Revista Innovación y Software
Vol. 3, No. 1, Mes Marzo-Agosto, 2022
ISSN: 2708-0935
Pág. 58-66
https://revistas.ulasalle.edu.pe/innosoft

Tipo de artículo: Artículos originales
Temática: Inteligencia artificial
Recibido: 18/11/2021 | Aceptado: 22/12/2021 | Publicado: 30/03/2022

Identificadores persistentes:
ARK: ark:/42411/s8/a53
PURL: 42411/s8/a53

Uso de árboles de decisión para detectar si una habitación está ocupada usando Python

Use of decision trees to detect if a room is occupied using Python

Joel Atamari Aguilar [1], Christian Flores Conde [2], Jhon Mamani Mamani [3], Sergio Rondon Polanco [4*]

[1] Universidad Nacional de San Agustín. jatamaria@unsa.edu.pe
[2] Universidad Nacional de San Agustín. cfloresc@unsa.edu.pe
[3] Universidad Nacional de San Agustín. jmamanim@unsa.edu.pe
[4] Universidad Nacional de San Agustín. srondonp@unsa.edu.pe

[*] Autor para correspondencia: srondonp@unsa.edu.pe

Resumen
En este artículo se presenta una descripción de los árboles de decisión para determinar si una habitación está ocupada o no. En esta investigación se demuestra empíricamente que es posible determinar si una habitación está ocupada o no, usando las variables temperatura, humedad, luminosidad, nivel de CO2 y el radio de humedad, mediante la utilización de árboles de decisión con las librerías SKLEARN en el lenguaje Python.

Palabras clave: Inteligencia artificial, árboles de decisión, CO2, etiquetas, Python.

Abstract
This article presents a description of the decision trees for determining whether a room is occupied or not. In this research it is empirically demonstrated that it is possible to determine whether a room is occupied or not, using the variables temperature, humidity, luminosity, CO2 level and the humidity ratio, by using decision trees with the SKLEARN libraries in the language Python.

Keywords: *Artificial Intelligence, Decision trees, CO2, labels, Python.*

Introducción

Según Marvin Minsky [1], la inteligencia artificial (IA) es la ciencia de construir máquinas para que hagan cosas que, si las hicieran humanos, requerirían inteligencia.

La IA tiene diferentes técnicas como sistemas expertos basados en reglas, redes neuronales artificiales, árboles de decisión, etc. [2]. A continuación, se presenta la implementación de un árbol de decisión para determinar si una habitación está ocupada o no con un dataset de más de 20000 registros utilizando una biblioteca de software automático llamada Sklearn.

Árboles de decisión

Los árboles de decisión es una de las técnicas de aprendizaje inductivo supervisado no paramétrico, se utiliza para la predicción y se emplea en el campo de inteligencia artificial, donde a partir de una base de datos se construyen diagramas de construcción lógica, muy similares a los sistemas de predicción basados en reglas, que sirven para representar y categorizar una serie de condiciones que ocurren en forma repetitiva para la solución de un problema.[3]

Nuestra inclinación por el uso de esta técnica se debe a sus ventajas:

- Se puede trabajar con valores cuantitativos (solo tuvimos que transformarlos).
- Al contar un número pequeño de características (cinco) es factible el uso de esta técnica.

Limpieza de datos

Inicialmente nuestra base de conocimiento estaba conformada por 7 columnas (No, Fecha, Temperatura, Humedad, Luminosidad, Nivel de CO2, Radio de humedad y la salida). Además, esta contaba con 20560 casos.

Con el objetivo de realizar un mejor entrenamiento se realizó la limpieza de datos utilizando como herramienta principal Microsoft Excel 2016. Se eliminó la columna Fecha ya que esta no tiene algún valor significativo para el entrenamiento al ser diferente para cada caso. Se eliminó filas repetidas usando la herramienta antes ya mencionada, con lo que el número de filas se redujo a 19119. Finalmente, no se encontraron valores muy alejados al promedio.

Después de haber realizado la fase de limpieza de datos se continuó con la transformación de estos.

En esta parte necesitamos hacer un mapeo de los datos en donde transformamos estos datos de entrada en valores categóricos. Esto debido a que nuestros datos son cuantitativos y no cualitativos.

