



UNIVERSIDAD
NACIONAL
DE COLOMBIA



Universidad Nacional de Colombia

Sede Medellín

Ajuste de hiperparámetros



Profesora: Patricia Jaramillo A. Ph.D

Ajuste de hiperparámetros

Se refiere a los aspectos del modelo que deben ser elegidos por el analista. Por ejemplo: número de capas, número de neuronas, número de clusters, etc.

Se denominan también parámetros no entrenables.

Generalmente, tiene interpretación intuitiva, por eso, suele ser difícil asignarles un valor óptimo, y requiere de muchas habilidades del analista.

Suelen tener un alto impacto en el desempeño del modelo.

Aún no se sabe cuál es la mejor manera de optimizarlos.

Algunos Hiperparámetros de una red neuronal

- Número de capas ocultas
- Función de activación: en una misma red neuronal puede usarse diferentes funciones.
- Pesos iniciales: pueden ser aleatorios, o usar heurísticas

Relativos al algoritmo de entrenamiento:

- Tasa de aprendizaje del algoritmo del gradiente
- Tamaño de lote (Batch) y forma de usarlo
- Número de iteraciones
- Algoritmo de optimización
- Tamaño de conjunto de entrenamiento y de validación

Métodos más comunes de asignarle valores “óptimos”:

- Ensayo y error
- Búsqueda sistemática (Nive grid search)
- Búsqueda aleatoria (sampling random combinations)
- Optimización bayesiana
- Métodos Metaheurísticos: Evolutivos, Ant Colony, Enfriamiento simulado, Búsqueda tabú, PSO, etc
- Best Arm Identification
- etc.

Ensayo y error: Se eligen manualmente, por ensayo y error, a partir de las habilidades del analista.

Grid Search

Se prueba, sistemáticamente, con diferentes valores de los hiperparámetros y, para cada conjunto de valores, se entrena la red. Se elige el conjunto con mejor desempeño.

Búsqueda aleatoria

Se asignan valores aleatoriamente. A veces, suele ser más efectivo que los anteriores.

Optimización Bayesiana

- Analiza todos los resultados parciales conseguidos hasta la iteración actual y elige probabilísticamente (mayor mejora esperada) cuáles regiones podrían tener mejores resultados.
- Se basa en modelos de procesos gaussianos: Para cada dimensión de entrada, “aprend”e una escala apropiada para medir similaridad y da como resultado una distribución gaussiana de valores en vez de un único valor.

Best Arm Identification

- Trata de encontrar el mejor “brazo”: un conjunto de valores de los hiperparametros
- Evalua el desempeño de un grupo de brazos, conserva la mitad y repite el proceso hasta que el mejor brazo permanece
- Permite implementar el algoritmo en paralelo.

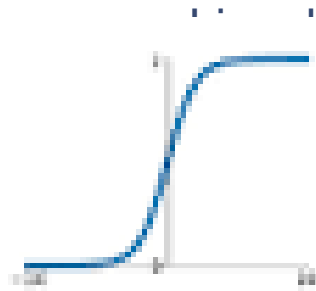
Experimentos

- Independiente del método que se usa, debe hacerse diferentes experimentos repetidamente para testear los resultados, cambiando datos del conjunto de entrenamiento y de validaciones.
- Estos experimentos pueden hacerse en paralelo en diferentes máquinas o en la nube.

Funciones de activación

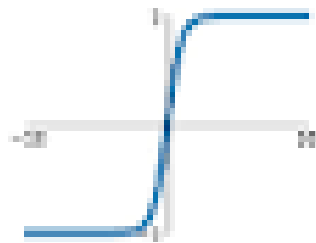
Sigmoid

$$\sigma(x) = \frac{1}{1+e^{-x}}$$



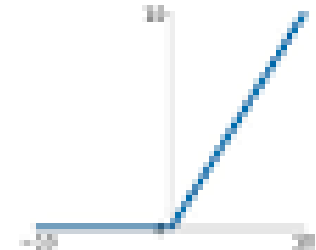
tanh

$$\tanh(x)$$



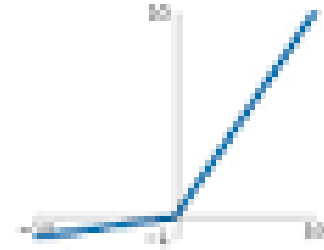
ReLU

$$\max(0, x)$$



Leaky ReLU

$$\max(-0.1x, x)$$

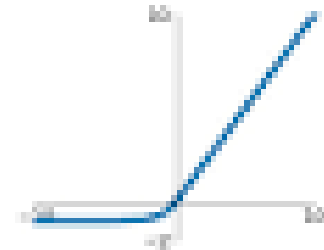


Maxout

$$\max(w_1^T x + b_1, w_2^T x + b_2)$$

ELU

$$\begin{cases} x & x \geq 0 \\ \alpha(e^x - 1) & x < 0 \end{cases}$$



Librerías para ayudar en la optimización de parámetros:

En Keras Hyperopt, Kopt , Talos

En Tensorflow, GPflowOpt (optimización Bayesiana, y soluciones comerciales como Google's Cloud Machine Learning Engine)

PyTorch

<https://medium.com/@mikkokotila/a-comprehensive-list-of-hyperparameter-optimization-tuning-solutions-88e067f19d9>