# Análisis del impacto de datos desbalanceados en el rendimiento predictivo de redes neuronales convolucionales

Francisco J. Pulgar, Antonio J. Rivera, Francisco Charte, María J. del Jesus
*Instituto Andaluz Interuniversitario en Data Science and Computational Intelligence (DaSCI)*
*Departamento de Informática*
*Universidad de Jaén*
Jaén, España
{fpulgar, fcharte, arivera, mjjesus}@ujaen.es

***Abstract*—En los últimos años han surgido nuevas propuestas basadas en Deep Learning para afrontar la tarea de clasificación. Estas propuestas han obtenido buenos resultados en algunos campos, por ejemplo, en reconocimiento de imágenes. Sin embargo, existen factores que deben ser analizados para valorar su influencia en los resultados obtenidos con estos nuevos modelos. En este trabajo se analiza la clasificación de datos desbalanceados con redes neuronales convolucionales (Convolutional Neural Networks-CNNs). Para hacerlo, se han llevado a cabo una serie de tests donde se reconocen imágenes mediante CNNs. Así mismo, se utilizan conjuntos de datos con diferente grado de desbalanceo. Los resultados demuestran que el desequilibrio afecta negativamente al rendimiento predictivo.**

***Index Terms*—Deep Learning, redes neuronales convolucionales, reconocimiento de imágenes, dataset desbalanceado.**

## I. Introducción

La tarea de clasificación es una de las más estudiadas dentro del aprendizaje automático, fundamentalmente debido a su gran aplicación para resolver problemas reales. El objetivo principal de esta tarea es obtener un modelo que permita clasificar correctamente nuevos ejemplos, a partir de una serie de instancias previamente etiquetadas [1].

En los últimos años se ha producido un auge en el uso de técnicas basadas en Deep Learning (DL) para afrontar el problema de clasificación. Esto se debe fundamentalmente a dos razones: la gran cantidad de datos disponible y el incremento en la capacidad de procesamiento. Estas técnicas han mostrado muy buenos resultados en clasificación, especialmente en campos como el reconocimiento de imágenes y audio [2], [3].

Uno de los modelos que han obtenido mejores resultados en el reconocimiento de imágenes son las redes neuronales convolucionales (CNNs) [4]. Debido a la naturaleza de la convolución, estas redes se adaptan a la forma en la que se distribuyen los datos de entrada en el caso de las imágenes.

A pesar de los buenos resultados obtenidos con CNNs, son modelos relativamente recientes y existen factores que deben ser estudiados. Uno de ellos es la necesidad de analizar cómo el desbalanceo de los datos afecta al rendimiento predictivo obtenido mediante CNNs. Muchos conjuntos de datos reales contienen algún grado de desbalanceo, de ahí la importancia de estudiar este factor.

Por ello, este estudio se centra en analizar los efectos del desbalanceo de los datos en la clasificación, usando CNNs, de imágenes reales correspondientes a señales de tráfico. En este sentido, se establece la hipótesis de que el rendimiento predictivo se verá afectado negativamente a medida que aumenta el desbalanceo de los datos. La razón que lleva a plantear esta hipótesis es una cuestión de similitud con otras técnicas tradicionales, si el desbalanceo influye negativamente en la clasificación de las redes neuronales tradicionales, podría influir de forma similar a la realizada mediante CNNs. Así mismo, la naturaleza de las CNNs puede verse influenciada por el desequilibrio, ya que el ajuste de los parámetros puede depender del número de ejemplos de cada una de las clases.

El artículo está estructurado de la siguiente forma: La Sección II explica cómo afecta el desbalanceo de los datos a la tarea de clasificación. En la Sección III se introduce el concepto de DL y de CNNs. La Sección IV se pretende verificar la hipótesis establecida, para ello se lleva a cabo una experimentación donde se clasifican conjuntos de imágenes reales mediante CNNs y con diferente grado de desbalanceo. Finalmente, en la Sección V se indican las conclusiones alcanzadas.

## II. El problema de desbalanceo en clasificación

La clasificación es una tarea de predicción que, normalmente, utiliza métodos de aprendizaje supervisados [5]. Su propósito es aprender, basándose en datos previos etiquetados, patrones que permitan predecir la clase a asignar a futuros ejemplos que no estén etiquetados. En la clasificación tradicional, los conjuntos de datos están compuestos por una serie de atributos de entrada y un único valor de salida, la clase o etiqueta.

En muchas situaciones reales donde se aplica la clasificación, existen diferencias significativas en el número de elementos correspondientes a las diferentes clases, por tanto, la probabilidad de que un ejemplo pertenezca a cada una de las clases es distinta. Esta situación se conoce como el problema

de desbalanceo [6]–[8]. En muchos casos, la clase minoritaria es la más interesante a la hora de clasificar y tiene un gran coste en caso de no hacerlo correctamente.

