

NOTAS INDIVIDUALES  
ROSALES MARTINEZ AARON.

## OVERVIEW STANFORD

Las máquinas serán capaces, dentro de veinte años, de hacer cualquier trabajo que un hombre puede hacer.- herbert simon dentro de 10 años los problemas de la inteligencia artificial serán sustancialmente resuelto.- marvin minsky

Visualizo un momento en el que seremos para los robots lo que los perros son para

humanos, y estoy apoyando a las máquinas.- Claude Shannon

El curso habla de ¿Qué tienen en común la búsqueda web, el reconocimiento de voz, el reconocimiento facial, la traducción automática, la conducción autónoma y la programación automática? Todos estos son problemas complejos del mundo real, y el objetivo de la inteligencia artificial (IA) es abordarlos con herramientas matemáticas rigurosas.

La IA se ha basado en muchas cosas diferentes, ya sabes, campos, muchas de las técnicas que veremos, por ejemplo, máxima probabilidad, provienen de sus estadísticas o juegos provienen de la economía, optimizaciones, descenso de gradiente, por lo tanto desde- era, ya sabes, en los años 50 sin ninguna relación con la IA.

Pero estas técnicas se desarrollaron en un contexto diferente.

puede hacer sistemas artificiales que tengan el lenguaje, o el motor, o las capacidades de percepción visual que tienen los humanos. Entonces, si observa cómo son las máquinas, cómo han tenido éxito, todo es con un conjunto limitado de tareas y, ya sabe, millones o miles de millones de ejemplos, y simplemente procesa una gran cantidad de cálculos, y realmente puedes optimizar, eh, todas las tareas que se te ocurran. Mientras que los humanos operan en un régimen muy diferente. No necesariamente hacen nada, ya sabes, una cosa bien, pero tienen una especie de diverso conjunto de, ya sabe, experiencias, puede resolver un conjunto diverso de tareas y aprender de cada tarea individual con muy pocos ejemplos. Y todavía es una especie de gran desafío, desde una perspectiva cognitiva, cómo se pueden construir sistemas con este nivel de capacidad que tienen los humanos.

## INTRODUCTION AI

Es la revolución más importante de la tecnología desde que se inventó la informática. La inteligencia artificial va a cambiarlo todo (ya lo está haciendo), aunque no tenemos claro cuándo, ni cómo... ni por qué. Es la gran paradoja de la IA. Todo el mundo habla de ella, pero pocos saben cómo funciona, o lo que realmente hace. En este artículo vamos a intentar explicar qué es de una forma clara y sencilla, para entender los conceptos básicos y descubrir sus posibilidades.

Uno de los padres de la inteligencia artificial, Marvin Lee Minsky, estaba convencido de que la IA salvaría a la Humanidad. Pero también profetizó en 1970: "Cuando los ordenadores tomen el control, quizá ya no lo podamos volver a recuperar. Sobreviviremos mientras ellos nos toleren. Si tenemos suerte, quizá decidan tenernos como sus mascotas".

¿Qué es entonces la inteligencia artificial?. Decimos que la inteligencia artificial se demuestra cuando una tarea que realiza un ser humano y que se considere que requiere la capacidad de aprender, razonar y resolver problemas se puede hacer con una máquina. Ésta es en sí misma una cuestión filosófica profunda. Los intentos de responder sistemáticamente caerán dentro de los fundamentos de la inteligencia artificial como un rico tema de análisis y debate. Este no es el objetivo de esta conferencia. Entonces, por el momento, tendremos una respuesta provisional. Una respuesta provisional podría ser la siguiente.

La inteligencia artificial es el campo dedicado a la construcción de artefactos capaces de mostrar en entornos controlados, bien entendidos y durante períodos prolongados de tiempo, comportamientos que podrían considerarse inteligentes; o más en general, comportamientos que están en el corazón de lo que es tener una mente. Dos cosas son muy importantes en esta definición de trabajo de IA con la que avanzamos.

## PROBLEM SOLVING AS STATE SPACE SEARCH

comprender los principales tipos de agentes y arquitecturas:

dirigido a objetivos vs basado en utilidades

deliberativo vs recreativo

basado en modelo vs sin modelo

aprenda el modelo y la construcción algorítmica, bloques para la creación de agentes

Modelar el problema de forma adecuada representación formal.

Especificar, analizar, aplicar e implementar razonamientos algoritmos para resolver las formulaciones de problemas.

Resolución de problemas como

Búsqueda de espacio de estado

Formulación de problemas (modelado)

Resolución de problemas como búsqueda de espacios de estados

Representación formal

Gráficos y árboles de búsqueda

Algoritmos de razonamiento

Búsqueda en profundidad y en amplitud

La mayoría de los bloques de creación de agentes Implementaciones

Usar búsqueda Operaciones robustas:

Planificación de actividades

Diagnóstico

Reparación

Planificación

Asignación de recursos

Movilidad:

- Planificación de ruta
- Localización
- Construcción de mapas
- Trayectoria de control

Diseño

\* Los problemas de inteligencia artificial es → la realización de que la inteligencia requiere conocimientos.

\* La técnica de inteligencia artificial es → un método que explota conocimiento y debe ser representado de tal manera que:

El conocimiento captura generalizaciones

Entendida por personas que lo proveen

Fácilmente modificable y usada para ayudar a superarse

\* Para crear un sistema que resuelva un problema particular (Genérico) se necesitan cuatro cosas:

- Definir precisamente el problema
- Analizar el problema
- Aislar y representar el conocimiento de la tarea

- Escoger las mejores técnicas de resolución de problemas
- \* Programación con IA
  - El programa solamente puede responder las preguntas específicas que tiene destinado a resolver
  - El programa con IA puede resolver preguntas genéricas que están diseñadas para resolver
  - Modificaciones en el programa conduce a cambios en la estructura
  - Modificaciones en el programa no cambia la estructura independiente de la información
  - Las modificaciones puede afectar negativamente el programa
  - El programa es rápido y fácil de modificar
  - Las tareas de resolución de problemas pueden ser formuladas como una búsqueda en un estado de espacio.
  - Un estado de espacio consiste en todos los estados de un dominio y a un conjunto de operadores que cambian un estado a otro.
  - Los estados se pueden pensar mejor como nodos en un grafo conectado y los operadores como bordes.

Espacio de estado:

Se define al espacio de estado aquel que contiene todas las posibles configuraciones de los objetos relevantes.

Estados iniciales:

Especifica uno o más estados dentro de ese estado como posibles soluciones a partir de las cuales la resolución de problemas pueda empezar.

Estados finales:

Especifica uno o más estados que pueden ser aceptables como soluciones al problema.

