

Redes de funciones de base radial

Luis Ballado

CINVESTAV - UNIDAD TAMAULIPAS

luis.ballado@cinvestav.mx

February 25, 2023

Contenido

① Introducción

② Paradigmas de Aprendizaje

③ Paradigmas de Aprendizaje

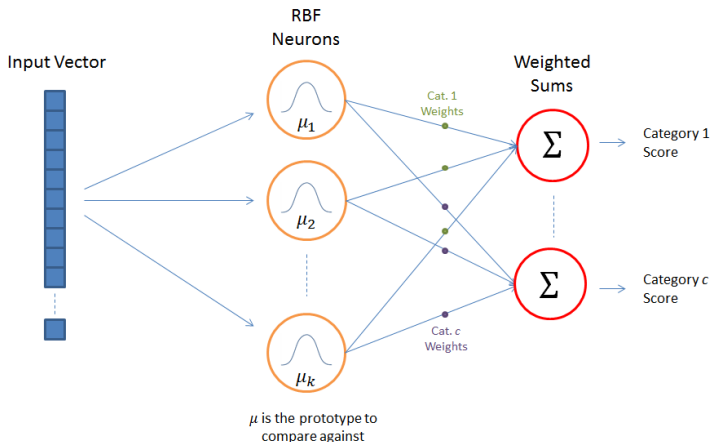
Aprendizaje supervisado

Aprendizaje no supervisado ó "clustering"

Aprendizaje por refuerzo

Introducción

Las redes neuronales de base radial están compuestas de tres capas: **capa de entrada**, **capa intermedia ó oculta**, en donde la función de activación de sus neuronas es Gaussiana, y **la capa de salida** con función de activación lineal (Perceptrón)



Introducción

Las funciones ϕ determinan las activaciones de las neuronas de la capa oculta en función de un muestra m .

Activación de las neuronas ϕ

Utilizando una ϕ Gaussiana la activación de las nueronas ocultas sería: $\phi = e^{\frac{x-c}{2n}}$

- p : Total de características
- C : Centroide perteneciente a la neurona oculta j
- x : Distancia euclidiana del vector de entrada y el centroide C_j
- Varianza

Salida de la red RBF (y)

$$y = W_i * \phi_i - \Theta$$

Donde:

- n: neuronas en la capa oculta
- m: patrón de entrada (seudomuestra)
- W_i : pesos sinápticos que conectan las neuronas de la capa oculta y la neurona de salida y
- Θ : Umbral de la neurona de salida
- Φ_j : Seudomuestra

En resumen: Para implementar una red RBF necesitamos:

- Un conjunto de muestras $\implies X$ (solo para la etapa de aprendizaje)
- Los centroides $\implies C$
- El valor de la variación $\implies \sigma$
- Las pseudomuestras $\implies Z$
- El valor de los pesos sinápticos $\implies w$

El proceso de aprendizaje de la red RBF

Este proceso se suele llamar híbrido ya que se divide en dos fases:

- Fase no supervisada
Se encarga de obtener el valor de los centroides C y la varianza Σ . Para esto se suele utilizar el método **k-means**
- Fase supervisada
Consiste en calcular los pesos w y umbral θ de las neuronas de la capa de salida. Para esto se suele utilizar el método de la **regla delta** o bien el método de la **matriz pseudoinversa**

Terminología:

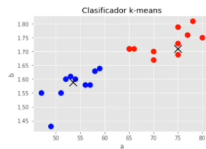
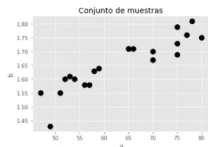
- m : Cantidad total de muestras
- X : Conjunto de muestras para entrenamiento $\Rightarrow X_0, X_1, X_2, \dots, X_m$
- Z : Conjunto de pseudomuestras $\Rightarrow Z_0, Z_1, Z_2, \dots, Z_m$
- Φ : Salida de una neurona en la capa de salida
- y : Salida de una neurona en la capa de salida
- w : Pesos sinápticos que enlazan la información de las neuronas de la capa oculta con las de la capa de salida.

Fase no supervisada

Clasificador k-means

Estableciendo 2 centroides o bien dos neuronas en la capa oculta

ϕ_1, ϕ_2 :



Obtener pseudomuestras

Una vez obtenidos los valores de centroides y varianza:
Generar pseudomuestras

- Obtener el conjunto de muestras X
- Generar la salida de las neuronas en la capa oculta por cada muestra del conjunto X

$$\Phi_j = e^{\frac{x}{y}}$$

- z_i : Pseudomuestra $\Rightarrow \begin{bmatrix} phi \\ phi \end{bmatrix}$
- Z : Pseudomuestras $\Rightarrow Z_0, Z_1, Z_2, \dots, Z_m$

Método por Matriz pseudoinversa

Debido a que la salida de la red depende linealmente de los pesos, es posible proporcionar una solución directa con lo siguiente:

$$w = G^+ * S \implies (G^t * G)^{-1} * G^t * S$$

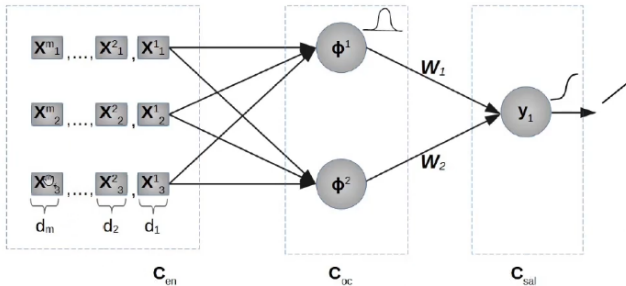
Donde:

$(G^t * G)^{-1} * G^t$, es la matriz pseudoinversa de G.

