

Redes Neuronales

Aprendizaje supervisado II

carlos.andres.delgado@correounivalle.edu.co

Carlos Andrés Delgado S.

Universidad San Buenaventura, Cali

Febrero de 2021

Contenido

1 Algoritmo BP con alfa variable

2 Problemas con los datos

3 Generalidades BP

Contenido

1 Algoritmo BP con alfa variable

2 Problemas con los datos

3 Generalidades BP

Problemas factor aprendizaje

Análisis

- No es posible conocer el valor del factor de aprendizaje, ya que depende del problema
- Por lo que, se debe seleccionar apropiadamente un valor de aprendizaje
 - Un valor muy pequeño hace que la convergencia sea muy lenta
 - Un valor adecuado, optimiza el número de iteraciones necesario para la convergencia
 - Un valor muy grande, hace que la convergencia sea muy lenta

Problemas factor aprendizaje

De acuerdo al error

- El factor de aprendizaje se debe ajustar en cada iteración
- Si el error se incrementa debemos disminuirlo
- Si el error se decrementa debemos incrementarlo

Algoritmo BP con alfa variable

Definición

- Se introducen tres parámetros:
 - ρ Es menor que 1 y mayor que 0.
 - α Es mayor que 1, pero cercano a 1. Ejemplo 1.01
 - ϵ_0 Factor de aprendizaje inicial

Estos valores son experimentales y si son mal escogidos puede hacer que el algoritmo no de buenos resultados.

Algoritmo BP con alfa variable

Definición

- Se aplica α para incrementar el entrenamiento cuando vamos por buena camino (el error se reduce), pero no exageramos para no dar incrementos exagerados en el error
- Se aplica ρ para abandonar valores elevados del aprendizaje cuando el error se incrementa de una iteración a la siguiente

Algoritmo BP con alfa variable

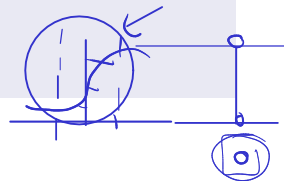
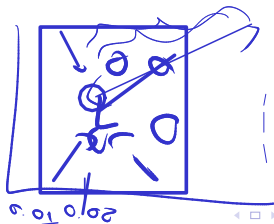
Definición

- La tasa de aprendizaje cambia de esta manera

$$\epsilon_{k+1} = \begin{cases} \alpha \epsilon_k & \text{si } E(w_{k+1}) < E(w_k) \\ \rho \epsilon_k & \text{si } E(w_{k+1}) \geq E(w_k) \end{cases}$$

$\alpha > 1$

$\rho < 1$



Contenido

1 Algoritmo BP con alfa variable

2 Problemas con los datos

3 Generalidades BP

Problemas con los datos

Generalidades

- Es importante que los valores deseados sean seleccionados dentro del rango de la función de activación (-1 o 1).
- Para mejorar el entrenamiento del algoritmo, los valores deben ser preprocesados para:
 - 1 Evitar que la entrada sea cero o cercana a cero, debido a que la regla delta depende de ella
 - 2 En ciertos casos las entradas pueden hacer que los pesos sólo aumenten o disminuyan, causando que no se pueda encontrar una solución (diverge)
 - 3 En otros casos las entradas pueden ocasionar un efecto de zig-zag con el error y nos quedemos en un óptimo local.

Problemas con los datos

Estrategia de preprocesamiento

Para tratar este problema normalizan las entradas así:

- 1 Elimina la media para que los datos no sean estrictamente positivos o negativos
- 2 Se ajusta cada una de las entradas para que estén entre -1 y 1.
 - 1 **Decorrelación:** Se busca con esto que las covarianzas de las entradas sean similares, para asegurar que los diferentes pesos sinápticos se ajusten a la misma velocidad.
 - 2 **Equalización:** Las variables de entrada son escaladas para que estén entre -1 y 1

Problemas con los datos

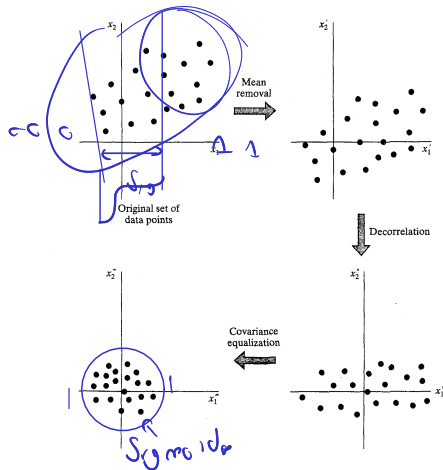


Figura: Estrategia de procesamiento de datos, tomado de [Haykin, 1998]

Contenido

1 Algoritmo BP con alfa variable

2 Problemas con los datos

3 Generalidades BP

¿Cuántas capas debe tener un MLP?

