

Proyecto final: Análisis de clasificadores de información.

Cristofher Solís, José David Soto
cristofhersj@gmail.com, josedavidsz@hotmail.com
Área académica de Ingeniería Mecatrónica
Instituto Tecnológico de Costa Rica

Resumen

El presente proyecto tiene como fin el demostrar los conocimientos adquiridos sobre los diferentes paradigmas de inteligencia artificial estudiados en clase, más específicamente para poder resolver problemas que involucren clasificación. Se ha estudiado en qué consiste la clasificación de datos y a su vez se ha hecho de múltiples formas utilizando varios paradigmas distintos. En este caso se tiene el objetivo de poner a prueba estos conocimientos al realizar un algoritmo que sea capaz de clasificar los datos de un censo disponible en la base de datos del repositorio UCI para machine learning, seguidamente se explica el funcionamiento de la clasificación realizada con la ayuda de un árbol de decisión que permite distinguir cuáles entradas poseen mayor importancia que otras a la hora de realizar dicha clasificación y una vez realizado eso, se vuelve a ejecutar el mismo ejercicio con otro paradigma de inteligencia artificial distinto con el fin de comparar la efectividad entre ellos y concluir cuál es el que realiza la clasificación de mejor forma.

Palabras clave

1.Algoritmo , 2.Clasificación, 3.Censo, 4. Redes Neuronales, 5.Python

I. INTRODUCCIÓN

El presente proyecto plantea la resolución de un problema de ingeniería mediante la selección de dos paradigmas de Inteligencia Artificial vistos a lo largo del curso. Primeramente se plantea presentar debidamente el problema (referente a clasificación), posteriormente se realiza el análisis respectivo del mismo y se detallan las razones del porqué se decide emplear uno de los paradigmas de inteligencia artificial. Luego de la debida justificación se describe el diseño y solución implementados, los resultados obtenidos, así como el respectivo análisis de estos. Finalmente se agregan las principales conclusiones del presente trabajo.

El *set* de datos, proviene de la base de dato UCI para machine learning y consta de un censo en el cuál se describen múltiples características de personas, como edad, trabajo, estado civil, lugar de procedencia, entre otros y los divide entre los que poseen un salario mayor o menor que 50 000 dólares, la meta de la red neuronal es lograr recrear esa división de la mejor forma posible a partir de las distintas entradas. [1]

En el presente curso se ha estudiado de múltiples formas, la ciencia detrás de las redes neuronales o sistemas conexionistas y el cómo se pueden interpretar datos a raíz de los resultados dados por ciertos diagnósticos como las curvas de entrenamiento y aprendizaje, o la matriz de confusión para verificar qué tanto se acerca el modelo diseñado a una inteligencia artificial capaz de resolver problemas de clasificación, estos conocimientos adquiridos son de gran utilidad ya que en este proyecto se muestra un *set* de datos complejo, el cual posee 14 entradas distintas y una sola salida, se plantea el realizar esta clasificación por medio de una red neuronal clásica, monitorizando los distintos parámetros que han sido estudiados como el learning rate, optimizadores, número de neuronas, y aspectos referentes a la topología como cantidad de neuronas por capa. De esta forma se realiza un estudio de parámetros para alcanzar una solución factible para el problema dado.

Una vez adquiridos los resultados dados por la clasificación de la red neuronal, es requerido un análisis especial con el fin de averiguar cuáles parámetros poseen mayor peso -importancia- para la clasificación que los demás. De esta manera se puede conocer el funcionamiento en profundidad de los parámetros del censo. Seguidamente se plantea la implementación de un árbol de decisiones empleando los atributos determinados como más importantes para la clasificación. El árbol fue realizado mediante técnicas computacionales y se estudiaron las combinaciones de hiper parámetros que den mejores resultados (en términos de precisión).

Se repite el experimento del primer punto, utilizando otro paradigma de inteligencia artificial, con el fin de comparar los resultados obtenidos -con la red neuronal- y definir cuál de los dos casos logra realizar la clasificación de manera más precisa. El paradigma implementado corresponde a un bosque aleatorio de clasificación, y se realiza un estudio de parámetros para determinar la mejor combinación de hiper parámetros para luego así realizar una comparación entre los resultados de la red neuronal clásica con los del bosque aleatorio para así determinar qué paradigma es más adecuado para realizar la clasificación de los datos utilizados.

II. PRESENTACIÓN DEL PROBLEMA

Primeramente, resulta imprescindible dar una breve explicación del conjunto de datos empleado, pues es importante contextualizar la naturaleza del ejercicio. El conjunto de datos original fue extraído de la base de datos de un censo estadounidense de 1994, el objetivo principal es predecir si una persona tiene ingresos superiores o inferiores a \$50 000 al año. Para esto se cuenta con 14 parámetros de entrada, dentro de los cuales hay datos categóricos y también de carácter numérico continuo, además se cuenta con 1 parámetro de salida que corresponde a la clasificación de ' $> 50K$ ' -para salarios mayores a \$50000 - y ' $\leq 50K$ ' -para salarios menores a \$50000. En la tabla I se resumen los atributos de entrada, en donde se detalla si la característica es de tipo continua y en caso de ser tipo categórica, se detallan las categorías que puede tomar.

Tabla I: Atributos de entrada y su respectiva clasificación

Nombre del atributo	Valor del atributo
Age	Continuous
Workclass	Private, Self-emp-not-inc, Self-emp-inc, Federal-gov, Local-gov, State-gov, Without-pay, Never-worked.
Fnlwgt	Continuous
Education	Bachelors, Some-college, 11th, HS-grad, Prof-school, Assoc-acdm, Assoc-voc, 9th, 7th-8th, 12th, Masters, 1st-4th, 10th, Doctorate, 5th-6th, Preschool.
Education-num	Continuous
Marital-Status	Married-civ-spouse, Divorced, Never-married, Separated, Widowed, Married-spouse-absent, Married-AF-spouse.
Occupation	Tech-support, Craft-repair, Other-service, Sales, Exec-managerial, Prof-specialty, Handlers-cleaners, Machine-op-inspct, Adm-clerical, Farming-fishing, Transport-moving, Priv-house-serv, Protective-serv, Armed-Forces
Relationship	Wife, Own-child, Husband, Not-in-family, Other-relative, Unmarried.
Race	White, Asian-Pac-Islander, Amer-Indian-Eskimo, Other, Black.
Sex	Female, Male.
Capital-gain	Continuous
Capital-loss	Continuous
Hours-per-week	Continuous
Native-country	United-States, Cambodia, England, Puerto-Rico, Canada, Germany, Outlying-US(Guam-USVI-etc), India, Japan, Greece, South, China, Cuba, Iran, Honduras, Philippines, Italy, Poland, Jamaica, Vietnam, Mexico, Portugal, Ireland, France, Dominican-Republic, Laos, Ecuador, Taiwan, Haiti, Columbia, Hungary, Guatemala, Nicaragua, Scotland, Thailand, Yugoslavia, El-Salvador, Trinidad&Tobago, Peru, Hong, Holand-Netherlands

Los datos para el presente proyecto son recuperados del repositorio para Machine Learning de UCI. Se tienen un total de 48842 datos en total, los cuales están de forma predeterminada divididos en 32561 datos para entrenamiento y 16263 para pruebas. En donde se tiene que la relación de categorías está dada por: aproximadamente 75 % presentan un salario menor a \$50000 y el resto (25 %) tienen ingresos superiores \$50000.

Teniendo claro que el problema tratado es de clasificación, ya que la salida es discreta -y en este caso consta de dos categorías- se procede a continuación a hacer el análisis y justificación del paradigma de inteligencia artificial considerado como más adecuado para la tarea.

III. ANÁLISIS DEL PROBLEMA

III-1. Red neuronal clásica: Desde su concepción, las redes neuronales han sido caracterizadas por su capacidad de poder realizar tareas de regresión y clasificación. Esto se debe a la lógica detrás del concepto de redes neuronales; el ser humano es capaz de reconocer patrones y utilizarlos para notar similitudes entre objetos, pues las neuronas en conjunto, al combinarse en una red neuronal, son capaces de resolver problemas complejos, dentro de las operaciones complejas que son capaces de manejar, se encuentran tanto la regresión como la clasificación [2]. Considerando que el problema planteado es de índole clasificatoria, se optó por considerar este paradigma de inteligencia artificial: sistemas conexionistas clásicos.

Con base al teorema de Cybenko, se conoce que cualquier tipo de problema puede ser resuelto por una red neuronal, pese a desconocer la topología y relación de pesos que la describan [3]. Sin embargo, esta característica de presentar la posibilidad de servir como mapeadores universales es un argumento a favor de emplear este tipo de inteligencia artificial para la resolución del problema. Por tanto, se escoge a las redes neuronales clásicas como el paradigma más adecuado para realizar la predicción referente a los ingresos en función de los distintos atributos mencionados previamente.

III-2. Árbol de decisión: Los árboles de decisión son representaciones gráficas utilizadas para la resolución de problemas, se suelen utilizar principalmente en modelos predictivos, la funcionalidad consiste en graficar todos los posibles sucesos que pueden ocurrir de la toma de una decisión y de los sucesos que derivan de las anteriores en forma de cascada. En *machinelearning*, los árboles de decisiones representan un paradigma de inteligencia artificial, donde el algoritmo creado se somete a un entrenamiento previo y a partir de este, genera relaciones entre los distintos parámetros de entrada para relacionarlos con las salidas y se decanta por la solución más eficaz posible. Los árboles de decisión como el que aparece en la figura 1 se componen por distintas partes las cuales son:

- **Nodos:** se da en el momento en el que se plantea una disyuntiva entre varias opciones.
- **Flechas:** son las conexiones entre los nodos
- **Vectores:** cada vector representa la opción escogida tras la pregunta planteada por el nodo.
- **Etiquetas:** unen nodos y flechas, además de nombrar las acciones que se ejecutan.

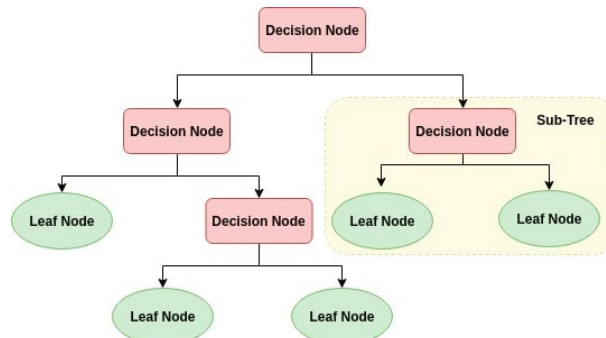


Figura 1: Ejemplo de árbol de decisiones. [4]

En este proyecto, se tiene la meta de que una vez se obtienen resultados satisfactorios con la red neuronal clásica, se seleccionen los parámetros que poseen más peso en la clasificación, por lo que se realizarán pruebas eliminando parámetros de entrada con el fin de averiguar cuáles representan una mayor importancia para el algoritmo. Una vez seleccionados los cinco parámetros más relevantes, se realizó un árbol de decisión diseñado para realizar la misma clasificación, pero este recibe únicamente los cinco parámetros de mayor relevancia, esto con el fin de comparar los resultados con los de la red clásica y así poder estudiar el comportamiento de estos árboles y la eficiencia que poseen a la hora de realizar clasificaciones.

III-3. Bosque aleatorio: El segundo paradigma de inteligencia artificial seleccionado fue el del bosque aleatorio, este consiste en un algoritmo basado en cálculos probabilísticos que utilizan diversos parámetros para generar múltiples árboles de decisiones, todos con estructura diferente y su propia sección de datos de entrenamiento, una vez se generó toda la lista de árboles de decisión, el resultado de la predicción de una salida que la mayoría de árboles estén de acuerdo, es el valor que se genera en la salida. Este paradigma de inteligencia artificial fue seleccionado por dos razones: primero, es una derivación de los árboles de decisión los cuales se estudiaron en este mismo proyecto, por lo que permite una mayor comprensión del funcionamiento de estos algoritmos (de hecho comparten distintos hiperparámetros) y además porque los bosques aleatorios pueden realizar operaciones de clasificación, considerando que el problema por resolver del presente proyecto es de este tipo, se aprovecha el hecho de que el algoritmo diseñado utilizando esta técnica sea capaz de realizar la clasificación de forma satisfactoria. [5]. La finalidad de este proyecto es, una vez se obtengan los resultados de la clasificación utilizando el bosque aleatorio, compararlos con los de la red neuronal clásica.

IV. CICLO DE DISEÑO EN INGENIERÍA

Como se explica en la sección de descripción del problema, el proyecto plantea el uso de un conjunto de datos de un censo, para realizar una predicción referente a los ingresos basándose en 14 atributos principales. Las categorías de salida corresponden a ingresos mayores a \$50000 e ingresos menores o iguales a dicha cifra.

Primeramente se debe realizar un clasificador empleando un paradigma de inteligencia artificial -considerado como el más adecuado. Como se justifica en la sección de análisis del problema, se escoge una red neuronal clásica para ello. Segundo, deben determinarse los cinco parámetros con mayor relevancia para la clasificación y con base a estos elaborar un árbol de decisión, el cuál debe ser comparado con los resultados obtenidos de la red. Tercero, debe utilizarse otro paradigma diferente al anterior para realizar la misma tarea de clasificación, los resultados deben compararse con el primer paradigma escogido. Como se explica en la sección anterior, se elige un bosque aleatorio -para clasificación.

Esto quiere decir que el proyecto presenta tres ejes principales sobre los cuales se van a trabajar, el primer problema por resolver es el diseño, implementación y definición de pruebas -para determinación de una mejor combinación de parámetros- de la red neuronal clásica. Seguidamente se define como segundo problema determinar los cinco parámetros más importantes, y cómo implementar el árbol de decisiones con base a estos, además este apartado incluye la comparación de resultados con

el primer paradigma. Por último se trata el problema de implementar un bosque aleatorio para clasificación, definir las pruebas necesarias para obtener la mejor combinación de parámetros y los mejores resultados compararlos con los frutos de la red neuronal.