Muchos algoritmos de clasificación obtienen buenos resultados cuando trabajan con elementos de la clase mayoritaria, pero los resultados con instancias de la clase minoritaria son erróneos frecuentemente. Esto implica que estos algoritmos, que obtienen buenos resultados con conjuntos de datos balanceados, no obtengan buen rendimiento con datos desbalanceados. Existen diferentes razones para ello:

- Muchas medidas de rendimiento usadas para guiar el proceso de entrenamiento penalizan a las clases minoritarias.
- Las reglas que predicen las clases minoritarias están muy especializadas y su cobertura es muy baja, por ello, a menudo son descartadas en favor de reglas más generales, es decir, aquellas que predicen a las clases mayoritarias.
- El tratamiento del ruido puede afectar a la clasificación de las clases minoritarias, ya que estas clases pueden ser identificadas como ruido y descartadas erróneamente o el ruido existente puede afectar en gran medida la clasificación de las clases minoritarias.

El principal obstáculo es que los algoritmos de clasificación se entrenan con un mayor número de instancias de la clase mayoritaria y cometen más errores cuando intenta clasificar ejemplos de la clase minoritaria. Para afrontar el problema han surgido muchas propuestas que pueden agruparse en [9]:

- **Muestreo de datos:** esta propuesta se basa en modificar el conjunto de entrenamiento proporcionado al algoritmo de clasificación. El objetivo es obtener datos de entrenamiento cuyas clases tengan una distribución más balanceada. En este caso, el algoritmo de clasificación no sufre ninguna modificación [10].
- **Adaptación de algoritmos:** el objetivo perseguido por este tipo de solución es adaptar los algoritmos tradicionales de clasificación para trabajar con datos desbalanceados [11]. En estos casos los datos no son modificados, es el algoritmo el que debe adaptarse.
- **Aprendizaje sensible al coste:** este tipo de solución puede incorporar modificaciones a nivel de datos y de algoritmo. Se basa en incluir penalizaciones más fuertes a los errores cometidos con la clase minoritaria que a los que se producen al clasificar la clase mayoritaria [12].

Al abordar problemas con datos desbalanceados, deben tenerse en cuenta otros factores que pueden tener gran influencia en los resultados obtenidos, por ejemplo, el solapamiento entre clases [13] o el ruido existente en los datos [14].

## III. Deep Learning

La necesidad de extraer información de más alto nivel de los datos analizados a través de métodos de aprendizaje ha hecho que emerjan nuevas áreas de estudio, en concreto DL [15]. Los modelos de DL están basados en una arquitectura profunda (multi-capa) cuyo objetivo es mapear las relaciones entre las características de los datos y los resultados esperados [16]. Este tipo de métodos aportan las siguientes ventajas:

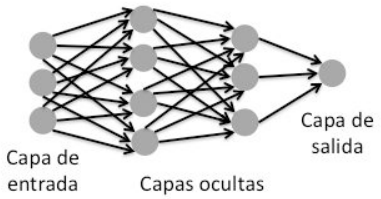
- Los modelos DL incorporan mecanismos para generar nuevas características por sí mismos, sin necesidad de realizar esto en fases externas.
- Las técnicas DL mejoran el rendimiento en cuanto a tiempo de cómputo, al realizar algunas de las tareas más costosas, como la generación de nuevas características.
- Los modelos basados en DL obtienen buenos resultados al afrontar problemas en ciertos campos como reconocimiento de imágenes o sonido, mejorando a técnicas tradicionales [2], [3].

Debido a los buenos resultados obtenidos usando propuestas basadas en DL, se han ido desarrollando diferentes arquitecturas, por ejemplo, CNNs [17] o redes neuronales recurrentes [18]. Estas arquitecturas han sido diseñadas para múltiples campos de aplicación, mostrando una gran eficiencia en el reconocimiento de imágenes. En la Sección III-A, se profundiza sobre el concepto de CNN, modelo que será usado en la experimentación asociada a este estudio.

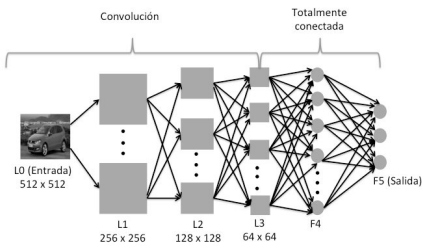
### A. Redes Neuronales Convolucionales

Las CNNs son un tipo de red neuronal profunda basada en la forma en la que los animales visualizan e identifican los objetos. Estas redes han mostrado un funcionamiento muy eficiente en ciertos campos de aplicación como en clasificación y reconocimiento de imágenes.

