

Abstracto

En este documento se explicará la metodología utilizada en la implementación de redes neuronales feedforward FNN, convolucionales CNN y recurrentes RNN para solucionar problemas de datos estructurados, espaciales y secuenciales o temporales.

1. Introducción

El problema 1 consiste en Clasificación de objetos (estrella, galaxia o cuásar) utilizando datos científicos obtenidos del observatorio Apache Point de Nuevo México. El set de datos forma parte del proyecto Sloan Digital Sky Survey SDSS. Dirección del set de datos es <https://www.kaggle.com/datasets/fedesoriano/stellar-classification-dataset-sdss17> y cuenta con 100,000 registros. El problema 2 consiste en utilizar un set de 1178 imágenes para detectar carros en una carretera (<https://www.kaggle.com/datasets/sshikamaru/car-object-detection>). Y el problema 3 consiste en realizar una predicción del precio del bitcoin utilizando los precios de Ethereum, Cardano, Binance Coin, TetherUSD, SP 500, Nasdaq, Dow Jones y el Oro.

2. Clasificación de objetos (estrella, galaxia o cuásar)

Las variables utilizadas de este set de datos son: ángulo de ascensión derecho α , ángulo de

declinación δ , filtro ultravioleta en sistema fotométrico u , filtro verde g , filtro rojo r , filtro cerca del ultravioleta i , filtro ultravioleta z y el valor de corrimiento al rojo basado en el incremento de longitud de onda $redshift$. Se eliminaron las variables que identificaban el registro de la base de datos o que no aportaban información al modelo.

La columna "class" se convirtió a one hot encoding para identificar con un 1 o 0 si el objeto es una estrella, galaxia o cuásar. Para el entrenamiento, se separó el 20% para validación y el otro 80% se separó para entrenamiento y validación.

El entrenamiento se realizó utilizando la librería *Keras*, el modelo utilizado fue *Sequential* con el optimizador *Adagrad*, el parámetro ϵ se estableció a $1e-07$ y para calcular el valor de la métrica $f1score$ se utilizó un *custom callback* con la formula $f1score = (2 * (recall * precision) / (recall + precision))$, en donde denominador es igual a $recall + precision$.

Los datos se normalizaron, por este motivo en el modelo de *keras* se utilizó la función de activación de tangente hiperbólica, para poder determinar esta función se realizaron varios experimentos demostrando ser la mejor opción. El modelo final utilizado consistió en 2 capas internas de 8 y 6 neuronas, se utilizó la capa de regularización *Dropout* y la función *softmax* en la capa de salida para clasificación múltiple. En la fase de validación de resultados, el parámetro $f1-score$ fue de 96%.

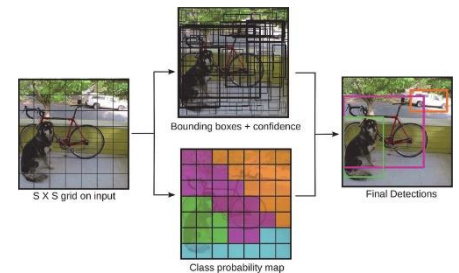
3. Detección de carros en imágenes

Las imágenes analizadas son del tamaño 676x380 pixeles de resolución, eran imágenes en movimiento e incluían más de un carro por imagen.



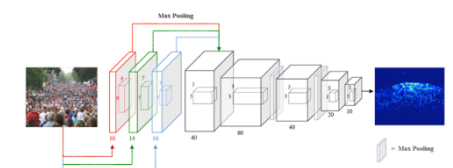
Las imágenes del set de datos muestran más de un carro por imagen (múltiples coordenadas).

Dado que una imagen podía contener más de un carro, el algoritmo debe separar la imagen en celdas y analizar de forma individual cada una, determinando si contiene o no un carro.



Análisis convolucional utilizando mallas.
Algoritmo YOLO

En la fase de experimentación se utilizó una red convolucional simple con tres capas paralelas y cuatro convolucionales con *MaxPooling2D* y *BatchNormalization*, la arquitectura utilizada es la siguiente:



Arquitectura convolucional simple utilizada en la fase de experimentación.

Seguido se utilizó una capa artificial para poder analizar los datos resultantes de la capa convolucional.

4. Predicción del precio del Bitcoin

Para realizar esta predicción se descargaron dos años de historial de las criptomonedas *Cardano ADA*, *Binance Coin BNB*, *Ethereum ETH*, *Theter USD USDT* y de los índices de valores *Down Jones*, *S&P 500*, *NASDAQ* y *oro*. De cada uno, se analizó el precio de cierre diario y el volumen.