A continuación se realiza una descripción de la solución implementada para cada punto y además la definición de las pruebas o pasos necesarios para llegar a esta. Cabe destacar que una vez obtenidos los resultados, estos son comentados y analizados según corresponda. Debido a que se trabajan tres problemas con cierto grado de independencia entre sí, se opta por programar la solución en tres códigos distintos: *Parte1*, *Parte2* y *Parte3* son los nombres de los archivos de programación (mediante *Python*) que contienen la respectiva implementación -junto con las pruebas definidas- para los puntos descritos previamente, respectivamente. Cabe destacar que la determinación de los parámetros más importantes se incluye en el código de parte 1 por orden y aprovechamiento del código de la red.

IV-1. Preparación de datos para red neuronal: El primer obstáculo para el problema por resolver, es el poder adaptar los datos del censo para poder ser manipulados por el algoritmo de la red neuronal. En el *dataset*, se encuentran 14 entradas, de las cuales 6 son valores continuos y 8 son categóricas (como se aprecia en I). No obstante, para un adecuado entrenamiento de la red neuronal, es necesario adaptar los datos para que estos sean representados de forma numérica, requisito fundamental debido a que el paradigma considera solamente entradas numéricas.

Es necesario realizar una codificación de los atributos categóricos, de forma tal que cada categoría pueda representarse numéricamente. Debido a que los parámetros de entrada de tipo categórico en su mayoría no presentan una relación ordinal entre sí -con una leve excepción en algunos aspectos de la categoría *Education*- se procede a emplear una técnica de codificación denominada *one – hot – encoding*, pues otros tipos como la codificación de enteros (donde a cada valor de un atributo se le asigna un valor decimal) resulta poco óptima para este tipo de categorías. [6]

Al aplicar *one – hot – encoding* a un atributo, se construye un vector de ceros y unos en donde una combinación específica representa una sub-categoría del atributo. Se asigna a cada elemento del vector una categoría, y cuando corresponda una categoría A, el respectivo elemento en el arreglo será 1. Esto para todos los elementos, a continuación, mediante el vector descrito en ecuación 1 y la tabla II, se realiza una breve demostración, basándose en el parámetro *Race*:

$$Race_{one_hot_encoded} = [a \ b \ c \ d \ e] \quad (1)$$

Tabla II: Representación mediante *one – hot – encoding* del atributo *Race*

Valor de atributo	Valores de los elementos del vector				
	a	b	c	d	e
Amer-Indian-Eskimo	1	0	0	0	0
Asian-Pac-Islander	0	1	0	0	0
Black	0	0	1	0	0
Other	0	0	0	1	0
White	0	0	0	0	1

Entonces, una vez realizada la codificación para los parámetros que lo requieran, cada entrada categórica se convierte en un arreglo cuya cantidad de columnas significa la cantidad de categorías presentes en esa entrada, importante destacar que se realiza una excepción con la categoría de sexo, pues se decide tomar la entrada solamente como un valor binario en donde 1 representa *Female* y *Male* se representa con 0. Tras realizar esto, la cantidad neta de entradas se incrementa, en la tabla III se evidencia la cantidad de entradas que representa cada atributo del conjunto de datos:

Tabla III: Cantidad de entradas según cada parámetro

Nombre del atributo	Número de columnas de entrada
Age	1
Workclass	9
Fnlwgt	1
Education	16
Education-num	1
Marital-Status	7
Occupation	14
Relationship	6
Race	5
Sex	1
Capital-gain	1
Capital-loss	1
Hours-per-week	1
Native-country	40
Total	104

Los datos de entrada categóricos pueden tomar el valor de 0 o 1, pero los valores continuos en el conjunto de datos deben tratarse para asegurar un mejor desempeño de la red [7], contemplando que existe una diferencia de magnitud -todos estos son

enteros positivos- entre los datos de valor continuos y los discretizados. Para ello se recurre a una normalización de los datos continuos, de esta forma se pueden situar entre valores que van de 0 a 1. Para esto se realiza una normalización $min - max$, cuya expresión matemática se describe en la ecuación 2.

$$x_{norm} = \frac{x - x_{min}}{x_{max} - x_{min}} \quad (2)$$

Una vez tratados los datos, se procedió a confeccionar la red neuronal utilizando la librería Keras, la cual se especializa en paradigmas de inteligencia artificial, especialmente en redes neuronales y derivados, con ayuda de los conocimientos adquiridos y con la referencia de otros experimentos similares, se confeccionó la red neuronal capaz de realizar la clasificación requerida.

IV-2. Implementación y definición de pruebas referentes a la red neuronal: Una vez habiendo codificado los datos de forma exitosa y confeccionado el algoritmo de la red neuronal clásica utilizando la librería *keras* de *Python*, se realizó un estudio de parámetros tomando en cuenta el conocimiento teórico-práctico previo sobre los estos, con el fin de obtener el mejor resultado posible para la clasificación de la red neuronal. Para cada combinación de parámetros se realizaron tres pruebas y se promediaron los resultados, a su vez se calculó la desviación estándar en los resultados de cada prueba para verificar la distribución de los resultados.

Los parámetros por estudiar y sus variantes son definidas a continuación:

- Optimizador: se varió entre *ADAM* ("Adaptive Moment Estimation") y *SGD* (*Gradient descent optimizer*), ambos optimizadores tienen una base teórica en el algoritmo de descenso de gradiente estudiado en el curso. Siendo el primero uno de los que tiene mayor popularidad en la ciencia de datos debido su potencial en las redes neuronales. [10]
- Batch size: se refiere al lote de datos de entrenamiento que se utilizan por cada iteración, se hicieron pruebas con un 100 % y con un 50 % de los datos de entrenamiento por cada iteración.
- Learning rate e iteraciones: este parámetro controla qué tan rápido aprende la red neuronal, entre más bajo, se requieren de más iteraciones para poder obtener resultados satisfactorios, por lo que se seleccionó un número fijo de iteraciones (250) que permite utilizar los valores de learning rates de 0,001 y 0,0001.
- Topología de la red: se plantea el uso de una topología con una única capa oculta, y otra topología con dos capas ocultas.

Cantidad de neuronas en la primera capa: se plantearon dos valores de la cantidad de neuronas: 104 ya que es el número de entradas en total que se tienen luego de realizar la codificación y normalización de datos, esto con el fin esperado de que cada neurona tenga la función de detectar un valor de entrada y realizar operaciones sencillas para lograr clasificar los datos, y también el valor 150, el cual brinda neuronas adicionales para poder realizar la clasificación.

Cantidad de neuronas en la segunda capa: De forma similar a la primera, se establecieron dos valores distintos para la cantidad de neuronas en esta capa, 50 y 150, con el fin de averiguar cuál combinación de neuronas es la más óptima. Una **(0 neuronas significa que la topología no presenta una segunda capa oculta)**.

- Función de activación: se emplea la función sigmoide para este aspecto. Se aprovecha el hecho de que el ámbito de esta función está en el rango de 0 a 1.
- Función de pérdida: para la función de pérdida, considerando que se trata de un problema de clasificación, se emplea *CrossEntropy*.

Los resultados se analizan para obtener la combinación de hiperparámetros que dé mejores resultados en términos de precisión. Una vez determinados estos hiperparámetros, se entrena una red con dicha características y se toman los resultados como la referencia para realizar la comparación con el árbol de decisión y el bosque aleatorio.

IV-3. Definición de parámetros más importantes: Una vez obtenidos los resultados de la red neuronal clásica (véase la sección de Resultados), se escoge la combinación de parámetros que presentase la mayor precisión promedio. Se entrena dicha red, empleando todos los datos de entrenamiento. Posteriormente se utiliza el modelo resultando para realizar pruebas utilizando un conjunto de datos reservado para dicha labor. Las pruebas consisten en determinar el peso de los atributos al medir el impacto en la precisión al *apagar* (igualar a cero) las entradas correspondientes al parámetro. De esta forma se realiza una comparación directa entre la precisión de la red bajo condiciones normales y la precisión con los datos modificados. Se obtiene un porcentaje de error relativo, el cual permite definir la relevancia de dicha entrada para la clasificación.

En la figura 2 se muestra un ejemplo ilustrativo referente a la idea. Se iguala a cero la entrada correspondiente a *Race*, debido a que esta está codificada mediante *one - hot - encoding*, distintas entradas asociadas al atributo se igualan a cero. De esta forma se espera observar desempeño de la red al aplicar la misma manipulación a cada atributo, mientras el resto permanece igual.

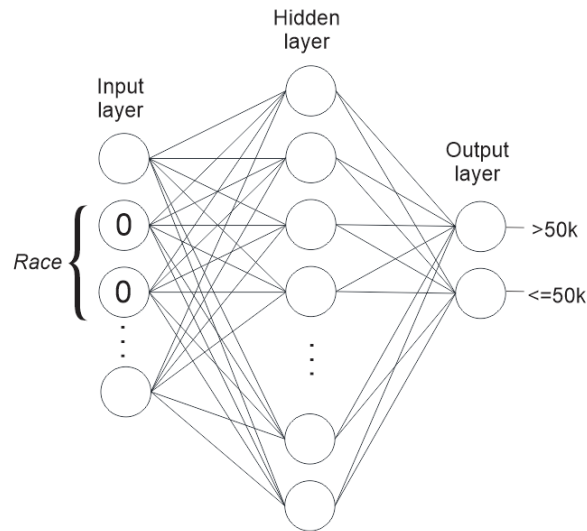


Figura 2: Ejemplo ilustrativo para determinación de parámetros de mayor relevancia (atributo *Race* igualado a cero, esto incluye todas las sub-categorías codificadas que lo conforman)

IV-4. Construcción e implementación del árbol de decisiones: El proyecto plantea la construcción de un árbol de decisiones considerando los cinco parámetros con mayor peso para la clasificación, por tanto, se parte de los cinco atributos más importantes determinados con ayuda de la red neuronal implementada. Se considera oportuno emplear recurso computacional para la generación de dicho árbol, puesto que la construcción basada meramente en observaciones y análisis de los datos, pese a presentar entradas reducidas, continúa siendo extensivamente laboriosa, además de que pueden presentarse sesgos y errores en el análisis que pueden afectar seriamente la precisión, asimismo los datos continuos deberían ser segregados en rangos debido a cómo funciona un árbol de decisiones.

Entonces, la aproximación escogida para la construcción del árbol de decisiones es mediante el paradigma de Inteligencia Artificial de *Decision tree*, para esto se utiliza Python y la librería de SciKit. Primeramente, a diferencia del caso de la red neuronal, para el árbol de decisiones los datos no requieren ser normalizados, ya que este algoritmo busca el mejor punto de división en cada característica. El punto de división está determinado por el porcentaje de etiquetas clasificadas correctamente mediante una característica. Por lo tanto, la estandarización no tiene ningún impacto significativo en este tipo de modelos de IA.[7]

Se toman los datos de entrenamiento con las cinco entradas más relevantes para la clasificación, según el estudio realizado con ayuda de la red neuronal, para generar árboles de decisión que permitan realizar la tarea de clasificación. Cabe destacar que existen distintos parámetros en este paradigma, por lo tanto se realiza un estudio de parámetros para encontrar la combinación más efectiva, cabe destacar que se realizan tres pruebas por cada combinación de parámetros, para determinar estadísticamente la solución más apropiada (empleando el promedio de precisión por combinación de parámetros). A continuación se mencionan los distintos parámetros y cómo estos se varían:

- Criterio de calidad: Se refiere al criterio que define los caminos que se abren a partir existen 2 criterios: Impureza Gini y entropía, ambos son medidas de la frecuencia con la que un elemento elegido al azar del conjunto se etiquetaría incorrectamente si se etiquetara al azar de acuerdo con la distribución de etiquetas en el subconjunto. la diferencia entre ambas está en la forma matemática de calcular esa frecuencia, ya que la impureza Gini utiliza valores cuadráticos, mientras que la entropía utiliza valores logarítmicos.
- Profundidad máxima: Este parámetro define las dimensiones del árbol generado, se puede tanto establecer un valor para definir una cantidad de decisiones que se deben tomar antes de obtener un resultado, o bien si no se escoge un valor, el algoritmo va a realizar un árbol del tamaño que sea necesario para que todos los nodos queden de forma terminal o en otras palabras, que no hayan más decisiones que tomar. Se probaron los valores de 5, 10 y 25 para la profundidad máxima, además se realizaron pruebas sin elegir un valor en específico.
- Número mínimo de muestras para dividir un nodo interno: Dicta el mínimo número de muestras que deben cumplir con una condición, para que esta se divida, se probaron con los números 2, 4 y 6, a grandes rasgos, un valor pequeño de este parámetro, genera árboles más extensos.
- La cantidad de atributos a considerar al buscar la mejor división: La documentación de sklearn se refiere a este parámetro como la cantidad de atributos que se deben tomar en cuenta para realizar cualquier tipo de decisión, existen varias formas equivalentes, por lo que para este experimento, se utilizaron las tres formas distintas que existen para este parámetro: sqrt, la cuál dice que el valor máximo de atributos a tomar en cuenta para cada árbol es la raíz cuadrado de la cantidad total atributos, log2, la cual lo define como el logaritmo en base 10 de la cantidad total de atributos existentes y finalmente el

caso estándar, que dice que la cantidad máxima de atributos a considerar es igual al total de atributos existentes.[8]

Una vez realizadas las pruebas, se determina la mejor combinación de hiperparámetros en términos de precisión (la mejor precisión promedio). Se procede a evaluar nuevamente el modelo, pero bajo dichos parámetros, y los resultados obtenidos se contrastan con los resultados de la red neuronal. El objetivo es medir el impacto que tiene la omisión de atributos en la tarea de clasificación, recordando que el modelo de la red neuronal fue entrenado con la totalidad de atributos mientras que el árbol solamente tomó los cinco considerados como más importantes.

