

“A definition of AI Main capabilities and scientific disciplines”,
High Level Expert Group on Artificial Intelligence, European Commission, April 2019

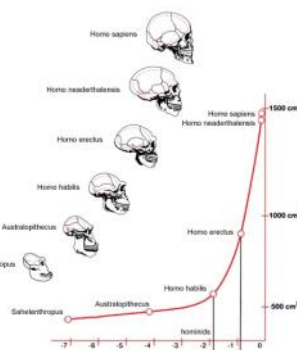
NEW NAVY DEVICE LEARNS BY DOING

Dr. Frank Rosenblatt, designer of the Perceptron, conducted the demonstration. He said the machine would be the first device to think as the human brain. As do human be-

Mr. Rosenblatt said in principle it would be possible to build brains that could reproduce themselves on an assembly line and which would be conscious of their existence.

The first Perceptron will have about 1,000 electronic "association cells" receiving electrical impulses from an eye-like scanning device with 400 photo-cells. The human brain has 10,000,000,000 responsive cells, including 100,000,000 connections with the eyes.

Be conscious of its existence?



Evolución do tamaño do cerebro en homínidos.

INTRODUCCIÓN E PERSPECTIVA HISTÓRICA (II)

luns, 23 de outubro de 2023 11:15

Elemento de comparación	Cerebro	Computador
Tamaño	Aproximadamente 1.500 cc	720 m ² de superficie ocupada e máis de 100 toneladas de peso (2)
Consumo de enerxía	20 watts (1)	17,6 MW -24 MW se se inclúe a unidade de refrixeración- (2)
Capacidade de almacenamento / memoria	3.500 terabytes (1)	1 petabyte de memoria asociada aos procesadores e 12,4 petabytes de capacidade de almacenamento total (2)
Capacidade de procesamento	2.200 teraflops (1)	33,86 petaflops (2)
Aprendizaxe	Enorme capacidade de aprendizaxe	Aprendizaxe en máquina implementadas via software
Evolución	Poucos cambios nos últimos 100.000 anos	Moi rápida, sobre todo en potencia de cálculo e capacidade de almacenamento
Cálculos matemáticos	Moi lento	Moi rápido
Problemas lóxicos – formalizables algorítmicamente-	Moi lento	Moi rápido
Percepción e acción sobre a contorna	Moi rápido	Moi lento
Razoamento de sentido común	Moi rápido	Moi lento

Comparación *wetware* vs hardware

ENFOQUES DA IA

IA	Como persoas	Racionalmente
Pensamento	Modelado cognitivo	Leis do pensamento lóxico
Comportamento	Test de Turing	Axentes racionais



Test de Turing

PERÍODOS NA EVOLUÇÃO DA IA



Período	Descripción
1940 - 1955	Primeiros pasos prometedores
1956 - 1973	Arranque moi optimista
1974 - 1980	Declive –primeiro "inverno"-
1981 - 1987	Recuperación
1988 - 1993	Estancamento –segundo "inverno"-
1994 - 2010	Relanzamento
2010 -	Ambición e realismo

Alan Turing publicou en 1950 na revista *Mind*: "Computing Machinery and Intelligence", que comeza así: "I propose to consider, **Can machines think?**"



Unde venis, AI?

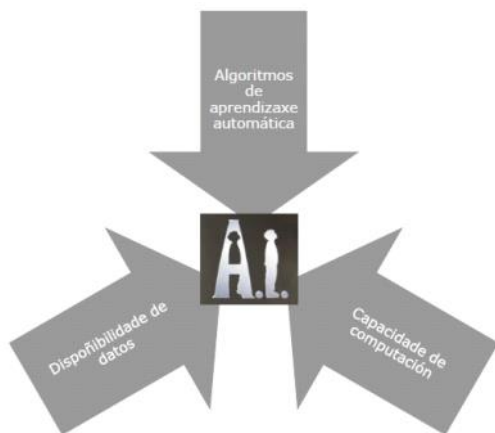


Dartmouth Summer Research Project on AI, 1956

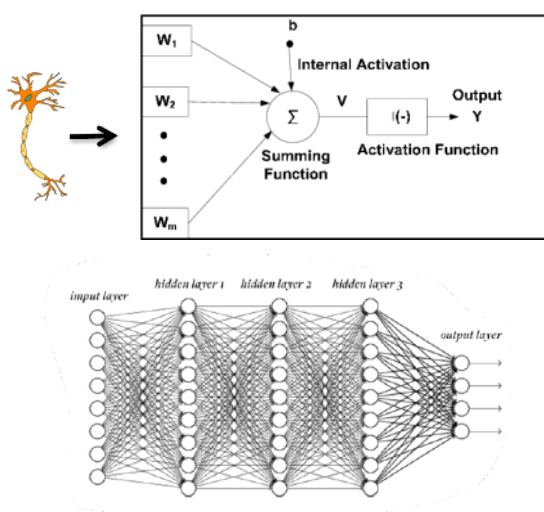
Quo vadis, AI?



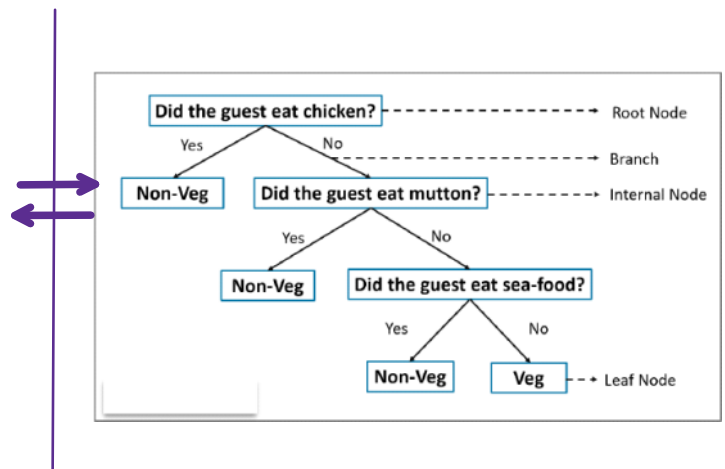
Figure 1. Trenchard More, John McCarthy, Marvin Minsky, Oliver Selfridge, and Ray Solomonoff.



DIFERENTES APROXIMACIÓNS NO TEMPO



Subsimbólica



Simbólica

ORIENTACIÓN CIENTÍFICO-TECNOLÓXICA



IA de propósito específico



IA basada en aprendizaxe [profunda] automática



Non temos un modelo de intelixencia nin de aprendizaxe xeral

REFLEXIÓN SOCIOECONÓMICAS SOBRE A IA

martes, 26 de decembro de 2023

12:31

REVOLUCIÓN INDUSTRIAL



1st Industrial Revolution WATER & STEAM

Steam and water power replace human and animal power with machines.



2nd Industrial Revolution ELECTRICITY

Electricity, internal combustion engines, airplanes, telephones, cars, radio, and mass production.



3rd Industrial Revolution AUTOMATION

Electronics, the internet and IT used to further the automation of mass production.



4th Industrial Revolution CYBER-PHYSICAL SYSTEMS

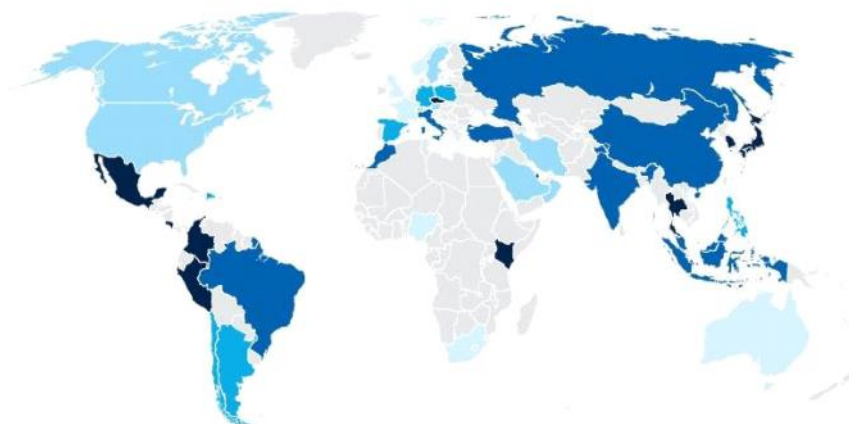
Driverless cars, smart robotics, materials that are lighter and tougher, and a manufacturing process built around 3D printing.

Automatización do emprego

Capacidades requeridas	Tipo de tarefa	
	Non sistemática	Sistemática
Cognitivas	Sistemas de elaboración de noticias	Sistemas expertos para el análisis de riesgos en la concesión de créditos bancarios
Manuales	Robots de exploración en entornos abiertos	Robots para soldadura y montaje en fábricas



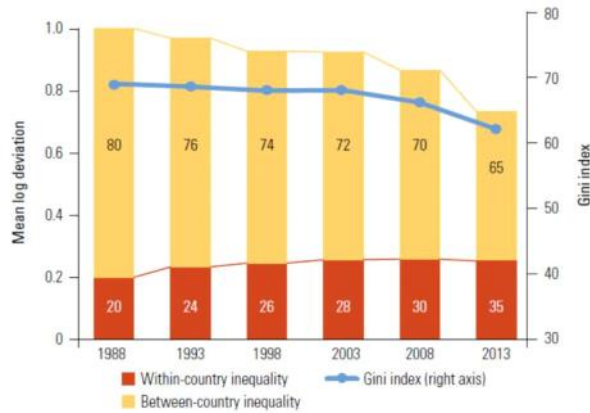
Employee weighted overall % of activities that can be automated by adapting currently demonstrated technologies¹



PARADOXA DA DESIGUALDADE

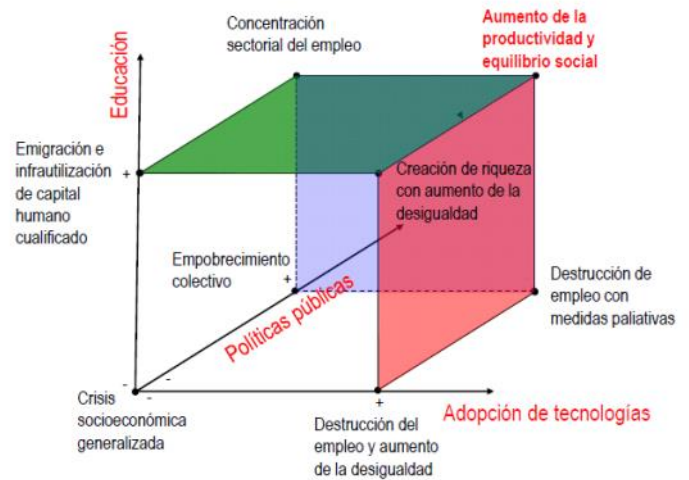
martes, 26 de decembro de 2023

13:06



Que accións tomar?

