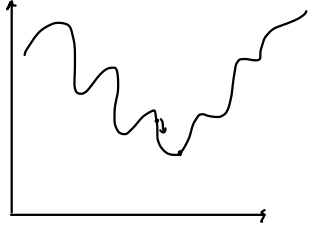


Incluye una jerarquía de componentes. Memoria de casos es pequeña para encontrar subconjuntos adecuados.
Encontrar en qué punto del modelo entran.

Están conectados de una manera diferente. Atacar tarea principal.
Tareas son diagnóstico, reparación y verificación.

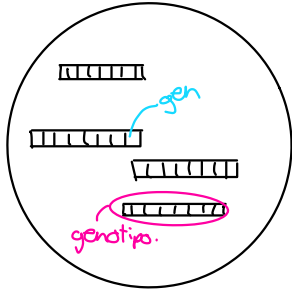
08/abril /19

Algoritmos Evolutivos.



- Búsqueda heurística tipo "Hill Climbing"
- **Función de aptitud:** Función heurística (fitness function)
- Búsqueda heurística paralela y estocástica tipo "Hill climbing".

Se basa en una población de posibles soluciones. Soluciones descritas en una cadena genética.



Genotipo: Soluciones, se denominan así por genética. Es la cadena de genes.

Gen: Cada elemento de la cadena de genes.

Cada instancia tiene un valor propio para cada gen.

Ejemplo: Genotipo: edificio, Genes: #pisos, escaleras...

Genotipo puede tener el número de genes necesarios. Algunos son esenciales para ciertos dominios, pero para otros dominios no, por lo que el número de genes puede ser el mismo siempre o variar.

Búsqueda en paralelo implica cada vez acercarnos más al mínimo. Se modifica quienes están dentro de la población.

El contenido evoluciona iterativamente y mejora.

Operadores genéticos proponen nuevas posibles soluciones y se determina si son mejores o no.

Los operadores genéticos principales:

- Cruzamiento
- Mutación

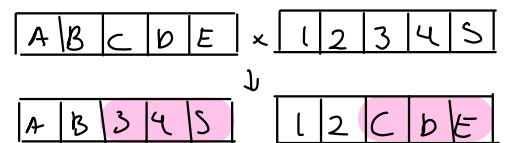
Cruzamiento: Se toman 2 individuos aleatoriamente:

Producir algo diferente con los genes de cada

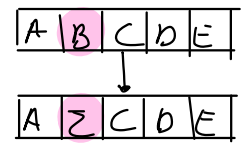
Genotipo. Es decir se usa material genético de

los padres. Cada uno tiene su fitness, su valor

de calidad. Si es baja la calidad se pueden descartar o reemplazan a otros genotipos. Reemplazo parcial, no completo.



Mutación: Se cambia un gen, eligiendo aleatoriamente por otro elemento



Criterio de Convergencia (PARO)

Elegir tamaño población y otras cosas muchas veces es prueba y error

Algunos parámetros que pueden variar:

- ¿se distingue o no entre genotipo y fenotipo?

Fenotipo: manifestación física del genotipo.

- Representación del Genotipo:

* Estructura (lineal, matricial, Jerárquica, ...)

* Tipo de valor(es) permitido(s) en los genes

* ¿Semántica de cada gen asociada a posición o no?

* ¿Genes divisibles o no?

* ¿Tamaño / Longitud variable o no?

- Operadores Genéticos disponibles?

* Cruzamientos:

• ¿de un punto o múltiples puntos de cruce?

• ¿Cómo elegir los padres?

• ¿Punto de cruce igual en los dos padres o no?

• ¿Punto de cruce a la mitad de un gen o no?

* Mutación:

• Circularidad: si solo se cambian gen o no

• ¿Usa conocimiento del dominio o no?

* ¿OTROS?

- Población:

* Semántica de cada individuo

* ¿Tamaño variable o no? sugerencia: no, por memoria.

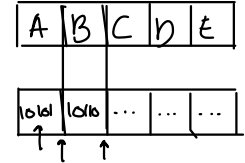
* Reemplazo parcial o completo de la población (parcial = elitismo)

- Convergencia:

* ¿Basado en número de generaciones, valor de aptitud mínimo/máximo. Individual, valor de aptitud mínimo/máximo promedio. Variedad genética, alguna combinación de los criterios anteriores, o...

- ¿Multiobjetivo o no?

- ¿Función de aptitud.



$A \rightarrow B \quad B \rightarrow C \quad C \rightarrow D$

$B \rightarrow D \quad D \rightarrow E \quad E \rightarrow C$

Algoritmos Evolutivos.

