**13**

Capítulo 2

Métodos de cálculo de errores

* Comprensión del cálculo de errores
* La función de error
* Métodos de cálculo de errores
* ¿Cómo se utiliza el error

En este capítulo veremos cómo calcular errores para una red neuronal. Al realizar el entrenamiento supervisado, la salida real de una red neuronal debe compararse con la salida ideal especificada en los datos de entrenamiento. La diferencia entre la salida real e ideal es el error de la red neuronal.

El cálculo de errores se produce en dos niveles. Primero está el error local. Esta es la diferencia entre la salida real de una neurona dual indiviy la salida ideal que se esperaba. El error local se calcula mediante una función de error.

Los errores locales se agregan juntos para formar un error global. El error global es la medición de lo bien que funciona una red neuronal a todo el conjunto de entrenamiento. Hay varios medios diferentes por los cuales se puede calcular un error global. Los métodos de cálculo de errores globales descritos en este capítulo se enumeran aquí.

* Error de suma de cuadrados (ESS)
* Error cuadrado medio (MSE)

**14 Métodos de cálculo**  **de** **errores**

* Cuadrado medio raíz (RMS)

Por lo general, simplemente usará MSE. MSE es el medio más común de calcular errores para una red neuronal. Más adelante en el libro usaremos ESS. El Algoritmo Levenberg Marquardt (LMA), que se cubrirá en el Capítulo 8, requiere ESS. Por último, RMS se puede utilizarful en ciertas situaciones.

* 1. **La función** **de error**

Comenzaremos con el error local. El error local proviene de la función de error. La función de error se alimenta de las salidas reales e ideales para una sola neurona de salida. A continuación, la función de error produce un número que representa el error de esa neurona de salida. Los métodos de entrenamiento verán para minimizar este error.

Este libro cubrirá dos funciones de error. La primera es la función de error lineal simple que se utiliza más comúnmente. La segunda es la función de error arctangent introducida por el método de entrenamiento de propagación rápida. Las funciones de error de Arctangent y la propagación rápida se discutirán en el Capítulo 6, "Propagación rápida". Este capítulo se centrará en la función de error lineal estándar. La fórmula para la función de error lineal se puede vern en la ecuación 2.1.

*E* = (*i* *−* *a*) (2.1)

La función de error lineal es muy simple. El error es simplemente la diferencia entre las salidas ideales (**i**) y reales(**a**) de la red neuronal. El único requisito de la función de error es que produzca un error que desea minimizar.

Para ver un ejemplo de esto, considere una neurona de salida de red neuronal que produjo 0.9 cuando debería haber producido 0.8. El error para esta red neuronal sería la diferencia entre 0.8 y 0.9, que es -0.1.

En algunos casos, es posible que no proporcione una salida ideal a la red neuronal y siga utilizando el entrenamiento supervisado. En este caso escribiría una función de error que de alguna manera evalúa la salida de la red neuronal para la entrada dada. Esta función de error de evaluación tendría que asignar algún tipo denúcleo s a lared neuronal. Un número más alto indicaría una salida menos deseable; un número más bajo indicaría una salida más deseable. El proceso de entrenamiento intentaría minimizar esta puntuación.

* 1. **Cálculo del** **error**global **15**

# Cálculo del error global

Ahora que hemos visto cómo calcular el error local, pasaremos a un error global. Debido a que el cálculo de errores MSE es el más común, comenzaremos aquí. Puede ver la ecuación que se utiliza para calcular el MSE en la ecuación 2.2.

*n*

1 

MSE = *E*2 (2.2)

*n i*=1

Como se puede ver en la ecuación anterior, hace uso del error local (E) que definimos en la última sección. Cada error local se cuadra y se suma. A continuación, la suma resultante se divide por el número total de casos. De esta manera, el error MSE es similar a un promedio tradicional, excepto que cada error local está cuadrado. El squaring niega el efecto de que algunos errores sean positivos y otros sean negativos. Esto se debe a que un número positivo al cuadrado es un número positivo, del igual que un cuadrado numérico negativo también es un número positivo. Si no está familiarizado con el operador de suma, que se muestra como una letra griega mayúscula sigma, consulte el Capítulo 1.

El error MSE normalmente se escribe como un porcentaje. El objetivo es disminuir este error porcentaje a medida que avanza la formación. Para ver cómo se utiliza esto, considere el siguiente programa Salida.

Comenzando T R Un Yo N Yo N G . . .

I T e R Un T Yo o N #1 Error : 51. 023786 % Error de destino : 1.0000000% Yo T e R Un T Yo o N #2 Error : 49 . 659291% Error de destino : 1.0000000% Yo T e R Un T Yo o N #3 Error : 43 . 140471% Error de destino : 1.0000000% Yo T e R Un T Yo o N #4 Error : 29 . 820891% Error de destino : 1.0000000% Yo T e R Un T Yo o N #5 Error : 29 . 457086 % Error de destino : 1.0000000% Yo T e R Un T Yo o N #6 Error : 19 . 421585 % Error de destino : 1.0000000% I t e r a t i o n #7 Error : 2 . 160925 % Error de destino : 1.0000000% I t e r a t i o n #8 Error : 0 . 432104 % Error de destino : 1.000000% Entrada = 0 . 0000 , 0 . 0000 , Real =0 . 0091 , i d e a l = 0.0000 Entrada = 1 . 0000 , 0 . 0000 , Real =0 . 9793 , i d e a l = 1.0000 Entrada = 0 . 0000 , 1 . 0000 , Real =0 . 9472 , Yo sería e a l = 1.0000 Entrada = 1 . 0000 , 1 . 0000 , Real =0 . 0731 , Yo D e Un L = 0.0000

Tipo de aprendizaje automático automático : F eedforward

Máquina Aprendizaje Ar C H Yo te C tu R e : ? : B*−>*Sigmoideo*−>*4:B*−>*Sigmoideo*−>*?