Las CNNs se basan en la idea de la correlación espacial mediante la aplicación de una serie de patrones de conectividad local entre neuronas de capas adyacentes [4], [19]–[21]. Esto implica que, al contrario que otras redes neuronales tradicionales donde cada neurona se conecta con todas las neuronas de las capas anteriores, en las CNNs cada neurona se conecta únicamente a una región concreta de la capa anterior. Otra diferencia fundamental es que las neuronas de las CNNs están distribuidas en tres dimensiones, mientras que las redes tradicionales sólo ocupan dos dimensiones. La Figura 1 muestra la diferencia entre ambos tipos de redes.




(a) Red Neuronal Tradicional




(b) Red Neuronal Convolucional

Fig. 1. Diferencias entre redes neuronales tradicionales y CNNs.

La arquitectura de una CNN está basada en una secuencia de capas, cada una de ellas transforma el volumen de entrada mediante la utilización de una función concreta. Hay tres tipos: capa de convolución, de *pooling* y totalmente conectada. Los tipos de capas indicados anteriormente se usan para formar una CNN compleja, para ello, capas de diferentes tipos se enlazan para formar la arquitectura del modelo que se quiera construir [4], [17], [20], [21].

## IV. Experimentación

Una vez expuestos los principales conceptos teóricos necesarios para establecer las bases de este trabajo, se presenta la hipótesis que se pretende verificar y la experimentación que se ha desarrollado para hacerlo.

Durante esta fase se pretende demostrar si un excesivo desequilibrio entre las instancias de las diferentes clases que forman el conjunto de datos afecta al rendimiento predictivo obtenido utilizando CNNs.

Existen diferentes estudios [9]–[12] que muestran que esta propiedad de los datos afecta a determinados modelos de clasificación y proponen soluciones para reducir los efectos al trabajar con datos que presenten desequilibrio. Sin embargo, debido a que las técnicas basadas en CNNs son relativamente recientes, apenas existen estudios que analicen la influencia de los datos desbalanceados en este tipo de modelos.

En este trabajo, se asume la idea de que los datos con desequilibrio entre las clases afectan a la clasificación mediante CNNs. Este hecho lleva a proponer la hipótesis inicial del trabajo: los resultados obtenidos mediante CNNs utilizando datos desbalanceados reducen el rendimiento predictivo de las mismas.

Por tanto, el objetivo de este estudio es analizar si los datos desbalanceados influyen negativamente en la clasificación de imágenes utilizando CNNs. Para hacerlo, se realizan una serie de test usando datos con diferente nivel de desbalanceo (ratio de desbalanceo-IR). Estos datos corresponden a imágenes de tráfico reales [22], siendo el objetivo asociar cada imagen de entrada a la señal correspondiente. El modelo utilizado para realizar la clasificación será una CNN, cuya red usada tendrá la misma arquitectura en todos los experimentos llevados a cabo. Así mismo, debe tenerse en cuenta que no se va a utilizar ningún mecanismo para reducir los efectos del desbalanceo, ya que el objetivo es analizar cómo afectan a estos modelos.

### A. Framework experimental

Para desarrollar el experimento se ha utilizado un dataset de señales de tráfico [22] con un total de 11 910 imágenes pertenecientes a 43 tipos de señales o clases diferentes. En primer lugar, es necesario realizar un pre-procesamiento de las imágenes. Esta fase tiene dos objetivos fundamentales: por un lado, recortar la imagen para seleccionar solo la parte correspondiente a la señal de tráfico (Figura 2); y, por otro lado, escalar las imágenes para hacer que todas ellas tengan la misma dimensión. En este sentido, se ha decidido escalar las imágenes a un tamaño de 32x32, ya que es el usado en otros estudios similares [19].

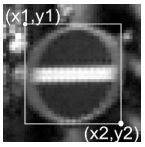



Fig. 2. Ejemplo de imagen recortada (fuente: http://benchmark.ini.rub.de).

Una vez que todas la imágenes del conjunto de datos han sido pre-procesadas, se seleccionan las correspondientes a 10 clases con el objetivo de acentuar el desbalanceo entre los datos. Las clases seleccionadas serán las 5 con más ejemplos y las 5 con menos ejemplos. De esta forma, se obtiene un subconjunto de datos a partir del conjunto completo original.

Una aspecto importante cuando se trabaja con datos desbalanceados es el IR. Esta medida es definida como el ratio entre el número de ejemplos de la clase mayoritaria y el de la clase minoritaria [23], [24].

