Clasificación automática de Galaxias mediante redes neuronales ocupando imágenes del catálogo astronómico del proyecto Galaxy Zoo.

Pruebas y Resultados de la Investigación

Gabriel Salazar, Ricardo Contreras, Neil Nagar,
Mauricio Solar, Marcelo Mendoza
Jorge Ibsen, Lars Nyman, Eduardo Vera, Diego Mardones, Guillermo Cabrera,
Paola Arellano, Karim Pichara, Nelson Padilla, Victor Parada.

Concepción, 10 de septiembre de 2014

Resumen

El presente documento muestra la configuración de las pruebas efectuadas en torno a la investigación realizada, implementadas en MATLAB, de las cuales se desprenden algunos resultados importantes. Asimismo, se extraen las conclusiones de los resultados obtenidos y los posibles trabajos a futuro a realizar en este tema.

Palabras Claves: Galaxias, Clasificación de Galaxias, Galaxy Zoo, SDSS, ChiVO, Minería de Datos, Machine Learning, Redes Neuronales Artificales.

1. Resumen Ejecutivo

Desde que Edwin Hubble en el año 1926 ideó su famosa secuencia, ha surgido la necesidad de clasificar las galaxias según ciertas características particulares. Con el tiempo la cantidad de galaxias descubiertas ha aumentado enormemente, elevando así la complejidad en términos de tiempo a ocupar para clasificarlas. Debido a esto, surge el problema de clasificar las galaxias en un tiempo razonable y con cierto grado de precisión. Las máquinas de aprendizaje cobran aquí gran valor, debido a sus características propias, ya sea de emulación de las neuronas cerebrales o la imitación del comportamiento de seres vivos.

La organización del informe será como sigue, primero se definirá la metodología de trabajo de la investigación. Posteriormente se procederá a mostrar la configuración de las pruebas efectuadas, junto con el diseño del plan de estas, además se describen los métodos de medición del rendimiento de la red y el plan de pruebas diseñado. En la sección que sigue se mostraran los resultados de cada una de las pruebas, junto a un análisis de los resultados. Finalmente, se mencionan las conclusiones obtenidas al finalizar el desarrollo de la investigación y los trabajos futuros que generaría el trabajo efectuado.

Índice

Índice de figuras

Índice de cuadros

2. Metodología de Trabajo

2.1. Fase Inicial y recolección de información

Como primera parte se construyó el marco teórico en que se sustenta la memoria. Se basó en la investigación de los trabajos hechos anteriormente para tratar la clasificación de galaxias basados en redes neuronales artificiales, y en el estudio de las redes neuronales. Todo esto con el fin de dejar los primeros cimientos de la investigación llevada a cabo.

2.2. Entrenamiento previo

Posteriormente se llevó a cabo una familiarización con el entorno desarrollador. Se estudió acerca de morfología de galaxias, teoría de Redes Neuronales Artificiales y sus algoritmos de aprendizaje, reducción de dimensionalidad de los datos, manejo de MATLAB y formulación de consultas en SQL para CasJobs.

2.3. Análisis y especificación

En esta parte se realizó la especificación y análisis de los requerimientos del sistema. Se procedió a analizar las fuentes de datos, como el repositorio ocupado por el proyecto Galaxy Zoo. Posteriormente se analizó la calidad de la información y en qué condiciones nos la otorga Galaxy Zoo, para poder trabajar con los datos proporcionados. Además, esto llevó también a la elección y especificación de los datos de entrada a la red (Selección de Features o Parámetros).

Posteriormente se procedió a la elección de la arquitectura y tipo de RNA a utilizar en las primeras pruebas, así como también las métricas a ocupar para medir los resultados obtenidos.

2.4. Implementación, desarrollo y pruebas del programa base (prototipo)

Esta parte del proceso se llevó a cabo en forma iterativa, siguiendo las directrices del Desarrollo Evolutivo de Ingeniería de Software. Esta fase contó con las siguientes subfases:

- **Diseño**: Se buscó las alternativas válidas para poder generar la red neuronal, y se encontró una arquitectura adecuada para ella basándose en el trabajo hecho por Banerji et al.[1] con redes neuronales.
- **Desarrollo**: Se pre-procesaron los parámetros de entrada a cada una de la respectivas redes neuronales, y se procedió a implementar cada red en el entorno MATLAB.
- **Pruebas**: Se ejecutó cada una de la pruebas planteadas, de acuerdo al plan de pruebas propuesto.
- **Análisis de Resultados**: Se realizó la recolección de los resultados entregados en cada prueba realizada, y se analizó cada una de ellas.

3. Pruebas

3.1. Configuración del entrenamiento

Tomando en cuenta el trabajo efectuado por Banerji et al. [1], se decidió emular parte de la implementación efectuada por ellos. La configuración de la red se fijó en $n : 24 : 24 : 3$, donde n es el número de parámetros de entrada para cada prueba, con dos capas ocultas de 24 neuronas cada una y tres neuronas de salida. El tipo de red neuronal elegida fue el Perceptrón multicapa, junto con el algoritmo de aprendizaje de quasi-Newton. La elección del algoritmo se debió a la limitación de memoria del equipo ocupado para las pruebas, ya que en primera instancia se había optado por el algoritmo de Levenberg-Marquard, pero como se mostró en el capitulo 1, a mayor cantidad de patrones de aprendizaje, mayor gasto de memoria efectúa. La función de activación elegida fue la función sigmoidal para las capas ocultas, y la función identidad para la capa de salida. La elección del modelo de arquitectura, que en nuestro caso fue Feedforward, influye en la elección de las funciones de activación, ya que MATLAB al elegir este tipo de arquitectura toma por defecto las funciones ya descritas. En tanto, los parámetros de detención para cada entrenamiento fueron los siguientes:

- Gradiente Mínimo: 10^{-6}
- Máximo número de comprobaciones de validación[1]: 10 comprobaciones
- MSE Objetivo: 0
- Número de épocas máxima: 1000

La red detendrá el entrenamiento cuando se cumpla uno de los criterios anteriores primero, ya sea porque se alcanzó el error mínimo, se alcanzó el número arbitrario de ciclos (épocas) elegido, se llegó al gradiente mínimo elegido, o se alcanzó el número máximo de comprobaciones de validación para prevenir el sobre-entrenamiento (Overfitting). Se normalizaron los parámetros de acuerdo a la función de MATLAB `mapstd`, la cual normaliza los datos en base a la varianza unitaria y a la media entre todos los datos de cada parámetro de entrada de la red. También se probó la normalización `mapminmax`, que transforma los datos de los parámetros al rango $[-1, 1]$, pero no hubo mayor diferencia de rendimiento con respecto al otro método durante el entrenamiento. La división de los datos de entrenamiento, de validación y de testing se hizo de acuerdo a la función `divideint`, la cual divide los datos utilizando bloques intercalados aleatorios. El porcentaje elegido para cada set de datos fue el siguiente: un 85 % del total de datos para el entrenamiento, un 10 % para la validación y un 5 % para el testing. Se consideró esta elección para los dos conjuntos de prueba descritos en el capitulo anterior.

