

ALGORITMO PARA MONITORIZAR Y DETECTAR LA ROYA EN LOS CULTIVOS DE CAFÉ

Juan Jose Madrigal Palacio

Universidad Eafit

Colombia

jjmadrigap@eafit.edu.co

Luis Ángel Jaimes Mora

Universidad Eafit

Colombia

lajaimesm@eafit.edu.co

Mauricio Toro

Universidad Eafit

Colombia

mtorobe@eafit.edu.co

RESUMEN

El café es una de las principales fuentes monetarias agrícolas del país tanto para exportación como para el consumo interno del mismo el cual permite el sostenimiento de miles de familias [8]; es por esta razón que el determinar en tiempo real la presencia o no de la roya se vuelve un factor determinante para estas familias, pero debido a la dificultad de determinar esto sin una examinación exhaustiva de un especialista el cual puede llegar a demorarse en tomar una decisión termina poniendo en peligro el cultivo.

Parecido a como sucede con las plantaciones de banano afectadas por el hongo fusarium el cual destruye plantaciones enteras de banano y deja el suelo inutilizable para plantar durante años [5], tenemos también a la roya la cual, aunque no deja inutilizable el suelo; sigue siendo difícil llegar a determinar a tiempo y con facilidad.

1. INTRODUCCIÓN

Debido a la importancia del café para el país, esta debe de ser cuidada y monitorizada para su correcto cultivo y crecimiento, con el fin de lograr un producto de calidad tanto para exportación como para consumo interno del país.

Con esto en cuenta es que se buscan formas más optimas de asegurar su crecimiento y salud manteniendo a raya enfermedades e infecciones de hongos como es el caso de la roya.

La roya es una enfermedad causada por el hongo *Hemileia vastatrix*; esta afecta a las plantas mediante la caída prematura de las hojas infectadas, reduciendo el rendimiento hasta un 50% o más.[6]

Los síntomas de la roya son: lesiones cloróticas, inicialmente con decoloración de la hoja, donde tiende a acumularse más agua, y posteriormente con gran presencia de urediniosporas del hongo que se reconoce como el polvillo amarillo o naranja ubicado por el envés de la hoja afectada.[6]

En este caso, vamos a llevar a cabo una demostración de nuestra ayuda hacia el problema de la detención de roya por medio del análisis de datos recolectados de las plantaciones.

2. PROBLEMA

El problema en el cual vamos a centrarnos es en el cómo lograr averiguar y concluir la presencia de la roya en una plantación de café por medio del análisis y estructuración de

los diferentes datos obtenidos de las condiciones de la zona del cultivo como son iluminación, pH, humedad, entre otros; para determinar por medio de un algoritmo lo más optimo y eficiente posible para así determinar la necesidad de una intervención y así evitar la propagación de la roya y la pérdida de un cultivo saludable.

3. TRABAJOS RELACIONADOS

3.1 C 4.5

C4.5 básicamente construye arboles de decisión a partir de un grupo de datos base o datos de ejemplo, usando la entropía de información, la cual básicamente es el nivel de incertidumbre, siendo 0 la incertidumbre nula y 1 la incertidumbre máxima.[3]

En cada nodo del árbol, C4.5 elige un atributo de los datos que más eficazmente dividen el conjunto de muestras en subconjuntos enriquecidos en una clase u otra. Su criterio es el normalizado para ganancia de información, que es la diferencia entre la entropía de un nodo y la de uno de sus descendientes, lo que conlleva a la división de datos a partir del atributo elegido. Este algoritmo resulta más eficiente que otros ya que una vez que ha sido creado el árbol intenta eliminar las ramas que no ayudan, reemplazándolos con los nodos de hoja.[4]

Además, es capaz de manejar tanto atributos discretos como atributos continuos, esto es muy bueno ya que en el proyecto se va a trabajar con estos dos tipos de atributos, la forma en que el algoritmo maneja los continuos es crear un umbral y luego se divide la lista en aquellos cuyo valor de atributo es superior al umbral y los que son menores o iguales a él.[4][7]

3.2 ID3 (Induction Decision Trees)

ID3 Crea un árbol de decisión como un método para aproximar una función objetivo de valores discretos, que es resistente al ruido en los datos y que es capaz de hallar o aprender de una disyunción de expresiones. Aquí ya se presenta una desventaja frente a otros algoritmos y es que trabaja solo con atributos o valores discretos, por lo cual para nuestro proyecto resultaría un poco más complicado, ya que tendríamos que pasar los atributos continuos a discretos. Aunque este algoritmo también implementa el uso de la entropía de información y de la ganancia de información, lo cual es bueno.[2]

Después de haber creado el árbol de decisión este se recorre desde la raíz y tanto en ella como en cada uno de

los demás nodos se decide cuál rama tomar basándonos en el valor de algún atributo del ejemplar que se esté clasificando, hasta llegar a un nodo terminal (hoja), que corresponde a la clase en que queda clasificado el ejemplar.[3]

3.3 Find-S

Find-S es reconocido en el sector de la IA, ya que se enfoca en hipótesis, a partir de un grupo de datos base o conjunto de ejemplos, el conjunto de ejemplos deberá estar conformado por una serie de tuplas de valores, denominados atributos. Pero uno de los atributos tiene que ser de tipo binario, el cual es el atributo objetivo a clasificar que diferencia el concepto, en nuestro caso será la etiqueta sobre si la planta contiene o no roya. De esta forma el algoritmo trata de obtener una hipótesis que sea la más específica a los atributos en común de todos los datos base. Una vez obtenida la hipótesis se puede determinar si una nueva instancia la cumple.

