño y desarrollo de un extractor que se conecte a HBase. Se desarrolló un programa en Java, utilizando el *framework* Hadoop MapReduce que permite procesar fácilmente grandes volúmenes de datos en forma distribuida. El programa toma como entrada:

- Archivo que indica para cada cliente: identificador de cliente y fecha de inspección.
- Resolución de la medida (quinceminutal / horario / diario).
- Cantidad de días de histórico.

Por cada cliente, se extraen medidas de energía activa y energía reactiva que se dieron previo a la realización de la inspección durante una cantidad de días. Internamente en el extractor se realiza chequeo de que para el periodo seleccionado, el medidor instalado sea teledorado y haya tenido conectividad. En caso de no encontrar una medida para cierta fecha y hora, se coloca el valor *NaN*. El extractor genera como salida un único archivo de formato csv. Cada línea tiene como formato: identificador de cliente, fecha-hora de la medida, energía activa, energía reactiva.

3.3.3. Características adicionales

Los tipos de irregularidades que se detectan día a día en las inspecciones son muy variables y no todos los tipos de fraude impactan directamente sobre la curva de consumo del medidor. Por ejemplo, en Uruguay, el tipo de fraude más común es la conexión directa a la red de baja tensión. Por esta razón, en los análisis, es importante considerar características extras de los clientes, que puedan aportar información que no se vea reflejada en la curva de consumo energético. Esto ayuda a los algoritmos a encontrar patrones que están vinculados a otros aspectos como ubicación, comportamiento frente al pago de facturas, antecedentes de irregularidades, y características del servicio contratado. Por lo tanto, además de datos de mediciones, se tomaron datos del sistema comercial *Customer Care & Billing* (CC&B) de UTE. De ahí se obtuvieron datos característicos y contractuales de los clientes, los cuales están almacenados en una base de datos relacional Oracle. La elección de estas características proviene de trabajos previos [37, 39]. Las características en cuestión son:

- Identificador de cliente.
- Fecha de inspección.
- Fecha de extracción de los datos.
- Los últimos 106 consumos facturados previos a la fecha de extracción de los datos.
- Cantidad de consumos que fueron reales (no estimados).
- Potencia contratada.
- Ubicación geográfica del medidor.
- Estado del servicio al momento de ser inspeccionado (indica si el contrato estaba vigente o dado de baja).
- Cantidad de días de mora que tuvo el cliente.

Capítulo 3. Uso de temperatura ambiente en la detección de NTL

Fuente	Promedio	Desviación estándar
Open Weather	18.4°C	6.3°C
Convencionales INUMET	17.8°C	6.5°C
Automáticas INUMET	18.4°C	6.5°C

Tabla 3.1: Promedio y desviación estándar para cada fuente de datos: Open Weather, estaciones convencionales INUMET y estaciones automáticas INUMET.

- Cantidad de irregularidades cometidas.
- Cantidad de días que pasaron desde el último inicio de contrato de servicio.
- Cantidad de días que pasaron desde la ante última inspección (en caso de haber tenido más de una inspección).
- Etiqueta que indica si se cometió fraude.

3.3.4. Recolección de datos de temperatura

Para poder armar el conjunto de datos de temperaturas de todos los departamentos de Uruguay con resolución horaria, fue necesaria la recolección de datos de diversas fuentes.

En primer lugar, se tomaron datos del sistema Meter Data Management (MDM), perteneciente a UTE. Dado que se tenían muy pocos datos en ese sistema fue necesario solicitar información al Instituto Uruguayo de Meteorología (INUMET) [26], el cual brindó datos de sus estaciones meteorológicas, donde algunas de ellas son convencionales y otras automáticas. Una estación convencional está dotada de personal encargado de tomar las mediciones. En las otras, las mediciones se realizan de forma automática. Y por último, también se tomaron datos históricos de Open Weather [47]. Open Weather es una compañía que provee servicios para obtener datos meteorológicos actuales e históricos y pronósticos, para cualquier ubicación geográfica. Open Weather utiliza una red neuronal convolucional para el pronóstico del tiempo y el cálculo de datos históricos. Esta red se alimenta de datos de: modelos numéricos de predicción meteorológica, estaciones meteorológicas, datos de radares y datos satelitales. Durante el desarrollo de este trabajo, surgió la duda de qué tan confiables son las estimaciones realizadas por Open Weather dado que sus datos son creados por un modelo. En [60] se puede encontrar parte de su estudio donde describen se los métodos y exponen los valores de algunas de las métricas que utilizan para medir qué tan precisos son sus pronósticos y la calidad de los mismos.

Para evaluar la calidad de los datos de OW se utiliza como referente los datos de INUMET. En primer lugar, se tomaron todos los datos en común y se calcularon valores estadísticos como promedio y desviación estándar, para cada una de las fuentes. El promedio brinda información sobre el valor típico o representativo de los datos, mientras que la desviación estándar mide la dispersión de los valores con respecto al promedio. En la Tabla 3.1 se puede ver cómo fue el promedio obtenido para cada una de las fuentes, siendo el mismo para OW y las estaciones automáticas de INUMET (18.4) y una diferencia de 0.6 con respecto a las estaciones convencionales de INUMET. También se puede ver la desviación estándar donde OW difiere en 0.2 respecto a las fuentes de INUMET.

En segundo lugar, se calcula el error cuadrático medio, ECM, entre las series de OW e INUMET. El ECM mide la diferencia promedio de los valores observados (INUMET) y los valores predichos (OW), elevados al cuadrado (ver Fórmula 3.2). El hecho de elevar al cuadrado amplifica los errores, por lo que si se comete errores grandes, va a tener un impacto grande en esta medida.

3.3. Datos utilizados

$$ECM = \frac{1}{N} \sum_{i=0}^N (x_i - \hat{x}_i)^2 \quad (3.2)$$

siendo x el valor real y \hat{x} el valor estimado.

Para poder interpretar mejor su resultado, se utiliza la raíz del error cuadrático medio (RECM) para poder obtener un valor en las mismas unidades del problema (ver Fórmula 3.3).

$$RECM = \sqrt{\frac{1}{N} \sum_{i=0}^N (x_i - \hat{x}_i)^2} \quad (3.3)$$

siendo x el valor real y \hat{x} el valor estimado.

El ECM o RECM es comúnmente utilizado para evaluar la precisión de modelos predictivos o de regresión. Cuanto menor sea el valor, mejor será el ajuste del modelo a los datos. En la Tabla 3.2 se puede apreciar cuál es el valor de RECM para los datos en común entre Open Weather y estaciones convencionales de INUMET, y, Open Weather y estaciones automáticas de INUMET.

Fuentes	RECM
Open Weather vs. Convencionales INUMET	2.4°C
Open Weather vs. Automáticas INUMET	2.3°C

Tabla 3.2: Valor de RECM obtenido al comparar datos de OW con estaciones convencionales y automáticas de INUMET.

Los cálculos indican que en promedio la diferencia de grados de Open Weather es de alrededor 2.3°C-2.4°C con respecto a los datos de INUMET, lo cual se considera aceptable.

3.3.5. Preparación del conjunto de datos

Para poder estudiar la correlación cruzada entre la temperatura y el consumo energético de los clientes de UTE, y ver cómo se ésta se comporta en clientes normales con respecto a clientes que cometen fraude, es necesario crear un conjunto de datos que contenga:

- Etiqueta indicando si el cliente cometió fraude.
- Serie de consumo (energía activa) del cliente previo a la inspección, con resolución horaria y con un historial de 3 meses.
- Serie de temperatura ambiente para el departamento donde se encuentra el cliente, con resolución horaria y para el mismo periodo de tiempo que la serie de consumos.

La preparación del conjunto de datos fue muy costosa, debido a que hubo que combinar datos de diferentes fuentes. La misma constó de varios pasos que se describen a continuación.

Se estudió la completitud de los datos de temperatura. Al considerar solamente la fuente de temperaturas provista por MDM, se constató que faltaban un 10.3% del total, que se corresponde con 65.768 mediciones. Por lo tanto, como primer paso se procedió a completar los datos faltantes usando los datos de INUMET. En caso de poder completar el faltante con dato de una estación convencional o automática, se le

Capítulo 3. Uso de temperatura ambiente en la detección de NTL

da prioridad a la automática porque se entiende que evita errores humanos que puedan existir tanto en la observación como en el ingreso del dato.

En las Figura 3.2 se puede apreciar las curvas de temperatura por año y por departamento. Se diferencia la fuente del dato según el color, siendo verde los datos que provienen de MDM, naranja los datos que provienen de estaciones automáticas de INUMET y rojo los datos que provienen de estaciones convencionales de INUMET.

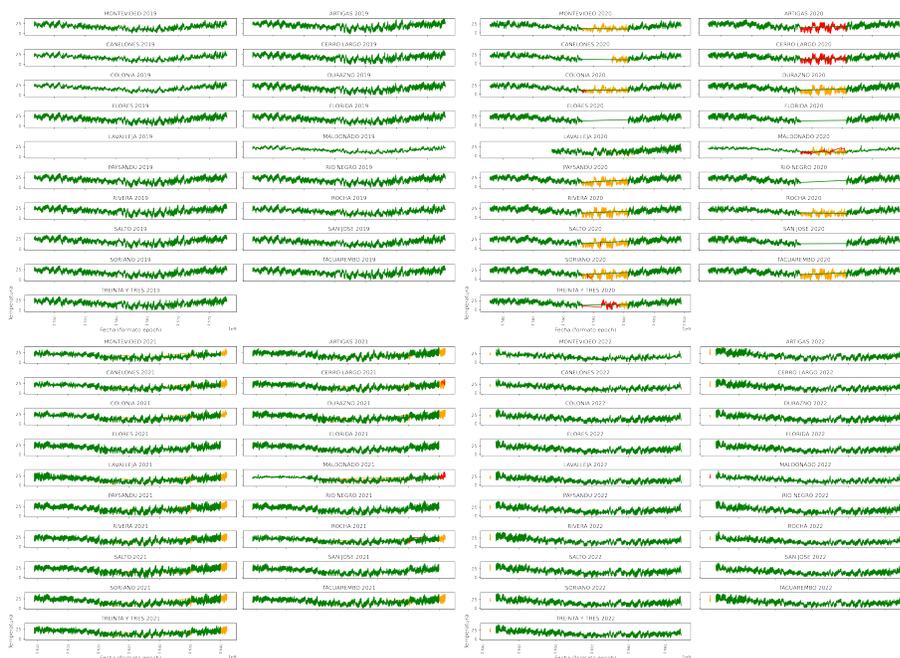


Figura 3.2: Curvas de temperatura por año y por departamento. Se diferencia la fuente del dato según el color, siendo verde los datos que provienen de MDM, naranja los datos que provienen de estaciones automáticas de INUMET y rojo los datos que provienen de estaciones convencionales de INUMET.

Luego de haber usado los datos de INUMET, se logró bajar el porcentaje de faltantes a 5.5%. Se realiza un histograma para ver la distribución de los huecos según el departamento y el año. En la Figura 3.3 se puede apreciar que el departamento de Lavalajea es el más afectado.

Como siguiente paso, se intenta completar el resto de faltantes usando los datos históricos de OW. En las Figura 3.4 se puede apreciar las curvas de temperatura por año y por departamento. Se diferencia la fuente del dato según el color, siendo verde los datos que ya fueron consolidados y provienen de MDM e INUMET, y en color naranja los datos históricos de OW.

Luego de haber usado los datos de OW, se logró bajar el porcentaje de faltantes a 0.6%, que se corresponde con un total de 4072 mediciones. Con el fin de llevar la cantidad de huecos a cero, se adoptaron dos estrategias más para lograrlo. Como siguiente paso, se realizó una interpolación lineal con algunas restricciones: los valores a interpolar deben estar entre valores válidos (es decir, los límites no pueden ser NaN) y se completa hasta 4 huecos consecutivos máximo. Si el hueco es mayor a 4, la interpolación no actúa. Esta decisión se debe a que existen huecos muy grandes y

3.3. Datos utilizados

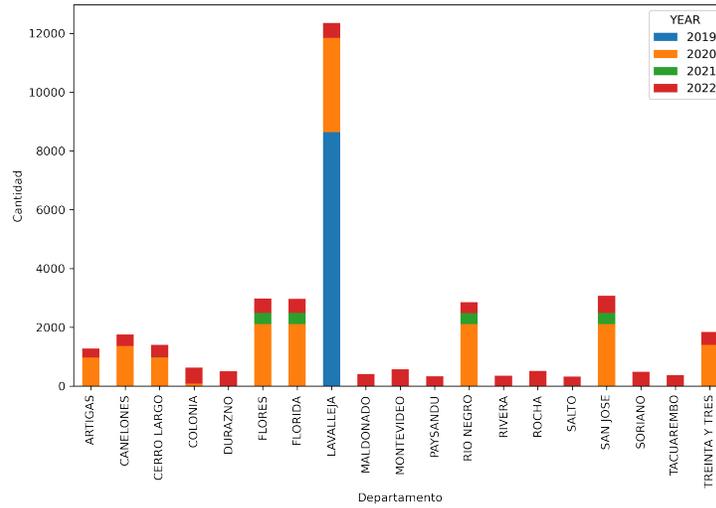


Figura 3.3: Cantidad de temperaturas faltantes por departamento y año.

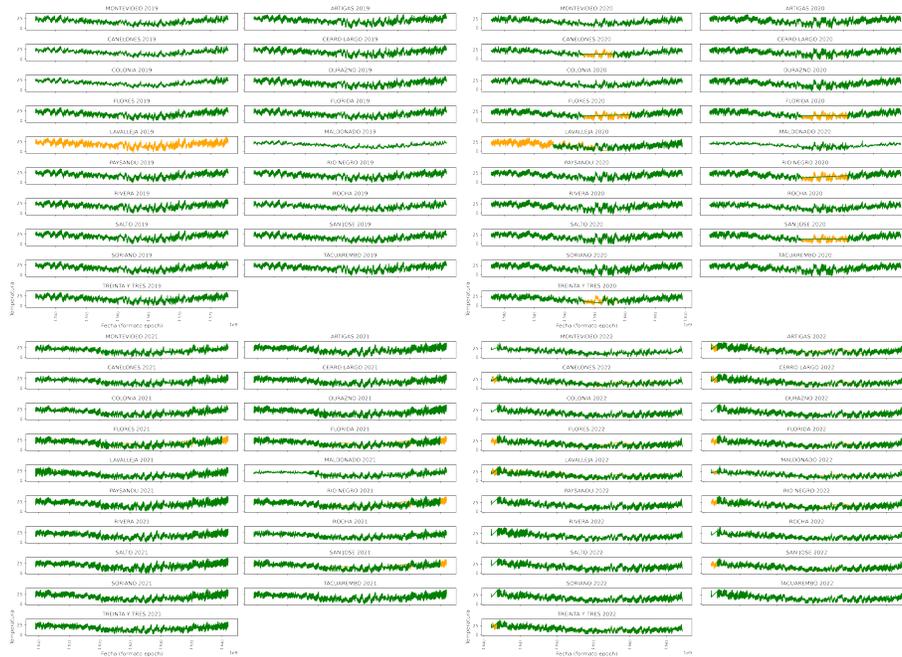


Figura 3.4: Curvas de temperatura por año y por departamento. Se diferencia la fuente del dato según el color, siendo verde los datos que provienen de MDM e INUMET, y naranja los datos históricos que provienen de OW.

el cálculo podría conducir a valores erróneos. Ejemplo: si se tiene un departamento X con una medición de 11°C para el día-hora 3-10-2023 11:00 y su próxima medición corresponde a 6°C para el día-hora 4-10-2023 5:00, la interpolación lineal sin restricciones

Capítulo 3. Uso de temperatura ambiente en la detección de NTL

completaría los huecos de forma decreciente cuando en realidad en las horas del mediodía-tarde correspondientes al 3-10-2023 podrían eventualmente seguir ascendiendo. Por esa razón es que se busca introducir el menor ruido posible, completando como máximo 4 huecos consecutivos y dejando sin completar aquellos huecos que tengan un largo mayor a 4.

Luego de la interpolación lineal, se logró seguir bajando el porcentaje de faltantes a 0.3%, que se corresponde con un total de 2177 mediciones. Finalmente, la estrategia adoptada para completar este remanente es utilizar los datos del vecino más cercano. Para ello, fue necesario el cálculo de distancia entre todos los departamentos y luego tomar para cada departamento aquél que se encuentre más cerca en términos de kilómetros. Una vez que se tiene el departamento más cercano, se copia la temperatura correspondiente al mismo día, hora y año. De esta forma se logró llevar a 0% los faltantes para el periodo desde Enero 2020 hasta Octubre 2022. Posterior a la completitud de datos, se hizo una validación de los datos a nivel general, la cual permitió detectar que no todas las fuentes tenían los datos reportados en la misma zona horaria.

Una vez que se tiene el conjunto de datos de temperaturas, se procede a armar un conjunto que centralice las series de consumo energético y de temperatura ambiente por cliente. Para ello, fue necesario combinar 3 fuentes:

- Series de consumo de energía activa (definidas en Sección 3.3.1).
- Características adicionales (definidas en Sección 3.3.3), de las cuales solamente se considera un subconjunto: identificador de cliente, etiqueta de si se cometió fraude, fecha de inspección, latitud y longitud.
- Datos de temperatura (definidas en Sección 3.3.4).

Para combinar las características adicionales con los datos de temperatura fue necesario calcular cuál es la estación meteorológica más cercana en términos de kilómetros según la ubicación del cliente.

En la Figura 3.5 y Tabla 3.3 se puede observar cómo es la distribución de clientes de la base, según el departamento, siendo Montevideo ampliamente superior en cuanto a la cantidad de inspecciones.

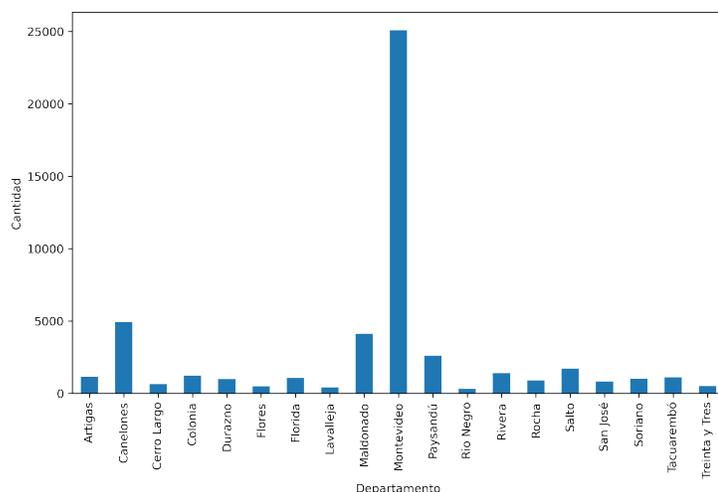


Figura 3.5: Cantidad de inspecciones por departamento.

3.4. Correlación entre temperatura ambiente y consumo energético

Departamento	Inspecciones		
	Total	Normales	Con Fraude
Montevideo	25078	23120	1958
Canelones	4919	4251	668
Maldonado	4102	3995	107
Paysandú	2601	2385	216
Salto	1714	1564	150
Rivera	1390	1322	68
Colonia	1230	1199	31
Artigas	1148	1066	82
Tacuarembó	1111	1086	25
Florida	1066	1030	36
Soriano	1006	925	81
Durazno	990	957	33
Rocha	891	884	7
San José	824	788	36
Cerro Largo	640	629	11
Treinta y Tres	513	508	5
Flores	491	459	32
Lavalleja	415	379	36
Rio Negro	303	298	5

Tabla 3.3: Cantidades de inspecciones según el departamento realizadas por UTE entre Enero 2020 y Octubre 2022.

