

Introducción a la asignatura

Perceptrón simple

Diego Milone
Inteligencia Computacional
Departamento de Informática

FICH-UNL

Organización

Introducción a la inteligencia computacional

Redes neuronales: generalidades

Percepción simple: modelo matemático y aprendizaje

Otros conceptos importantes



Inteligencia Natural vs Artificial

- A la hora de resolver problemas...
¿Qué nos diferencia de una computadora?
 - ¿En qué situaciones el ser humano es marcadamente mejor?
 - ¿Qué tipo de problemas son mejor resueltos por la computadora?

Inteligencia Natural vs Artificial

- A la hora de resolver problemas...
¿Qué nos diferencia de una computadora?
 - ¿En qué situaciones el ser humano es marcadamente mejor?
 - ¿Qué tipo de problemas son mejor resueltos por la computadora?
- Algunas capacidades del ser humano:
 - Aprendizaje, Generalización,
 - Adaptabilidad, Experiencia,
 - Razonamiento, Creatividad,
 - ...?

Inteligencia Natural vs Artificial

- A la hora de resolver problemas...
¿Qué nos diferencia de una computadora?
 - ¿En qué situaciones el ser humano es marcadamente mejor?
 - ¿Qué tipo de problemas son mejor resueltos por la computadora?
- Algunas capacidades del ser humano:
 - Aprendizaje, Generalización,
 - Adaptabilidad, Experiencia,
 - Razonamiento, Creatividad,
 - ...?
- ¿Qué fin buscamos en Inteligencia Artificial?

Inteligencia Artificial e Inteligencia “Computacional”

- Los orígenes de la inteligencia artificial

Inteligencia Artificial e Inteligencia “Computacional”

- Los orígenes de la inteligencia artificial
- ¿Qué modelamos?
 - Modelos del contenido: → sistemas basados en conocimientos
 - Modelos del continente: → modelos bio-inspirados
 - ...?



Inteligencia Artificial e Inteligencia “Computacional”

- Los orígenes de la inteligencia artificial
- ¿Qué modelamos?
 - Modelos del contenido: → sistemas basados en conocimientos
 - Modelos del continente: → modelos bio-inspirados
 - ...?
- ¿Cómo lo modelamos?
 - Modelos con énfasis en lo simbólico
 - Modelos con énfasis en lo numérico
 - ...?



Inteligencia Artificial e Inteligencia “Computacional”

- Los orígenes de la inteligencia artificial
- ¿Qué modelamos?
 - Modelos del contenido: → sistemas basados en conocimientos
 - Modelos del continente: → modelos bio-inspirados
 - ...?
- ¿Cómo lo modelamos?
 - Modelos con énfasis en lo simbólico
 - Modelos con énfasis en lo numérico
 - ...?
- La inteligencia computacional
 - Redes neuronales (Neural Networks)
 - Lógica borrosa (Fuzzy Logic)
 - Computación evolutiva (Evolutionary Computation)
 - Inteligencia distribuida (Swarm Intelligence)

La inspiración biológica

- La corteza cerebral ... 10^{11} ...
- Redes de neuronas
 - No linealidad, Paralelismo
 - Aprendizaje, Generalización
 - Adaptabilidad, Robustez

La inspiración biológica

- La corteza cerebral ... 10^{11} ...
- Redes de neuronas
 - No linealidad, Paralelismo
 - Aprendizaje, Generalización
 - Adaptabilidad, Robustez
- La neurona biológica: soma, dendritas, axón...
- Fisiología de la neurona:
 - sinapsis, neurotransmisores
 - despolarización, comportamiento todo/nada
 - propagación del impulso
 - refuerzo de las sinapsis... aprendizaje

La inspiración biológica

- La corteza cerebral ... 10^{11} ...
- Redes de neuronas
 - No linealidad, Paralelismo
 - Aprendizaje, Generalización
 - Adaptabilidad, Robustez
- La neurona biológica: soma, dendritas, axón...
- Fisiología de la neurona:
 - sinapsis, neurotransmisores
 - despolarización, comportamiento todo/nada
 - propagación del impulso
 - refuerzo de las sinapsis... aprendizaje
- Modelo simplificado de neurona

