

Un juego de lógica, El Buscaminas

Lucía del Carmen Fuentes
dpto. Ciencias de la Computación e Inteligencia Artificial
Universidad de Sevilla
Sevilla, España
lucdelfue@alum.us.es
luciadeldcarmenfuentes@gmail.com

Fernando Manuel Ruiz Pliego
dpto. Ciencias de la Computación e Inteligencia Artificial
Universidad de Sevilla
Sevilla, España
ferruipli@alum.us.es
fdomruizp@gmail.com

El objetivo principal del trabajo es implementar un método que determine las casillas con mayor probabilidad de no contener una mina y sugerirlas como siguiente paso a dar en el juego. Las conclusiones obtenidas del mismo corresponde a una buena capacidad de aprendizaje que tiene la red al analizar las variables en el entorno especificado y sugerir siguientes pasos como mejor elección.

I. INTRODUCCIÓN

Buscaminas [1] (Minesweeper) es un videojuego inventado por Robert Donner en 1989 y que se hizo popular por venir instalado en todas las versiones de Windows desde la famosa 3.1. Es un juego de lógica donde tendremos que despejar todas las casillas de la pantalla sin denotar ninguna mina.

Para saber dónde están las minas tendremos como pista un número que saldrá al destapar una casilla, este número nos indica la cantidad de minas que hay en las casillas colindantes (arriba, abajo, izquierda, derecha y en diagonal).

La estrategia de este juego tiene relación con la probabilidad, ya que se entiende encontrar una estrategia que nos asegure la mejor probabilidad de resolver una partida. A raíz de esto, muchas personas han intentado usar la Inteligencia Artificial para crear partidas y que se auto-resuelvan, es decir, intentar que el ser humano le gane a la máquina.

Una de las problemáticas de dicho asunto es intentar resolver partidas con un tamaño de tablero considerable ya que cuanto más grande es, más difícil es resolverlo y con total probabilidad dan lugar a malos resultados.

En relación con esto, tras estudiar las redes bayesianas [2] en el campo de la Inteligencia Artificial se plantea la resolución de una partida a través de algoritmos de inferencia probabilística, en este caso, se usa eliminación de variables. Es importante destacar que al igual que hemos contado con anterioridad, al proponer resolver tableros de gran tamaño, la red bayesiana puede verse alterada ya que usa nodos interconectados, lo que se refleja en un aumento de carga de trabajo que puede afectar a los resultados obtenidos.

Por tanto, se propone una simplificación del modelo para corregir problemas de ineficiencia en los resultados.



Fig 1. Imagen del juego del buscaminas

II. PRELIMINARES

A. Métodos empleados

Redes Bayesianas [1]

Las redes bayesianas están formadas por un conjunto de variables aleatorias, estas variables expresan nuestra intuición al representar nuestro “mundo” mediante la red bayesiana correspondiente.

En nuestro trabajo existen dos tipos de variables aleatorias:

- $X_{\text{fila, columna}}$: la cual representa la presencia de una mina en una fila y columna (casilla del tablero). Esta variable es de tipo booleano, donde el valor **True** indica la presencia de la mina en la casilla y el valor **False** indica que la casilla está libre, sus valores se distribuyen de la misma forma y la probabilidad de que tome el valor **True** dependerá del número de minas escondidas y del número de casillas del tablero.
- $Y_{\text{fila, columna}}$: asociada a la cantidad de minas que hay en las casillas colindantes a una casilla. Esta variable es de tipo discreto y toma valores de 0 hasta 8. Si la casilla se encuentra en una esquina del tablero, solo puede tomar valores hasta 3 y si la casilla se encuentra en uno de los lados, solo puede tomar valores hasta 5.

La red bayesiana en un grafo acíclico dirigido es una modelo grafo probabilístico que representa un conjunto de variables aleatorias y sus dependencias condicionales.

Dicha red consta de un conjunto de nodos, uno por cada variable del mundo y un conjunto de arcos dirigidos o aristas que conectan dichos nodos. Cada nodo contiene su distribución de probabilidad condicional.