Otro aspecto a tener en cuenta es que el número de imágenes usadas en cada experimento será el mismo, únicamente cambiará el IR del conjunto de datos. Esto es importante, ya que si el número de ejemplos varía de forma significativa puede afectar a los resultados obtenidos, ocultando así los efectos del desequilibrio de los datos. Por tanto, se han seleccionado 2 700 imágenes para cada ejecución. La razón por la que se selecciona este número viene dada por la cantidad de imágenes de las clases minoritarias en el dataset original. Este conjunto contiene unas 270 imágenes de las clases minoritarias y unas 2 700 de las mayoritarias. Al balancear el conjunto de datos, se seleccionan 270 imágenes de cada clase, por lo que el número total de instancias es 2 700, que se mantiene constante durante todas las ejecuciones, a pesar de cambiar el IR.

Finalmente, se deben indicar que las métricas de evaluación usadas para evaluar el rendimiento predictivo de la CNN en cada caso serán: Tasa de Error, *Precision* y *Recall*.

### B. Arquitectura de la CNN

La arquitectura de la CNN usada para clasificar debe ser la misma en todos los casos, a pesar de que el conjunto de imágenes variará en cada uno de ellos. Esto es necesario para evaluar los efectos del desequilibrio en los datos. Esta CNN tiene una arquitectura cuya secuencia de capas es la siguiente:

- **Capa convolucional 1:** 32 filtros son aplicados sobre la imagen original. Tienen un tamaño de 5x5. Esto produce 32 mapas de características.
- **Capa pooling 1:** Los 32 mapas anteriores se reducen usando la función máximo con una ventana de pooling de tamaño 2 y un paso 2.
- **Capa convolucional 2:** Esta capa aplica 64 filtros con un tamaño de 5x5 sobre la salida de la capa anterior y genera un nuevo conjunto de mapas de características.
- **Capa pooling 2:** Se reduce la dimensionalidad de los mapas de características anteriores. Se utiliza la función máximo con una ventana de tamaño 2 y un paso 2.
- **Capa completamente conectada:** En esta capa todos los elementos de la fase anterior son combinados y

usados para realizar la clasificación. Esta capa tiene tantos elementos como clases tenga el problema.

Durante el proceso de entrenamiento, se usa la entropía cruzada para evaluar la red y, posteriormente, la propagación hacia atrás para modificar los pesos de la red. La configuración elegida para la red es la usada por defecto en el software utilizado, TensorFlow [1], considerando el tamaño de las imágenes y los diferentes valores de salida que puede tener el problema.

### C. Análisis de resultados

La experimentación llevada a cabo consiste en diferentes ejecuciones en las que se clasifican imágenes reales mediante una CNN, cada una de las cuales tiene diferente IR. En concreto, se han realizado cuatro experimentos con IR 1/10, 1/5, 1/3 y 1/1.

Como se ha descrito en la Sección IV-A, el conjunto de datos seleccionado tiene un total de 11 910 imágenes. Sin embargo, se ha justificado el hecho de seleccionar únicamente 2 700 imágenes en cada uno de los experimentos, con el objetivo de que todas las ejecuciones tengan el mismo número de instancias. Por ello, el primer paso consiste en reducir el número de imágenes de cada clase de forma proporcional y aleatoria, para obtener el número establecido sin afectar a la tasa de desbalanceo. Así, los distintos conjuntos de datos presentan la distribución de ejemplos que puede verse en la Tabla I para cada clase.

Tabla I
INSTANCIAS DE ENTRENAMIENTO Y TEST POR EXPERIMENTACIÓN.

| | IR 1/10 | | IR 1/5 | | IR 1/3 | | IR 1/1 | |
|---|---|---|---|---|---|---|---|---|
| Clase | Train | Test | Train | Test | Train | Test | Train | Test |
| 1 | 36 | 11 | 66 | 20 | 98 | 31 | 203 | 67 |
| 2 | 351 | 115 | 320 | 106 | 288 | 95 | 203 | 67 |
| 3 | 392 | 130 | 361 | 118 | 325 | 106 | 203 | 67 |
| 4 | 365 | 121 | 334 | 111 | 301 | 100 | 203 | 67 |
| 5 | 376 | 125 | 345 | 115 | 311 | 103 | 203 | 67 |
| 6 | 36 | 11 | 65 | 21 | 97 | 32 | 203 | 67 |
| 7 | 39 | 13 | 72 | 24 | 108 | 36 | 203 | 67 |
| 8 | 36 | 11 | 65 | 21 | 97 | 32 | 203 | 67 |
| 9 | 360 | 120 | 330 | 110 | 297 | 99 | 203 | 67 |
| 10 | 39 | 13 | 72 | 24 | 108 | 36 | 203 | 67 |
| Total | 2030 | 670 | 2030 | 670 | 2030 | 670 | 2030 | 670 |

La Tabla I muestra el número de ejemplos de cada una de las clases para los conjuntos de datos de cada ejecución. En ella, se puede ver que todas tienen un total de 2 700 imágenes de las que 2 030 serán usadas para entrenar la red y 670 para evaluar el modelo. Sin embargo, se puede apreciar como el ratio entre las instancias de las clases mayoritarias y minoritarias es diferente. A continuación, se exponen los resultados obtenidos en dichos experimentos.