El rendimiento se midió mediante el MSE o *Error Cuadrático Medio*. El objetivo del MSE es minimizar el error entre la salida deseada y la salida que otorga el entrenamiento, y está definido como sigue:

$$MSE = \frac{1}{N} \sum_{i=1}^{N} (e_i)^2,$$

[1]Número máximo de comprobaciones sucesivas en donde el error del rendimiento del set de validación durante el entrenamiento no decrece.

donde e_i es la diferencia entre la salida deseada y la salida de la red y N es el número total de objetos ocupados durante el entrenamiento.

Por otra parte, se decidió también ocupar otro método de medición de rendimiento para validar las redes neuronales, la matriz de confusión [6]. Esta matriz muestra la cantidad de objetos bien clasificados durante el entrenamiento, validación y testing, por cada clase. Un ejemplo de esta matriz para dos clases se muestra en la tabla 1. En el ejemplo el valor a representa la cantidad de clasificaciones de Clase 1 bien clasificadas, el valor d la cantidad de clasificaciones de Clase 2 bien clasificadas, y los valores c y b representan la cantidad de clasificaciones mal efectuadas por la red, con respecto al valor deseado o esperado.

	Deseado Clase 1	Deseado Clase 2
Predicho Clase 1	a	c
Predicho Clase 2	b	d

Cuadro 1: Matriz de Confusión - Ejemplo

También se considera el porcentaje de éxito, el cual mide qué tan precisa es la red. Éste se calcula considerando la matriz de confusión de la red. Tomando el ejemplo de la tabla 1, su porcentaje de éxito se calcula como sigue:

$$\%\, de\, éxito = \frac{a+d}{a+b+c+d} \times 100.$$

Por ultimo, tomando en cuenta la cantidad de clasificaciones mal efectuadas de la matriz de confusión, se calcula el porcentaje de error de la red. La siguiente ecuación muestra el cálculo de este porcentaje:

$$\%\, de\, error = \frac{b+c}{a+b+c+d} \times 100.$$

Anexo a estos métodos de medición de rendimiento, se tienen otros métodos los cuales no se consideran en esta memoria, que también toman como base la matriz de confusión. Estos son: la tasa de falsos positivos, de falsos negativos, de verdaderos positivos y de verdaderos negativos [5], los cuales se definen en base a la tabla 1 y se mostrarán a continuación.

- Tasa de Falsos Positivos: es la proporción de casos Clase 1 que se clasificaron incorrectamente como Clase 2.
$$Tasa\, de\, falsos\, positivos = \frac{b}{a+b}$$
- Tasa de Falsos Negativos: es la proporción de casos Clase 2 que fueron clasificados incorrectamente como Clase 1.
$$Tasa\, de\, falsos\, negativos = \frac{c}{c+d}$$
- Tasa de Verdaderos Positivos: es la proporción de casos Clase 2 que fueron identificados correctamente.
$$Tasa\, de\, verdaderos\, positivos = \frac{d}{c+d}$$

- Tasa de Verdaderos Negativos: se define como la proporción de casos Clase 1 que fueron clasificados correctamente.

$$Tasa\,de\,verdaderos\,negativos = \frac{a}{a+b}$$

3.2. Plan de pruebas

Para ver el rendimiento de las redes neuronales en el problema objetivo, se procedió a diseñar un plan de pruebas, ocupando la información de los filtros fotométricos u y r. Estos filtros se eligieron ya que se quiso variar el problema resuelto en Banerji et al.[1] y ver qué tan efectivas son las redes neuronales en otros filtros fotométricos distinto al filtro i ocupado por ellos.

Se idearon ocho pruebas, tomando distintos números de parámetros y distintos filtros. Estas pruebas se muestran a continuación:

- **Prueba 1**: Parámetros basados en el color y en los perfiles ajustados, en el filtro u.
- **Prueba 2**: Parámetros basados en la forma y textura, en el filtro u.
- **Prueba 3**: Los parámetros de Prueba 1 y Prueba 2 juntos, en el filtro u.
- **Prueba 4**: Parámetros basados en el color y en los perfiles ajustados, en el filtro r.
- **Prueba 5**: Parámetros basados en la forma y textura, en el filtro r.
- **Prueba 6**: Los parámetros de Prueba 4 y Prueba 5 juntos, en el filtro r.
- **Prueba 7**: Los parámetros de Prueba 4 y Prueba 5 juntos, en el filtro r, pero considerando los objetos brillante, es decir, tomando sólo los objetos con un $r < 17$[2].
- **Prueba 8**: Los parámetros de Prueba 4 y Prueba 5, pero eliminando el parámetro *texture*, en el filtro r.

Las pruebas con el filtro u (**Pruebas 1**, **2** y **3**), se eligieron para probar el rendimiento de la red con las dos clases de parámetros propuestos. En este caso se quiere probar si el filtro u, debido a su naturaleza, clasifica bien las galaxias espirales y elípticas. Por otro lado, las pruebas con el filtro r (**Pruebas 4**, **5** y **6**) deben servir para probar si ocupando ese filtro, se obtienen tan buenos resultados como los mostrados en Banerji et al.[1] ocupando el filtro i. Estos últimos resultados no se pueden comparar directamente, ya que Banerji et al. ocupó un sesgo un poco más complejo que el ocupado aquí, ademas de ocupar una variación de la matriz de confusión al entregar los resultados.

Se agregó la prueba de eliminar el parámetro *texture* (**Prueba 8**) y la prueba con el conjunto de objetos brillantes (**Prueba 7**), para mostrar cómo afectan cada una el rendimiento de la red, y compararlas con la **Prueba 6**. La idea de eliminar el parámetro *texture* proviene de analizar el histograma de distribución de los tipos de galaxias en el parámetro en cuestión, mostrados en Banerji et al.[1]. Se puede apreciar que la distribución de los objetos no difiere mucho al considerar *texture*, y lo que se busca analizar es la necesidad de mantener este parámetro o no.

Para cada prueba se procedió a realizar 10 ejecuciones[3] de la red, eligiéndose la última de ellas. Esto

[2] El tamaño de este subconjunto es de 157,919 objetos, cantidad menor a la considerada en las otras pruebas.

[3] Una ejecución corresponde al entrenamiento, validación y testing de una prueba de la red neuronal.

debido al análisis de los valores obtenidos en cada medidor de rendimiento entre las diez ejecuciones, y se llegó a la conclusión de que no diferían demasiado entre cada ejecución como para afectar el rendimiento global de la prueba.