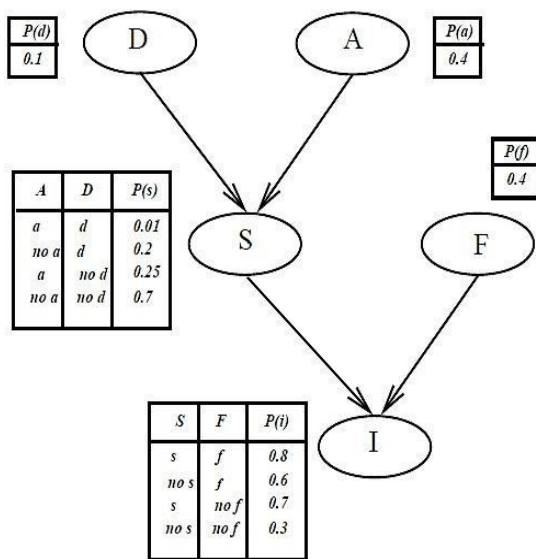


Fig 2. Ejemplo de red bayesiana con sus nodos y su distribución de probabilidad condicional

Inferencia probabilística [2]

Se entiende el cálculo de la probabilidad de una proposición dada condicionada por la observación de determinadas evidencias. Es decir, cálculos del tipo $P(a|b)$ donde a es la proposición que se consulta y b es la proposición que se ha

marcado. El conocimiento base vendrá dado por una DCC, distribución de probabilidad conjunta y completa (probabilidad de cada evento atómico).

El problema de la inferencia en una red bayesiana:

- Calcular la probabilidad a posteriori para un conjunto de variables de consulta, dado que se han observado algunos valores para las variables de evidencia.

Debido a las dimensiones que pueden alcanzar los algoritmos, se ve necesarios evitar cálculos redundantes. Para ellos utilizaremos el algoritmo de eliminación de variables cuya labor es detectar y eliminar las variables que no son determinantes en el modelo para reducir los costes de cálculo. Además, realiza operaciones entre tablas, multiplica o suma lo cual reduce aún más dichos cálculos.

B. Trabajo Relacionado

A lo largo de la historia reciente, diversos científicos han intentado simular la auto-resolución del juego.

Dichos científicos se centran en la creación de una red bayesiana como en la inferencia estadística. [3].

III. METODOLOGÍA

Como hemos dicho con anterioridad, el objetivo es, dado un estado del juego (se define estado como una instancia, un determinado momento de la partida) sugerir una casilla a destapar cuya probabilidad de no contener una mina sea alta.

Para llevar a cabo este procedimiento, se usará los métodos dichos anteriormente:

- Nodos:
 - $\text{Tapada}_{\text{fila, columna}}$
 - $\text{Destapada}_{\text{fila, columna}}$
- Aristas:
 - $\text{padres}(\text{Destapada}_{\text{fila, columna}})$:
 $\{\text{ListaColindantesOcultos}(\text{Destapada}_{\text{fila, columna}})\}$

ListaColindantesOcultos

Entrada:

- Nodo $\text{Destapada}_{\text{fila, columna}}$ (donde fila es el orden de cuadrantes en sentido horizontal y columna en vertical).

Salidas:

- Conjunto de casillas colindantes a la casilla $\text{Destapada}_{\text{fila, columna}}$ distanciadas en una unidad.

- Distribución de la probabilidad condicional:

- Para Tapada fila, columna :

$$P(Tapada_{fila, columna}) = \begin{cases} 1 & \text{si } Tapada_{fila, columna} = Mina \\ \frac{Minas - Minas_Encontradas}{CasillasOcultasTotales} & \text{e.o.c} \end{cases}$$

$$P(\neg Tapada_{fila, columna}) = \begin{cases} 0 & \text{si } Tapada_{fila, columna} = Mina \\ 1 - (P(Tapada_{fila, columna})) & \text{e.o.c} \end{cases}$$

- Para Destapada fila, columna :

Destapada fila, columna	\sum Tapada fila, columna vecinos	
$\forall i \in [0-8]$	$i == \sum x_{ij}$	1
	e.o.c	0

Se comienza con la definición de un tablero que con tamaño $n \times m$, nótese que la n es el número de filas y m el número de columnas.

Partiendo del conocimiento de que cada fila y columna del tablero va a representar un nodo de Tapada fila, columna y otro de Destapada fila, columna los cuales están conectados unos con otros tenemos en consecuencia una gran complejidad a la hora de resolver este problema ya que la cantidad de trabajo es lo suficientemente grande como para no alcanzar un tiempo razonable al calcular la solución.