Un sistema que consta de:

Una base de datos separado, operaciones y componentes de control representan un bloque de construcción metafórico apropiado para construir descripciones lucidas de sistemas de inteligencia artificial.

Un sistema que consta de una base de datos separado, operaciones y componentes de control representan un bloque de construcción metafórico apropiado para construir descripciones lucidas de sistemas de inteligencia artificial.

Los sistemas de inteligencia artificial muestran una separación más o menos rígida entre datos, operaciones y control.

Varias generalizaciones de los formalismos computacionales conocidos como sistemas de producción envuelve a una separación limpia de esos componentes computacionales y así parecen capturar la esencia del funcionamiento de muchos sistemas de inteligencia artificial

## UNINFORMED SEARCH

- Revisión
- Análisis - Búsqueda en profundidad primero - Búsqueda en amplitud primero
- Profundización iterativa
- Sistemas autónomos:
  - Planificar secuencias complejas de acciones
  - Programe recursos limitados
  - Monitorear y diagnosticar el comportamiento
  - Reparar o reconfigurar el hardware

### Formalización de la búsqueda de gráficos

Input: A search problem  $SP = \langle g, S, G \rangle$  where  $g$  = graph,  $S$  = start vertex in  $V$ , and  $G$  = goal vertex in  $V$ . Output: A simple path  $P = \langle v_0, v_1, \dots, v_n \rangle$  in  $g$  such that  $v_0 = S$  and  $v_n = G$ .

Pseudocódigo para búsqueda simple

Sea  $g$  un Gráfico  $G$  el vértice de la meta de  $g$ .

$S$  el vértice inicial de  $g$   $Q$  sea una lista de senderos parciales simples en  $GR$ ,

1. Inicialice  $Q$  con la ruta parcial  $(S)$  como única entrada; establecer  $Visitas = \{ \}$ ;
2. Si  $Q$  está vacío, falla. De lo contrario, elija algún camino parcial  $N$  de  $Q$ ;
3. Si  $cabecera(N) = G$ , devuelve  $N$ ; (¡meta alcanzada!)
4. Más a) Eliminar  $N$  de  $Q$ ; b) Encuentre todos los hijos de la cabeza  $(N)$  (sus vecinos en  $g$ ) que no estén en  $Visitas$  y crear una extensión de un paso de  $N$  para cada niño; c) Sumar a  $Q$  todos los trayectos extendidos; d) Agregar a los hijos del jefe  $(N)$  a los visitados; Brian Williams, otoño 10 9

## Busqueda Heuristica

### Heurísticos

El estudio de los metodos y reglas para la invencion y los descubrimientos. Utilizar nuestro conocimiento en el problema para considerar algunos sucesores del estado actual. Esto haría que se podara el espacio de estados ganando velocidad pero pudiendo perder la solucion. Para hacer búsqueda heurística para trabajar, se tiene que ser posible clasificar a los hijos de un nodo.

Una función heurística toma un estado y regresa un valor numérico -- una evaluación compuesta de este estado. Entonces se elige a un hijo con el mejor resultado posible(ya sea un máximo o un mínimo).

Una función heurística puede ayudar ganando o perdiendo mucho pero encontrar la función correcta no siempre es facil.

En una búsqueda de inteligencia artificial, la palabra heurística tiene una definición muy especifica. Una heurística es una función que estima que tan cerca un estado esta en la meta. La función heurística toma un estado y regresa un estimado de su distancia (costo de camino) del estado final.

La principal ganancia es la reducción del espacio de estados

- Por ejemplo el árbol completo de Gato tiene 9! ramas, si consideramos simetrías el árbol se vuelve seis veces más pequeño pero sigue siendo largo.
- Con una función heurística vamos a obtener un árbol de tan solo cuarenta estados.

## INFORMED SEARCH

La información heurística es usada para hacer búsquedas expansivas a lo largo de estos sectores de la frontera que son más prometedores.

En una búsqueda heurística cada estado es asignado a un valor heurístico que el buscador utiliza seleccionando el mejor paso siguiente

Para ordenar nodos para la expansión se necesita un método para calcular la promesa de un nodo. Esto está hecho usando una función de evaluación de valor real.

Las funciones de evaluación están basadas en una variedad de ideas.

## Búsqueda informada o heurísticas

La búsqueda no informada es ineficiente en la mayoría de los casos. El propósito de la búsqueda informada es utilizar conocimiento específico del problema para alcanzar el objetivo de manera más eficiente. La idea es ser capaces de medir la “calidad” de un estado.

La idea es ser capaces de medir la “calidad” de un estado. Eso nos permitirá dirigir la búsqueda por los estados mejores que estarán “más cerca” del objetivo y no seguir estrategias en anchura o profundidad que no tienen en cuenta la calidad de los estados. Las estrategias de búsqueda informada son mucho más eficientes que las no informadas.

### Función de evaluación

Función de evaluación  $f(n)$ , mide la calidad de  $n$ .

Un nodo tendrá calidad cuanto menor sea la distancia al objetivo.  $f(n)$  estima la distancia desde ese nodo  $n$  a un nodo objetivo. Las búsquedas informadas expanden primero los nodos que están más cerca del objetivo; i.e. aquellos en los que la función  $f(n)$  asigna un menor valor.

### Búsqueda primero el mejor

Idea: usar la función de evaluación  $f(n)$  para cada nodo, de modo que se pueda estimar su “deseabilidad” y sea el nodo más deseable que no ha sido expandido el que se elija para expandir.

Implementación: los nodos en la frontera deben ordenarse de forma decreciente con respecto a su deseabilidad. Casos especiales: Búsqueda voraz primero el mejor  $A^*$ .

### Algoritmo $A^*$

1 Minimiza el costo estimado total de la solución

Evalúa los nodos combinando  $g(n)$  y  $h(n)$

$g(n)$ : costo de haber alcanzado  $n$

$h(n)$ : costo para llegar desde  $n$  hasta el objetivo

$f(n) = g(n) + h(n)$  costo más barato estimado de la solución a través de  $n$



## CONSTRAINT SATISFACTION PROBLEMS

Son problemas matemáticos definidos como un conjunto de objetos tal que su estado debe satisfacer un número de restricciones o limitaciones. Los CSP representan las entidades de un problema como una colección homogénea finita de restricciones sobre variables las que son resueltas por métodos de satisfacción de restricciones.

- En una búsqueda de estado de espacios se explora la idea de que problemas pueden ser resueltos buscando en un espacio de estados.
  - Estos estados pueden ser evaluados por una heurística de dominio específico y probada para ver si son estados finales
  - Cada estado es una caja negra accedida únicamente por rutinas de problemas específicos, la función sucesora, funciones heurísticas y pruebas finales.
  -
- Problemas de satisfacción de constantes examina problemas donde estados y pruebas finales concuerdan con la representación estándar y estructurada.
  - Se pueden definir los algoritmos de búsqueda que aprovechen la estructura de estados y usar heurísticas de propósito general en vez de específicas de problemas para generar las soluciones de problemas largos.