- Educación
- Adopción intelixente das tecnoloxías intelixentes
- Políticas públicas



É O MESMO DE SEMPRES?

We are being afflicted with a new disease of which some readers may not yet have heard the name, but of which they will hear a great deal in the years to come namely, technological unemployment.

- John Maynard Keynes

Pode agora ser diferente?

Motor tecnolóxico de transformación: IA	Automatización de tarefas de medio e alto nivel cognitivo	Proceso en aceleración
Emprego en sectores que poden acelerar o proceso	Difícil reubicación de traballadores desempregados	Pensamento ilusorio sobre o tema



"SUPERINTELIXENCIA"

martes, 26 de decembro de 2023

13:18

Definamos unha máquina **ultraintelixente** como aquela que pode superar con creces todas as **actividades intelectuais de calquera home** por moi listo que sexa. Posto que o deseño de máquinas é unha desas actividades intelectuais, **unha máquina ultraintelixente podería deseñar máquinas incluso mellores**; entón habería, sen dúbida, unha **"explosión de intelixencia"**, e a intelixencia humana quedaría moi atrás. Por isto, a primeira máquina ultraintelixente é o último invento que o home necesita crear, contando con que a máquina sexa o **suficientemente dócil** como para dicirnos cómo mantela baixo control.

- I.J. God, 1965

AI Act, 2022

Categorías	Ejemplos	Implicaciones	Sanciones
Riesgo inaceptable	Uso de sistemas de clasificación o puntuación social Sistemas de identificación biométrica a distancia "en tiempo real" utilizados por los gobiernos	Prohibición absoluta	Multas de hasta el 6% de los ingresos totales o 30 millones de euros (la cantidad más alta)
Riesgo alto	Infraestructuras críticas, como el suministro de agua, electricidad, gas... o el tráfico rodado Educación (sistemas de admisión o evaluación, por ejemplo) Administración de justicia Dispositivos médicos en general	Evaluación de la conformidad, que garantice el cumplimiento	Multas de hasta el 4% de los ingresos totales o 20 millones de euros (la cantidad más alta). En el caso de que se violen requerimientos asociados a datos las multas serían las mismas que para riesgos inaceptables
Riesgo limitado	Cualquier sistema basado en IA que genere o manipule imágenes, audios o vídeos con apariencia de reales Sistemas que interacciones con personas (por ejemplo, los asistentes conversacionales)	Obligaciones de transparencia: por ejemplo, el usuario debe ser informado de que está tratando con un sistema basado en IA	Multas de hasta el 4% de los ingresos totales o 20 millones de euros (la cantidad más alta)
Riesgo mínimo	Los que no son sistemas basados en IA prohibidos ni de alto riesgo	Queda a criterio del proveedor (de productos, servicios, soluciones...) el cumplir con ciertos códigos de conducta, buenas prácticas, consejos éticos...	No hay multas

BUSCA DE SOLUCIÓN

martes, 26 de decembro de 2023

13:29

Aproximacións á resolución de problemas:

Método	Descrición	Tipo de solución
Preciso	Método específico do problema e que aporta unha solución óptima	Exacta
Heurístico	Utilízase un método de resolución de problemas que aplica coñecemento específico do problema con obxecto de aproximar unha solución ao mesmo ou aumentar a eficiencia na procura dunha solución	Aproximada
Metaheurístico	Melloran os procedementos heurísticos xeralizándoos e obtendo melloras na eficiencia. En xeral utilizan estratexias que diversifican a busca para evitar mínimos locais, intensificándoa en zonas prometedoras	Aproximada

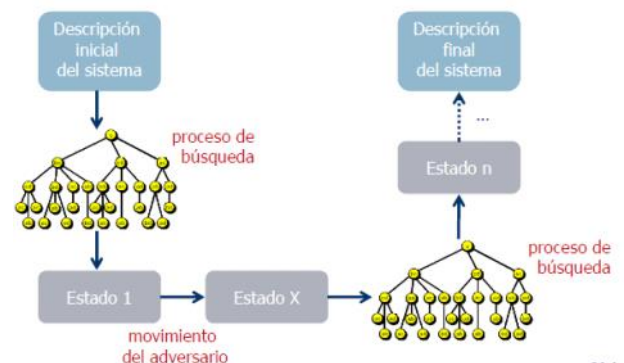


BUSCA EN ESPAZOS DE ESTADOS



Solución: estado **n** ou secuencia de estados de **1** a **n**.

p.e.: Xadrez



- **Espazo de estados:** estados aos que se pode chegar dende o estado inicial mediante calquera secuencia de operadores.
- O proceso de **busca** trata de atopar unha meta ou solución ao problema -ás veces a ruta seguida-.
- **Proba de meta:** permite saber se un estado dado é meta ou solución do problema.
- **Condición de parada** -parcial ou total-: criterio(s) para deter a busca.

Características:

- Imposibilidade de explorar todas as alternativas en **problemas complexos**.
- Se existen **múltiples solucións**, normalmente é suficiente con buscar unha solución aceptable -o mellor é inimigo do bo-.
- **Heurísticas:** criterios para seleccionar un operador ou acción prometedora.

BUSCA A CEGAS (exploración exhaustiva de estados)

martes, 26 de decembro de 2023

14:19

Función: Búsqueda en espacio de estados()

Datos: El estado inicial

Resultado: Una solución

Seleccionar el primer estado como el estado actual

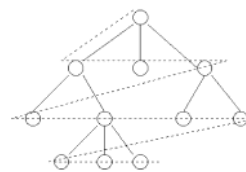
mientras estado actual \neq estado final **hacer**

 Generar y guardar sucesores del estado actual (expansión)

 Escoger el siguiente estado entre los pendientes (selección)

fin

- La selección del siguiente nodo determinará el tipo de búsqueda (orden de selección o expansión)
- Es necesario definir un orden entre los sucesores de un nodo (orden de generación)



Busca en amplitud



Busca en profundidad

Algoritmo de busca en amplitud:

1. Crear lista de nós ABERTA co nó raíz.
2. Ata que ABERTA estea baleira ou se atope nun estado solución, realizar:
 - a) Extraer o primeiro nó de ABERTA (**m**) -FIFO-.
 - b) Expandir **m**. Para cada operador aplicable e cada forma de aplicación:
 - i. Aplicar operador a **m**, obtendo un novo estado e creando un punteiro que permita saber que o seu predecesor é **m**.
 - ii. Se o novo estado é solución, saír e devolver dito estado.
 - iii. Incluír o novo estado ao final de ABERTA.

Algoritmo de busca en profundidad [limitada]:

1. Crear lista de nós ABERTA co nó raíz.
2. Ata que se atope un estado solución ou un erro, realizar:
 - a) Se ABERTA está baleira, erro. Noutro caso:
 - b) Extraer o primeiro nó de ABERTA (**m**) -LIFO-.
 - c) Se a profundidade de **m** é igual a **p**, retroceder a b. Noutro caso:
 - d) Expandir **m** creando punteiros cara el dende os seus sucesores. Introducir os sucesores en ABERTA:
 - i. Se algún sucesor de **m** é solución, devolver camiño da solución.
 - ii. Se algún sucesor de **m** está nun "punto morto", elimínalo de ABERTA.
 - iii. Retroceder a b.

AVALIACIÓN DAS ESTRATEXIAS SEGUIDAS

- Busca **completa**: se existe solución, atópaa.
- Busca **óptima**: atopa a mellor solución, se existe.
- **Complexidade**:
 - **Temporal**: número de nós explorados.
 - **Espacial**: Máximo número de nós en memoria de forma simultánea.
- **Terminoloxía**:
 - **Factor de refracción (r)**: número medio de sucesores dos nós.
 - **Profundidade da solución (p)**.
 - **Profundidade máxima da árbore de busca (m)**.

Profundidade	Nós	Tempo	Memoria
0	1	0,001 mseg	100 Byte
2	111	0,1 mseg	11 KB
4	11.111	11 mseg	1 MB
6	10 ⁶	1 seg	111 MB
8	10 ⁸	100 seg	11 GB
10	10 ¹⁰	0,128 días	1 TB
12	10 ¹²	12,8 días	111 TB
14	10 ¹⁴	3,5 anos	11 PB

Avaliación das estratexias de busca a cegas

martes, 26 de decembro de 2023

15:50

	Amplitude	Profundidade		Preferente por amplitude	Profundidade	Limitada en profundidade	Profundización iterativa	Bidireccional (cando é posible)
Busca completa	Sí	Non, agás en espazos finitos e sen bucles						
Busca óptima	Sí	Non	Tempo	r^p	r^m	r^l	r^p	$r^{p/2}$
Complexidade temporal	$O(r^p)$	$O(r^m)$	Espazo	r^p	r^*m	r^*l	r^*p	$r^{p/2}$
Complexidade espacial	$O(r^p)$	$O(r^*m)$	Óptima?	Si	Non	Non	Si	Si
			Completa?	Si	Non	Si, cando $l > p$	Si	Si

BUSCA HEURÍSTICA

É unha busca con **información potencialmente útil**. De modo xeral, podemos avaliar os nós a través dunha **función de avaliación**:

$$f(n) = g(n) + h(n)$$

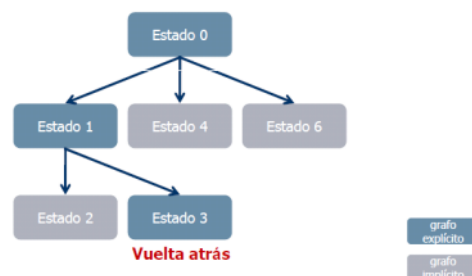
Onde $g(n)$ é o **custo de camiño** dende o nó de inicio e $h(n)$ é unha **función heurística** que lle asigna a cada nó un valor de utilidade estimada na busca dunha solución -en xeral unha distancia a un nó solución-.

- $f(n) = g(n)$, custo uniforme; busca non informada ou **a cegas**.
- $f(n) = h(n)$, algoritmo **primeiro o mellor**.
- $f(n) = g(n) + h(n)$, algoritmo A^* .

BUSCA RETROACTIVA (backtracking)

Segue unha estratexia tentativa con información:

- Só gardamos un nó fillo de cada estado -mantense o camiño dende o estado inicial ata o estado actual-.
- O grafo explícito é unha lista.
- Parada cando chegamos á meta ou non tempos máis operadores que aplicar.



