

Identificación de partículas captadas por un detector de radiación Cherenkov por medio de una red neuronal

Mayra Betsabé Silva Carranza / sil17276@uvg.edu.gt, Juan F. Mancilla-Caceres / juanfernandomansilla@gmail.com, Luis Guillermo Mijangos Fuentes / lgmijangos@uvg.edu.g

RESUMEN: Desde el espacio se nos está bombardeando constantemente con partículas sub-atómicas que tienen su origen en fenómenos físicos extremadamente interesantes y misteriosos. Estas partículas son llamadas rayos cósmicos. Cuando estos rayos interactúan con nuestra atmósfera producen una *cascada* de partículas secundarias que son detectables y que traen información muy útil de los fenómenos que las produjeron. Como estas partículas son muy diversas es extremadamente importante tener la capacidad para clasificarlas para su estudio. En este artículo presentaremos la una herramienta de clasificación de partículas captadas por un detector de radiación Cherenkov de agua. La tecnología que se usó es *Machine Learning*, o más concretamente, una red neuronal. Estas estrategias de Inteligencia Artificial están ganando mucha popularidad en el gremio científico y por eso consideramos de vital importancia para el científico guatemalteco adquirir estas destrezas que son de gran ayuda en el mundo moderno. Como estos algoritmos requieren entrenamiento, se utilizaron datos provenientes de simulaciones de estas cascadas de partículas generadas por las aplicaciones CORSIKA y Geant4 (utilizados en laboratorios de renombre como CERN y *Fermilab*). Además de los datos de la simulación también se utilizaron datos de detección del tanque Cherenkov de agua que está instalado en el campus central de la Universidad del Valle de Guatemala. Una porción de estos datos se usó como datos de entrenamiento de la red neuronal y otra porción se usó para validar los resultados. Las partículas elegidas para este estudio fueron las 7 más comunes en la latitud, longitud y altitud sobre el nivel del mar de Ciudad de Guatemala, las cuales son: muón, electrón, positrón, anti muón, protón, pión y anti-pión. La precisión de la red neuronal en la clasificación de muones fue de 0.905 comparado contra los 0.813 del algoritmo *Random Forest*, y el valor del *Recall* para muones para nuestra red fue de 0.729 comparado con el 0.537 para *Random Forest*. Los resultados muestran que la red neuronal utilizada en este estudio es significativamente mejor clasificando la mayoría de las partículas del experimento si se le compara contra los algoritmos *Naive Bayes*, *Random Forest* y *Regresión Logarítmica*.

PALABRAS CLAVE: partículas, *Machine Learning*, Cherenkov, Geant4, red neuronal, cascadas de área extensa, rayos cósmicos, radiación Cherenkov.

The use of animals in research and teaching in Guatemala: exploring the perception of animal welfare in the veterinary academic community

ABSTRACT: From outer space we are being flooded by subatomic particles that are originated by mysterious and fascinating physical phenomenon. Those particles are called Cosmic Rays. When those Cosmic Rays interact with Earth's atmosphere, they produce a *shower* of secondary particles that are detectable. Those new particles carry a lot of useful information from the phenomenon that originate them. These secondary particles are extremely diverse, and because of that is extremely important to have the tools to classify which type of particle is the one that is traveling through the detector tank. In this article we will present one tool capable of performing that task. The tool implemented is called a *Neural Network* and it belongs to the very popular field of *Machine Learning*. These tools of Artificial Intelligence are gaining popularity in the Scientific Community and because of that we believe that it is of vital importance to Guatemalan and Latin American Scientists to acquire the capabilities to use them. These powerful algorithms require validated data that will be used to train the network because the network need to *learn* how to classify the information by itself. To provide that training data, we used the data provided by simulations implemented in the computer programs CORSIKA and Geant4. Those programs are used in state-of-the-art laboratories around the world such as CERN and *Fermilab*. Also, there were used data obtained from the Water Cherenkov Detector installed in the main campus of Universidad del Valle de Guatemala. One portion of the detected data was used as training data and another one was used as validating data. The chosen particles for this study were the 7 most common particles that can be detected in Guatemala City's latitude, longitude, and elevation over sea-level. Those 7 particles are: muons, electrons, positrons, anti-muons, protons, pions, and anti-pions. The accuracy of the Neural Network classification was 0.905 that is significantly better than the 0.813 obtained by an algorithm called *Random Forest*. The *Recall* value (a very important parameter of a classification algorithm) for our Network was 0.729. If you compare it to the 0.537 from *Random Forest*, you get a 36% improvement. The performance of our Neural Network was significantly better than the other algorithms used to this type of task, such as *Naive Bayes*, *Random Forest*, and *Logarithmic Regression*.