Luego de vincular la información de características con la de temperatura, se procede a combinar con la serie de consumos. Esta combinación es más simple debido a que se hace por identificador de cliente. Un detalle a tener en cuenta es que siempre se valida que los consumos vinculados a un cliente sean todos previos a la fecha de inspección. Este paso es fundamental para evitar la introducción de ruido en los estudios y modelos, ya que una vez que se lleva a cabo la inspección en clientes fraudulentos, es sabido que los mismos cambian su comportamiento por el hecho de haber sido descubiertos y penalizados.

3.4. Correlación entre temperatura ambiente y consumo energético

En base al conjunto de datos armado en Sección 3.3.5, el cual contiene datos de 50432 clientes de UTE inspeccionados por técnicos de la empresa entre Enero 2020 y Octubre 2022, se procede a verificar si existe una relación hipotética en forma de U entre la variación de temperatura y la demanda de electricidad que es mencionada en la literatura existente [17]. Esta curva en forma de U, debería estar centrada alrededor de 18.3°C, que también es un valor muy utilizado en la literatura desde 1952 [55] y aún se sigue usando en la actualidad [15, 18, 58]. Li et al. [34] indicaron que la curva de respuesta a la temperatura del consumo eléctrico es plana cuando la temperatura

Capítulo 3. Uso de temperatura ambiente en la detección de NTL

ronda alrededor de 18.3°C porque se supone que para esta temperatura exterior, no se necesita de calefacción ni refrigeración para sentirse cómodo.

En la Figura 3.6 se presenta cuál es el consumo energético promedio (línea azul) en base a la temperatura ambiente (histograma verde). En el mismo se puede ver como el promedio más bajo se centra alrededor de 18°C y fuera de ese valor, el consumo energético tiende a incrementarse, lo cual va de la mano con lo mencionado en la literatura.

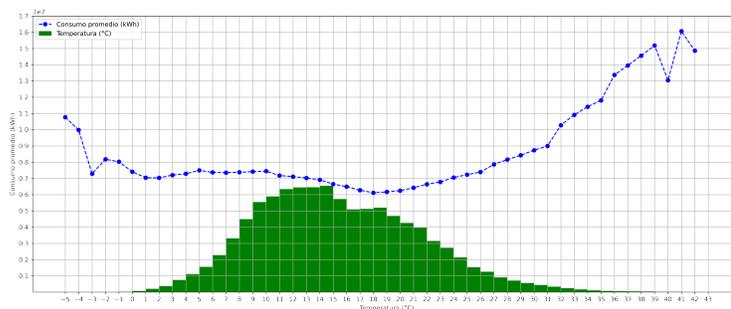


Figura 3.6: Consumo promedio según temperatura registrada

Siguiendo con el estudio, se procede a calcular la correlación cruzada, se considera un adelantamiento/retraso de hasta 12 horas, esto va de la mano del concepto de inercia térmica, que al no tener información en cuanto a la construcción de las viviendas de los clientes, se estudia la correlación cruzada para los desfases entre $[-12, 12]$ y se elige aquel que haga que la correlación sea máxima.

Como ya se ha mencionado, es importante tener en cuenta que la correlación entre temperatura y consumo de energía cambia según la estación del año. Por lo tanto, para poder estudiar en paralelo los meses de invierno y verano, se estudia la correlación utilizando el valor absoluto $|temperatura\ ambiente - 18,3^{\circ}\text{C}|$. Se consideró un umbral de temperatura de 18.3°C en base a su prevalencia en la literatura científica.

Para analizar los resultados obtenidos de máxima correlación, se dividieron los datos en dos conjuntos. Por un lado se agrupan los clientes normales y por otro aquellos que cometieron fraude. De esta forma, lo que se intenta ver es si la correlación entre temperatura y consumo de energía se comporta igual en ambos conjuntos.

Para este estudio primero se tomó un enfoque visual, para poder evaluar y explorar rápidamente las diferencias. Graficando el histograma y la función de densidad se brinda una idea de cómo se distribuye la correlación cruzada en ambas distribuciones. Se representa en color verde los clientes normales y en color rojo los clientes que han cometido fraude, ver Figura 3.7.

Si bien la forma de la distribución es muy similar, parece ser que las correlaciones en el rango $[0,0625 - 0.25]$ corresponden en mayor medida a clientes normales mientras que fuera de ese rango corresponden en mayor medida a clientes con fraude. Este dato combinado con otros, como por ejemplo, la ubicación geográfica puede llegar a aportar mucha información. Dentro del conjunto de datos, se puede segmentar los clientes en distintos subconjuntos tales como: ubicación geográfica, potencia contratada, dado que esos segmentos determinan en cierta parte cómo es el comportamiento del cliente en cuanto al consumo energético. Por esa razón, al combinar esos datos con la máxima correlación puede aportar información muy útil.

También se realizó el test de Kolmogorov-Smirnov que permite, dada una variable,

3.5. Evaluación de nueva característica de correlación frente a características ya conocidas

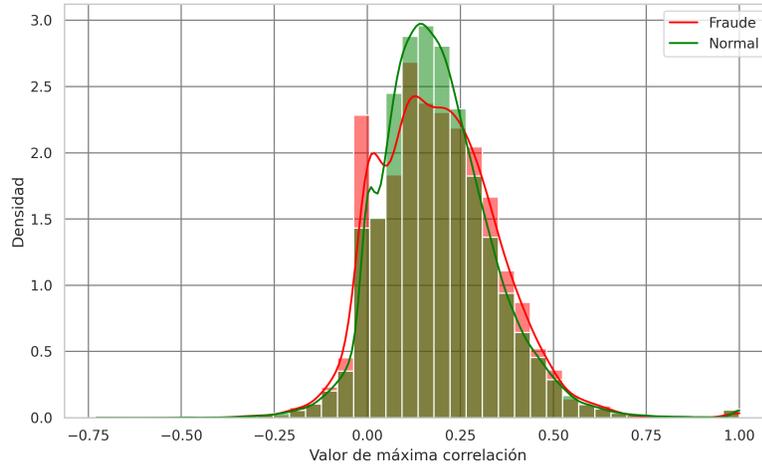


Figura 3.7: Función de densidad e histograma de la máxima correlación en clientes normales (verde) y clientes que cometen fraude (rojo)

inferir si ambos conjuntos de datos son muestras de una misma función de distribución. Para esto evalúa la diferencia entre las funciones de distribución de ambos conjuntos de datos. Este test da como resultado dos valores, uno llamado *valor p* y otro llamado *estadístico*. Cuando el valor de p es menor a 0.5, se dice que se rechaza la hipótesis nula que en este caso trata de ver si los datos provienen de la misma distribución. Además, cuanto menor sea el valor de p , mayor será la evidencia estadística que se tenga para rechazar la hipótesis nula y concluir que las distribuciones son diferentes. El estadístico representa la diferencia absoluta máxima entre las funciones de distribución acumulada empíricas de las dos muestras. En este caso, el resultado del test da que el estadístico es de 0,048 (ver Figura 3.8) y el valor de p es $3,753e^{-7}$, por lo cual se concluye que las distribuciones son diferentes.

Por lo tanto, hasta aquí se tiene que este nuevo valor que representa la máxima correlación entre temperatura y consumo de energía, tiene sus diferencias entre los clientes normales y los que cometen fraude. Esas diferencias se pueden apreciar tanto en el resultado del test de Kolmogorov-Smirnov así como en el histograma. Por lo tanto, como siguiente paso, se decide crear una nueva característica, que será incorporada al conjunto de características adicionales definidas en Sección 3.3.3, con el objetivo de poder evaluar y confirmar si es una característica que puede aportar información útil frente a características que ya son de utilidad en la tarea de detección de NTL.

3.5. Evaluación de nueva característica de correlación frente a características ya conocidas

En la Sección 3.4, se calculó un valor de máxima correlación entre el consumo de energía y la temperatura ambiente, para cada cliente. Como este valor presentó evidencias de que es diferente cuando el cliente es normal respecto a cuando el cliente comete fraude, se desea estudiar qué sucede cuando se incorpora esta información al conjunto de datos, en términos de qué tanto puede mejorar en la tarea de detección

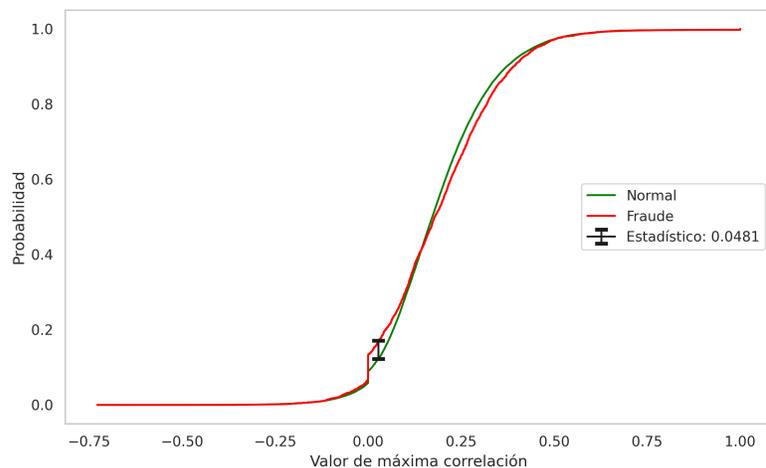


Figura 3.8: Funciones de distribución acumulada (clientes normales en color verde y clientes que comenten fraude en rojo) para ambos conjuntos junto con su estadístico del test de Kolmogorov-Smirnov

de NTL y qué tanto peso tiene la nueva característica en la decisión tomada por el clasificador.

Para descubrirlo, se plantea realizar un experimento el cual consiste en crear un clasificador sencillo, usando el algoritmo *random forest*. En las siguientes secciones se detalla todo el proceso: preparación de los datos, definición del modelo y de métricas para evaluarlo y por último se realiza la evaluación del modelo en un conjunto de test para medir el desempeño.

3.5.1. Preparación del conjunto de datos

El proceso de preparación de los datos consiste de varias etapas. En primer lugar se considera el conjunto de datos definido en Sección 3.3.3, al cual se le incorpora como característica adicional la máxima correlación calculada en Sección 3.4. Es importante tomar en cuenta que en realidad se crearán dos conjuntos de datos, uno que contiene la nueva característica y otro que la omite, para así evaluar el impacto. Los siguientes pasos se realizan en ambos conjuntos.

En primer lugar, fue necesario ajustar el contador que indica la cantidad de irregularidades cometidas. Este contador vale al menos uno para todos los clientes que han cometido fraude, por lo cual es necesario restarle 1 para que no incluya el último fraude cometido. De lo contrario, sería una característica que posiblemente tome mucho peso en la decisión del clasificador.

En segundo lugar, fue necesario hacer un ajuste en los consumos de energía facturados al cliente. Tal como sucede con las mediciones de energía activa y reactiva, siempre es importante asegurar que los consumos de energía sean previos a la fecha de inspección, debido a que los clientes cuando son descubiertos luego de cometer fraude, cambian su comportamiento. En este caso se considera los últimos tres años de consumos de energía facturados previo a ser inspeccionados. El motivo de tomar tres años viene de trabajos previos [37, 38].

3.5. Evaluación de nueva característica de correlación frente a características ya conocidas

En tercer lugar, se procede a hacer la división del conjunto de datos en dos: entrenamiento y test. Esta división se realiza en base a la fecha de inspección, de manera tal que el 80 % de las muestras queden en el conjunto de entrenamiento y el 20 % en el conjunto de test. El conjunto de entrenamiento va a permitir entrenar y ajustar los hiperparámetros del modelo. Si bien no se tiene un conjunto de datos pequeño, aún así se decide usar validación cruzada dado que el modelo tiene un gran desempeño en términos de velocidad de entrenamiento. Al hacer validación cruzada algunos de sus beneficios son: evitar que el modelo no se vea muy afectado en caso de existir valores atípicos; evitar el sobreajuste y mejorar la generalización porque el modelo se evalúa en diferentes conjuntos de validación; permite una comparación justa de las diferentes configuraciones que se prueban a la hora de ajustar un modelo. Dado que el conjunto de datos se encuentra altamente desbalanceado (6.9 % de fraude en entrenamiento y 8.1 % de fraude en test), para realizar la validación cruzada se utiliza el método *StratifiedKfold* que toma en cuenta el desbalance en cada subdivisión que realiza. El conjunto de test, contiene datos que no fueron vistos anteriormente por el modelo, por lo tanto, es una forma de hacer una evaluación objetiva de cómo se va a comportar el modelo frente a datos nuevos del mundo real.

3.5.2. Definición del modelo y métricas para su evaluación

El algoritmo *random forest* es un método de aprendizaje automático que se puede usar tanto para clasificación como para regresión. Brevemente explicando, el modo en que realiza la clasificación es a través de múltiples árboles de decisión, siendo la cantidad de árboles uno de los hiperparámetros a definir. En este ensamblado de clasificadores, cada árbol se entrena con un subconjunto de muestras aleatorio del conjunto de entrenamiento. La técnica de generar múltiples subconjuntos a partir del conjunto original se conoce como *bootstrap* y fue creada por Bradley Efron [16]). La técnica de entrenar un árbol diferente por cada subconjunto creado se la conoce como *bagging* y fue propuesta por Leo Breiman [10]. A su vez, en el caso de *random forest* cada árbol considera solo un subconjunto de características elegidas de forma aleatoria, esto hace que cada árbol sea un modelo diferente no solo porque utiliza distintos datos sino porque también utiliza distinto conjunto de características. Al momento de evaluar una instancia, la misma se evalúa en cada árbol. Cada árbol dará su clasificación y luego la clase con más votos será la elegida para la clasificación final. Lo interesante de este método es que los errores que puedan llegar a cometer algunos árboles, pueden ser corregidos por otros árboles del conjunto, lo cuál hace que el modelo resultante sea más robusto y menos propenso al sobreajuste. Los métodos basados en ensambles de clasificadores han sido muy utilizados en la detección de NTL [21, 36, 48]. En esta tesis se utiliza este algoritmo no solo como clasificador, sino como una forma de medir el impacto que tiene en la detección de NTL la incorporación de la nueva característica de correlación entre temperatura y consumo energético.

Una parte fundamental del proceso de desarrollo de un modelo, consiste en elegir una buena métrica. Una métrica bien seleccionada está fuertemente vinculada y refleja los objetivos del problema en cuestión. Es decir, si por ejemplo se desea clasificar imágenes de animales en distintas categorías puede que *precision* sea una métrica adecuada mientras que si se desea clasificar imágenes de lunares cancerígenos puede que *recall* sea la más adecuada dado que importa saber qué porcentaje de la clase positiva se está detectando. Tener en cuenta que la métrica es la que guía el proceso de entrenamiento del modelo, ayudando a realizar los ajustes necesarios para que el modelo se adecúe bien a los datos, asegurando que los resultados sean útiles y relevantes para la aplicación en cuestión. También facilita a la hora de comunicar los resultados del modelo a las partes interesadas. Una métrica comprensible y relevante facilita la

Capítulo 3. Uso de temperatura ambiente en la detección de NTL

interpretación y toma de decisiones que a menudo se traducen en términos de impacto comercial.

En este caso la métrica considerada es PR AUC, debido a su historial en detección de NTL [36–38] y porque se adecúa a las características del conjunto de datos que se encuentra fuertemente desbalanceado (en Sección 2.6 se describen más detalles acerca de esta métrica).

Dado que se tienen dos conjuntos de datos diferentes, se definen dos clasificadores *random forest* y se realizan dos búsquedas de hiperparámetros para encontrar en cada caso la cantidad de árboles y profundidad que mejor se ajusta a los datos de entrenamiento.

3.5.3. Evaluación del modelo y conclusiones

En la Figura 3.9 se puede observar los resultados obtenidos de PR AUC para ambos clasificadores *random forest*. Se puede apreciar en color naranja como el clasificador que posee la nueva característica de correlación tiene un mejor desempeño en la gran mayoría de los umbrales de decisión.

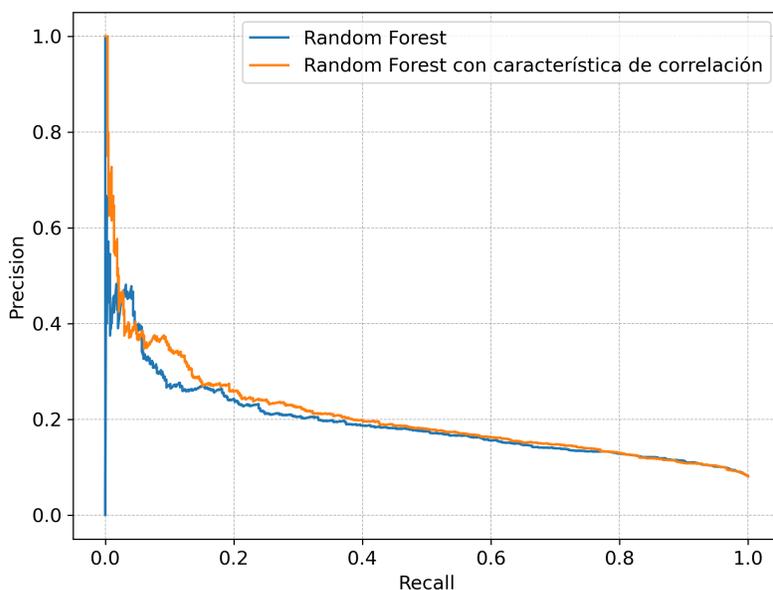


Figura 3.9: Curvas de *precision-recall* para los modelos *random forest* que consideran las características definidas en Sección 3.3.3 con y sin la nueva característica de correlación.

El PR AUC para el clasificador *random forest* sin característica es de 19.1% y para el clasificador *random forest* con característica de correlación es de 20.6%. En la Figura 3.10 se pueden apreciar las curvas de ROC, las cuales son muy similares pero nuevamente el clasificador *random forest* con característica de correlación es quien obtiene mayor área bajo la curva. El ROC AUC para el clasificador *random forest* sin característica es de 72.5% y para el clasificador *random forest* con característica de correlación es de 73.1%.

3.5. Evaluación de nueva característica de correlación frente a características ya conocidas

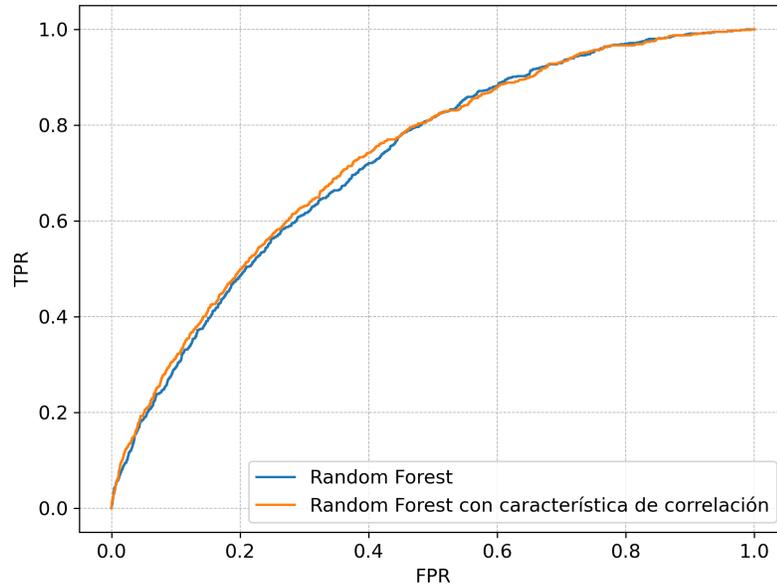


Figura 3.10: Curvas de ROC para los modelos *random forest* que consideran las características definidas en Sección 3.3.3 con y sin la nueva característica de correlación.