#### 1) Resultados con IR 1/10:

El primer experimento de clasificación de imágenes utilizando CNN llevado a cabo comienza con un conjunto de datos con un gran nivel de desbalanceo entre clases. La Tabla I muestra el número de instancias de cada una de ellas. Así mismo, se puede ver cómo, en este primer experimento, existe un IR de aproximadamente 1/10 entre las clases minoritarias y mayoritarias. Una vez establecido el conjunto de datos, se utiliza una CNN para realizar la clasificación, obteniendo los resultados presentados en las Tablas II y III.

[1]https://www.tensorflow.org/

Tabla II
NÚMERO DE INSTANCIAS TOTAL Y ERRORES EN TEST POR EXPERIMENTACIÓN.

| | IR 1/10 | | IR 1/5 | | IR 1/3 | | IR 1/1 | |
|---|---|---|---|---|---|---|---|---|
| Clase | Test | Error | Test | Error | Test | Error | Test | Error |
| 1 | 11 | 4 | 20 | 4 | 31 | 2 | 67 | 1 |
| 2 | 115 | 1 | 106 | 3 | 95 | 2 | 67 | 0 |
| 3 | 130 | 1 | 118 | 2 | 106 | 2 | 67 | 3 |
| 4 | 121 | 2 | 111 | 0 | 100 | 2 | 67 | 0 |
| 5 | 125 | 0 | 115 | 2 | 103 | 0 | 67 | 0 |
| 6 | 11 | 4 | 21 | 2 | 32 | 2 | 67 | 0 |
| 7 | 13 | 2 | 24 | 0 | 36 | 0 | 67 | 0 |
| 8 | 11 | 4 | 21 | 1 | 32 | 0 | 67 | 0 |
| 9 | 120 | 1 | 110 | 1 | 99 | 1 | 67 | 1 |
| 10 | 13 | 3 | 24 | 0 | 36 | 0 | 67 | 3 |
| Total | 670 | 22 | 670 | 15 | 670 | 11 | 670 | 8 |

La Tabla II muestra el número de ejemplos de test por clase y el número de errores del modelo al clasificar dichos ejemplos. Además, en la Tabla III se puede ver las métricas Tasa de Error, *Precision* y *Recall* por clase. Ambas tablas presentan los resultados para los 4 experimentos realizados.

En el primero experimento con IR 1/10, los resultados obtenidos muestran que la Tasa de Error por clase tiene un valor global de 0.033, en concreto, de las 670 imágenes de test son clasificadas erróneamente 22. Así mismo, el valor medio de *Precision* es de 0.963 y el de *Recall* 0.848. Estos resultados servirán de base para determinar si las ejecuciones realizadas con menor grado de desbalanceo tienen mejor rendimiento.

#### 2) Resultados con IR 1/5:

El siguiente paso es reducir el IR a 1/5. Para hacerlo, en primer lugar se parte del dataset original con 11 910 imágenes y, posteriormente, se realiza una selección aleatoria del 50% de ejemplos de las clases mayoritarias. De está forma obtenemos un subconjunto con 6 510 imágenes. Una vez hecho esto, se seleccionan 2 700 para que todos los experimentos tengan el mismo número de instancias.

En la Tabla I puede verse el número de ejemplos por clase para este experimento y puede verificarse que el IR es de 1/5. Los resultados obtenidos al clasificar con la CNN esta nueva distribución de ejemplos se muestra en las Tablas II y III.

Estos resultados con un IR 1/5 muestran cómo disminuye la Tasa de Error con respecto a la ejecución anterior. La Tasa de Error obtenida es 0.022, ya que 15 imágenes del total de 670 de test han sido clasificadas erróneamente. Además, los resultados muestran que tanto *Precision* como *Recall* aumentan respecto al experimento previo. El valor medio de *Precision* obtenido es de 0.984 y el de *Recall* es 0.958. Los resultados refuerzan la hipótesis inicial, por lo que se continúa reduciendo el IR.

#### 3) Resultados con IR 1/3:

Para continuar verificando la hipótesis, se reduce el IR a 1/3. Para ello, a partir del dataset de partida con 11 910 imágenes, se selecciona de forma aleatoria el 30% de los ejemplos de las clases mayoritarias, obteniendo un subconjunto con 4 350 imágenes. Posteriormente, se seleccionan 2 700 para que todas las ejecuciones tengan el mismo tamaño.