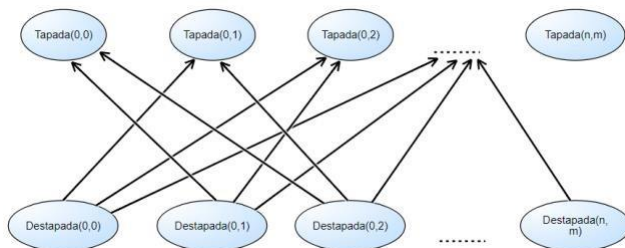


Fig 3. Red Bayesiana sin simplificar

Debido a esto, se ha pensado en reducir el tamaño de la red bayesiana al reducir a la mitad en cuanto al número de nodos conectados por ambas variables.

La división de nodos se consigue partiendo de los siguientes puntos:

- Se parte de los $n \times m$ nodos totales, los cuales se dividirán en nodos formados por casillas tapadas del tablero y las destapadas.
- Por cada destapado de casilla se genera una nueva red bayesiana la cual agrega dicha casilla a la lista de casillas destapadas y la correspondiente actualización del resto de variables de la red.

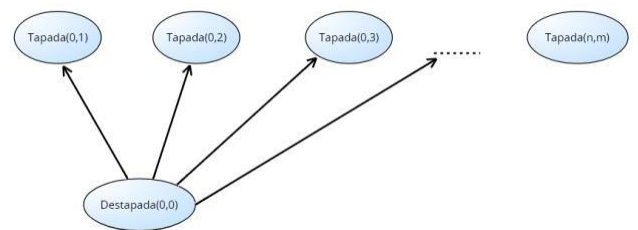


Fig 4. Red Bayesiana simplificada

Con respecto a la automatización del proceso para resolver el juego, cabe destacar que:

- Al destapar una casilla Tapada fila, columna si las casillas colindantes de la misma tienen probabilidad nula de ser minas, se destapan.
- Según varias consultas a diversas páginas webs, se concluye que el juego en la vida real está implementado de manera que la primera casilla que destapa un jugador no puede ser mina, por tanto, esta consideración se ha tenido en cuenta en la solución propuesta.

Al realizarse este procedimiento se simplifica notoriamente el grafo por lo que el algoritmo es capaz de mejorar sus prestaciones.

Por cada consulta a la red, se muestran un listado de casillas ocultas vecinas, un listado de minas encontradas hasta el momento; la cual al principio está vacía ya que la red todavía no es capaz de averiguar con las evidencias que tiene cuáles de ellas son mina, esta lista empieza a dejar de estar vacía conforme avanza el juego ya que es cuando al quedar menos casillas por descubrir es capaz de averiguar cuáles son mina consultando las probabilidades. Por último, devuelve el conjunto de probabilidades de las casillas colindantes y la casilla que tiene probabilidad mayor de no ser mina.

Mejor casilla oculta con alta probabilidad de no ser mina:

```
x: Tapada(00) -> Probabilidad de no ser mina: 0.9583333333333334
x: Tapada(01) -> Probabilidad de no ser mina: 0.6
x: Tapada(03) -> Probabilidad de no ser mina: 0.6
x: Tapada(04) -> Probabilidad de no ser mina: 0.9583333333333334
x: Tapada(10) -> Probabilidad de no ser mina: 0.9583333333333334
x: Tapada(11) -> Probabilidad de no ser mina: 0.6
x: Tapada(12) -> Probabilidad de no ser mina: 0.6
x: Tapada(13) -> Probabilidad de no ser mina: 0.6
x: Tapada(14) -> Probabilidad de no ser mina: 0.9583333333333334
x: Tapada(20) -> Probabilidad de no ser mina: 0.9583333333333334
x: Tapada(21) -> Probabilidad de no ser mina: 0.9583333333333334
x: Tapada(22) -> Probabilidad de no ser mina: 0.9583333333333334
x: Tapada(23) -> Probabilidad de no ser mina: 0.9583333333333334
x: Tapada(24) -> Probabilidad de no ser mina: 0.9583333333333334
x: Tapada(30) -> Probabilidad de no ser mina: 0.9583333333333334
x: Tapada(31) -> Probabilidad de no ser mina: 0.9583333333333334
```

Fig 5. Imagen salida por pantalla de listas de las casillas ocultas hasta el momento junto con la probabilidad de que no sean minas

Tapada(00)_0	0
Tapada(00)_1	1

Tapada(01)_0	0.916667
Tapada(01)_1	0.0833333

Fig 7. Ejemplo de tabla CPD mostrada

Se observa que cada casilla posee dos valores, el primero de ellos, con la extensión ‘_0’ determina la probabilidad de que la casilla no contenga una mina. En el otro caso, determina la probabilidad de que exista mina en la casilla.