Continuando con el análisis, se busca ver qué importancia le da cada uno de los modelos a cada característica a la hora de clasificar las instancias. Esto permite entender cómo funciona cada clasificador y ver cómo influye la nueva característica de correlación. En primer lugar se grafica el histograma para el clasificador común, ver Figura 3.11, de ahí se puede observar cómo las coordenadas geográficas se llevan la mayor parte del peso y sobresalen por encima del resto. También es interesante observar que los consumos de energía más recientes son los más relevantes con respecto a los más antiguos. En Tabla 3.4 se detalla numéricamente el peso que tienen las primeras 10 características según el histograma.

En segundo lugar, se grafica el histograma para el clasificador que considera la nueva característica de correlación, ver Figura 3.12. De allí se puede observar cómo las coordenadas geográficas se siguen llevando la mayor parte del peso pero si se consideran en simultáneo pierden peso con respecto al experimento anterior. La nueva característica de correlación adquiere un nivel de importancia no tan alejado con respecto a las coordenadas, ocupando un tercer lugar en nivel de importancia frente al resto de las características.

En la Tabla 3.6 se presentan los resultados obtenidos en test de ambos modelos. En lo que refiere a Exactitud, *Precision* y *Recall* se define como umbral de clasificación aquel que maximiza *F1-score*. Estos resultados confirman que la nueva característica aporta información relevante en la clasificación de NTL y que además en combinación el resto de las características logra mejorar la capacidad del modelo en un 1.5%. Otro detalle que se desprende es que la máxima correlación entre consumo energético y temperatura ambiente correspondiente a los últimos tres meses tiene mayor potencial con respecto a los últimos tres años de consumos de energía mensuales.

Capítulo 3. Uso de temperatura ambiente en la detección de NTL

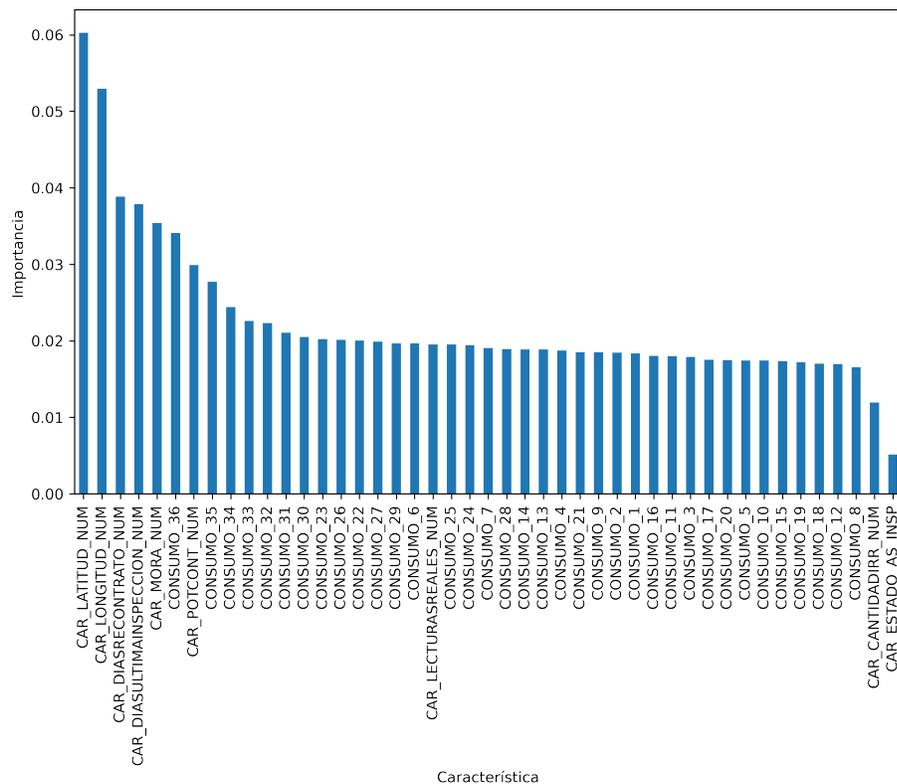


Figura 3.11: Peso que le da el modelo *random forest* a cada una de las características a la hora de clasificar instancias.

Característica	Peso
CAR_LATITUD_NUM	0.060264
CAR_LONGITUD_NUM	0.052941
CAR_DIASRECONTRATO_NUM	0.038854
CAR_DIASULTIMAINSPECCION_NUM	0.037884
CAR_MORA_NUM	0.035418
CONSUMO_36	0.034109
CAR_POTCONT_NUM	0.029905
CONSUMO_35	0.027732
CONSUMO_34	0.024399
CONSUMO_33	0.022609

Tabla 3.4: Las 10 características más importantes a la hora de clasificar instancias por el clasificador *random forest*. Para cada una se detalla numéricamente cual es el peso que tienen en la clasificación.

3.6. Conclusiones

Al inicio de este capítulo, se presentaron varias interrogantes: *¿Los clientes cambian su comportamiento junto con los cambios en la temperatura?, ¿El comportamiento es*

3.6. Conclusiones

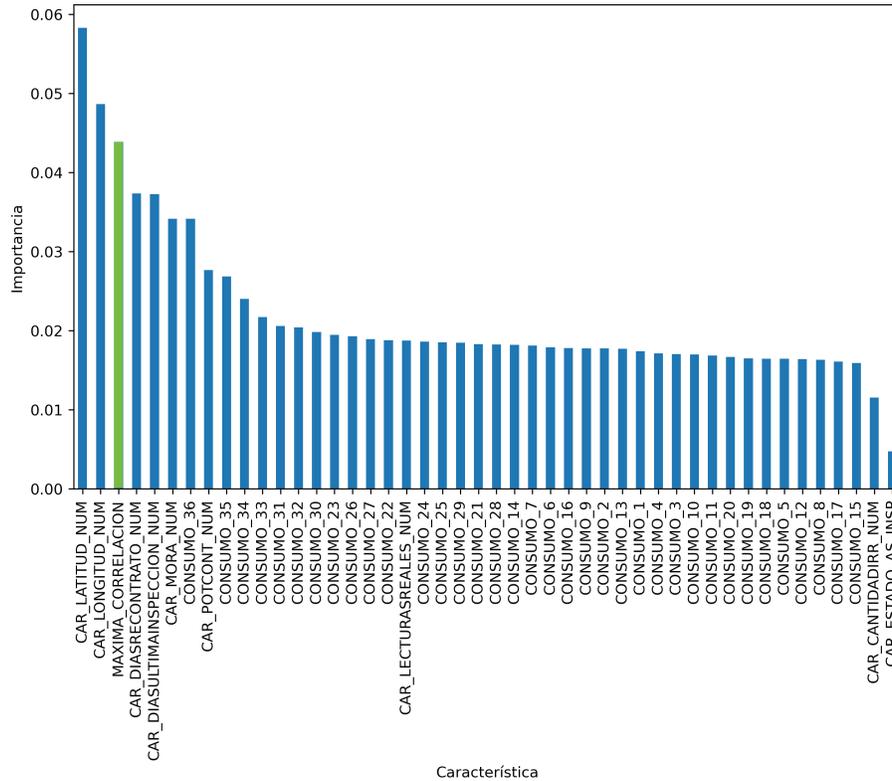


Figura 3.12: Peso que le da el modelo *random forest* a cada una de las características a la hora de clasificar instancias utilizando el conjunto definido en Sección 3.3.3.

Característica	Peso
CAR_LATITUD_NUM	0.058303
CAR_LONGITUD_NUM	0.048679
MAXIMA_CORRELACION	0.043881
CAR_DIASRECONTRATO_NUM	0.037346
CAR_DIASULTIMAINSPERCCION_NUM	0.037274
CAR_MORA_NUM	0.034152
CONSUMO_36	0.034144
CAR_POTCONT_NUM	0.027675
CONSUMO_35	0.026875
CONSUMO_34	0.024025

Tabla 3.5: Las 10 características más influyentes a la hora de clasificar instancias por el clasificador *random forest* utilizando el conjunto definido en Sección 3.3.3 y además la nueva característica de máxima correlación. Para cada una se detalla numéricamente cual es el peso que tienen en la clasificación.

el mismo en los meses de invierno y de verano?, ¿Este comportamiento es el mismo en aquellos clientes que cometen fraude?, las cuales se pudieron responder tomando como

Capítulo 3. Uso de temperatura ambiente en la detección de NTL

Modelo	Exactitud	Precisión	Recall	F1-score	PR AUC	ROC AUC
Random Forest	0.783	0.180	0.470	0.261	0.191	0.725
Random Forest con correlación	0.811	0.196	0.424	0.268	0.206	0.731

Tabla 3.6: Resultados de los clasificadores *random forest* sobre el conjunto de test.

referencia y estudiando el conjunto de datos de consumos de energía de 50432 clientes de UTE que fueron inspeccionados por técnicos de la empresa entre Enero 2020 y Octubre 2022.

A lo largo del estudio, se pudo constatar que los clientes cambian su comportamiento junto con los cambios de temperatura. Tanto ante temperaturas elevadas de verano como en temperaturas muy bajas de invierno, se observa como aumenta el consumo energético de los clientes. Se estudió la correlación cruzada entre la temperatura y el consumo de energía de estos clientes, y se tomó como referencia el máximo de esta nueva serie resultante, ya que sería el instante en el cual se encuentran más sincronizadas las series. Al analizar la máxima correlación en los conjuntos de clientes normales con respecto a los que cometen fraude, se observaron ciertas diferencias al graficar sus distribuciones y a su vez se comprobó a través del test de Kolmogorov-Smirnov que ambos conjuntos de datos no provienen de una misma distribución, lo cual ya brindó un indicio que el comportamiento es diferente. Continuando el estudio y buscando reafirmar si la nueva característica de máxima correlación puede aportar información útil en la detección de NTL, se procedió a crear dos clasificadores simples *random forest*, la diferencia entre ellos es que uno utiliza los datos de Sección 3.3.3 y el otro además de estos datos incorpora la nueva característica. De este experimento no solo se pudo comprobar que la máxima correlación mejora el desempeño del clasificador, sino que también la misma toma un peso relevante en las decisiones del clasificador. Esto abre nuevas posibilidades de trabajo y motiva a seguir trabajando con la temperatura como una de las fuentes de datos para la detección de NTL.

Surgen distintas direcciones de trabajo para seguir, como por ejemplo: incorporación de la nueva característica a los modelos ya existentes y utilizados por UTE para evaluar su desempeño en campo, diseñar nuevos modelos de aprendizaje profundo que tengan como entrada la serie de temperatura ambiente y consumo energético. No se descarta que a futuro, una vez que se obtengan los datos del Censo 2023 en Uruguay, se podría seguir mejorando este análisis ya que se dispondrá más información acerca de las características principales de las viviendas. Información adicional acerca de la construcción o métodos de calefacción/refrigeración de las mismas, permitiría un análisis más profundo que involucre estudiar aspectos como: ¿cuál es el retraso que hay que hacer para que las series alcancen su máxima correlación? y ver si esa información se condice con las características y/o equipamiento de la vivienda o si resulta sospechosa para la detección de NTL.

En esta tesis, se continua el trabajo apuntando al estudio de series temporales multivariadas.

Capítulo 4

Detección de NTL en series temporales multivariadas

4.1. Introducción

La evolución tecnológica en la industria energética ha dado paso a la implementación de uno de los desarrollos más notables, la infraestructura avanzada de medición que, entre otras cosas, permite recopilación de datos de medidores inteligentes. El despliegue masivo de medidores inteligentes en una red de AMI tiene como resultado la generación constante de grandes volúmenes de datos. Esta abundancia de información enriquece la comprensión del consumo y presenta oportunidades sin precedentes para mejorar la detección de NTL.

La capacidad de extraer conocimiento valioso de datos complejos se ha vuelto esencial para la toma de decisiones en una variedad de campos. Entre las fuentes más ricas de información se encuentran las series temporales. En el caso de detección de NTL, se ataca el problema usando series temporales de energía activa, reactiva y temperatura ambiente, y utilizando modelos de aprendizaje profundo capaces de soportar múltiples entradas. Estas arquitecturas tienen la tarea de clasificar las entradas en dos categorías: fraude o normal.

Como ya se ha mencionado, la razón de optar por modelos de aprendizaje profundo radica en su capacidad para aprender de manera automática a partir de los datos, sin conocimiento previo del dominio.

En este capítulo se describe el conjunto de datos utilizado para entrenar los modelos de detección de fraude, se detallan los pasos de su preparación, destacando varias técnicas de normalización, y se llevan a cabo experimentos para validar la técnica elegida. Además, se establece la estrategia de búsqueda de hiperparámetros.

4.2. Datos utilizados

En esta sección se describen los datos utilizados y su preparación.

4.2.1. Organización del conjunto de datos

Para los experimentos de esta tesis se consideraron un total de 50432 clientes con medidor inteligente instalado e inspeccionados por técnicos de UTE entre Enero 2020

Capítulo 4. Detección de NTL en series temporales multivariadas

y Octubre 2022. Los datos se encuentran etiquetados con 1 si se corroboró la existencia de fraude en la instalación (clase positiva) y 0 si la instalación se encontró normal (clase negativa). La base tiene 3587 clientes fraudulentos que representan aproximadamente el 7% del total. Se armaron distintos conjuntos de datos para contestar las preguntas de investigación definidas en Sección 1.2.1, donde se varía la cantidad de series involucradas y la cantidad de días de histórico.

Dada la relevancia de contar con datos lo más completos posible en cualquier análisis para garantizar resultados confiables, es esencial comprender el grado de disponibilidad de datos a lo largo del tiempo en este conjunto de datos. Se define como completitud de las series temporales, al porcentaje de datos no nulos para cierta inspección a lo largo de un periodo. En la Figura 4.1 se puede observar la completitud de las series temporales cuando se considera un periodo de tres meses, considerando el total de inspecciones y luego el subconjunto de aquellas inspecciones donde se constató fraude. Dado que el conjunto de datos se encuentra altamente desbalanceado, es crucial conocer la disponibilidad de datos sobre los casos de fraude, ya que éstos son de suma importancia para el entrenamiento efectivo del modelo de detección de NTL. Dado un histórico de tres meses, se tiene que el 87,6% de todo el universo de inspecciones presentan un grado de completitud de sus series temporales mayor o igual a 90%. Esto indica que la gran mayoría de las inspecciones se cuentan con mediciones para casi los tres meses de estudio. En las inspecciones donde se ha confirmado fraude, se observa que el 92.2% de ellas posee un alto grado de completitud en sus series, con un porcentaje igual o superior al 90%. Este dato es especialmente significativo dada la naturaleza minoritaria de esta clase.

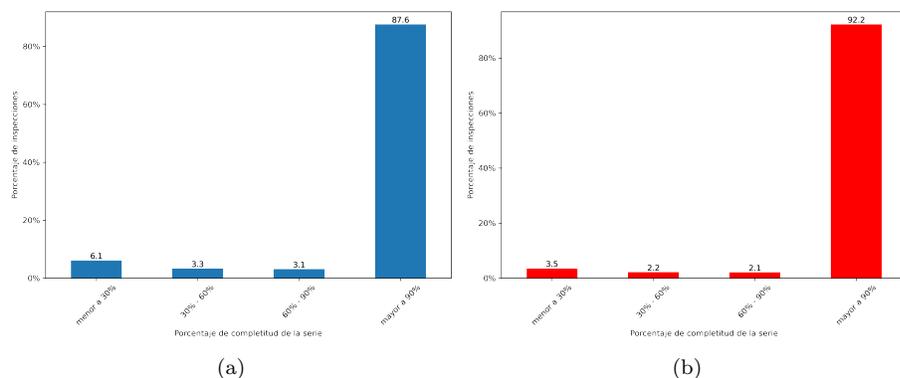


Figura 4.1: La completitud de las series temporales se define como el porcentaje de datos no nulos para cada inspección durante un período determinado. En los gráficos de barra se presenta los rangos de porcentaje de completitud de las series temporales para cada inspección en el eje X, mientras que en el eje Y se muestra el porcentaje de inspecciones que cumplen con cada rango. El periodo considerado es de tres meses. La Figura (a) es sobre el total de inspecciones y la Figura (b) considera solamente las inspecciones donde se constató fraude.

Cuando se consideran doce meses de histórico, el porcentaje de completitud de todo el universo de inspecciones se reduce a 58.7% y también se reduce a 63.6% el que considera las inspecciones donde se confirmó fraude (ver Figura 4.2). El motivo principal de esta reducción se debe a que, en el momento de la inspección, el cliente aún no había completado un año con el medidor inteligente instalado.

Los datos se organizan en tres archivos: uno para el conjunto de características especiales descriptas en 3.3.3, otro para las series de energía activa y reactiva con

4.2. Datos utilizados

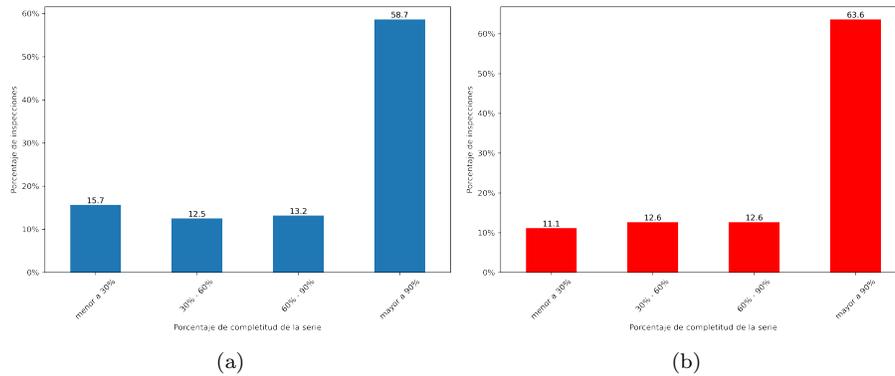


Figura 4.2: La completitud de las series temporales se define como el porcentaje de datos no nulos para cada inspección durante un período determinado. En los gráficos de barra se presenta los rangos de porcentaje de completitud de las series temporales para cada inspección en el eje X, mientras que en el eje Y se muestra el porcentaje de inspecciones que cumplen con cada rango. El periodo considerado es de doce meses. La Figura (a) es sobre el total de inspecciones y la Figura (b) considera solamente las inspecciones donde se constató fraude.

resolución horaria correspondiente a doce meses y por último, otro que contiene las temperaturas en resolución horaria para cada uno de los departamentos de Uruguay desde Enero 2020 hasta Octubre 2022.

4.2.2. Preparación de los datos

La preparación de los datos es una etapa fundamental en el desarrollo de cualquier trabajo de aprendizaje, ya que la calidad y adecuación de los datos de entrada va a determinar el rendimiento y la capacidad de generalización del modelo.

En cuanto a la adquisición de los datos, la misma se aborda en las Secciones 3.3.2 y 3.3.4.