IV-5. Construcción e implementación del Random Forest Classifier: Finalmente, se planteó la confección de un clasificador de bosque aleatorio con el fin de comparar los resultados con los obtenidos por la red neuronal clásica, al realizar el mismo ejercicio que se realizó al inicio. De forma similar al árbol de decisiones, se utilizó la librería *scikit – learn* para construir un clasificador de bosque aleatorio capaz de realizar la tarea de clasificar los datos de entrada en personas que ganan más o menos de \$ 50 000. Nuevamente fue requerido un estudio de parámetros para definir la mejor combinación posible para esta tarea y así compararlo con la mejor combinación de parámetros para la red neuronal clásica. Cabe destacar que para al igual que para el árbol de decisiones, el bosque aleatorio no requiere normalización de los datos continuos, sin embargo, se preservan los atributos de forma codificada (igualmente, mediante *one – hot – encoding*).

Los hiperparámetros por estudiar fueron los mismos que en el caso del árbol de decisiones ya que el bosque aleatorio es una derivación del paradigma anteriormente mencionado, con la adición del parámetro definido como la cantidad de árboles, el cual como lo dice su nombre, define cuántos árboles de decisión se van a crear para luego comparar sus salidas y así escoger la predicción más aceptada por todos los árboles confeccionados, se probaron tres valores de este parámetro, 10, 50 y 100 árboles.

Para determinar la mejor combinación de parámetros, se tabularon los resultados de tres pruebas distintas con todas las combinaciones planteadas, se obtuvieron los valores promedio, desviación estándar y el valor de la prueba individual máxima, se escogió la combinación de parámetros que cumpliera con el valor máximo promedio y si diera el caso de que otra combinación posea el valor de la prueba individual más alta, se define la mejor combinación como la que posea la menor desviación estándar entre las pruebas que destaquen.

V. RESULTADOS Y ANÁLISIS

V-1. Resultados de la red neuronal: Los resultados obtenidos por todas las combinaciones de los parámetros mencionadas anteriormente se encuentran en la sección de anexos, descritos en la tabla X, de los cuales destacan dos combinaciones de parámetros cuyos resultados sobresalieron, los entrenamientos número 3 y número 24, ambos presentes en la tabla IV, poseen los dos valores más altos de la precisión máxima, no obstante, la combinación número 24, posee una desviación estándar menor y mayor precisión promedio, por lo que esta tiene una mayor consistencia en sus resultados y se concluyó que dicho experimento presenta los mejores resultados.

Tabla IV: Mejor combinación de parámetros de la red neuronal según precisión y estadística

Combinación	Optimizador	Batch size	Learning Rate	Cantidad de neuronas en capa oculta 1	Cantidad de neuronas en capa oculta 2.	Función de activación de capas ocultas	Precisión promedio	Desviación estándar de la precisión	Precisión máxima
3	ADAM	Full	0,001	104	50	Relu	0,84660796	0,001838028	0,84868962
25	ADAM	50 %	0,001	104	0	Relu	0,84763171	0,000387581	0,84807533

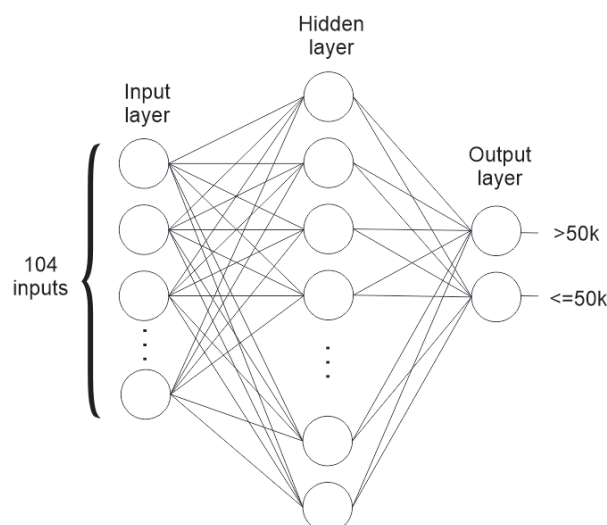


Figura 3: Topología de la red con mejores resultados

En la figura 3 se presenta la topología de la red con el mejor desempeño y a continuación se enlista la combinación de hiper parámetros que la caracterizan:

- Optimizador: ADAM
- Batch size: 50 %
- Learning rate: 0.001
- Cantidad de neuronas en capa oculta 1: 104
- La topología no presenta segunda capa oculta.

La combinación de parámetros según la teoría se puede justificar, ya que el optimizador ADAM utiliza una variación del algoritmo del descenso de gradiente el cual se ha comprobado que permite optimizar datos de una forma muy eficiente, como se puede apreciar en otros experimentos como el presente en la investigación de Potdar, K. y Pai, C. [9], el learning rate utiliza un valor que permite que la red aprenda a la velocidad adecuada para poder comprender las características que definen si un individuo pertenece a un grupo o a otro y además la cantidad de neuronas en la primera capa resultó ser la ideal en 104 neuronas ya que si se utilizaba un valor mayor, se sobre complicaba la solución del problema hecha por la red, y al tener una menor cantidad, la red no es capaz de resolver la clasificación de forma eficaz, no se coloca segunda capa oculta debido al mismo caso que cuando la primera capa tiene un valor mayor al seleccionado.

En la figura 4 se presentan tres curvas de precisión de entrenamiento y validación de ensayos distintos de la mejor combinación de parámetros, con el fin de verificar con mayor seguridad los resultados, en los tres casos se puede notar una precisión mayor a 0,8 y cercana a 0,85, como fue denotado en la tabla de los resultados (tabla X de la sección de anexos), además de que se puede apreciar el comportamiento deseado de este tipo de curvas, un ascenso constante hasta llegar a la asíntota de un valor. Además en la figura 4, también se presentan las matrices de confusión respectivas a las curvas descritas anteriormente, donde se puede apreciar que la red es capaz de clasificar las personas con salarios menores o iguales a \$ 50 000 con hasta un 93 % de precisión, pero con el otro caso, (salarios mayores a \$ 50 000) tiene problemas al reconocerlos debido a la dispersión de los datos, por lo que posee una precisión cercana al 60 %. Estos datos tienen sentido si se toma en cuenta el hecho de que existe un sesgo de datos, ya que existe una gran cantidad de casos (75 % del *dataset*) para los casos de personas con salarios menores o iguales a \$ 50 000, por lo que la red siempre va a tener más oportunidades de aprender patrones para reconocer si un individuo cumple con las características que lo hacen tener un salario menor o igual a \$ 50 000 que con el caso contrario.

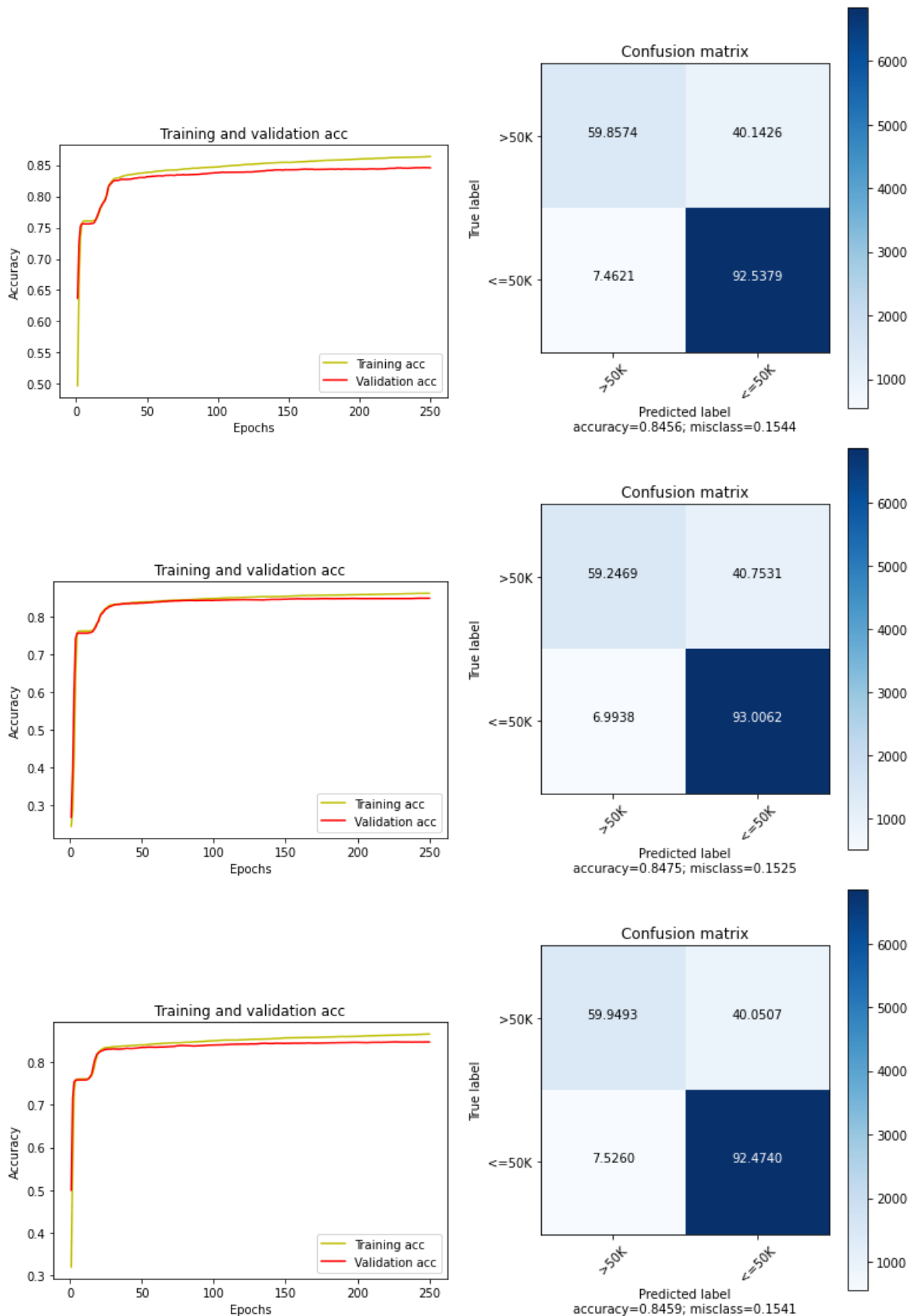


Figura 4: Curvas de precisión para el entrenamiento y validación de la red neuronal con respectivas matrices de confusión para tres pruebas utilizando la mejor combinación de parámetros obtenida.

V-2. *Resultados de la determinación de los cinco atributos más importantes:* En la tabla V se pueden apreciar los resultados que se generan al eliminar uno de los parámetros y a su vez la comparación con la precisión del mejor resultado obtenido. Destacaron cinco entradas que representaron errores de más de 2 %, estos fueron las entradas del *dataset* que en el censo representan la edad de la persona, el nivel de educación de la misma, su estado marital, su etnia y las horas de trabajo por semana. Estas entradas se utilizaron para realizar el entrenamiento de un árbol de selección el cuál tendrá la meta de realizar la misma clasificación, con la diferencia de que únicamente tendrá cinco entradas en lugar de catorce.

Tabla V: Pruebas de la red neuronal eliminando un atributo de entrada para encontrar las entradas más relevantes en la clasificación.

Atributo eliminado	Precisión	Porcentaje de error
age	0,827702703	2,331481833
<i>workclass</i>	0,84531122	0,253685281
<i>fnlwgt</i>	0,848587224	0,13288152
education	0,815724816	3,744866697
<i>education – num</i>	0,838349713	1,075139732
marital-status	0,820741196	3,152936284
<i>occupation</i>	0,838554464	1,050979307
<i>relationship</i>	0,842137592	0,628171869
race	0,813472563	4,010631372
<i>sex</i>	0,84049959	0,821455269
<i>capital – gain</i>	0,835892711	1,365064832
<i>capital – loss</i>	0,846130221	0,157043581
hours-per-week	0,806408681	4,844166036
<i>native – country</i>	0,843673219	0,446968681

V-3. *Resultados del árbol de clasificación:* Una vez realizada la segregación de las entradas que representan un menor impacto para la clasificación, se confeccionó el árbol de clasificación utilizando los cinco atributos de mayor peso (edad, educación, estado marital, etnia y horas por semana). El siguiente paso fue averiguar la mejor combinación de parámetros posible para realizar la clasificación deseada, por lo que se realizó el estudio de parámetros, cuyos resultados se pueden observar en la tabla XI, de los cuales destacó la combinación número 10, (precisión promedio de 0.820263) la cual cumple tanto con el valor promedio más alto como con el valor de precisión más elevado, por lo que fue seleccionada como la mejor combinación de parámetros para el árbol de decisiones. Los hiper parámetros de la mejor combinación se detallan en la tabla VI.

Tabla VI: Combinación más adecuada de parámetros para el árbol de decisión

Combinación	Criterio	Atributos máx.	División mín. de muestras	Máx. profundidad	Promedio	Desviación estándar	Valor máximo
10	Gini	None	6	10	0,820263	0,000213	0,820434

Se puede apreciar que dentro de los hiper parámetros empleados, el criterio Gini fue ganador, sin embargo, teóricamente el efecto de usar Entropía o Gini no son tan significativos, pero el Gini aporta eficiencia computacional al evitar el uso de logaritmos [11]. Al tener la cantidad máxima de atributos se contribuye en la cantidad de datos evaluados. La división mínima de muestras para la separación (igual a 6) era la mayor dentro de las opciones dadas, asimismo la profundidad máxima del árbol, limitada a 10 nodos, dan indicios de que el árbol se vio beneficiado de una exploración, pero no extensivamente profunda. Es decir no vio mejores resultados al emplear mayores profundidades o criterios de separación en donde se tuviera menores cantidades de muestras.

En la figura 5 se encuentran tres matrices de confusión obtenidas con la mejor combinación de parámetros obtenida para el árbol de clasificación.