Lista de mejores casillas ocultas con probabilidad de no ser minas:

```
['Tapada(00)', 'Tapada(04)', 'Tapada(10)', 'Tapada(14)', 'Tapada(20)', 'Tapada(21)', 'Tapada(22)', 'Tapada(23)', 'Tapada(24)', 'Tapada(30)', 'Tapada(31)', 'Tapada(32)', 'Tapada(33)', 'Tapada(34)', 'Tapada(40)', 'Tapada(41)', 'Tapada(42)', 'Tapada(43)', 'Tapada(44)']
```

Fig 6. Imagen salida de sugerencias de casillas a destapar como siguiente paso.

Toda la implementación de la red y sus métodos externos que nos permiten un buen funcionamiento de la misma se ha conseguido usando el lenguaje de programación Python, dentro del mismo se ha usado el paquete pgmpy [4] el cual proporciona un marco de trabajo para las redes bayesianas.

Dicha herramienta contiene diversas funcionalidades como modelar los gráficos de probabilidad, las tablas CPD y obtener la inferencia probabilística exacta.

Respecto a las tablas CPD mostradas al ejecutar una consulta, informan de la probabilidad de que una mina esté contenida en la casilla en cuestión.

Existió la posibilidad de hacer el sistema de una forma alternativa a la propuesta como solución. Estas alternativas fueron desde la subdivisión de la red en diversas subredes formadas solo por nodos colindantes a la casilla consultada en cuestión, así como una segunda alternativa consistente en la adicción a la alternativa anterior de los vecinos colindantes a los propios colindantes de la casilla consultada.

Ambas no llegaron a tomarse como posibles procedimientos a hacer ya que surgían problemas con las probabilidades y no mejoran el rendimiento conseguido con la solución propuesta en el trabajo.

IV. RESULTADOS

Con respecto a los resultados cabe destacar que se ha usado la librería time [5]. Esta librería nos permite consultar el tiempo de ejecución que ha tardado la red bayesiana en crearse.

Para ser exactos en cuanto al tiempo de ejecución se ha creado una variable al inicio de la red y otra al final, restando el tiempo final menos el inicial sale el tiempo de ejecución que nos sirve para estudiar la eficiencia del código implementado.

A continuación, realizaremos pruebas para distintos tableros. Para cada uno de ellos, la implementación prueba varios intentos. Se mostrará la información de las pruebas para un intento aleatorio para cada tablero.

En concreto, para cada uno de los tableros se mostrará un ejemplo del destapado de una casilla. A través del nuevo tablero oculto el algoritmo sigue destapando la casilla con mayor probabilidad de no ser mina.

Finalmente se mostrará el resultado de la partida.

Los resultados obtenidos de las distintas pruebas que se ha sometido la red son los siguientes:

- Tamaño: 5 x 5 con 5 minas

Tablero Destapado	Tablero oculto
1 1 1 0 0	1 T T T T
1 M 2 2 1	T T T T T
1 3 M 3 M	T T T T T
0 2 M 4 2	T T T T T
0 1 1 2 M	T T T T T

```
Lista de casillas vecinas ocultas:
['Tapada01', 'Tapada02', 'Tapada10', 'Tapada11']

Lista de casillas destapadas:
['Destapada00']

Lista de minas encontradas:
[]

Oculta con mayor probabilidad de no ser mina
x: Tapada01 -> Probabilidad de no ser mina: 0.6666666666666666
x: Tapada02 -> Probabilidad de no ser mina: 0.7916666666666666
x: Tapada10 -> Probabilidad de no ser mina: 0.6666666666666666
x: Tapada11 -> Probabilidad de no ser mina: 0.6666666666666666

Lista de mejores ocultas con alta probabilidad de no ser minas: ['Tapada02']
```

```
Destapando casilla: ['Tapada02']
Nuevo tablero oculto
1 T 1 T T
T T T T T
T T T T T
T T T T T
T T T T T
```

```
Has ganado
Terminado en 5.0 segundos
```