Dado que se trabaja con series temporales multivariadas, un punto clave en la preparación de los datos es ver qué ventana de tiempo considerar, qué hacer cuando la longitud de la secuencia no alcanza el mínimo establecido y cómo organizar los datos para que el modelo lo procese. En esta tesis se define que todas las series deben tener el mismo largo fijo. Este largo puede variar según el experimento: tres meses o doce meses. La decisión de empezar a considerar doce meses fue porque al incorporar la temperatura, se cree que puede ser valioso ver cómo el cliente se comporta en los meses de invierno y de verano en simultáneo. Por lo tanto, pueden suceder dos escenarios que impliquen manejar valores nulos: pérdida de alguna medida para el periodo seleccionado (con valor NaN), o puede suceder que no se tenga suficiente historial de medidas y la serie temporal sea más corta. Para ello, todo valor NaN , se sustituye por cero y para las secuencias que no alcancen el largo mínimo se les realiza *zero-padding*.

En cuanto a la normalización, es otro paso crítico en la preparación de los datos para que los mismos tengan una escala similar, lo que ayuda a mejorar la convergencia del modelo. En este escenario es necesario tener dos normalizaciones, una que involucre a las medidas de energías y otra a la temperatura. El detalle y justificación se encuentra en la Sección 4.2.3.

Con relación a dividir el conjunto de datos en entrenamiento, validación y test, es

Capítulo 4. Detección de NTL en series temporales multivariadas

otra práctica esencial en el desarrollo de modelos de aprendizaje. Esta división tiene varios propósitos que son fundamentales para ajustar, evaluar y seleccionar modelos de manera efectiva.

El objetivo de tener un conjunto de validación separado del entrenamiento, es evitar el temido sobreajuste. El sobreajuste se da cuando el modelo termina memorizando los datos de entrenamiento en vez de aprender sus patrones. La validación permite ajustar los hiperparámetros del modelo de forma que éstos maximicen el rendimiento del modelo y eludir el sobreajuste. Dependiendo del tipo de arquitectura, se definen qué conjunto de hiperparámetros se van a ajustar.

El hecho de separar un conjunto de test, permite simular cómo será el comportamiento en el mundo real, cuando el modelo se enfrente a datos nuevos que no fueron vistos anteriormente. La evaluación del rendimiento del modelo en un conjunto de test independientemente proporciona una medida más realista de la capacidad del modelo para generalizar ante situaciones desconocidas.

En este caso, la división de estos conjuntos se hace a través de la fecha de inspección y cumpliendo con ciertos porcentajes. En problemas de clasificación, en general, la división se hace de forma aleatoria, tratando de que el conjunto de test tenga una muestra representativa de todo el conjunto de datos. Pero en este caso, si se hiciera lo mismo, puede introducir sesgos relacionados con campañas que están enfocadas a ciertos tipos de fraude. Por lo tanto, se hace la división a través de la fecha de inspección buscando aprender del pasado para predecir el futuro. En la Tabla 4.1 se puede ver el total de muestras, su distribución por clase (normal es negativa y fraude es positiva) y su porcentaje de fraude con respecto al total de muestras, para cada conjunto.

Conjunto	Muestras	Muestras negativas	Muestras positivas	Porcentaje de fraude
Entrenamiento	32474	30251	2223	6.8 %
Validación	8010	7455	555	6.9 %
Testeo	9948	9139	809	8.1 %

Tabla 4.1: Total de muestras, total de muestras para cada clase (normal es negativa y fraude es positiva) y porcentaje de fraude con respecto al total de muestras, para los conjuntos de: entrenamiento, validación y test.

Por último, otro punto muy importante es cómo tratar el desbalance de clases. Existen diferentes técnicas para abordar este tema, las más conocidas son: métodos de submuestreo, sobremuestreo o generación sintética de datos o sino puede ser un enfoque a nivel del algoritmo. Este último es el elegido para esta tesis e involucra a la función de costo (conocida también como función de *loss*) de un modelo. La técnica consiste en penalizar más fuerte cuando el modelo se equivoca en la clase minoritaria. El peso de la clase minoritaria es asignado con la proporción de muestras por clase de la base de entrenamiento. A su vez, se eligen métricas que contemplan el desbalance y fueron definidas en Sección 2.6.

Luego de la preparación de los datos se hicieron verificaciones de calidad para garantizar que los cambios realizados no hayan introducido errores o distorsiones en los datos.

4.2.3. Normalización en series temporales

La normalización es una técnica que se aplica a menudo como parte de la preparación de datos para el aprendizaje automático. La técnica consiste en cambiar los valores numéricos del conjunto de datos para que sigan una distribución específica o se encuentren dentro de un rango determinado, sin distorsionar las diferencias en los

4.2. Datos utilizados

rangos de valores ni perder información. Su importancia radica en su capacidad para mejorar la estabilidad, la convergencia y el rendimiento de los modelos, así como para asegurar que los datos sean comparables y manejables. Cuando las diferentes características están en una escala similar, ayuda a estabilizar el descenso por gradiente porque hace posible utilizar tasas de aprendizaje más altas, o ayuda a que los modelos converjan más rápido para una tasa de aprendizaje determinada. Cuando los datos no están normalizados, los valores con escalas muy diferentes pueden llevar a gradientes de magnitudes dispares y provocar que el proceso de optimización sea inestable y lento. La técnica de normalización va a depender del tipo de datos y de los requisitos del modelo con el cual se esté trabajando.

En el caso de series temporales, algunos de los métodos de normalización son:

- *Min-Max scaler*: es una técnica muy simple de implementar, que transforma los valores a un rango específico, comúnmente entre $[0, 1]$ utilizando la Fórmula 4.1. Sin embargo, esta técnica puede ser sensible a los valores atípicos (*outliers*) en los datos, lo que puede afectar la distribución de los valores normalizados. En general, se recomienda combinar la normalización Min-Max junto con otras técnicas de preprocesamiento de datos, como la detección y eliminación de valores atípicos. En este caso la normalización es por columnas.

$$x_{scaled} = \frac{x - x_{min}}{x_{max} - x_{min}} \quad (4.1)$$

- *Normalize*: esta técnica consiste en tomar los datos de entrada como si fueran vectores y transformarlos individualmente para que cada uno de ellos tenga norma igual a uno. La norma más utilizada es la euclidiana (L2). Esta normalización se logra a través de la Fórmula 4.2. En este caso la normalización puede ser por filas o columnas.

$$x_{normalized} = \frac{x}{\|x\|_2} \quad (4.2)$$

siendo $\|x\|_2 = \sqrt{x_1^2 + x_2^2 + \dots + x_n^2}$

- *Estandarización z-score*: este método consiste en transformar los valores de una característica (o varias) en un conjunto de datos que cumple que tenga media igual a cero y desviación estándar igual a uno. Esto se logra restando la media de los datos y dividiendo el resultado por la desviación estándar, ver Fórmula 4.3. En este caso, los datos normalizados son más robustos a los valores atípicos que otros métodos de normalización. En este caso la normalización es por columnas o global (según el escenario y los tipos de datos).

$$x_{z-score} = \frac{x - \mu}{\sigma} \quad (4.3)$$

siendo μ la media y σ la desviación estándar.

Al trabajar con series temporales resulta muy importante preservar la relación que hay entre las distintas series y además que cada una conserve su forma. En esta tesis, se evalúan las técnicas *Normalize* (utilizada ya en trabajos previos) y *Estandarización z-score*. Para ello se realiza un experimento (explicado en Sección 4.2.4) que primero busca reproducir el resultado obtenido en trabajos previos y segundo, medir el impacto que tiene modificar la normalización.

4.2.4. Experimento con distintas normalizaciones

Para poder medir el impacto que tiene un cambio en la normalización, fue necesario reproducir el trabajo realizado en [36] con la arquitectura CNN2D. Se utilizó el mismo

Capítulo 4. Detección de NTL en series temporales multivariadas

conjunto de datos y arquitectura de ese entonces. El conjunto de datos se caracteriza por tener un total de 20726 clientes con medidor inteligente instalado e inspeccionados por técnicos de UTE entre Diciembre 2018 y Octubre 2021. Los datos se encuentran etiquetados con 1 si se corroboró la existencia de fraude en la instalación (clase positiva) y 0 si la instalación se encontró normal (clase negativa). La base tiene 1707 clientes fraudulentos que representan aproximadamente el 8% del total. Para cada uno de los clientes se tiene los consumos de energía activa quinceminutales correspondientes a tres meses, es decir que por cada día se tiene 96 mediciones por cliente y en los tres meses se tiene 8640 mediciones por cliente. El conjunto de datos fue dividido en tres conjuntos: entrenamiento, validación y test, según la fecha de inspección y de modo que en test se encuentre el 20% del total de muestras y en validación el 20% del total del conjunto de entrenamiento. En cuanto a la arquitectura, se utiliza una red convolucional, que se puede apreciar en la Figura 4.3. Este modelo recibe como entrada imágenes por lo tanto, es necesario reorganizar la entrada de 8640 mediciones en una grilla de 90x96, donde 90 es la cantidad de días y 96 la cantidad de mediciones por día. Cada fila de la grilla es la medición de un día completo.

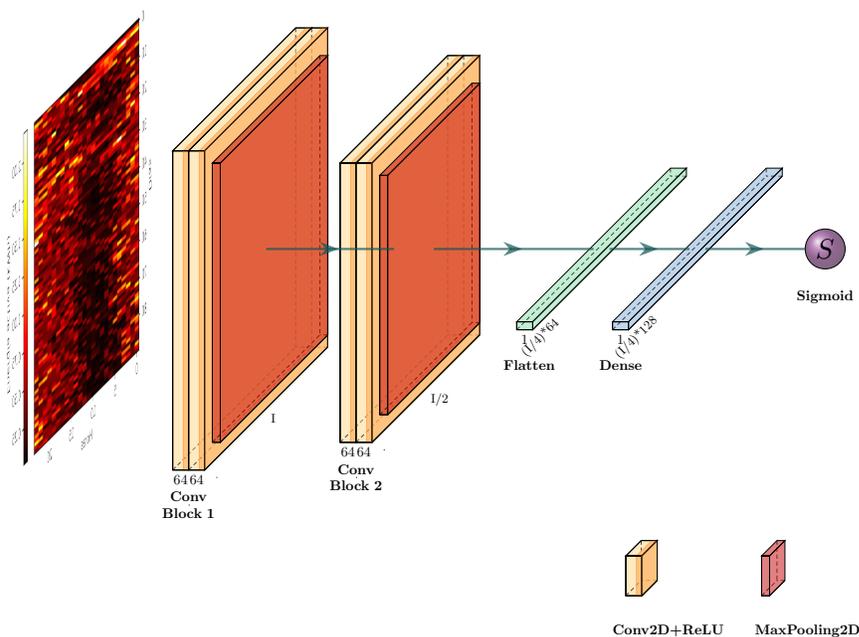


Figura 4.3: Arquitectura CNN2D para detección de NTL que se alimenta de consumo de energía activa en resolución quinceminutal.

La normalización realizada durante la preparación de los datos fue Normalize por filas, lo que fuerza a que cada fila tenga norma 1. La razón de que se hiciera por fila y no por columnas es porque las muestras no se encuentran todas alineadas temporalmente (ya que cada una tiene su propia fecha de inspección), entonces se estaría utilizando datos de distintos instantes temporales para normalizar, lo cual agregaría ruido. Para el entrenamiento, la métrica elegida para evaluar el desempeño es PR AUC, el modelo

4.2. Datos utilizados

elegido para probar en test es aquel que obtenga el máximo valor de esta métrica para el conjunto de validación. Como resultado del entrenamiento, se obtiene un 20 % de PR AUC para el conjunto de validación. El desempeño en test es de un 12 % de PR AUC. Mientras que si se cambia la normalización por el método Estandarización *z-score* por columna, el mismo se implementa tomando todos los valores de energía activa de todos los clientes y calculando su media y desviación estándar y por último aplicando la Fórmula 4.3. Aplicado este cambio, si se vuelve a entrenar el modelo, los números obtenidos son: el desempeño es 49 % de PR AUC para el conjunto de validación y es 40 % de PR AUC para el conjunto de test. Lo cual marca una diferencia considerable. En la Figura 4.4 se puede apreciar ambas curvas de PR AUC.

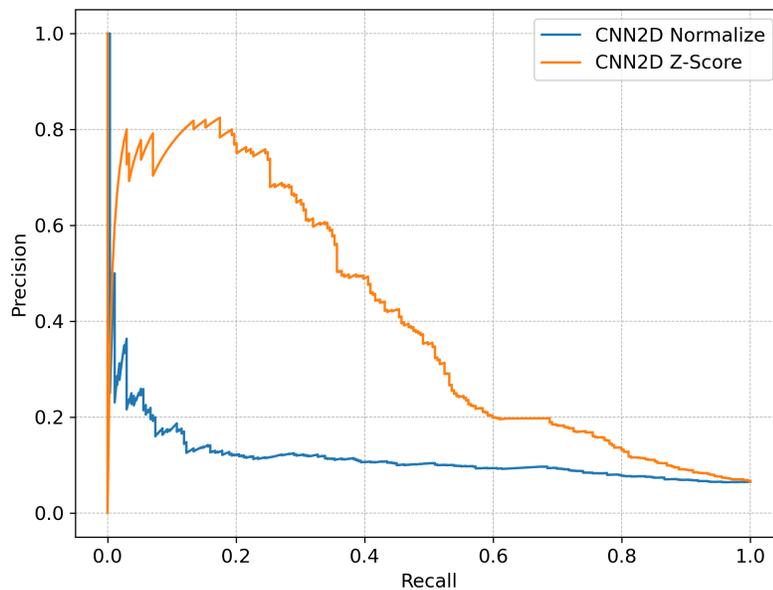


Figura 4.4: Curvas de *precisión-recall* para el conjunto de test de los modelos: CNN2D con método de normalización Normalize y CNN2D con método de estandarización *z-score*.

A su vez, en la Figura 4.5 se pueden ver las curvas de ROC, aquí también se puede apreciar cómo la curva del modelo que preparó los datos usando la estandarización *z-score* está por encima, lo cual indica que es mejor en cuanto a la capacidad de distinguir las clases positivas de las negativas.

Además, en Tabla 4.2 se puede observar el resto de las métricas. En lo que refiere a Exactitud, *Precision* y *Recall* se define como umbral de clasificación aquel que maximiza *F1-score*. El experimento con estandarización *z-score* es superior en todas. Una explicación a esto puede ser que cuando se fuerza a que cada muestra tenga una norma igual a 1, si bien no se pierde la forma de la curva, sí se está perdiendo información acerca del valor real, se pierde noción acerca de qué tan más grande o más chico es un valor con respecto al resto de las muestras. Esa pérdida de información puede ser la que esté explicando el cambio en el desempeño.

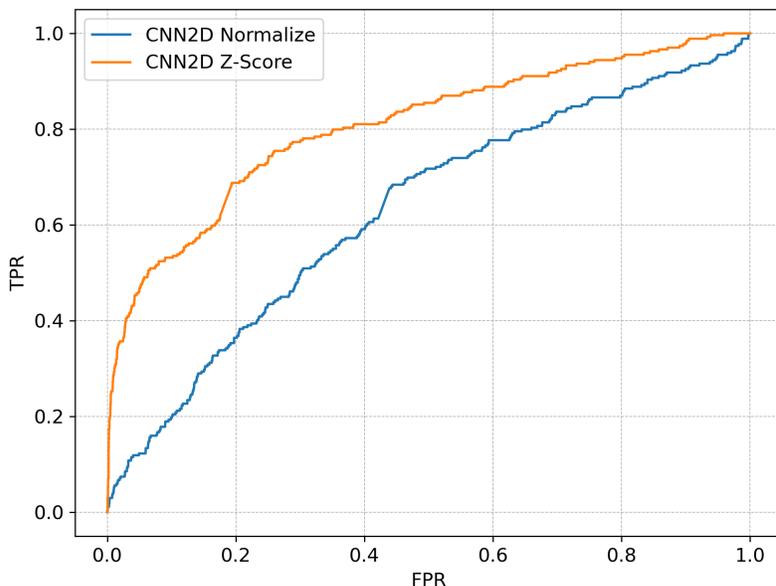


Figura 4.5: Curvas de ROC sobre el conjunto de test de los modelos: CNN2D con método de normalización Normalize y CNN2D con método de estandarización *z-score*.

Escenario	Exactitud	Precision	Recall	F1 score	PR AUC	ROC AUC
CNN2D + Normalize	0.803	0.121	0.323	0.176	0.116	0.630
CNN2D + z-score	0.934	0.491	0.401	0.442	0.399	0.806

Tabla 4.2: Valor obtenido para cada una de las métricas sobre el conjunto de test tanto para CNN2D con método de normalización Normalize y para CNN2D con método de estandarización *z-score*. Siendo este último el modelo que mejor rendimiento tiene a nivel de todas las métricas.

4.3. Estrategia para la búsqueda de hiperparámetros

Al momento de construir una red neuronal profunda, se pueden diferenciar tres tipos de parámetros: parámetros de entrada que son las muestras que alimentan el modelo, ya sea en la fase de entrenamiento, validación o test; los parámetros internos de la red (pesos y sesgos) que se van ajustando automáticamente durante la fase de entrenamiento para minimizar la función de *loss*; y los hiperparámetros, que a diferencia de los anteriores, no se aprenden automáticamente de los datos sino que se establecen antes de iniciar el proceso de entrenamiento.

Los hiperparámetros varían según la arquitectura elegida. Algunos ejemplos son: tasa de aprendizaje, número de capas, número de neuronas, función de activación, cantidad de épocas, tamaño de lote (*batch*), entre otros. El número de hiperparámetros a configurar puede ser muy elevado dependiendo de la complejidad de la red y no existe una receta que permita saber de antemano los mejores valores, sino que requiere de un duro trabajo de ensayo y error. Idealmente, se deben probar todos los valores posibles para conocer los valores óptimos, pero hacer esto manualmente podría requerir una cantidad considerable de tiempo y recursos, lo cual no sería viable. Para solucionarlo,

4.3. Estrategia para la búsqueda de hiperparámetros

existen diferentes técnicas para la búsqueda de hiperparámetros. En esta tesis se describen las dos que fueron utilizadas: búsqueda en cuadrícula (*grid search*) y búsqueda aleatoria.

Para la búsqueda en grilla, se define un conjunto de valores posibles para cada hiperparámetro y el algoritmo hace una búsqueda exhaustiva, probando todas las combinaciones posibles. El costo computacional va a estar directamente ligado a la cantidad de valores posibles que se defina para hiperparámetro y a la cantidad de hiperparámetros. Por otro lado, la búsqueda aleatoria, selecciona un valor para cada hiperparámetro de forma independiente utilizando una distribución de probabilidad.

Dado que en esta tesis se experimentan con múltiples arquitecturas de aprendizaje profundo y las búsquedas se pueden volver muy complejas, se decide dejar con valor fijo ciertos valores de hiperparámetros y para el resto se realiza una búsqueda que combina búsqueda en cuadrícula y búsqueda aleatoria.

- Hiperparámetros fijos:
 - Funciones de activación
 - Tasas de dropout
 - Tamaño de lote
 - Cantidad de épocas
 - Paciencia para la parada temprana (*early stopping*)
 - Algoritmo de optimización
- Hiperparámetros con búsqueda en cuadrícula:
 - Cantidad de bloques de convolución (2, 3, 4)
 - Cantidad de filtros (32, 64, 128)
 - Tamaño de los filtros (3, 5)
 - Cantidad de neuronas (64, 128)
 - Agregar o no capa de *Batch normalization*

Usar tamaños de filtros pequeños (3 o 5) es por varias razones:

- Capacidad de representación: a pesar de tener menos parámetros, aún pueden aprender gran variedad de patrones locales como bordes, texturas. Dado que se utilizan múltiples filtros, la composición de múltiples filtros pequeños conduce a una mayor capacidad de representación en comparación con un solo filtro grande.
- Invarianza: el tener filtros pequeños permiten capturar patrones locales independientemente del lugar donde se encuentre en la imagen. Con filtros grandes, esta tarea se vuelve más compleja.
- Costo computacional: a medida que el filtro se vuelve más grande, aumenta la cantidad de cálculos que se deben realizar. Cuando se aplica un filtro a una entrada en una capa convolucional, se realiza una operación de convolución que implica desplazar el filtro sobre la entrada y realizar operaciones de multiplicación y suma. Cuanto más grande es el filtro, más valores deben multiplicarse y sumarse en cada paso del desplazamiento. Esto resulta en un aumento significativo en la cantidad de cálculos requeridos. Cuando las arquitecturas se componen de varios bloques convolucionales, muchas veces se utilizan combinaciones de tamaños. Las capas iniciales suelen utilizar filtros pequeños de 3x3 porque tienen a capturar detalles locales, mientras que las capas más profundas pueden utilizar filtros más grandes como 5x5, 7x7 para capturar patrones más complejos y abstracciones a un nivel superior.