4. Resultados

La presentación de los resultados están organizados de la siguiente forma. En primer lugar se presentan los porcentajes de éxito y de error, MSE alcanzado, el motivo de la detención del entrenamiento, el número máximo de épocas alcanzado, y el tiempo de ejecución de cada prueba. Por otra parte se presenta la matriz de confusión junto al gráfico de comparación, el cual compara la cantidad de objetos de cada clase clasificados por la red neuronal contra la cantidad de objetos clasificados por los usuarios de Galaxy Zoo 1. Debido a esto se puede apreciar el grado de acercamiento a la clasificación objetivo (deseada) por parte de cada prueba de la red neuronal. La ejecución de cada prueba entregó los resultados que se muestran a continuación.

4.1. Prueba 1

Parámetros basados en el color y en los perfiles ajustados de brillo, en el filtro u. Para la prueba 1 los resultados totales del entrenamiento, la validación y testing, fueron los siguientes:

- Porcentaje de éxito: 94,3 %
- Porcentaje de error: 5,7 %
- MSE alcanzado: 0,029
- Motivo de detención del entrenamiento: por validación.
- Número de épocas de entrenamiento alcanzado: 642 épocas.
- Tiempo aprox. de entrenamiento de una ejecución: 29 minutos.

		Galaxy Zoo		
		Espiral	Elíptica	Objeto Desconocido/Estrella
RNA	Espiral	89076	5795	551
	Elíptica	9085	179785	420
	Objeto Desconocido/Estrella	349	186	2650

Cuadro 2: Matriz de Confusión - Prueba 1

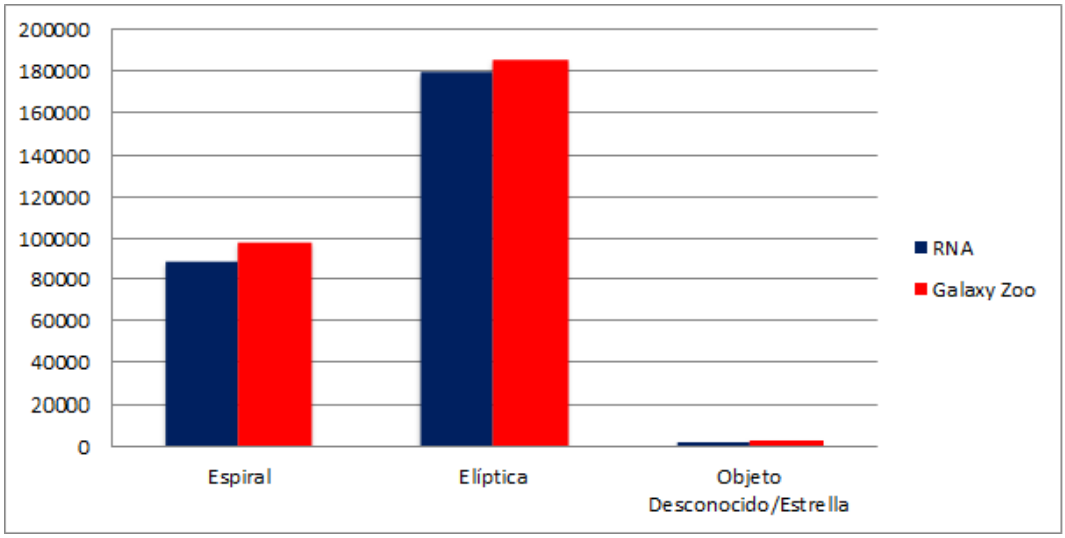

Figura 1: Clasificación Red Neuronal v/s Clasificación GZ1 - Prueba 1

4.2. Prueba 2

Parámetros basados en la forma y textura, en el filtro u. Para la prueba 2 los resultados totales del entrenamiento, la validación y testing, fueron los siguientes:

- Porcentaje de éxito: 89,8 %
- Porcentaje de error: 10,2 %
- MSE alcanzado: 0,053
- Motivo de detención del entrenamiento: por validación.
- Número de épocas de entrenamiento alcanzado: 707 épocas.
- Tiempo aprox. de entrenamiento de una ejecución: 25 minutos.

		Galaxy Zoo		
		Espiral	Elíptica	Objeto Desconocido/Estrella
RNA	Espiral	82036	10174	546
	Elíptica	16418	175371	2051
	Objeto Desconocido/Estrella	56	221	1024

Cuadro 3: Matriz de Confusión - Prueba 2

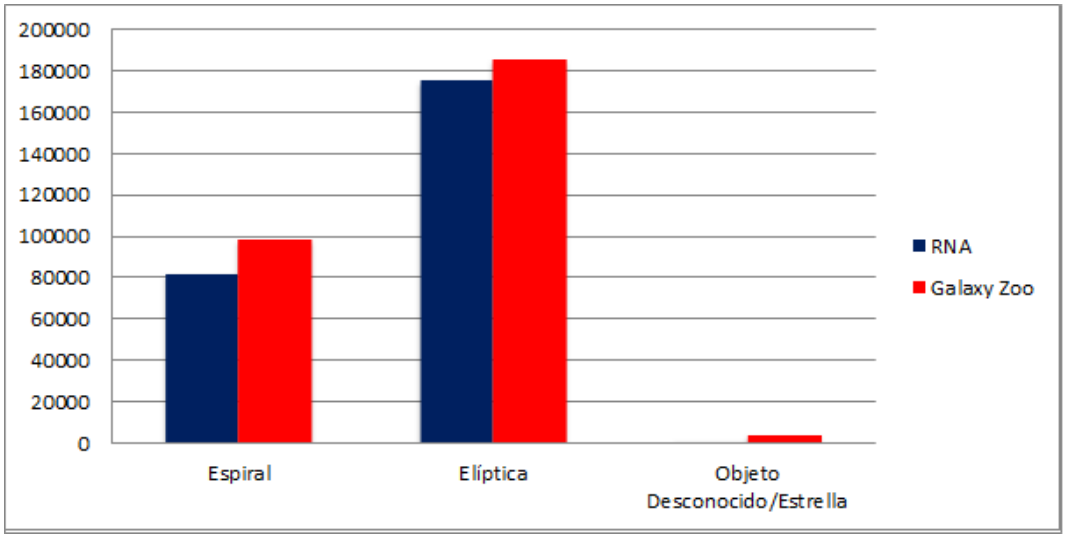

Figura 2: Clasificación Red Neuronal v/s Clasificación GZ1 - Prueba 2

4.3. Prueba 3

Los parámetros de Prueba 1 y Prueba 2 juntos, en el filtro u. Para la prueba 3 los resultados totales del entrenamiento, la validación y testing, fueron los siguientes:

- Porcentaje de éxito: 95,9 %
- Porcentaje de error: 4,1 %
- MSE alcanzado: 0,0215
- Motivo de detención del entrenamiento: por validación.
- Número de épocas de entrenamiento alcanzado: 349 épocas.
- Tiempo aprox. de entrenamiento de una ejecución: 18 minutos.