Un problema de satisfacción de restricción es definido como una tripla  $\{X, D, C\}$  donde  $X$  es un conjunto de variables,  $D$  es el dominio de valores, y  $C$  es un conjunto de restricciones

## SEARCHING

- Representan la solución de problemas que pueden ser resueltos descomponiendo los en partes de pequeños problemas, todos de ellos tienen que ser resueltos
  - el arco AND puede apuntar a cualquier número del nodo sucesor, todos tienen que ser solucionados en el orden del arco para poner una solución
  - los arcos AND son indicados con una línea conectando todos los componentes

los grafos AND OR son definidos como hipografos:

- En vez de arcos conectando pares de nodos, estos son hiper arcos conectando un nodo pariente con un conjunto de nodos sucesores.
- Estos hiper arcos son llamados conectores

Para encontrar una solución en un grafo AND OR se necesita un algoritmo similar al best first search pero con la habilidad de manejar el arco AND apropiadamente.

El algoritmo debería encontrar un camino desde el nodo inicial a un conjunto de nodos representando la solución de estados.

### Redes Bayesianas

Esencialmente una red bayesiana representa, mediante un grafo dirigido, las relaciones de dependencia entre un conjunto de variables relevantes para cierto dominio.

La programación bayesiana es una metodología y formalismo que permite especificar y resolver modelos probabilistas



## Minimax Alpha-Beta

minimax es un método de decisión para minimizar la pérdida máxima esperada en juegos co adversario y con información perfecta este cálculo se hace de forma recursiva

el algoritmo Minimax es:

1. Generación del árbol de juego: A partir del nodo que representa el estado actual, se generan todos los nodos hasta llegar a un estado terminal (si no podemos afrontar la generación del árbol completo, es posible aplicar los pasos siguientes sobre una sección del mismo, aunque entonces no podremos asegurar la optimalidad de los resultados).
2. Se calculan los valores de la función de evaluación para cada nodo terminal (o las hojas que hayamos conseguido si no hemos podido construirlo entero) del árbol construido.
3. Se evalúan los nodos superiores a partir del valor de los inferiores. Según si estos nodos pertenecen a un nivel MAX o un nivel MIN, se elegirán los valores mínimos y máximos representando los movimientos del jugador y del oponente.
4. Se repite el paso 3 hasta llegar al nodo superior (estado actual).
5. Se selecciona la jugada-nodo directamente accesible desde el nodo actual que optimiza el valor de la evaluación.

El minimax aporta una herramienta de proceso recursiva muy útil

Debe destacarse que, si no es posible realizar el árbol completo de jugadas, pero sí disponemos de una función que permite evaluar lo bueno que es un estado intermedio (por medio de una heurística), aún podemos hacer uso del algoritmo minimax para decidir cuál es el mejor movimiento posible a partir del estado actual. Por supuesto, dependerá de la calidad de la función heurística el que la estrategia que proporciona la ejecución del algoritmo sea ganadora o no. El problema de muchos juegos es que no es posible trabajar con el árbol completo de jugadas posibles y tampoco se conoce una heurística que asegure una estrategia ganadora.

Poda alfa-beta de la búsqueda Minimax. El problema de la búsqueda Minimax es que el número de estados a explorar puede llegar a ser exponencial respecto al

número de movimientos. Si  $r$  es el factor de ramificación (cuántos hijos tiene cada nodo), y  $m$  es el nivel de profundidad que vamos a alcanzar, entonces la complejidad en tiempo es del orden  $O(r^m)$  y la complejidad en espacio del orden  $O(rm)$ .

La Poda alfa-beta es una técnica que reduce el número de nodos evaluados en el árbol de juego construido por el algoritmo Minimax. Para ello, trata de eliminar partes grandes del árbol que se va construyendo de forma que se devuelva el mismo movimiento que devolvería este, podando ramas que se sepa que no van a influir en la decisión final.

La idea de esta técnica es que cada nodo se analiza teniendo en cuenta el valor que por el momento tiene y el valor que por el momento tiene su padre, lo que determina en cada momento un intervalo  $(\alpha, \beta)$

de posibles valores que podría tomar el nodo. El significado intuitivo de estos parámetros en cada momento es:

- En los nodos MAX:  $\alpha$  es el valor actual del nodo (que tendrá ese valor o superior), y  $\beta$  es el valor actual del padre (que tendrá ese valor o inferior).
- En los nodos MIN:  $\beta$  es el valor actual del nodo (que tendrá ese valor o inferior), y  $\alpha$  es el valor actual del padre (que tendrá ese valor o superior).
- La poda se produce si en algún momento  $\alpha \geq \beta$ , y en ese momento no hace falta analizar los restantes sucesores del nodo. En nodos MIN, se denomina poda  $\beta$ , y en nodos MAX, poda  $\alpha$ .

## Lógica proposicional

Los enunciados más simples se tratan como unidades individuales y eso hace es fundamentalmente diferente de la lógica aristotélica,

Una proposición es un enunciado que en sí mismo es verdadero o falso.

La lógica proposicional es un sistema matemático para razonar sobre proposiciones y mirar en cómo se relacionan entre sí.

- Cada enunciado en la lógica proposicional consta de variables de proposiciones que luego se combinan a través de conectivos proposicionales
- Cada variable representa alguna proposición
- Los conectivos codifican cómo se relacionan las proposiciones

VARIABLE PROPOSICIONAL: representa una proposición arbitraria.  
Representamos variables proposicionales con letra mayúscula

CONECTIVOS PROPOSICIONALES:

LÓGICA NOT:  $\neg P$

- READ “NOT P”
- $\neg P$  es verdadero si y solo si P es falso
- También llamado negación lógica

Lógica AND:  $P \wedge Q$

- Read “P and Q”
- $P \wedge Q$  es verdadero si ambos P y Q son verdaderos
- También llamado conjunción lógica

Lógica OR:  $P \vee Q$

- Read “P or Q”
- $P \vee Q$  son verdaderos si al menos uno de P o Q es verdadero lógico
- También denominada lógica disyunción

Implicación:  $P \rightarrow Q$

- Read “si P entonces Q”
- Falso cuando P es verdadero y Q es falso; y es verdadero de otra manera
- También llamado material operador condicional

Bicondicional:  $P \leftrightarrow Q$

- Read “P si y solo si Q”
- Verdadero si P y Q tienen lo mismo con valores y falso en caso contrario
- También llamado material operador bicondicional

Verdadero y falso

- El símbolo T es un valor que siempre es verdadero
- Mientras el símbolo  $\perp$  es un símbolo que siempre es falso

## First Order Logic -I

Uno de los lenguajes formales más potentes para la representación del conocimiento anteoría de la computación es el lenguaje formal de la Lógica de primer orden que carece de imprecisiones y posee una forma clara de representación basada en la forma y no en el contenido. Formaliza hechos o proposiciones acerca del mundo obteniendo fórmulas lógicas. Para ello, considerados niveles de abstracción, quedan lugar al

lenguaje proposicional y predicativo. El lenguaje proposicional tiene en cuenta sólo las posibles conexiones entre ellas; sin embargo el lenguaje predicativo además de tener en cuenta dichas conexiones, considera los sujetos o individuos que aparecen en las proposiciones, las propiedades o características que les afectan y las posibles relaciones entre ellos.