Capítulo 4. Detección de NTL en series temporales multivariadas

En cuanto a la cantidad de filtros o neuronas a utilizar, no existe una regla estricta a seguir que determine cuánto se necesita. Es común utilizar valores que son potencia de dos por motivos de hardware. Las redes neuronales profundas suelen entrenarse en GPU para acelerar el tiempo de entrenamiento. Usar valores que son potencia de dos ayuda a la GPU a tener un rendimiento más eficiente.

- Hiperparámetros con búsqueda aleatoria:
 - Tasa de aprendizaje, valor entre $[1e^{-5}, 1e^{-2}]$ y función de muestreo 'log'.

La idea detrás del muestreo logarítmico es que los valores se incrementan o disminuyen en una proporción constante en una escala logarítmica. Es muy útil este tipo de muestreo con hiperparámetros que siguen una relación exponencial o logarítmica respecto al rendimiento del modelo.

Cada tipo de arquitectura utilizada en los experimentos tiene definido su método de búsqueda de hiperparámetros, de forma que experimentos que vinculen a la misma arquitectura realicen las mismas búsquedas y ninguno tenga ventaja sobre otro para poder hacer una comparación justa a la hora de evaluar resultados.

Capítulo 5

Experimentos con redes neuronales profundas

5.1. Introducción

Como se ha mencionado a lo largo de este documento, el aprendizaje profundo ha emergido como un paradigma poderoso para modelar patrones en datos secuenciales, gracias a su capacidad para capturar relaciones y características complejas en los datos. El enfoque de aprendizaje profundo ha demostrado su eficacia en una amplia gama de aplicaciones, desde el procesamiento de lenguaje natural hasta la visión por computadora. Sin embargo, su aplicación en el análisis de series temporales multivariadas sigue siendo un área de investigación activa y desafiante.

En este capítulo se describen una serie de experimentos que involucraron la evaluación de diferentes tipos de arquitecturas de aprendizaje profundo en el contexto específico de series temporales multivariadas. Los experimentos diseñados para esta investigación abordaron cuestiones clave sobre cómo tratar las series cuando poseen distintos largos o resoluciones, cómo normalizarlas, cómo modelarlas según el tipo de arquitectura que se use.

Los resultados obtenidos proporcionan una visión general de cómo diferentes arquitecturas de aprendizaje profundo abordan las particularidades de las series temporales multivariadas y cómo pueden aplicarse en contextos del mundo real. Adicionalmente se describen los recursos computacionales utilizados y costos en tiempo en cuanto a la búsqueda de hiperparámetros y entrenamiento final de cada uno de los modelos.

5.1.1. Datos utilizados

Para llevar a cabo los experimentos con distintas arquitecturas se utilizó el conjunto de datos que contiene un total de 50432 inspecciones que corresponden a clientes con medidor inteligente instalado e inspeccionado por técnicos de UTE entre Enero 2020 y Octubre 2022. En la base, 3587 inspecciones fueron etiquetadas con fraude (aproximadamente un 7% del total). Por cada cliente se intenta extraer doce meses de medidas pero pueden existir casos donde se hayan encontrado menor cantidad debido a que su medidor inteligente lleva menor tiempo instalado o por algún problema de conectividad haya enviado menor cantidad de medidas. Si se tiene un valor de consumo *NaN*, se sustituye por cero y para las secuencias que no alcancen el largo mínimo se les realiza *zero-padding*. Se decide trabajar con resolución horaria por dos motivos: la serie

Capítulo 5. Experimentos con redes neuronales profundas

de temperatura ambiente se encuentra en resolución horaria y las mediciones en esta resolución tienen menos datos faltantes porque los algoritmos de estimación de MDM se encargan de calcular aquellas mediciones que se pierden. En la preparación de este conjunto de datos se realizan dos estandarizaciones *z-score*, una para las energías y otra para la temperatura ambiente. La razón de no hacerlas juntas es porque la temperatura ambiente se encuentra en otro orden de magnitud. Y el motivo de estandarizar las energías juntas, calculando una única media y desviación estándar, es porque se desea preservar la forma que tienen ambas curvas y no se quiere perder cómo es la relación y proporción de ambas magnitudes que corresponden a energía consumida.

5.2. Experimentos con distintas arquitecturas de redes neuronales usando series de consumo de energía activa, reactiva y temperatura ambiente.

En esta sección, se detallan los experimentos llevados a cabo para evaluar el rendimiento de distintas arquitecturas de aprendizaje profundo en la detección de NTL usando series temporales multivariadas. Se comparan tres tipos de arquitecturas: redes convolucionales, residuales y recurrentes. Cada una de estas arquitecturas ofrece enfoques diferentes para el procesamiento de datos secuenciales y la extracción de características relevantes. A través de estos experimentos, se busca identificar qué arquitectura es la más adecuada para abordar el desafío usando series temporales de energía activa, energía reactiva y temperatura ambiente, todas en resolución horaria y con un historial de tres meses.

5.2.1. Definición de arquitectura basada en redes convolucionales

Se construye un modelo basado en la arquitectura planteada en [38]. Para este tipo de redes convolucionales 2D se organizan los datos como imágenes. Por cada serie temporal se arma una imagen 90×24 , donde se tienen 90 filas (una por cada día) y 24 columnas (una por cada hora) con 3 canales en total (una por cada serie temporal).

La arquitectura obtenida después de buscar los hiperparámetros sobre el espacio de valores definido en Tabla 5.1, será llamada CNN2D_AE_Q1_TEMP y se puede ver en la Figura 5.1. Sus resultados serán presentados en Sección 5.2.4.

Parámetro	Espacio
Filtros	64, 128
Tamaño filtro	3, 5
Neuronas capa densa	64, 128
Tasa de aprendizaje	$[1e^{-5}, \dots, 1e^{-3}]$
Cantidad de bloques	2, 3, 4

Tabla 5.1: Espacio de búsqueda de hiperparámetros

5.2. Experimentos con distintas arquitecturas de redes neuronales usando series de consumo de energía activa, reactiva y temperatura ambiente.

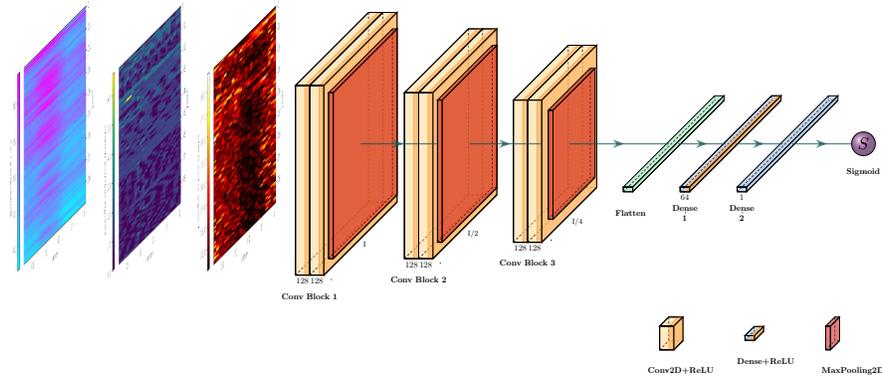


Figura 5.1: Arquitectura CNN2D_AE.Q1_TEMP para detección de NTL, resultante de la búsqueda de hiperparámetros para series de energía activa, reactiva y temperatura ambiente con resolución horaria e historial de tres meses.

5.2.2. Definición de arquitectura basada en redes residuales

Se desarrolla un modelo utilizando la arquitectura propuesta en [61] para su aplicación en clasificación de series temporales. El modelo utiliza las series de entrada en su formato original, presentadas como vectores unidimensionales.

La arquitectura resultante luego de la búsqueda de hiperparámetros sobre el espacio de valores definido en Tabla 5.2, será llamada ResNet_AE.Q1_TEMP y se puede ver en la Figura 5.2. Sus resultados serán presentados en Sección 5.2.4.

Parámetro	Espacio
Tasa de aprendizaje	$[1e^{-5}, \dots, 1e^{-2}]$
Aplicar Batch Normalization a la entrada	Si/No

Tabla 5.2: Espacio de búsqueda de hiperparámetros sobre redes residuales

5.2.3. Definición de arquitectura basada en redes recurrentes

Este experimento surge del artículo [24], donde se plantea resolver el problema de detección de NTL a través de una red LSTM con tres tipos de datos de entrada: consumo diario de los clientes, pérdidas diarias de los transformadores para las zonas donde se encuentran los clientes y registros de temperatura mínima y máxima. El historial de mediciones considerado abarca casi un periodo de dos años.

En esta tesis, no se cuenta con datos vinculados a pérdidas diarias de transformadores, por lo que se plantea una variante de la arquitectura donde solamente recibe dos series temporales: energía activa y temperatura ambiente. El modelo utiliza las series de entrada en su formato original, presentadas como vectores unidimensionales.

La arquitectura obtenida después de buscar los hiperparámetros sobre el espacio de valores definido en Tabla 5.3, será llamada LSTM_AE_TEMP y se puede ver en la Figura 5.3. Sus resultados serán presentados en Sección 5.2.4.

Capítulo 5. Experimentos con redes neuronales profundas

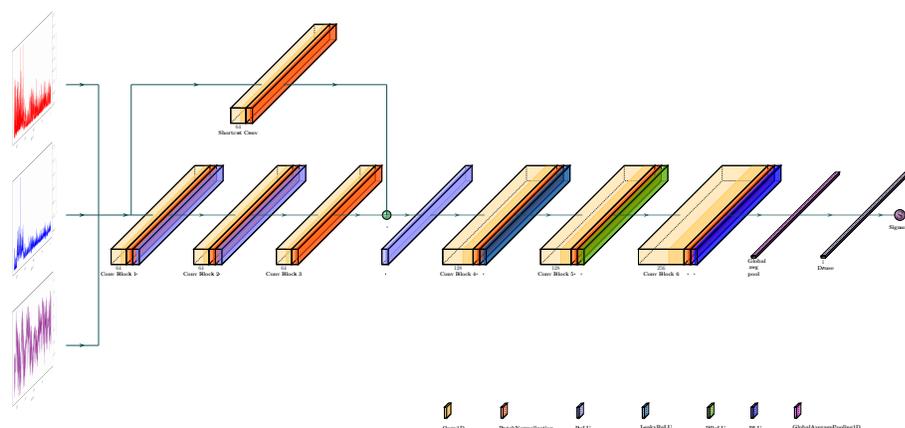


Figura 5.2: Arquitectura ResNet-AE-Q1-TEMP para detección de NTL, resultante de la búsqueda de hiperparámetros para series de energía activa, reactiva y temperatura ambiente con resolución horaria e historial de tres meses.

Parámetro	Espacio
Cantidad de neuronas	32, 64
Tasa de aprendizaje	$[1e^{-5}, \dots, 1e^{-2}]$

Tabla 5.3: Espacio de búsqueda de hiperparámetros sobre redes recurrentes

5.2.4. Comparativa entre arquitecturas

En esta sección se presentan los resultados obtenidos para cada uno de los modelos luego de ser evaluados en el conjunto de test.

Una consideración a tener en cuenta es que luego de las búsquedas de hiperparámetros, se decidió hacer un último entrenamiento de la red con la particularidad de que para entrenar se considera el conjunto conformado por la unión de entrenamiento y validación. La condición de parada del entrenamiento es por cantidad de épocas, su valor está determinado por la cantidad de épocas que fueron necesarias durante el proceso de búsqueda para esa configuración de hiperparámetros. La razón de cambiar la técnica en este último entrenamiento es debido a que se quiere aprovechar el conocimiento detrás del conjunto de validación, aprender de tipos de fraudes que quizás no estén presentes en el conjunto de entrenamiento y eventualmente poder mejorar el desempeño cuando se pruebe en test.

En la Figura 5.4 se puede observar las curvas de *precision-recall* para cada uno de los tipos de arquitecturas. Se puede apreciar como el modelo convolucional es el que mejor rendimiento obtiene frente al resto de las arquitecturas para todos los umbrales. Esto pone de manifiesto el potencial de estas redes para identificar patrones en imágenes, sugiriendo que tratar las series como imágenes resulta más ventajoso que considerarlas como vectores unidimensionales.

En la Figura 5.5 se pueden ver las curvas de ROC, aquí también se puede apreciar cómo la curva del modelo convolucional está por encima del resto, lo cual indica que es mejor en cuanto a la capacidad de distinguir las clases positivas de las negativas.

En la Tabla 5.4 se puede ver el valor obtenido para cada una de las métricas. En lo que refiere a Exactitud, Precision, Recall y F1-score se define como umbral de clasificación aquel que maximiza F1-score, ya que de esta manera se logra encontrar

5.2. Experimentos con distintas arquitecturas de redes neuronales usando series de consumo de energía activa, reactiva y temperatura ambiente.

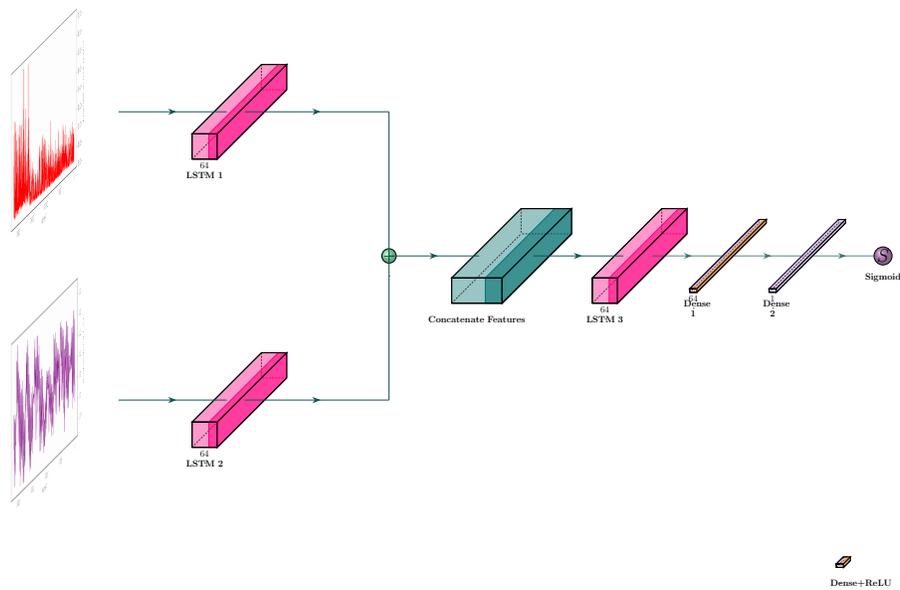


Figura 5.3: Arquitectura LSTM-AE_TEMP para detección de NTL, resultante de la búsqueda de hiperparámetros para series de energía activa y temperatura ambiente con resolución horaria e historial de tres meses.

un balance ideal entre las métricas *recall* y *precision*. Esto es crucial para problemas con conjuntos de datos desbalanceados. En los resultados se puede apreciar como el modelo convolucional es superior en todas las métricas a excepción de *recall* donde el modelo LSTM es quien es superior. Sin embargo, su *precision* es la más baja lo cual sugiere que el modelo tiende a clasificar con frecuencia la clase positiva, incluso cuando no corresponde.

Modelo	Exactitud	Precision	Recall	F1-score	PR AUC	ROC AUC
CNN2D_AE_Q1_Temp	0.857	0.247	0.368	0.296	0.282	0.736
ResNet_AE_Q1_Temp	0.797	0.178	0.413	0.249	0.210	0.706
LSTM_AE_Temp	0.757	0.168	0.506	0.253	0.177	0.688

Tabla 5.4: Valor obtenido para cada una de las métricas sobre el conjunto de test para los experimentos variando el tipo de arquitectura.

Siguiendo, se procede a calcular qué valor se obtiene de *precision* en diferentes umbrales. Como ya se ha mencionado, por un tema de capacidad operativa no es posible inspeccionar todo el universo de clientes, sino que se debe priorizar sobre aquellos donde se sospecha más. Se evaluaron los umbrales 1 %, 5 %, 10 % y 15 %, los cuales representan 99, 497, 995 y 1492 inspecciones respectivamente. Los resultados se puede ver en Tabla 5.5. De la misma se desprende que el modelo basado en redes convolucionales obtienen un mejor valor de *precision@k* sobre el resto.

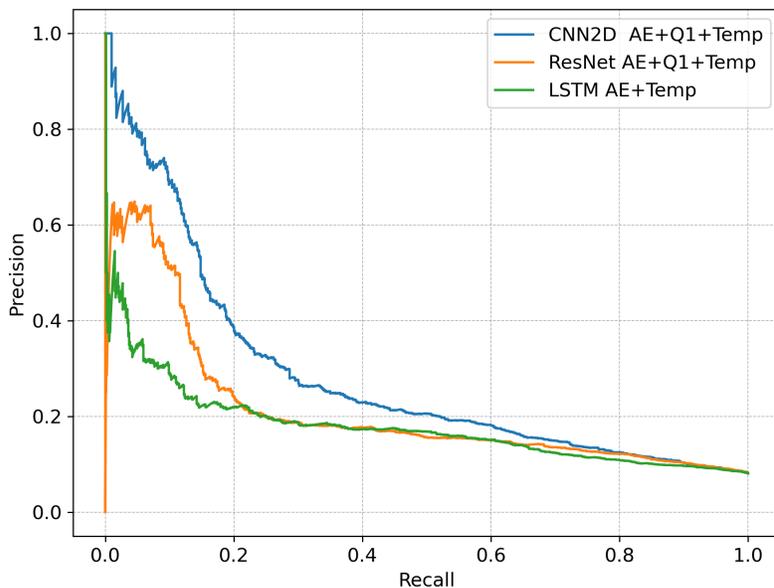


Figura 5.4: Curvas de *precision-recall* para los distintos tipos de arquitecturas sobre el conjunto de test.

Modelo	P@1 % 99 insp	P@5 % 497 insp	P@10 % 995 insp	P@15 % 1492 insp
CNN2D_AE_Q1_Temp	0.737	0.356	0.263	0.225
ResNet_AE_Q1_Temp	0.596	0.278	0.202	0.181
LSTM_AE_Temp	0.343	0.231	0.201	0.186

Tabla 5.5: Valor obtenido para cada una de las métricas p@k sobre el conjunto de test para los experimentos variando el tipo de arquitectura.

5.3. Experimentos variando la cantidad de series temporales a utilizar

Con el objetivo de entender qué impacto tuvo el agregado de las series temporales de energía reactiva y temperatura ambiente, se estudia dicho efecto sobre el modelo convolucional.