Método método de entrenamiento : Lma Entrenamiento Args :

Lo anterior muestra un programa que aprende el operador XOR. Observe cómo el error MSE cae cada iteración? Finalmente, pero la iteración ocho el error está por debajo del uno por ciento, el entrenamiento hormiga se detiene.

**16 Métodos de cálculo**  **de** **errores**

# Otros métodos de cálculo de errores

Aunque MSE es el método más común para calcular errores globales, no es el único método. En esta sección veremos otros dos métodos de cálculo de errores globales.

## Error de suma de cuadrados

El método sum of squares (ESS) utiliza una fórmula similar a la del método de error MSE. Sin embargo, ESS no se divide por el número de elementos. Como resultado, el ESS no es un porcentaje. Es simplemente un número que es más grande dependiendo de lo grave que sea el error. La ecuación 2.3 muestra la fórmula de error MSE.

Ess = 1  *E*2 (2.3)

*P*

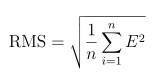
2

Como se puede ver anteriormente, la suma no se divide por el número de elementos. Más bien, la suma se divide simplemente a la mitad. Esto da lugar a un error que no es un porcentaje, sino más bien un total de los errores. Cuadrar los errores elimina el efecto de errores positivos y negativos.

Algunos métodos de entrenamiento requieren que utilice ESS. El algoritmo Levenberg Marquardt (LMA) requiere que el método de cálculo de errores sea ESS. LMA estará cubierta en el Capítulo 8.

## Error cuadrado medio raíz

El método de error root mean square (RMS) es muy similar al método MSE descrito anteriormente. La principal diferencia es que la raíz cuadrada se toma de la suma. Puede ver la fórmula RMS en la ecuación 2.4.



Rms = 1 *n*

'. , 

*n i*=1

*E*2 (2.4)

El error cuadrado medio raíz siempre será mayor que MSE. La siguiente salida muestra el error calculado para los tres métodos de cálculo de errores. Los tres casos utilizaron el mismo Y Ideal Valores.

Intentando de *−*1.00 Un 1 . 0 0

* 1. **Resumen**del Capítulo **17**

Real : [ *−* 0 . 36 , 0 . 07 , 0 . 55 , 0 . 05 , *−* 0 . 37 , 0 . 34 , *−* 0 . 72 , *−* 0 . 10 , *−* 0 . 41 , *−* 0 . 32 ]

Error ( ESS) : 0 . 00312453 Error (MSE) : 0.062491% Error (RMS) : 2.499810%

I D e Un L : [ *−* 0 . 37 , 0 . 06 , 0 . 51 , 0 . 06 , *−* 0 . 36 , 0 . 35 , *−* 0 . 67 , *−* 0 . 09 , *−* 0 . 43 , *−* 0 . 33 ]

RMS no se utiliza muy a menudo para el cálculo de errores de red neuronal. RMS fue creado originalmente en el campo de la ingeniería eléctrica. No he usado mucho RMS. Muchos artículos de investigación que involucran RMS muestran que se utiliza para el análisis de forma de onda.

# 2.4 Resumen del capítulo

Las redes neuronales comienzan con valores aleatorios para pesos. Estas redes se entrenanhasta quese encuentra un conjunto de pesos que proporciona salida de la red neuronal que coincide estrechamente con los valores ideales de los datos de entrenamiento. Para el entrenamiento para progresar se necesita un medio para evaluar el grado en que la salida real de la red neuronal coincide con la salida ideal esperada de la red neuronal.

Este capítulo comenzó introduciendo el concepto de error local y global. El error local es el error utilizado para medir el diversoencaje entre la salida real e ideal de una neurona de salida individual. Este error se calcula mediante una función de error. Las funciones de error solo se utilizan para calcular el error local.

El error global es el error total de la red neuronal en todas las neuronas de salida y elementos de conjunto de entrenamiento. En este capítulo se presentaron tres técnicas diferentes para el cálculo del error global. Error cuadrado medio (MSE) es el más utilizado. Algunos métodos de entrenamiento utilizan la suma de errores cuadrados (ESS) para calcular el error. Root Mean Square (RMS) se puede utilizar para calcular el error para ciertas aplicaciones. RMS fue creado para el campo de la ingeniería electrical para el análisis de forma de onda.

El siguiente capítulo introducirá un concepto matemático conocido como derivados. Los derivados provienen de Calculus y se utilizarán para analizar las funciones de error y ajustar las ponderaciones para minimizar este error. En este libro we aprenderá sobre varias técnicas de entrenamiento de propagación. Todas las técnicas de entrenamiento de propagación utilizan derivados para calcular los valores de actualización de los pesos de la red neuronal.