		Galaxy Zoo		
		Espiral	Elíptica	Objeto Desconocido/Estrella
RNA	Espiral	92053	4418	445
	Elíptica	6209	181175	421
	Objeto Desconocido/Estrella	248	173	2755

Cuadro 4: Matriz de Confusión - Prueba 3

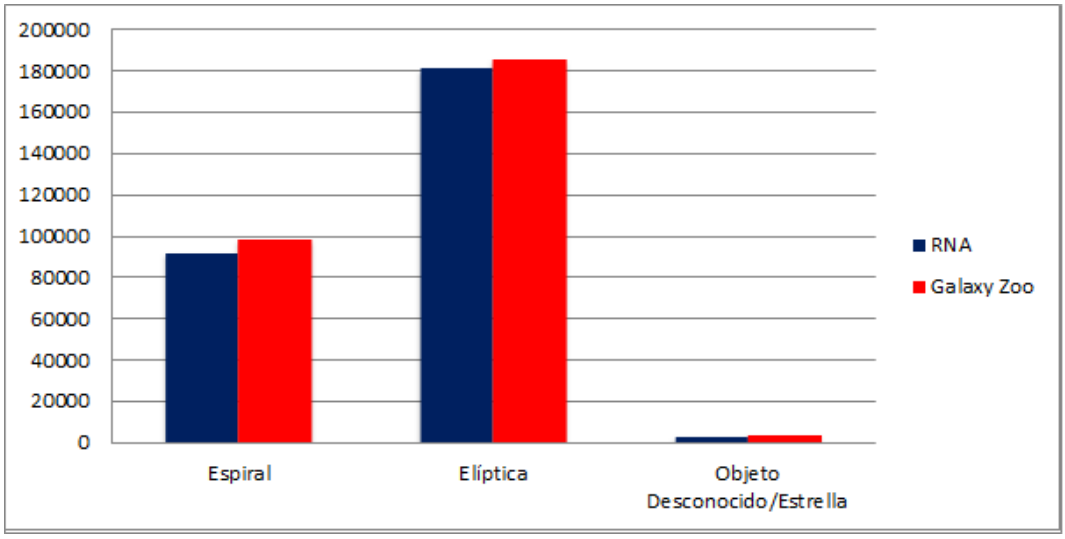

Figura 3: Clasificación Red Neuronal v/s Clasificación GZ1 - Prueba 3

4.4. Prueba 4

Parámetros basados en el color y en los perfiles ajustados de brillo, en el filtro r. Para la prueba 4 los resultados totales del entrenamiento, la validación y testing, fueron los siguientes:

- Porcentaje de éxito: 97,4 %
- Porcentaje de error: 2,6 %
- MSE alcanzado: 0,0133
- Motivo de detención del entrenamiento: por validación.
- Número de épocas de entrenamiento alcanzado: 454 épocas.
- Tiempo aprox. de entrenamiento de una ejecución: 20 minutos.

		Galaxy Zoo		
		Espiral	Elíptica	Objeto Desconocido/Estrella
RNA	Espiral	95104	3135	518
	Elíptica	3024	182482	256
	Objeto Desconocido/Estrella	382	149	2847

Cuadro 5: Matriz de Confusión - Prueba 4

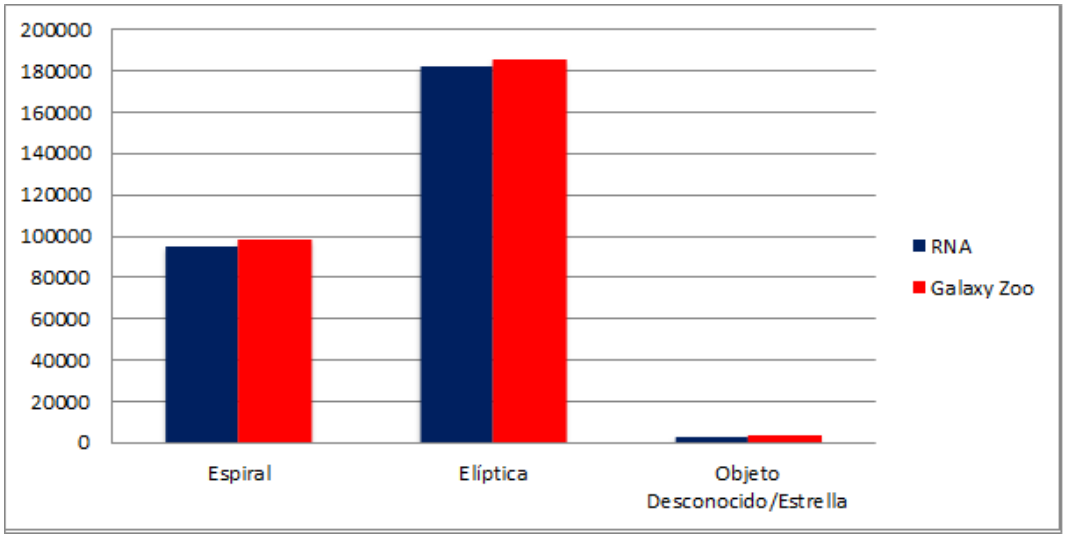

Figura 4: Clasificación Red Neuronal v/s Clasificación GZ1 - Prueba 4

4.5. Prueba 5

Parámetros basados en la forma y textura, en el filtro r. Para la prueba 5 los resultados totales del entrenamiento, la validación y testing, fueron los siguientes:

- Porcentaje de éxito: 95,6 %
- Porcentaje de error: 4,4 %
- MSE alcanzado: 0,0227
- Motivo de detención del entrenamiento: por validación.
- Número de épocas de entrenamiento alcanzado: 440 épocas.
- Tiempo aprox. de entrenamiento de una ejecución: 19 minutos.

		Galaxy Zoo		
		Espiral	Elíptica	Objeto Desconocido/Estrella
RNA	Espiral	91321	4522	515
	Elíptica	6956	181146	358
	Objeto Desconocido/Estrella	233	98	2748