KEY WORDS: Sub-atomic particles, *Machine Learning*, Cherenkov, Geant4, *Neural Network*, Cosmic Showers, Cosmic Rays, Cherenkov Radiation.

Introducción

La radiación Cherenkov es un fenómeno que ocurre cuando una partícula cargada eléctricamente atraviesa un medio viajando a una rapidez superior a la velocidad de la luz en el vacío. Esta radiación se utiliza para poder detectar lluvias de rayos cósmicos (Obodovskiy, 2004). Para este fin se construye un tanque de agua, completamente aislado de la luz con fotomultiplicadores que registran el paso de una partícula generando una señal de voltaje que tiene una forma específica que depende, entre otras cosas, del tipo de partícula que pasó por el tanque. Estas partículas altamente energéticas provienen de las cascadas de área extensa, que a su vez son el resultado de la interacción de rayos cósmicos con la atmósfera terrestre (Morison, 2008). Se estima que cada minuto, cada metro cuadrado de la superficie terrestre es bombardeado por cerca de 10,000 muones (Wolverton, 2007)

Cuando un detector de rayos cósmicos percibe una partícula genera una gran cantidad de datos. Según nuestra experiencia, durante una hora de funcionamiento del tanque Cherenkov de agua instalado en el Campus Central de la Universidad del Valle de Guatemala, se registran alrededor de doscientos mil eventos asociados a partículas. Cada evento es una curva que consta de 12 valores numéricos. La forma de estas curvas es extremadamente diversa y está asociada a muchas variables, entre las cuales, una de las principales es **el tipo de partícula** que está siendo detectada. Por eso es extremadamente importante poder determinar qué partículas son las que están siendo detectadas, y para ello se implementaron y evaluaron varios modelos de *Machine Learning*. De esta manera el software será capaz de determinar de una manera rápida y efectiva la naturaleza de la cascada cósmica detectada.

Esta tecnología de detección y clasificación puede ser aplicada en una gran variedad de campos, como, por ejemplo: la muongrafía, el estudio del clima solar, la caracterización de la radiación de fondo, física de partículas de altas energías, entre otros. Con respecto a los muones y su detección, se encuentran varias aplicaciones basadas en la absorción del flujo de muones cósmicos como lo son: análisis geológicos de plantas nucleares e inspecciones arqueológicas (Checchia, 2016)

Materiales y Métodos

Cascadas de Aire Extensas y Simulaciones

Las partículas primarias presentes en un rayo cósmico, al incidir en la atmósfera e interactuar con sus moléculas dan lugar a las cascadas de aire extensa (EAS por sus siglas en inglés *Extensive Air Shower*). Las partículas que componen las cascadas se les conoce como partículas secundarias.

Muchas de estas partículas tienen vidas medias muy breves, por lo que decaen, produciendo a su vez otras partículas secundarias. Como resultado de estos continuos decaimientos, se obtiene una cascada formada por nuevas partículas con menor energía (Durán, 2011)

Para modelar este fenómeno físico, se cuentan con programas que permiten simular el decaimiento de las partículas y así generar la composición exacta de las cascadas. Se tiene por un lado el software diseñado por el CERN llamado *Geant4*, el cual tiene como objetivo la simulación de la trayectoria de partículas a través de la materia (CERN, 2021). Dicha herramienta se utilizará para simular la respuesta de nuestro tanque detector una vez que la partícula secundaria ingresa en él. Otro recurso que se utilizó en esta investigación fue CORSIKA, el cual es un programa que fue diseñado para la simulación de cascadas de aire extensas originadas por partículas provenientes de rayos cósmicos de altas energías (KIT, 2021)