Lenguaje formal lógico:

- ❖ Es un lenguaje que se define completamente.
- ❖ Se identifica por el conjunto de sus fórmulas lógicas bien formadas que denotaremos por  $\mathcal{F}$ .
- ❖ Una fórmula lógica es una expresión lógica que representa o formaliza a una proposición.

Para definir un lenguaje formal se requiere:

- ❖ Designar el conjunto de símbolos (alfabeto) del lenguaje.
- ❖ Definir el conjunto de reglas de formación de fórmulas de lenguaje.
- ❖ Definir la interpretación de cada símbolo que interviene en la  $\mathcal{F}$ .

Lenguaje de la lógica de proposición es el lenguaje propio de la lógica de proposiciones, nivel básico de la lógica de primer orden. Su origen se remonta a finales del siglo XIX, coincidiendo con la aparición de las obras de G.Boole(1815-1864) y de G.Frege(1848-1925)

Permite construir fórmulas lógicas proposicionales a partir de la formulación de proposiciones atómicas y moleculares que definen un determinado problema de razonamiento. Su sintaxis se define a partir de un alfabeto, con los elementos básicos del lenguaje, y unas reglas de formación de fórmulas proposicionales.

Su Semántica se define a partir del significado de cada una de las componentes que aparecen en una fbf. Su unidad básica de información es la proposición simple que sintácticamente estará definida por un afbf formada por una variable proposicional y semánticamente se interpretará como verdadera o falsa.

## FIRST ORDER LOGIC - II

Logica de primer orden conocida como logica de predicados, logicca cuantificacional y calculo de predicados de primer orden, es una coleccion de sistemas formales usados en matematicas, filosofia, linguistica e informatica. La logica de primer orden usa variables cuantificadas sobre objetos no logicos y permite el uso de oraciones que contienen variables de modo que en lugar de proposiciones como '*Socrates es un hombre*' se pueden hacer expresiones en la forma '*Existe X tal que X es Socrates y X es un hombre*' donde '*existe*' es un cuantificador, mientras que x es una variable.

Esto la distingue de la plogica propisicional que no usa cuantificadores ni relaciones, en este sentido, la logica proposicional es la base de la logica de primer orden.

### CONCEPTUALIZACION

Se ha argumentado que la inteligencia depende del conocimiento de la entidad que tiene sobre su entorno

- Conocimiento Declarativo
  - Mucho de el conocimiento del entorno es descriptivo y puede ser expresado en una forma declarativa
- La formalizacion del conocimiento en una forma declarativa empieza con la *conceptualizacion*.

- Incluido los objetos que presumen existir o hipoteticamente existen en el mundo y en sus interrelaciones
- ¡Los objetos pueden ser cualquier cosa de la que queramos decir de algo!

Muchas funciones se pueden definir. El conjunto de funciones enfatizadas a la conceptualización es llamado el conjunto de bases funcionales.

En el mismo ejemplo si queremos conceptualizar una parte, se puede declarar una funcion como *hat* que puede tomar la parte de un bloque que esta arriba de otro si alguno llega a existir *hat*:  $\{ \langle B,A \rangle, \langle C,B \rangle, \langle E,D \rangle \}$

La relacion que es el segundo tipo de interrelación entre objetos en este *Universo de discurso*, muchas relaciones se pueden definir, un conjunto de relaciones enfatizadas en una conceptualizacion se llama conjunto de bases relacionales.

Clear: significa que ningún bloque está encima del otro bloque

Above: alrededor de dos cuadras si y solo si está encima de la otra.

On: se mantiene si y solo si uno está inmediatamente encima del otro

Para la escena los elementos para las diferentes relaciones son:

on:  $\{ \langle A,B \rangle, \langle B,C \rangle, \langle D,E \rangle \}$

above:  $\{ \langle A,B \rangle, \langle B,C \rangle, \langle A,C \rangle, \langle D,E \rangle \}$

clear:  $\{ \langle A,D \rangle \}$

No hay una respuesta comprensible sobre que conceptualizacion es mejor que otra asi que se considera “correcta” si es consistente con el mundo.

Nosotros tenemos un conjunto de sentencias y una conceptualizacion de el mundo; nosotros asociamos simbolos usados en las sentencias con objetos, funciones y relaciones de nuestra conceptualizacion.

Con las semanticas declarativas que son la perspectiva del observador se evalua el conocimiento base con el valor verdadero de las sentencias en el mundo se puede determinar que es cierto o no