Cuadro 6: Matriz de Confusión - Prueba 5

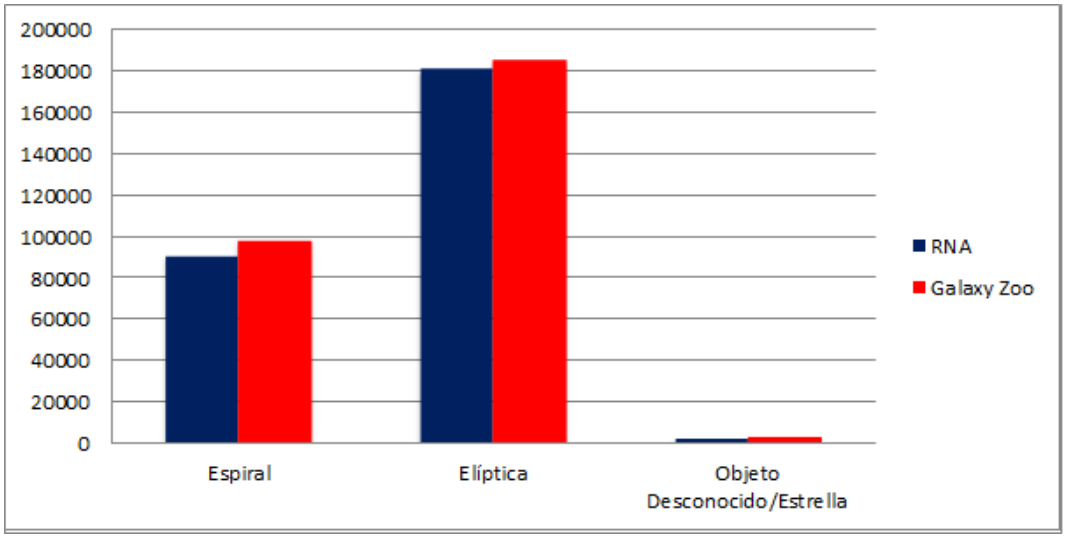

Figura 5: Clasificación Red Neuronal v/s Clasificación GZ1 - Prueba 5

4.6. Prueba 6

Los parámetros de Prueba 4 y Prueba 5 juntos, en el filtro r. Para la prueba 6 los resultados totales del entrenamiento, la validación y testing, fueron los siguientes:

- Porcentaje de éxito: 98,0 %
- Porcentaje de error: 2,0 %
- MSE alcanzado: 0,0105
- Motivo de detención del entrenamiento: por validación.
- Número de épocas de entrenamiento alcanzado: 417 épocas.
- Tiempo aprox. de entrenamiento de una ejecución: 20 minutos.

		Galaxy Zoo		
		Espiral	Elíptica	Objeto Desconocido/Estrella
RNA	Espiral	95810	2481	358
	Elíptica	2470	183177	184
	Objeto Desconocido/Estrella	230	108	3079

Cuadro 7: Matriz de Confusión - Prueba 6

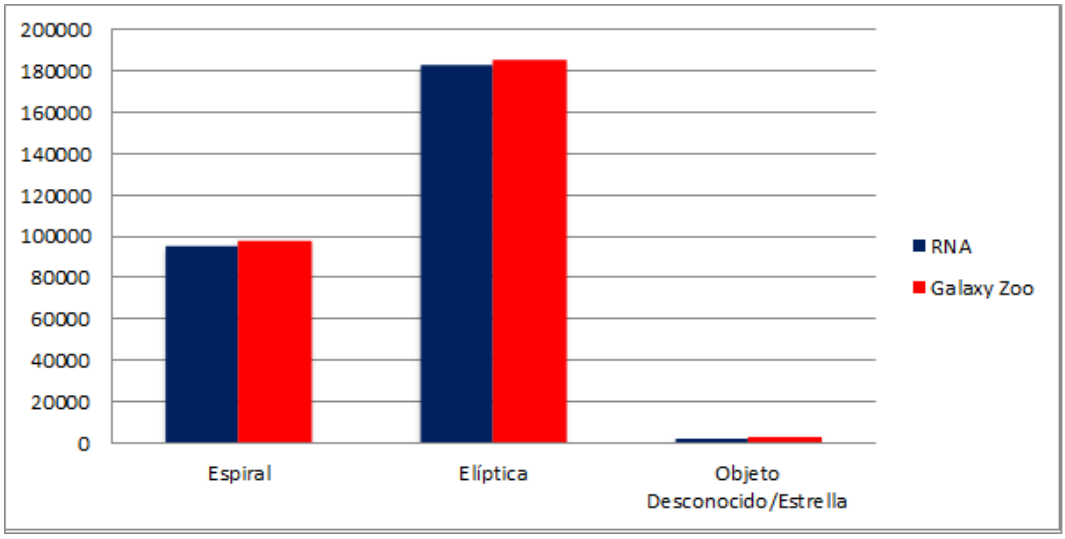

Figura 6: Clasificación Red Neuronal v/s Clasificación GZ1 - Prueba 6

4.7. Prueba 7

Los parámetros de Prueba 4 y Prueba 5 juntos, en el filtro r, pero considerando los objetos brillante, es decir, tomando sólo los objetos con un $r < 17$. Para la prueba 7 los resultados totales del entrenamiento, la validación y testing, fueron los siguientes:

- Porcentaje de éxito: 97,5 %
- Porcentaje de error: 2,5 %
- MSE alcanzado: 0,0130
- Motivo de detención del entrenamiento: por validación.
- Número de épocas de entrenamiento alcanzado: 351 épocas.
- Tiempo aprox. de entrenamiento de una ejecución: 13 minutos.

		Galaxy Zoo		
		Espiral	Elíptica	Objeto Desconocido/Estrella
RNA	Espiral	64591	1837	171
	Elíptica	1760	87582	40
	Objeto Desconocido/Estrella	125	39	1773

Cuadro 8: Matriz de Confusión - Prueba 7

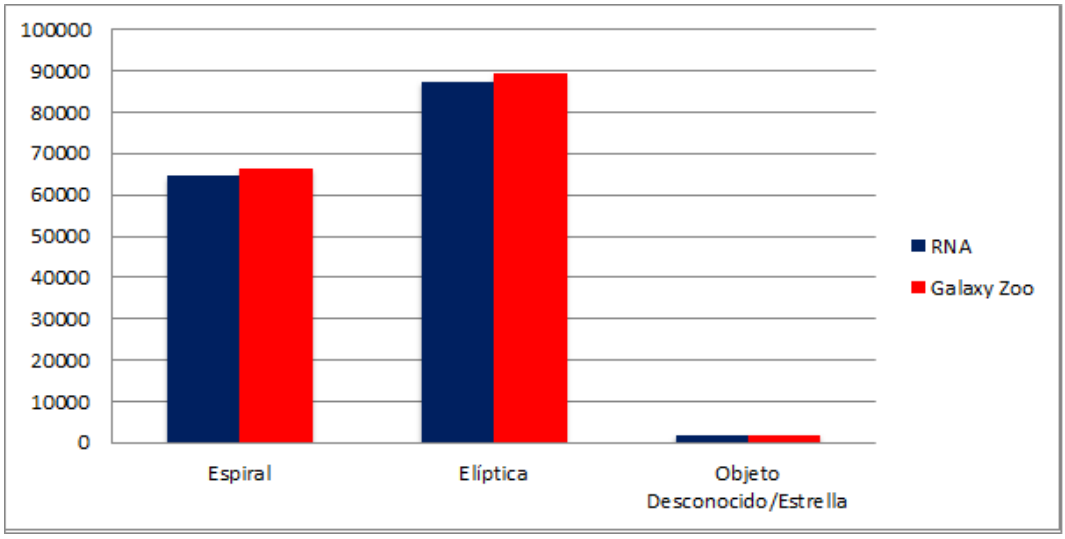

Figura 7: Clasificación Red Neuronal v/s Clasificación GZ1 - Prueba 7

4.8. Prueba 8

Los parámetros de Prueba 4 y Prueba 5, pero eliminando el parámetro *texture*, en el filtro r. Para la prueba 8 los resultados totales del entrenamiento, la validación y testing, fueron los siguientes:

- Porcentaje de éxito: 98,0 %
- Porcentaje de error: 2,0 %
- MSE alcanzado: 0,0130
- Motivo de detención del entrenamiento: por validación.
- Número de épocas de entrenamiento alcanzado: 351 épocas.
- Tiempo aprox. de entrenamiento de una ejecución: 13 minutos.