Para llevarlo a cabo, se realizan dos nuevas búsquedas de hiperparámetros sobre el mismo espacio de valores definidos en Tabla 5.1. Una búsqueda corresponde al conjunto de datos que solamente contiene la serie temporal de energía activa, y la otra búsqueda corresponde al conjunto de datos con energías activa y reactiva. Las arquitecturas resultantes son una variante de la que considera las 3 series temporales (ver Figura 5.2).

En la Figura 5.6, se aprecia que la inclusión de la energía reactiva parece deteriorar el modelo, aunque esta situación cambia cuando se añade la temperatura ambiente.

5.3. Experimentos variando la cantidad de series temporales a utilizar

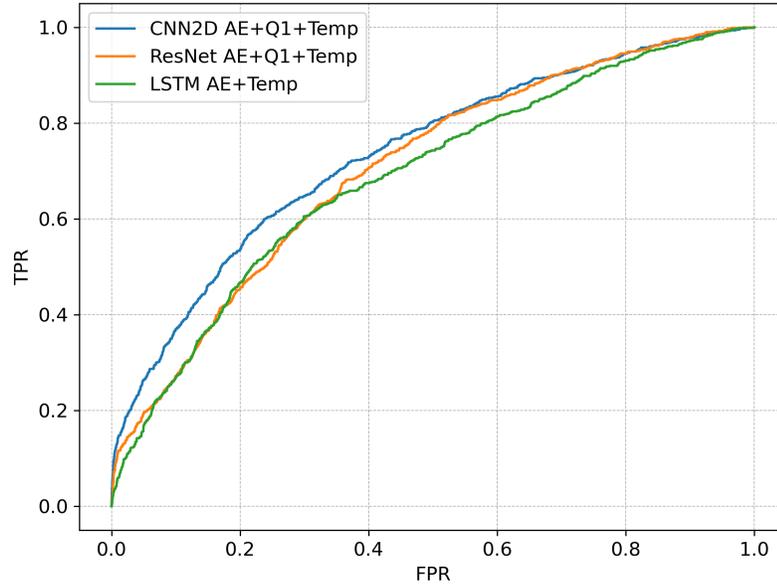


Figura 5.5: Curvas de ROC para los distintos tipos de arquitecturas sobre el conjunto de test.

En la Tabla 5.6 se puede ver el valor obtenido para cada una de las métricas. En lo que refiere a Exactitud, *Precision* y *Recall* se define como umbral de clasificación aquel que maximiza *F1-score*. Resulta interesante observar que, aunque las diversas arquitecturas logran valores similares de *F1-score*, la incorporación de nueva información conduce a un mejor rendimiento en métricas que no dependen del umbral, como PR AUC y ROC AUC.

Modelo	Exactitud	Precision	Recall	F1-score	PR AUC	ROC AUC
CNN2D_AE	0.874	0.275	0.336	0.303	0.265	0.722
CNN2D_AE.Q1	0.854	0.246	0.384	0.300	0.251	0.727
CNN2D_AE.Q1.Temp	0.857	0.247	0.368	0.296	0.282	0.736

Tabla 5.6: Valor obtenido para cada una de las métricas sobre el conjunto de test para redes de convolución variando las series consideradas.

El incremento de desempeño al usar esta información dependerá del diseño de la campaña de inspección, si la estadística de estos datos es representativa de todos los clientes, inspeccionar al 1% de los clientes de UTE implica una campaña del orden de las mil inspecciones. En este caso la diferencia de precisión sería de 10 puntos porcentuales como se puede ver en la Tabla 5.7.

Por lo tanto, se puede concluir que la inclusión en conjunto de la energía reactiva y la temperatura ambiente mejora el rendimiento del modelo. A nivel general, estas mejoras se observan sobre las métricas PR AUC y ROC AUC, siendo PR AUC la métrica más importante a lo largo de esta tesis. Es importante destacar que incluso un incremento del 1.7% en la métrica PR AUC entre el modelo que utiliza una sola serie y el que incorpora las tres series, puede ser considerable en escenarios con un elevado

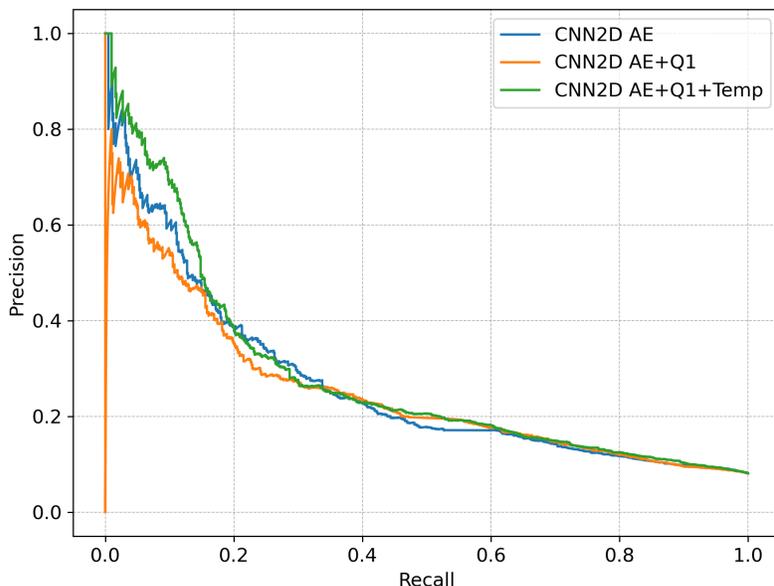


Figura 5.6: Curvas de *precision-recall* para el conjunto de datos de test y modelos basados en redes convolucionales variando la cantidad de series que considera cada uno.

Modelo	P@1 % 99 insp	P@5 % 497 insp	P@10 % 995 insp	P@15 % 1492 insp
CNN2D_AE	0.636	0.360	0.274	0.221
CNN2D_AE_Q1	0.566	0.336	0.259	0.227
CNN2D_AE_Q1_Temp	0.737	0.356	0.263	0.225

Tabla 5.7: Valor obtenido para cada una de las métricas $p@k$ sobre el conjunto de test para cada una de las métricas sobre el conjunto de test para redes de convolución variando las series consideradas.

número de inspecciones, lo que resalta las mejoras efectivas logradas con el modelo.

5.4. Experimentos aumentando el largo de las series temporales

Con el objetivo de mejorar los resultados y tomando en cuenta que se dispone de mediciones de energías desde el momento en que queda instalado el medidor inteligente hasta la actualidad, surge la idea de probar qué sucede cuando se amplía el historial de mediciones de tres a doce meses. Esto permite explorar cómo la inclusión de datos a más largo plazo afecta la capacidad predictiva de los modelos. Al considerar doce meses, se tendrán presentes las variaciones estacionales de verano e invierno en simultáneo para cada cliente, lo cual puede ayudar al modelo en la detección de NTL. Si bien se vio en la Sección 4.2.1 que la disponibilidad de datos anuales para este conjunto de datos se reduce bastante, se cree que aquellas inspecciones que si tienen presente las

5.4. Experimentos aumentando el largo de las series temporales

variaciones, pueden aportar conocimiento a los modelos.

En estos experimentos, se opta por reevaluar las diversas arquitecturas utilizadas. Específicamente, en el caso de las arquitecturas ResNet y LSTM, que procesan las series temporales en forma de vectores unidimensionales, se pretende determinar si un incremento en la cantidad de datos conduce a una mejora en su rendimiento.

Se lanzan tres nuevas búsquedas de hiperparámetros, una por cada tipo de arquitectura y variando sobre los mismos espacios de valores ya definidos en Tablas 5.1, 5.2, 5.3.

Evaluando el impacto de aumentar el historial de medidas, se muestra en la Figura 5.7 las curvas de *precision-recall* vinculadas a mismas arquitecturas. A nivel general no es posible concluir que el efecto sea el mismo para todos los tipos de arquitecturas. Para las redes convolucionales, el desempeño empeora nivel de curva cuando se amplía el historial. Para las redes residuales, el desempeño mejora cuando se amplía el historial. Y en el caso de las redes recurrentes, se observó que al aumentar el historial el desempeño baja drásticamente para casi todos los umbrales. Un motivo de esto puede ser que la secuencia de doce meses es demasiado larga y la red no está siendo capaz de recordar información valiosa para una secuencia tan larga.

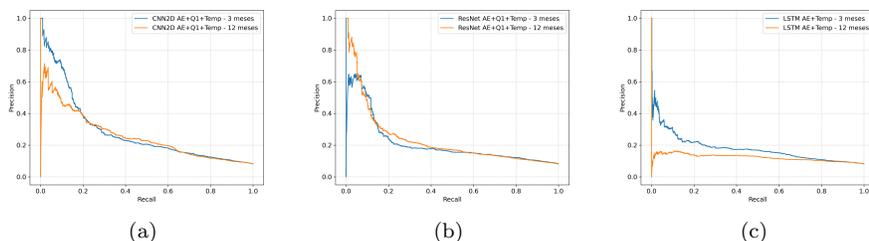


Figura 5.7: Curvas de *precision-recall* para los tres tipos de arquitecturas para evaluar el impacto del historial de mediciones. En la Figura (a) se presentan los modelos basados en redes convolucionales, en la Figura (b) los modelos basados en redes residuales y en la Figura (c) los modelos basados en redes recurrentes. En color azul siempre se grafican los modelos con tres meses de historial y en naranja los modelos con doce meses de historial.

En la Tabla 5.8 se puede ver el valor obtenido para cada una de las métricas. En lo que refiere a Exactitud, *Precision* y *Recall* se define como umbral de clasificación aquel que maximiza *F1-score*. El efecto de ampliar el historial varió entre las diferentes arquitecturas. Si se pone el foco en las métricas PR AUC y ROC AUC, para las redes convolucionales, extender el historial no presentó ventajas. Para las redes residuales, si generó ganancias un poco más significativas, logrando pasar de 0.210 a 0.230 de PR AUC. Y para las redes residuales, tuvo un efecto negativo, siendo inferior para todas las métricas.

Modelo	Historial	Exactitud	Precision	Recall	F1-score	PR AUC	ROC AUC
CNN2D_AE_Q1_Temp	3 meses	0.857	0.247	0.368	0.296	0.282	0.736
CNN2D_AE_Q1_Temp	12 meses	0.837	0.241	0.466	0.318	0.254	0.735
ResNet_AE_Q1_Temp	3 meses	0.797	0.178	0.413	0.249	0.210	0.706
ResNet_AE_Q1_Temp	12 meses	0.837	0.208	0.355	0.262	0.231	0.706
LSTM_AE_Temp	3 meses	0.757	0.168	0.506	0.253	0.177	0.688
LSTM_AE_Temp	12 meses	0.700	0.131	0.480	0.206	0.122	0.636

Tabla 5.8: Valor obtenido para cada una de las métricas sobre el conjunto de test variando el tipo de arquitectura y el historial de mediciones.

Capítulo 5. Experimentos con redes neuronales profundas

En la Tabla 5.9 se pueden ver los valores obtenidos para la métrica $precision@k$. Si se compara el efecto sobre mismo tipo de arquitectura se puede volver a corroborar que solo para LSTM existe una clara ventaja del modelo de tres meses sobre el de doce meses. A pesar de haber ampliado el historial, los modelos ResNet y LSTM siguen sin alcanzar el desempeño que obtienen los modelos convolucionales.

Modelo	Historial	P@1 % 99 insp	P@5 % 497 insp	P@10 % 995 insp	P@15 % 1492 insp
CNN2D_AE_Q1_Temp	3 meses	0.737	0.356	0.263	0.225
CNN2D_AE_Q1_Temp	12 meses	0.566	0.348	0.277	0.241
ResNet_AE_Q1_Temp	3 meses	0.596	0.278	0.202	0.181
ResNet_AE_Q1_Temp	12 meses	0.596	0.292	0.231	0.200
LSTM_AE_Temp	3 meses	0.343	0.231	0.201	0.186
LSTM_AE_Temp	12 meses	0.131	0.151	0.146	0.133

Tabla 5.9: Valor obtenido para cada una de las métricas $p@k$ sobre el conjunto de test para todas las arquitecturas, variando el historial de mediciones.

Como conclusión, es importante destacar que el impacto del aumento en el historial de mediciones varía según la arquitectura utilizada, generando un impacto positivo solamente para la arquitectura ResNet en términos de PR AUC. A pesar de la expectativa inicial de que la inclusión de las variaciones estacionales de cada cliente tendría un impacto positivo, es posible que este efecto no se haya visto debido a la disponibilidad limitada de datos anuales. Aún así, se observa claramente que las redes convolucionales continúan mostrando un rendimiento superior en comparación con las demás arquitecturas. En cuanto a las redes LSTM, su desempeño más bajo genera dudas sobre si el espacio de búsqueda de hiperparámetros fue adecuado para la secuencia de doce meses.

5.5. Evaluación de tiempos de optimización y entrenamiento de los modelos

En esta sección se evalúan otros aspectos vinculados a los tiempos empleados tanto en optimización como en entrenamiento de los modelos. Todo el trabajo de esta tesis fue realizado haciendo uso de los recursos del Centro Nacional de Supercomputación [57] (cluster.uy).

Para optimizar y entrenar las distintas arquitecturas se hizo uso de GPUs Nvidia Tesla P100 con 12Gb de memoria disponibles en cluster.uy. Referente a los tiempos empleados, se puede apreciar en Tabla 5.10 como aumenta considerablemente el tiempo en ambas etapas a medida que se aumenta el historial de mediciones. Sin embargo, el aumento gradual en cantidad de series en lo que refiere a modelos CNN2D no ocasionó aumentos en los tiempos en ninguna de las etapas, lo cual posiblemente se deba a que paraleliza los cálculos de los distintos canales. Por otro lado, las redes LSTM son las que más tiempo requieren en ambas etapas si se las compara con otras arquitecturas que consideran misma cantidad de series e historial. Esta demora considerable se cree que se debe a la complejidad computacional que tienen las neuronas de este tipo de arquitectura.

5.5. Evaluación de tiempos de optimización y entrenamiento de los modelos

Modelo	Historial	Tiempo búsqueda de hiperparámetros	Tiempo entrenamiento
CNN2D_AE	3 meses	13h 45m	27m
CNN2D_AE_Q1	3 meses	13h 21m	36m
CNN2D_AE_Q1_Temp	3 meses	13h 17m	51m
CNN2D_AE_Q1_Temp	12 meses	1d 13h 14m	4h 30m
ResNet_AE_Q1_Temp	3 meses	8h 21m	1h 07m
ResNet_AE_Q1_Temp	12 meses	1d 3h 18m	5h 12m
LSTM_AE_Temp	3 meses	15h 12m	1h 49m
LSTM_AE_Temp	12 meses	2d 13h 53m	9h 8m

Tabla 5.10: Tiempo empleado en búsqueda de hiperparámetros y entrenamiento para todos los experimentos realizados, variando el tipo de arquitectura, las series consideradas y el historial de mediciones.

Esta página ha sido intencionalmente dejada en blanco.

Capítulo 6

Conclusiones

6.1. Conclusiones

El presente trabajo de investigación ha abordado el problema de cómo detectar las NTL usando redes neuronales profundas que se alimentan de series temporales multivariadas.

Se analizó el impacto de integrar datos de temperatura ambiente en los algoritmos de detección, lo cual requirió la recopilación de datos de diferentes fuentes y un arduo trabajo de preparación de los datos, incluyendo el diseño de algoritmos para la imputación de datos faltantes. Como resultado, se encontró que el comportamiento de los clientes cambia según las variaciones de temperatura, con un aumento en el consumo de energía durante tanto los periodos de calor en verano como en los de frío en invierno. Al estudiar la correlación entre la temperatura y el consumo, se pudo observar que la distribución de correlaciones entre clientes normales y aquellos que cometen fraude, son diferentes y esto se confirmó mediante un test estadístico. Estos resultados impulsaron a probar si la nueva característica de correlación calculada podía mejorar la detección de NTL. Mediante la creación de clasificadores *random forest* se encontró que la nueva característica no solo mejora el rendimiento del clasificador, sino que también es una de las principales influyentes en sus decisiones.

Se estudió cómo procesar las diferentes series temporales usando modelos de aprendizaje profundo. Se trabajó con las series en diferentes formatos, para las arquitecturas convolucionales se crearon imágenes con varios canales y para las redes residuales y recurrentes se las trató en su formato tradicional como vectores unidimensionales.

A su vez, fue necesario examinar de qué forma era conveniente normalizar los datos y para ello se estudiaron dos alternativas: *Normalize* y *z-score*. A partir de los resultados obtenidos se comprobó que el método *z-score* provoca una mejora considerable del rendimiento del modelo y como explicación de este fenómeno se entiende que el hecho de forzar a que cada muestra tenga una norma igual a 1, si bien no se pierde la forma de la curva, sí se está perdiendo información acerca del valor real, se pierde noción acerca de qué tan más grande o más chico es un valor con respecto al resto de las muestras. Esa pérdida de información se entiende que es la causante de la caída del desempeño.

A lo largo de este trabajo, se probaron un total de 13 modelos de aprendizaje profundo, variando entre ellos el tipo de arquitectura, la cantidad de series temporales y el historial de mediciones considerado. Se realizaron experimentos con diversas arquitecturas de redes neuronales, incluyendo redes convolucionales, residuales y recurrentes. De los experimentos se concluye que:

Capítulo 6. Conclusiones

El tipo de red convolucional es el que mejor rendimiento obtiene frente al resto de las arquitecturas para todos los umbrales. Esto pone de manifiesto el potencial de estas redes para identificar patrones en imágenes, sugiriendo que tratar las series como imágenes resulta más ventajoso que considerarlas como vectores unidimensionales para la detección de NTL.

En cuanto a la incorporación de la serie temporal de energía reactiva, se observó que empeora el desempeño en comparación con el modelo que utiliza solamente la energía activa. Esta situación se revierte cuando se incorpora en conjunto la energía reactiva y la temperatura ambiente, logrando un aumento del 1.7% en la métrica PR AUC entre el modelo que utiliza una sola serie y el que incorpora las tres series, lo cual puede ser considerable en escenarios con un elevado número de inspecciones, resaltando así las mejoras efectivas logradas con el modelo.

En relación a ampliar el historial de mediciones a considerar, el resultado varía según el tipo de arquitectura, generando un impacto positivo solamente para la arquitectura ResNet en términos de PR AUC. Aunque inicialmente se esperaba que incorporar las variaciones estacionales de cada cliente generara un impacto positivo, es probable que este resultado no se haya concretado debido a la disponibilidad restringida de datos anuales. Este experimento siguió demostrando la superioridad de las redes convolucionales frente al resto en ambos historiales.

6.2. Trabajo Futuro

El estudio realizado hasta la fecha ha proporcionado una visión experimental sobre la detección de NTL, centrándose particularmente en el análisis de series temporales multivariadas con modelos de aprendizaje profundo. Sin embargo, este trabajo es solo el primer paso en un campo de investigación en constante evolución, y hay varias áreas que podrían explorarse en futuros estudios para mejorar aún más la detección y mitigación de NTL. En esta sección, se presentan algunas posibles líneas de trabajo a seguir en el futuro, con el objetivo de ampliar nuestro entendimiento sobre este fenómeno y seguir desarrollando estrategias para abordarlo.