Revista Innovación y Software
Vol. 3, No. 1, Mes Marzo-Agosto, 2022
ISSN: 2708-0935
Pág. 58-66
https://revistas.ulasalle.edu.pe/innosoft

Se utilizó la Ecuación 1 para esta transformación, la cual resta el menor valor al mayor valor y lo divide entre la cantidad de categorías. Esto para posteriormente ir sumando el resultado al valor mínimo y definir los rangos de cada categoría.

$$Rango\ de\ cada\ categoría = (Valor\ maximo - Valor\ minimo)/Cantidad\ de\ categorías \quad (1)$$

- Temperatura : Donde se dividió en 5 categorías, la primera categoría donde los datos tienen una lectura menor igual a 20.0816666,en la segunda los datos tienen una lectura mayor a 20.0816666 y menor igual a 21.1633332, en la tercera categoría los datos tienen una lectura mayor a 21.1633332 y menor igual a 22.2449998, en la cuarta categoría los datos tienen una lectura mayor a 22.2449998 y menor igual a 23.3266664,finalizando en la quinta categoría donde datos tienen una lectura mayor a 23.3266664.

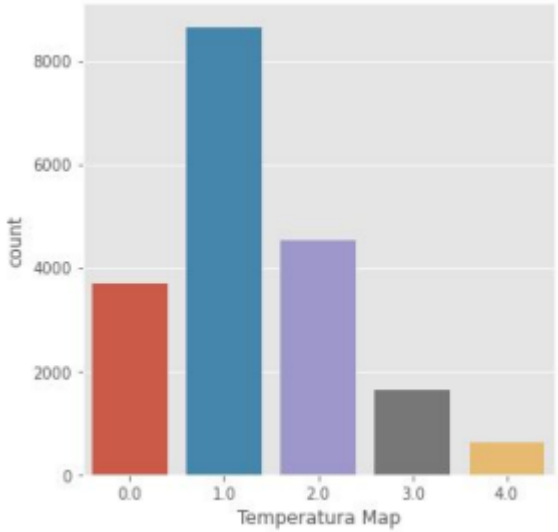

Figura 1. Cantidad de datos de Temperatura dividido en 5 categorías.

- Humedad: Donde se dividió en 5 categorías, la primera categoría donde los datos tienen una lectura menor igual a 21.296, en la segunda los datos tienen una lectura mayor a 21.296 y menor igual a 25.847, en la tercera categoría los datos tienen una lectura mayor a 25.847 y menor igual a 30.398, en la cuarta categoría los datos tienen una lectura mayor a 30.398 y menor igual a 34.949, finalizando en la quinta categoría donde datos tienen una lectura mayor a 34.949.

Revista Innovación y Software
Vol. 3, No. 1, Mes Marzo-Agosto, 2022
ISSN: 2708-0935
Pág. 58-66
https://revistas.ulasalle.edu.pe/innosoft

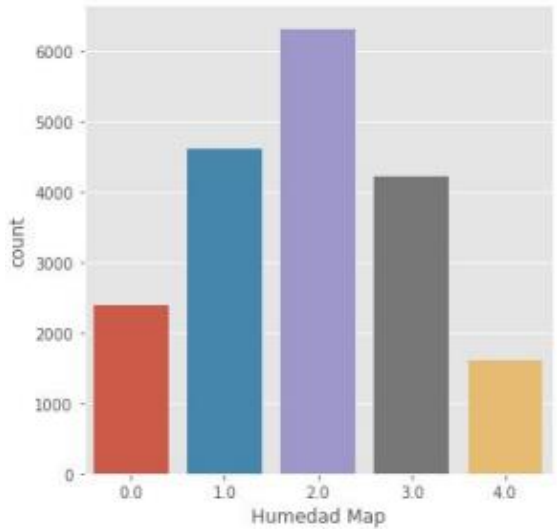

Figura 2. Cantidad de datos de Humedad dividido en 5 categorías.

- Luminosidad: Donde se dividió en 5 categorías, la primera categoría donde los datos tienen una lectura menor igual a 339.45, en la segunda los datos tienen una lectura mayor a 339.45 y menor igual a 678.9, en la tercera categoría los datos tienen una lectura mayor a 678.9 y menor igual a 1018.35, en la cuarta categoría los datos tienen una lectura mayor a 1018.35 y menor igual a 1357.8, finalizando en la quinta categoría donde datos tienen una lectura mayor a 1357.8.