Clasificación automática de partículas en cascadas de Aire Extensas

En 2016 un grupo de estudiantes de la Universidad del Valle de Guatemala diseñó e implementó un detector de radiación Cherenkov de agua para la toma y procesamiento de datos provenientes de rayos cósmicos, figura 1. Dicho tanque estuvo operacional en forma no continuada desde 2016 hasta 2018 generando varias decenas de gigabytes de datos de detecciones de partículas secundarias de las cascadas. Uno de los aspectos requeridos por este proyecto de investigación es la clasificación de los eventos, de modo que al detectar una partícula dentro del tanque se sea capaz de determinar con mucha precisión la naturaleza de la partícula detectada.

En el trabajo de graduación de 2016 se aplicaron algoritmos de clasificación de aprendizaje supervisado y no supervisado. Para la *Clasificación No Supervisada* se utilizó *K-Means* y para la *Clasificación Supervisada* se utilizó el algoritmo *Naive Bayes*. Es de esta forma que surge la necesidad de comparar e implementar un método de clasificación de más partículas puesto que anteriormente se trabajó con electrones de 37 MeV, muones de 4 GeV y rayos gamma de 100 MeV (Conde et al, 2016). En el caso de esta investigación, se buscaba identificar el método que presentara el mejor rendimiento, tomando en cuenta suposiciones como que los eventos entre las partículas son dependientes y no independientes como se asume en (Conde, Duque, et al, 2016)

Descripción de datos

A partir de una simulación creada en CORSIKA para conocer la distribución típica de partículas para Ciudad de Guatemala, se generaron alrededor de doscientos mil eventos en una

