ENTREGA FINAL DEL PROYECTO

POR

Natalia Marcela Henao

43.463.025

Andrés Felipe Duque Daza

1.152.706.282

Julián Camilo Duque Ospina

1.036.656.349

Introducción A la inteligencia Artificial

Raul Ramos Pollan

Universidad De Antioquia

Facultad De Ingeniería

Medellín, noviembre de 2023

Contenido

Introducción

La naturaleza nos presenta su diversidad de muchas maneras diferentes y diversidad en formas, texturas, tamaños, etc. despierta la curiosidad por conocer sus características y cómo se puede expresar esta diversidad. En este proyecto, el tipo de cubierta forestal (tipo de cubierta arbórea dominante) tuvo que predecirse a partir de variables cartográficas precisas (a diferencia de los datos de teledetección). El conjunto de datos es producido y proporcionado por el Sistema de información de recursos de la Región 2 del Servicio Forestal de los Estados Unidos (USFS) y

el Servicio Geológico de los Estados Unidos. Los datos están en formato bruto (no escalados) y contienen columnas de datos binarios para variables cualitativas independientes, como la naturaleza y el tipo de suelo.

El área de estudio son cuatro áreas silvestres ubicadas en el Bosque Nacional Roosevelt del norte de Colorado. Cada observación es un parche de 30m x 30m. Se le pide que prediga una clasificación entera para el tipo de cubierta forestal. Los siete tipos son:

1. Picea/abeto
2. Pino Lodgepole
3. Pino Ponderosa
4. Álamo/Sauce
5. Álamo temblón
6. Abeto de Douglas
7. Krummholz

El conjunto de entrenamiento (15120 observaciones) contiene tanto entidades como

Cover_Type (el tipo de cubierta forestal). El conjunto de prueba contiene solo las funciones. Debe predecir Cover_Type para cada fila en el conjunto de prueba (565892 observaciones).

Exploración Descriptiva

El dataset que usaremos es de una competencia de kaggle en la cual se proporcionan datos cartográficos para cada celda de 30m x 30m de corteza forestal, estos son:

- **Elevation** - Elevación en metros.
- **Aspect** - Aspecto en grados de acimut.
- **Slope** - Pendiente en grados.
- **Horizontal_Distance_To_Hydrology** - Horz Dist a las características de agua superficial más cercanas.
- **Vertical_Distance_To_Hydrology -** Vert Dist a las características de agua superficial más cercanas.
- **Horizontal_Distance_To_Roadways -** Horz Dist a la carretera más cercana.
- **Hillshade_9am (índice 0 a 255)** - Hillshade índice a las 9 a. m., solsticio de verano.
- **Hillshade_Noon (índice de 0 a 255)** - Índice de sombreado al mediodía, solsticio de verano.
- **Hillshade_3pm (índice de 0 a 255)** - Índice de sombreado a las 3 p.m., solsticio de verano.
- **Horizontal_Distance_To_Fire_Points-** Dist Horz a los puntos de ignición de incendios forestales más cercanos.
- **Wilderness_Area (4 columnas binarias, 0 = ausencia o 1 = presencia)**

- Designación de área silvestre.

• **Soil_Type (40 columnas binarias, 0 = ausencia o 1 = presencia)** -

Designación de tipo de suelo.

• **Cover_Type (7 tipos, números enteros 1 a 7)** - Designación del tipo de

cubierta forestal.

- Para conocer cuáles son los demás datos del dataset los invito a que visiten

la página oficial de kaggle de este proyecto:

• https://www.kaggle.com/competitions/forest-cover-type-prediction/data

Archivos Utilizados

- train.csv:

Es un archivo con los datos de entrenamiento (con 15120 instancias), descritos

anteriormente.

- test.csv:

Son los datos de prueba (con más de 500.000 instancias), que tiene la misma naturaleza que los datos de entrenamiento, en este caso hay que predecir la columna Cover_Type, con el tipo de corteza a la que pertenezca del 1 al 7.

- sample_submission.csv:

Un archivo de envío de muestra en el formato correcto.

Métricas

La métrica de evaluación principal para el modelo será el porcentaje de precisión

multiclases que se representa de la siguiente manera:

$$ACC = \frac{TP + TN}{P + N} = \frac{TP + TN}{TP + TN + FP + FN}$$

Donde:

P: Condición positiva. El número de casos positivos reales en los datos.

N: Condición negativa. El número de casos negativos reales en los datos.

TP: True positive. Un resultado de prueba que indica correctamente la presencia de una condición o característica.

TN: True negative. Un resultado de prueba que indica correctamente la ausencia de una condición o característica.

FP: False positive. Un resultado de prueba que indica erróneamente que una condición o atributo en particular está presente.

FN: False negative. Un resultado de prueba que indica erróneamente que una condición o atributo en particular está ausente.

Iteraciones de Desarrollo

Una vez que entendimos el problema que teníamos que resolver, comenzamos a revisar los datos del archivo train.csv. Para ello, primero los descargamos desde Kaggle a Google Colab. Luego, utilizamos la librería pandas para leer y analizar los datos.

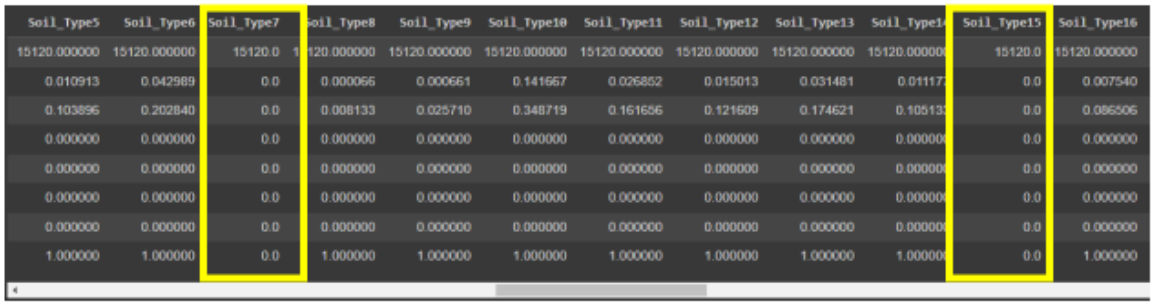
Nuestro primer hallazgo fue que cada columna del archivo train.csv contiene 15.120 datos. Además, todos los datos son de tipo entero, no hay datos flotantes, y no hay datos faltantes ni nulos. Esto es una buena señal, ya que indica que los datos se recolectaron cuidadosamente.

Descripción Estadística

Continuamos haciendo una revisión a fondo del dataframe y realizamos una descripción estadística que nos permita obtener de cada una de las columnas de entrenamiento (train.csv) los siguientes datos:

- Conteo de datos almacenados.
- Media aritmética.
- Desviación estándar.
- Valor mínimo.
- Percentil 25%.
- Percentil 50%.
- Percentil 75%.
- Valor máximo.