- Normalmente, tener tres capas es más que suficiente.
- Aumentar capas, aumenta en gran medida la complejidad computacional
- Es más viable aumentar el tamaño de la capa oculta que agregar capas

Velocidad de convergencia

- Se navega a través del gradiente descendiente para encontrar el error
- El entrenamiento da la velocidad de convergencia, entre más pequeño mejor. Sin embargo requiere más datos
- Un valor entrenamiento muy alto produce oscilaciones



Función de activación

- Se recomienda en las capas ocultas utilizar función no lineal (sigmoidal) para garantizar un procesamiento no-lineal
- Así mismo, una función no lineal, da más poder de clasificación de datos
- Para mejorar la precisión del algoritmo, se recomienda estandarizar los datos mediante una distribución gaussiana con media 0 y desviación igual a 1. Sin embargo para este curso, los trabajaremos tal cual.



Generalización

- El objetivo de una red neuronal es generalizar, es decir identificar patrones
- Si la red memoriza los patrones de entrada (da respuesta sin error), significa que la red se especializa o se aprende de memoria
- El sobre-entrenamiento trae como consecuencia que el error en los datos de entrenamiento sea pequeño y de prueba sea grande

2x2

2x2

2x

2x9

3x5

80%

20% prueba

Número optimo de Neuronas Ocultas

- Para seleccionar el número de neuronas en la capa oculta, para una entrada con n parámetros, se puede generar que la primera capa tenga n neuronas, la segunda $\frac{n}{2}$, la tercera $\frac{n}{3}$ y así, de acuerdo al teorema de Hecht-Neilson. En el caso de teorema de Kolmogorov la segunda tiene $2n - 1$ neuronas, la tercera $\frac{n}{2}$, la cuarta $\frac{n}{3}$ y así.
- Sin embargo, en algunos problemas se requiere seleccionar el número de neuronas de capa de acuerdo a la experiencia de los ingenieros.

Generalización

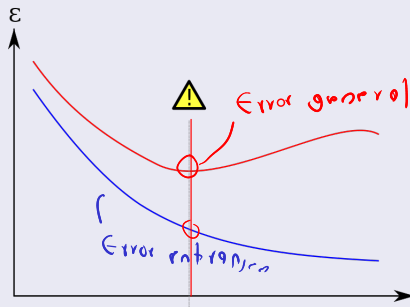


Figura: Sobreentrenamiento. Línea azul. Error de entrenamiento. Línea roja. Error general. Tomado de Wikipedia

Generalización

Para enfrentar este problema, existen dos estrategias:

- **Regularización por parada temprana:** Se separa el conjunto de entrenamiento, en dos conjuntos, uno de entrenamiento y otro de validación. Con el conjunto de entrenamiento realizamos el entrenamiento de la red y con el de validación calculamos el error de entrenamiento. Normalmente se toma un 20 % a 30 % de los datos totales para validación.
- **Regularización por limitación de magnitud de paso:** Se realiza un ajuste en el calculo del error introduciendo un parámetro λ que se multiplica los pesos. Este parámetro es menor que 1 y mayor que 0.

Referencias I



Eduardo, C. and Jesus Alfonso, L. (2009).

Una aproximación práctica a las redes neuronales artificiales.

Colección Libros de Texto. Programa Editorial Universidad del Valle.



Haykin, S. (1998).

Neural Networks: A Comprehensive Foundation (2nd Edition).

Prentice Hall.



Widrow, B. and Winter, R. (1988).

Neural nets for adaptive filtering and adaptive pattern recognition.

Computer, 21(3):25–39.

¿Preguntas?

Próximo tema:
Procesamiento de datos