- Tamaño: 5 x 5 con 6 minas

Tablero Destapado	Tablero oculto
M M M 2 1	T T T 2 T
2 3 3 M 1	T T T T T
1 1 1 1 1	T T T T T
M 2 1 1 0	T T T T T
1 2 M 1 0	T T T T T

```
Lista de casillas ocultas:
['Tapada00', 'Tapada02', 'Tapada04', 'Tapada12', 'Tapada13', 'Tapada14']

Lista de casillas destapadas:
['Destapada03']

Lista de minas encontradas:
[]

Oculta con mayor probabilidad de no ser mina
x: Tapada00 -> Probabilidad de no ser mina: 0.75
x: Tapada02 -> Probabilidad de no ser mina: 0.6
x: Tapada04 -> Probabilidad de no ser mina: 0.6
x: Tapada12 -> Probabilidad de no ser mina: 0.6
x: Tapada13 -> Probabilidad de no ser mina: 0.6
x: Tapada14 -> Probabilidad de no ser mina: 0.6

Lista de mejores ocultas con alta probabilidad de no ser minas: ['Tapada00']
```

```
Tiempo de paso: 0.0 segundos

Destapando casilla: ['Tapada00']
¡Has encontrado una mina! Has perdido.

Ultimo tablero resuelto
T T T 2 T
T T T T T
T T T T T
T T T T T
T T T T T

Has perdido!
Terminado en 0.0 segundos
```

- Tamaño: 5 x 5 con 7 minas

Tablero Destapado	Tablero oculto
M 1 0 1 1	T 1 T T T
1 2 1 2 M	T T T T T
2 4 M 4 2	T T T T T
M M M 3 M	T T T T T
2 3 2 2 1	T T T T T

```
Lista de casillas ocultas:
['Tapada00', 'Tapada02', 'Tapada03', 'Tapada10', 'Tapada11', 'Tapada12']

Lista de casillas destapadas:
['Destapada01']

Lista de minas encontradas:
[]

Oculta con mayor probabilidad de no ser mina
x: Tapada00 -> Probabilidad de no ser mina: 0.7999999999999999
x: Tapada02 -> Probabilidad de no ser mina: 0.7999999999999999
x: Tapada03 -> Probabilidad de no ser mina: 0.7083333333333333
x: Tapada10 -> Probabilidad de no ser mina: 0.7999999999999999
x: Tapada11 -> Probabilidad de no ser mina: 0.7999999999999999
x: Tapada12 -> Probabilidad de no ser mina: 0.7999999999999999

Lista de mejores ocultas con alta probabilidad de no ser minas: ['Tapada00']
```



```
Tiempo de paso: 0.0 segundos
Destapando casilla: ['Tapada00']
¡Has encontrado una mina! Has perdido.

Ultimo tablero resuelto
T 1 T T T
T T T T T
T T T T T
T T T T T
T T T T T
```

- [illegible]

[illegible]

- | Tablero Destapado | Tablero oculto |
|-------------------|-----------------|
| 1 1 1 M M 2 0 0 | 1 T T T T T T T |
| M 1 2 4 M 2 0 0 | T T T T T T T |
| 1 1 1 M 3 3 1 1 | T T T T T T T |
| 0 0 1 2 M 3 M 3 | T T T T T T T |
| 0 1 2 3 2 3 M M | T T T T T T T |
| 0 1 M M 2 2 3 2 | T T T T T T T |
| 1 2 2 3 3 M 1 0 | T T T T T T T |
| M 1 0 1 M 2 1 0 | T T T T T T T |

```
Tiempo de paso: 0.0 segundos
Destapando casilla: ['Tapada02']

Nuevo tablero oculto
1 T T T T T T T
T T T T T T T
T T T T T T T
T T T T T T T
T T T T T T T
T T T T T T T
T T T T T T T
T T T T T T T
```

```
Ultimo tablero resuelto
1 T 1 T T T T T
T T T T T T T T
T T T T T T T T
T T T T T T T T
T T T T T T T T
T T T T T T T T
T T T T T T T T
T T T T T T T T
```