En este punto nos encontramos con unos datos inquietantes en las columnas de tipo de suelo 7 y 15 (Soil_Type7, Soil_Type15), en las cuales todos los datos en el análisis estadístico son 0.



Soil_Type5	Soil_Type6	Soil_Type7	oil_Type8	Soil_Type9	Soil_Type10	Soil_Type11	Soil_Type12	Soil_Type13	Soil_Type1	Soil_Type15	Soil_Type16
15120.000000	15120.000000	15120.0	1 120.000000	15120.000000	15120.000000	15120.000000	15120.000000	15120.000000	15120.00000	15120.0	15120.000000
0.010913	0.042989	0.0	0.000066	0.000661	0.141667	0.026852	0.015013	0.031481	0.01117	0.0	0.007540
0.103896	0.202840	0.0	0.008133	0.025710	0.348719	0.161656	0.121609	0.174621	0.10513	0.0	0.086506
0.000000	0.000000	0.0	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.00000	0.0	0.000000
0.000000	0.000000	0.0	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.00000	0.0	0.000000
0.000000	0.000000	0.0	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.00000	0.0	0.000000
0.000000	0.000000	0.0	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.00000	0.0	0.000000
1.000000	1.000000	0.0	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.00000	0.0	1.000000

Nos enfrentamos a un desafío intrigante: la incertidumbre respecto a los valores iguales a 0 en nuestros datos. Es crucial determinar si estos ceros indican una medición incorrecta o si se han introducido intencionadamente para evitar campos vacíos y problemas asociados con los datos nulos. Esta ambigüedad representa una de las dificultades clave en nuestro proyecto en este punto. A pesar de esta incertidumbre, no tenemos más opción que trabajar con los datos tal como

están hasta el momento. Para abordar esta situación, verificamos el mensaje presente en cada columna para evaluar la necesidad de correcciones posteriores.

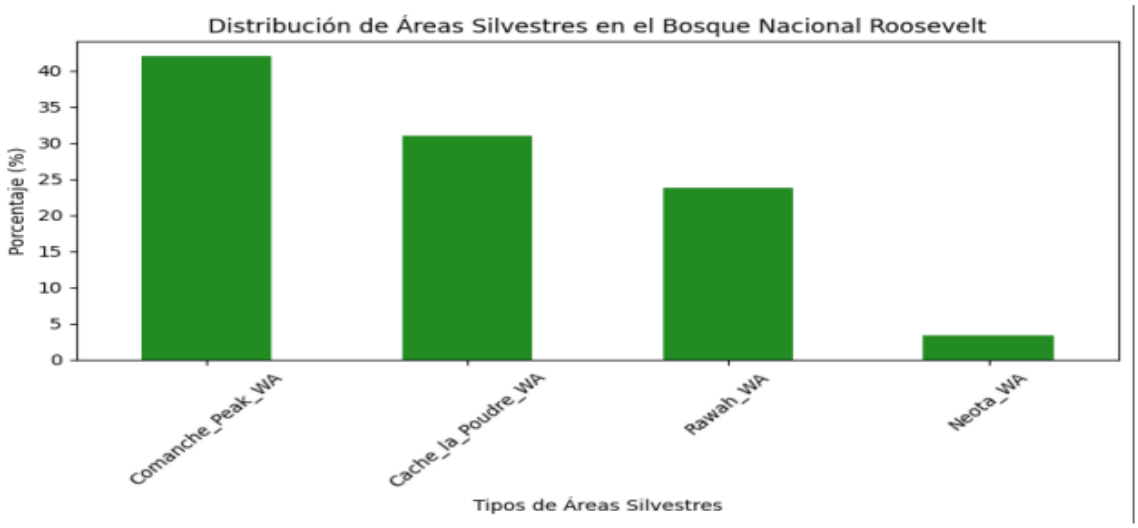
Este análisis nos permitirá entender mejor la naturaleza de los valores iguales a 0 y determinar si tienen un impacto significativo en nuestro conjunto de datos.

Además, aunque nos encontramos con esta incertidumbre en los datos almacenados en el conjunto de entrenamiento, seguimos avanzando con nuestra exploración.

Realizamos un análisis detallado y descubrimos un patrón interesante: cada tipo de corteza forestal designado en el archivo de entrenamiento cuenta con exactamente 2,160 instancias. Esta uniformidad en la distribución de las clases es valiosa, ya que nos proporciona un equilibrio inicial en nuestro conjunto de datos, lo que puede ser crucial para el entrenamiento efectivo de nuestro modelo en etapas posteriores del proyecto.

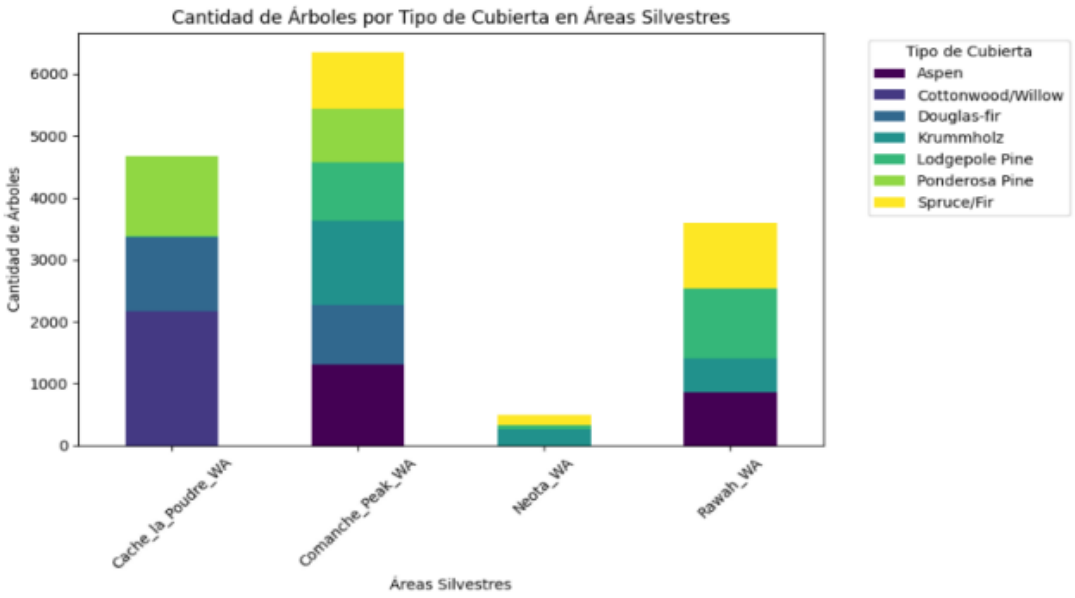
A pesar de las incertidumbres iniciales, esta consistencia en la distribución de las clases es un hallazgo valioso. Nos brinda una base sólida sobre la cual construir nuestras investigaciones y análisis futuros.