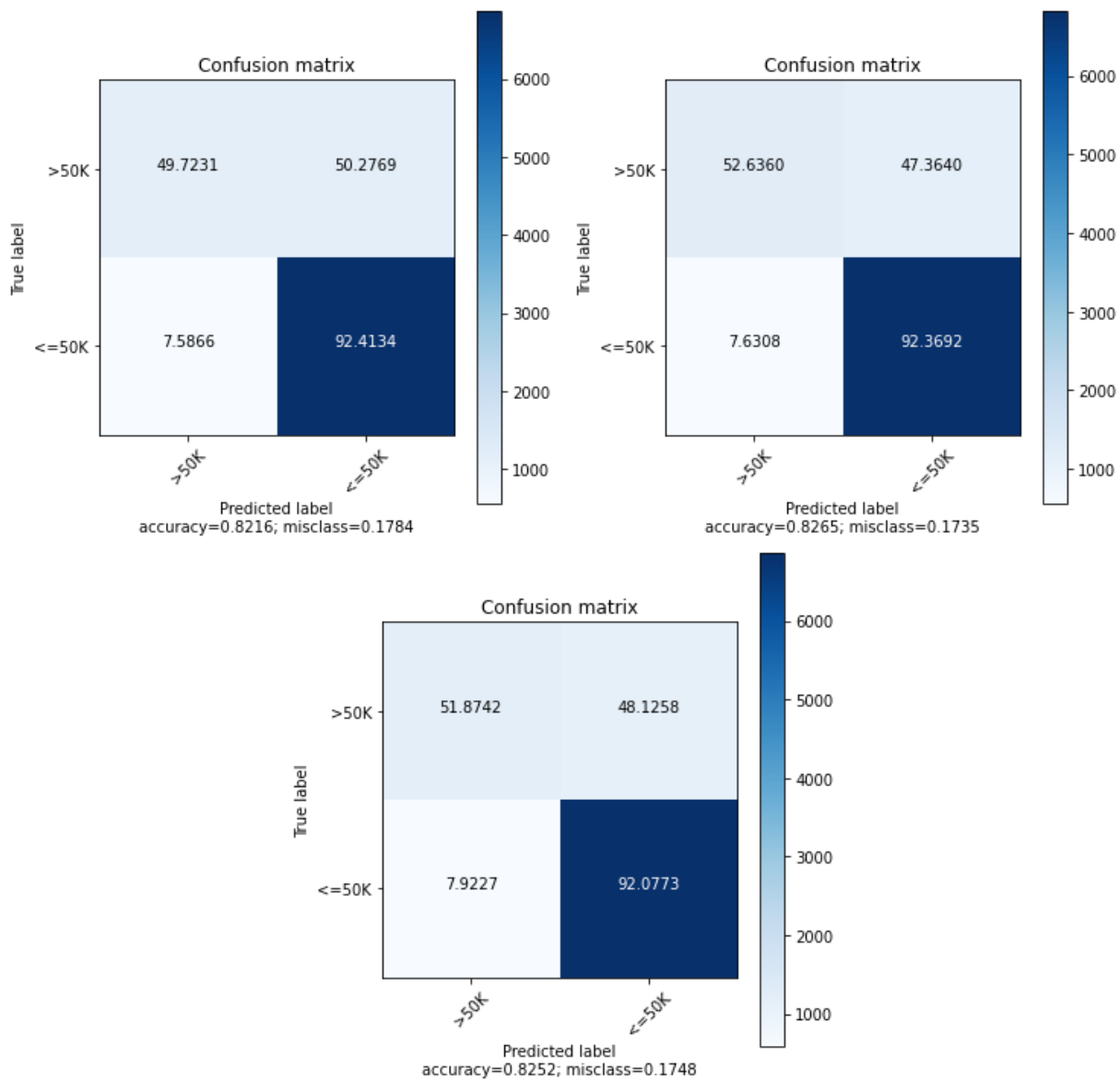


Figura 5: Matrices de confusión para los árboles de decisión confeccionados con la mejor combinación de parámetros obtenida.

Se puede apreciar para todas las matrices de la figura 5 que la categoría " $< \$ = 50K$ " presenta gran porcentaje de aciertos (alrededor de 92 %) en contraposición con la otra categoría, que presenta verdaderos positivos de alrededor de 50 % (cabe destacar que se presenta una desviación estándar de 1.51 para los datos clasificados adecuadamente en esta categoría). Esto coincide con los resultados obtenidos de la red neuronal, en donde se presenta un comportamiento similar y la categoría de ingresos menores o iguales a 50000 USD presenta mayor cantidad de aciertos que la predicción para ingresos mayores a 50000 USD. Esto puede justificarse de la misma forma que para la red: se presenta un desbalance entre la cantidad de datos clasificados con una categoría y otra (relación 25:75), presentándose un mejor resultado en la categoría más frecuente en el conjunto de datos, es decir, " $<=50$ ". Ocurre para ambos casos, que la tendencia de tener más datos de una categoría sesga los resultados.

Además, en la figura 6 se presentan tres versiones simplificadas de los árboles de decisión confeccionados con la mejor combinación de parámetros obtenida, las versiones completas tienen una forma muy extensa y no es posible incluirlas en este informe debido a su tamaño.

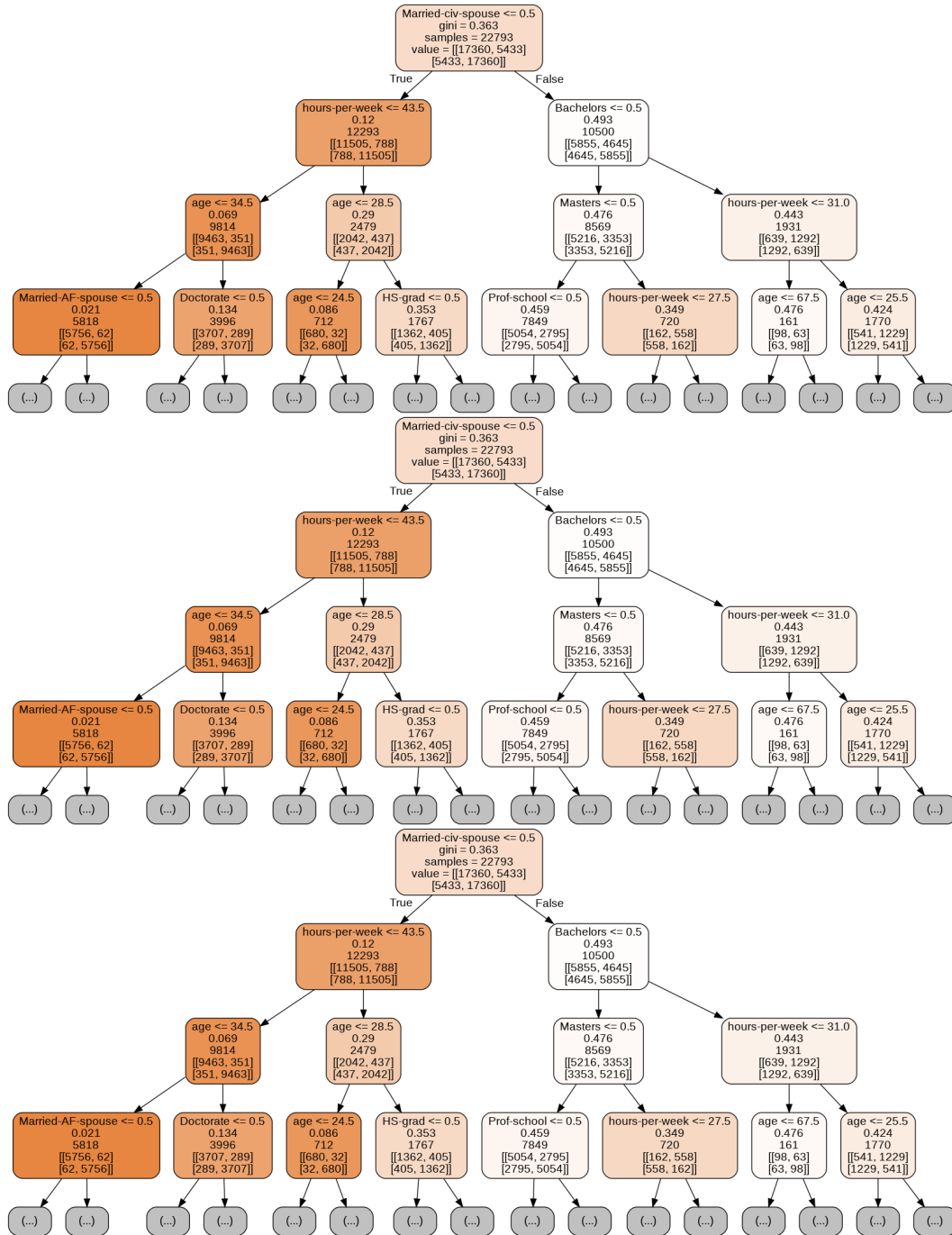


Figura 6: Árboles de decisión confeccionados con las mejores combinaciones de parámetros para estos árboles.

Puede apreciarse que para todos los árboles de la figura 6 los primeros nodos son iguales, es decir, la base de la toma de decisiones es igual para los casos presentados al utilizar la misma combinación de parámetros. Esto a su vez reitera la importancia de los valores que toman los atributos para la clasificación. Por ejemplo, la primera decisión corresponde a verificar si la persona contestó en el censo si su situación marital es estar casado con un cónyuge civil (no militar). A partir de dicha respuesta se ramifica en dos aspectos relevantes o de impacto en la clasificación, como lo es la cantidad de horas trabajadas por semana y la educación. Se puede apreciar cómo se van ramificando las distintas decisiones, pero el aspecto más destacable es la homogeneidad de los nodos entre distintos árboles para los nodos expuestos (profundidad igual a 3).

V-4. Comparación entre red neuronal y árbol de decisión: Se procede a colocar en la tabla VII los resultados de precisión promedio de las tres pruebas empleando la mejor combinación de parámetros para la red neuronal y el árbol de decisiones.

Tabla VII: Comparación entre resultados de red neuronal y árboles de decisiones implementados

Prueba	Precisión red clásica	Precisión árbol
1	0,8456	0,8216
2	0,8475	0,8265
3	0,8459	0,8252
Promedio	0,8463	0,8244
Desv. Est.	0,0010	0,0025
Porcentaje de error	2.59 %	

Como aclaratoria se considera que el valor real o teórico corresponde al resultado de la red neuronal clásica. Se puede apreciar que en términos de precisión el porcentaje de error es de 2.59 %. Esto indica que al emplear un árbol de decisiones, considerando solamente los cinco parámetros que fueron determinados como más impactantes, se obtiene un resultado de precisión promedio relativamente cercano. Sin embargo, la precisión sí es inferior al omitir estos atributos, lo cual quiere indicar que son necesarios atributos de entrada adicionales para realizar una predicción más precisa, es decir, para asegurar un mejor desempeño de clasificación, se requieren de atributos catalogados de menor relevancia. Pese a ello es importante destacar que, aunque se presente una precisión inferior, los resultados tampoco difieren de forma excesiva y podrían hasta considerarse como válidos -en especial si la predicción no es utilizada para aplicaciones críticas que puedan afectar la vida de los encuestados.

Ambos casos, como se expuso previamente, presentan el mismo sesgo hacia la clasificación de ingresos inferiores o iguales a 50000 USD, puesto que dicha condición se repite para ambos casos, se reafirma el argumento de que el desbalance en el conjunto de datos propicia la obtención de mejores resultados en una categoría que en otra.

V-5. *Resultados del Bosque aleatorio:* El paradigma del bosque aleatorio, una vez fue confeccionado en *Python* de forma exitosa, se realizó un estudio de parámetros similar al hecho en los otros dos experimentos, en este caso se utilizó el mismo *set* de datos que para la red neuronal clásica, los resultados se pueden apreciar en la tabla XII de la sección de anexos, se puede notar que existe una cantidad de combinaciones notoriamente mayor a los demás casos debido a que existen más parámetros que probar que en el caso del árbol de decisión. En estas destaca la combinación número 69 (precisión promedio 0.862032 y precisión máxima 0.862510238) que posee tanto la mayor precisión promedio como el mayor valor de precisión máxima, por lo que se escogió como la mejor combinación de parámetros para este experimento. Este resultado se puede apreciar en la tabla VIII.

Tabla VIII: Combinación más adecuada de parámetros para el bosque aleatorio

Combinación	Criterio	Atributos máx.	División mín. de muestras	Máx. profundidad	Cantidad de árboles	Precisión máxima promedio	Desviación estándar	Precisión máxima
69	gini	sqrt	6	25	100	0,862032487	0,000827488	0,862510238

Los parámetros que mejor dieron desempeño para la clasificación pueden ser justificados ya que el criterio Gini genera valores más sencillos de interpretar para el algoritmo que los generados por entropía, ya que esta última utiliza logaritmos para calcular las frecuencias por lo que hay mayor probabilidad de generar datos flotantes en vez de enteros, la cantidad máxima de atributos seleccionada fue sqrt ya que esta se encarga de repartir la misma cantidad de variables a todos los árboles (la raíz cuadrada del total de variables), la división máxima de muestras favorecida fue el valor de seis ya que esta permite la realización de árboles menos extensos, este caso favoreció también para el experimento utilizando un único árbol de decisión por la misma razón, finalmente cantidad de árboles seleccionada es de 100, y tiene sentido ya que la forma de funcionar de un árbol aleatorio requiere de varios árboles que posean resultados similares para poder obtener un consenso sobre cuál debería ser la salida ante ciertas entradas [5].

