

1. Concepto de Juego. Elementos y clasificación de los juegos.

- Perfecto para investigar en técnicas de resolución de problemas
- Fácil de medir el éxito o el fracaso.
- Fascinación para una amplia variedad de gente.
- Cada jugador intenta conseguir el mayor beneficio para su interés

DEFINICION:

Cualquier decisión con varios agentes (jugadores) gobernada por un conjunto de reglas y con un resultado bien definido, caracterizada porque ningún jugador con su sola actuación puede determinar el resultado.

La solución de un juego es la determinación de una sucesión de actuaciones que indican a cada jugador que resultado puede esperar y como alcanzarlo

Conclusión: un juego puede plantearse como un problema de maximización, aunque finalmente y en muchos casos solo pueda llegar a una satisfacción.

Elementos de un juego:

- 1) **Estado Inicial:** Posición del tablero e identifica al jugador que mueve.
- 2) **Función sucesor:** Devuelve una lista de pares(movimiento,estado) indicando un movimiento legal y el estado que resulta
- 3) **Test terminal:** Determina cuando se termina el juego
- 4) **Función utilidad:** función objetivo/rentabilidad. Da un valor a los estados terminales

Características de los juegos:

- Numero de jugadores:
 - Dos jugadores competitivo y cooperativo
 - Mas de dos no puede serlo mas de dos cosas.
- **Tipo de juego:**
 - Información Perfecta: Sabes en todo momento lo que tienes delante. Ej Ajedrez
 - Información imperfecta: Ej póker: no sabes las cartas de tu adversario.
- Orden Actuación de los jugadores:
 - Depende de quien juegue primero actuara de una manera u otra. Ej Ajedrez blanca atacan, negras defienden
- Juego de suma nula o no nula:
 - Nula(bipersonal): Se pierde,gana o empata. Lo que gana uno lo pierde el otro. Ej Ajedrez
 - No nula: Hay pagos colaterales(dinero).El dinero que se circula se reparte entre los Jugadores

2.El algoritmo Minimax. Componentes y funcionamiento

El algoritmo de minimax consiste en la elección del mejor movimiento para el computador, suponiendo que el contrincante escogerá uno que lo pueda perjudicar, para escoger la mejor opción este algoritmo realiza un árbol de búsqueda(búsqueda en profundidad) con todos los posibles movimientos, luego recorre todo el árbol de soluciones del juego a partir de un estado dado, es decir, según las casillas que ya han sido rellenadas.

Componentes:

Estado inicial: Es una configuración inicial del juego, es decir, un estado en el que se encuentre el juego.

Operadores: Corresponden a las jugadas legales que se pueden hacer en el juego, en el caso del tres en raya no puedes marcar una casilla ya antes marcada.

Condición Terminal: Determina cuando el juego se acabó, en nuestro ejemplo el juego termina cuando un jugador marca tres casillas seguidas iguales, ya se

horizontalmente, verticalmente o en diagonal, o se marcan todas las casillas (empate) .

Función de Utilidad: Da un valor numérico a una configuración final de un juego. En un juego en donde se puede ganar, perder o empatar, los valores pueden ser 1, 0, o -1.

Implementación Minimax: Los pasos que sigue minimax pueden variar, pero lo importante es tener una idea clara de cómo es su funcionamiento.

1. Generación del **árbol de juego**. Se generarán todos los nodos hasta llegar a un estado terminal.
2. Cálculo de los valores de la **función de utilidad** para cada nodo terminal.
3. Calcular el valor de los nodos superiores a partir del valor de los inferiores. Alternativamente se elegirán los valores mínimos y máximos representando los movimientos del jugador y del oponente, de ahí el nombre de Minimax.
4. Elegir la jugada valorando los valores que han llegado al nivel superior.
 - El algoritmo explorará los nodos del árbol asignándoles un valor numérico mediante una función de utilidad, empezando por los nodos terminales y subiendo hacia la raíz.
 - Colocar 0 ó 1 en los nodos terminales dependiendo si gana MIN o MAX
 - La función de utilidad definirá lo buena que es la posición para un jugador cuando la alcanza.
 - Se requiere de una estrategia que garantice llegar a estados terminales ganadores independientemente de lo que haga el oponente.
 - Un valor positivo indica la ventaja de un jugador y uno negativo la ventaja del otro.