Algoritmos Genéticos
(Genotipos discretos (binarios), lineales, énfasis en cruzamiento,

Programación Genética.
(Genotipos jerárquicos (simbólicos), aplicación específica: escritura automatizada de problemas, Población de posibles soluciones)

Estrategias evolutivas
(Genotipos lineales con valores reales, énfasis en mutación, población de posibles soluciones)

Otros (Por ejemplo: inteligencia de enjambres)
(Población de agentes que buscan soluciones)

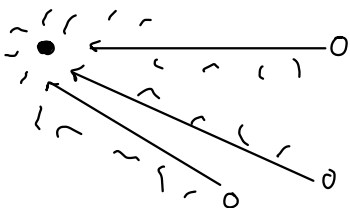
10 / abril / 19

Sistemas Multi-agente:

- Varios agentes inteligentes simulados.
- Cada agente tiene su "personalidad" (atributos) y "comportamiento" (método)
- Normalmente los agentes no son demasiado complejos
- Puede haber varios agentes del mismo tipo o de tipos distintos.
- El punto clave son las interacciones entre los agentes.
 - Cooperativas
 - Competitivas
 - Pueden ser de los dos tipos / variables.
- Las interacciones pueden ser:
 - Directas (comunicación entre los agentes)
 - Indirectas (observación de otros agentes + inferencias sobre sus metas / motivaciones)
- Como resultado de los comportamientos individuales de los agentes y (b) las interacciones entre ellos pueden emerger comportamientos a nivel macro / sociedad (de forma distribuida descentralizada)

Ant Colony, Optimization, Algorithms.

- Cada hormiga individual sigue las reglas:
 - 1) si encuentras comida, levántala y regresa al nido.
 - 2) si encuentras una pista de feromonas, siguela.
 - 3) en cualquier otro caso camina aleatoriamente.
- } Dejar una gota de feromona en cada lugar que visitaste.



Tercer Proyecto de Programación

- Proyecto escrito y presentación oral.
- Router
- Sistema multiagente para simular un Safari.
- Algoritmo heurístico de propagación de restricciones para la resolución de problemas.

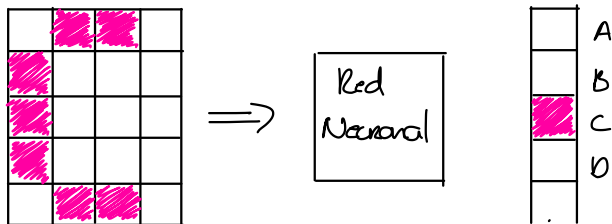
22 / abril / 19

Aprendizaje Automatizado (Machine Learning)

- Se pueden aprender muchos tipos de cosas:
 - Hechos o conceptos aislados.
 - Relaciones entre hechos / conceptos incluidas las jerarquías.
 - Generalizaciones / Patrones.
 - Analogías.
 - Comportamientos → Aumenta la eficiencia del razonamiento *Speed up learning.*
- Se puede aprender de distintas maneras:
 - Aprendizaje supervisado (e.g. viendo ejemplos, recibiendo explicaciones).
 - Aprendizaje no supervisado (e.g. buscar patrones en el conocimiento que ya se tiene → descubrimiento). *↳ minería de datos.*
- Dos tipos de problema que deben tratar de resolver los algoritmos de aprendizaje:
 - Problema de la utilidad ("*utility problem*"): ¿vale la pena aprender algo tomando en cuenta el "trade-off" entre cantidad/calidad del conocimiento y la eficiencia del razonamiento?
 - Asignación de crédito o culpa ("*credit or blame assignment*"): Decidir que factores son responsables del éxito o fracaso.

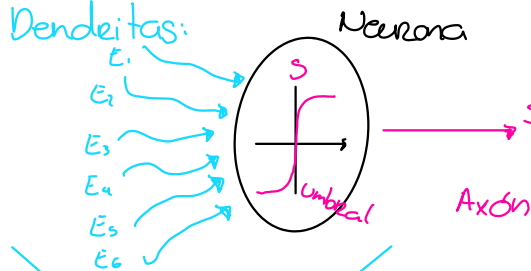
Ejemplo 1: Redes neuronales.

- Aprenden asociaciones del conjunto total de ejemplos.