Continuamos con la obtención de los datos de las distintas áreas silvestres (4). Obtenemos un gráfico de barras que nos muestra desde el área silvestre con mayor cantidad de datos hasta la con menor cantidad de datos:



Gráfica 1. Porcentaje de las áreas silvestres con sus cantidades de datos obtenidos.

Realizamos un histograma compuesto en el cual evidenciamos que los 7 tipos de corteza que se encuentran en el parque están distribuidos por las cuatro zonas y que la zona con mayor variedad de cortezas está en Comanche Peak WA teniendo 6 de las siete variedades y Neota Wasolo 3.



Gráfica 2. Presencia de los diferentes tipos de corteza en cada una de las áreas silvestres.

Algoritmos de Clasificación

Finalmente, utilizando la biblioteca Scikit-Learn, hemos avanzado a la fase de generación de modelos de clasificación. Dado que existen nueve tipos de modelos predictivos, nuestro objetivo es determinar cuál de estos métodos se adapta mejor a las necesidades de nuestro proyecto. Para lograr esto, hemos implementado un bloque de código que emplea un algoritmo de clasificación para probar cada método. Esta estrategia nos proporciona una evaluación precisa de la precisión de cada modelo en relación con los datos del archivo de entrenamiento. Este proceso nos permite

comparar y contrastar la eficacia de diversas técnicas de clasificación, lo que es crucial para seleccionar el modelo más adecuado para nuestro escenario

específico. Esta fase de evaluación es fundamental, ya que nos brinda perspectivas valiosas para la selección del modelo final que emplearemos en nuestro proyecto de predicción de cobertura forestal. Estamos entusiasmados por los resultados que obtendremos y por la posibilidad de aplicar un enfoque sólido y bien fundamentado en nuestro análisis predictivo.

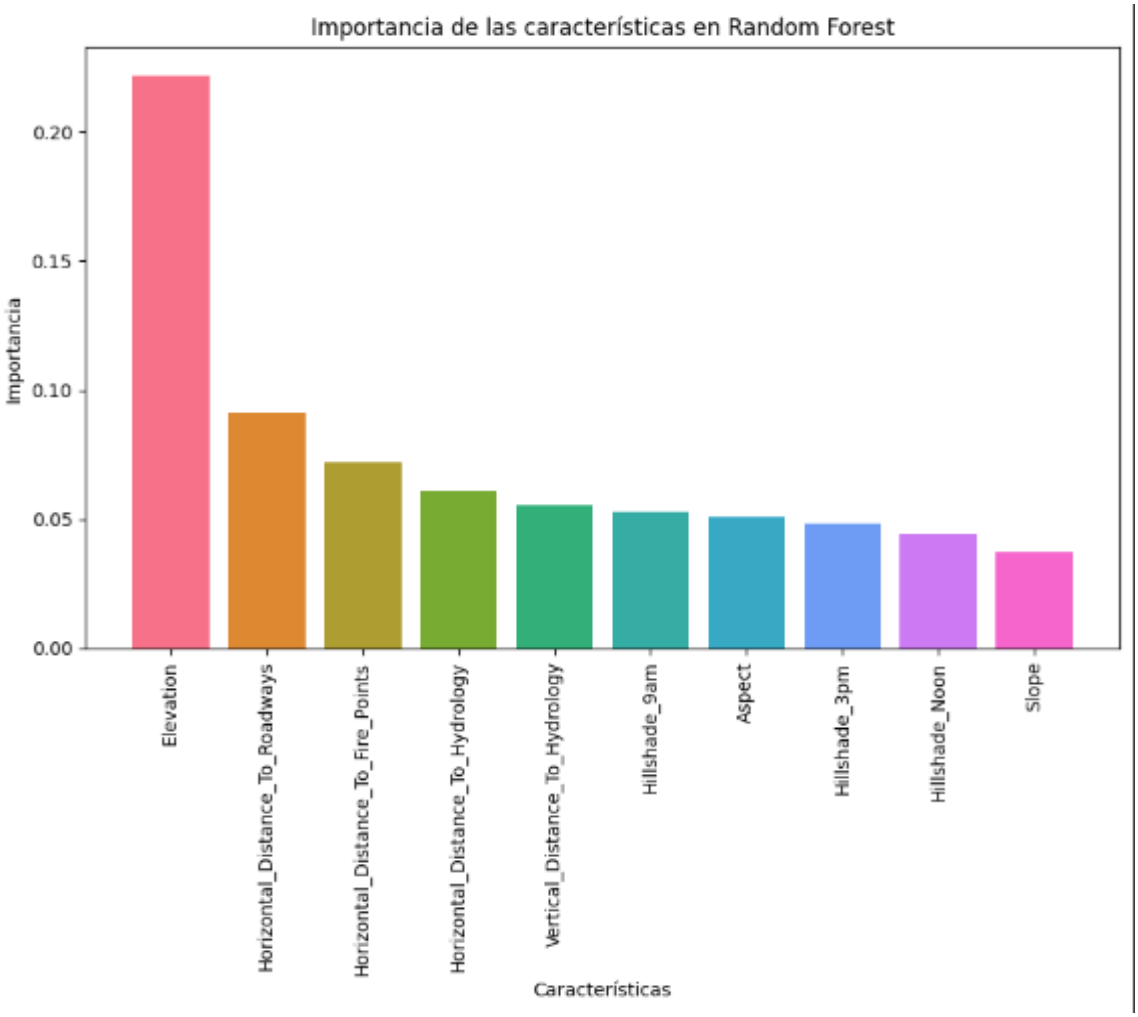
	AccuracyScore	PrecisionScore	RecallScore	f1_Score
RandomForest	0.869378	0.869378	0.869378	0.869378
KNeigbors	0.804894	0.804894	0.804894	0.804894
GradientBoosting	0.802249	0.802249	0.802249	0.802249
DecisionTree	0.787368	0.787368	0.787368	0.787368
SVC	0.625661	0.625661	0.625661	0.625661
Bernoulli	0.610780	0.610780	0.610780	0.610780
Gaussian	0.600860	0.600860	0.600860	0.600860
LogisticRegr	0.474537	0.474537	0.474537	0.474537
AdaBoost	0.346561	0.346561	0.346561	0.346561

Tabla 1. Algoritmos de clasificación ordenados según su puntaje de predicción.

La tabla de resultados nos permite identificar los modelos más y menos precisos. Basándonos en esta información, seleccionamos diferentes modelos para profundizar en sus resultados y comprender las razones de su desempeño.

Randomforest

El primer modelo que analizamos es RandomForest. Según la puntuación, este modelo es el que mejor predice para estos datos. La primera gráfica que generamos muestra la importancia de las características de los datos para la predicción. Para esta gráfica, utilizamos las 10 características más importantes.