3. Modelos de representación del conocimiento

1) **Modelos de Conducta.**

Reglas de Produccion o reglas de if then
Si A entonces B (A:condición B: Accion)

2) **Modelos de Razonamiento.**

Esta basado en la lógica.

Una lógica es un modelo de representación de conocimiento que consta de:

1. Lenguaje formal con:
 - a. Simbolos
 - b. Sintaxis: Reglas para construir cadenas validas (formula bien formada)
 - c. Semantica: Relación entre fbf de lógica y el conocimiento que se quiere representar.
2. Unas reglas de inferencia(Teoría de la demostración) que permitan obtener nuevo conocimiento a partir de conocimiento previo

Las lógicas mas sencillas son: proposiciones y primero orden

Predicados, la descriptiva

3) **Modelos de Memoria.**

Heredable

Dos básicos: simple ()y heredable(clases organizada en forma de arbol) redes semánticas y frames.

Conocer esteriotipos

5. Paradigmas de aprendizaje

El aprendizaje consiste en utilizar las percepciones no solo para actuar , si no también para mejorar la habilidad del agente para actuar en un futuro.

El aprendizaje entra en juego cuando el agente observa sus interacciones con el mundo y sus procesos de toma de decisiones.

Tipos de aprendizaje según su conocimiento:

I. Supervisado.

Para cada entrada se dispone de un profesor/supervisor que proporciona una salida deseada, ya sea una clase o un valor a aproximar (clasificación vs regresión)

Tengo un maestro que me dice que salida le corresponde a cada entrada

II. No supervisado

No se dispone de una salida deseada para cada entrada.

Consiste en aprender a partir de patrones de entrada para los que no se especifica los valores de sus salidas.

III. Refuerzo

En vez de que un profesor indique al agente que hacer, el agente de aprendizaje por refuerzo debe aprender a partir de la información obtenida al realizar procesos de ensayo error.

Ej para aprender a conducir debes practicar y aprender de los errores.

Aprendizaje memorístico.

Es la acción de introducir en la memoria un concepto o idea sin saber o entender su significado, sin tener ideas previas, teniendo como instrumento de aprendizaje las múltiples repeticiones de dicho concepto, para así de este modo poder recordarlo.

Osea retener mentalmente datos sin procesarlos detenidamente.

Aprendizaje deductivo.

- Aplicamos la deducción para obtener descripciones generales a partir de un ejemplo de concepto y su explicación

Aprendizaje analógico.

- Buscamos soluciones a problemas nuevos basándonos en encontrar similitudes con problemas ya conocidos y adaptando sus soluciones.
- Este tipo de aprendizaje intenta emular algunas de las capacidades humanas más sorprendente poder entender una situación por su parecido con situaciones anteriores conocidas.

- Ejemplo: sistemas eléctricos

6. Aprendizaje Inductivo. Empleo de arboles de decisión en Aprendizaje inductivo.

Se trata de aprender un concepto o una clasificación a partir de ejemplos y contraejemplos. El objetivo es aprender la función f .

Un ejemplo es un par $(x, f(x))$.

Dada una colección de ejemplos de f , devolver una función h (hipótesis) que aproxime a f .

La razón por la cual el aprendizaje es difícil es porque no es fácil determinar si una función H es una buena aproximación de F .

Una buena hipótesis estará bien generalizada si puede predecir ejemplos que no se conocen

Arboles de decisión como herramienta de desarrollo

Toma como entrada un objeto o una situación descrita a través de un conjunto de atributos y devuelve una decisión (valor previsto de la salida dada por la entrada).

Los atributos de entradas y salida pueden ser discretos o continuos.

Aprender una función de valores discretos se denomina **clasificación**.

Aprender una función continua se denomina **regresión**.

Resumiendo, un árbol de decisión, desarrolla una secuencia de tests para poder alcanzar una decisión.

Cada nodo corresponde con un test sobre el valor de una de las propiedades, y las ramas que salen de cada nodo están etiquetadas con los posibles valores de dicha propiedad.

Cada nodo hoja representa el valor que ha de ser devuelto si dicho nodo hoja es alcanzado.