Avaliación das estratexias de busca con heurísticas

	Primeiro o mellor	A^*
Busca completa	Non	Si
Busca óptima	Non	Si, se $h(n)$ é admisible

SISTEMAS BASEADOS EN COÑECEMENTO

martes, 26 de decembro de 2023

17:00

Comparación entre sistemas baseados en coñecemento e en espazos de estados

Sistemas baseados en coñecemento	Busca en espazos de estados
Base de coñecemento	Estados de representación do problema (y heurísticas)
Método de razoamento – aplica o coñecemento dispoñible-	Algoritmo de busca – explora estados na busca dunha solución-
Feitos coñecidos	Espazo de estado inicial
Feitos inferidos	Espazos de estados explorados e estado actual
Resposta a unha consulta	Estado solución



SISTEMAS EXPERTOS

- Nivel de **competencia** equivalente ou superior a un experto humano nun dominio concreto do saber.
- Representación e utilización - razoamento- de **coñecemento explícito**.

Primeiros sistemas expertos

- **Dendral**, *Universidade de Stanford*, 1965-1975; identificación de compostos orgánicos sobre espectroscopía de masas.
- **Mycin**, *Stanford Research Institute*, anos 70; identificación de bacterias causantes de infeccións.

Cándo usar un sistema experto?

- En problemas en **dominios complexos e especializados do saber**.
- En casos nos que **non existe unha aproximación algorítmica ou unha forma de aprender** a resolver o problema mediante aprendizaxe automática.
- En problemas para os que **exista coñecemento** para a súa solución, tanto documentado como aportable por expertos humanos.

Cómo se diseña?

- **Obtención do coñecemento** potencialmente útil sobre o problema e a súa solución por parte dunha "enxeñeira do coñecemento".
- **Representación do coñecemento** dun modo computacionalmente tratable para o seu uso en procesos de "razoamento" en máquinas.



Cómo funciona?

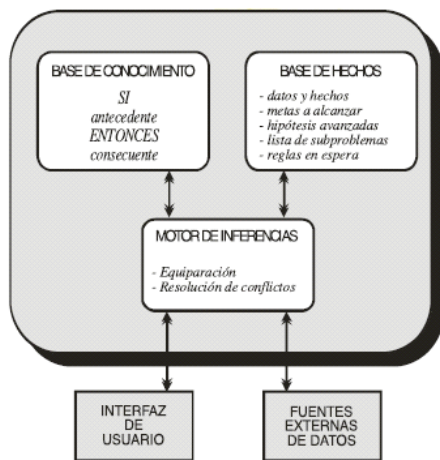
martes, 26 de decembro de 2023 17:27

- **Base de feitos:** o que se sabe sobre o caso que estea sendo considerado dentro da tipoloxía de problemas abordados polo sistema experto.
- **Base de coñecemento:** coñecemento potencialmente útil para a resolución de problemas no dominio específico considerado.
- **Mecanismo de razoamento:** aplica o coñecemento aos feitos para obter unha solución ao problema dado.

Exemplo:

- **Coñecemento:** se a temperatura corporal é superior a 36'5 graos, o paciente ten febre.
- **Feito:** temperatura do paciente é 38'5 graos
- **Conclusión:** "o paciente ten febre"
- **A ter en conta:** os feitos son dinámicos, e tamén pode selo o coñecemento.

ESTRUTURA DUN SISTEMA EXPERTO



Exemplo:

sistema experto para reparación de coches.

SISTEMAS BASEADOS EN REGLAS

- **Regla de produción:**
SE situación ENTÓN acción
- **Situación:** condicións a satisfacer
- **Accións típicas:**
 - Engadir algún feito á memoria de traballo.
 - Suprimir algún feito da memoria de traballo.
 - Executar algún procedemento.

```
Regla 1.    IF coche no arranca,
            THEN comprobar batería
Regla 2.    IF coche no arranca
            THEN comprobar combustible
...
Regla 75.   IF comprobar batería
            AND voltaje batería < 10V
            THEN cambiar batería
...
Regla 120.  IF comprobar combustible
            AND depósito de combustible vacío
            THEN llenar depósito.
```

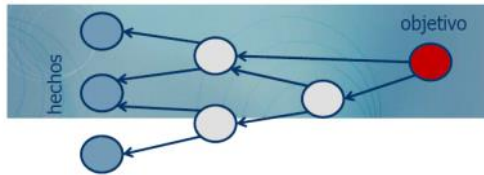
Encadeamento cara adiante:

- 1) Pártese dos feitos na **memoria de traballo**.
- 2) Emparéllanse os feitos cos antecedentes das regras.
- 3) **Aplicanse regras** até que se acada un obxectivo ou non se pode seguir.
- 4) Aplicación de criterios de selección das regras aplicables dentro do "**conxunto conflito**".
- 5) **Principio de refracción:** non aplicación reiterada dunha regra aplicable.

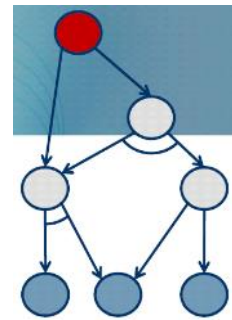
Encadeamento cara atrás:

martes, 26 de decembro de 2023

17:57



Para a representación dos obxectivos a demostrar úsase, normalmente, un grafo E/OU.

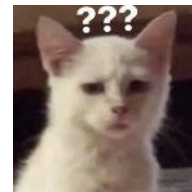


MECANISMO BÁSICO DE RAZOAMENTO

```
1: BH = HechosIniciales;
2: mientras NoVerificaCondiciónFinalización(BH) o NoseEjecutaAccióndeParada hacer
3:   ConjuntoConflicto = Equiparar(BC,BH);
4:   R=Resolver(ConjuntoConflicto);
5:   NuevosHechos = Aplicar(R,BH);
6:   Actualizar(BH,NuevosHechos);
7: fin mientras
```

Verificación de regras

```
1: Verificado=Falso;
2: si Contendida (Meta,BH) entonces
3:   devolver Verdadero;
4: si no
5:   ConjuntoConflicto = Equiparar(Consecuentes(BC),Meta);
6:   mientras NoVacio(ConjuntoConflicto) y No(Verificado) hacer
7:     R=Resolver(ConjuntoConflicto);
8:     Eliminar(R,ConjuntoConflicto);
9:     NuevasMetas=ExtraerAntecedentes(R), Verificado=Verdadero;
10:    mientras NoVacio(NuevasMetas) y Verificado hacer
11:      Meta=SeleccionarMeta(NuevasMetas);
12:      Eliminar(Meta, NuevasMetas);
13:      Verificado=Verificar(Meta,BH);
14:      si Verificado entonces
15:        Añadir(Meta,BH);
16:      fin si
17:    fin mientras
18:  fin mientras
19:  devolver(Verificado);
20: fin si
```



(cara adiante)

```
1: BH = HechosIniciales, ConjuntoConflicto = ExtraeCualquierRegla(BC);
2: mientras NoContendida(Meta,BH) y NoVacio(ConjuntoConflicto) hacer
3:   ConjuntoConflicto = Equiparar(Antecedentes(BC),BH);
4:   si NoVacio(ConjuntoConflicto) entonces
5:     R=Resolver(ConjuntoConflicto);
6:     NuevosHechos = Aplicar(R,BH);
7:     Actualizar(BH,NuevosHechos);
8:   fin si
9: fin mientras
10: si Contendida(Meta,BH) entonces
11:   devolver "éxito";
12: fin si
```

(cara atrás)

```
1: BH = HechosIniciales;
2: si Verificar (Meta,BH) entonces
3:   devolver "éxito";
4: si no
5:   devolver "fracaso";
6: fin si
```


COÑECEMENTO VS. FEITOS

martes, 26 de decembro de 2023

18:26



Feitos	Coñecemento
Específicos –do problema concreto-	De carácter xeral –do dominio-
Dinámicos	Relativamente estáticos
Aumentan durante a resolución do problema	En xeral non aumenta durante a resolución do problema
Necesidade de almacenamento e recuperación eficientes	Necesidade de razoamento eficiente
Búscase que os feitos obtidos directamente do problema abordado sexan precisos e certos	Pode ser impreciso e incerto; polo tanto, tamén os feitos inferidos

SISTEMAS EXPERTOS VS. PROGRAMACIÓN CONVENCIONAL

Programación convencional	Sistemas expertos
Programación imperativa	Programación declarativa
Modificación por reprogramación	Modificación da base de coñecemento
Solución algorítmica	Solución por razoamento baseado en coñecemento
Normalmente solución precisa e certa –solución óptima-	Normalmente imprecisa e con graos de certeza –solución probable, posible...-
Execución guiada polo fluxo de execución do código	Execución guiada polo “motor” de inferencia

CLIPS

C Language Integrated Production System
NASA Johnson Space Center, 1985 - Gary Riley

LÓXICAS

Son a representación formal das relacións existentes entre conceptos, obxectos, propiedades, valores...

Lóxica proposicional

- As **proposicións** son afirmacións verdadeiras ou falsas.
- Únense con **operadores lóxicos**: "e", "ou", "non", "implicación lóxica"
- Utilízanse **mecanismos de razoamento** ou inferencia da lóxica: *modus ponens*, *modus tollens*, ...

Modus ponens:

Se p

e p implica q

entón q

Modus tollens:

Se non q

e p implica q

entón non p

Pazo

Lóxica de predicados

martes, 26 de decembro de 2023

19:02

- Engade a posibilidade de empregar cuantificadores: **para todo** e **existe**.
- **Constantes** (lista de obxectos): *Xosé, María, Xoana...*
- **Predicados** (relacións entre obxectos): *quere-a, pai-de, nai-de...*
- **Coñecemento**: normalmente en forma de regras cuantificadas universalmente:

$$\forall x \forall y \forall z \\ \text{es-padre}(x, y) \wedge \text{es-padre}(y, z) \rightarrow \text{es-abuelo}(x, z)$$

MODELOS DE REPRESENTACIÓN DE COÑECIMIENTO: REDES SEMÁNTICAS



A IMPORTANCIA DO "PROMPTING"

Chain-of-Thought Prompting

Model Input

Q: Roger has 5 tennis balls. He buys 2 more cans of tennis balls. Each can has 3 tennis balls. How many tennis balls does he have now?

A: Roger started with 5 balls. 2 cans of 3 tennis balls each is 6 tennis balls. $5 + 6 = 11$. The answer is 11.

Q: The cafeteria had 23 apples. If they used 20 to make lunch and bought 6 more, how many apples do they have?

Model Output

A: The cafeteria had 23 apples originally. They used 20 to make lunch. So they had $23 - 20 = 3$. They bought 6 more apples, so they have $3 + 6 = 9$. The answer is 9. ✓

(d) Zero-shot-CoT (Ours)

Q: A juggler can juggle 16 balls. Half of the balls are golf balls, and half of the golf balls are blue. How many blue golf balls are there?