		Galaxy Zoo		
		Espiral	Elíptica	Objeto Desconocido/Estrella
RNA	Espiral	95913	2558	342
	Elíptica	2402	183119	186
	Objeto Desconocido/Estrella	195	89	3093

Cuadro 9: Matriz de Confusión - Prueba 8

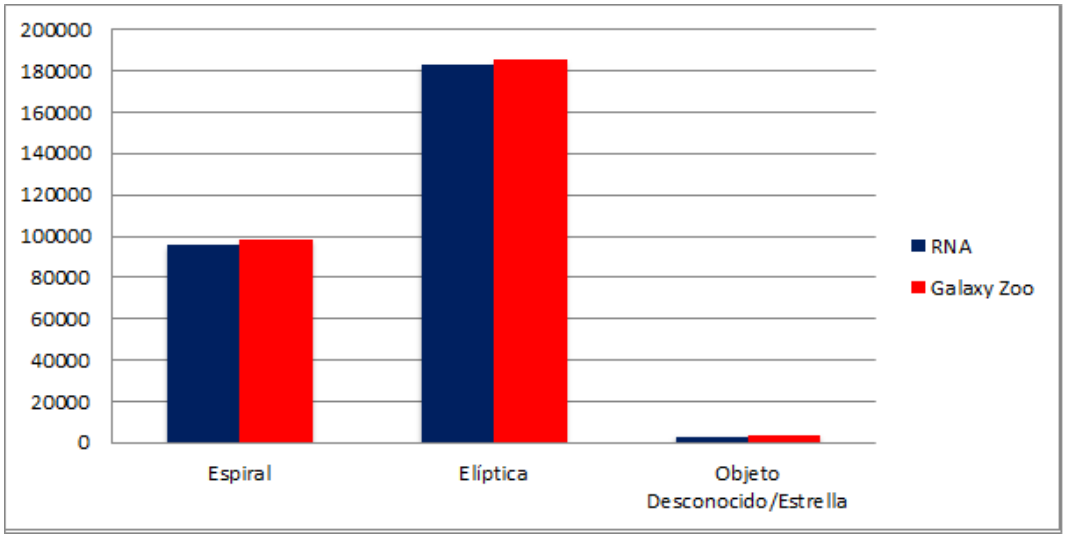

Figura 8: Clasificación Red Neuronal v/s Clasificación GZ1 - Prueba 8

4.9. Resumen

En esta sección se mostraran las tablas resumen de las pruebas efectuadas. Las tablas se presentarán separadas en tres tipos, dependiendo de las características de cada prueba efectuada para su posterior análisis. Cada tipo se describe a continuación.

- Tipo A: Toma las pruebas 1, 2 y 3, las cuales tienen en común el filtro fotométrico u, y se diferencian por los parámetros de entrada a la red.

	Filtro u		
	Prueba 1	Prueba 2	Prueba 3
% de éxito	94,3	89,8	95,9
% de error	5,7	10,2	4,1
MSE	0,029	0,053	0,0215

Cuadro 10: Resumen de Pruebas Tipo A

- Tipo B: Toma las pruebas 4, 5 y 6, que son similares a las pruebas del Tipo A, pero que ocupan la información del filtro r.

	Filtro r		
	Prueba 4	Prueba 5	Prueba 6
% de éxito	97,4	95,6	98,0
% de error	2,6	4,4	2,0
MSE	0,0133	0,0227	0,0105

Cuadro 11: Resumen de Pruebas Tipo B

- Tipo C: Son de este tipo las pruebas 7 y 8, ya que son variaciones a la prueba 6 del Tipo B.

	Filtro r	
	Prueba 7	Prueba 8
% de éxito	97,5	98,0
% de error	2,5	2,0
MSE	0,013	0,010

Cuadro 12: Resumen de Pruebas Tipo C

.

5. Análisis de Resultados

El análisis de resultados de un experimento conlleva a evaluar las similitudes y diferencias entre cada par de pruebas efectuadas. En ese contexto, se procedió a agrupar las pruebas en tres tipos, dependiendo de sus características particulares.

Uno de los primeros análisis, fue ver las diferencias entre las pruebas de Tipo A y Tipo B. Como se puede apreciar en las tablas 10 y 11, las pruebas de Tipo B resultaron mejores que las de Tipo A, significando así que las pruebas realizadas ocupando los parámetros asociados al filtro r fueron superiores a todas las pruebas que ocuparon los parámetros del filtro u. Lo que nos lleva a resumir que la red se comporta de mejor manera con el filtro fotométrico r.

Por otro lado, comparando las pruebas individuales dentro de su mismo tipo, es decir, viendo las pruebas que tienen distintos tipos de parámetros elegidos para el entrenamiento independiente del filtro ocupado, se obtiene lo siguiente. Para las pruebas de Tipo A (Filtro u), la prueba 3 fue la que mejor comportamiento tuvo, sin embargo, la prueba 2 tuvo un pobre desempeño, incluso comparándola con todas las demás pruebas. Esto dice, que por si solos, los parámetros basados en la forma y textura de la galaxia no son tan buenos parámetros en comparación con los basados en el color y perfiles ajustados del objeto. Sin embargo, la unión de estos tipos de parámetros logran una mejora en el rendimiento de la red en comparación con los dos tipos de parámetros por si solos. Lo mismo ocurre con las pruebas de Tipo B (Filtro r), donde si bien la diferencia entre los tipos de parámetros son menores, se mantienen los mejores resultados en la prueba con parámetros basados en color y perfiles ajustados. También se puede ver que la unión de los dos tipos de parámetros nos brinda un mejor rendimiento de la red, al igual que en las pruebas de Tipo A.

Analizando las pruebas de Tipo C, éstas se deben ver por separado y compararlas con respecto a la prueba 6 de Tipo B. Teniendo en cuenta esto último, la prueba 7 obtuvo una diferencia de un $0,5\,\%$ en el porcentaje de éxito con respecto a la prueba 6. Si bien es un porcentaje pequeño, podemos decir que la disminución del set de entrenamiento, debido a la selección de objetos brillantes, contribuye a esta diferencia, pero esto es algo que debiese comprobarse con la realización de otras pruebas adicionales, que en este caso, no es el objetivo del estudio. También podemos decir que la selección de objetos brillantes mantiene en cierta medida los resultados obtenidos en la prueba con los demás objetos con un filtro $r > 17$, por lo que esta selección no aporta a la mejora de rendimiento de la red.