Una vez escogida la mejor combinación de parámetros, se realizó la comprobación de los resultados generando tres matrices de confusión para distintas corridas del mismo algoritmo con los parámetros definidos como los mejores para este ejercicio, los resultados se encuentran en la figura 7

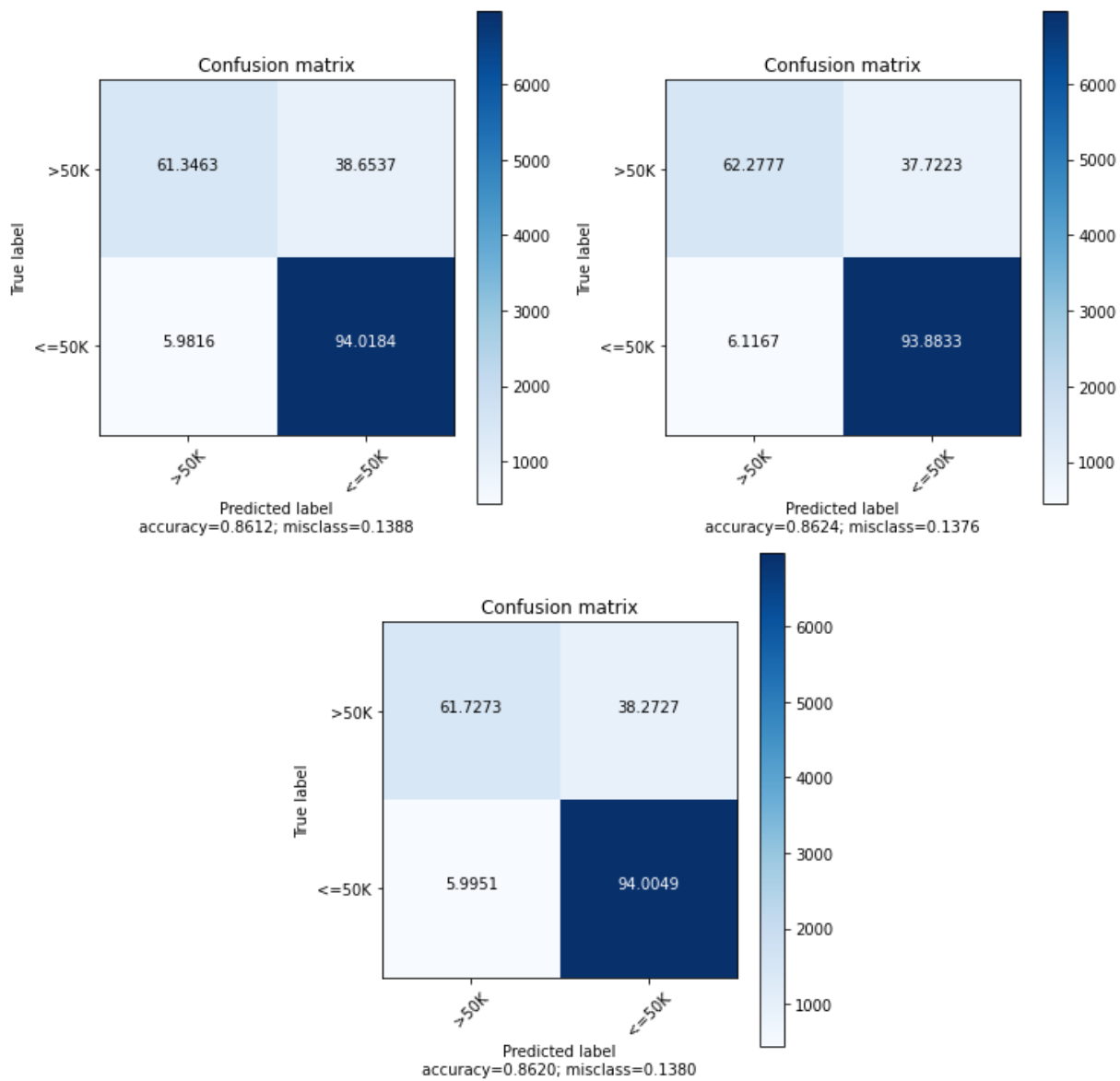


Figura 7: Matrices de confusión para los bosques aleatorios confeccionados con la mejor combinación de parámetros obtenida.

Se puede notar una precisión para las personas que tienen un salario menor o igual a \$ 50 000 de aproximadamente 94 % en todas las pruebas realizadas con la combinación de parámetros denominada como la mejor, además de que se ve una precisión entre 61 y 62 % para el otro caso, nuevamente, sale a reducir la dispersión de los datos como un factor de error para los individuos con salarios mayores a \$ 50 000.

V-6. *Comparación entre red neuronal y bosque aleatorio:* Con los resultados obtenidos en las figuras 4 y 7 se puede realizar una comparación del desempeño de ambos paradigmas a la hora de clasificar los datos del censo utilizados, se puede observar en la tabla IX el cómo el bosque aleatorio es capaz de constantemente obtener mejores resultados que la red neuronal clásica, además de poseer mayor precisión ya que la desviación estándar de este parámetro es considerablemente más baja, por lo que se puede concluir con bases numéricas experimentales, que el uso de un clasificador de bosque aleatorio fue el paradigma de inteligencia artificial mejor adaptado para la solución del problema planteado.

Tabla IX: Comparación de resultados entre la red neuronal clásica con el bosque aleatorio

Paradigma utilizado	Resultado 1	Resultado 2	Resultado 3	Promedio	Desviación estándar
Red neuronal clásica	0,8456	0,8475	0,8459	0,8463	0,00102144
Bosque aleatorio	0,8612	0,8624	0,8620	0,8619	0,00061101

VI. CONCLUSIONES

- Se realizaron una red neuronal clásica, un árbol generacional y un bosque aleatorio en Python con la ayuda de las librerías Keras y scikit-learn para la resolución del problema práctico planteado.
- Se demostraron distintos conocimientos y habilidades referentes a las redes neuronales clásicas, árboles de decisión y bosques aleatorios.
- Se comprobó que para simplificar un problema de clasificación, es posible utilizar únicamente las entradas más relevantes o que tengan mayor peso y se pueden obtener resultados satisfactorios.
- Se determinó la importancia de los múltiples parámetros que tienen peso en el funcionamiento de las redes neuronales, los árboles de decisión y los bosques aleatorios.
- Se evidenció el efecto de presentarse datos desbalanceados en un conjunto de datos (una categoría se presenta en mayor proporción que otra).
- Se logró determinar que los bosques aleatorios se adaptan de mejor manera para realizar una clasificación como la planteada en la realización de este proyecto que las redes neuronales clásicas.

REFERENCIAS

- [1] Dua, D. and Graff, C. "Census Income Data Set" 2019. [Online]. Disponible en: <https://archive.ics.uci.edu/ml/datasets/Census+Income> [Accedido]
- [2] Ramos, D. "Real-Life and Business Applications of Neural Networks" Oct. 17, 2018. [Online]. Disponible en: <https://www.smartsheet.com/neural-network-applications>
- [3] Lorke, A. "Cybenko's Theorem and the capability of a neural network as function approximator" Sep. 24, 2019. Disponible en: https://www.mathematik.uni-wuerzburg.de/fileadmin/10040900/2019/Seminar__Artificial_Neural_Network_24_9_.pdf
- [4] sitiobigdata.com. 2021. Árbol de decisión en Machine Learning (Parte 1). [Online] Disponible en: <https://sitiobigdata.com/2019/12/14/arbol-de-decision-en-machine-learning-parte-1/> [Accedido Junio 21, 2021].
- [5] UJWALPAWAR913., 2020 "Lets Open the Black Box of Random Forests"[Blog] *analyticsvidhya*, Disponible en: <https://www.analyticsvidhya.com/blog/2020/12/lets-open-the-black-box-of-random-forests/>
- [6] Brownlee, J., 2017. "Why One-Hot Encode Data in Machine Learning?". [Blog] *Machine Learning Mastery*, Disponible en: <https://machinelearningmastery.com/why-one-hot-encode-data-in-machine-learning/> [Accedido Julio 19, 2021].
- [7] Zhang, Z., 2019. Understand Data Normalization in Machine Learning. [Blog] *Towards Data Science*, Disponible en: <https://towardsdatascience.com/understand-data-normalization-in-machine-learning-8ff3062101f0> [Accedido Junio 19, 2021].
- [8] Pedregosa, F. and Varoquaux, G. and Gramfort, A. and Michel, V. and Thirion, B. and Grisel, O. and Blondel, M. and Prettenhofer, P. and Weiss, R. and Dubourg, V. and Vanderplas, J. and Passos, A. and Cournapeau, D. and Brucher, M. and Perrot, M. and Duchesnay, E., 2011 "Scikit-learn: Machine Learning in Python". Disponible en: <https://scikit-learn.org/stable/index.html>
- [9] Potdar, K. Pai, C. "A Comparative Study of Categorical Variable Encoding Techniques for Neural Network Classifiers" vol. 175, no. 4, Oct. 2017, doi: 10.5120/ijca2017915495
- [10] Brownlee, J., 2017. Gentle Introduction to the Adam Optimization Algorithm for Deep Learning. [Blog] *Machine Learning Mastery*, Recuperado de: <https://machinelearningmastery.com/adam-optimization-algorithm-for-deep-learning/> [Accedido Junio 19, 2021].
- [11] L. Raileanu and K. Stoffel, "Theoretical Comparison between the Gini Index and Information Gain Criteria", vol. 41, no. 1, pp. 77–93, Mayo 2004, doi: 10.1023/B:AMAI.0000018580.96245.c6.

VII. ANEXOS

En esta sección se adjuntan las distintas tablas que no fueron directamente colocadas en el cuerpo del documento. Cabe destacar que los archivos de código (3 en total) se adjuntan en una carpeta con el mismo nombre de este archivo PDF.

Tabla X: Tabla con los resultados del estudio de parámetros para la red neuronal clásica.