Arquitecturas neuronales

- Redes hacia adelante (feed–forward networks)
- Redes recurrentes (feedback networks)
- Otros modelos híbridos

Arquitecturas neuronales

- Redes hacia adelante (feed–forward networks)
 - Perceptrones simples y de una capa
 - Perceptrón multicapa
 - Redes con funciones de base radial
- Redes recurrentes (feedback networks)

- Otros modelos híbridos

Arquitecturas neuronales

- Redes hacia adelante (feed-forward networks)
 - Perceptrones simples y de una capa
 - Perceptrón multicapa
 - Redes con funciones de base radial
- Redes recurrentes (feedback networks)
 - Redes competitivas de una capa (SOM-1D)
 - Redes bidimensionales de Kohonen
 - Redes de Hopfield
 - Modelos de resonancia adaptativa
- Otros modelos híbridos

Arquitecturas neuronales

- Redes hacia adelante (feed–forward networks)
 - Perceptrones simples y de una capa
 - Perceptrón multicapa
 - Redes con funciones de base radial
- Redes recurrentes (feedback networks)
 - Redes competitivas de una capa (SOM-1D)
 - Redes bidimensionales de Kohonen
 - Redes de Hopfield
 - Modelos de resonancia adaptativa
- Otros modelos híbridos
 - Parcialmente recurrentes
 - Parcialmente conectados
 - Redes modulares, ...

Procesos de aprendizaje

- Tipos de aprendizaje
 - Supervisado
 - No supervisado
 - Híbridos
- Reglas de aprendizaje

Procesos de aprendizaje

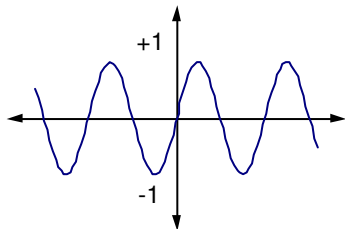
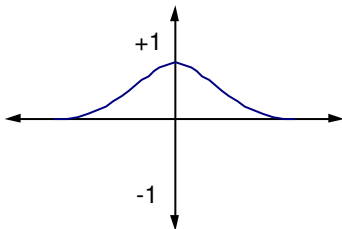
- Tipos de aprendizaje
 - Supervisado
 - No supervisado
 - Híbridos
- Reglas de aprendizaje
 - Corrección de error
 - Competitivo
 - Hebbiano
 - Boltzman

Perceptrón simple

- Modelo matemático del perceptrón simple
 - Producto interno
 - Entrada extendida
 - Funciones de activación

Perceptrón simple

- Modelo matemático del perceptrón simple
 - Producto interno
 - Entrada extendida
 - Funciones de activación



Perceptrón simple

- Modelo matemático del perceptrón simple
 - Producto interno
 - Entrada extendida
 - Funciones de activación
- Ejemplo: un PS de dos entradas
 - Espacio (plano) de soluciones
 - Hiperplano (recta) de decisión
 - Mapeo de la función OR
 - Importancia del bias
 - Cálculo directo de los w_i para el problema OR

Perceptrón simple

- Modelo matemático del perceptrón simple
 - Producto interno
 - Entrada extendida
 - Funciones de activación
- Ejemplo: un PS de dos entradas
 - Espacio (plano) de soluciones
 - Hiperplano (recta) de decisión
 - Mapeo de la función OR
 - Importancia del bias
 - Cálculo directo de los w_i para el problema OR
- ¿No era que iba a aprender solo a partir de los datos?