A: **Let's think step by step.**

(Output) There are 16 balls in total. Half of the balls are golf balls. That means that there are 8 golf balls. Half of the golf balls are blue. That means that there are 4 blue golf balls. ✓

SISTEMAS CONEXIONISTAS

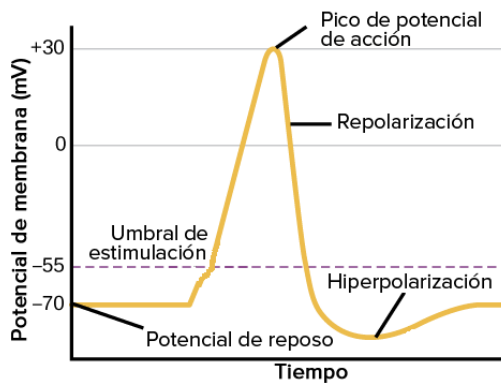
martes, 26 de decembro de 2023 19:21

Nos sistemas conexionistas ou **Redes Neuronais Artificiais (RNA)** apréciase especialmente a dualidade científica e tecnolóxica da IA:

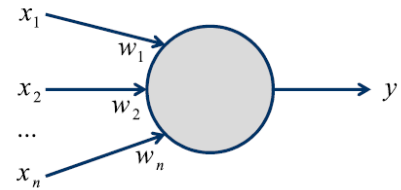
- Bioinspiración
- Tecnoinspiración



Potencial de acción (ideal) dunha neurona



Modelo máis simple de neurona artificial

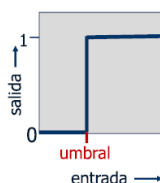


$$y = \sum_i x_i w_i = x_1 w_1 + x_2 w_2 + \dots + x_n w_n$$

Neurona binaria con umbral

[McCulloch & Pitts, 1943]

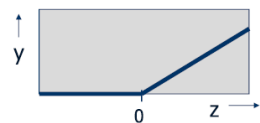
$$z = b + \sum_i x_i w_i$$
$$y = \begin{cases} 1 & \text{si } z \geq 0 \\ 0 & \text{en otro caso} \end{cases}$$



Umbral $\theta = -b$

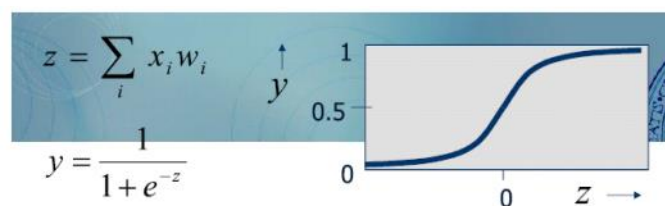
Neurona lineal rectificada

$$z = \sum_i x_i w_i$$
$$y = \begin{cases} z & \text{si } z \geq 0 \\ 0 & \text{en otro caso} \end{cases}$$



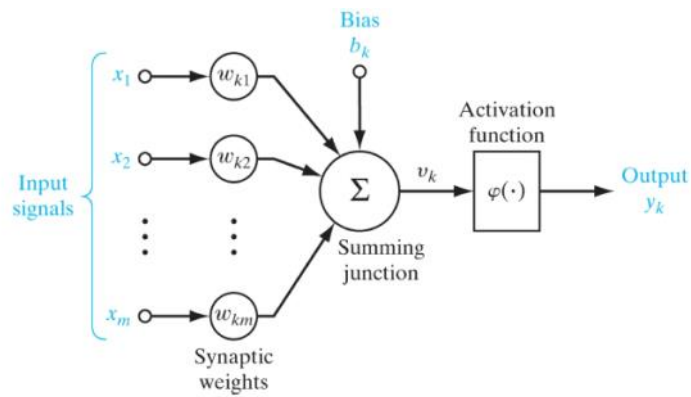
Asumiendo $x_0=1$ y $w_0=b$ (umbral $\theta = -b$)

Neurona sigmoidal

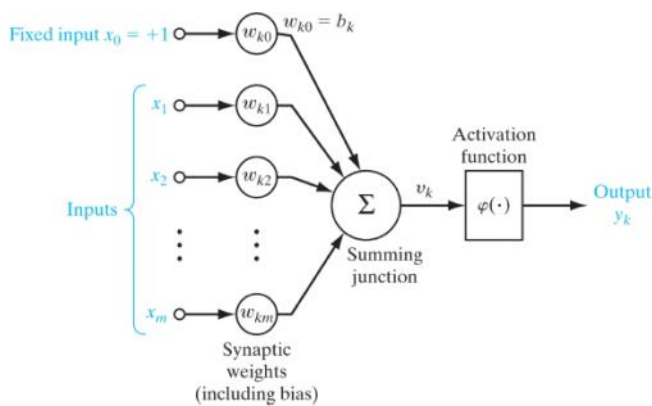


Representación común dunha Neurona Artificial

martes, 26 de decembro de 2023 19:46



Representación común dunha Neurona Artificial con *bias* integrado como peso

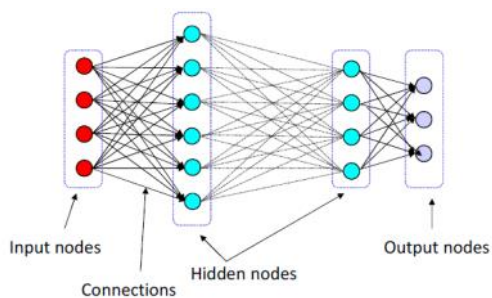


Cronoloxía dos sistemas conexionistas

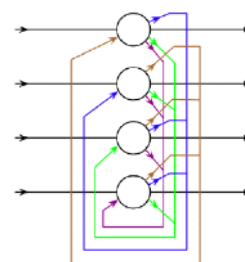
- 1943 Neurona formal de McCulloch e Pitts
- 1957 Perceptrón, Frank Rosenblatt
- 1969 Publicación do libro "Perceptrons", Minsky e Papert
- 1974-1986 Retropropagación (*backpropagation*)
- 1982 Redes de Hopfield
- 2006 Aprendizaxe profunda (*Deep Learning*)



Rede Neuronal Multicapa



Rede de Hopfield



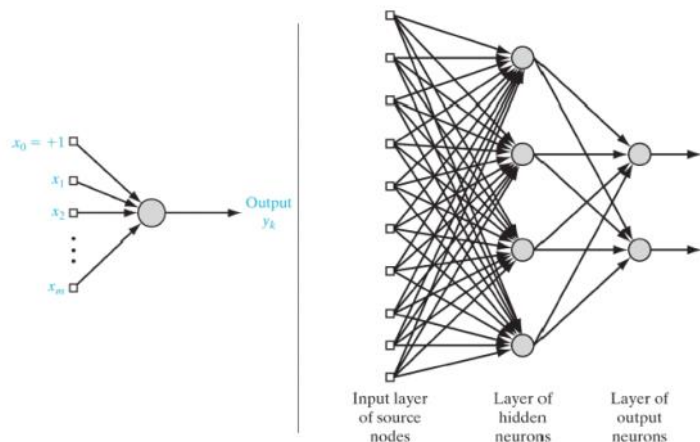
REDES NEURONAIS ARTIFICIAIS (RNA)

martes, 26 de decembro de 2023

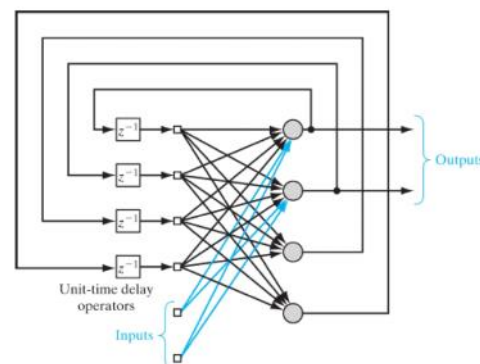
20:25

- **Estrutura** composta por Neuronas Artificiais (NA), profusamente conectadas entre si.
- Diferentes **modelos** ou arquitecturas dependendo do tipo de NA e da súa organización.
- Arquitecturas de **cómputo** paralelo e distribuído.
- **Aprendizaxe** a partir de datos de entrada e, en xeral, mediante modificación de pesos sinápticos.

Perceptrón e arquitectura multicapa cara adiante



Rede recorrente



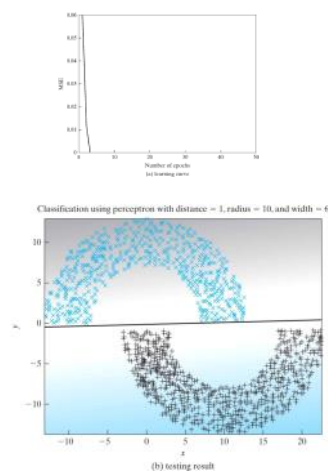
Cómo aprende unha RNA con Perceptróns?

- **Aprendizaxe supervisada** a partir de exemplos -conxunto de adestramento-.
- **Conxunto de adestramento** formado por pares (entrada, saída desexada), suficientemente representativo do problema a resolver.
- **Apréndese axustando os pesos** das entradas dos Perceptróns para que no posible a Rede responda coa saída desexada ante cada exemplo do conxunto de adestramento.

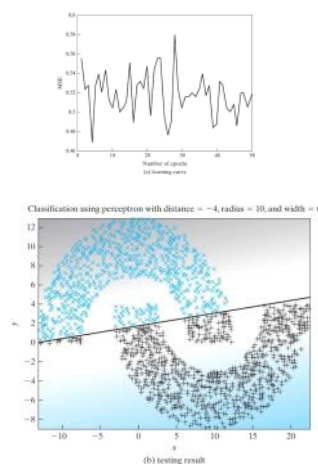
Algoritmo de converxencia do Perceptrón

- **Atopa unha solución se esta existe** -conxunto de pesos que clasifican correctamente o conxunto de entrenamento-.
- Só pode distinguir **categorías linealmente separables**.
- Nunha **arquitectura multicapa** pode discriminar categorías non linealmente separables, pero necesita outro tipo de algoritmos de aprendizaxe.

(clases linealmente separables)



(clases non linealmente separables)



Pazo

APRENDIZAXE AUTOMÁTICA

martes, 26 de decembro de 2023 21:09

"Un programa informático aprende da experiencia E en relación a unha tarefa T empregando unha medida de rendemento P se mellora as súas prestacións, medidas mediante P , na realización á tarefa T a través da experiencia E ." (Tom Mitchell)

Por exemplo: xogo de damas

E = experiencia adquirida mediante o xogo de moitas partidas

T = xogar ás damas

P = probabilidade de que o programa gañe a próxima vez



Importancia e protagonismo da aprendizaxe máquina:

- Facer viable o desenvolvemento de certas aplicacións.
- Abordar unha IA de propósito xeral a través da aprendizaxe, tamén de propósito xeral.
- A potencia de cálculo e a capacidade de memoria, a dispoñibilidade de datos e información e os novos e mellores algoritmos están permitindo un desenvolvemento inédito da Aprendizaxe Automática.