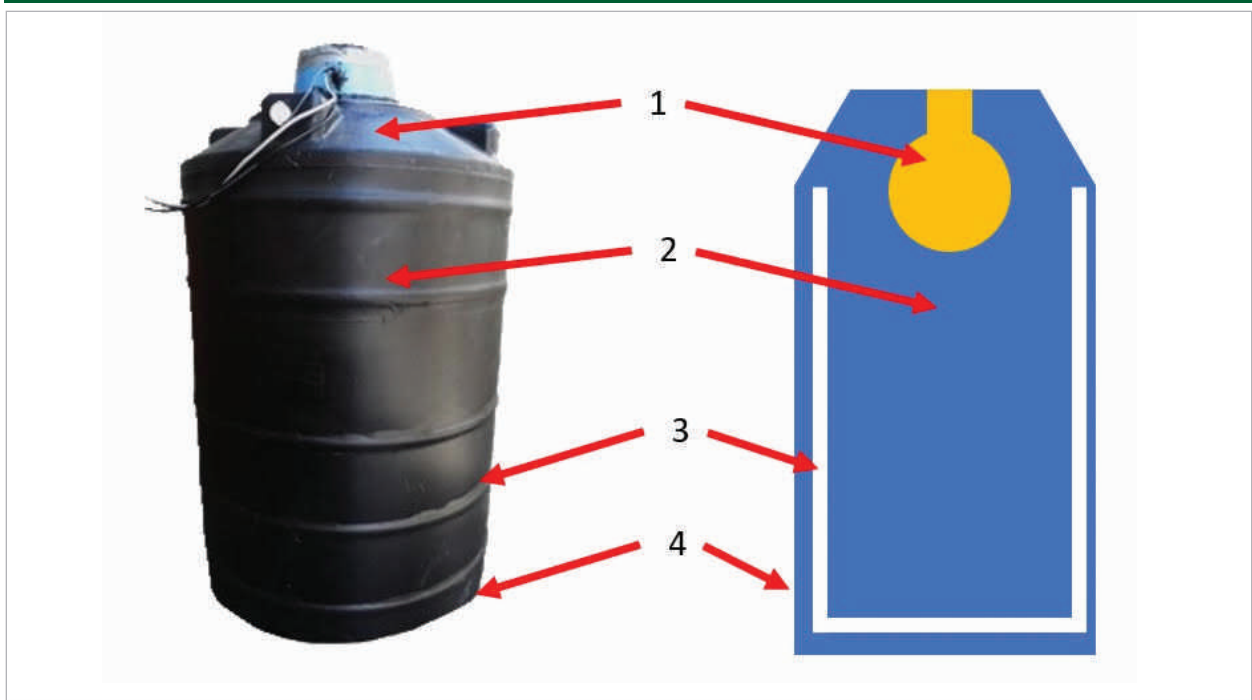


Figura 1. Tanque detector de radiación cósmica Cherenkov construido e implementado en el campus central de UVG. Las partes más importantes: (1) Tubo fotomultiplicador (PMT), (2) Agua, (3) Cubierta reflectante de Tyvek, (4) Recubrimiento oscuro de fibra de carbono para evitar filtraciones de luz.

simulación de un tanque de radiación Cherenkov de agua implementada en *Geant4*. Dicha simulación del tanque contiene los materiales y parámetros del tanque que se encuentre en el campus de la Universidad del Valle. A continuación, se muestran los objetos utilizados en *Geant4* para la simulación del tanque:

- **Tyvek:** material reflectivo que recubre el interior del tanque con forma cilíndrica, con una altura de 57.1 cm y un radio de 40.1 m.
- **Agua**
- **Tubo fotomultiplicador y un fotocatodo:** se seleccionó los materiales de Borosilicato y Aluminio y se les asignó un radio de 11.1 cm.

Se delimitó el área de estudio a las siete partículas más abundantes en la simulación de *CORSIKA*, las cuales son: muón, anti muón, positrón, electrón, protón, pión y pión negativo. Para cada tipo de partícula, se contaba con las curvas detectadas por el simulador del tanque.

Resultados y discusión

Clasificación de partículas

En esta sección describimos los algoritmos utilizados para clasificar las partículas, así como los resultados obtenidos con cada algoritmo.

- **Naive Bayes.** Es un modelo de clasificación supervisada generativo, que asume la independencia condicional entre cada una de las variables de entrada. Este es uno de los modelos que fue utilizado en la construcción del tanque detector de rayos cósmicos que fue instalado en el Campus Central de la Universidad del Valle de Guatemala y del cual se habla en (Conde, Duque, et al, 2016).

Se aplicó el algoritmo de *Naive Bayes*, con el propósito de poder comparar y evaluar el método que se adaptaba mejor al conjunto de datos. En el cuadro 1 se muestran los resultados.

Cuadro 1. Métricas obtenidas luego de evaluar *Naive Bayes*.

| Clase | Precisión | Recall | F-1 |
|-----------|-----------|--------|-------|
| Anti-muón | 0.550 | 0.356 | 0.432 |
| Anti-pión | - | 0 | - |
| Electrón | 0.795 | 0.803 | 0.799 |
| Muón | 0.482 | 0.613 | 0.540 |
| Pión | 0 | 0 | 0 |
| Positrón | 0.214 | 0.748 | 0.422 |
| Protón | 0.016 | 0 | 0 |

Cuadro 2. Matriz de confusión obtenida al evaluar *Naive Bayes*.

| | Electrón | Anti-muón | Protón | Positrón | Muón | Pión | Anti-pión |
|-----------|----------|-----------|--------|----------|------|------|-----------|
| Electrón | 3906 | 0 | 0 | 955 | 3 | 0 | 0 |
| Anti-muón | 331 | 0 | 68 | 2694 | 6145 | 95 | 0 |
| Protón | 14 | 1565 | 0 | 2260 | 334 | 0 | 0 |
| Positrón | 291 | 303 | 426 | 3031 | 0 | 0 | 0 |
| Muón | 359 | 2246 | 13 | 1291 | 6181 | 0 | 0 |
| Pión | 5 | 49 | 17 | 53 | 69 | 0 | 0 |
| Anti-pión | 10 | 52 | 11 | 30 | 89 | 0 | 0 |