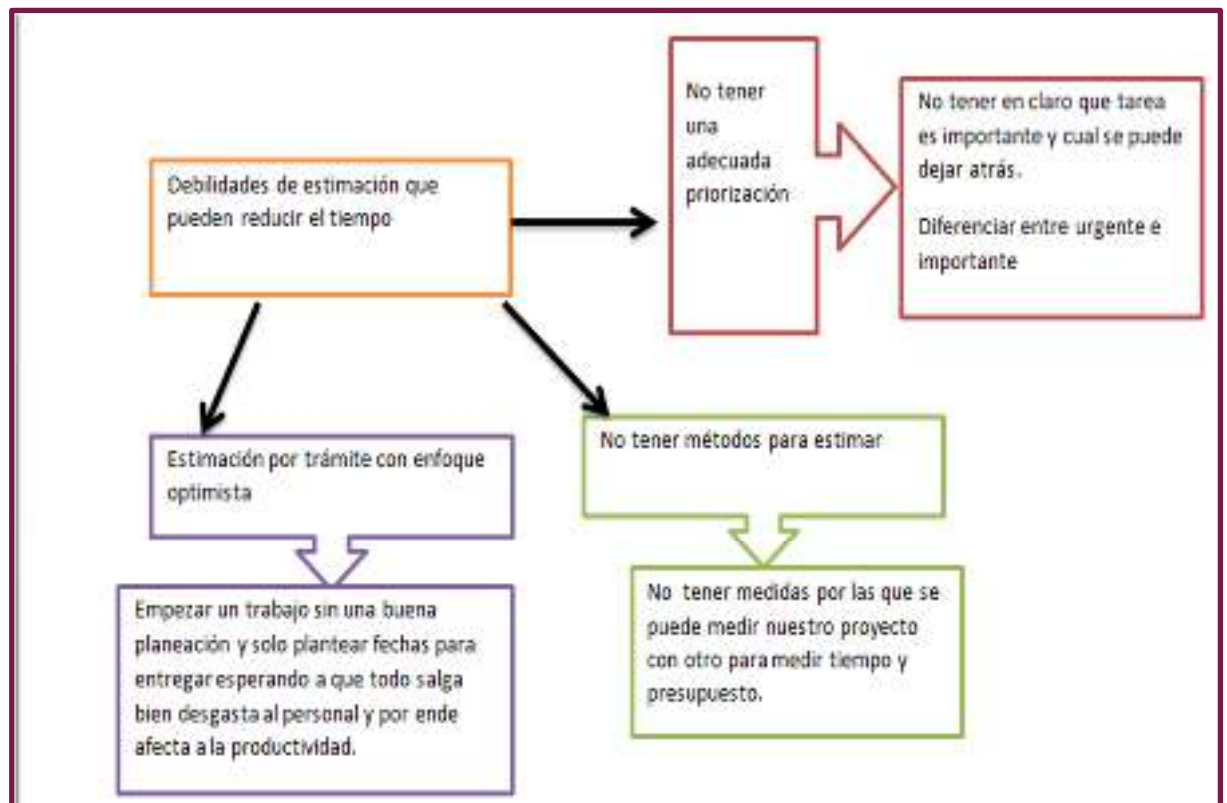
### ***Interpretacion***

Una interpretación es un trazado de mapas entre elemetos de el lenguaje y elementos de la conceptualizacion.



## Combatiendo barreras de productividad a través de planeación práctica para desarrollos de Software

Planeación del desarrollo del software >>En el video nos plantean que cuando iniciamos un proyecto en la parte de la planeación del desarrollo de software en ocasiones se hacen estresantes por tomar una planeación general que en ocasiones no es para el proyecto planteado es decir en la mayoría de las prácticas es por lo general tomarlo como un producto único después están las malas prácticas que afectan negativamente al producto final, a la productividad y a la calidad del proyecto. Existen muchas malas prácticas que se pueden encontrar en la industria por lo general en los proyectos se refuerzan en 3 áreas: estimación, administración de tiempo y comunicación.



### Comunicación

Iniciar sin identificar todos los requerimientos y separarlos por su prioridad enfocado al negocio con ayuda del cliente.

- o Cliente es de mucha ayuda y el que prioriza los requerimientos con nuestra ayuda
- o Evaluar las capacidades del equipo de trabajo.

Delegar tareas sin tener en claro una idea o tiempos claros y transmitirla de forma correcta a los demás.

No poder negarse a una idea o situación que se desarrolle a lo largo del proyecto que afecten de manera importante y negativa al proyecto.

## ADMINISTRADOR DE PROYECTOS

Un proyecto se define como el conjunto de las actividades que desarrolla una persona o una entidad para alcanzar un determinado objetivo, se encuentran interrelacionadas y se desarrollan de manera coordinada.

El proyecto es igual a un objetivo en común más un calendario más un costo (los dos últimos son finitos).

Normalmente en un grupo de trabajo se encuentra un **líder**, siendo este el que da dirección al proyecto, pero ¿Qué es un líder de proyecto?

El líder del proyecto es el responsable de encontrar las necesidades de los usuarios y gestionar los recursos económicos, materiales y humanos, con el fin de obtener los resultados esperados en los plazos que se hayan previsto y con la calidad necesaria, el líder debe tener ciertas habilidades para poder cumplir su función, tales como:

- Orientación al logro de objetivos
- Manejo de conflictos
- Capacidad de comunicación
- Capacidad para trabajar bajo presión

Y no solo eso, también debe de tener competencias técnicas como:

- Metodologías de desarrollo e implementación de proyectos
- Gestión de recursos humanos

Lo importante a destacar es que el líder del proyecto no es el responsable de todo lo que pase en el grupo de trabajo, todas las personas que están en el grupo deben estar comprometidos con el proyecto para que salga bien, ya de lo contrario, los cambios que puedan haber en el transcurso del proyecto o los problemas que surjan van a ser más difíciles de sobrepasar, afectando la calidad del mismo e incluso que el proyecto sea cancelado por falta de tiempo.

## PROCEDURAL CONTROL OF REASONING

Dentro de la representación y el razonamiento del conocimiento se encuentra la resolución lógica de primer orden como mecanismo de inferencia, refutación de resolución pruebas y extracción de respuestas.

Es necesario recordar que la implicación en la lógica de primer orden es semidecidible, es decir, los algoritmos existen que pueden decir que si para cada oración implicada. pero no existe algoritmo que puedan decir no a toda la oración no implicada, la resolución es la refutación completa. eso es para un insatisfactorio conjunto de cláusulas alguna rama contendría la cláusula vacía, se garantiza una búsqueda en amplitud para mostrar que el conjunto de cláusulas no es satisfactorio. sin embargo, para un conjunto de cláusulas satisfactorias, la búsqueda puede o no terminar.

nuestra base de conocimientos consta de una sola fórmula, a qui esta fórmula es:

$$\forall x y z [(R(x,y) \wedge R(y,z)) \rightarrow R(x,z)]$$

mostrando la relacion R es transitiva, se podria pensar en R como una relacion que define relativo, entonces R(x,y) podria significar que X es el relativo de Y, ahora si X es una pariente de Y y Y es un pariente de Z entonces sabemos que C es una pariente de Z que es la regla en nuestra base de conocimiento.

$$(\neg R(x,y) \vee \neg R(y,z) \vee R(x,z))$$

la existencia de alguien pára todos los que no sean parientes

$$\exists x \forall y \neg R(x,y)$$

si existiera X para todo Y y no R de X,Y.

se negara la meta y se convertira en forma clausal.

$$\neg [\exists x \forall y \neg R(x,y)]$$

$$\forall x \exists y R(x,y)$$

la forma clausal aquí y la única regla disponible que puede ver que la base de conocimientos no conlleva la consulta si es negación así que una resolución debería fallar, el problema es si plantearlo como una resolución terminamos generando una secuencia infinita.

$$R(x, f(x))$$

nunca se llegará a la cláusula vacía para la derivación de la resolución.

$$(\neg R(x,y) \vee \neg R(y,z) \vee R(x,z)) \quad R(x, f(x))$$

la primera cláusula que proviene de la regla en la base de conocimientos y la otra que la negación de la meta tenemos efectos de sustitución para Y y que se resuelve en la resultante que está aquí.