Combinación	Optimizador	Parámetros					Resultados					
		Batch size	Learning Rate	Cantidad de neuronas en capa oculta 1	Cantidad de neuronas en capa oculta 2.	Función de activación	Primera prueba	Segunda prueba	Tercera prueba	Precisión promedio	Desviación estándar	Precisión máxima
1	ADAM	Full	0,001	104	0	Relu	0,84326375	0,84152335	0,84029484	0,84169398	0,00149179	0,84326375
2	ADAM	Full	0,001	104	0	Sigmoide	0,83159298	0,83456182	0,83404994	0,83340158	0,00158707	0,83456182
3	ADAM	Full	0,001	104	50	Relu	0,84868962	0,84592545	0,84520882	0,84660796	0,001838028	0,84868962
4	ADAM	Full	0,001	104	50	Sigmoide	0,83804262	0,83220720	0,83640456	0,83555146	0,003009792	0,83804262
5	ADAM	Full	0,001	104	150	Relu	0,84653974	0,84295660	0,84643734	0,84531122	0,002039805	0,84653974
6	ADAM	Full	0,001	104	150	Sigmoide	0,83896399	0,83538085	0,83476657	0,83637047	0,002266955	0,83896399
7	ADAM	Full	0,001	150	0	Relu	0,84387797	0,84244472	0,84162569	0,84264946	0,001140014	0,84387797
8	ADAM	Full	0,001	150	0	Sigmoide	0,83230960	0,83026206	0,83230960	0,83162709	0,001182147	0,83230960
9	ADAM	Full	0,001	150	50	Relu	0,84684682	0,84797299	0,84551597	0,84677859	0,001229931	0,84797299
10	ADAM	Full	0,001	150	50	Sigmoide	0,83241194	0,83323097	0,83292383	0,83285558	0,000413757	0,83323097
11	ADAM	Full	0,001	150	150	Relu	0,84684682	0,83988535	0,84418511	0,84363910	0,003512706	0,84684682
12	ADAM	Full	0,001	150	150	Sigmoide	0,83599508	0,83548319	0,83753073	0,83633633	0,001065573	0,83753073
13	ADAM	Full	0,0001	104	0	Relu	0,78194106	0,82176495	0,77006555	0,79125718	0,027079506	0,82176495
14	ADAM	Full	0,0001	104	0	Sigmoide	0,75819004	0,75767815	0,75767815	0,75784878	0,000295537	0,75819004
15	ADAM	Full	0,0001	104	50	Relu	0,82043409	0,82442671	0,82555282	0,82347121	0,002689811	0,82555282
16	ADAM	Full	0,0001	104	50	Sigmoide	0,75819004	0,75767815	0,75767815	0,75784878	0,000295537	0,75819004
17	ADAM	Full	0,0001	104	150	Relu	0,828862881	0,82616705	0,82862407	0,82787331	0,001481208	0,828862881
18	ADAM	Full	0,0001	104	150	Sigmoide	0,75819004	0,75767815	0,75767815	0,75784878	0,000295537	0,75819004
19	ADAM	Full	0,0001	150	0	Relu	0,82585996	0,81879610	0,82524568	0,82330058	0,003913068	0,82585996
20	ADAM	Full	0,0001	150	0	Sigmoide	0,75819004	0,75767815	0,75767815	0,75784878	0,000295537	0,75819004
21	ADAM	Full	0,0001	150	50	Relu	0,82432431	0,82626945	0,82719082	0,82592819	0,001463406	0,82719082
22	ADAM	Full	0,0001	150	50	Sigmoide	0,75819004	0,75767815	0,75767815	0,75784878	0,000295537	0,75819004
23	ADAM	Full	0,0001	150	150	Relu	0,82893121	0,82790744	0,83015972	0,82899946	0,00112769	0,83015972
24	ADAM	Full	0,0001	150	150	Sigmoide	0,81756759	0,77457005	0,75767815	0,78327193	0,030878442	0,81756759
25	ADAM	50 %	0,001	104	0	Relu	0,84735870	0,84807533	0,84746110	0,84763171	0,000387581	0,84807533
26	ADAM	50 %	0,001	104	0	Sigmoide	0,83609748	0,83527845	0,83497131	0,83544908	0,000582151	0,83609748
27	ADAM	50 %	0,001	104	50	Relu	0,84653974	0,84510648	0,84613019	0,84592547	0,000738232	0,84653974
28	ADAM	50 %	0,001	104	50	Sigmoide	0,83579034	0,83568794	0,83691645	0,83613157	0,000681646	0,83691645
29	ADAM	50 %	0,001	104	150	Relu	0,84592545	0,84131861	0,84357083	0,84360496	0,002303611	0,84592545
30	ADAM	50 %	0,001	104	150	Sigmoide	0,83732599	0,83804262	0,83609748	0,83715536	0,000983731	0,83804262
31	ADAM	50 %	0,001	150	0	Relu	0,84674448	0,84817773	0,84715396	0,84735872	0,000738241	0,84817773
32	ADAM	50 %	0,001	150	0	Sigmoide	0,83763307	0,83722359	0,83538085	0,83674584	0,001199712	0,83763307
33	ADAM	50 %	0,001	150	50	Relu	0,84694222	0,84377557	0,84152335	0,84408271	0,002725944	0,84694222
34	ADAM	50 %	0,001	150	50	Sigmoide	0,83609748	0,83681411	0,83845210	0,83712123	0,001206982	0,83845210
35	ADAM	50 %	0,001	150	150	Relu	0,83579034	0,84111387	0,84285420	0,83991947	0,003680284	0,84285420
36	ADAM	50 %	0,001	150	150	Sigmoide	0,83804262	0,84029484	0,83814496	0,83882747	0,001271807	0,84029484
37	ADAM	50 %	0,0001	104	0	Relu	0,82780510	0,82493860	0,82995498	0,82756623	0,00251671	0,82995498
38	ADAM	50 %	0,0001	104	0	Sigmoide	0,75839478	0,75788289	0,75859952	0,75829240	0,00036912	0,75859952
39	ADAM	50 %	0,0001	104	50	Relu	0,83497131	0,83312857	0,83374286	0,83394758	0,000938272	0,83497131
40	ADAM	50 %	0,0001	104	50	Sigmoide	0,75819004	0,75767815	0,75767815	0,75784878	0,000295537	0,75819004
41	ADAM	50 %	0,0001	104	150	Relu	0,83732599	0,83916873	0,83783782	0,83811084	0,000951225	0,83916873
42	ADAM	50 %	0,0001	104	150	Sigmoide	0,75819004	0,75767815	0,75767815	0,75784878	0,000295537	0,75819004
43	ADAM	50 %	0,0001	150	0	Relu	0,82985258	0,82923836	0,82975018	0,82961371	0,00032907	0,82985258
44	ADAM	50 %	0,0001	150	0	Sigmoide	0,76064700	0,77979118	0,76658475	0,76900764	0,009799371	0,77979118
45	ADAM	50 %	0,0001	150	50	Relu	0,83937347	0,83640456	0,83619982	0,83732595	0,001776156	0,83937347
46	ADAM	50 %	0,0001	150	50	Sigmoide	0,75819004	0,75767815	0,75767815	0,75784878	0,000295537	0,75819004
47	ADAM	50 %	0,0001	150	150	Relu	0,83886158	0,83865684	0,83783782	0,83845208	0,00054173	0,83886158
48	ADAM	50 %	0,0001	150	150	Sigmoide	0,75819004	0,75767815	0,75767815	0,75784878	0,000295537	0,75819004
49	SGD	Full	0,001	104	0	Relu	0,75941849	0,76996315	0,77948403	0,76962189	0,010037125	0,77948403
50	SGD	Full	0,001	104	0	Sigmoide	0,75819004	0,75767815	0,75767815	0,75784878	0,000295537	0,75819004
51	SGD	Full	0,001	104	50	Relu	0,76689190	0,75767815	0,75767815	0,76074940	0,005319558	0,76689190
52	SGD	Full	0,001	104	50	Sigmoide	0,75819004	0,75767815	0,75767815	0,75784878	0,000295537	0,75819004
53	SGD	Full	0,001	104	150	Relu	0,75870186	0,75767815	0,75767815	0,75801939	0,000591039	0,75870186
54	SGD	Full	0,001	104	150	Sigmoide	0,75819004	0,75767815	0,75767815	0,75784878	0,000295537	0,75819004
55	SGD	Full	0,001	150	0	Relu	0,75982803	0,76842749	0,75911140	0,76245564	0,00518417	0,76842749
56	SGD	Full	0,001	150	0	Sigmoide	0,75819004	0,75767815	0,75767815	0,75784878	0,000295537	0,75819004
57	SGD	Full	0,001	150	50	Relu	0,76167077	0,77436525	0,75788289	0,76463964	0,008632938	0,77436525
58	SGD	Full	0,001	150	50	Sigmoide	0,75819004	0,75767815	0,75767815	0,75784878	0,000295537	0,75819004
59	SGD	Full	0,001	150	150	Relu	0,75819004	0,77098691	0,75778049	0,76231915	0,007509299	0,77098691
60	SGD	Full	0,001	150	150	Sigmoide	0,75819004	0,75767815	0,75767815	0,75784878	0,000295537	0,75819004
61	SGD	Full	0,0001	104	0	Relu	0,75685912	0,75327599	0,75368547	0,75460686	0,001961231	0,75685912
62	SGD	Full	0,0001	104	0	Sigmoide	0,75819004	0,75767815	0,75767815	0,75784878	0,000295537	0,75819004
63	SGD	Full	0,0001	104	50	Relu	0,75778049	0,75747341	0,75685912	0,75737101	0,000469142	0,75778049
64	SGD	Full	0,0001	104	50	Sigmoide	0,75819004	0,75767815	0,75767815	0,75784878	0,000295537	0,75819004
65	SGD	Full	0,0001	104	150	Relu	0,75819004	0,75593776	0,75624490	0,75679090	0,001221384	0,75819004
66	SGD	Full	0,0001	104	150	Sigmoide	0,75819004	0,75767815	0,75767815	0,75784878	0,000295537	0,75819004
67	SGD	Full	0,0001	150	0	Relu	0,71642095	0,74375510	0,75501639	0,73839748	0,019847671	0,75501639
68	SGD	Full	0,0001	150	0	Sigmoide	0,75819004	0,75767815	0,75767815	0,75784878	0,000295537	0,75819004
69	SGD	Full	0,0001	150	50	Relu	0,75767815	0,75624490	0,75747341	0,75713215	0,000775175	0,75767815
70	SGD	Full	0,0001	150	50	Sigmoide	0,75819004	0,75767815	0,75767815	0,75784878	0,000295537	0,75819004
71	SGD	Full	0,0001	150	150	Relu	0,75819004	0,75685912	0,75767815	0,75757577	0,000671337	0,75819004
72	SGD	Full	0,0001	150	150	Sigmoide	0,75819004	0,75767815	0,75767815	0,75784878	0,000295537	0,75819004
73	SGD	50 %	0,001	104	0	Relu	0,80210894	0,80671579	0,81101555	0,80661343	0,004454184	0,81101555
74	SGD	50 %	0,001	104	0	Sigmoide	0,75819004	0,75767815	0,75767815	0,75784878	0,000295537	0,75819004
75	SGD	50 %	0,001	104	50	Relu	0,82299346	0,80927521	0,80937755	0,81388207	0,007890856	0,82299346
76	SGD	50 %	0,001	104	50	Sigmoide	0,75819004	0,75767815	0,75767815	0,75784878	0,000295537	0,75819004
77	SGD	50 %	0,001	104	150	Relu	0,81572479	0,81236466	0,81050366	0,81285830	0,00264793	0,81572479
78	SGD	50 %	0,001	104	150	Sigmoide	0,75819004	0,75767815	0,75767815	0,75784878	0,000295537	0,75819004
79	SGD	50 %	0,001	150	0	Relu	0,81644142	0,81347257	0,81347257	0,81446218	0,001714065	0,81644142
80	SGD	50 %	0,001	150	0	Sigmoide	0,75819004	0,75767815	0,75767815	0,75784878	0,000295537	0,7581

Tabla XI: Tabla con los resultados del estudio de parámetros para el árbol de decisiones.

Parámetros						Resultados				
Combinación	Criterio	Atributos máx	División mín de muestras	Máx profundidad	Prueba 1	Prueba 2	Prueba 3	Precisión máxima promedio	Desviación estándar	Precisión máxima
1	Gini	None	2	5	0,809582	0,809787	0,809582	0,809651	0,000118	0,809787
2	Gini	None	2	10	0,820178	0,819820	0,819871	0,819956	0,000194	0,820178
3	Gini	None	2	25	0,787930	0,787776	0,786906	0,787538	0,000552	0,787930
4	Gini	None	2	None	0,783886	0,785217	0,785063	0,784722	0,000728	0,785217
5	Gini	None	4	5	0,809582	0,809582	0,809582	0,809582	0,000000	0,809582
6	Gini	None	4	10	0,819410	0,819973	0,820127	0,819837	0,000377	0,820127
7	Gini	None	4	25	0,788698	0,788749	0,789824	0,789090	0,000636	0,789824
8	Gini	None	4	None	0,787520	0,787725	0,787623	0,787623	0,000102	0,787725
9	Gini	None	6	5	0,809582	0,809582	0,809582	0,809582	0,000000	0,809582
10	Gini	None	6	10	0,820332	0,820025	0,820434	0,820263	0,000213	0,820434
11	Gini	None	6	25	0,792127	0,792383	0,791974	0,792161	0,000207	0,792383
12	Gini	None	6	None	0,791155	0,790541	0,791052	0,790916	0,000329	0,791155
13	Gini	sqrt	2	5	0,776566	0,781634	0,783170	0,780457	0,003455	0,783170
14	Gini	sqrt	2	10	0,807740	0,787725	0,775747	0,790404	0,016163	0,807740
15	Gini	sqrt	2	25	0,789670	0,790745	0,789619	0,790012	0,000636	0,790745
16	Gini	sqrt	2	None	0,789926	0,788595	0,790950	0,789824	0,001181	0,790950
17	Gini	sqrt	4	5	0,789312	0,798526	0,766585	0,784808	0,016440	0,798526
18	Gini	sqrt	4	10	0,808712	0,806460	0,798731	0,804634	0,005235	0,808712
19	Gini	sqrt	4	25	0,794021	0,793765	0,792076	0,793288	0,001057	0,794021
20	Gini	sqrt	4	None	0,786650	0,787316	0,788186	0,787384	0,000770	0,788186
21	Gini	sqrt	6	5	0,773853	0,794738	0,780098	0,782897	0,010720	0,794738
22	Gini	sqrt	6	10	0,800317	0,803900	0,813575	0,805931	0,006858	0,813575
23	Gini	sqrt	6	25	0,794226	0,797400	0,794891	0,795506	0,001674	0,797400
24	Gini	sqrt	6	None	0,794789	0,793612	0,793970	0,794124	0,000603	0,794789
25	Gini	log2	2	5	0,763002	0,789210	0,762336	0,771516	0,015327	0,789210
26	Gini	log2	2	10	0,798270	0,794277	0,791308	0,794618	0,003493	0,798270
27	Gini	log2	2	25	0,792486	0,792793	0,791769	0,792349	0,000525	0,792793
28	Gini	log2	2	None	0,788391	0,789056	0,787418	0,788288	0,000824	0,789056
29	Gini	log2	4	5	0,783067	0,787469	0,797041	0,789193	0,007145	0,797041
30	Gini	log2	4	10	0,805538	0,811169	0,795199	0,803969	0,008100	0,811169
31	Gini	log2	4	25	0,792486	0,792588	0,792025	0,792366	0,000300	0,792588
32	Gini	log2	4	None	0,785268	0,787520	0,790541	0,787776	0,002645	0,790541
33	Gini	log2	6	5	0,782914	0,761005	0,783989	0,775969	0,012970	0,783989
34	Gini	log2	6	10	0,791871	0,796069	0,791052	0,792998	0,002691	0,796069
35	Gini	log2	6	25	0,798884	0,795147	0,796581	0,796871	0,001885	0,798884
36	Gini	log2	6	None	0,791104	0,790336	0,789722	0,790387	0,000692	0,791104
37	Entropy	None	2	5	0,808763	0,808559	0,808559	0,808627	0,000118	0,808763
38	Entropy	None	2	10	0,820127	0,819615	0,820434	0,820059	0,000414	0,820434
39	Entropy	None	2	25	0,790336	0,789773	0,790233	0,790114	0,000300	0,790336
40	Entropy	None	2	None	0,787367	0,786650	0,786599	0,786872	0,000429	0,787367
41	Entropy	None	4	5	0,808763	0,808763	0,808559	0,808695	0,000118	0,808763
42	Entropy	None	4	10	0,819973	0,819359	0,819206	0,819513	0,000406	0,819973
43	Entropy	None	4	25	0,791769	0,792434	0,792434	0,792213	0,000384	0,792434
44	Entropy	None	4	None	0,789414	0,788954	0,788903	0,789090	0,000282	0,789414
45	Entropy	None	6	5	0,808559	0,808763	0,808559	0,808627	0,000118	0,808763
46	Entropy	None	6	10	0,820229	0,820178	0,819717	0,820042	0,000282	0,820229
47	Entropy	None	6	25	0,792127	0,792895	0,792127	0,792383	0,000443	0,792895
48	Entropy	None	6	None	0,789670	0,790182	0,789466	0,789773	0,000369	0,790182
49	Entropy	sqrt	2	5	0,791667	0,762695	0,773546	0,775969	0,014637	0,791667
50	Entropy	sqrt	2	10	0,798935	0,783681	0,807074	0,796564	0,011875	0,807074
51	Entropy	sqrt	2	25	0,791923	0,794994	0,794072	0,793663	0,001576	0,794994
52	Entropy	sqrt	2	None	0,788800	0,791360	0,790541	0,790233	0,001307	0,791360
53	Entropy	sqrt	4	5	0,797809	0,764128	0,792537	0,784825	0,018117	0,797809
54	Entropy	sqrt	4	10	0,806050	0,812705	0,790387	0,803047	0,011458	0,812705
55	Entropy	sqrt	4	25	0,793765	0,796325	0,795710	0,795267	0,001336	0,796325
56	Entropy	sqrt	4	None	0,791206	0,789005	0,787162	0,789124	0,002025	0,791206
57	Entropy	sqrt	6	5	0,761568	0,763309	0,776208	0,767028	0,007997	0,776208
58	Entropy	sqrt	6	10	0,816595	0,811067	0,797656	0,808439	0,009739	0,816595
59	Entropy	sqrt	6	25	0,797451	0,792793	0,794072	0,794772	0,002407	0,797451
60	Entropy	sqrt	6	None	0,792639	0,789414	0,794584	0,792213	0,002611	0,794584
61	Entropy	log2	2	5	0,778460	0,768428	0,787572	0,778153	0,009576	0,787572
62	Entropy	log2	2	10	0,799908	0,798731	0,791923	0,796854	0,004311	0,799908
63	Entropy	log2	2	25	0,789722	0,793458	0,791155	0,791445	0,001885	0,793458
64	Entropy	log2	2	None	0,787674	0,786292	0,791923	0,788630	0,002934	0,791923
65	Entropy	log2	4	5	0,780917	0,791974	0,784296	0,785729	0,005666	0,791974
66	Entropy	log2	4	10	0,805948	0,803593	0,807791	0,805777	0,002104	0,807791
67	Entropy	log2	4	25	0,791308	0,794482	0,795762	0,793851	0,002293	0,795762
68	Entropy	log2	4	None	0,786599	0,790541	0,788903	0,788681	0,001980	0,790541
69	Entropy	log2	6	5	0,783118	0,759265	0,773444	0,771942	0,011997	0,783118
70	Entropy	log2	6	10	0,784193	0,807586	0,795199	0,795659	0,011703	0,807586
71	Entropy	log2	6	25	0,797553	0,793765	0,801085	0,797468	0,003661	0,801085
72	Entropy	log2	6	None	0,792946	0,791411	0,791769	0,792042	0,000803	0,792946

Tabla XII: Tabla con los resultados del estudio de parámetros para el bosque aleatorio.

