GUIÓN

Diap 1. Redes neuronales, qué son. Son un modelo (o estructura) inspirado en el comportamiento observado en el cerebro humano​. Consiste en un conjunto de unidades, llamadas neuronas artificiales, conectadas entre sí para transmitirse señales. La información de entrada atraviesa la red neuronal (donde se somete a diversas operaciones) produciendo unos valores de salida.

Diap 2. Neurona, basada en neurona humana. Entradas -> cálculos -> salida. Unidad mínima de la red neuronal, realiza los cálculos e interacciona con otras neuronas (o no) para colaborar con el objetivo de la red.

Diap 3. Explicación de entradas, bias, pesos, y funciones de activación. Limitaciones de una sola neurona.

Entradas: datos del problema.

Pesos: valor de “relevancia que se le da a cada entrada”

Bias: actúa como sesgo ya que controla qué tan predispuesta está la neurona a disparar una determinada salida independiente de los pesos. Un sesgo alto hace que la neurona requiera una entrada más alta para generar una salida de alta.

Función de activación: Cálculo que se realiza con la salida de la neurona en función de la naturaleza del problema.

Diap 4. Gradient Descend: es el algoritmo de aprendizaje más utilizado (definición de 1º de carrera de algoritmo). Su objetivo es reducir la función de error, es decir, lo que falla la red. “La red actúa, se equivoca, tiene en cuenta si se ha pasado se ha quedado corta y para la siguiente vez, lo hará mejor” “El equivalente cotidiano podría decirse que es calcular a ojo”

Diap 5. Red neuronal, como aprenden: error, forward y backward

Forward: la red avanza, hace sus cálculos y expone su respuesta. LOS PESOS AQUÍ NO CAMBIAN. Lo que importa para que la red funcione bien son los pesos.

Bachprop: se calcula el error, y se propaga hacia atrás, modificando levemente o no (mencionar lr) los valores definidos en el forward.

Diap 6. Ya ejemplos del cancer y el spam. La cosa de la clasificación es que las salidas son o 0/1.

Cancer:

las 9 features son: lo pondrá en el notebook. Esta red tiene esta arquitectura, la última función es el sigmoid porque quiero un 0 o un 1.

Mirar como aprende, esta es la función de error a lo largo del tiempo. Decir que son las epochs(es el número de veces que vamos ahacer el gradient descend, el número de veces que va a intentar predecir algo)

Explicar confusión matrix

Spam:

Aquí digo que la arquitectura es diferente. Ya está la red ejecutada.

Enseño el Dataset, la confusión matrix y paso directamente a probar con una frase.