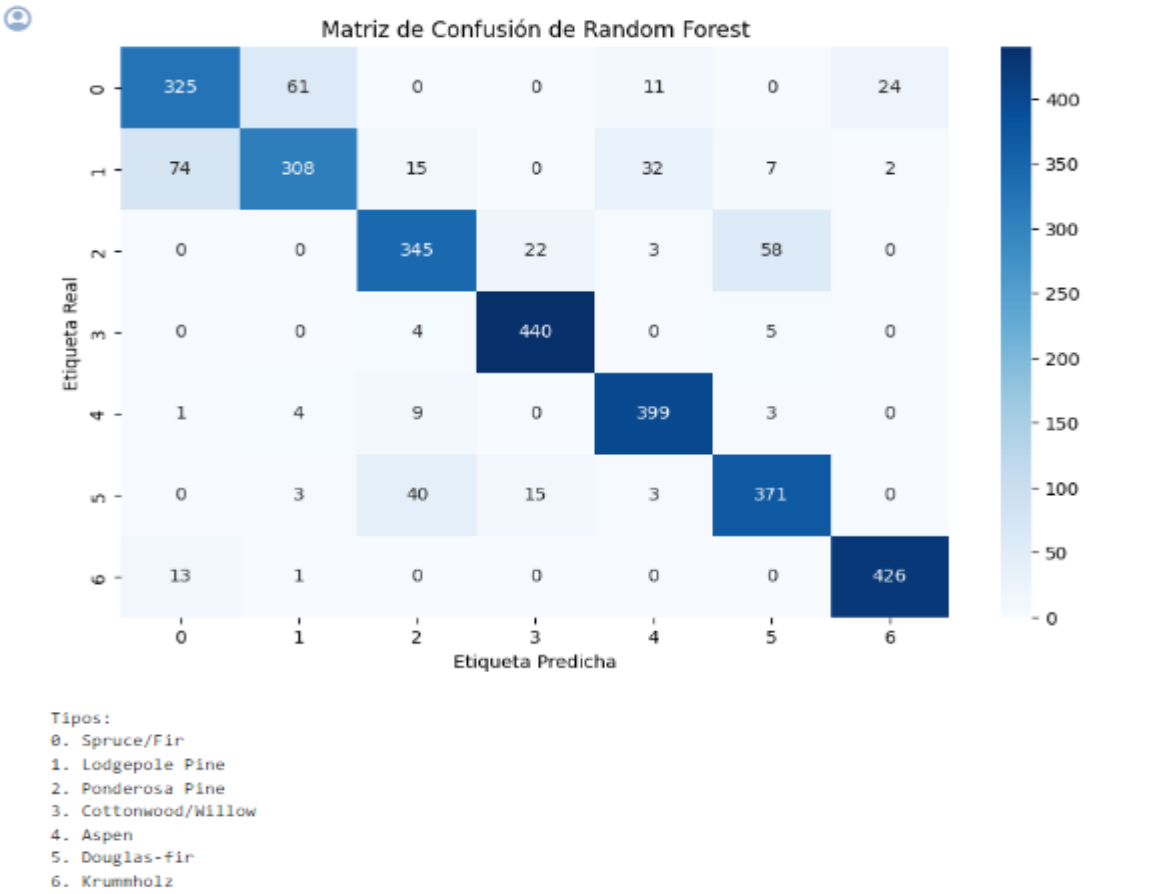


Gráfica 3. Top 10 características más importantes para RandomForest.

A continuación, generamos una matriz de dispersión para visualizar los datos de una manera más intuitiva. El clasificador tuvo dificultades para distinguir entre los dos primeros tipos de corteza, Spruce/Fir y Lodgepole Pine. Esto se refleja en la matriz, donde las cuadrículas correspondientes a estas dos categorías se superponen.

En la matriz, los colores más oscuros representan una mayor densidad de datos. La diagonal de la matriz debería contener todos los valores diferentes de 0, ya que representa las predicciones correctas. Los valores fuera de la diagonal representan las predicciones incorrectas.

Por lo tanto, una matriz de dispersión con muchos 0 fuera de la diagonal indica que el clasificador es preciso. En este caso, el clasificador tuvo algunas predicciones incorrectas, pero en general es preciso.

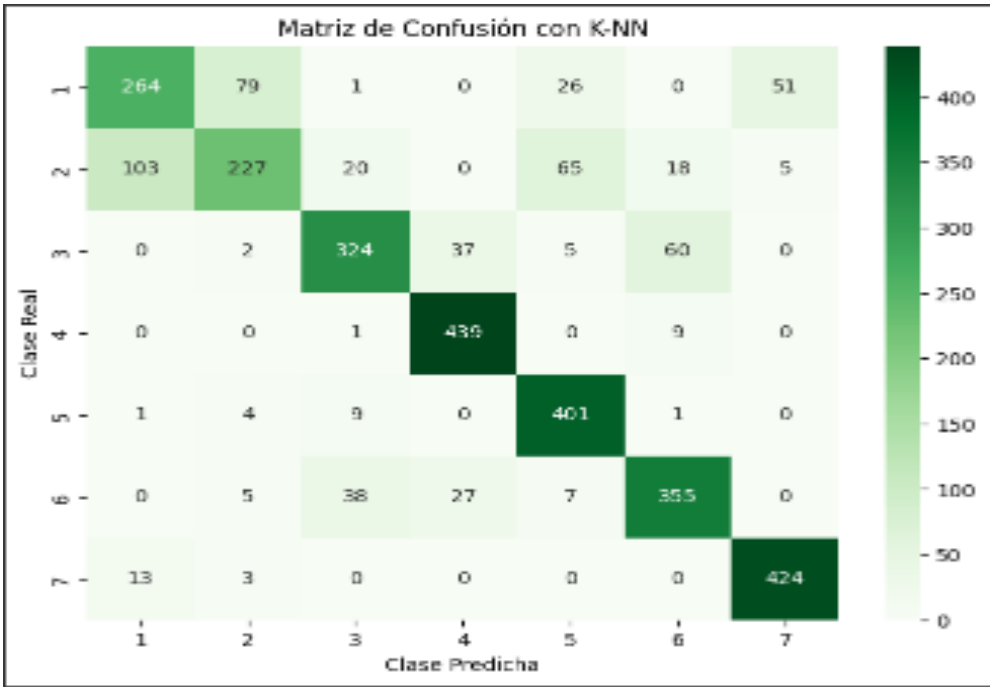


Gráfica 4. Matriz de confusión de RandomForest.

KNN

El siguiente modelo que evaluamos es el algoritmo **KNeighbors**. Para este modelo, elegimos **k = 5**, lo que significa que el clasificador considera los tres vecinos más cercanos de un nuevo punto de datos para predecir su clase.