ESTRATEGIAS

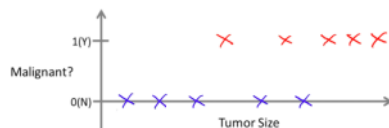
- **Aprendizaxe supervisada:**
Durante a experiencia E indícase cómo se debe realizar a tarefa T .
- **Aprendizaxe non supervisada:**
Durante a experiencia E non se indica cómo se debe realizar a tarefa T .
- **Aprendizaxe por reforzo:**
Durante a experiencia E danse indicios de se se está a realizar a tarefa T ben ou mal.

Aprendizaxe supervisada

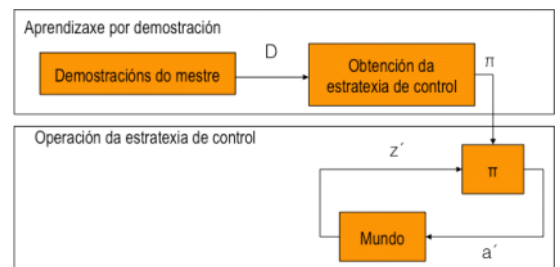
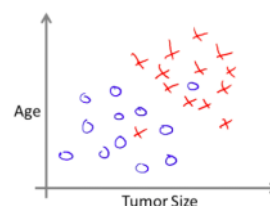
- **Aprendizaxe por demostración:** Robot Baxter
- Problema de **regresión**: aprender a función que mellor represente os datos de adestramento.



- Problema de **clasificación**: aprender o decisor que mellor discrimine os datos de adestramento.



Aumentar o número de características adoita mellorar a aprendizaxe.



Aprendizaxe non supervisada

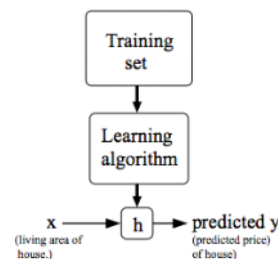
mércores, 27 de decembro de 2023

12:29

- É o tipo de aprendizaxe automática máis empregada despois da supervisada.
- Apréndese sobre datos que non están previamente etiquetados.
- O que se aprende é, polo tanto, cómo agrupalos en función de certos criterios de semellanza ou outros de categorización.

REGRESIÓN LINEAL

- Aprendizaxe **supervisada** dunha **función**.
- Conxunto de adestramento etiquetado.
- Notación:
 - m := número de exemplos de adestramento
 - x := valores de entrada / **características**
 - y := valor de saída / **resposta**
 - (x, y) : par entrada-saída xenérico
 - (x^i, y^i) : iésimo par entrada-saída



Utilización do conxunto de adestramento:

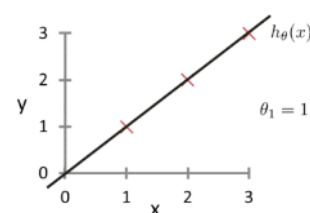
- Pártese dun **conxunto de adestramento**.
- Aplícaselle un **algoritmo de aprendizaxe**.
- O resultado do algoritmo é unha función, que chamaremos **h** ou **hipótese**.
- A **aprendizaxe** consiste en construír **h** de tal modo que aporte o valor de saída **y** asociado á entrada **x** de acordo co especificado no conxunto de adestramento -par (x, y) -.

Cómo representar a hipótese h?

Se, por exemplo:

$$y = h_{\theta}(x) = (\theta_0 + \theta_1 x)$$

- y será unha función lineal de x
- θ_i son os parámetros que definen a recta (neste caso)
 - θ_0 é o seu valor na orixe ($x = 0$)
 - θ_1 é a súa pendente (gradiente)



Función de custo

Necesitamos definir unha **función de custo** que guíe ao algoritmo de aprendizaxe no axuste dos parámetros θ_j que definen a recta que, á súa vez, aproximará o mellor posible os exemplos do conxunto de adestramento:

Trátase de que $h_{\theta}(x)$ se aproxime o posible a y , se o par (x, y) é un exemplo de comportamento na resolución do problema. Basicamente, $h_{\theta}(x)$ é un "imitador de y ". Ademais, debemos avaliar en que medida o é para poder guiar o proceso de aprendizaxe.

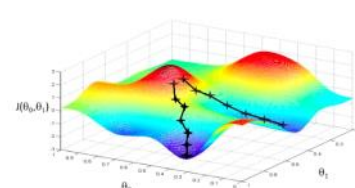
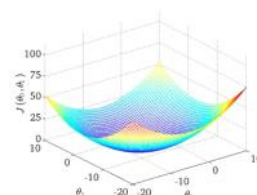
Trátase dun problema de **minimización**:

Minimizar $(h_{\theta}(x) - y)^2$, por exemplo, para cada par de valores de adestramento:

Polo tanto, a función de custo -neste caso, o "**erro cadrático medio**", será:

$$J(\theta_0, \theta_1) = \frac{1}{2m} \sum_{i=1}^m (h_{\theta}(x^{(i)}) - y^{(i)})^2$$

Debemos axustar os valores θ_1 e θ_2 e obter os que reduzan $J(\theta_1, \theta_2)$.
(podería tratarse dun mínimo local)



Pazo

Algoritmo regresión lineal

mércores, 27 de decembro de 2023 13:28

Formalmente, trátase de repetir os seguintes pasos ata que non se poida continuar:

$$\theta_j := \theta_j - \alpha \frac{\partial}{\partial \theta_j} J(\theta_0, \theta_1), \text{ para } j = 0 \text{ y } j = 1$$

$$\text{temp0} := \theta_0 - \alpha \frac{\partial}{\partial \theta_0} J(\theta_0, \theta_1)$$

$$\text{temp1} := \theta_1 - \alpha \frac{\partial}{\partial \theta_1} J(\theta_0, \theta_1)$$

$$\theta_0 := \text{temp0}; \theta_1 := \text{temp1}$$

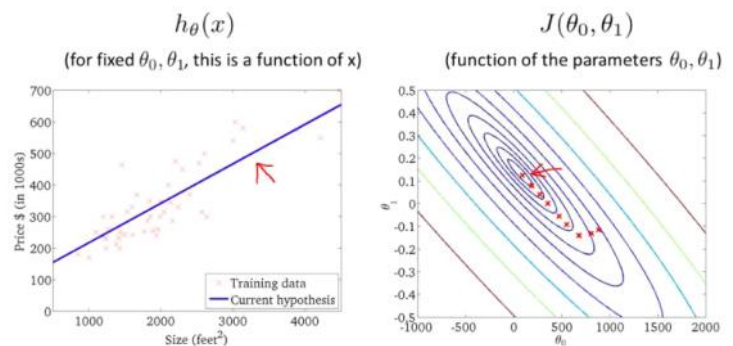


Método de descenso de gradiente:

$$\begin{aligned} \frac{\partial}{\partial \theta_j} J(\theta_0, \theta_1) &= \frac{\partial}{\partial \theta_j} \cdot \frac{1}{2m} \sum_{i=1}^m (h_{\theta}(x^{(i)}) - y^{(i)})^2 \\ &= \frac{\partial}{\partial \theta_j} \cdot \frac{1}{2m} \sum_{i=1}^m (\theta_0 + \theta_1 x^{(i)} - y^{(i)})^2 \end{aligned}$$

$$j = 0: \frac{\partial}{\partial \theta_0} J(\theta_0, \theta_1) = \frac{1}{m} \sum_{i=1}^m (h_{\theta}(x^{(i)}) - y^{(i)})$$

$$j = 1: \frac{\partial}{\partial \theta_1} J(\theta_0, \theta_1) = \frac{1}{m} \sum_{i=1}^m (h_{\theta}(x^{(i)}) - y^{(i)}) \cdot x^{(i)}$$



Descenso de gradiente ata axustar a recta que aproxima a "nube de puntos" de adestramento.

A idea consiste en tomar pasos na dirección oposta á do gradiente; que sigan o descenso máis empinado.

Problemas Multivariable

A aprendizaxe pódese **xeneralizar** para calquera número de características de entrada.

- **n** é o número de características.
- **m** é o número de exemplos de adestramento.
- **Xⁱ** é o vector de características para o i-ésimo exemplo.
- **X_jⁱ** é a característica j-ésima do i-ésimo exemplo.

$$y = \begin{bmatrix} 460 \\ 232 \\ 315 \\ 172 \end{bmatrix} \quad X = \begin{bmatrix} 2104 & 5 & 1 & 45 \\ 1416 & 3 & 2 & 40 \\ 1534 & 3 & 2 & 30 \\ 852 & 2 & 1 & 36 \end{bmatrix}$$

Deste xeito, a **hipótese h** ten agora a forma:

$$h_0(x) = \theta_0 x_0 + \theta_1 x_1 + \theta_2 x_2 + \dots + \theta_n x_n$$

$$h_0(x) = \theta^T X$$

Sendo θ^T o vector trasposto de θ e $x_0 = 1$ por conveniencia.

Método de descenso de gradiente para múltiples variables:

Función de custo:

$$J(\theta_0, \theta_1, \dots, \theta_n) = \frac{1}{2m} \sum_{i=1}^m (h_{\theta}(x^{(i)}) - y^{(i)})^2$$

Axuste de parámetros da hipótese:

$$\text{Repetir } \left\{ \theta_j := \theta_j - \alpha \frac{\partial}{\partial \theta_j} J(\theta_0, \dots, \theta_n) \right\},$$

actualizando simultáneamente los valores para $j = 0, \dots, n$

$$\text{Repetir } \left\{ \theta_j := \theta_j - \alpha \frac{1}{m} \sum_{i=1}^m (h_{\theta}(x^{(i)}) - y^{(i)}) \cdot x_j^{(i)} \right\},$$

actualizando simultáneamente los valores para $j = 0, \dots, n$



Algunhas consideracións prácticas:

xoves, 28 de decembro de 2023 17:35

- É bo que as variables de entrada (características) tomen valores en **rangos semellantes**, xa que facilita a converxencia do algoritmo. Para isto, pódense **normalizar** os valores do seguinte modo:

$$x_j^i := \frac{x_j^i - \mu_j}{s_j}, i = 1, \dots, m; j = 1, \dots, n$$

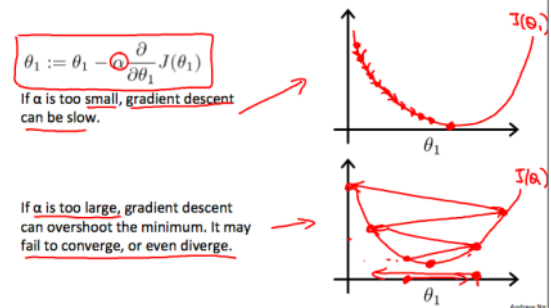
$$\mu_j = \frac{1}{m} \sum_{i=1}^m x_j^i, \quad s_j = \sqrt{\frac{1}{m} \sum_{i=1}^m (x_j^i - \mu_j)^2}$$

Á característica j-ésima de cada un dos m valores do conxunto de adestramento réstaselle o valor medio dos mesmos e divídese pola súa desviación estándar.