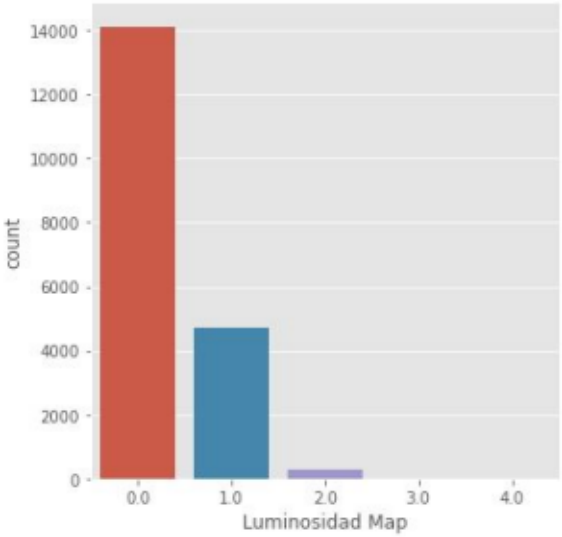

Figura 3. Cantidad de datos de Luminosidad dividido en 5 categorías.

Revista Innovación y Software
Vol. 3, No. 1, Mes Marzo-Agosto, 2022
ISSN: 2708-0935
Pág. 58-66
https://revistas.ulasalle.edu.pe/innosoft

Nivel de CO2: Donde se dividió en 5 categorías, la primera categoría donde los datos tienen una lectura menor igual a 745.5, en la segunda los datos tienen una lectura mayor a 745.5 y menor igual a 1078.25, en la tercera categoría los datos tienen una lectura mayor a 1078.25 y menor igual a 1411, en la cuarta categoría los datos tienen una lectura mayor a 1411 y menor igual a 1743.75, finalizando en la quinta categoría donde datos tienen una lectura mayor a 1743.75.

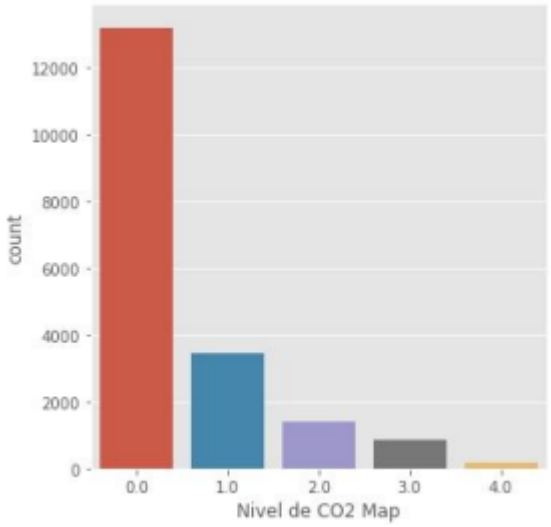

Figura 4. Cantidad de datos de CO2 dividido en 5 categorías

- Radio de Humedad :Donde se dividió en 5 categorías, la primera categoría donde los datos tienen una lectura menor igual a 0.0034344, en la segunda los datos tienen una lectura mayor a 0.0034344 y menor igual a 0.0041948, en la tercera categoría los datos tienen una lectura mayor a 0.0041948 y menor igual a 0.0049552, en la cuarta categoría los datos tienen una lectura mayor a 0.0049552 y menor igual a 0.0057156, finalizando en la quinta categoría donde datos tienen una lectura mayor a 0.0057156.

Revista Innovación y Software
Vol. 3, No. 1, Mes Marzo-Agosto, 2022
ISSN: 2708-0935
Pág. 58-66
https://revistas.ulasalle.edu.pe/innosoft

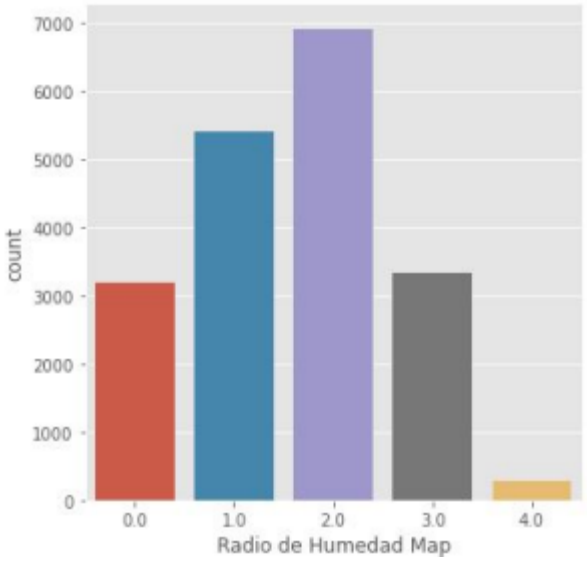

Figura 5. Cantidad de datos de Radio de Humedad dividido en 5 categorías

- Habitación Ocupada: Habiendo en estas 2 categorías ,1 para los datos que tienen el valor “si” y 0 para los que tienen el valor “no”