A continuación, generamos una matriz de confusión para evaluar el desempeño del modelo. La matriz de confusión muestra las predicciones correctas e incorrectas del clasificador para cada tipo de corteza.



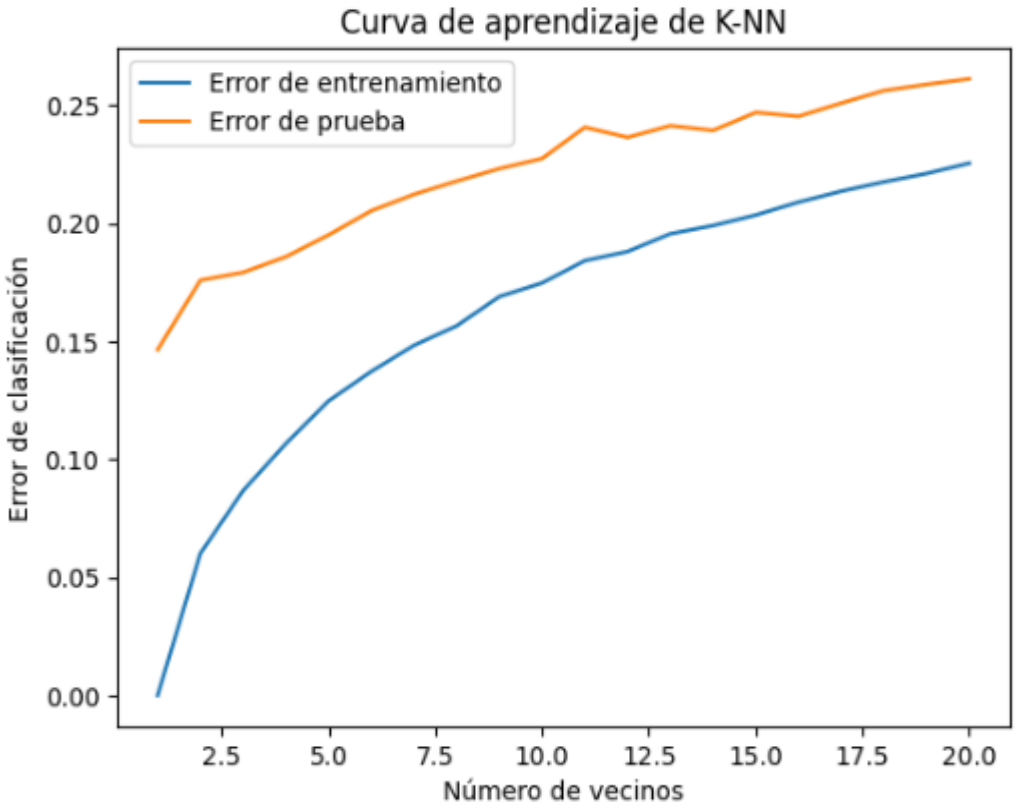

```
Tipos:
0. Spruce/Fir
1. Lodgepole Pine
2. Ponderosa Pine
3. Cottonwood/Willow
4. Aspen
5. Douglas-fir
6. Krummholz

Matriz de confusión:
[[264  79   1   0  26   0  51]
 [103 227  20   0  65  18   5]
 [  0   2 324  37   5  60   0]
 [  0   0   1 439   0   9   0]
 [  1   4   9   0 401   1   0]
 [  0   5  38  27   7 355   0]
 [ 13   3   0   0   0   0 424]]
```

Gráfica 5. Matriz de confusión de KNN.

Si bien **KNeighbors** es el segundo clasificador mejor ubicado en nuestra tabla, presenta algunos cambios que podrían haber afectado el resultado final. Algunas predicciones no difieren de las realizadas por **RandomForest**, y las que sí difieren lo hacen en cantidades bajas.

Para evaluar el desempeño de **KNeighbors**, generamos una curva de aprendizaje. Esta curva muestra cómo el rendimiento del modelo mejora a medida que se incrementa el tamaño del conjunto de entrenamiento. En este caso, la curva tiene una tendencia exponencial, lo que significa que el rendimiento del modelo mejora rápidamente al principio, pero luego se desacelera.



Gráfica 6. Curva de aprendizaje de KNN.

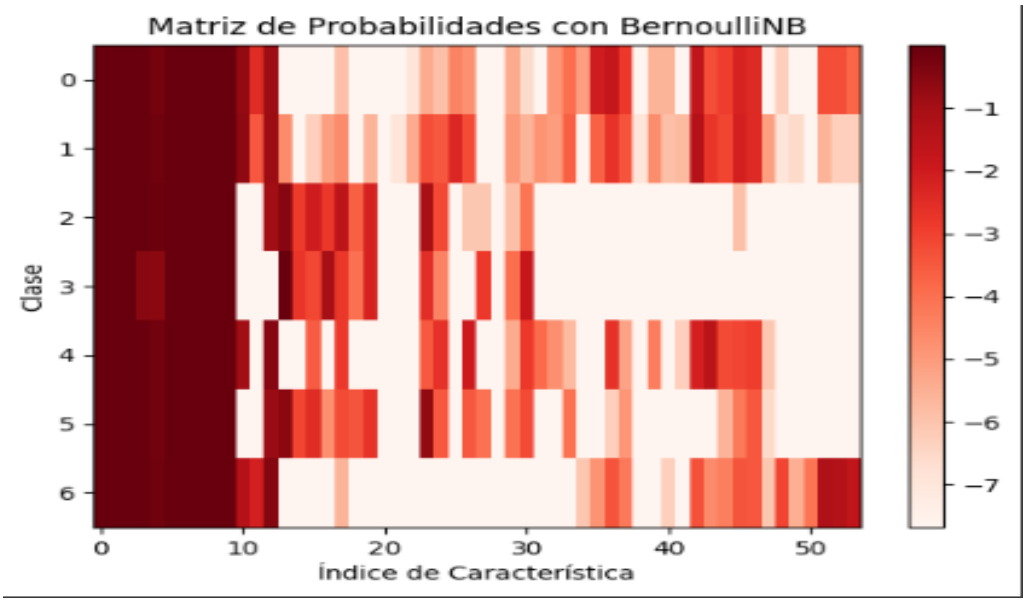
Bernoulli

Pasamos ahora al clasificador **Bernoulli**. Este clasificador es el que obtiene los resultados más bajos de los tres que hemos analizado. Por lo tanto, es de esperar que cometa más errores en las predicciones.

Para evaluar el desempeño de este clasificador, generamos una matriz de probabilidades. Esta matriz muestra la probabilidad de que un punto de datos pertenezca a cada clase, dada la distribución de probabilidad del clasificador.

Al observar la matriz de probabilidades, podemos concluir que las primeras 10 características no son relevantes para el clasificador. Esto se debe a que la mayoría de las clases tienen probabilidades similares para estas características. A medida que cambian las características, podemos observar una mayor variedad de predicciones. Esto se debe a que las características posteriores son más informativas sobre la clase del punto de datos.