Líneas de trabajo para continuar:

- Evaluar el desempeño en campo del modelo CNN2D_AE_Q1_Temp con historial de tres meses.
- Determinar si los modelos desarrollados en esta tesis son complementarios en términos de detección, es decir, si todos están identificando el mismo universo de fraudes o no.
- Dado el crecimiento continuo de la base de inspecciones y que casi totalidad de los clientes posee un medidor inteligente, se hace imprescindible desarrollar un módulo que alimente los modelos utilizando un generador de Python. Esto posibilitará el entrenamiento de los modelos con grandes volúmenes de datos.
- Incorporar la nueva característica de correlación a los modelos ya existentes y utilizados por UTE para evaluar su desempeño en campo.
- Dado el éxito que tuvieron las redes convolucionales frente a los otros tipos de arquitecturas, se puede explorar otras representaciones de los datos de series temporales como imágenes, como por ejemplo Gramian Angular Field (GAF). Esta técnica genera una imagen que preserva las características originales de los datos y añade correlaciones temporales.
- Estudiar si es factible la incorporación de una nueva serie temporal con datos de microbalances.

6.2. Trabajo Futuro

- Ante nuevas búsquedas de hiperparámetros se puede modificar la estrategia. Por ejemplo, en lo que respecta a redes convolucionales se puede hacer que la cantidad de filtros sean independientes entre las capas y que a medida que se vaya profundizando en la red, se vaya intentando con valores más grandes.
- Estudiar alternativas de cómo utilizar las series de energías en resolución quinceminutal y la serie de temperatura ambiente en resolución horaria.
- En caso de disponer datos del Censo 2023 en Uruguay, se podría seguir mejorando el análisis de correlación entre temperatura ambiente y consumo, ya que se dispondrá más información acerca de las características principales de las viviendas. La información adicional acerca de la construcción o métodos de calefacción/refrigeración de las mismas, permitiría un análisis más profundo que involucre estudiar aspectos tales como: ¿cuál es el retraso que hay que hacer para que las series alcancen su máxima correlación? y ver si esa información se condice con las características y/o equipamiento de la vivienda o si resulta sospechosa para la detección de NTL.

Esta página ha sido intencionalmente dejada en blanco.

Apéndice A

Experimento con Fully Convolutional Neural Network

Un experimento realizado que no tuvo éxito fue con la arquitectura Fully Convolutional Neural Networks (FCN) tomada del artículo [27]. Dicha arquitectura presentó muy buenos resultados para una gran variedad de conjuntos de datos con series temporales multivariadas. Por lo tanto, se decidió hacer una prueba con el conjunto de datos compuesto por medidas horarias y un historial de tres meses, se puede visualizar la arquitectura en la Figura A.1.

La arquitectura se mantuvo tal cual está descrita en el artículo y para la búsqueda de hiperparámetros solamente se varió la tasa de aprendizaje en el espacio $[1e^{-5}, \dots, 1e^{-3}]$. Lo que se observó durante la búsqueda de hiperparámetros es que los resultados en validación eran muy inestables. En la Figura A.2 se puede ver cómo variaban las métricas de PR AUC tanto para el conjunto de entrenamiento y validación. También se puede observar cómo varía el *loss* tanto para el conjunto de entrenamiento y validación en la Figura A.3. Los resultados de la búsqueda dieron que la tasa de aprendizaje que mejor se ajusta al conjunto de entrenamiento es $3.5e^{-5}$ y en la Figura A.4 se puede apreciar como este entrenamiento puntual los gráficos oscilan significativamente sobre el conjunto de validación, lo cual indica que el modelo no está convergiendo de manera consistente hacia una solución óptima. Esto puede ser causado por una serie de factores, como por ejemplo: tasa de aprendizaje demasiado alta o baja, un tamaño de lote inapropiado, una arquitectura de red inadecuada, entre otros. La estabilidad del entrenamiento es importante para garantizar que el modelo pueda generalizar bien a nuevos datos y producir resultados consistentes.

Los resultados obtenidos para el conjunto de test se puede observar en la Tabla A.1.

Modelo	Exactitud	Precision	Recall	PR AUC	ROC AUC
FCN	0.748	0.163	0.507	0.221	0.713

Tabla A.1: Valor obtenido para cada una de las métricas sobre el conjunto de test para la arquitectura FCN con resolución horaria e historial de tres meses.

Apéndice A. Experimento con Fully Convolutional Neural Network

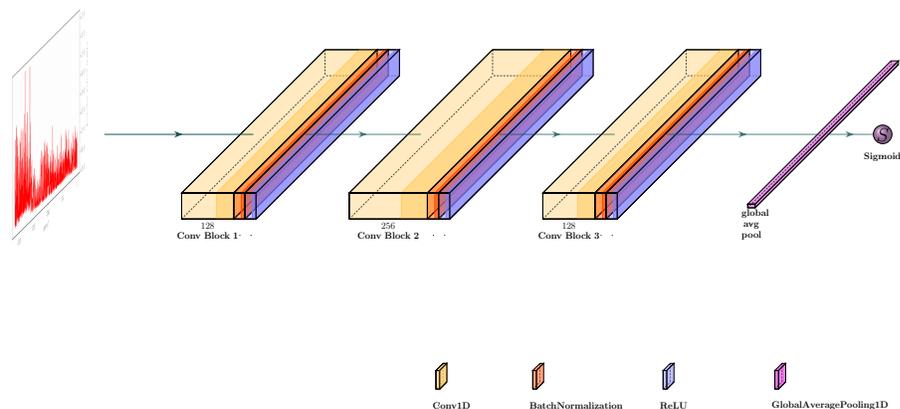


Figura A.1: Arquitectura FCN para detección de NTL, tomada del artículo [27] adaptada para procesar serie temporal de energía activa en resolución horaria e historial de tres meses.

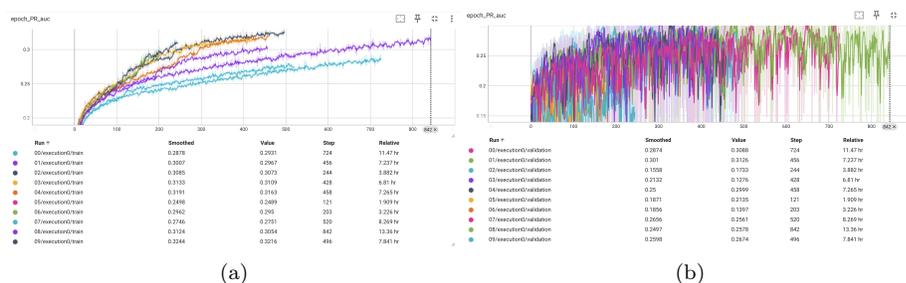


Figura A.2: Curvas de *precision-recall* para el modelo FCN durante la búsqueda de hiperparámetros. En la Figura (a) las curvas corresponden al conjunto de entrenamiento. En la Figura (b) las curvas corresponden al conjunto de validación.

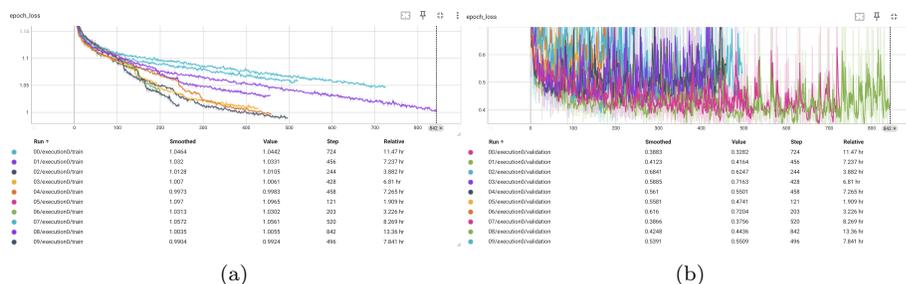


Figura A.3: Curvas de *loss* para el modelo FCN durante la búsqueda de hiperparámetros. En la Figura (a) las curvas corresponden al conjunto de entrenamiento. En la Figura (b) las curvas corresponden al conjunto de validación.

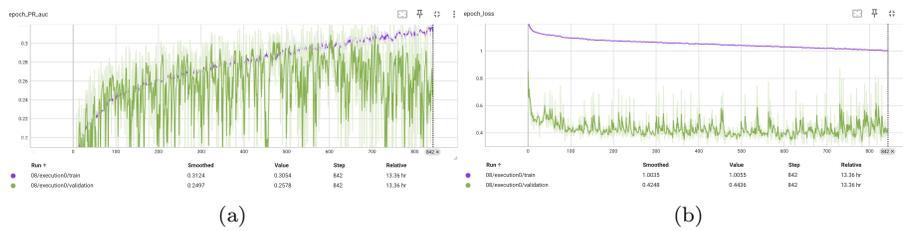


Figura A.4: Curvas de *precision-recall* y *loss* para el modelo FCN con tasa de aprendizaje $3.5e^{-5}$, producto de la búsqueda de hiperparámetros. En la Figura (a) se grafica *precision-recall* y en la Figura (b) se grafica *loss*.

Esta página ha sido intencionalmente dejada en blanco.

Apéndice B

Experimento con serie de energía activa en resolución quinceminutal

A pesar de que en este trabajo se definió trabajar con medidas en resolución horaria porque la serie de temperaturas se encuentra en resolución horaria y porque en esta resolución hay menos datos faltantes, se decide probar qué sucede cuando se trabaja con resolución quinceminutal.

En este experimento se considera solamente la serie de energía activa y se utiliza una red neuronal convolucional 2D. Para la base de nombre 50k_h (definida en B.1) se considera solamente los últimos tres meses de mediciones. A pesar de que ambas bases deberían tener el mismo tamaño, la razón por la cual la base 50k_h es más chica se debe a que no se posee un historial de doce meses para todos los medidores. Esto se pudo comprobar cuando se hizo el análisis del largo de las series temporales en Sección 4.2.1. Se realizan dos búsquedas de hiperparámetros, una para cada conjunto de datos y en ambas se explora sobre el mismo espacio (ver Tabla 5.1). El tiempo de búsqueda para la base 50k_15min fue de 1d 02h 24m, mientras que para la base 50k_h fue de 12h 12m, es decir, le la mitad del tiempo. En la Figura B.1 se puede ver la arquitectura resultante para ambas resoluciones.

Nombre base	#Clientes	Resolución	#Meses	Tamaño
50k_15m	50432	quinceminutal	3 meses	13 Gb
50k_h	50432	horaria	12 meses	11 Gb

Tabla B.1: Detalles de los datos de medidores extraídos de HBase

De la misma forma que se ha hecho a lo largo de esta tesis, la métrica elegida para evaluar el desempeño es PR AUC y el modelo elegido para probar en test es aquel que obtenga el máximo valor de esta métrica para el conjunto de validación. Los resultados obtenidos en el conjunto de test se pueden observar en la Tabla B.2. En lo que refiere a Exactitud, Precision, Recall y F1-score se define como umbral de clasificación aquel que maximiza F1-score. Se puede concluir que el modelo que trabaja con resolución quinceminutal presenta un mejor desempeño en las métricas de AUC y *recall*. En la Figura B.2 se pueden apreciar las curvas de *precision-recall* y ROC para entender mejor cómo es el desempeño de ambos modelos. En cuanto a las curvas de *precision-recall* se desprende que la diferencia de rendimiento se encuentra para umbrales bajos de *recall*. Esto quiere decir que cuando el objetivo se focaliza en detectar menos del

Apéndice B. Experimento con serie de energía activa en resolución quinceminutal

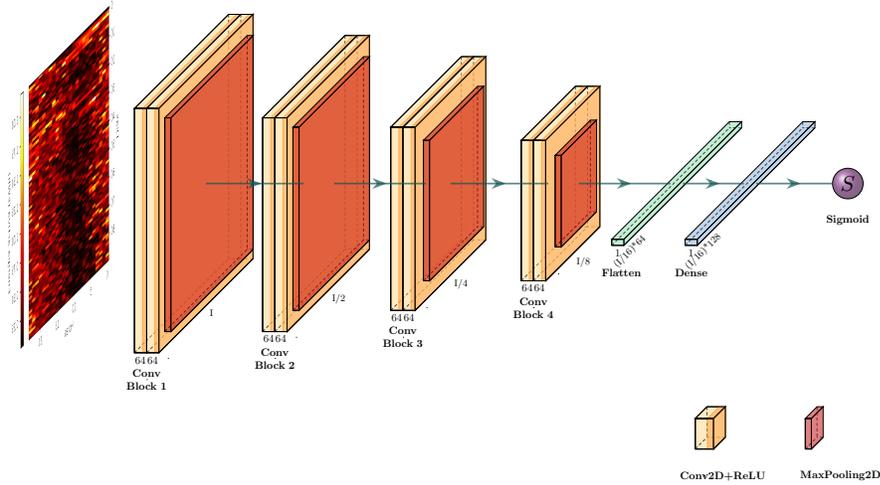


Figura B.1: Arquitectura CNN2D para la detección de NTL, resultante de la búsqueda de hiperparámetros para datos con resolución quinceminutal y horaria.

30% de los clientes con fraude, el modelo que trabaja con medidas quinceminutales es quien tendrá más *precision* al momento de clasificar. En cuanto a la curvas de ROC AUC, ambos modelos se encuentran muy semejantes, lo cual significa que la capacidad de ambos para distinguir entre ambas clases a clasificar es muy similar.

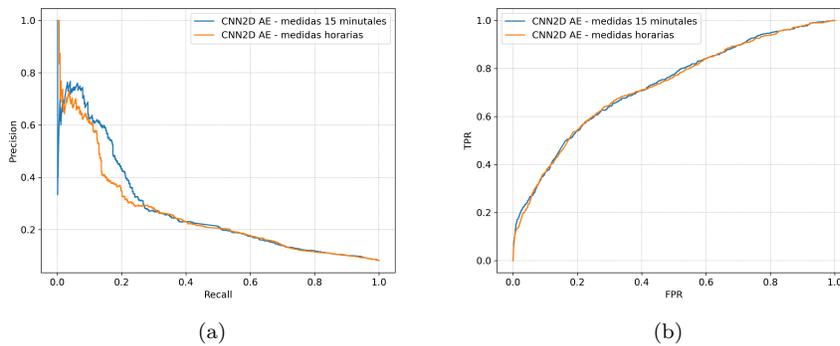


Figura B.2: La Figura (a) muestra las curvas de *precision-recall* para el conjunto de test de los modelos CNN2D con resoluciones quinceminutal y horaria. La Figura (b) muestra las curvas de ROC para el conjunto de test de los modelos CNN2D con resoluciones quinceminutal y horaria.

Por lo tanto, tal como se sospechaba, el modelo que utiliza datos con mayor resolución es el que obtiene un mejor desempeño.

Modelo	Exactitud	Precision	Recall	F1-score	PR AUC	ROC AUC
CNN2D + quinceminutal	0.834	0.225	0.425	0.294	0.274	0.729
CNN2D + horario	0.863	0.256	0.357	0.298	0.259	0.726

Tabla B.2: Valor obtenido para cada una de las métricas sobre el conjunto de test para la arquitectura CNN2D con resolución quinceminutal y horaria.

Esta página ha sido intencionalmente dejada en blanco.

Referencias

- [1] Energías renovables en uruguay. [https://www.uruguayxxi.gub.uy/uploads/informacion/79870b5679e4f9634944f6b8daca8ee6c3d45df.pdf#:~:text=MATRIZ%20EL%C3%89CTRICA%3A,1a%20tradicional%20hidroel%C3%A9ctrica%20\(45%25\)](https://www.uruguayxxi.gub.uy/uploads/informacion/79870b5679e4f9634944f6b8daca8ee6c3d45df.pdf#:~:text=MATRIZ%20EL%C3%89CTRICA%3A,1a%20tradicional%20hidroel%C3%A9ctrica%20(45%25)) ., 2023. Online; accedido 16 Abril 2024.
- [2] Iea heatpumps. <https://www.iea.org/energy-system/buildings/heat-pumps>, 2024. Online; accedido 12 Febrero 2024.
- [3] Worldenergyoutlook2023. <https://iea.blob.core.windows.net/assets/86ede39e-4436-42d7-ba2a-edf61467e070/WorldEnergyOutlook2023.pdf>, 2024. Online; accedido 12 Febrero 2024.
- [4] Emiliano Acevedo, Pablo Massaferro, Alicia Fernández, Alexander Martins, and Gonzalo Caudullo. Fraud detection using event logs with lstm and gradient boosting. In *2023 IEEE Power & Energy Society Innovative Smart Grid Technologies Conference (ISGT)*, pages 1–5. IEEE, 2023.
- [5] International Energy Agency. Energy system - buildings - heating. <https://www.iea.org/energy-system/buildings/heating>, ltimoacceso:30Marzode2024.
- [6] International Energy Agency. Space cooling. <https://www.iea.org/energy-system/buildings/space-cooling>, ltimoacceso:30Marzode2024.
- [7] International Energy Agency. Electricity market report 2023. Technical report, 2023.
- [8] Paula Bezerra, Fabio da Silva, Talita Cruz, Malcolm Mistry, Eveline Vasquez-Arroyo, Leticia Magalar, Enrica De Cian, André FP Lucena, and Roberto Schaefer. Impacts of a warmer world on space cooling demand in brazilian households. *Energy and Buildings*, 234:110696, 2021.
- [9] Filippo Maria Bianchi, Simone Scardapane, Sigurd Løkse, and Robert Jenssen. Reservoir computing approaches for representation and classification of multivariate time series. *IEEE transactions on neural networks and learning systems*, 32(5):2169–2179, 2020.
- [10] Leo Breiman. Bagging predictors. *Machine learning*, 24:123–140, 1996.
- [11] Yanping Chen, Eamonn Keogh, Bing Hu, Nurjahan Begum, Anthony Bagnall, Abdullah Mueen, and Gustavo Batista. The ucr time series classification archive, July 2015. www.cs.ucr.edu/~eamonn/time_series_data/.
- [12] Organización Latinoamericana de Energía. Sistema de información energética de latinoamérica y el caribe. <https://sielac.olade.org/>, Último acceso: 26 Julio de 2023.
- [13] Matías Di Martino, Federico Decia, Juan Molinelli, and Alicia Fernández. Improving electric fraud detection using class imbalance strategies. In *ICPRAM (2)*, pages 135–141, 2012.

Referencias

- [14] Matías Di Martino, Guzmán Hernández, Marcelo Fiori, and Alicia Fernández. A new framework for optimal classifier design. *Pattern Recognition*, 46(8):2249–2255, 2013.
- [15] Kerui Du, Ying Yu, and Chu Wei. Climatic impact on china’s residential electricity consumption: Does the income level matter? *China Economic Review*, 63:101520, 2020.
- [16] Bradley Efron. Bootstrap methods: another look at the jackknife. In *Breakthroughs in statistics: Methodology and distribution*, pages 569–593. Springer, 1992.
- [17] Chukwuemeka Chinonso Emenekwe and Nnaemeka Vincent Emodi. Temperature and residential electricity demand for heating and cooling in g7 economies: A method of moments panel quantile regression approach. *Climate*, 10(10):142, 2022.
- [18] Nnaemeka Vincent Emodi, Taha Chaiechi, and ABM Rabiul Alam Beg. The impact of climate change on electricity demand in australia. *Energy & Environment*, 29(7):1263–1297, 2018.
- [19] Oliver Faust, Yuki Hagiwara, Tan Jen Hong, Oh Shu Lih, and U Rajendra Acharya. Deep learning for healthcare applications based on physiological signals: A review. *Computer methods and programs in biomedicine*, 161:1–13, 2018.
- [20] Kuniyuki Fukushima. Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. *Biological cybernetics*, 36(4):193–202, 1980.
- [21] Sravan Kumar Gunturi and Dipu Sarkar. Ensemble machine learning models for the detection of energy theft. *Electric Power Systems Research*, 192:106904, 2021.
- [22] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.
- [23] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.
- [24] Wenjie Hu, Yang Yang, Jianbo Wang, Xuanwen Huang, and Ziqiang Cheng. Understanding electricity-theft behavior via multi-source data. In *Proceedings of The Web Conference 2020*, pages 2264–2274, 2020.
- [25] David H Hubel and Torsten N Wiesel. Receptive fields, binocular interaction and functional architecture in the cat’s visual cortex. *The Journal of physiology*, 160(1):106, 1962.
- [26] Instituto Uruguayo de Meteorología. <https://www.inumet.gub.uy/>, Último acceso: 25 de Agosto de 2023.
- [27] Hassan Ismail Fawaz, Germain Forestier, Jonathan Weber, Lhassane Idoumghar, and Pierre-Alain Muller. Deep learning for time series classification: a review. *Data mining and knowledge discovery*, 33(4):917–963, 2019.
- [28] Herbert Jaeger and Harald Haas. Harnessing nonlinearity: Predicting chaotic systems and saving energy in wireless communication. *science*, 304(5667):78–80, 2004.
- [29] Jieyi Kang and David M Reiner. What is the effect of weather on household electricity consumption? empirical evidence from ireland. *Energy Economics*, 111:106023, 2022.
- [30] Amir Kavousian, Ram Rajagopal, and Martin Fischer. Determinants of residential electricity consumption: Using smart meter data to examine the effect of climate, building characteristics, appliance stock, and occupants’ behavior. *Energy*, 55:184–194, 2013.