Como se observa el rendimiento no es bueno. Esto es porque las suposiciones para la utilización de este algoritmo no se cumplen. Para este tipo de clasificador, todos los atributos se clasifican como independientes unos de los otros, así como también se considera que cada contribución de los atributos es igual para todos. Ambos aspectos no se cumplen para el conjunto de datos dado debido a que cada valor de voltaje es dependiente de la posición y el tiempo y por lo mismo, no es el método adecuado.

En la matriz de confusión, mostrada en el cuadro 2, también se puede observar que para las categorías de muón y anti-muón no se clasifican bien estas partículas, además cabe resaltar que no se tiene una matriz diagonal. Es de especial atención ver que, en la matriz de confusión, el algoritmo no es capaz de clasificar los piones, así como también se tiene un rendimiento pobre para los muones positivos.

Considerando lo anterior, se decidió crear nuevas variables independientes, tales como: el voltaje máximo y la energía. A continuación, se presentan en el cuadro 3 los resultados,

luego de haber aplicado nuevamente *Naive Bayes* al conjunto de datos. Se puede observar cierta mejoría con respecto a la aplicación de *Naive Bayes* anterior, basándose en el valor de precisión se tiene que, para los muones negativos, electrones, protones si hay una mejor precisión. Sin embargo, a la matriz de confusión no es una una matriz diagonal. Por lo que se puede concluir que *Naive Bayes* no es el mejor algoritmo para utilizar con este conjunto de datos.

Regresión logística

La regresión logística también es uno de los modelos utilizados (Conde et al. 2016). Este algoritmo supone que las variables no están correlacionadas entre sí. En el cuadro 4 se muestran las métricas de evaluación del algoritmo luego de aplicar una regresión logística para la clasificación de datos.

Si se compara con el algoritmo de *Naive Bayes* la precisión para la clasificación de partículas aumentó. Sin embargo, al observar la matriz de confusión en el cuadro 5 se puede ver que se sigue sin poder clasificar e identificar a los piones, y

Cuadro 3. Métricas obtenidas al evaluar *Naive Bayes* con variables independientes.

| Clase | Precisión | Recall | F-1 |
|-----------|-----------|--------|-------|
| Anti-muón | 0.707 | 0.322 | 0.442 |
| Anti-pión | 0 | - | - |
| Electrón | 0.314 | 0.115 | 0.169 |
| Muón | 0.767 | 0.514 | 0.289 |
| Pión | 0 | - | - |
| Positrón | 0.616 | 0.6489 | 0.603 |
| Protón | 0.120 | 0.147 | 0.132 |

Cuadro 4. Métricas obtenidas al evaluar *regresión logística*.

| Clase | Precisión | Recall | F-1 |
|-----------|-----------|--------|-------|
| Anti-muón | 0.621 | 0.792 | 0.696 |
| Anti-pión | - | 0 | - |
| Electrón | 0.811 | 0.768 | 0.789 |
| Muón | 0.571 | 0.402 | 0.472 |
| Pión | - | 0 | - |
| Positrón | 0.584 | 0.775 | 0.668 |
| Protón | 0.729 | 0.440 | 0.549 |

Cuadro 5. Matriz de confusión obtenida al evaluar *regresión logística*.

| | Electrón | Anti-muón | Protón | Positrón | Muón | Pión | Anti-pión |
|-----------|----------|-----------|--------|----------|------|------|-----------|
| Electrón | 3737 | 70 | 0 | 955 | 102 | 0 | 0 |
| Anti-muón | 58 | 11471 | 415 | 134 | 2412 | 0 | 0 |
| Protón | 303 | 1071 | 1836 | 2953 | 10 | 0 | 0 |
| Positrón | 63 | 215 | 167 | 3139 | 467 | 0 | 0 |
| Muón | 401 | 5410 | 85 | 140 | 4054 | 0 | 0 |
| Pión | 23 | 112 | 15 | 17 | 26 | 0 | 0 |
| Anti-pión | 22 | 132 | 1 | 8 | 29 | 0 | 0 |

no se clasifica bien a los protones. Esto es porque, nuevamente no se cumplen las suposiciones: para este algoritmo los datos deberían ser independientes unos de otros y este no es el

caso. Además, es recomendable aplicar este algoritmo para casos de clasificación binaria y en este caso se tiene una clasificación de multiclases.