Como recordatorio, hay alrededor de 40 tipos de suelo diferentes. Por lo tanto, es difícil para un clasificador binario, como **Bernoulli**, distinguir entre todas estas clases.

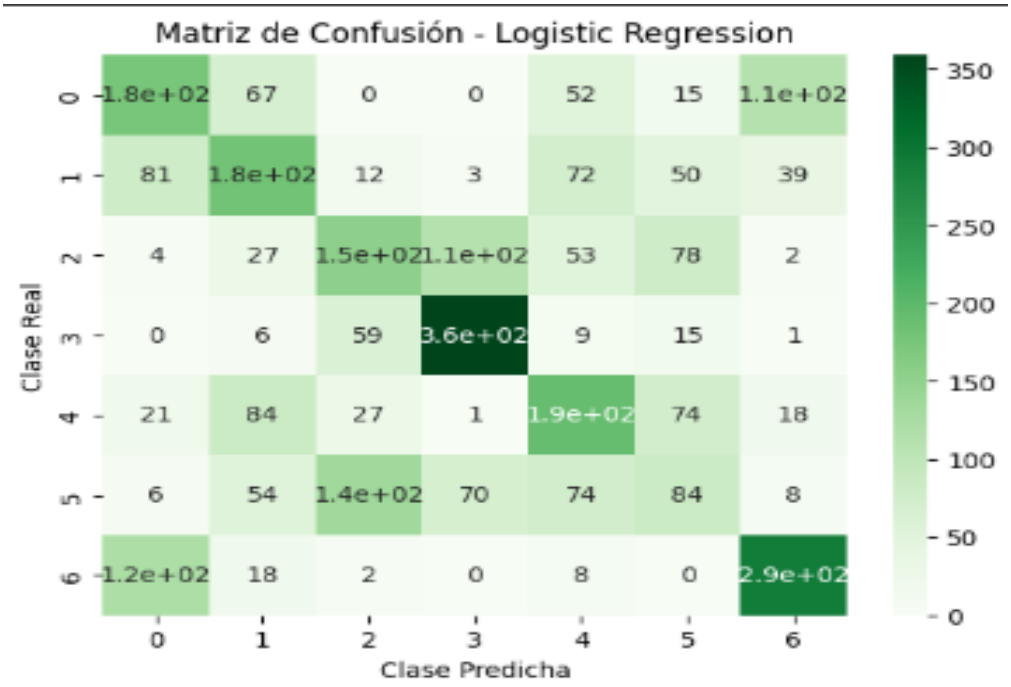


Gráfica 7. Matriz de probabilidades de Bernoulli.

LogisticRegression.

Continuando con el segundo clasificador menos efectivo, **LogisticRegression**, generamos una matriz de confusión. En comparación con las matrices de los otros dos clasificadores, esta matriz

muestra un cambio significativo. Los colores son más claros y están más repartidos por toda la matriz, lo que indica una menor calidad de predicción. Por ello, **LogisticRegression** ocupa el penúltimo lugar en nuestra tabla.



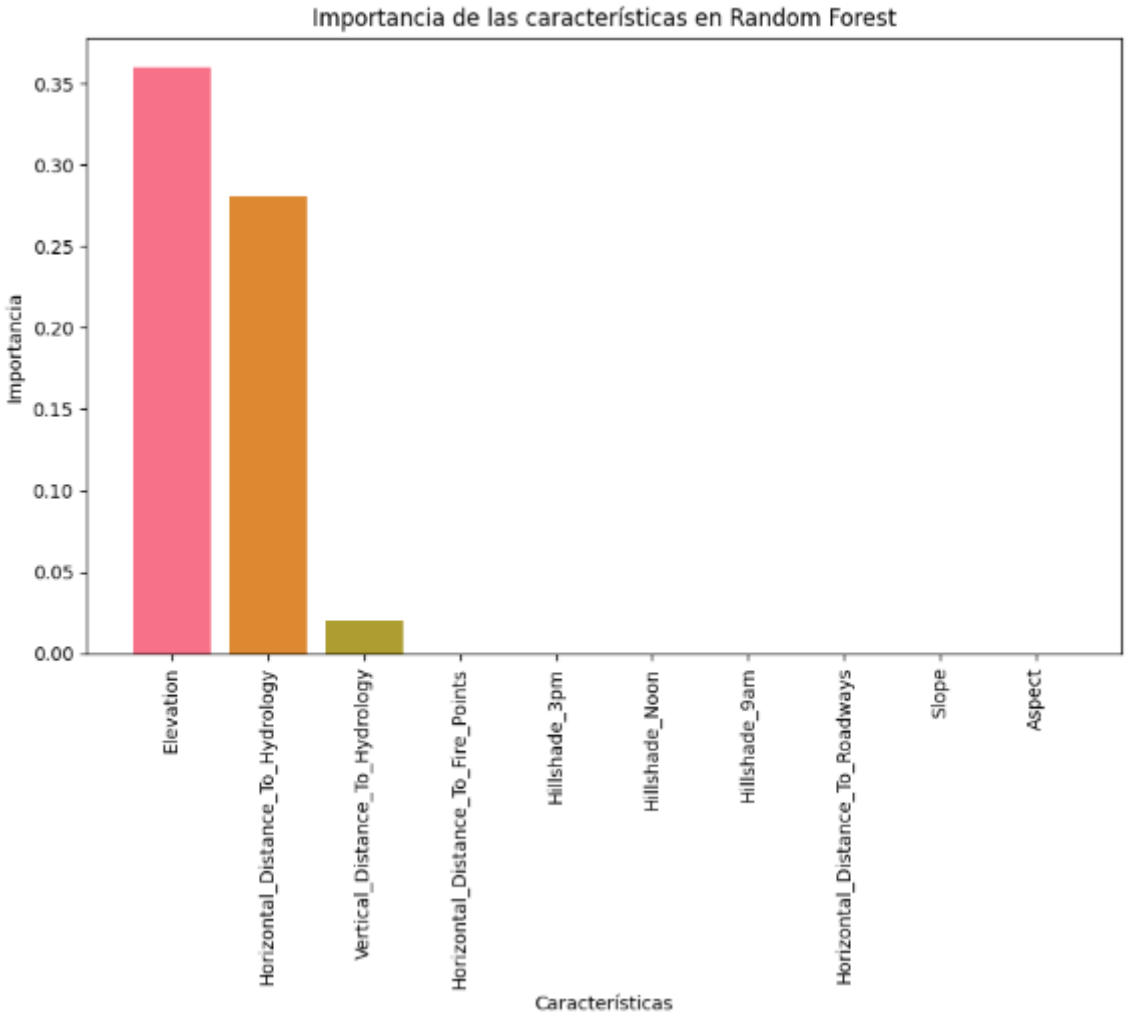

```
Tipos:
0. Spruce/Fir
1. Lodgepole Pine
2. Ponderosa Pine
3. Cottonwood/Willow
4. Aspen
5. Douglas-fir
6. Krummholz

Matriz de confusión:
[[176  67   0   0  52  15 111]
 [ 81 181  12   3  72  50  39]
 [  4  27 153 111  53  78   2]
 [  0   6  59 359   9  15   1]
 [ 21  84  27   1 191  74  18]
 [  6  54 136  70  74  84   8]
 [121  18   2   0   8   0 291]]
```

Gráfica 8. Matriz de confusión de Logistic Regression.