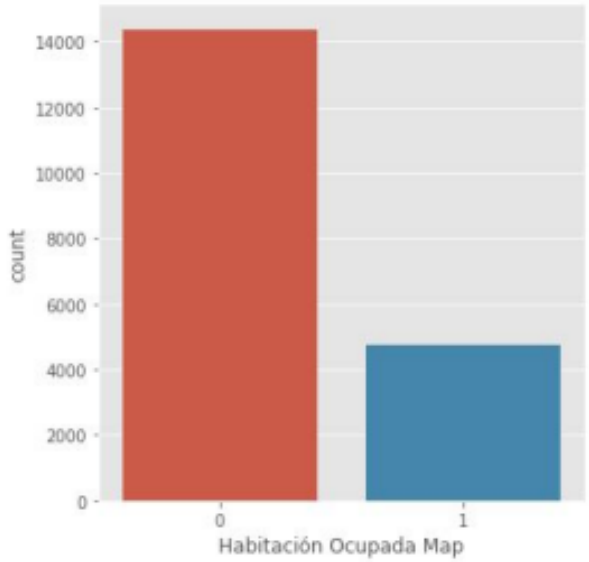

Figura 6. Cantidad de datos de Habitación Ocupada dividido en 2 categorías

Revista Innovación y Software
Vol. 3, No. 1, Mes Marzo-Agosto, 2022
ISSN: 2708-0935
Pág. 58-66
https://revistas.ulasalle.edu.pe/innosoft

Una categorizado todos los datos empezamos con la fase de entrenamiento.

Fase de entrenamiento

Después de haber realizado el mapeo de los datos, se realizó la fase de entrenamiento en el lenguaje Python utilizando Jupyter. Las librerías utilizadas son Numpy, Pandas y Sklearn.

Para la creación del árbol de decisión se utilizaron los siguientes parámetros citerion=”gini”, min_samples_split=100, min_samples_leaf=20 y class weight={1:3:22}.

Respecto a los datos usados para el entrenamiento, se usó el 75% para este y el 25% restante para la fase de comprobación Se desarrolló la siguiente interfaz para el ingreso de datos:

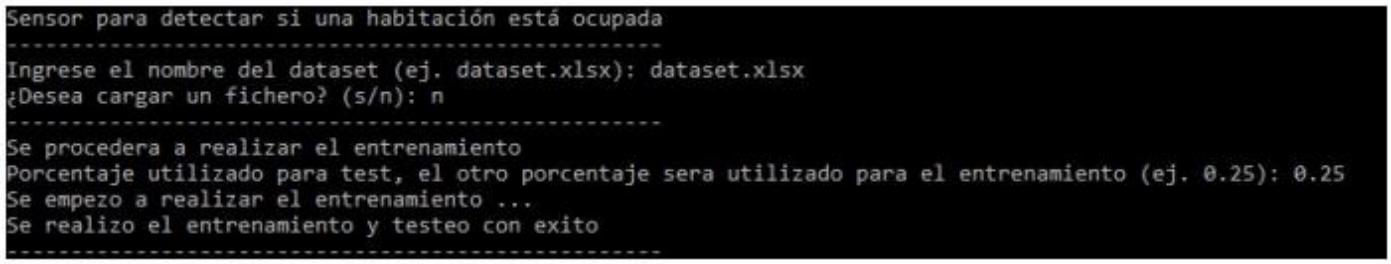

Figura 7. Interfaz para entrenamiento

Fase de comprobación

Para la comprobación se trabajó con el 25% de información restante. En esta fase se comprobó la salida esperada con la salida obtenida con lo que cada vez que no coincidan el error acumulado aumenta. Se muestra a continuación la interfaz para la fase comprobación.

Revista Innovación y Software
Vol. 3, No. 1, Mes Marzo-Agosto, 2022
ISSN: 2708-0935
Pág. 58-66
https://revistas.ulasalle.edu.pe/innosoft

```
--------------------------------------------------
Resultados
Numero de casos de prueba: 5140
Numero de errores: 55
Porcentaje de error: 1.07 %
¿Desea guardar un fichero con el modelo entrenado? (s/n): s
--------------------------------------------------
Se procedera a guardar el fichero
Ingrese el nombre del fichero (sin extencion): train
Se guardo el fichero con el nombre train
¿Desea salir (s/n): s
Vuelva pronto
```

Figura 8. Interfaz para comprobación

Análisis de resultados

Como se observa en la Figura 8 el porcentaje de error después de la fase de entrenamiento es de 1.07%. A continuación, se muestra una comparación entre los datos esperados y los obtenidos gráficamente. La figura 9 muestra una comparación solo entre los 160 primeros casos para una mejor visualización. En una comparación de los 19119 casos se repetirá.