Parámetros						Resultados					
Combinación	Criterio	Atributos máx	División mín de muestras	Máx profundidad	Cantidad de árboles	Prueba 1	Prueba 2	Prueba 3	Precisión máxima promedio	Desviación estándar	Precisión máxima
1	gini	All	2	5	10	0,850839476	0,853091728	0,852784603	0,852238602	0,001221371	0,853091728
2	gini	All	2	5	50	0,851453726	0,851146601	0,850634726	0,851078351	0,000413744	0,851453726
3	gini	All	2	5	100	0,852886978	0,850634726	0,851146601	0,851556102	0,001180647	0,852886978
4	gini	All	2	10	10	0,856521294	0,857698608	0,858364046	0,857527983	0,00093315	0,858364046
5	gini	All	2	10	50	0,85744267	0,859029484	0,859899672	0,858790609	0,001245797	0,859899672
6	gini	All	2	10	100	0,858568796	0,859029484	0,85995086	0,859183047	0,000703712	0,85995086
7	gini	All	2	25	10	0,844543407	0,846590909	0,847563473	0,846232596	0,001541587	0,847563473
8	gini	All	2	25	50	0,853808354	0,855036855	0,852477477	0,853774229	0,00128003	0,855036855
9	gini	All	2	25	100	0,855804668	0,856316544	0,854985667	0,855702293	0,000671318	0,856316544
10	gini	All	2	Full	10	0,840141278	0,844389844	0,841318591	0,841949904	0,002193512	0,844389844
11	gini	All	2	Full	50	0,853142916	0,852733415	0,851248976	0,852375102	0,000996515	0,853142916
12	gini	All	2	Full	100	0,852272727	0,851044226	0,850993038	0,851436664	0,000724504	0,852272727
13	gini	All	4	5	10	0,850634726	0,850737101	0,851146601	0,850839476	0,000270859	0,851146601
14	gini	All	4	5	50	0,853501229	0,850941851	0,851146601	0,851863227	0,00142224	0,853501229
15	gini	All	4	5	100	0,854217854	0,852067977	0,853398853	0,853228228	0,001085047	0,854217854
16	gini	All	4	10	10	0,856981982	0,858415233	0,858108108	0,857835108	0,000754619	0,858415233
17	gini	All	4	10	50	0,858619984	0,858927109	0,858312858	0,858619984	0,000307125	0,858927109
18	gini	All	4	10	100	0,858517609	0,858210483	0,858517609	0,858415233	0,000177319	0,858517609
19	gini	All	4	25	10	0,852886978	0,850685913	0,849866912	0,851146601	0,00156185	0,852886978
20	gini	All	4	25	50	0,857698608	0,856777232	0,854780917	0,856418919	0,001491483	0,857698608
21	gini	All	4	25	100	0,857596233	0,856930794	0,857289107	0,857272045	0,000333047	0,857596233
22	gini	All	4	Full	10	0,849406224	0,846744472	0,851095414	0,849082037	0,002193512	0,851095414
23	gini	All	4	Full	50	0,855292793	0,854883292	0,853910729	0,854695605	0,000709891	0,855292793
24	gini	All	4	Full	100	0,857493857	0,854320229	0,854217854	0,85534398	0,001862552	0,857493857
25	gini	All	6	5	10	0,851351351	0,850429975	0,853194103	0,851658477	0,001407425	0,853194103
26	gini	All	6	5	50	0,850634726	0,851146601	0,853603604	0,851794977	0,001587089	0,853603604
27	gini	All	6	5	100	0,851556102	0,851248976	0,853603604	0,852136227	0,00128003	0,853603604
28	gini	All	6	10	10	0,858108108	0,857698608	0,857954545	0,85792042	0,000206872	0,858108108
29	gini	All	6	10	50	0,860053235	0,858619984	0,858415233	0,859029484	0,000892485	0,860053235
30	gini	All	6	10	100	0,859234234	0,858824734	0,859029484	0,859029484	0,00020475	0,859234234
31	gini	All	6	25	10	0,849696287	0,851402539	0,847768223	0,84971335	0,001830626	0,851402539
32	gini	All	6	25	50	0,857698608	0,859844845	0,857545045	0,858364046	0,001287853	0,859844845
33	gini	All	6	25	100	0,856060606	0,859029484	0,856981982	0,857357357	0,001519618	0,859029484
34	gini	All	6	Full	10	0,84700041	0,847307535	0,849355037	0,84788766	0,00128003	0,849355037
35	gini	All	6	Full	50	0,858517609	0,856674857	0,855241605	0,856811357	0,001642262	0,858517609
36	gini	All	6	Full	100	0,856674857	0,855753481	0,856470106	0,856299481	0,000483806	0,856674857
37	gini	sqrt	2	5	10	0,828009828	0,830671581	0,826167076	0,828282828	0,002264627	0,830671581
38	gini	sqrt	2	5	50	0,843366093	0,838349713	0,839578215	0,84043134	0,002614744	0,843366093
39	gini	sqrt	2	5	100	0,839475839	0,838349713	0,840192465	0,839339339	0,000928928	0,840192465
40	gini	sqrt	2	10	10	0,854422604	0,851044226	0,851760852	0,852409227	0,001780071	0,854422604
41	gini	sqrt	2	10	50	0,855548731	0,855958231	0,85534398	0,855616981	0,000312761	0,855958231
42	gini	sqrt	2	10	100	0,856674857	0,855855856	0,856060606	0,856197106	0,000426222	0,856674857
43	gini	sqrt	2	25	10	0,848894349	0,853296478	0,852938165	0,851709664	0,002444708	0,853296478
44	gini	sqrt	2	25	50	0,858927109	0,859844845	0,858875921	0,859217172	0,000547332	0,859844845
45	gini	sqrt	2	25	100	0,859592547	0,859080672	0,858005733	0,858892984	0,000809886	0,859592547
46	gini	sqrt	2	Full	10	0,843263718	0,842547093	0,842649468	0,842820093	0,000387586	0,843263718
47	gini	sqrt	2	Full	50	0,852477477	0,850634726	0,851351351	0,851487851	0,000928928	0,852477477
48	gini	sqrt	2	Full	100	0,851351351	0,853705979	0,851453726	0,852170352	0,001330876	0,853705979
49	gini	sqrt	4	5	10	0,838759214	0,82964783	0,832309582	0,833572209	0,004685083	0,838759214
50	gini	sqrt	4	5	50	0,837428337	0,840601966	0,841420966	0,83981709	0,002108861	0,841420966
51	gini	sqrt	4	5	100	0,839066339	0,845208845	0,837530713	0,840601966	0,004062886	0,845208845
52	gini	sqrt	4	10	10	0,853910729	0,849815725	0,85452498	0,852750478	0,00256006	0,85452498
53	gini	sqrt	4	10	50	0,854320229	0,856470106	0,855958231	0,855582856	0,00112302	0,856470106
54	gini	sqrt	4	10	100	0,856572482	0,856060606	0,855855856	0,856162981	0,000369119	0,856572482
55	gini	sqrt	4	25	10	0,852989353	0,852272727	0,857698608	0,854320229	0,002947621	0,857698608
56	gini	sqrt	4	25	50	0,857186732	0,861076986	0,859029484	0,859097734	0,001946025	0,861076986
57	gini	sqrt	4	25	100	0,861588862	0,860462735	0,85954136	0,860530986	0,001025456	0,861588862
58	gini	sqrt	4	Full	10	0,846898034	0,844952907	0,844645782	0,845498908	0,001221371	0,846898034
59	gini	sqrt	4	Full	50	0,855241605	0,854371417	0,85493448	0,854849167	0,000441323	0,855241605
60	gini	sqrt	4	Full	100	0,854832105	0,855446355	0,856470106	0,855582856	0,000827488	0,856470106
61	gini	sqrt	6	5	10	0,836609337	0,831900082	0,821560197	0,830023205	0,007698127	0,836609337
62	gini	sqrt	6	5	50	0,833333333	0,839066339	0,840704341	0,837701338	0,003870447	0,840704341
63	gini	sqrt	6	5	100	0,842239967	0,840909091	0,839987715	0,841045591	0,001132314	0,842239967
64	gini	sqrt	6	10	10	0,852784603	0,853808354	0,852886978	0,853159978	0,000563838	0,853808354
65	gini	sqrt	6	10	50	0,855241605	0,855958231	0,854115479	0,855105105	0,000928928	0,855958231
66	gini	sqrt	6	10	100	0,856162981	0,856367731	0,856674857	0,856401856	0,000257638	0,856674857
67	gini	sqrt	6	25	10	0,856930794	0,854422604	0,855190418	0,855514606	0,001285137	0,856930794
68	gini	sqrt	6	25	50	0,860257985	0,859642735	0,860462735	0,860121485	0,000426222	0,860462735
69	gini	sqrt	6	25	100	0,862510238	0,861076986	0,862510238	0,862032487	0,000827488	0,862510238
70	gini	sqrt	6	Full	10	0,851300164	0,851248976	0,850890663	0,851146601	0,000223121	0,851300164
71	gini	sqrt	6	Full	50	0,855241605	0,856060606	0,855958231	0,855753481	0,000446243	0,856060606
72	gini	sqrt	6	Full	100	0,856470106	0,857289107	0,859336609	0,857698608	0,001476475	0,859336609