Adaboost.

Para concluir, analizamos el clasificador con menor puntaje, **Adaboost**. Generamos una gráfica de importancia para el modelo, utilizando las 10 primeras características según su importancia.

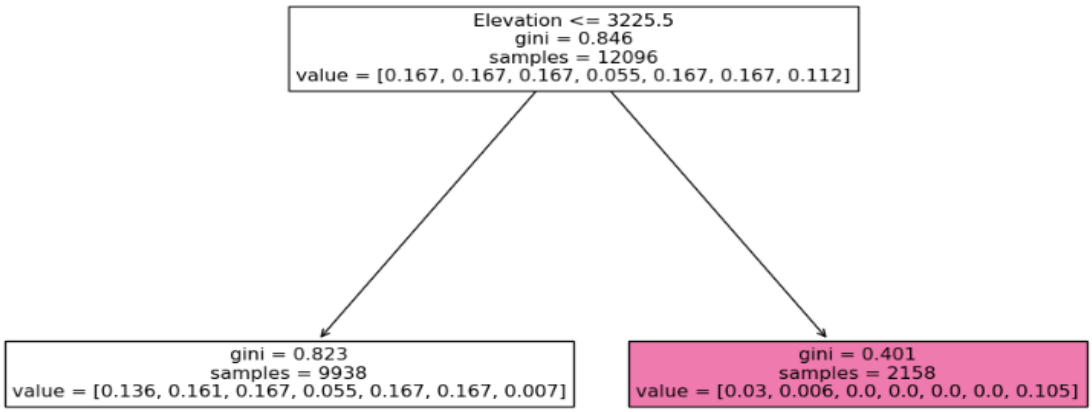


Gráfica 9. características más importantes para Adaboost.

Al ver la gráfica, se observa que 10 clases tienen importancia 0 o mucho menor a 0.05, lo que nos acerca un poco a saber por qué este clasificador ocupa el último lugar de la tabla y la razón para sus predicciones erróneas. Para mostrar más detallado el proceso de este caso en particular, en el notebook se grafican los 50 árboles de decisión que tiene el algoritmo, para este caso y por

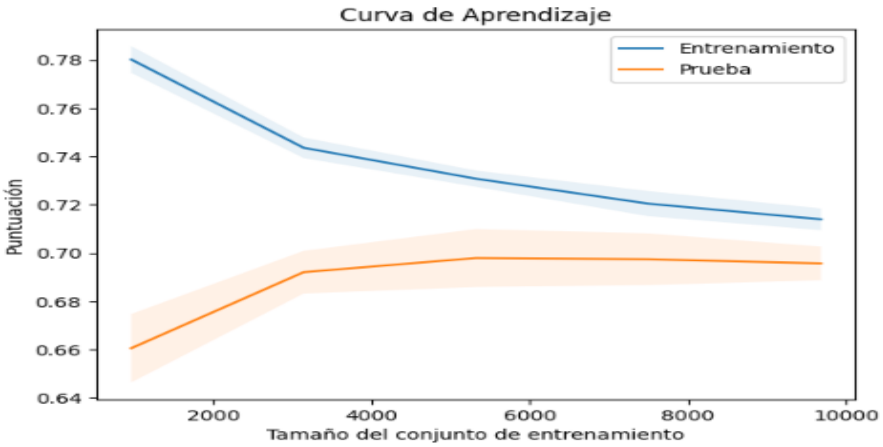
practicidad se decide solo mostrar el primero como parte del ejercicio de análisis. Para conocer más a fondo este algoritmo se puede dirigir al programa y ejecutarlo o simplemente observarlo ya graficado.

https://colab.research.google.com/github/naty0611/Proyecto_IA_Forest_Cover_Type/blob/main/Algoritmos%20clasificadores.ipynb#scrollTo=N2lcOjtu6aTn

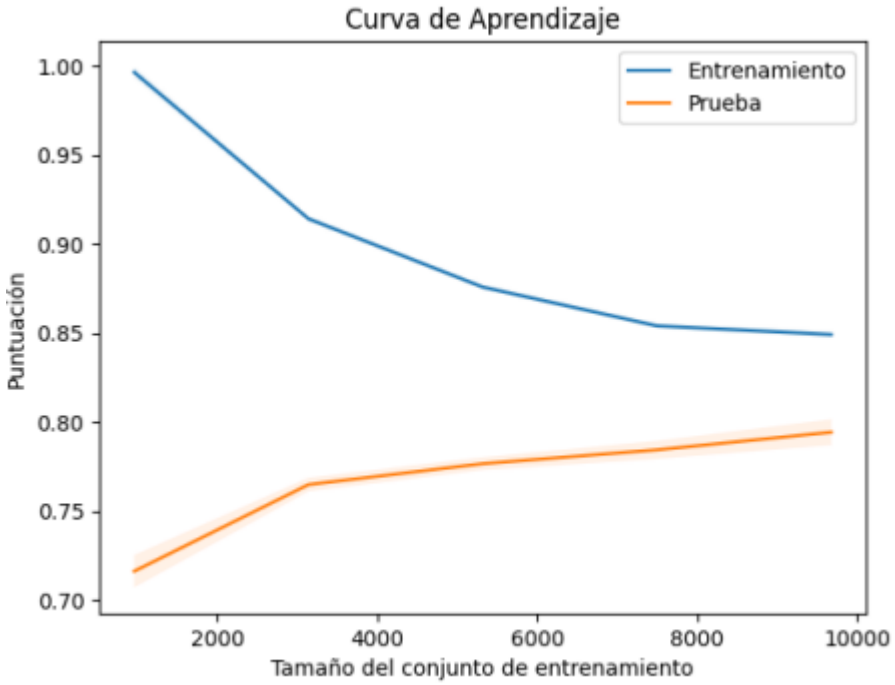


Gráfica 10. Primer árbol de decisión de Adaboost.