$$f(x)/y$$

$$(\neg R(f(x), z) \vee R(x, z))$$

si se resuelve mas lejos, se obtiene una expresion que diga R de Y de F de F de Y una vez alli

$$y/x; f(f(y))/z \quad f(y)/x$$

$$R(y, f(f(y)))$$

Si se resuelve más se obtendrá una disyunción y cuando esta disyunción se resuelva tendremos una más.

$$\begin{array}{c} f(f(x))/y \quad x/y \\ \diagdown \quad \diagup \\ (\neg R(f(f(x)), z) \vee R(x, z)) \end{array}$$

cuando se resuelve la negación de la meta se obtendrá la expresión

$$\begin{array}{c} y/x; f(f(f(y)))/z \quad f(f(y))/x \\ \diagdown \quad \diagup \\ R(y, f(f(f(y)))) \end{array}$$

R de Y de F de F de F de Y

si se razona en las cláusulas posteriores que se han obtenido

$$R(y, f(f(y))) \quad R(y, f(f(f(y))))$$

se puede observar que cada segundo se repite, así que no podemos utilizar una profundidad primer primer procedimiento para buscar la cláusula vacía, se puede quedar atascado.

para buscar una cláusula vacía en una ruta dada por la lógica de primer orden no hay forma de detectar si una rama continuará indefinidamente, esto se debe a:

la lógica de primer orden que uno necesita comprender es un lenguaje muy poderoso, al igual que no hay forma de detectar cuando un programa está en bucle no hay forma de detectar si una rama continuará indefinidamente.

esto es problemático desde un conocimiento perspectiva de representación, no existen procedimientos que dado un conjunto de cláusulas devuelven satisfactorio cuando las cláusulas son resolución satisfactoria sin embargo es insatisfactorio y cuando las cláusulas son satisfactorias, puede o no terminar.

por eso la resolución no es una panacea, la resolución no proporciona una eficacia general

## REASONING UNDER UNCERTAINTY

### Redes bayesianas

Las redes bayesianas son una representación gráfica de dependencias para razonamiento probabilístico en el cual los nodos y arcos representan:

Nodos variables proposicionales

Arcos: Dependencia probabilística

La variable a la que apunta el arco es dependiente (causa->efecto ) de la que esta en el origen de esta

Podemos interpretar una RB de dos formas.

1. distribución de probabilidad conjunta de las variables representadas en la red representa la distribución de la probabilidad.

$$P(C,I,F,D)=P(C)P(I|C)P(F|I)P(D|I)$$

2. Base de reglas

Cada arco representa un conjunto de reglas que asocian las variables involucradas,

Por ejemplo:

Si I entonces F    Dichas reglas están cuantificadas por las probabilidades respectivas.

### Estructura

- La topología o estructura de la red nos da información sobre las dependencias probabilísticas entre las variables.
- La red también representa las independencias condicionales de una variable (o conjunto de variables) dadas otra variable(s).

### Representación Gráfica

- Una red bayesiana representa en forma gráfica las dependencias e independencias entre variables aleatorias, en particular las independencias condicionales
- Independencia en la distribución

–  $P(X | Y, Z) = P(X | Z)$

• Independencia en el grafo

– X “separada” de Y por Z



Separación

• El conjunto de variables A es independiente del conjunto B dado el conjunto C, si no existe trayectoria entre A y B en que

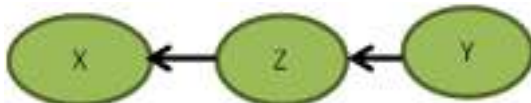
1. Todos los nodos convergentes están o tienen descendientes en C
2. Todos los demás nodos están fuera de C

Tres casos básicos

Arcos divergentes->



Arcos en secuencia->



Lec 19: Bayesian Network

Puede resultar muy difícil especificar las probabilidades de los eventos atómicos; Se requiere una gran cantidad de datos a partir de los cuales se recopilan estimaciones estadísticas.K

Es útil usar una notación gráfica para representar la condicional relación de independencia que se mantienen en el dominio.

En una red bayesiana :

Cada nodo representa una de las variables aleatorias del dominio.

Un arco dirigido entre dos nodos representa una relación de influencia directa entre las dos variables.

(El arco va de la variable 'causa' a la variable 'efecto').

Los nodos hermanos en el gráfico representan variables que son condicionalmente independiente dada la variable padre.

No se permite que una red bayesiana tenga ciclos.

Nodos y valores

¿Cuáles son los nodos a representar y qué valores pueden tomar, o en qué estado pueden estar? Consideraremos solo los nodos que toman valores discretos.

Los valores deben ser mutuamente excluyentes y exhaustivos.

Los nodos pueden ser nodos booleanos discretos o continuos: representan proposiciones que toman valores binarios

Ejemplo: el nodo de cáncer representa la proposición "el paciente tiene cáncer "

Valores ordenados

Ejemplo: nodo de contaminación con valores bajo, medio, alto

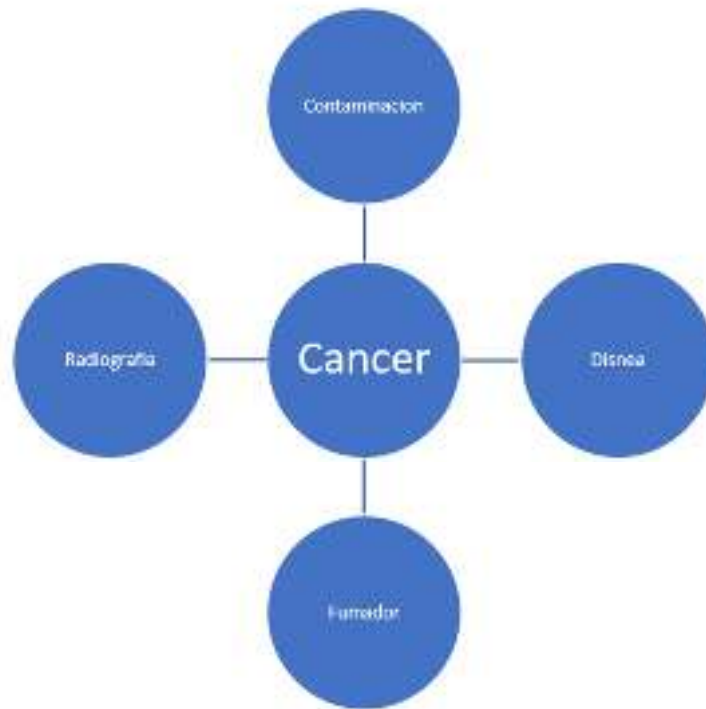
Valores integrales

Ejemplo: Edad con valores posibles 1-120

Ejemplo:



11111111



Dos nodos deben estar conectados directamente si uno afecta o causa al otro, con el arco indicando la dirección del efecto.