Por otro lado, un resultado interesante tuvo que ver con la prueba 8. El objetivo de esta prueba era verificar si, eliminando el parámetro *texture*, se obtenía un resultado similar o no. En este caso, por lo visto en las tablas 11 y 12, los valores de rendimiento de la redes, en ambas pruebas, 6 y 7, son similares, variando los resultados solamente en la cuarta cifra significativa del MSE. Dado lo revisado en esta comparación, se puede decir que el parámetro *texture*, no contribuye a la clasificación que ocupa parámetros asociados al filtro r, por lo que se puede prescindir de él.

6. Documentación del código fuente

A continuación se describirá la serie de pasos para ejecutar el código fuente de las pruebas efectuadas durante el trabajo efectuado.

Primeramente se extrae el Sample principal desde el sitio de CasJobs[4], mediante una consulta SQL, la cual se muestra a continuación:

```
SELECT p.objID, p.dered_u, p.dered_g, p.dered_r, p.dered_i, p.dered_z,
       p.petroR50_u, p.petroR90_u, p.mE1_u, p.mE2_u, p.deVAB_u, p.expAB_u,
       p.lnLExp_u, p.lnLDeV_u, p.lnLStar_u, p.mRrCc_u, p.mCr4_u, p.texture_u,
       p.petroR50_r, p.petroR90_r, p.mE1_r, p.mE2_r, p.deVAB_r, p.expAB_r,
       p.lnLExp_r, p.lnLDeV_r, p.lnLStar_r, p.mRrCc_r, p.mCr4_r, p.texture_r,
       m.p_el, m.p_dk, m.p_mg, m.p_cs

FROM DR7..PhotoObjAll p, Votos m

WHERE p.objID=m.dr7objid and (m.p_el > 0.8 or m.p_dk > 0.8 or m.p_cs > 0.8)
and m.p_mg < 0.8 and p.dered_u != -9999 and p.dered_g != -9999 and
p.dered_r != -9999 and p.dered_i != -9999 and p.dered_z != -9999 and

p.petroR50_u != -9999 and p.petroR90_u != -9999 and p.mE1_u != -9999 and
p.mE2_u != -9999 and p.deVAB_u != -9999 and p.expAB_u != -9999 and
p.lnLExp_u != -9999 and p.lnLDeV_u != -9999 and p.lnLStar_u != -9999 and
p.mRrCc_u != -9999 and p.mCr4_u != -9999 and p.texture_u != -9999 and

p.petroR50_i != -9999 and p.petroR90_i != -9999 and p.mE1_i != -9999 and
p.mE2_i != -9999 and p.deVAB_i != -9999 and p.expAB_i != -9999 and
p.lnLExp_i != -9999 and p.lnLDeV_i != -9999 and p.lnLStar_i != -9999 and
p.mRrCc_i != -9999 and p.mCr4_i != -9999 and p.texture_i != -9999 and

p.petroR50_r != -9999 and p.petroR90_r != -9999 and p.mE1_r != -9999 and
p.mE2_r != -9999 and p.deVAB_r != -9999 and p.expAB_r != -9999 and
p.lnLExp_r != -9999 and p.lnLDeV_r != -9999 and p.lnLStar_r != -9999 and
p.mRrCc_r != -9999 and p.mCr4_r != -9999 and p.texture_r != -9999;
```

En la consulta se ocupan los votos extraídos desde Galaxy Zoo[5], los cueles se deben subir a la base de datos personal que otorga CasJobs a cada usuario. Además se eliminan las galaxias que tiene datos nulos (los cuales en la base de datos tiene el valor −9999). Una vez hecho esto se tiene el Main Sample a ocupar para cada prueba. El formato ocupado para la extracción de estos datos fue de Valores separados por coma (CSV). Para efectos de este documento se otorgará el set de datos Main Sample, ya extraído desde CasJobs.

[4] http://skyserver.sdss3.org/CasJobs/
[5] http://data.galaxyzoo.org/

Posterior a ello se procede a normalizar los datos para cada una de las 8 pruebas en el entorno MATLAB. Además dentro del mismo script [6] se incluye la elección de los subset de datos para cada prueba, dependiendo de los parámetros que requiere cada una. Cada uno de los ocho scripts recibe de entrada el archivo `MainSample.csv`, y entrega dos archivo CSV, uno con los parámetros requeridos para cada prueba[7], y otro archivo `salida.csv` que entrega los votos que se le asignó en el proyecto Galaxy Zoo a cada galaxia. Esto dos archivos servirán para el entrenamientos de las redes neuronales de cada prueba, como entrada y salida respectivamente.

En los scripts de las redes, se modificó el archivo estándar que entrega MATLAB para poder adaptarlas a los requerimientos de nuestra investigación. En cada script de red neuronal, primeramente se cargan los archivos de entrada y salida de la red, y posteriormente se ejecuta la prueba, la cual entrega los resultados del entrenamiento, la validación y el testing de cada una. Estos resultados los entrega como matriz de confusión, además de otras opciones de visualización, como gráfico de performance, ROC, entre otros. Para efectos de este trabajo se utilizó la matriz de confusión entregada en cada prueba y el valor de performance MSE.

La estructura de los ficheros entregados en esta documentación es la siguiente:

- ☐ Prueba 1
  - ↳ Prueba1.m
  - ↳ Red1.m
  - ↳ MainSample.csv
- ☐ Prueba 2
  - ↳ Prueba2.m
  - ↳ Red2.m
  - ↳ MainSample.csv
- ☐ Prueba 3
  - ↳ Prueba3.m
  - ↳ Red3.m
  - ↳ MainSample.csv
- ☐ Prueba 4
  - ↳ Prueba4.m
  - ↳ Red4.m
  - ↳ MainSample.csv

[6] El script lleva por nombre `PruebaN.m`, donde `N` es el número de la prueba donde se efectúa.
[7] El nombre del archivo dependerá de cada prueba.