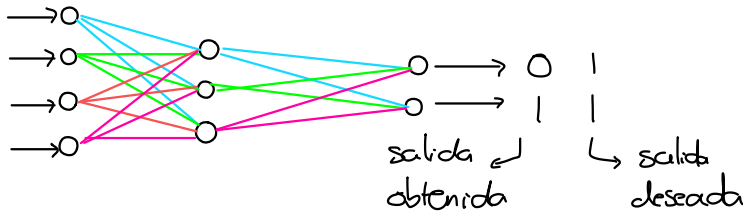
- Se entrenan mostrandoles ejemplos
- Recuerdan ejemplos anteriores aún después de ver ejemplos nuevos.
- Poco a poco aprenden una generalización que cubre todos los ejemplos a través de modificar los pesos de sus conexiones
- No son muy sensibles al ruido.

Dendritas:



$$s = \begin{cases} 0 & \text{si } \sum_{i=1}^n P_i E_i < \text{umbral} \\ 1 & \text{si } \geq \text{umbral} \end{cases}$$

Capa entrada Capa intermedia Capa salida



Algoritmo de retropropagación.

One-hot encoding (solo un bit de salida prendido a la vez)

GAN: Generative Adversarial Network. } Generan salidas no existentes anteriormente. Ideas nuevas.

Ejemplo 2: "Concept Learning" Aprendizaje de conceptos / Generalizaciones.

Temperatura Ambiente	Cantidad de viento	Fin de semana / Día no laborable	Contingencia Ambiental	Espacio de Instancias 3 variables } $2^3 = 8$ 2 valores/var } Tamaño del espacio de instancias
Alta	Alta	Si	No	
Alta	Baja	Si	No	
Baja	Alta	Si	No	
Baja	Baja	Si	Si	
Alta	Alta	No	Si	
Alta	Baja	No	Si	
Baja	Alta	No	Si	
Baja	Baja	No	Si	

← variable de Salida

- Queremos mostrar solo algunas instancias y obtener una sola descripción general. (Representada como una conjunción de restricciones)
- Para esto hay que hacer una búsqueda dentro de un espacio de hipótesis (donde cada estado es una descripción general distinta/alternativa/posible)

Cada restricción (que forma parte de la hipótesis) es de la forma:

- $Var = Val$: (Alguna var. tiene que tener cierto valor específico)
- $var = ?$: (Don't care, no importa que valor tome esa variable)
- $var = \emptyset$: (Ningún valor es válido para esa variable)

Tamaño del espacio de hipótesis: $4^3 \rightarrow \text{variables} = 64 \rightarrow \# \text{ total de hipótesis}$ sintácticamente diferentes
 \hookrightarrow Forma que puede tomar cada restricción.

$(?, \emptyset, \text{si}) \Rightarrow \text{si}$
 $(\emptyset, ?, \text{si}) \Rightarrow \text{si}$

- La hipótesis más general posible: $(?, ?, ?)$
- La hipótesis más específica posible: $(\emptyset, \emptyset, \emptyset)$
- La "mejor" hipótesis tendrá (probablemente) un nivel de generalización intermedio entre estas 2.

Algoritmo:

- Inicializar h para que sea la hipótesis más específica posible.
- Para cada ejemplo de entrenamiento positivo x :
 - Para la restricción asociada con cada atributo en y :
 - * Si la restricción es satisfecha por x
No hacer nada
 - * En caso contrario:
Reemplazar en h la restricción por la que este más general que si sea satisfecha por x .
- Regresa h .

$h = (\emptyset, \emptyset, \emptyset) \xrightarrow{④} (\text{Baja}, \text{Baja}, \text{si}) \xrightarrow{①} (\text{Baja}, \text{Baja}, \text{si}) \xrightarrow{⑦} (\text{Baja}, ?, ?)$
Temperatura = baja and viento = ? and fin de semana = ? \Rightarrow Contingencia.

$h = (\emptyset, \emptyset, \emptyset) \xrightarrow{②} (\emptyset, \emptyset, \emptyset) \xrightarrow{③} (\text{Baja}, \text{Baja}, \text{no}) \xrightarrow{④} (\text{Baja}, ?, \text{no})$

Niveles de generalidad diferente. Si importa el orden de entrada de los ejemplos.

24/abril/19

Procesamiento de lenguaje natural

- Varios modalidades en la que se expresan los lenguajes:
 - Voz
 - Texto (ASCII)
 - Escritura (Caligrafía)

$\left. \begin{array}{l} \text{Oraciones} \\ \text{Párrafos} \\ \text{Textos completos.} \\ \text{Palabras} \\ \text{(Varios niveles jerárquicos)} \end{array} \right\}$
- Dos direcciones
 - Entendimiento
 - Generación.