Find-S realiza esta labor tomando una tupla de valores con el mismo número de atributos, menos el del atributo objetivo, que los de entrenamiento. Pero de forma adicional define un nuevo tipo de valores que puede adoptar un atributo. \emptyset que representa ningún valor. ¿Este es el valor más específico posible, y? que representa cualquier valor. Este es el valor más general posible.[1]

Un ejemplo de Find-S sería el siguiente:

Cielo	Temperatura	Humedad	Viento	Agua	Previsión	Hacer Deporte
Soleado	Templada	Normal	Fuerte	Templada	Igual	Sí
Soleado	Templada	Alta	Fuerte	Templada	Igual	Sí
Lluvia	Fría	Alta	Fuerte	Templada	Cambio	No
Soleado	Templada	Alta	Fuerte	Fría	Cambio	Sí

la hipótesis inicial es: $\langle \emptyset, \emptyset, \emptyset, \emptyset, \emptyset, \emptyset \rangle$

Ante el primer ejemplo: $\langle \text{Soleado, Templada, Normal, Fuerte, Templada, Igual} \rangle$

Ante el segundo ejemplo: $\langle \text{Soleado, Templada, ?, Fuerte, Templada, Igual} \rangle$

Como el tercer ejemplo tiene su atributo objetivo a falso no se hace nada

Y Ante el último ejemplo: $\langle \text{Soleado, Templada, ?, Fuerte, ?, ?} \rangle$

Él cual construye la hipótesis final. Que puede leerse como que para hacer deporte debe cumplirse que el cielo este soleado, la temperatura templada y el viento fuerte. Todos los demás atributos no se tienen en cuenta.

Como se pudo observar al estar enfocado en hipótesis depende demasiado de la cantidad de datos base, por lo cual si esta es muy pequeña es bastante probable que la hipótesis falle. Pero si por el contrario cuenta con un conjunto de

ejemplos grande obtendría la hipótesis y lo único que haría es comparar las entradas sin el atributo binario y así determinar su atributo binario, esto resulta más efectivo que en otros algoritmos ya que en otros hay que realizar todo el algoritmo para cada una de las entradas.

3.4 CHAID

Para comenzar CHAID examina las tablas de tabulación cruzada entre los campos de entrada y los resultados, para luego comprobar la importancia mediante una comprobación de independencia, esto con el fin de tomar solo los atributos de mayor impacto en el atributo objetivo, CHAID es capaz de aceptar tanto los datos de entrada como el objetivo de manera discreta o continua, e incluso acepta variables de frecuencia o ponderaciones de casos. Además, al no generar arboles binarios, lo que trae arboles más grandes, pero a cambio es mucho más asertivo que otros algoritmos. También cabe resaltar que CHAID es mucho más visual que otros algoritmos lo que genera una explicación más didáctica.[9]

REFERENCIAS

1. Algoritmo find-s. 2018/06/03. Wikipedia. Recuperado de https://es.wikipedia.org/wiki/Algoritmo_find-s
2. Algoritmo ID3. 2018/11/21. Wikipedia. Recuperado de https://es.wikipedia.org/wiki/Algoritmo_ID3
3. Bruno López Takeyas. Algoritmo ID. S.f. Recuperado de <http://www.itnuevolaredo.edu.mx/takeyas/Apuntes/Inteligencia%20Artificial/Apuntes/IA/ID3.pdf>
4. C4.5. 2018/04/09. Wikipedia. Recuperado de <https://es.wikipedia.org/wiki/C4.5>
5. Catalina Oquendo. Un hongo que afecta al cultivo de banano dispara las alarmas en Colombia. 2019/07/20. El País. Recuperado de https://elpais.com/internacional/2019/07/19/colombia/1563570288_279873.html
6. CropLife Latin America. Roya del cafeta. S.f. Recuperado de <https://www.croplifela.org/es/plagas/listado-de-plagas/roya-del-cafeto>
7. Gilaberth. atributos discretos y continuos. 2009/04/14. SlideShare. Recuperado de <https://es.slideshare.net/gilaberth/atributos-discretos-y-continuos>
8. María Paula Martínez. La importancia del café en Colombia. 2017/11/08. Blogs El Insignia. Recuperado de <https://blog.elinsignia.com/2017/11/08/ea-la-importancia-del-cafe-en-colombia/>
9. Nodo C5.0. S.f. IBM. Recuperado de https://www.ibm.com/support/knowledgecenter/es/SS3RA7_sub/modeler_mainhelp_client_ddita/clementine/c50node_general.html