- [31] Juan Pablo Kosut, Fernando Santomauro, Andrés Jorysz, Alicia Fernández, Federico Lecumberry, and Fernanda Rodríguez. Abnormal consumption analysis for fraud detection: Ute-udelar joint efforts. In *2015 IEEE PES Innovative Smart Grid Technologies Latin America (ISGT LATAM)*, pages 887–892. IEEE, 2015.
- [32] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25, 2012.
- [33] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.
- [34] Jianglong Li, Lisha Yang, and Houyin Long. Climatic impacts on energy consumption: Intensive and extensive margins. *Energy Economics*, 71:332–343, 2018.
- [35] Yuanzheng Li, Wenjing Wang, Yating Wang, Yashu Xin, Tian He, and Guosong Zhao. A review of studies involving the effects of climate change on the energy consumption for building heating and cooling. *International Journal of Environmental Research and Public Health*, 18(1):40, 2021.
- [36] Pablo Massaferró. *Detección de pérdidas no técnicas en redes eléctricas en un contexto de migración tecnológica y maximizando el retorno económico*. PhD thesis, Facultad de Ingeniería - UdeLaR, 2022.
- [37] Pablo Massaferró, J Matías Di Martino, and Alicia Fernández. Ntl detection: Overview of classic and dnn-based approaches on a labeled dataset of 311k customers. In *2021 IEEE Power & Energy Society Innovative Smart Grid Technologies Conference (ISGT)*, pages 1–5. IEEE, 2021.
- [38] Pablo Massaferró, J Matías Di Martino, and Alicia Fernández. Fraud detection on power grids while transitioning to smart meters by leveraging multi-resolution consumption data. *IEEE Transactions on Smart Grid*, 13(3):2381–2389, 2022.
- [39] Pablo Massaferró, Henry Marichal, Matias Di Martino, Fernando Santomauro, Juan Pablo Kosut, and Alicia Fernandez. Improving electricity non technical losses detection including neighborhood information. In *2018 IEEE Power & Energy Society General Meeting (PESGM)*, pages 1–5. IEEE, 2018.
- [40] George M Messinis and Nikos D Hatziargyriou. Review of non-technical loss detection methods. *Electric Power Systems Research*, 158:250–266, 2018.
- [41] Iñigo Monedero, Félix Biscarri, Carlos León, Juan I Guerrero, Jesús Biscarri, and Rocío Millán. Using regression analysis to identify patterns of non-technical losses on power utilities. In *Knowledge-Based and Intelligent Information and Engineering Systems: 14th International Conference, KES 2010, Cardiff, UK, September 8-10, 2010, Proceedings, Part I 14*, pages 410–419. Springer, 2010.
- [42] Organización Meteorológica Mundial. Los indicadores del cambio climático alcanzaron niveles sin precedentes en 2023: Omm. <https://wmo.int/es/news/media-centre/los-indicadores-del-cambio-climatico-alcanzaron-niveles-sin-precedentes-en-2023-omm>, ltimoacceso:25Abrilde2024.
- [43] Organización Meteorológica Mundial. Los últimos ocho años han sido los más cálidos jamás registrados a nivel mundial. <https://wmo.int/es/media/news/los-ultimos-ocho-anos-han-sido-los-mas-calidos-jamas-registrados-nivel-mundial>, ltimoacceso:25Abrilde2024.
- [44] J Nagi, AM Mohammad, Keem Siah Yap, Sieh Kiong Tiong, and Syed Khaleel Ahmed. Non-technical loss analysis for detection of electricity theft using support

Referencias

- vector machines. In *2008 IEEE 2nd International Power and Energy Conference*, pages 907–912. IEEE, 2008.
- [45] Jawad Nagi, Keem Siah Yap, Sieh Kiong Tiong, Syed Khaleel Ahmed, and Malik Mohamad. Nontechnical loss detection for metered customers in power utility using support vector machines. *IEEE transactions on Power Delivery*, 25(2):1162–1171, 2009.
- [46] Energy Regulatory Commission of Ireland. Cer smart metering project—electricity customer behaviour trial, 2009-2010 [dataset]. irish social science data archive. <https://www.ucd.ie/issda/data/commissionforenergyregulationcer>, Último acceso: 01 Abril de 2024.
- [47] Open Weather. <https://openweathermap.org/>, Último acceso: 25 de Agosto de 2023.
- [48] Rajiv Punmiya and Sangho Choe. Energy theft detection using gradient boosting theft detector with feature engineering-based preprocessing. *IEEE Transactions on Smart Grid*, 10(2):2326–2329, 2019.
- [49] Lawrence R Rabiner and Bernard Gold. Theory and application of digital signal processing. *Englewood Cliffs: Prentice-Hall*, 1975.
- [50] Deepta Rajan and Jayaraman J Thiagarajan. A generative modeling approach to limited channel ecg classification. In *2018 40th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pages 2571–2574. IEEE, 2018.
- [51] Caio César Oba Ramos, André Nunes de Sousa, João Paulo Papa, and Alexandre Xavier Falcao. A new approach for nontechnical losses detection based on optimum-path forest. *IEEE Transactions on Power Systems*, 26(1):181–189, 2010.
- [52] David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams. Learning representations by back-propagating errors. *nature*, 323(6088):533–536, 1986.
- [53] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115:211–252, 2015.
- [54] Muhammad Salman Saeed, Mohd Wazir Mustafa, Nawaf N Hamadneh, Nawa A Alshammari, Usman Ullah Sheikh, Touqeer Ahmed Jumani, Saifulnizam Bin Abd Khalid, and Ilyas Khan. Detection of non-technical losses in power utilities—a comprehensive systematic review. *Energies*, 13(18):4727, 2020.
- [55] HCS Thom. Seasonal degree-day statistics for the united states. *Monthly Weather Review*, 80(9):143–147, 1952.
- [56] Munenori Uemura, Morimasa Tomikawa, Tiejun Miao, Ryota Souzaki, Satoshi Ieiri, Tomohiko Akahoshi, Alan K Lefor, Makoto Hashizume, et al. Feasibility of an ai-based measure of the hand motions of expert and novice surgeons. *Computational and mathematical methods in medicine*, 2018, 2018.
- [57] Cluster UY. Centro nacional de supercomputación. <https://cluster.uy/>, Último acceso: 28 Agosto de 2023.
- [58] Haojie Wang and Qingyan Chen. Impact of climate change heating and cooling energy use in buildings in the united states. *Energy and Buildings*, 82:428–436, 2014.
- [59] Zhiguang Wang and Tim Oates. Imaging time-series to improve classification and imputation. *arXiv preprint arXiv:1506.00327*, 2015.

Referencias

- [60] Open Weather. Accuracy and quality of weather data. <https://openweathermap.org/accuracy-and-quality>, Último acceso: 26 Julio de 2023.
- [61] Xiaowu Zou, Zidong Wang, Qi Li, and Weiguo Sheng. Integration of residual network and convolutional neural network along with various activation functions and global pooling for time series classification. *Neurocomputing*, 367:39–45, 2019.

Esta página ha sido intencionalmente dejada en blanco.

Índice de tablas

3.1. Promedio y desviación estándar para cada fuente de datos: Open Weather, estaciones convencionales INUMET y estaciones automáticas INUMET.	24
3.2. Valor de RECM obtenido al comparar datos de OW con estaciones convencionales y automáticas de INUMET.	25
3.3. Cantidades de inspecciones según el departamento realizadas por UTE entre Enero 2020 y Octubre 2022.	29
3.4. Las 10 características más importantes a la hora de clasificar instancias por el clasificador <i>random forest</i> . Para cada una se detalla numéricamente cual es el peso que tienen en la clasificación.	36
3.5. Las 10 características más influyentes a la hora de clasificar instancias por el clasificador <i>random forest</i> utilizando el conjunto definido en Sección 3.3.3 y además la nueva característica de máxima correlación. Para cada una se detalla numéricamente cual es el peso que tienen en la clasificación.	37
3.6. Resultados de los clasificadores <i>random forest</i> sobre el conjunto de test.	38
4.1. Total de muestras, total de muestras para cada clase (normal es negativa y fraude es positiva) y porcentaje de fraude con respecto al total de muestras, para los conjuntos de: entrenamiento, validación y test.	42
4.2. Valor obtenido para cada una de las métricas sobre el conjunto de test tanto para CNN2D con método de normalización Normalize y para CNN2D con método de estandarización <i>z-score</i> . Siendo este último el modelo que mejor rendimiento tiene a nivel de todas las métricas.	46
5.1. Espacio de búsqueda de hiperparámetros	50
5.2. Espacio de búsqueda de hiperparámetros sobre redes residuales	51
5.3. Espacio de búsqueda de hiperparámetros sobre redes recurrentes	52
5.4. Valor obtenido para cada una de las métricas sobre el conjunto de test para los experimentos variando el tipo de arquitectura.	53
5.5. Valor obtenido para cada una de las métricas p@k sobre el conjunto de test para los experimentos variando el tipo de arquitectura.	54
5.6. Valor obtenido para cada una de las métricas sobre el conjunto de test para redes de convolución variando las series consideradas.	55
5.7. Valor obtenido para cada una de las métricas p@k sobre el conjunto de test para cada una de las métricas sobre el conjunto de test para redes de convolución variando las series consideradas.	56
5.8. Valor obtenido para cada una de las métricas sobre el conjunto de test variando el tipo de arquitectura y el historial de mediciones.	57
5.9. Valor obtenido para cada una de las métricas p@k sobre el conjunto de test para todas las arquitecturas, variando el historial de mediciones.	58

Índice de tablas

5.10. Tiempo empleado en búsqueda de hiperparámetros y entrenamiento para todos los experimentos realizados, variando el tipo de arquitectura, las series consideradas y el historial de mediciones.	59
A.1. Valor obtenido para cada una de las métricas sobre el conjunto de test para la arquitectura FCN con resolución horaria e historial de tres meses.	65
B.1. Detalles de los datos de medidores extraídos de HBase	69
B.2. Valor obtenido para cada una de las métricas sobre el conjunto de test para la arquitectura CNN2D con resolución quinceminutal y horaria. .	71

Índice de figuras

2.1. <i>Framework</i> unificado de aprendizaje profundo para la clasificación de series temporales - Imagen tomada del artículo citado en [27].	6
2.2. Taxonomía con los diferentes enfoques para la clasificación de series temporales - Imagen tomada del artículo citado en [27].	8
2.3. Arquitectura de multiresolución para detección de NTL. La parte superior es la red vinculada a los datos de alta resolución (medidas quincenales), la misma posee cuatro capas de convolución 2D de 64 filtros. En color naranja se resalta la aplicación de la función de activación ReLU y en rojo las capas de pooling. La parte inferior de la red es la vinculada a los datos de baja resolución (medidas mensuales), la misma posee cuatro capas de convolución 1D de 64 filtros. Las salidas de ambas redes de convolución (de 1D y 2D) se concatenan y son la entrada a una serie de capas de redes neuronales totalmente conectadas con activación ReLU. La salida queda determinada por una función sigmoide. Imagen obtenida del artículo <i>Fraud detection on power grids while transitioning to smart meters by leveraging multi-resolution consumption data</i>	11
2.4. Arquitectura multiresolución para detección de NTL. La arquitectura recibe como entrada datos de diferentes fuentes: clima, pérdida en el área y consumos de clientes. La misma se organiza en tres niveles, en cada uno de ellos se ingresan diferentes secuencias en diferentes capas recurrentes, cuyas representaciones latentes se fusionan en pares. Las líneas discontinuas indican la dirección de la fusión. La capa de salida calcula la probabilidad de fraude a través de la función softmax. Imagen obtenida del artículo <i>Understanding electricity-theft behavior via multi-source data</i>	12
2.5. Capa de convolución, imagen tomada de las notas del curso <i>Stanford CS class CS231n Convolutional Neural Networks for Visual Recognition</i>	13
2.6. Capa de convolución con múltiples filtros y mapas de activación, imagen tomada de las notas del curso <i>Stanford CS class CS231n Convolutional Neural Networks for Visual Recognition</i>	14
2.7. Unidad básica de una arquitectura LSTM, imagen obtenida del artículo <i>Application of Long Short-Term Memory (LSTM) Neural Network for Flood Forecasting</i>	15
2.8. Bloque residual. Imagen obtenida del artículo <i>Deep Residual Learning for Image Recognition</i>	16
3.1. Mediciones durante 7 días de energía activa (color rojo) y energía reactiva (color azul) para diferentes clientes.	22

Índice de figuras

3.2. Curvas de temperatura por año y por departamento. Se diferencia la fuente del dato según el color, siendo verde los datos que provienen de MDM, naranja los datos que provienen de estaciones automáticas de INUMET y rojo los datos que provienen de estaciones convencionales de INUMET.	26
3.3. Cantidad de temperaturas faltantes por departamento y año.	27
3.4. Curvas de temperatura por año y por departamento. Se diferencia la fuente del dato según el color, siendo verde los datos que provienen de MDM e INUMET, y naranja los datos históricos que provienen de OW.	27
3.5. Cantidad de inspecciones por departamento.	28
3.6. Consumo promedio según temperatura registrada	30
3.7. Función de densidad e histograma de la máxima correlación en clientes normales (verde) y clientes que comenten fraude (rojo)	31
3.8. Funciones de distribución acumulada (clientes normales en color verde y clientes que comenten fraude en rojo) para ambos conjuntos junto con su estadístico del test de Kolmogorov-Smirnov	32
3.9. Curvas de <i>precision-recall</i> para los modelos <i>random forest</i> que consideran las características definidas en Sección 3.3.3 con y sin la nueva característica de correlación.	34
3.10. Curvas de ROC para los modelos <i>random forest</i> que consideran las características definidas en Sección 3.3.3 con y sin la nueva característica de correlación.	35
3.11. Peso que le da el modelo <i>random forest</i> a cada una de las características a la hora de clasificar instancias.	36
3.12. Peso que le da el modelo <i>random forest</i> a cada una de las características a la hora de clasificar instancias utilizando el conjunto definido en Sección 3.3.3.	37
4.1. La completitud de las series temporales se define como el porcentaje de datos no nulos para cada inspección durante un período determinado. En los gráficos de barra se presenta los rangos de porcentaje de completitud de las series temporales para cada inspección en el eje X, mientras que en el eje Y se muestra el porcentaje de inspecciones que cumplen con cada rango. El periodo considerado es de tres meses. La Figura (a) es sobre el total de inspecciones y la Figura (b) considera solamente las inspecciones donde se constató fraude.	40
4.2. La completitud de las series temporales se define como el porcentaje de datos no nulos para cada inspección durante un período determinado. En los gráficos de barra se presenta los rangos de porcentaje de completitud de las series temporales para cada inspección en el eje X, mientras que en el eje Y se muestra el porcentaje de inspecciones que cumplen con cada rango. El periodo considerado es de doce meses. La Figura (a) es sobre el total de inspecciones y la Figura (b) considera solamente las inspecciones donde se constató fraude.	41
4.3. Arquitectura CNN2D para detección de NTL que se alimenta de consumo de energía activa en resolución quinceminutal.	44
4.4. Curvas de <i>precision-recall</i> para el conjunto de test de los modelos: CNN2D con método de normalización Normalize y CNN2D con método de estandarización <i>z-score</i>	45
4.5. Curvas de ROC sobre el conjunto de test de los modelos: CNN2D con método de normalización Normalize y CNN2D con método de estandarización <i>z-score</i>	46

5.1.	Arquitectura CNN2D_AE_Q1_TEMP para detección de NTL, resultante de la búsqueda de hiperparámetros para series de energía activa, reactiva y temperatura ambiente con resolución horaria e historial de tres meses.	51
5.2.	Arquitectura ResNet_AE_Q1_TEMP para detección de NTL, resultante de la búsqueda de hiperparámetros para series de energía activa, reactiva y temperatura ambiente con resolución horaria e historial de tres meses.	52
5.3.	Arquitectura LSTM_AE_TEMP para detección de NTL, resultante de la búsqueda de hiperparámetros para series de energía activa y temperatura ambiente con resolución horaria e historial de tres meses.	53
5.4.	Curvas de <i>precision-recall</i> para los distintos tipos de arquitecturas sobre el conjunto de test.	54
5.5.	Curvas de ROC para los distintos tipos de arquitecturas sobre el conjunto de test.	55
5.6.	Curvas de <i>precision-recall</i> para el conjunto de datos de test y modelos basados en redes convolucionales variando la cantidad de series que considera cada uno.	56
5.7.	Curvas de <i>precision-recall</i> para los tres tipos de arquitecturas para evaluar el impacto del historial de mediciones. En la Figura (a) se presentan los modelos basados en redes convolucionales, en la Figura (b) los modelos basados en redes residuales y en la Figura (c) los modelos basados en redes recurrentes. En color azul siempre se grafican los modelos con tres meses de historial y en naranja los modelos con doce meses de historial.	57
A.1.	Arquitectura FCN para detección de NTL, tomada del artículo [27] adaptada para procesar serie temporal de energía activa en resolución horaria e historial de tres meses.	66
A.2.	Curvas de <i>precision-recall</i> para el modelo FCN durante la búsqueda de hiperparámetros. En la Figura (a) las curvas corresponden al conjunto de entrenamiento. En la Figura (b) las curvas corresponden al conjunto de validación.	66
A.3.	Curvas de <i>loss</i> para el modelo FCN durante la búsqueda de hiperparámetros. En la Figura (a) las curvas corresponden al conjunto de entrenamiento. En la Figura (b) las curvas corresponden al conjunto de validación.	66
A.4.	Curvas de <i>precision-recall</i> y <i>loss</i> para el modelo FCN con tasa de aprendizaje $3.5e^{-5}$, producto de la búsqueda de hiperparámetros. En la Figura (a) se grafica <i>precision-recall</i> y en la Figura (b) se grafica <i>loss</i> . . .	67
B.1.	Arquitectura CNN2D para la detección de NTL, resultante de la búsqueda de hiperparámetros para datos con resolución quinceminutal y horaria.	70
B.2.	La Figura (a) muestra las curvas de <i>precision-recall</i> para el conjunto de test de los modelos CNN2D con resoluciones quinceminutal y horaria. La Figura (b) muestra las curvas de ROC para el conjunto de test de los modelos CNN2D con resoluciones quinceminutal y horaria.	70

Esta es la última página.
Compilado el jueves 11 julio, 2024.
<http://iie.fing.edu.uy/>