$\left. \begin{array}{l} \text{Traducción automatizada.} \end{array} \right\}$ **ELIZA:** Simulaba un psiquiatra rogeriano.
- Características de los lenguajes que causan dificultades:
 - Ambigüedades.
 - Información oculta/implícita
 \rightarrow cómo que como como cochino/chancho/morcuno?
 - Excepciones

→ $O \rightarrow S \vee P = ¡Hola!$

Ejemplos: La caja de Kleenex está vacía.
Cobré micheque en la caja

El cobra le dió una paliza a Blue Demon
El cobra su sueldo el día 15 de cada mes

Las ideas verdes incoloras duermen furiosamente → sin sentido

Niño compear sandwich jamón en la changareros. → mal escrito

Al astredromo le gustó la estrella → no se entiende con precisión

María nació en 2 de octubre
El cumpleaños de María es el 2 de octubre. → son iguales

BORIS: Proyecto que duró 11 años. Fue tesis doctoral y solo es capaz de contestar preguntas acerca de 2 historias.

El planeta estaba siendo observado por el extraterrestre; él era café
↳ ambiguo

Conclusión: Se necesita conocer (implícita o explícitamente) y tomar en cuenta:

- Léxico (Vocabulario)
- Semántica (significado)
- Sintaxis (gramática)
- Puntuación / Acentuación / Pronunciación.
- Variantes válidas de todas estas cosas
- Contexto.

29 / abril / 19

Domingo 12 de abril se entrega proyecto y reporte escrito y presentación.

Visión por computadora.

Que pasa si en fondo blanco se forman líneas negras que forman figuras geométricas
Buscar espigas. Reglas dependiendo la forma.

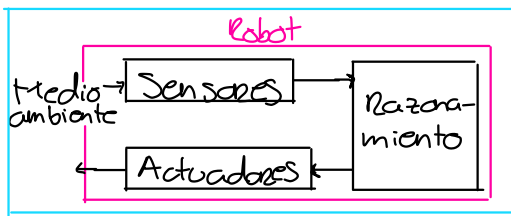
Procesar imágenes 18 configuraciones distintas que dan a cabo a diferentes inferencias.

David Marr: Vision (años 70): Libro de inicios de visión por computadora.

Robótica.

Tiene un poco de machine learning, procesamiento de lenguaje natural, visión por computadora para sensor, planean rutas, etc. Utilizan todo lo visto en el curso.

Robótica asume que todo ya esté funcionando. Enfócate por parte del razonamiento.



- Enfoques:

- Deliberativo: enfatiza el razonamiento, análisis, procesamiento detallado de la información proporcionada por los sensores así como el conocimiento que tienen para tomar decisiones y planear cuidadosamente las acciones.

Requiere un mapa/modelo completo, detallado del medio ambiente.

Shakey: Robot que cruzaba un salón en Stanford.

Enfoque muy bueno para ambientes estáticos y conocidos.

- Reactivo: enfatiza una conexión directa entre sensores y actuadores (modelando "intuiciones", "reflejos" pre-programados), minimizando lo más posible el razonamiento (y la necesidad de almacenar mucho conocimiento).

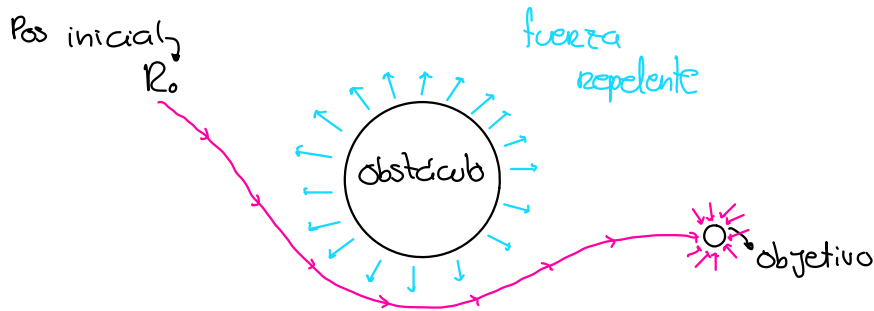
Enfoque muy bueno para ambientes dinámicos y desconocidos.

- Híbrido: combina aspectos / ventajas de los dos enfoques extremos.

Ejemplo: comenzar sin mapa o con un mapa básico / parcial y luego aprender más detalles.

Un enfoque híbrido: arquitectura propuesta por Ron Arkin de Georgia Tech (basado en comportamientos).

- Si se combinan los comportamientos evita-obstáculos y dirígete-hacia-objetivo.



Fusión de sensores: creación de un solo mapa (aunque sea aproximado) basándose en la info que proporcionan múltiples sensores del mismo tipo o tipos distintos.