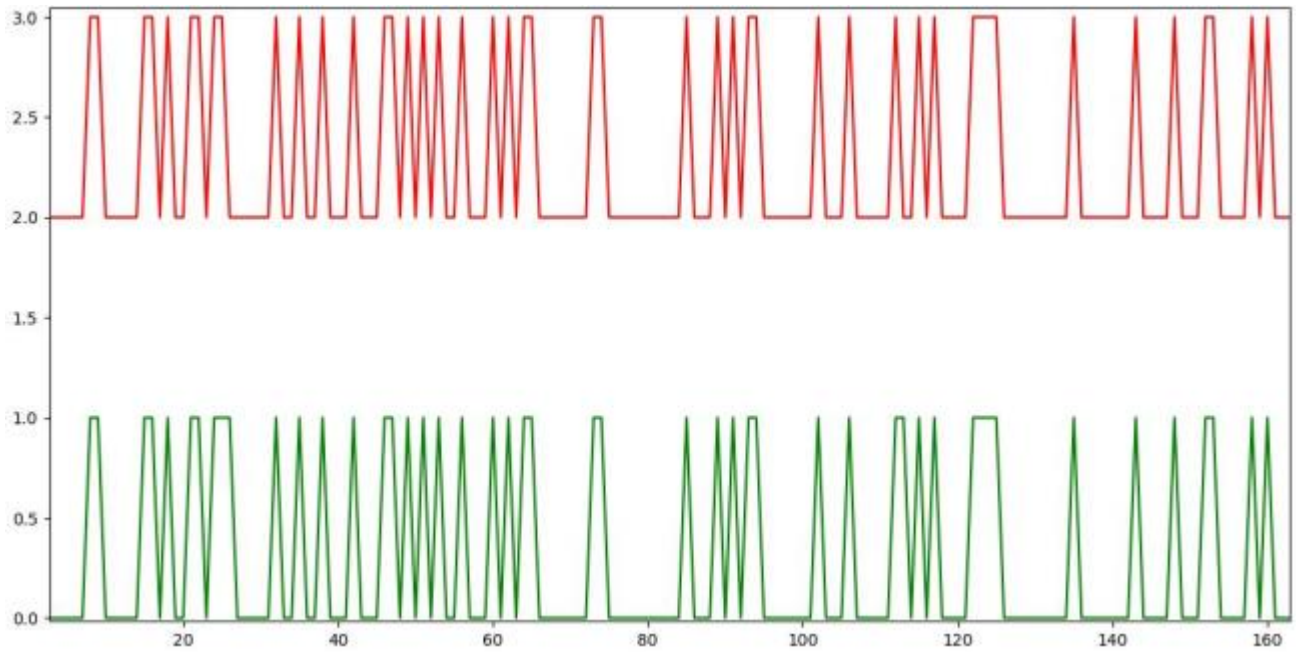

Figura 9. Gráfico comparativo de los resultados esperados (verde) y los datos obtenidos (rojo).

Como trabajo próximo se sugiere la comparación de diversas técnicas de aprendizaje para el mismo contexto.

Conclusiones

- El uso de árboles de decisión como técnica de entrenamiento para determinar si una habitación está ocupada tuvo 1% de error. Por lo que al usar este algoritmo obtuvimos buenos resultados.
- Los árboles de decisión son una técnica muy fácil de entender y aplicar a diversos contextos, ya que para este caso, aunque no tuviésemos valores cualitativos pudimos utilizarla sin problemas gracias a una adecuada transformación de datos.
- El uso de Python y sus librerías facilitan la aplicación de muchas técnicas de inteligencia artificial.

Referencias

[1] M. Minsky, «The age of Intelligent Machines: Thoughts About Artificial Intelligence,» KurzweilAl.net., 1990.

[2] J. De Andres Oviedo, «Tecnicas de inteligencia artificial aplicadas al analsis de la solvencia empresarial,» Universidad de Oviedo, Oviedo, 2000.

[3] G. R. Solarte Martinez y J. Soto Mejia, «Árboles de decisiones en el diagnóstico de enfermedades cardiovasculares,» Scientia et Technica, vol. XVI, nº 49, pp. 104-109, 2011.