73	gini	log2	2	5	10	0,821253071	0,807022932	0,789619165	0,805965056	0,015843464	0,821253071
74	gini	log2	2	5	50	0,808353808	0,805794431	0,797911548	0,804019929	0,005442595	0,808353808
75	gini	log2	2	5	100	0,809479934	0,813165438	0,810606061	0,811083811	0,001888629	0,813165438
76	gini	log2	2	10	10	0,848382473	0,846130221	0,842137592	0,845550096	0,003162601	0,848382473
77	gini	log2	2	10	50	0,851658477	0,851044226	0,846642097	0,8497816	0,002736181	0,851658477
78	gini	log2	2	10	100	0,850532351	0,850941851	0,850225225	0,850566476	0,00035953	0,850941851
79	gini	log2	2	25	10	0,850481163	0,851658477	0,853910729	0,85201679	0,001742634	0,853910729
80	gini	log2	2	25	50	0,855702293	0,855958231	0,858927109	0,856862544	0,001792539	0,858927109
81	gini	log2	2	25	100	0,857903358	0,858722359	0,858824734	0,858483483	0,000505004	0,858824734
82	gini	log2	2	Full	10	0,840704341	0,837479525	0,841728092	0,839970652	0,002217274	0,841728092
83	gini	log2	2	Full	50	0,848024161	0,847102785	0,8495086	0,848211848	0,00121384	0,8495086
84	gini	log2	2	Full	100	0,847358722	0,851197789	0,847921785	0,848826099	0,002073148	0,851197789
85	gini	log2	4	5	10	0,819512695	0,816339066	0,829135954	0,821662572	0,006663825	0,829135954
86	gini	log2	4	5	50	0,81541769	0,819922195	0,817669943	0,817669943	0,002252252	0,819922195
87	gini	log2	4	5	100	0,813370188	0,822993448	0,806920557	0,814428064	0,008088497	0,822993448
88	gini	log2	4	10	10	0,847461097	0,842444717	0,848382473	0,846096096	0,003195568	0,848382473
89	gini	log2	4	10	50	0,846539722	0,849610975	0,846642097	0,847597598	0,001744387	0,849610975
90	gini	log2	4	10	100	0,84971335	0,84971335	0,850839476	0,850088725	0,000650169	0,850839476
91	gini	log2	4	25	10	0,852733415	0,850685913	0,852886978	0,852102102	0,001228857	0,852886978
92	gini	log2	4	25	50	0,858517609	0,858517609	0,858671171	0,858568796	8,86594E-05	0,858671171
93	gini	log2	4	25	100	0,858108108	0,86015561	0,859029484	0,859097734	0,001025456	0,86015561
94	gini	log2	4	Full	10	0,846795659	0,847102785	0,844185094	0,846027846	0,001603241	0,847102785
95	gini	log2	4	Full	50	0,853296478	0,851146601	0,853603604	0,852682228	0,001338728	0,853603604
96	gini	log2	4	Full	100	0,853194103	0,854780917	0,853552416	0,853842479	0,000832224	0,854780917
97	gini	log2	6	5	10	0,823198198	0,794226044	0,804258804	0,807227682	0,014712481	0,823198198
98	gini	log2	6	5	50	0,806203931	0,810503686	0,811220311	0,809309309	0,002713101	0,811220311
99	gini	log2	6	5	100	0,811629812	0,816748567	0,814291564	0,814223314	0,00256006	0,816748567
100	gini	log2	6	10	10	0,850941851	0,846334971	0,843673219	0,846983347	0,003677438	0,850941851
101	gini	log2	6	10	50	0,849201474	0,844799345	0,848791974	0,847597598	0,002431992	0,849201474
102	gini	log2	6	10	100	0,853194103	0,850020475	0,851351351	0,851521977	0,001593679	0,853194103
103	gini	log2	6	25	10	0,853040541	0,851965602	0,851300164	0,852102102	0,000878181	0,853040541
104	gini	log2	6	25	50	0,857289107	0,859848485	0,857596233	0,858244608	0,001397461	0,859848485
105	gini	log2	6	25	100	0,859438984	0,857596233	0,856879607	0,857971608	0,001320335	0,859438984
106	gini	log2	6	Full	10	0,848382473	0,847717035	0,849559787	0,848553099	0,00093315	0,849559787
107	gini	log2	6	Full	50	0,855446355	0,854013104	0,852375102	0,853944854	0,001536764	0,855446355
108	gini	log2	6	Full	100	0,854115479	0,854627355	0,855241605	0,85466148	0,000563838	0,855241605
109	entropy	All	2	5	10	0,847256347	0,843673219	0,847051597	0,845993721	0,00201222	0,847256347
110	entropy	All	2	5	50	0,848484848	0,848382473	0,847870598	0,848245973	0,00032909	0,848484848
111	entropy	All	2	5	100	0,848280098	0,848280098	0,848382473	0,848314223	5,91063E-05	0,848382473
112	entropy	All	2	10	10	0,85954136	0,856060606	0,856828419	0,857476795	0,001828717	0,85954136
113	entropy	All	2	10	50	0,857698608	0,858005733	0,859131859	0,858278733	0,000754619	0,859131859
114	entropy	All	2	10	100	0,857391482	0,858824734	0,858415233	0,858210483	0,000738237	0,858824734
115	entropy	All	2	25	10	0,85012285	0,846232596	0,847102785	0,84781941	0,002041735	0,85012285
116	entropy	All	2	25	50	0,858261671	0,854883292	0,857596233	0,856913732	0,001789613	0,858261671
117	entropy	All	2	25	100	0,85744267	0,858824734	0,855855856	0,85737442	0,001485615	0,858824734
118	entropy	All	2	Full	10	0,845208845	0,84645782	0,847972973	0,845942533	0,001780807	0,847972973
119	entropy	All	2	Full	50	0,853296478	0,854013104	0,852938165	0,853415916	0,000547332	0,854013104
120	entropy	All	2	Full	100	0,854320229	0,85452498	0,856418919	0,855088043	0,00115711	0,856418919
121	entropy	All	4	5	10	0,843570844	0,842137592	0,843980344	0,843229593	0,000967612	0,843980344
122	entropy	All	4	5	50	0,847461097	0,848689599	0,847563473	0,847904723	0,000681647	0,848689599
123	entropy	All	4	5	100	0,847256347	0,845925471	0,847256347	0,846812722	0,000768382	0,847256347
124	entropy	All	4	10	10	0,857289107	0,856674857	0,85954136	0,857835108	0,001509237	0,85954136
125	entropy	All	4	10	50	0,858927109	0,858210483	0,858005733	0,858381108	0,000483806	0,858927109
126	entropy	All	4	10	100	0,857493857	0,857698608	0,857800983	0,857664483	0,000156381	0,857800983
127	entropy	All	4	25	10	0,850788288	0,850634726	0,852989353	0,851470789	0,001317355	0,852989353
128	entropy	All	4	25	50	0,859234234	0,859131859	0,858568796	0,858978296	0,000358313	0,859234234
129	entropy	All	4	25	100	0,858005733	0,856879607	0,858312858	0,857732733	0,000754619	0,858312858
130	entropy	All	4	Full	10	0,848536036	0,84720516	0,848433661	0,848058286	0,0007406	0,848536036
131	entropy	All	4	Full	50	0,855804668	0,855804668	0,853654791	0,855088043	0,001241232	0,855804668
132	entropy	All	4	Full	100	0,855651106	0,856772232	0,856265356	0,856231231	0,000563838	0,856772232
133	entropy	All	6	5	10	0,843161343	0,849201474	0,845823096	0,846061971	0,003027143	0,849201474
134	entropy	All	6	5	50	0,849406224	0,848280098	0,846232596	0,847972973	0,001608951	0,849406224
135	entropy	All	6	5	100	0,848894349	0,848177723	0,843980344	0,847017472	0,002654524	0,848894349
136	entropy	All	6	10	10	0,857596233	0,858875921	0,859029484	0,858500546	0,000786913	0,859029484
137	entropy	All	6	10	50	0,858108108	0,858005733	0,858619984	0,858244608	0,00032909	0,858619984
138	entropy	All	6	10	100	0,858108108	0,858210483	0,857800983	0,858039858	0,000213111	0,858210483
139	entropy	All	6	25	10	0,851197789	0,852323915	0,847256347	0,85025935	0,002660932	0,852323915
140	entropy	All	6	25	50	0,856674857	0,86015561	0,861588862	0,859473109	0,002527096	0,861588862
141	entropy	All	6	25	100	0,857186732	0,860257985	0,861281736	0,859575485	0,002131108	0,861281736
142	entropy	All	6	Full	10	0,848382473	0,850993038	0,848126536	0,849167349	0,001586264	0,850993038
143	entropy	All	6	Full	50	0,856367731	0,856674857	0,854627355	0,855889981	0,001104197	0,856674857
144	entropy	All	6	Full	100	0,856879607	0,856674857	0,856674857	0,856743107	0,000118213	0,856879607
145	entropy	sqrt	2	5	10	0,815929566	0,828931204	0,832719083	0,825859951	0,008806044	0,832719083
146	entropy	sqrt	2	5	50	0,838144963	0,838452088	0,838042588	0,838213213	0,000213111	0,838452088
147	entropy	sqrt	2	5	100	0,83527846	0,840397215	0,831900082	0,835858586	0,004278169	0,840397215
148	entropy	sqrt	2	10	10	0,855036855	0,853910729	0,852784603	0,853910729	0,001126126	0,855036855
149	entropy	sqrt	2	10	50	0,855446355	0,85472973	0,855446355	0,85520748	0,000413744	0,855446355
150	entropy	sqrt	2	10	100	0,856060606	0,855651106	0,856162981	0,855958231	0,000270859	0,856162981

151	entropy	sqrt	2	25	10	0,849764537	0,8495086	0,856418919	0,851897352	0,003917882	0,856418919
152	entropy	sqrt	2	25	50	0,85744267	0,855497543	0,859797297	0,85757917	0,002153125	0,859797297
153	entropy	sqrt	2	25	100	0,85954136	0,85995086	0,860053235	0,859848485	0,000270859	0,860053235
154	entropy	sqrt	2	Full	10	0,843877969	0,844338657	0,843519656	0,843912094	0,000410565	0,844338657
155	entropy	sqrt	2	Full	50	0,851760852	0,849559787	0,85012285	0,850481163	0,001143443	0,851760852
156	entropy	sqrt	2	Full	100	0,850993038	0,850941851	0,849969287	0,850634726	0,000576854	0,850993038
157	entropy	sqrt	4	5	10	0,842956593	0,835892711	0,832616708	0,837155337	0,005284314	0,842956593
158	entropy	sqrt	4	5	50	0,836302211	0,836609337	0,838247338	0,837052962	0,001045697	0,838247338
159	entropy	sqrt	4	5	100	0,840397215	0,839066339	0,839168714	0,83954409	0,0007406	0,840397215
160	entropy	sqrt	4	10	10	0,852886978	0,850225225	0,850020475	0,851044226	0,00159915	0,852886978
161	entropy	sqrt	4	10	50	0,853808354	0,853398853	0,855446355	0,854217854	0,001083436	0,855446355
162	entropy	sqrt	4	10	100	0,856162981	0,85472973	0,855446355	0,855446355	0,000716626	0,856162981
163	entropy	sqrt	4	25	10	0,856367731	0,855599918	0,855855856	0,855941168	0,000390951	0,856367731
164	entropy	sqrt	4	25	50	0,86015561	0,860872236	0,859234234	0,86008736	0,000821131	0,860872236
165	entropy	sqrt	4	25	100	0,859643735	0,860462735	0,862100737	0,860735736	0,001251044	0,862100737
166	entropy	sqrt	4	Full	10	0,846488534	0,84781941	0,845976658	0,846761534	0,000951226	0,84781941
167	entropy	sqrt	4	Full	50	0,853603604	0,854064292	0,854371417	0,854013104	0,000386458	0,854371417
168	entropy	sqrt	4	Full	100	0,853603604	0,853091728	0,85513923	0,853944854	0,001065554	0,85513923
169	entropy	sqrt	6	5	10	0,831490581	0,834357084	0,830569206	0,832138957	0,001975424	0,834357084
170	entropy	sqrt	6	5	50	0,841523342	0,841318591	0,839168714	0,840670216	0,001304362	0,841523342
171	entropy	sqrt	6	5	100	0,838861589	0,837325962	0,84049959	0,838895714	0,001587089	0,84049959
172	entropy	sqrt	6	10	10	0,850941851	0,850941851	0,855036855	0,852306852	0,002364252	0,855036855
173	entropy	sqrt	6	10	50	0,854627355	0,855958231	0,853910729	0,854832105	0,001038994	0,855958231
174	entropy	sqrt	6	10	100	0,855958231	0,856060606	0,857084357	0,856367731	0,000622723	0,857084357
175	entropy	sqrt	6	25	10	0,85723792	0,854320229	0,854371417	0,855309855	0,001669949	0,85723792
176	entropy	sqrt	6	25	50	0,861384111	0,860872236	0,861281736	0,861179361	0,000270859	0,861384111
177	entropy	sqrt	6	25	100	0,85995086	0,86015561	0,860769861	0,86029211	0,000426222	0,860769861
178	entropy	sqrt	6	Full	10	0,849150287	0,848740786	0,848791974	0,848894349	0,000223121	0,849150287
179	entropy	sqrt	6	Full	50	0,855036855	0,856981982	0,856674857	0,856231231	0,001045697	0,856981982
180	entropy	sqrt	6	Full	100	0,857289107	0,856060606	0,855958231	0,856435981	0,0007406	0,857289107
181	entropy	log2	2	5	10	0,796171171	0,809889435	0,806203931	0,804088179	0,007099648	0,809889435
182	entropy	log2	2	5	50	0,803644554	0,803746929	0,812244062	0,806545182	0,004935641	0,812244062
183	entropy	log2	2	5	100	0,813370188	0,807841933	0,802313677	0,807841933	0,005528256	0,813370188
184	entropy	log2	2	10	10	0,845413595	0,846744472	0,843775594	0,84531122	0,001487084	0,846744472
185	entropy	log2	2	10	50	0,845823096	0,850941851	0,847563473	0,848109473	0,002602691	0,850941851
186	entropy	log2	2	10	100	0,84971335	0,849406224	0,849201474	0,849440349	0,000257638	0,84971335
187	entropy	log2	2	25	10	0,853091728	0,849047912	0,847665848	0,849935162	0,002819655	0,853091728
188	entropy	log2	2	25	50	0,858415233	0,855599918	0,85785217	0,857289107	0,001489725	0,858415233
189	entropy	log2	2	25	100	0,859183047	0,85534398	0,856674857	0,857067295	0,001949388	0,859183047
190	entropy	log2	2	Full	10	0,838503276	0,840448403	0,838298526	0,839083402	0,001186551	0,840448403
191	entropy	log2	2	Full	50	0,849201474	0,849355037	0,851248976	0,849935162	0,001140384	0,851248976
192	entropy	log2	2	Full	100	0,849815725	0,848894349	0,848945536	0,849218537	0,000517813	0,849815725
193	entropy	log2	4	5	10	0,812653563	0,805589681	0,799242424	0,805828556	0,00670876	0,812653563
194	entropy	log2	4	5	50	0,809070434	0,800880426	0,806818182	0,805589681	0,004230954	0,809070434
195	entropy	log2	4	5	100	0,807841933	0,804258804	0,805794431	0,805965056	0,001797648	0,807841933
196	entropy	log2	4	10	10	0,851658477	0,852375102	0,840909091	0,848314223	0,006423035	0,852375102
197	entropy	log2	4	10	50	0,84531122	0,849303849	0,848484848	0,847699973	0,002108861	0,849303849
198	entropy	log2	4	10	100	0,846744472	0,848996724	0,849406224	0,848382473	0,001433251	0,849406224
199	entropy	log2	4	25	10	0,854064292	0,851095414	0,855855856	0,853671854	0,002404362	0,855855856
200	entropy	log2	4	25	50	0,860872236	0,859029484	0,858517609	0,859473109	0,001238414	0,860872236
201	entropy	log2	4	25	100	0,85954136	0,858415233	0,857596233	0,858517609	0,000976596	0,85954136
202	entropy	log2	4	Full	10	0,844645782	0,847563473	0,844236282	0,845481845	0,001814332	0,847563473
203	entropy	log2	4	Full	50	0,851351351	0,852579853	0,852886978	0,852272727	0,000812577	0,852886978
204	entropy	log2	4	Full	100	0,854013104	0,854422604	0,853296478	0,853910729	0,00057	0,854422604
205	entropy	log2	6	5	10	0,795966421	0,813574939	0,789414414	0,799651925	0,012494796	0,813574939
206	entropy	log2	6	5	50	0,797502048	0,806408681	0,807534808	0,803815179	0,00549625	0,807534808
207	entropy	log2	6	5	100	0,817260442	0,811015561	0,809684685	0,812653563	0,004044788	0,817260442
208	entropy	log2	6	10	10	0,836916462	0,844594595	0,841420966	0,840977341	0,003858242	0,844594595
209	entropy	log2	6	10	50	0,851453726	0,848177723	0,848996724	0,849542725	0,001704886	0,851453726
210	entropy	log2	6	10	100	0,84971335	0,8495086	0,848177723	0,849133224	0,000833797	0,84971335
211	entropy	log2	6	25	10	0,855190418	0,854166667	0,854780917	0,854712667	0,000515277	0,855190418
212	entropy	log2	6	25	50	0,854627355	0,860462735	0,857903358	0,857664483	0,002925015	0,860462735
213	entropy	log2	6	25	100	0,857903358	0,860462735	0,858824734	0,859063609	0,001296302	0,860462735
214	entropy	log2	6	Full	10	0,84700041	0,850429975	0,848126536	0,848518974	0,001748138	0,850429975
215	entropy	log2	6	Full	50	0,854217854	0,852579853	0,852989353	0,853262353	0,000852443	0,854217854
216	entropy	log2	6	Full	100	0,855036855	0,85452498	0,852477477	0,854013104	0,001354295	0,855036855