Tabla III
Resultados para conjunto de test.

| | IR 1/10 | | | IR 1/5 | | | IR 1/3 | | | IR 1/1 | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Clase | Error | Precision | Recall | Error | Precision | Recall | Error | Precision | Recall | Error | Precision | Recall |
| 1 | 0.364 | **1.000** | 0.636 | 0.200 | **1.000** | 0.800 | 0.065 | **1.000** | 0.935 | **0.015** | 0.985 | **0.985** |
| 2 | 0.009 | 0.966 | 0.991 | 0.028 | 0.954 | 0.972 | 0.021 | 0.989 | 0.979 | **0.000** | **1.000** | **1.000** |
| 3 | **0.008** | 0.963 | **0.992** | 0.017 | 0.951 | 0.983 | 0.019 | 0.990 | 0.981 | 0.045 | **1.000** | 0.955 |
| 4 | 0.017 | 0.983 | 0.983 | **0.000** | **1.000** | **1.000** | 0.020 | 0.990 | 0.980 | **0.000** | 0.985 | **1.000** |
| 5 | **0.000** | 0.977 | **1.000** | 0.017 | 0.983 | 0.983 | **0.000** | 0.956 | **1.000** | **0.000** | **0.985** | **1.000** |
| 6 | 0.364 | 0.875 | 0.636 | 0.095 | **1.000** | 0.905 | 0.062 | 0.968 | 0.937 | **0.000** | 0.985 | **1.000** |
| 7 | 0.154 | 0.917 | 0.846 | **0.000** | 0.960 | **1.000** | **0.000** | 0.947 | **1.000** | **0.000** | **1.000** | **1.000** |
| 8 | 0.364 | **1.000** | 0.636 | 0.048 | **1.000** | 0.952 | **0.000** | **1.000** | **1.000** | **0.000** | 0.985 | **1.000** |
| 9 | **0.008** | 0.952 | **0.992** | 0.009 | 0.991 | 0.991 | 0.010 | **1.000** | 0.990 | 0.015 | 0.956 | 0.985 |
| 10 | 0.231 | **1.000** | 0.769 | **0.000** | **1.000** | **1.000** | **0.000** | **1.000** | **1.000** | 0.045 | **1.000** | 0.955 |
| Media | 0.033 | 0.963 | 0.848 | 0.022 | 0.984 | 0.958 | 0.016 | 0.984 | 0.980 | **0.012** | **0.988** | **0.988** |

En la Tabla I, se verifica que la distribución de ejemplos por clase de la tercera experimentación tiene un IR de 1/3, ya que se ha llevado a cabo una reducción mayor de las instancias de la clase mayoritaria. Los resultados obtenidos con esta nueva distribución pueden verse en las Tablas II y III.

Observando los resultados para este experimento con un IR de 1/3, se puede ver que la tendencia vista en la sección previa continúa, ya que mejora el rendimiento. En este caso, la tasa de Error global obtenida es de 0.016, en concreto, se clasifican mal 11 imágenes del total de 670 del conjunto de test. Así mismo, el valor medio de *Precision* es de 0.984 y el de *Recall* es 0.980. De esta forma, se verifica la tendencia general que confirma que, a medida que decrece el IR, los resultados de clasificación mediante CNNs mejoran. Este hecho vuelve a confirmar la hipótesis inicial y, por ello, se pasa a realizar el último experimento con el dataset completamente balanceado.

*4) Resultados con IR 1/1:*

El objetivo de este último test es, como se ha descrito anteriormente, verificar la mejora de los resultados obtenidos al clasificar con CNN utilizando un dataset balanceado (IR 1/1). Por tanto, el primer paso es balancear el conjunto de datos inicial de 11 910 imágenes. Para hacerlo, se selecciona la clase con menos ejemplos y se eliminan elementos aleatoriamente del resto de clases hasta que tengan el mismo número de instancias. De esta forma, se obtiene un subconjunto de imágenes para llevar a cabo la experimentación con 2 700, cuya distribución puede verse en la Tabla I.

Las Tablas II y III muestran los resultados obtenidos en clasificación usando una CNN y un conjunto de datos balanceado. La Tasa de Error es de 0.012, solo se han clasificado mal 8 imágenes del total de 670 de test, el valor de *Precision* es 0.988 y el de *Recall* 0.988. Estos resultados vuelven a mejorar los obtenidos en las experimentaciones previas.

Las evidencias mostradas por las distintas experimentaciones confirman la hipótesis inicial: a medida que el conjunto de datos está más balanceado, los resultados de la clasificación realizada con CNNs mejoran.