Cuadro 6. Matriz de confusión obtenida al evaluar *Random Forest*.

| | Electrón | Anti-muón | Protón | Positrón | Muón | Pión | Anti-pión |
|-----------|----------|-----------|--------|----------|------|------|-----------|
| Electrón | 4864 | 0 | 0 | 0 | 0 | 0 | 0 |
| Anti-muón | 175 | 13768 | 0 | 101 | 446 | 0 | 0 |
| Protón | 14 | 1635 | 2517 | 0 | 7 | 0 | 0 |
| Positrón | 728 | 88 | 53 | 3182 | 0 | 0 | 0 |
| Muón | 430 | 7478 | 0 | 94 | 2088 | 0 | 0 |
| Pión | 7 | 138 | 0 | 25 | 23 | 0 | 0 |
| Anti-pión | 20 | 140 | 0 | 16 | 16 | 0 | 0 |

Cuadro 7. Métricas obtenidas al evaluar la *red neuronal*.

| Clase | Precisión | Recall | F-1 |
|-----------|-----------|--------|-------|
| Anti-muón | 0.817 | 0.941 | 0.875 |
| Anti-pión | 1.0 | 0.063 | 0.118 |
| Electrón | 0.974 | 0.100 | 0.989 |
| Muón | 0.905 | 0.729 | 0.808 |
| Pión | 0.900 | 0.028 | 0.055 |
| Positrón | 0.902 | 0.947 | 0.923 |
| Protón | 0.869 | 0.836 | 0.852 |

Random Forest

En el cuadro 6 se muestran los resultados luego de aplicar el algoritmo *Random Forest* al conjunto de datos, comparándolos con los resultados de la red neuronal. Los algoritmos de *Random Forest* tienden a sobreajustar el modelo dado que solo es posible para el usuario modificar los hiperparámetros. Si los hiperparámetros se dejan sin ajustar se puede lograr una precisión hasta del 99% pero ese valor sería a todas luces engañoso. Si se usa un solo árbol de decisión se crearían reglas que se basan en valores específicos de voltaje para clasificar cada tipo de partícula, pero la naturaleza de los datos lleva bastante aleatoriedad. Las partículas, al decaer dentro del tanque no registran los mismos y exactos valores de voltaje en el PMT para cada tipo de partícula. Este es el caso que considera tan solo un árbol de decisión, y para *Random Forest* se tienen que utilizar cerca de cien árboles de decisión, lo cual en definitiva llevaría al sobreajuste del modelo. Nótese en el cuadro 6 que nuevamente, el algoritmo falla en el momento de clasificar los piones, a pesar de que esta matriz tiene una forma más parecida a una matriz diagonal.

Método de Clasificación: red neuronal

Debido a las debilidades encontradas en los métodos descritos anteriormente, se decidió implementar una red neuronal que pueda capturar las características necesarias de los datos. En particular, se decidió implementar la red de tal forma que tome en cuenta el orden de los valores de voltaje (considerándolos como series de tiempo).

Para la red neuronal se utilizó un modelo secuencial el cual consta de un conjunto de capas donde cada capa tiene como entrada y salida un tensor. Tiene un total de 10 capas, donde 3 de ellas son de normalización, utilizando la capa *Batch Normalization de Keras*. Estas tres capas se encargan de mantener el promedio de los datos de salida cercano a 0 y la desviación estándar de los mismos cercano a 1, esto ayuda a evitar el sobreajuste. La última capa de la red neuronal incluye una función de activación tipo sigmoide. Simplemente se aplica la función sigmoide a cada entrada. Cabe mencionar que los valores de salida están comprendidos en el intervalo de (0,1).