Tener cáncer afectará la respiración del paciente y las posibilidades de obtener un resultado radiográfico positivo. Entonces agregamos arcos de Cáncer a Disnea y Rayos X.

### Terminología de estructura

Nodo es un padre de un hijo, si hay un arco desde el primero al segundo.

Para una cadena dirigida de nodos, un nodo es un antepasado de otro si aparece antes en la cadena, mientras que un nodo es un descendiente de otro nodo si viene más adelante en la cadena.

El nódulo de cáncer tiene dos padres, Contaminación y Fumador, mientras que el fumador es un antepasado de rayos X Y disnea

Rayos X es hijo de Cáncer y descendiente de Fumador y Contaminacion.

## DECISION NETWORK

Una red de decisión (también llamada diagrama de influencia) es una representación gráfica de un problema de decisión secuencial finito. Las redes de decisión amplían las redes de creencias para incluir variables de decisión y utilidad. Una red de decisiones extiende la red de decisiones de una sola etapa para permitir decisiones secuenciales.

En particular, una red de decisión es un gráfico acíclico dirigido (DAG) con nodos de azar, nodos de decisión y un nodo de utilidad. Esto amplía las redes de decisión de una sola etapa al permitir que tanto los nodos de azar como los de decisión sean padres de los nodos de decisión.

- Los arcos que entran en los nodos de decisión representan la información que estará disponible cuando se tome la decisión.
- Los arcos que entran en nodos aleatorios representan una dependencia probabilística.
- Los arcos que ingresan al nodo de la utilidad representan de qué depende la utilidad.

Un **agente que no olvida** es un agente cuyas decisiones están totalmente ordenadas, y el agente recuerda sus decisiones anteriores y cualquier información que estuviera disponible para una decisión anterior.

**Una red de decisión sin olvidar** es una red de decisión en la que los nodos de decisión están totalmente ordenados y, si el nodo de decisión  $D_i$  está antes de  $D_j$  en el orden total, entonces  $D_i$  es un padre de  $D_j$ , y cualquier padre de  $D_i$  también es un padre de  $D_j$ . Por lo tanto, cualquier información disponible para  $D_i$  está disponible para  $D_j$ , y la acción elegida para la decisión  $D_i$  es parte de la información disponible en la decisión  $D_j$ . La condición de no olvidar es suficiente para garantizar que las siguientes definiciones tengan sentido y que los siguientes algoritmos funcionen.

Uno de las partes mas importantes de tomar una decisión es conociendo que preguntas hacer, para conducir pruebas criticas dependen de dos factores:

- Que las diferentes entradas puedan marcar una gran diferencia para el curso optimo de la accion
- las probabilidades de las diferentes entradas

## INTRODUCTION TO PLANNING

Planear es ir razonando sobre eventos futuros en orden para establecer una serie de acciones para cumplir una meta.

Un acercamiento comun para planificar es representando un estado actual y determinar las series de acciones necesarias para alcanzar el estado final (o vice versa).

Los planes son creados buscando a traves del espacio posibles acciones hasta que la secuencia necesaria para acompletar la tarea es descubierta, planear es un tipo especifico de busqueda espacial.

Planear involucra buscar a travez del espacio lo cual utiliza:

- Progresion: Elige una accion donde las precondiciones se conocen hasta llegar al estado final
- Regresion: Elige una accion que marque una sub meta incompleta mientras añade precondiciones desconocidas para el grupo de sub metas. Continuar hasta que el grupo de sub metas este vacio.
  - Un enfoque hacia atras, con orientacion a objetivos, tiende ser mas eficiente

### Planeadores no herarquicos

- Metodo mas facil de planear
  - No hace distincion entre planes mas o menos importantes
  - Se ralentiza por atender los elementos menos importantes
  - Falta de estructura entregando mal rendimiento con problemas complejos

### Planeadores herarquicos

- Toma una distincion entre las partes mas importantes del plan
  - Ejemplo: Cuando compras un carro nuevo primero necesita decidir donde encuentra los fondos. No tiene sentido encontrar un buen lugar de estacionamiento en la escuela antes de tener el dinero.

### Supuestos clasicos de planificacion

- Informacion perfecta
- Efectos deterministicos
- Ejecucion instantanea
- Agente en solitario
- No preocuparse del tiempo extra, costo, recursos
- Estatico y finito

## PLAN SPACE PLANNING

### Planificación de la pila de objetivos

- Uno de los planeadores más fáciles STRIPS considera acciones razonando hacia atras; pero comprometido en acciones anticipadas.
- El algoritmo de planificación discutido es referente a la planificación de la pila de objetivos
  - Empuja sub metas y acciones a una pila; selecciona una acción solamente cuando todas las precondiciones están satisfechas.
  - Funciona en descripciones de estado donde siempre es consistente para planes crecientes, y descripciones de metas para el crecimiento del árbol de búsqueda.
- La planificación de la pila de objetivos esta incompleta, en el sentido de que puede terminar sin encontrar ningún plan.
- La planificación de la pila de objetivos rompe un conjunto de predicados de meta en submetas individuales e intenta resolverlos individualmente uno por uno.
- Este enfoque es llamado planificación lineal
  - Se refiere al hecho que las submetas se intentan y se resuelven en un orden lineal.
  - Aunque no siempre es posible.

## PLANNING GRAPH AND GRAPHPLAN

Buscaremos soluciones en el diagrama de planificación. En la discusión, nos centraremos en la relación entre el diagrama de planificación y el espacio, y luego un brazo robótico simple que puede tomar y colocar algunos bloques, y tengo una representación mundial de los bloques. El diagrama de planificación está ubicado En su forma más simple, se considera una estructura que incorpora estados gráficos. En términos más simples, el horario es una estructura que combina los estados generados por diferentes acciones. Por lo tanto, la diferencia básica entre el espacio de estados del problema de planificación y el gráfico del plano correspondiente es que el espacio de estados se refiere a la operación del estado que genera un nuevo estado. El nuevo estado se denomina estado sucesor, que en adelante se denomina como cada acción vuelve a ser Estado, este nuevo estado se denomina estado sucesor.

## SEQUENTIAL DECISION PROBLEMS

En lugar de buscar soluciones en el espacio de estados o en el espacio de planificación, verá un plano gráfico con un método de planificación completamente diferente. El plano crea una estructura gráfica en la que se representan todas las posibles soluciones. A esto se le llama plan. Producido por diferentes factores. Por lo tanto, la diferencia básica entre el espacio de estados del problema de planificación y el gráfico del plan correspondiente es que el espacio de estados representa las operaciones realizadas en el estado que genera el nuevo estado, que se llama estado sucesor, y después de eso, cada acción se aplicará nuevamente. El estado de seguimiento, este es en realidad el punto de partida para una mayor exploración.

