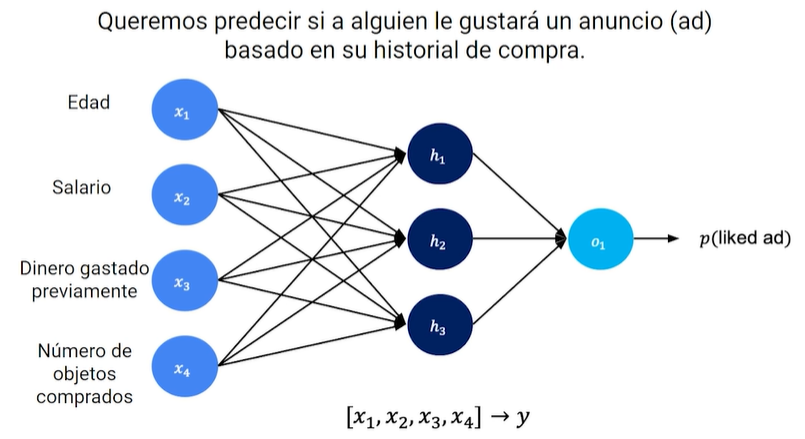
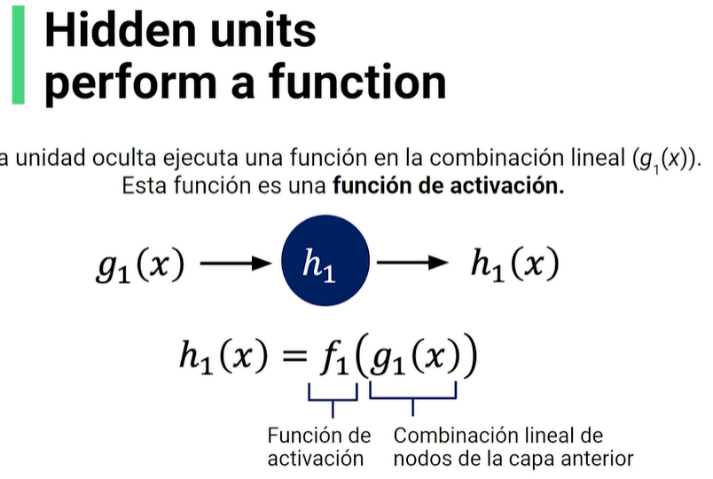
* Herramientas: pyTorch y TensorFlow
* Subcategoría de ML que crea diferentes niveles, de abstracción que representa los datos.
* Se usan tensores para representar estructuras de datos más complejas.
* Neuronas: Capa de entrada, capas ocultas y capa de salida.
* Para poder aprender se necesita una función de activación: ReLU: Permite el paso de todos los valores positivos sin cambiarlos, pero asigna todos los valores negativos a 0.
* TensorFlow: Biblioteca de código abierto desarrollado por google, capaz de construir y entrenar redes neuronales.

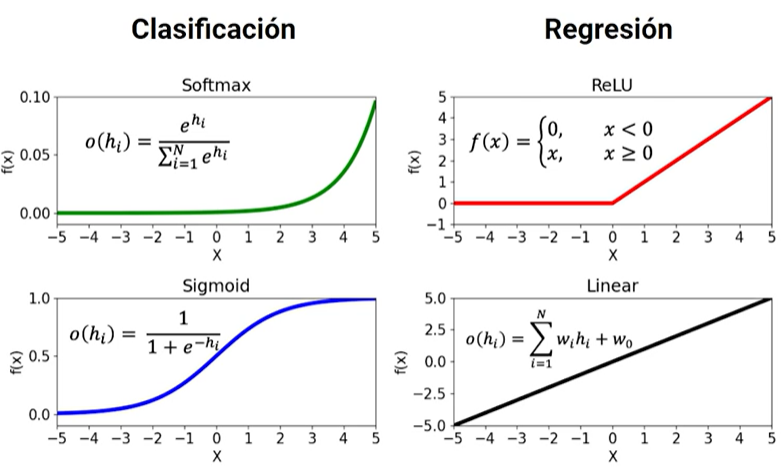


<https://www.youtube.com/watch?v=aircAruvnKk&list=PLZHQObOWTQDN52m7Y21ePrTbvXkPaWVSg&index=4>