Cuadro 8. Matriz de confusión obtenida al evaluar la *red neuronal*.

| | Electrón | Anti-muón | Protón | Positrón | Muón | Pión | Anti-pión |
|-----------|----------|-----------|--------|----------|------|------|-----------|
| Electrón | 5632 | 114 | 36 | 0 | 5 | 5 | 453 |
| Anti-muón | 147 | 19432 | 3846 | 8 | 96 | 116 | 137 |
| Protón | 12 | 455 | 10517 | 0 | 110 | 150 | 374 |
| Positrón | 58 | 84 | 30 | 6852 | 0 | 8 | 0 |
| Muón | 0 | 0 | 0 | 0 | 17 | 0 | 0 |
| Pión | 0 | 0 | 0 | 0 | 1 | 9 | 0 |
| Anti-pión | 100 | 565 | 0 | 0 | 43 | 31 | 4917 |

Cuadro 9. Cuadro comparativo entre las métricas del mejor algoritmo de *Machine Learnig* utilizado en estudios previos (*Random Forest*) contra la red neuronal implementada en este estudio

| Clase | Red Neuronal | | | <i>Random Forest</i> | | |
|-----------|--------------|--------|-------|----------------------|--------|--------|
| | Precisión | Recall | F-1 | Precisión | Recall | F-1 |
| Anti-muón | 0.817 | 0.941 | 0.875 | 0.694 | 0.903 | 0.784 |
| Anti-pión | 1.0 | 0.063 | 0.118 | - | 0 | - |
| Electrón | 0.974 | 0.100 | 0.989 | 0.845 | 1 | 0.916 |
| Muón | 0.905 | 0.729 | 0.808 | 0.813 | 0.537 | 0.3647 |
| Pión | 0.900 | 0.028 | 0.055 | - | 0 | - |
| Positrón | 0.902 | 0.947 | 0.923 | 0.952 | 0.869 | 0.909 |
| Protón | 0.869 | 0.836 | 0.852 | 0.990 | 0.730 | 0.840 |

Para la compilación del modelo se utilizó un modelo de pérdida de entropía cruzada (para datos categóricos). La cual en este caso se encargó de hacer un promedio de la distribución de probabilidades entre la diferencia del valor real y el valor predicho. En cuanto al parámetro de optimización, se utilizó la clase *Adam* para esta tarea porque se ajusta bien para problemas con bastantes datos o parámetros.

Luego de haber entrenado la red neuronal, se probó su rendimiento con el conjunto de datos de prueba. En los cuadros 7 y 8 se muestran los resultados de las métricas y la matriz de confusión, respectivamente.

Como se puede observar en el cuadro 8, el algoritmo de clasificación implementado sí es capaz de clasificar piones. Otro aspecto importante por resaltar de estos resultados es que la matriz de confusión, en efecto cumple con que el elemento de la diagonal para cada tipo de partícula es el

valor máximo de cada fila. Como son de especial interés los muones, se puede ver que, en efecto, son las partículas que el algoritmo es capaz de identificar en su mayoría.

Comparando los resultados de las métricas obtenidas en el cuadro 8 con los resultados luego de aplicar *Naive Bayes* del cuadro 1 se tiene que hay un incremento en la precisión para la clasificación de partículas, lo cual implica que la red neuronal es una mejoría para el análisis de partículas planteado en trabajos anteriores.

Haciendo referencia a la clasificación de muones en el cuadro 9, se tiene que, para el algoritmo de *Random Forest* la precisión es mayor, teniendo un 0.813 comparado con un 0.905 de la *red neuronal* lo cual es un indicador de que, si el algoritmo clasifica una partícula como muón, hay mayor probabilidad de que en efecto sea muón. También, al comparar el valor de *Recall* (el cual indica la fracción de clases que el modelo es capaz de identificar) para ambos

algoritmos, se tiene que hay un valor mayor para la red neuronal en comparación a *Random Forest*. Este resultado implica que el algoritmo de *Random Forest*, deja pasar una mayor cantidad de muones y no los clasifica como tal.