Aset	ULR
BTC	https://finance.yahoo.com/quote/BTC-USD/history/
ETH	https://finance.yahoo.com/quote/ETH-USD/history/
ADA	https://finance.yahoo.com/quote/ADA-USD/history/
BNB	https://finance.yahoo.com/quote/BNB-USD/history/
USDT	https://finance.yahoo.com/quote/USDT-USD/history/
DJ	https://finance.yahoo.com/quote/%5EDJI/history
S&P 500	https://finance.yahoo.com/quote/%5EGSPC/history
NASDAQ	https://finance.yahoo.com/quote/%5EIXIC/history
Oro	https://es.finance.yahoo.com/quote/GC%3DF/history

Tabla de origen de datos.

Los datos cargados en formato *CSV* se ordenaron de forma ascendente respecto a la fecha y se normalizaron con la formula $(valor - min)/(max - min)$ con el objetivo de tener valores de 0 a 1.

Debido a la variabilidad del origen de datos (más de una fecha de un registro faltaba), se utilizaron solo los registros que compartían la misma fecha para poder realizar el entrenamiento recurrente.

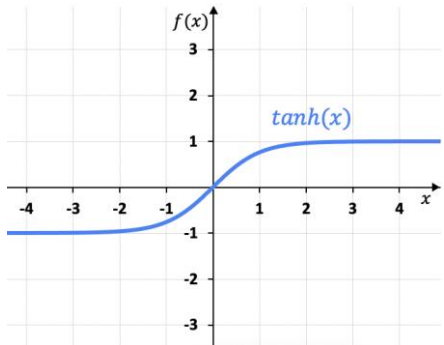
El entrenamiento con *keras* se utilizaron cuatro capas *LSTM Long Short-Term Memory network*, con la función de activación *relu* y 128 neuronas y cuatro capas *FeedForward* con 64, 32, 16 y 8 neuronas. La función de perdida utilizada fue *MSE* y el parámetro épsilon de 0.000005 configurado en un *custom callback*.

5. Resultados

Los resultados del entrenamiento de las diferentes redes neuronales se muestran a continuación.

5.1 FNN

Al utilizar la función de activación de tangente hiperbólica para datos con codominio de -1 a 1 los resultados de la predicción fueron mejores.



Función de tangente hiperbólica.

El modelo permite ingresar los datos obtenido del observatorio y predecir si es una galaxia, estrella o cuásar. La siguiente predicción indico que el objeto tiene una probabilidad de 0.98395 de ser galaxia, 0.011795 de ser cuásar y 0.004255 de ser estrella.

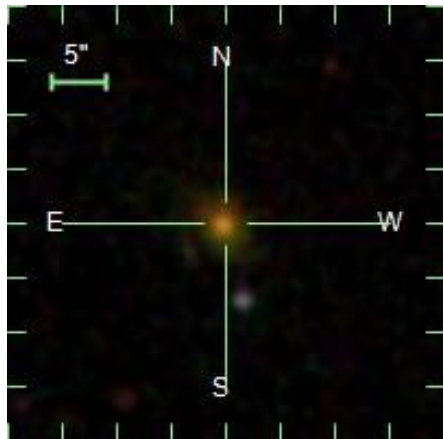
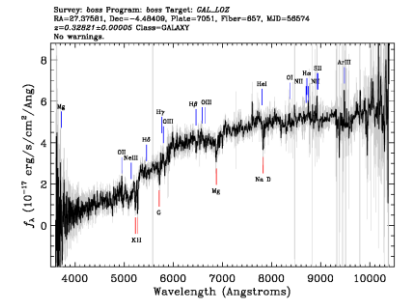


Imagen de la galaxia analizada.

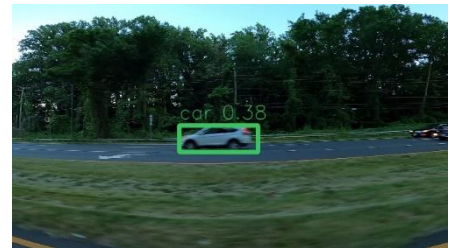
Utilizando *SQL* desde el sitio oficial del [proyecto \(http://skyserver.sdss.org/dr17/\)](http://skyserver.sdss.org/dr17/) es posible obtener la imagen real y el espectro de los objetos de estudio.



Espectro de frecuencias de la galaxia analizada.

5.2 CNN

Al implementar el algoritmo de detección de objetos *YOLO You Only Look Once*, se lograron detectar los diferentes objetos que estaban presentes en cada imagen, obteniendo la probabilidad y clase de cada objeto en cuestión.



Predicción del algoritmo YOLO.

Es necesario profundizar en el diseño de redes neuronales para poder crear una arquitectura desde cero que permita identificar si un objeto está en la imagen, que tipo de objeto es (probabilidad) y sus coordenadas.

5.3 RNN

El error cuadrático medio de los datos de validación fue de *MSE*: 0.0118 y la predicción para los siguientes tres días del precio de bitcoin fue 18,547.4 USD, 23,205.16 USD y 18,547.4 USD. Estos resultados sugieren un intento de romper la resistencia, pero regresará al soporte establecido.