*5) Discusión de Resultados:*

En las Subsecciones previas, se ha confirmado la hipótesis inicial de este artículo. A continuación se muestra una representación visual de los resultados obtenidos en las Figuras 3 y 4, así mismo, se realiza una discusión de los mismos.

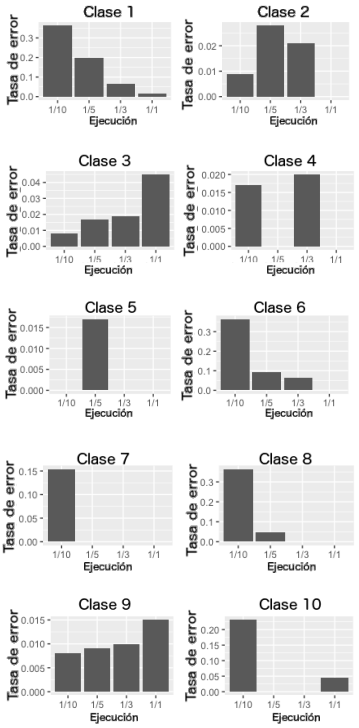



Fig. 3. Tasa de Error por clase y experimento.

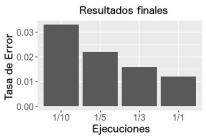



Fig. 4. Tasa media de Error por experimento.

Por un lado, la Figura 3 muestra el Error obtenido por clase para cada experimento, puede verse que en todos los casos, exceptuando las clases 3 y 9, los resultados obtenidos con el conjunto balanceado son los mejores. Por otro lado, la Figura 4 muestra el Error medio para cada experimento, se observa que se obtiene mejor rendimiento a medida que decrece el IR.

Los análisis generales muestran que la reducción del grado de desbalanceo produce una mejora en el rendimiento cuando se clasifica con CNN, lo que confirma la hipótesis inicial. Realizando un estudio por clase, se observa que los mejores resultados se obtienen con el dataset balanceado, excepto en las clases 3 y 9. En estos casos los mejores resultados se obtienen con el conjunto de datos desbalanceado. El motivo puede deberse a que son clases mayoritarias, por lo que al balancear el dataset el número de imágenes de estas clases es menor que en el conjunto de datos desbalanceado, factor que puede afectar al rendimiento predictivo.

Una vez realizado el análisis previo, se concluye que uno de los aspectos que más podría afectar al rendimiento es el cálculo de los pesos de la última capa de la red convolucional. Esta capa completamente conectada determina la clase en una última fase supervisada, y los pesos obtenidos podrían priorizar las clases mayoritarias con respecto a las minoritarias.

Otra característica de la CNN que podría influir es el cálculo de los pesos correspondientes a los diferentes filtros en las capas convolucionales. Estos se mueven a lo largo del espacio de entrada, modificando los pesos durante el proceso, de modo que se podrían adaptar excesivamente a las clases mayoritarias cuando hay un mayor desequilibrio. Estas conclusiones abren nuevas vías de estudio, ya que son necesarios análisis más detallados para verificar si se cumplen o no.

## V. Conclusiones

Uno de los principales problemas cuando se afronta la tarea de clasificación con datos reales es que, en muchos casos, no están balanceados, lo que influye negativamente en el rendimiento predictivo de gran cantidad de modelos. En este trabajo, se verifica si este problema de desequilibrio afecta también a la clasificación realizada con CNNs.

Las ejecuciones realizadas han confirmado la hipótesis inicial: a medida que el desbalanceo en los datos es minimizado, los resultados obtenidos a través de la CNN mejoran. Esto implica que debe tenerse en cuenta la distribución de los datos cuando se usan este tipo de técnicas, ya que un excesivo desequilibrio puede afectar negativamente.

Los resultados derivados de este estudio abren nuevas vías de trabajo. Una primera aproximación para afrontar el problema asociado a clasificar datos desbalanceados con CNNs es la aplicación de métodos clásicos: técnicas de muestreo, métodos de aprendizaje sensibles al coste o ensembles, estas técnicas tienen como objetivo reducir los efectos del desequilibrio de los datos. Así mismo, existe la posibilidad de crear nuevos modelos que combinen técnicas tradicionales que afronten el problema de desbalanceo con CNNs, generando algoritmos híbridos que tengan en cuenta este factor.

Este trabajo es una primera aproximación al problema utilizando un conjunto de datos particular y una técnica concreta, por lo que este estudio debe ser ampliado en trabajos futuros para establecer una conclusiones sólidas.

## Agradecimientos

Este trabajo de F. Pulgar ha sido financiado por el Ministerio de España de Educación bajo el Programa Nacional FPU (Ref. FPU16/00324). Este trabajo ha sido parcialmente financiado por el Ministerio de España de Ciencia y Tecnología bajo el proyecto TIN2015-68454-R.