La mejora en los resultados se debe a que la red neuronal logra capturar las dependencias entre los valores de voltaje de mejor manera que los otros algoritmos. Como trabajo futuro se planea utilizar este algoritmo para clasificar nuevos datos obtenidos del detector de agua Cherenkov de la Universidad del Valle de Guatemala con la nueva tarjeta digitalizadora la cual tomará más del doble de muestras por evento y comparar así con las distribuciones esperadas.

Conclusiones

- Con la construcción de la red neuronal se pudo clasificar las partículas con base en el pulso que generan luego de ser captadas por el tubo fotomultiplicador de un detector Cherenkov de agua instalado en el Campus Central de la Universidad del Valle de Guatemala. Esta red neuronal, cumple con que al ver la matriz de confusión el elemento de la diagonal es el valor máximo de cada fila. A pesar de tener un conjunto de datos de entrenamiento (provenientes de la simulación) bastante desbalanceado, la red neuronal sí es capaz de detectar las partículas cuya presencia es mínima en el conjunto de datos.
- Se logró afinar el procedimiento utilizado anteriormente para la clasificación de partículas. Se tuvo una mejora en la precisión y en el *recall* para todas las partículas del estudio si se le compara contra el algoritmo *Random Forest*, el algoritmo *Naive Bayes* y el de *Regresión Logística*. Se recomienda profundizar el estudio entrenando a la red con una mayor cantidad de datos medidos y volver a obtener las métricas para mejorar la comparación, lo cual se logrará cuando el tanque vuelva a estar operativo.

Bibliografía

CERN. (2021) *Geant4: A simulation toolkit*. Extraído de: <https://geant4.web.cern.ch/>.

Cecchia, P. (2016). *Principles of Artificial Neural Networks: Basic Designs to Deep Learning* Review of possible applications of cosmic muon tomography. *Journal of Instrumentation* 11 (12): C12072-C12072. doi: 10.1088/1748-0221/11/12/c12072

Conde, D., Duque, P., Guarcaz, K., et al. (2016) *Diseño, construcción y caracterización de Detector de Radiación Vavilov-Cherenkov de Agua*. Guatemala: Universidad del Valle de Guatemala.

de León, R. (2012) *Informe final: Simulación de un detector Cherenkov de Agua*. Guatemala: Universidad de San Carlos de Guatemala.

Durán, M. (2011). *Instalación de un Detector Cherenkov de Agua para la detección de trazas de Rayos cósmicos a 956 metros sobre el nivel del mar*. Colombia: Bucamaranga

Graupe, D. (2019) *Principles of Artificial Neural Networks: Basic Designs to Deep Learning* World Scientific

KIT (2021) *CORSIKA: an Air Shower Simulation Program* Extraído de: <https://www.iap.kit.edu/corsika/> [11/03/2021]

Morrison, I. (2008) *Introduction to astronomy and cosmology* Wiley.

Obodovskiy, I. (2019) *Radiation: Fundamentals, Applications, Risks and Safety* Elsevier.

Wolverton, M. (2007). *Muons for Peace*. *Scientific American* 297 (3): 26-28. <http://www.jstor.org/stable/26069483>

Aporte de autores

Autor 1 (Mayra Silva): Estuvo a cargo de la experimentación, generación de datos, pruebas con algoritmos, análisis de datos y redacción del artículo.

Autor 2 (Juan Mancilla): Diseñó algoritmos para automatizar el proceso de generación de datos, contribuyó con la red neuronal y la revisión de esta, revisión del artículo y supervisor de los resultados.

Autor 3 (Luis Mijangos): Contribuyó en la obtención de datos, revisión del artículo en la parte de física, aportes en la explicación de resultados, facilitó contactos de soporte para las simulaciones utilizadas. (Geant4 y CORSIKA). También contribuyó en la redacción y estilo de este artículo.