Tipos de activacion

Algunas funcines de activacion tienen un rango limitado (softmax/sigmoide) y otros indefinidamente (ReLUs, lineal)



COMO SE ENTRENA REDES NEURALES

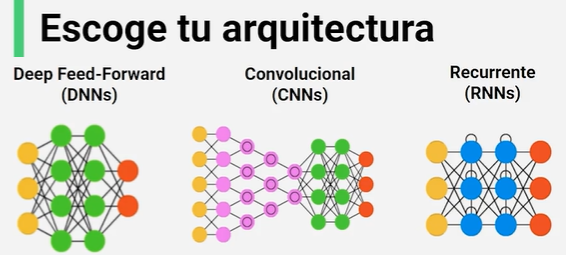
El siguiente artículo brinda información para construir una red neuronal utilizando Python

[https://towardsdatascience.com/how-to-build-your-own-neural-network-from-scratch-in-python-68998a08e4f6?gi=c278a6a3944](https://platzi.com/clases/2459-machine-learning/40699-como-es-el-entrenamiento-de-las-redes-neuronales/url)

1. Escoge arquitectura

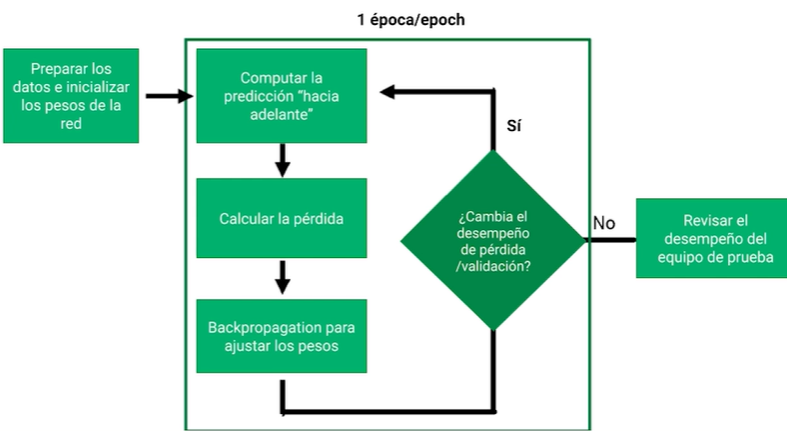
2. La receta de entrenamiento

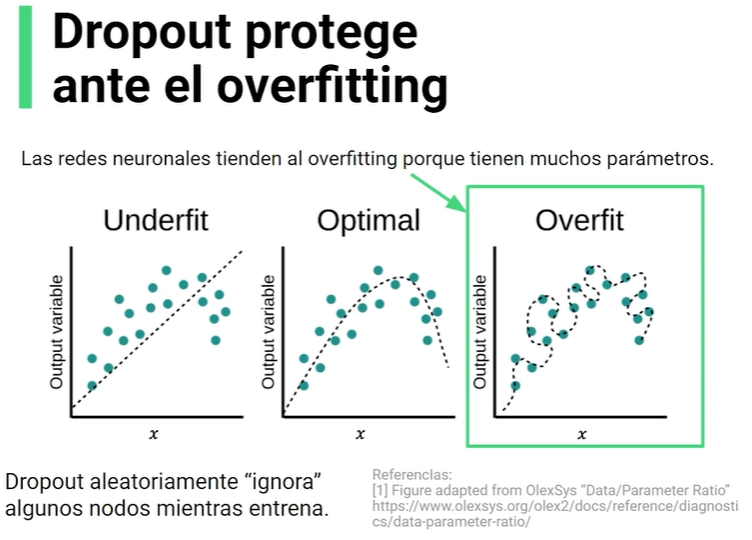
3. Ajusta tasa de entrenamiento



Recurrente se utiliza para lenguaje

RECETA DE ENTRENAMIENTO





RESUMEN

* Redes neurales tienen 3 capas generales: entrada, ocultas, salida
* Funciones de activacion para capas ocultas y de salida
* Entrenar involucre avanzar en la red, calcular la Perdida y backprogpagacion
* Tasa de aprendizaje y dropout son importantes para el entrenamiento
* Revisar la Perdida y el rendimiento en el set de validacion