Perceptrón simple: aprendizaje

Entrenamiento por corrección de error:

1. Inicialización al azar
2. Se muestran muchos ejemplos con las salidas esperadas y en cada caso:

Perceptrón simple: aprendizaje

Entrenamiento por corrección de error:

1. Inicialización al azar
2. Se muestran muchos ejemplos con las salidas esperadas y en cada caso:
 - **SI** la salida de la red es correcta
no se hacen cambios: principio de mínima perturbación

Perceptrón simple: aprendizaje

Entrenamiento por corrección de error:

1. Inicialización al azar
2. Se muestran muchos ejemplos con las salidas esperadas y en cada caso:
 - **SI** la salida de la red es correcta
no se hacen cambios: principio de mínima perturbación
 - **SI** la salida de la red es incorrecta
penalización: se actualizan los w_i en el sentido opuesto al cual con el que contribuyeron a la salida incorrecta

Perceptrón simple: aprendizaje

Entrenamiento por corrección de error:

1. Inicialización al azar
2. Se muestran muchos ejemplos con las salidas esperadas y en cada caso: $y(n) = \phi(\langle \mathbf{w}(n), \mathbf{x}(n) \rangle)$
 - **SI** $y(n) = y_d(n) \Rightarrow$ *no se hacen cambios*
 - **SI** $y(n) \neq y_d(n)$ (supongamos $\mathbf{x}_i(n) > 0$)

Perceptrón simple: aprendizaje

Entrenamiento por corrección de error:

1. Inicialización al azar
2. Se muestran muchos ejemplos con las salidas esperadas y en cada caso: $y(n) = \phi(\langle \mathbf{w}(n), \mathbf{x}(n) \rangle)$
 - **SI** $y(n) = y_d(n) \Rightarrow$ *no se hacen cambios*
 - **SI** $y(n) \neq y_d(n)$ (supongamos $\mathbf{x}_i(n) > 0$)
 - **SI** $y(n) = +1$ mientras $y_d(n) = -1$
 $\Rightarrow \mathbf{w}(n+1) = \mathbf{w}(n) - \eta \mathbf{x}(n)$

Perceptrón simple: aprendizaje

Entrenamiento por corrección de error:

1. Inicialización al azar
2. Se muestran muchos ejemplos con las salidas esperadas y en cada caso: $y(n) = \phi(\langle \mathbf{w}(n), \mathbf{x}(n) \rangle)$
 - **SI** $y(n) = y_d(n) \Rightarrow$ *no se hacen cambios*
 - **SI** $y(n) \neq y_d(n)$ (supongamos $\mathbf{x}_i(n) > 0$)
 - **SI** $y(n) = +1$ mientras $y_d(n) = -1$
 $\Rightarrow \mathbf{w}(n+1) = \mathbf{w}(n) - \eta \mathbf{x}(n)$
 - **SI** $y(n) = -1$ mientras $y_d(n) = +1$
 $\Rightarrow \mathbf{w}(n+1) = \mathbf{w}(n) + \eta \mathbf{x}(n)$

Perceptrón simple: aprendizaje

Algoritmo del perceptrón simple:

1. Inicialización al azar: $\mathbf{w}(1) \in [-0.5 \ 0.5]$
2. Para cada patrón de entrenamiento $\mathbf{x}(n) | y_d(n)$:

- Se obtiene la salida:

$$y(n) = \phi(\langle \mathbf{w}(n), \mathbf{x}(n) \rangle)$$

- Se adaptan los pesos:

$$\mathbf{w}(n+1) = ?$$

3. Volver a 2 hasta satisfacer algún criterio de finalización.

Perceptrón simple: aprendizaje

Algoritmo del perceptrón simple:

1. Inicialización al azar: $\mathbf{w}(1) \in [-0.5 \ 0.5]$
2. Para cada patrón de entrenamiento $\mathbf{x}(n) | y_d(n)$:

- Se obtiene la salida:

$$y(n) = \phi(\langle \mathbf{w}(n), \mathbf{x}(n) \rangle)$$

- Se adaptan los pesos:

$$\mathbf{w}(n+1) = \mathbf{w}(n) + \frac{\eta}{2} [y_d(n) - y(n)] \mathbf{x}(n)$$

3. Volver a 2 hasta satisfacer algún criterio de finalización.

Otros conceptos importantes

- Espacio de soluciones
- Superficie de error
- Mínimos locales vs. mínimos globales
- Capacidad de generalización
- Métodos de validación cruzada