G^t , es la matriz transpuesta de G

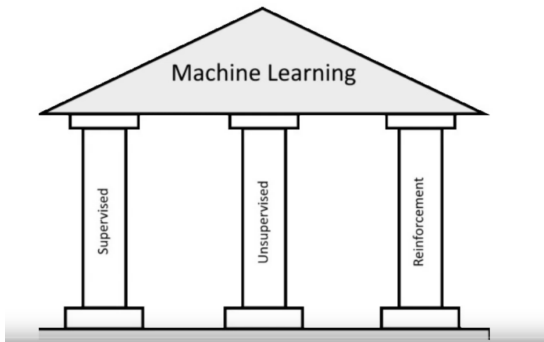
G, es una matriz de tamaño (X^m, n_{oc}) que contiene las activaciones de las neuronas de la capa oculta pero los patrones de entrada

S, es la matriz de salidas deseadas, de tamaño (X^m, n_{csal})



Paradigmas de Aprendizaje

Mecanismos que permiten que podamos procesar toda aquella información nueva que percibimos para acabar transformándola en conocimiento. La mayoría de los algoritmos dentro del ML se pueden clasificar en estos tres grupos.



¿Qué es el aprendizaje?

Norbert Wiener en su libro "Dios y el Golem" define aprendizaje como:

Definition

Un sistema organizado puede definirse como aquel que transforma un cierto mensaje de entrada en uno de salida, de acuerdo con algún principio de transformación. Si tal principio está sujeto a cierto criterio de validez de funcionamiento, y si el método de transformación se ajusta a fin de que tienda a mejorar el funcionamiento del sistema de acuerdo con ese criterio, se dice que el sistema aprende"



Aprendizaje supervisado

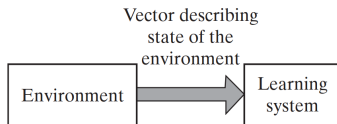
Surge en enseñarle a estos algoritmos cual es el resultado que quieres obtener para un determinado valor tras mostrarles muchos ejemplos si se dan las condiciones, el algoritmo será capaz de dar un resultado correcto incluso cuando le muestres valores que no ha visto antes.

Se denomina "clasificación"

- Los datos deben estar etiquetados, requiere supervisión humana.
- Se usan datos etiquetados para entrenamiento y se predicen individuos fuera de la muestra de entrenamiento

A partir de nuestros datos etiquetados entrenar un algoritmo que aprenda en relación a sus atributos.

Aprendizaje no supervisado



No existe alguien que enseñe o evalúe el proceso de aprendizaje, la red desarrolla la habilidad de formar representaciones internas para abstraer características de la entrada y crear nuevas clases automáticamente.

- Los datos no están etiquetados
- El algoritmo busca por su cuenta grupos o cluster

Nuestro algoritmo es ciego y sólo conoce los atributos y en base a esas características crear grupos

Se hace uso de reglas de aprendizaje competitivas
"winner-takes-all"

- Capa de entrada recibe la información disponible
- Capa competitiva consiste de neuronas que compiten entre ellas (en base a la regla de aprendizaje) para la esperanza de responder, la neurona con el mejor acierto gana y se activará, dejando a las demás neuronas apagadas.

Aprendizaje por refuerzo ¿Fuerza Bruta?

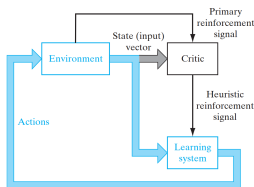
No tenemos una "etiqueta de salida", por lo que no es de tipo supervisado y si bien estos algoritmos aprenden por sí mismos, tampoco son de tipo no supervisado, en donde se intenta clasificar grupos teniendo en cuenta alguna distancia entre muestras.

El Aprendizaje por refuerzo, intentará hacer aprender a la máquina basándose en un esquema de "premios y castigos" - en un entorno en donde hay que tomar acciones y que está afectando por múltiples variables que cambian con el tiempo.

En modelos de Aprendizaje Supervisado (o no supervisado) se intenta minimizar la función de coste,.. es decir reducir el error.

En cambio en Aprendizaje Reforzado intenta "maximizar la recompensa" con la continua interacción con el ambiente para minimizar el index scalar de desempeño.

Componentes del RL



- Agente - será nuestro modelo que queremos entrenar y que aprenda a tomar decisiones
- Ambiente - será el entorno en donde interactúa y se mueve el agente. Contiene las limitaciones y reglas posibles en cada momento
- Acción - posibles acciones que puede tomar en un momento determinado el agente
- Recompensas (ó castigos) - podremos obtener un premio ó penalización que oriente al agente en si lo está haciendo bien o mal.

Q-Learning, el algoritmo más usado

El objetivo es entrenar nuestro modelo a través de simulaciones de manera que las decisiones que vaya tomando nuestro agente obtengan "la mayor recompensa"

- Políticas - Tabla que indicará al modelo como actuar en cada estado
- Acciones - las diversas acciones que puede hacer el agente
- Recompensas - si sumamos ó restamos puntaje con la acción tomada
- Comportamiento greedy - Tener grandes recompensas inmediatas o ir explorando y valorando las riquezas a largo plazo