- Coeficiente de aprendizaxe α :**

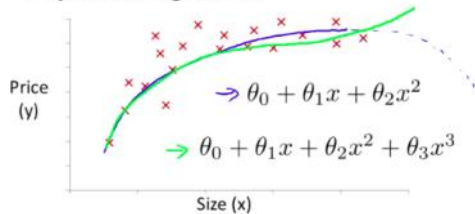
Se o algoritmo funciona adecuadamente debe irse reducindo o valor de $J(\theta)$ en cada iteración.

- Se α é moi pequeno, a converxencia pode ser moi lenta.
- Se α é moi grande, pode non converxer.



REGRESIÓN POLINÓMICA

Polynomial regression



Examples: $m = 4$.

	Size (feet ²)	Number of bedrooms	Number of floors	Age of home (years)	Price (\$1000)
x_0	x_1	x_2	x_3	x_4	y
1	2104	5	1	45	460
1	1416	3	2	40	232
1	1534	3	2	30	315
1	852	2	1	36	178

$X = \begin{bmatrix} 1 & 2104 & 5 & 1 & 45 \\ 1 & 1416 & 3 & 2 & 40 \\ 1 & 1534 & 3 & 2 & 30 \\ 1 & 852 & 2 & 1 & 36 \end{bmatrix}$ $y = \begin{bmatrix} 460 \\ 232 \\ 315 \\ 178 \end{bmatrix}$

$\theta = (X^T X)^{-1} X^T y$

$m \times (n+1)$

m -dimensional vector

Construímos unha función (hipótese) polinómica. Cálculo analítico de θ mediante a ecuación normal.

CÁLCULO ANALÍTICO VS DESCENSO DE GRADIENTE

Descenso de gradiente	Ecuación normal
Necesidad de definir α	No es necesario definir α
Se recomienda escalar las características de entrada	No es necesario escalarlas
Muchas iteraciones	Sin iteraciones
$O(kn^2)$	$O(n^3)$, necesidad de calcular la inversa de $X^T X$
Funciona bien cuando n es grande	Es lento cuando n es muy grande
No tiene este problema	En ocasiones $X^T X$ no es invertible
--	No es aplicable en general a otros algoritmos de aprendizaje

REGRESIÓN LOGÍSTICA

xoves, 28 de decembro de 2023 18:01

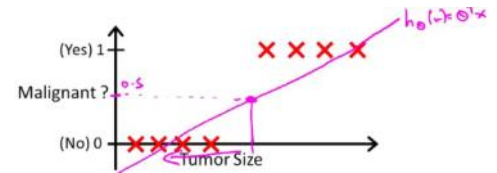
?

Podemos aprender a discriminar entre elementos de dúas ou máis clases?

Por exemplo, fixando un umbral para os valores de h_θ .

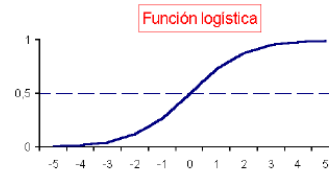
Problemas:

- $h_\theta(x)$ non está de partida acotada entre 0 e 1
- En xeral, a aproximación lineal con umbrais non é un bo clasificador.



Como solución, podemos empregar a función **sigmoide** ou logística para representar h:

$$h_\theta(X) = \frac{1}{1 + e^{-\theta^T X}}$$



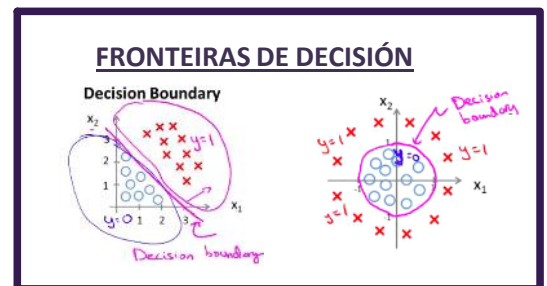
$h_\theta(x)$ é unha "estimación" da probabilidade de que $y = 1$ ante unha entrada dada x .

Función de custo

Sexa a función de custo: $J(\theta) = \frac{1}{m} \sum_{i=1}^m \frac{1}{2} (h_\theta(x^{(i)}) - y^{(i)})^2$

Se: $h_\theta(X) = \frac{1}{1 + e^{-\theta^T X}}$

$J(\theta)$ será non convexa.

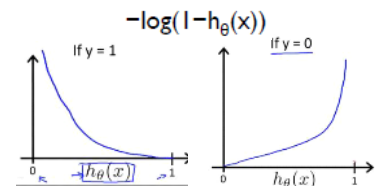


Como alternativa, podemos empregar:

$\text{Cost}(h_\theta(x), y) = -\log(h_\theta(x))$, si $y=1$

$\text{Cost}(h_\theta(x), y) = -\log(1 - h_\theta(x))$, si $y=0$

É dicir, $\text{Cost}(h_\theta(x), y) = -y \cdot \log(h_\theta(x)) - (1-y) \cdot \log(1 - h_\theta(x))$



Agora, a **función de custo** é:

$$J(\theta) = -\frac{1}{m} \sum_{i=1}^m [y^{(i)} \log(h_\theta(x^{(i)})) + (1 - y^{(i)}) \log(1 - h_\theta(x^{(i)}))]$$

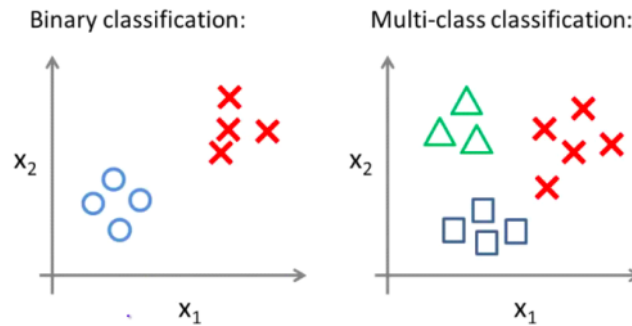
E o **algoritmo de axuste de parámetros** -mediante **descenso de gradiente**:- Repetir $\left\{ \theta_j := \theta_j - \frac{\alpha}{m} \sum_{i=1}^m (h_\theta(x^{(i)}) - y^{(i)}) \cdot x_j^{(i)} \right\}$

Fixémonos en que é igual ca a de regresión lineal, pero con

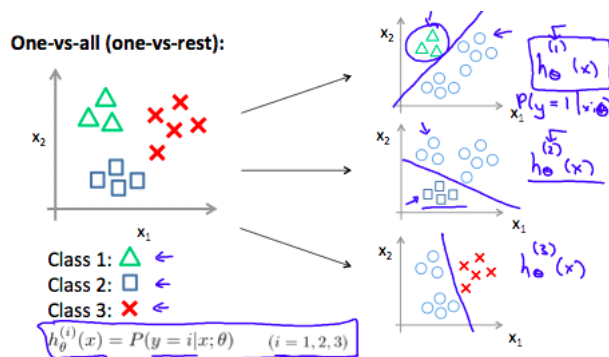
$$h_\theta(X) = \frac{1}{1 + e^{-\theta^T X}}$$

Clasificación multiclase mediante regresión logística

xoves, 28 de decembro de 2023 19:06

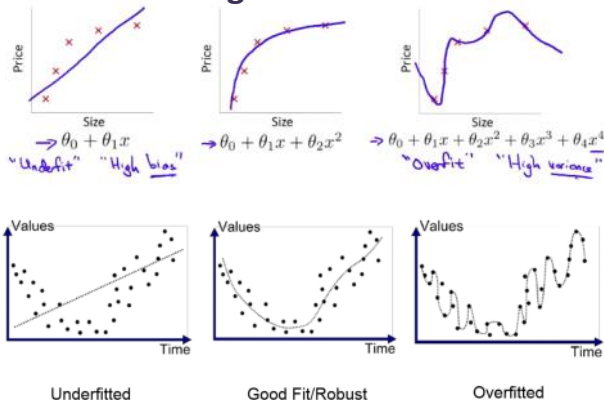


→ Deséñase un clasificador mediante regresión logística para cada clase, asignando cada novo x de entrada á clase para a que $h_{\theta}(x)$ é maior.

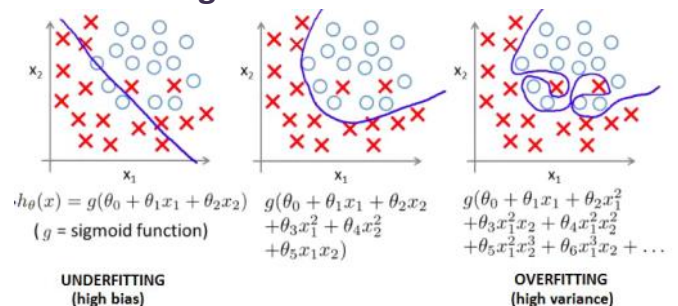


Sobreaxuste

Regresión lineal



Regresión logística



Cómo abordar o sobreaxuste?