- ☐ Prueba 5
  - ↳ Prueba5.m
  - ↳ Red5.m
  - ↳ MainSample.csv
- ☐ Prueba 6
  - ↳ Prueba6.m
  - ↳ Red6.m
  - ↳ MainSample.csv
- ☐ Prueba 7
  - ↳ Prueba7.m
  - ↳ Red7.m
  - ↳ MainSample_Brillante.csv[8]
- ☐ Prueba 8
  - ↳ Prueba8.m
  - ↳ Red8.m
  - ↳ MainSample.csv
- ☐ ConsultaCasjobs.sql
- ☐ MainSample.csv
- ☐ MainSample_Brillante.csv
- ☐ Tabla_UR_Completa.csv[9]

El orden de ejecución de cada prueba se mostrará en el siguiente ejemplo. Si queremos ejecutar la prueba 1, se ejecuta el script `Prueba1.m`, y posteriomente el script `Red1.m`, esto se puede emular en todas las demás pruebas. El tiempo de ejecución por prueba es de alrededor de 20 minutos, pero como cada prueba se repitió un número determinado de veces, en nuestro caso fueron diez[10], este tiempo fue de $3,3$ horas por prueba aproximadamente. Las pruebas del trabajo se ejecutaron en una maquina Intel Core i5-2415M de 4 núcleos de 2.30 GHz, con 10 GB de memoria RAM, en sistema operativo Windows 7 Ultimate 64 bits.

[8] El archivo Main Sample Brillantes se utiliza en la prueba 7, lo razón se explica en el plan de pruebas.

[9] Este archivo contiene los nombres de las cabeceras de columna del Main Sample, que se ocupa sólo de referencia y respaldo.

[10] Esta cantidad de veces se controla en el script de cada red mediante un ciclo `for`, y para elegir un resultado se compararon las diez ejecuciones, de las cuales se tomo la ejecución promedio.

7. Conclusiones

El trabajo efectuado permitió profundizar en el estudio teórico y práctico de las redes neuronales artificiales, además de llevar ese conocimiento al trabajo en el ámbito astronómico. Por otra parte, el conocer acerca del trabajo de los astrónomos ha resultado ser una experiencia alucinante que motiva a profundizar más en su campo.

Con respecto a las pruebas, los resultados de ellas nos han otorgado conclusiones interesantes acerca del uso de ciertos parámetros, como el color, los perfiles ajustados de brillo, la forma y la textura, para el entrenamiento de la redes neuronales. Una de las conclusiones vista en el sub-capítulo de análisis de resultados, es acerca de los parámetros asociados al filtro r, y es que estos se comportan de mejor manera que los parámetros asociados al filtro u. Por otro lado, el rendimiento de las redes, que ocupan parámetros basados en color y perfiles ajustados de brillo, es mejor que el rendimiento de las que usan los parámetros de forma y textura. Otro resultado, y no menos importante, es que la variación de volumen del set de entrada no influye demasiado en el rendimiento global de la red que ocupa parámetros asociados al filtro r, lo cual se puede ver en la prueba con el set de datos de objetos brillantes. Por último, uno de los resultados más interesantes, es el visto en la prueba en cual se eliminaba el parámetro *texture*, que mide la gama de fluctuaciones en el brillo de la superficie de los objetos astronómicos. Este parámetro no aporta mayor información, a la red neuronal, que ayude a clasificar los objetos como galaxias de algún tipo, por lo que se puede prescindir de él para obtener la clasificación de galaxias ocupando parámetros asociados al filtro r. La eliminación del parámetro *texture* asociado a los demás filtros, se deja para una futura investigación.

Estos resultados obtenidos son importantes en términos de la clasificación de galaxias, ya que nos aseguran que ocupando un filtro fotométrico adecuado, del sistema *ugriz* de SDSS, y eligiendo los parámetros de entrada presentados en este trabajo, podemos obtener una buena clasificación de este tipo de objetos. A su vez, nos pone en un buen camino para desarrollar una máquina de aprendizaje que permita distinguir fusiones de galaxias tomando como base el trabajo realizado, lo cual era uno de los principales objetivos de esta investigación.

8. Trabajos Futuros

Como se demostró, las redes neuronales artificiales poseen un gran potencial para clasificar objetos astronómicos, como las galaxias. Esta ventaja se podría ocupar para otro tipo de clasificación o detección dentro del campo de la astronomía. Por ejemplo la identificación de fusiones de galaxias asoma como un gran desafío, ya que para los astrónomos, detectar esta mezcla de galaxias ha sido de gran dificultad. Se sabe que se han hecho algunos avances, como lo mostrado en Darg et al.[2][3][4], con respecto a fusiones y multi-fusiones de galaxias, lo cual invita a investigar sobre la posible ayuda que puedan ofrecer las máquinas de aprendizaje en la detección automática de estos objetos.

En el ámbito de esta investigación, un posible trabajo sería mostrar el efecto de la elección de un umbral distinto en la clasificación de GZ1, lo cual podría generar importantes cambios en el rendimiento de la red neuronal. Esto porque, es decisión del investigador elegir el umbral adecuado para realizar sus pruebas, lo que podría provocar distintos rendimientos en la clasificación dependiendo del umbral elegido. Entonces, un posible trabajo sería cambiar el umbral en base a ciertos criterios, como lo visto en Banerji et al.[1]. Otra de las posibles propuestas sería repetir este experimento utilizando el catálogo del proyecto Galaxy Zoo 2, lo cual permitiría añadir otro tipo de parámetros que podrían mejorar la clasificación. Por otro lado, el procesamiento de las imágenes de galaxias sería otro de los trabajos a efectuar en el futuro, ya que se podría analizar y elegir otra clase de parámetros distintos a los ocupados en este trabajo. Por otro lado, se podrían realizar las mismas pruebas de este trabajo, pero considerando el algoritmo de entrenamiento de Levenberg-Marquardt. Ésto, para asegurar que converja la red neuronal en un tiempo menor al ocupado en las pruebas de este trabajo.

Referencias

[1] Banerji, M., Lahav, O., et al., *"Galaxy Zoo: Reproducing galaxy morphologies via machine learning"*. Monthly Notices of the Royal Astronomical Society (MNRAS),Vol. 406, pag. 342-353, 2010.

[2] Darg, D., Kaviraj, S., Lintott, C., et al., *"Galaxy Zoo: the fraction of merging galaxies in the SDSS and their morphologies"*. Monthly Notices of the Royal Astronomical Society (MNRAS),Vol. 401, pag. 1043-1056, 2010.

[3] Darg, D., Kaviraj, S., Lintott, C., et al., *"Galaxy Zoo: the properties of merging galaxies in the nearby Universe-local environments, colours, masses, star formation rates and AGN activity"*. Monthly Notices of the Royal Astronomical Society (MNRAS),Vol. 401, pag. 1552-1563, 2010.

[4] Darg, D., Kaviraj, S., Lintott, C., et al., *"Galaxy Zoo: multimergers and the Millennium Simulation"*. Monthly Notices of the Royal Astronomical Society (MNRAS),Vol. 416, pag. 1745-1755, 2011.

[5] Provost, F., Fawcett, T., Kohavi, R., *"The case against accuracy estimation for comparing induction algorithms "*. Proceedings of the Fifteenth International Conference on Machine Learning, 1997.

[6] Visa, S., Ramsay, B. et al., *"Confusion Matrix-based Feature Selection"*. Midwest Artificial Intelligence and Cognitive Science Conference, USA, 2011.