- | Tablero Destapado | Tablero oculto |
|-------------------|-----------------|
| 2 M 2 0 1 M 3 2 | 2 T T T T T T T |
| 2 M 2 0 1 3 M M | T T T T T T T |
| 2 2 1 1 2 4 M 4 | T T T T T T T |
| M 1 0 1 M M 4 M | T T T T T T T |
| 2 2 1 1 3 M 4 2 | T T T T T T T |
| 1 M 1 0 1 1 2 M | T T T T T T T |
| 2 2 2 0 0 1 2 2 | T T T T T T T |
| 1 M 1 0 0 1 M 1 | T T T T T T T |

```
Tiempo de paso: 0.0 segundos
Destapando casilla: ['Tapada02']
Nuevo tablero oculto
2 T 2 T T T T
T T T T T T T
T T T T T T T
T T T T T T T
T T T T T T T
T T T T T T T
T T T T T T T
T T T T T T T
```

```
Tiempo de paso: 0.0 segundos
Destapando casilla: ['Tapada96']
Nuevo tablero oculto
1 1 3 T 2 0 0 0 1 T
1 T 3 T 2 0 1 1 2 1
1 1 3 2 2 0 1 T 2 1
0 1 2 T 1 0 1 1 3 T
1 2 T 4 3 2 1 1 2 T
T 2 2 T T 2 T 2 2 2
1 1 1 2 3 3 3 3 T 1
0 0 0 0 2 T 3 T 3 2
0 0 0 0 3 T 5 3 T 1
0 0 0 0 2 T 3 T 2 1

Has ganado
Terminado en 19.0 segundos
```

- | Tablero | Destapado | | Tablero oculto |
|---------------------|-----------|--|-----------------------|
| 1 1 3 M 2 0 0 0 1 M | | | 1 T T T T T T T T T T |
| 1 M 3 M 2 0 1 1 2 1 | | | T T T T T T T T T T |
| 1 1 3 2 2 0 1 M 2 1 | | | T T T T T T T T T T |
| 0 1 2 M 1 0 1 1 3 M | | | T T T T T T T T T T |
| 1 2 M 4 3 2 1 1 2 M | | | T T T T T T T T T T |
| M 2 2 M M 2 M 2 2 2 | | | T T T T T T T T T T |
| 1 1 1 2 3 3 3 3 M 1 | | | T T T T T T T T T T |
| 0 0 0 0 2 M 3 M 3 2 | | | T T T T T T T T T T |
| 0 0 0 0 3 M 5 3 M 1 | | | T T T T T T T T T T |
| 0 0 0 0 2 M 3 M 2 1 | | | T T T T T T T T T T |

- Tamaño: 10 x 10 con 22 minas

Tablero Destapado									
M	M	2	2	2	M	M	1	0	0
3	4	M	2	M	3	2	1	0	0
M	2	1	2	1	1	0	0	1	1
2	2	1	2	2	1	0	1	2	M
M	1	2	M	M	1	0	1	M	2
1	1	2	M	3	1	0	1	1	1
0	0	2	2	2	1	1	1	0	0
0	1	2	M	1	2	M	4	2	1
0	2	M	3	1	2	M	M	M	2
0	2	M	2	0	1	3	M	4	M

- [illegible]

```

Lista de casillas ocultas:
['Tapada00', 'Tapada01', 'Tapada03', 'Tapada11', 'Tapada12', 'Tapada13']
Lista de casillas destapadas:
['Destapada02']
Lista de minas encontradas:
[]
Oculta con mayor probabilidad de no ser mina
x: Tapada00 -> Probabilidad de no ser mina: 0.7777777777777778
x: Tapada01 -> Probabilidad de no ser mina: 0.6
x: Tapada03 -> Probabilidad de no ser mina: 0.6
x: Tapada11 -> Probabilidad de no ser mina: 0.6
x: Tapada12 -> Probabilidad de no ser mina: 0.6
x: Tapada13 -> Probabilidad de no ser mina: 0.6
Lista de mejores ocultas con alta probabilidad de no ser minas: ['Tapada00']
Tiempo de paso: 0.0 segundos
Destapando casilla: ['Tapada00']
¡Has encontrado una mina! Has perdido.
Ultimo tablero resuelto
T T T T T T T T
T T T T T T T T
T T T T T T T T
T T T T T T T T
T T T T T T T T
T T T T T T T T
T T T T T T T T
T T T T T T T T
T T T T T T T T
T T T T T T T T
T T T T T T T T
T T T T T T T T
Has perdido!
Terminado en 0.0 segundos

```

- Tamaño: 10 x 10 con 25 minas

```

Tablero Destapado
1 2 M 4 M 3 M 1 2 M
2 M 4 M M 3 1 1 2 M
2 M 3 2 2 1 0 0 2 2
1 2 2 1 1 1 1 0 2 M
1 3 M 2 1 M 2 1 2 M
M 3 M 3 3 4 M 2 3 2
1 2 1 2 M M 4 M 2 M
0 1 1 3 4 M 4 2 4 2
0 1 M 2 M 2 2 M 2 M
0 1 1 2 1 1 1 1 2 1