## MAKING COMPLEX DECISIONS

en una situación de toma de decisiones no existe una opción obviamente correcta. En cambio hay respuestas que justifican una mayor experimentación antes de comprometerte con un solo enfoque

### *INTRODUCTION TO MACHINE LEARNING*

El cálculo situacional permite que los comportamientos basados en el conocimiento infieran las consecuencias de las acciones, logrando así una representación puramente del cálculo situacional objetivos.

En su forma más simple, el diagrama de planta se considera como una estructura que combina los estados generados por diferentes operaciones. Por lo tanto, la diferencia básica entre el espacio de estados del problema del plan y el diagrama de plan correspondiente es que el espacio de estados genera nuevos estados en el estado. El estado de la acción del estado se

llama sucesor. Desde entonces, cada acción es aplicable al estado posterior, que en realidad es el punto de partida para una mayor exploración. Por otro lado, el gráfico del plan incorpora los estados producidos por diferentes acciones aplicables, por lo que tengo un conjunto de proposiciones expresadas como una sola capa. Ahora puedes ver que antes de este nivel de propuesta, habrá un conjunto de acciones que conduzcan a este nivel.

## LEARNING DECISION TREES

En el aprendizaje supervisado, al elemento de aprendizaje se le da el valor correcto (o aproximadamente correcto) de la función de entradas particulares, y cambia su representación de la función para tratar de calcular la información proporcionada por la retroalimentación.

### Inferencia inductiva pura.

Dada una colección de ejemplos de  $f$ , devuelva una función  $h$  que se aproxime a  $f$ . La función  $h$  se llama hipótesis.

### **Supuesto fundamental**

Supuesto: la distribución de los ejemplos de formación es idéntica a la distribución de los ejemplos de prueba (incluidos los ejemplos futuros no vistos).

- En la práctica, estas suposiciones a menudo se violan hasta cierto punto.
- Las violaciones graves claramente darán como resultado una clasificación deficiente.
- Para lograr una buena precisión sobre los datos de la prueba, los ejemplos de entrenamiento deben ser suficientemente representativos de los datos de la prueba.

### **árboles de decisión de aprendizaje**

El aprendizaje del árbol de decisiones es una de las técnicas de clasificación más utilizadas.

- Su precisión de clasificación es competitiva con otros métodos;
- Es muy eficaz.

- Sirve como una buena introducción al área del aprendizaje inductivo.
- El aprendizaje del árbol de decisiones utiliza un árbol de decisiones (como modelo predictivo) para pasar de observaciones sobre un elemento (representado en las ramas) a conclusiones sobre el valor objetivo del elemento (representado en las hojas).

## LINEAR REGRESSION

### Variable independiente y dependiente

- Variable independiente:
  - Variable cuyo valor no cambia por efecto de otras variables y se utiliza para manipular la variable dependiente. A menudo se denota por  $X$ .
- Variable dependiente:
  - Una variable cuyo valor cambia cuando hay alguna manipulación en los valores de la variable independiente. A menudo se denota como  $Y$ .
- Las variables explicativas se denominan variables independientes y las variables a explicar se denominan variables dependientes.

### Proceso de estimación

- Nuestro objetivo es conocer los parámetros del modelo que minimizan el error en las predicciones del modelo.
- Para encontrar los mejores parámetros:
  1. Defina una función de costo, o función de pérdida, que mida cuán inexactas son las predicciones de nuestro modelo.

2. Encuentre los parámetros que minimizan la pérdida, es decir, hacen que nuestro modelo sea lo más preciso posible.

Descenso gradiente: aprende los parámetros

El objetivo del descenso de gradiente es encontrar el mínimo de la función de pérdida de nuestro modelo obteniendo iterativamente una mejor y mejor aproximación de la misma.

## SUPPORT VECTOR MACHINES

### **Máquina de vector de soporte lineal**

- Para maximizar el ancho de la separación, es necesario minimizar  $w$ , sin dejar de respetar las restricciones con respecto a los valores en los bordes de la separación.
- Un posible enfoque para encontrar el mínimo es utilizar el método ideado por Lagrange.
- La maximización del ancho está asegurada si se minimiza lo siguiente, respetando las restricciones en los valores de los bordes.

### **Clasificador de margen máximo**

- Formalizar la noción del mejor separador lineal.
- El mejor hiperplano es el que representa la mayor separación, o margen, entre las dos clases: se maximiza la distancia desde él hasta el punto de datos más cercano en cada lado.
- si existe tal hiperplano, se conoce como el hiperplano de margen máximo y el clasificador lineal que define se conoce como clasificador de margen máximo

### **Multiplificadores lagrangianos**

- Manera de convertir un problema de optimización restringido en uno más fácil de resolver



## Kernels

- Proyectar datos en un espacio de dimensiones superiores los hace linealmente separables.

## UNSUPERVISED LEARNING

### Aprendizaje supervisado frente a aprendizaje no supervisado

- Aprendizaje supervisado: descubre patrones en los datos que relacionan los atributos de los datos con un atributo objetivo (clase).
  - Luego, estos patrones se utilizan para predecir los valores del atributo objetivo en instancias de datos futuras.
- Aprendizaje no supervisado: los datos no tienen ningún atributo objetivo; descubrir la estructura subyacente a los datos.
  - Explore los datos para encontrar algunas estructuras intrínsecas en ellos.
  - El aprendizaje no supervisado se trata más de actividades creativas: exploración, comprensión y refinamiento.

### Agrupación

- La agrupación en clústeres es una técnica para encontrar grupos de similitud en los datos, llamados grupos, es decir,
  - Agrupa las instancias de datos que son similares (cercanas) entre sí en un clúster y las instancias de datos que son muy diferentes (lejanas) entre sí en diferentes grupos.
  - En los problemas de agrupamiento, no se especifican las etiquetas de clase; sólo se conocen los vectores de características que representan diferentes objetos / instancias / registros o situaciones.
- La agrupación a menudo se denomina una tarea de aprendizaje no supervisada ya que no se dan valores de clase que denoten una agrupación a priori de las instancias de datos, que es el caso del aprendizaje supervisado.

## **Métodos básicos de agrupación en clústeres**

- Hay muchos algoritmos agrupados. Es difícil una categorización precisa de los métodos de agrupamiento.
- Los principales métodos de agrupación fundamentales incluyen:
  1. Agrupación en clústeres particional
  2. Agrupación jerárquica
  3. Agrupación espectral
  4. Agrupación en clústeres mediante mapas autogestionados