- 1) Reducindo o número de características
 - a. Manualmente
 - b. Mediante métodos de selección das características menos relevantes
- 2) "Regularización"
 - a. Mantendo as características, pero reducindo o valor dos parámetros θ



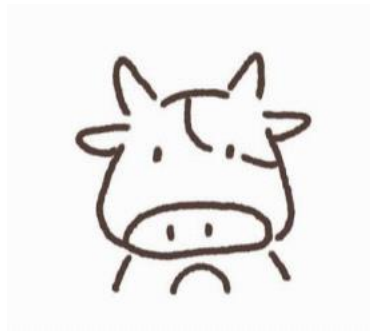
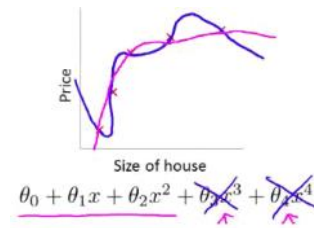
REGULARIZACIÓN EN REGRESIÓN LINEAL

xoves, 28 de decembro de 2023

19:35

Incluimos na función de custo os valores de θ_j ao cadrado, **ponderados** por un factor λ , de valor alto. Isto fai que só os valores de θ_j máis **relevantes** prevalezan.

$$J(\theta) = \frac{1}{2m} \left[\sum_{i=1}^m (h_{\theta}(x^{(i)}) - y^{(i)})^2 + \lambda \sum_{j=1}^m \theta_j^2 \right], \text{ con } \lambda \gg$$



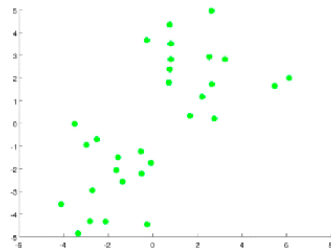
APRENDIZAXE NON SUPERVISADA

xoves, 28 de decembro de 2023

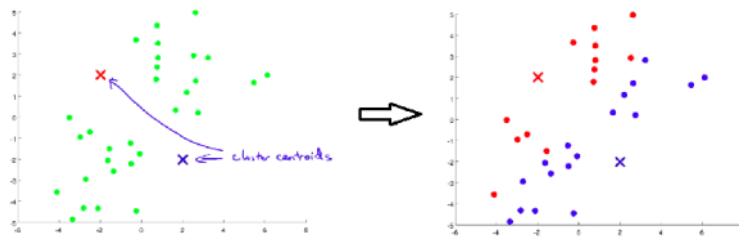
19:50

- Apréndese sobre **datos non etiquetados**.
- É o deseñador o que lle atribúe un significado (**pertinencia**) e un valor (**utilidade**) ao resultado.
- Apréndese **identificando determinadas estruturas** nos datos de adestramento.
- Despois do supervisado é o tipo de aprendizaxe máis empregado.

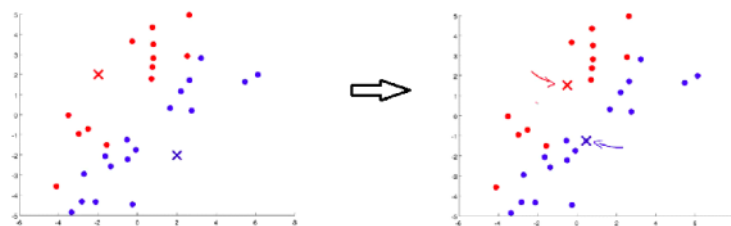
MÉTODO K-MEDIAS



Conxunto de datos de entrada sen etiquetar.



Pártese de dous "centroides" e etiquétanse os datos segundo a súa proximidade a eles.



Despois da (re)asignación dos datos, recálculanse os centroides e repítese o proceso ata a converxencia.

Algoritmo básico K-medias

Entrada:

- K é o número de agrupamentos.
- $\{x^{(1)}, x^{(2)}, \dots, x^{(m)}\}$ é o conxunto de adestramento, con $x^{(i)} \in R^n$

Inicialízanse aleatoriamente os K centroides dos agrupamentos $\mu_1, \mu_2, \mu_3, \dots, \mu_k \in R^n$

```
Repeat {  
  for  $i = 1$  to  $m$   
     $c^{(i)}$  := index (from 1 to  $K$ ) of cluster centroid  
      closest to  $x^{(i)}$   
  for  $k = 1$  to  $K$   
     $\mu_k$  := average (mean) of points assigned to cluster  $k$  }
```

Optimización da función obxectivo

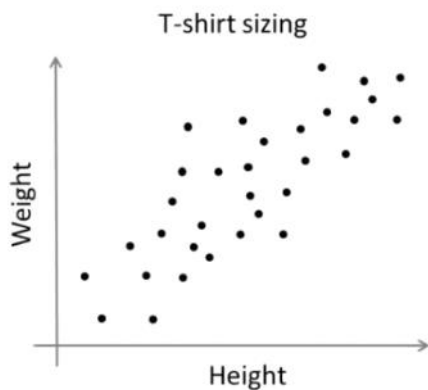
xoves, 28 de decembro de 2023 20:02

- $c^{(i)}$ = índice do agrupamento (1, 2, ..., K) ao que se lle asignou o valor de entrada $x^{(i)}$ [temporalmente].
- μ_k = centroide do agrupamento k ($\mu_k \in R^n$).
- $\mu_{c^{(i)}}$ = centroide do agrupamento ao que se lle asignou $x^{(i)}$ [temporalmente].

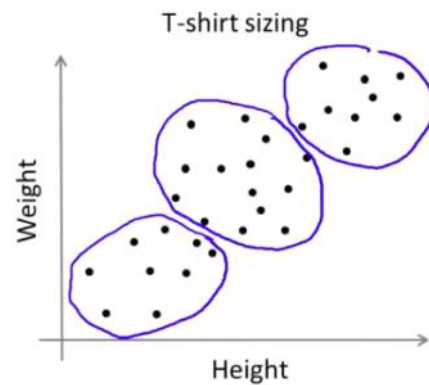
Obxectivo a optimizar:

$$J(c^{(1)}, \dots, c^{(m)}, \mu_1, \dots, \mu_K) = \frac{1}{m} \sum_{i=1}^m \|x^{(i)} - \mu_{c^{(i)}}\|^2$$

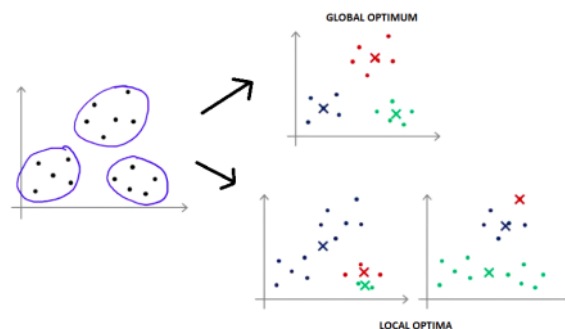
$$\min_{\substack{c^{(1)}, \dots, c^{(m)}, \\ \mu_1, \dots, \mu_K}} J(c^{(1)}, \dots, c^{(m)}, \mu_1, \dots, \mu_K)$$



Conxunto de entrada con agrupamentos non ben diferenciados.



A decisión sobre o número K de categorías pode ser dada a priori -figura- ou calcularse.



Influencia da inicialización aleatoria dos K centroides.

```

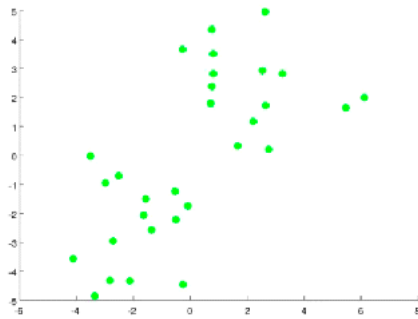
For i = 1 to 100 {
  Randomly initialize K-means.
  Run K-means. Get  $c^{(1)}, \dots, c^{(m)}, \mu_1, \dots, \mu_K$ 
  Compute cost function (distortion)
   $J(c^{(1)}, \dots, c^{(m)}, \mu_1, \dots, \mu_K)$ 
}
  
```

Inicialización aleatoria múltiple dos K centroides.
Quedamos coa que dea lugar ao J menor.

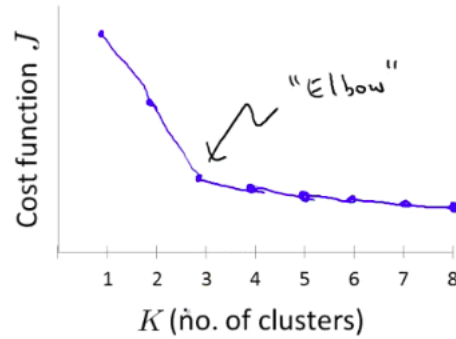
MÉTODO DO CÓBADO

xoves, 28 de decembro de 2023

20:16

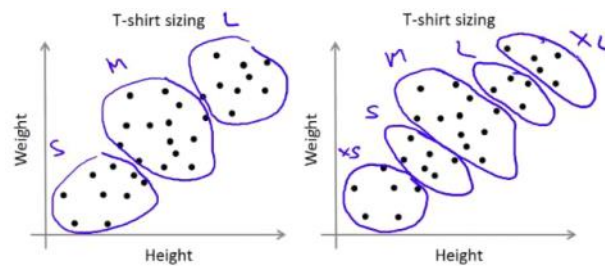


Selección do número K de agrupamentos.
Cál sería o mellor valor de K para os datos mostrados?



Método do "cóbado" para a determinación do valor de K .

É un método baseado no coñecemento do problema para a determinación do valor de K .

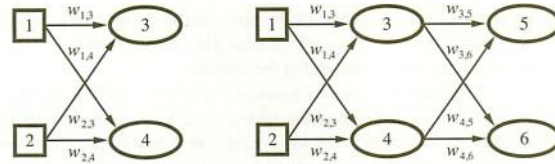
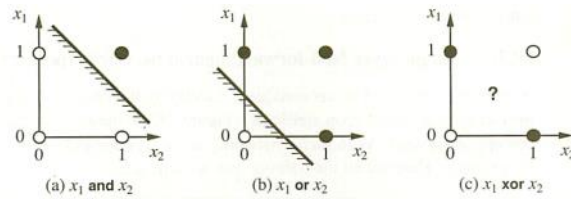


APRENDIZAXE AUTOMÁTICA EN RNA POR RETROPROPAGACIÓN

xoves, 28 de decembro de 2023 20:30

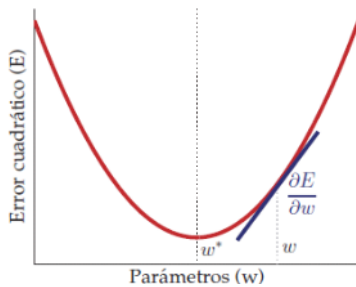
REDE PERCEPTRÓN MONOCAPA

x_1	x_2	y_{acarreo}	y_{xuma}
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0

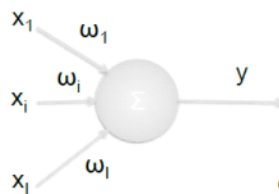


RETROROPAGACIÓN DE ERROS EN REDES MONOCAPA

Regra Delta para *Perceptróns* con NA lineais



$$\Delta w_i = -\eta \frac{\partial E}{\partial w_i}$$



$$y = \sum_{i=1}^I x_i w_i$$

$$E = (t - y)^2$$

$$\Delta w_i = -\mu \frac{\partial E}{\partial w_i}$$

$$\frac{\partial E}{\partial w_i} = \frac{\partial y}{\partial w_i} \frac{\partial E}{\partial y} = x_i \cdot (-2(t - y))$$

$$w_i(t + 1) = w_i(t) + 2\mu \cdot x_i \cdot (t - y)$$

Sumatorio de erros sobre o conxunto de adestramento:

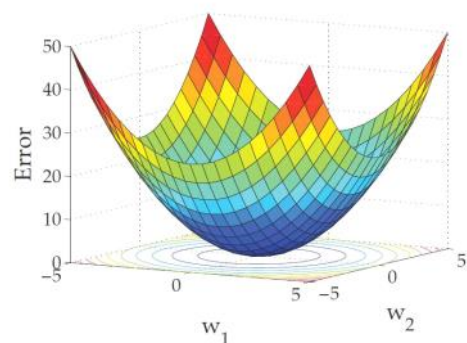
$$E = \frac{1}{2} \sum_{n=1}^N (t^n - y^n)^2$$

$$\frac{\partial E}{\partial w_i} = \frac{\partial y^n}{\partial w_i} \frac{\partial E}{\partial y^n}$$

$$\frac{\partial E}{\partial w_i} = \frac{\partial \left(\frac{1}{2} \sum_{n=1}^N (t^n - y^n)^2 \right)}{\partial w_i} = \frac{1}{2} \sum_{n=1}^N \frac{\partial y^n}{\partial w_i} \frac{\partial (t^n - y^n)^2}{\partial y^n} =$$

$$\frac{1}{2} \sum_{n=1}^N x_i^n \cdot (-2(t^n - y^n)) = - \sum_{n=1}^N x_i^n (t^n - y^n)$$

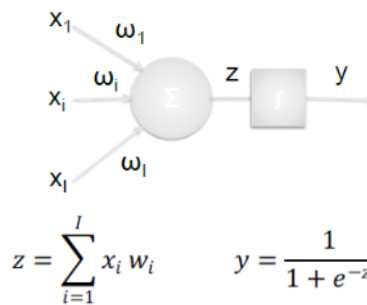
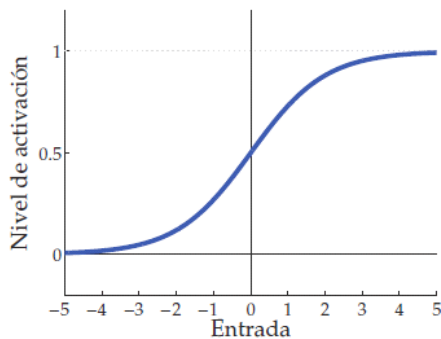
$$w_i(t + 1) = w_i(t) + \mu \sum_{n=1}^N x_i^n (t^n - y^n)$$



Regra Delta para *Perceptróns* con NA sigmoidais

venres, 29 de decembro de 2023

11:48



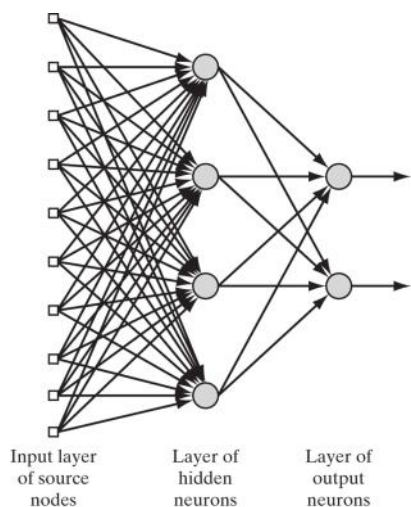
$$\frac{\partial y}{\partial z} = \frac{0 - (-e^{-z})}{(1 + e^{-z})^2} = \frac{e^{-z}}{(1 + e^{-z})^2} = \frac{1}{1 + e^{-z}} \cdot \frac{e^{-z}}{1 + e^{-z}} = y(1 - y)$$

Sumatorio de erros sobre o conxunto de adestramento:

$$\begin{aligned} \Delta w_i &= -\mu \frac{\partial E}{\partial w_i} = -\mu \frac{1}{2} \sum_{n=1}^N \frac{\partial y^n}{\partial w_i} \frac{\partial (t^n - y^n)^2}{\partial y^n} \\ &= -\mu \frac{1}{2} \sum_{n=1}^N \frac{\partial z^n}{\partial w_i} \frac{\partial y^n}{\partial z^n} \frac{\partial (t^n - y^n)^2}{\partial y^n} = \\ &= -\mu \frac{1}{2} \sum_{n=1}^N x_i^n (y^n (1 - y^n)) \cdot (-2(t^n - y^n)) \\ &= \mu \sum_{n=1}^N x_i^n (y^n (1 - y^n)) \cdot (t^n - y^n) \end{aligned}$$

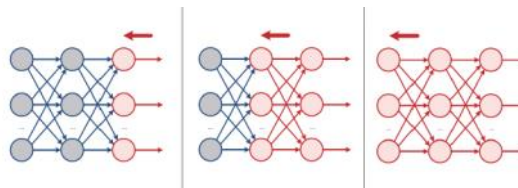


RETROPAGACIÓN DE ERROS EN REDES MULTICAPA



As **RNA multicapa** poden superar as limitacións das redes monocapa, singularmente do *Perceptrón*, pero **cómo lograr que aprendan?**

Retropropagación de erros $\partial E / \partial Y$



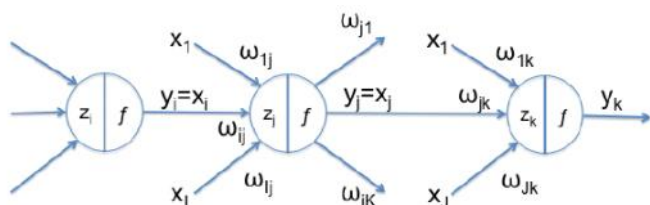
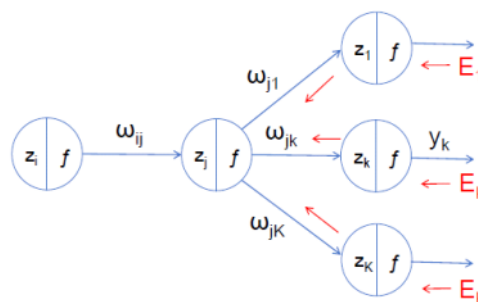
* Ver explicación detallada subida ao campus virtual !!

Algoritmo de retropropagación do erro para RNA

venres, 29 de decembro de 2023

12:14

Nunha **RNA multicapa**, as neuronas das **capas ocultas** afectanlle á saída da rede e, polo tanto, aos erros cometidos pola mesma. Deste xeito, os pesos das súas conexións vanlle afectar a ditos erros e deberán ser axustados para minimalizalos.



$$E_k = (t_k - y_k)^2$$

$$\frac{\partial E_k}{\partial w_{jk}} = \frac{\partial y_k}{\partial w_{jk}} \frac{\partial E_k}{\partial y_k} \quad \frac{\partial E_k}{\partial y_k} = -2(t_k - y_k)$$

$$\frac{\partial y_k}{\partial w_{jk}} = \frac{\partial f(z_k)}{\partial w_{jk}} = f'(z_k) \cdot \frac{\partial z_k}{\partial w_{jk}} = f'(z_k) \cdot \frac{\partial (\sum_j y_j w_{jk})}{\partial w_{jk}} = f'(z_k) \cdot y_j$$

$$\Delta w_{jk} = -\mu \frac{\partial E_k}{\partial w_{jk}} = \mu(t_k - y_k) \cdot f'(z_k) \cdot y_j = \mu \Delta_k \cdot y_j$$

Sumatorio de erros sobre o conxunto de adestramento:

$$\Delta w_{ij} = -\mu \sum_k \frac{\partial E_k}{\partial w_{ij}}$$

$$\frac{\partial E_k}{\partial w_{ij}} = \frac{\partial y_k}{\partial w_{ij}} \frac{\partial E_k}{\partial y_k}$$

$$\frac{\partial y_k}{\partial w_{ij}} = \frac{\partial f(z_k)}{\partial w_{ij}} = f'(z_k) \cdot \frac{\partial z_k}{\partial w_{ij}} = f'(z_k) \cdot w_{jk} \cdot f'(z_j) \cdot y_i$$

$$\frac{\partial E_k}{\partial w_{ij}} = -2(t_k - y_k) \cdot f'(z_k) \cdot w_{jk} f'(z_j) \cdot y_i = -2 \Delta_k \cdot w_{jk} \cdot f'(z_j) \cdot y_i$$

$$\Delta w_{ij} = -\mu \sum_k \frac{\partial E_k}{\partial w_{ij}} = \mu y_i \cdot f'(z_j) \cdot \sum_k (w_{jk} \Delta_k) = \mu y_i \cdot \Delta_j$$

Function **RETROPROPAGACIÓN** (ejemplos, red_neuronal) returns red_neuronal
inputs:

ejemplos /* conjunto de valores (X, T) del conjunto de entrenamiento

red_neuronal /* red neuronal con M capas, función de activación f y pesos w_{ij}

variables locales: Δ /* vector de errores

repeat

for each w_{ij} in red_neuronal do w_{ij} ← número pequeño aleatorio /*inicializa pesos

for each (X, T) in ejemplos do

for each nodo i in capa 1 do y_i ← x_i /* asigna valores de entrada a la red

for m=2 to M do /* propaga las entradas hacia la capa de salida

for each node j in capa m do

z_j ← ∑_i w_{ij} y_i

y_j ← f(z_j)

for each nodo k in capa M do Δ_k ← f'(z_k) · (t_k - y_k)

for m=M-1 to 1 do

for each nodo j in capa m do

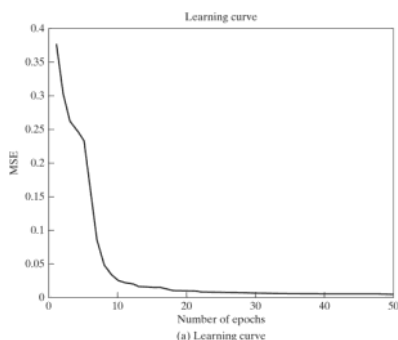
Δ_j ← f'(z_j) · ∑_k w_{jk} · Δ_k

for each w_{ij} in red_neuronal do w_{ij} ← w_{ij} + μ y_i · Δ_j /* actualiza los pesos

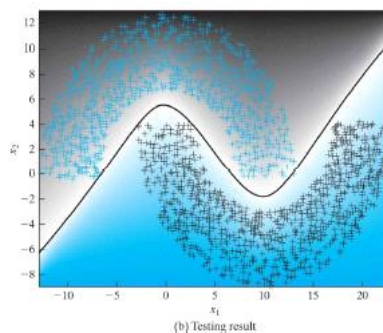
until criterio de finalización

return red_neuronal

Algoritmo de retropropagación del error para RNA



(a) Learning curve



(b) Testing result

Pazo