```

```

Tablero oculto
1 T T T T T T T T
T T T T T T T T
T T T T T T T T
T T T T T T T T
T T T T T T T T
T T T T T T T T
T T T T T T T T
T T T T T T T T
T T T T T T T T
T T T T T T T T
T T T T T T T T
T T T T T T T T

```

```

Lista de casillas ocultas:
['Tapada01', 'Tapada02', 'Tapada10', 'Tapada11']
Lista de casillas destapadas:
['Destapada00']
Lista de minas encontradas:
[]
Oculta con mayor probabilidad de no ser mina
x: Tapada01 -> Probabilidad de no ser mina: 0.6666666666666667
x: Tapada02 -> Probabilidad de no ser mina: 0.7474747474747474
x: Tapada10 -> Probabilidad de no ser mina: 0.6666666666666667
x: Tapada11 -> Probabilidad de no ser mina: 0.6666666666666667
Lista de mejores ocultas con alta probabilidad de no ser minas: ['Tapada02']
Tiempo de paso: 0.0 segundos
Destapando casilla: ['Tapada02']
¡Has encontrado una mina! Has perdido.
Ultimo tablero resuelto
1 T T T T T T T T
T T T T T T T T
T T T T T T T T
T T T T T T T T
T T T T T T T T
T T T T T T T T
T T T T T T T T
T T T T T T T T
T T T T T T T T
T T T T T T T T
T T T T T T T T
T T T T T T T T
Has perdido!
Terminado en 2.0 segundos

```

Con respecto a los resultados obtenidos en las distintas pruebas del tablero se considera lo siguiente:

- La gran eficiencia de tiempo en los resultados obtenidos para tamaño de tableros de bajas dimensiones ya que, por ejemplo, para el tablero con tamaño 5x5x5 lo ha resuelto en apenas 5 segundos.
- Con respecto a tableros de más dimensiones, cabe destacar el buen resultado en su mayoría ya que es capaz de resolver tablero de 8x8x15 en 19 segundos dándose un resultado de victoria.
- Para el tablero de mayor dimensión se ha observado que el tiempo sube considerablemente por la complejidad de dicho tablero. Ha habido intentos que se ha resultado en minutos y otros que directamente han tardado mucho más.

Como conclusión se debe saber que estos tiempos no son del todo reales ya que varían conforme a la situación de cada uno de los tableros, pueden enfrentarse a la situación de tener todas las casillas con la misma probabilidad de contener una mina y eso puede alargar o acortar el tiempo de juego en función de que si la casilla que destapa de manera arbitraria ante tal situación sea una mina. Conforme a las victorias pasa de igual manera con respecto a la situación comentada anteriormente ya que al destapar de forma aleatoria la casilla puede contener una mina y darse por finalizado el juego.

V. CONCLUSIONES

Una vez realizadas las comprobaciones con un número determinado de intentos, podemos llegar a la conclusión de que, generalmente, la implementación del algoritmo nos proporciona un alto porcentaje de victorias.

Un dato importante es la rapidez en resolver los tableros, ya que, por ejemplo, un tablero de 8 x 8 con 15 minas lo ha resuelto en apenas 19 segundos.

Cabe destacar que, a pesar de que resolviendo automáticamente el tablero dado numerosos intentos terminen en victoria, otro porcentaje no nos devuelve el mismo resultado. Por ello pensamos que quizás exista otro método que, aplicándolo a este problema, obtengamos un resultado aún mejor. Como mejoras, podría implementarse una interfaz gráfica que haga más visual el procedimiento de resolución automática del tablero.

REFERENCIAS

- [1] Wikipedia – Buscaminas. <https://es.wikipedia.org/wiki/Buscaminas>
- [2] Departamento de Ciencias de la Computación e Inteligencia Artificial
Redes Bayesianas: <https://www.cs.us.es/cursos/iais-2017/temas/RedesBayesianas.pdf>
- [3] Applying Bayesian Networks in the game of Minesweeper
<http://staff.utia.cas.cz/vomlel/vomlel-ova-cze-jap-2009.pdf>
- [4] Librería pgmpy – Python <http://pgmpy.org/>
- [5] Librería time – Python <https://docs.python.org/2/library/time.html>