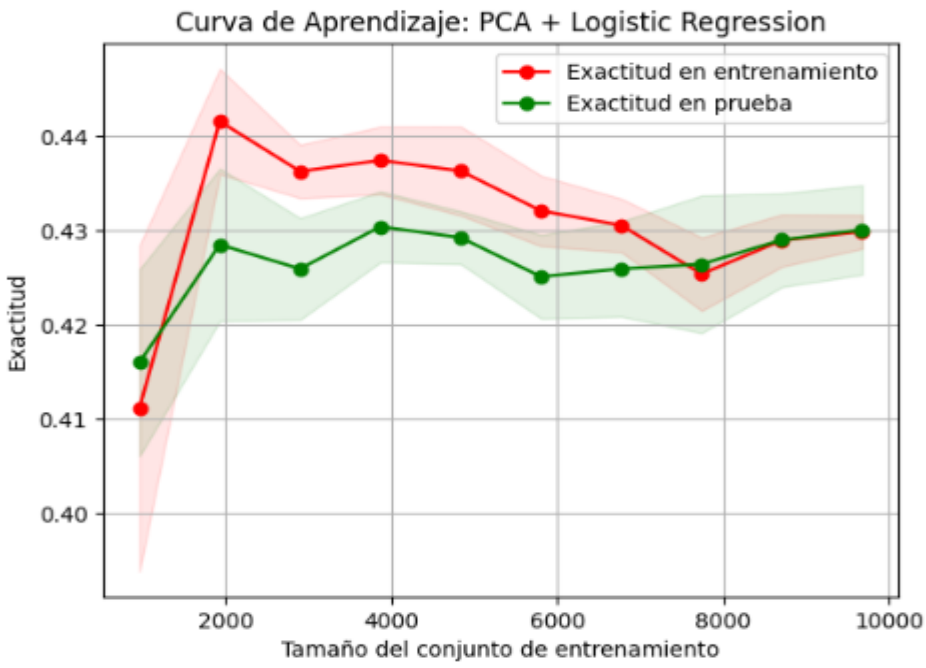
Hiperparámetros



Gráfica 11. Curva de aprendizaje para RandomForest.



Gráfica 12. Curva de aprendizaje para GradientBoosting.




```
No se detecta overfitting en el modelo.

Mejor valor de 'n_components': 3
Mejor valor de 'C': 10
Mejor valor de 'gamma': 0.001
Exactitud en el conjunto de prueba: 0.6058201058201058
```

Gráfica 13. Curva de aprendizaje para PCA + LogisticRegression.

En general, es difícil dar una opinión acertada sobre cualquier proyecto con tan poco tiempo de exploración. En el caso específico de la investigación sobre árboles con características especiales, el grupo recomienda tomar muchos más datos para conocer mejor estos tipos de árboles. La obtención de más datos reduce el sesgo y nos da una visión más acertada de lo que deberían ser estas características.

Al seleccionar los modelos, se pueden presentar similitudes que pueden indicar nada. Por lo tanto, es recomendable utilizar muchos más modelos para diferenciar mejor los datos. El tiempo es corto, pero se logró desarrollar una cantidad de modelos que nos aproximan un poco a la realidad. Sin embargo, consideramos que lo tratado en esta asignatura debería tener mucho más tiempo para experimentar y aportar a los proyectos.

El sesgo de los datos hace que algunos modelos presenten problemas en "aprender". Por ello, es necesario una mejor máquina que permita el correcto funcionamiento de los programas más pesados. Al ser un dataset tan pesado, en ocasiones el equipo responde lento. Por este motivo, también recomendamos realizar divisiones de los datos.

En base a las recomendaciones anteriores, el grupo considera que se pueden mejorar los resultados de la investigación sobre árboles con características especiales. Para ello, es necesario aumentar la cantidad de datos, utilizar más modelos, contar con una mejor máquina y dividir los datos.

Dificultades en el proyecto

A lo largo de la realización del proyecto de investigación sobre árboles con características especiales, el grupo se enfrentó a una serie de dificultades las cuales nos ayudaron a mejorar nuestra capacidad de análisis y decisión.

Una de las dificultades más importantes fue elegir los datos adecuados para el proyecto. Para ello, el grupo tuvo que considerar una serie de factores, como la calidad de los datos, la cantidad de datos disponibles y la relevancia de los datos para el objetivo del proyecto. Finalmente, el grupo decidió utilizar el dataset de la competencia Kaggle "Tree Species Classification".

Otro obstáculo que enfrentó el grupo fue la selección de los modelos de clasificación. Para ello, el grupo tuvo que evaluar una serie de modelos diferentes, teniendo en cuenta factores como la complejidad del modelo, el tiempo de entrenamiento y la precisión del modelo. Finalmente, el grupo decidió utilizar los modelos Random Forest y XGBoost.

El grupo también se enfrentó a la problemática de analizar los datos estadísticos del dataset. Al analizar las columnas "tipo de suelo #7" y "tipo de suelo #15", el grupo encontró que todos los valores de estas columnas eran iguales a 0. Esto podría deberse a que las mediciones de estas variables no se realizaron correctamente o a que simplemente se colocaron valores de 0 para evitar que los campos quedaran vacíos.

El grupo decidió trabajar con estos datos, a pesar de la incertidumbre que planteaban. Para ello, el grupo verificó que en cada tipo de corteza designada en el archivo "train" hay 2160 instancias para cada uno de los 7 tipos de cobertura forestal. Esto indica que los datos no fueron introducidos por error, sino que tienen algún significado.

En general, el grupo considera que las dificultades que enfrentó fueron positivas, ya que les ayudaron a aprender y mejorar sus habilidades. El grupo cree que los resultados del proyecto son prometedores y que pueden contribuir a la investigación sobre árboles con características especiales.

Conclusiones

- **Obtener más datos:** La escasez de datos es uno de los principales problemas que limitan la precisión de los modelos de clasificación. El grupo recomienda obtener más datos de árboles con características especiales, especialmente de aquellos que son más difíciles de identificar.
- **Conocer la razón de los valores nulos:** Los valores nulos en las columnas "tipo de suelo #7" y "tipo de suelo #15" pueden deberse a errores de medición o a la ausencia de información. El grupo recomienda investigar la razón de estos valores nulos para poder tomar decisiones informadas sobre cómo tratarlos.
- **Utilizar una variedad de modelos:** El grupo utilizó 9 modelos de clasificación diferentes, lo que le permitió evaluar la efectividad de cada uno en diferentes escenarios. El grupo recomienda utilizar una variedad de modelos para tener una mejor comprensión del problema y obtener mejores predicciones.
- **Interpretar los resultados de los modelos:** Los resultados de los modelos de clasificación pueden ser difíciles de interpretar. El grupo recomienda utilizar técnicas de visualización para obtener una mejor perspectiva visual de los resultados.

Bibliografía

* FOREST COVER TYPE PREDICTION – Use cartographic variables to classify forest categories | Kaggle. (2022). Retrieved 4 July 2022, from www.kaggle.com/competitions/forest-cover-typeprediction/overview/description