## Referencias

[1] R. Duda, P. Hart, D. Stork, Pattern Classification, 2nd edition, John Wiley, 2000.
[2] D. Ciresan, U. Meier, Multi-column Deep Neural Networks for Image Classification, Technical Report No. IDSIA-04-12, 2012.
[3] R. McMillan, How Skype Used AI to Build Its Amazing New Language Translator, Wire, 2014.
[4] Y. LeCun, K. Kavukcuoglu, C. Farabet, Convolutional networks and applications in vision, in Circuits and Systems (ISCAS), in Proceedings of 2010 IEEE International Symposium on, p. 253-256, 2010.
[5] S. Kotsiantis, Supervised machine learning: A review of classification techniques, Proceedings of the 2007 conference on Emerging Artificial Intelligence Applications in Computer Engineering, p. 3-24, 2007.
[6] N.V. Chawla, N. Japkowicz, A. Kotcz, Editorial: special issue on learning from imbalanced data sets, SIGKDD Explorations, 6 (1), p. 1-6, 2004.
[7] H. He, E.A. García, Learning from imbalanced data, IEEE Transactions on Knowledge and Data Engineering, 21 (9), p. 1263-1284, 2009.
[8] Y. Sun, A.K.C. Wong, M.S. Kamel, Classification of imbalanced data: a review, International Journal of Pattern Recognition and Artificial Intelligence, 23 (4), p. 687-719, 2009.
[9] M. Galar, A. Fernández, E. Barrenechea, H. Bustince, F. Herrera, A review on ensembles for class imbalance problem: bagging, boosting and hybrid based approaches, IEEE Transactions on Systems, Man, and Cybernetics, 42 (4), p. 463-484, 2012.
[10] G.E.A.P.A. Batista, R.C. Prati, M.C. Monard, A study of the behaviour of several methods for balancing machine learning training data, SIGKDD Explorations, 6 (1), p. 20-29, 2004.
[11] B. Zadrozny, Learning and making decisions when costs and probabilities are both unknown, Proceedings of the 7th International Conference on Knowledge Discovery and Data Mining, p. 204-213, 2001.
[12] B. Zadrozny, J. Langford, N. Abe, Cost–sensitive learning by cost–proportionate example weighting, in Proceedings of the 3rd IEEE International Conference on Data Mining (ICDM'03), p. 435-442, 2003.
[13] Prati, R. C., Batista, G. E., Monard, M. C., Class imbalances versus class overlapping: an analysis of a learning system behavior, in Mexican international conference on artificial intelligence, p. 312-321, 2004.
[14] Frénay, B., Verleysen, M., Classification in the presence of label noise: a survey, IEEE transactions on neural networks and learning systems, 25(5), p. 845-869, 2014.
[15] Y. Bengio, A. Courville, P. Vincent, Representation Learning: A Review and New Perspectives. Pattern Analysis and Machine Intelligence, IEE Transactions, 3 (8), p. 1798-1828, 2013.
[16] I. Goodfellow, Y. Bengio, A. Courville, Deep Learning, 2016.
[17] Y. LeCun, Y. Bengio, Convolutional networks for images, speech, and time-series, M. A. Arbib, The Handbook of Brain Theory and Neural Networks, 1995.
[18] H. Sak, A. Senior, F. Beaufays, Long short-term memory recurrent neural network architectures for large scale acoustic modeling, Proc. Interspeech, p. 338-342, 2013.
[19] P. Sermanet, Y. LeCun, Traffic sign recognition with multi-scale convolutional networks, in Proceedings of International Joint Conference on Neural Networks, 2011.
[20] Krizhevsky, A., Sutskever, I., Hinton, G. E., Imagenet classification with deep convolutional neural networks, in Advances in neural information processing systems, p. 1097-1105, 2012.
[21] Jin, K. H., McCann, M. T., Froustey, E., Unser, M., Deep convolutional neural network for inverse problems in imaging, IEEE Transactions on Image Processing, 26(9), p. 4509-4522, 2017.
[22] J. Stallkamp, M. Schlipsing, J. Salmen, and C. Igel, Man vs. computer: Benchmarking machine learning algorithms for traffic sign recognition, Neural Networks, vol. 32, p. 323-332, 2012.
[23] V. García, J.S. Sánchez, R.A. Mollineda, On the effectiveness of preprocessing methods when dealing with different levels of class imbalance, Knowledge Based Systems, 25 (1), p. 13-21, 2012.
[24] A. Orriols-Puig, E. Bernadó-Mansilla, Evolutionary rule-based systems for imbalanced datasets, Soft Computing, 13 (3), p. 213-225, 2009.