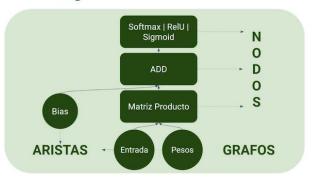


Lógica de TensorFlow



- TensorFlow es un flujo entre tensores.
- Keras es una capa que incorpora todo lo que hace TensorFlow y lo simplifica.
- Keras fue creado por un ingeniero de Google.



El ecosistema de TensorFlow



F

Comunidad de TF

- 56,000 preguntas en StackOverflow.
- 109 ML Google Developer Experts.
- 46 TensorFlow user groups en todo el mundo.
- 25 publicaciones no blog de TensorFlow.
- ~ 70,000 commits.
- > 2,200 contribuyentes.

-



Generator en Python

Es una función que devuelve un objeto, el cual te permite iterar en el mismo, sin almacenarlos a la vez en memoria.



Donde buscar bases de datos para Deep Learning?

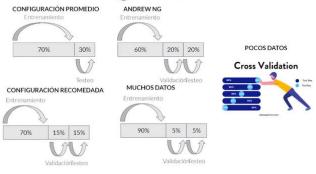
- Kaggle
- Datos abiertos
- Data world
- Data.gov
- Google: https://datasetsearch.research.google.com/
- Github: https://github.com/awesomedata/awesome-public-datasets



Distribución de bases de datos



Distribución de los datos según casos



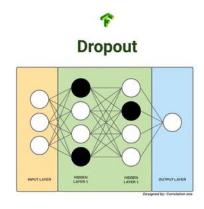


Errores más comunes

- Agregar datos de entrenamiento a testeo.
- Bases de datos no balanceadas en clases.
- Muy pocos datos.



Métodos de regularización





Recomendaciones practicas para ajustar modelos de Redes Neuronales

- Pre-procesamiento:

Recomendaciones para preprocesamiento

- Busca datos null.
- Archivos corruptos.
- Balancea tu base de datos.
- Aplica normalización.
- Visualiza la base de datos.
- Entiende los datos.



- Parámetros:



Recomendaciones de valores

- Convoluciones (3x3)
- Pooling (2x2)
- Flatten (imágenes)
- Neuronas (64, 128, 256, 512)
- Learning Rate (0.001) ADAM
- L1_2 (1e-5)
- Dropout (0.2)
- Regularización:



- L1, L2, L1+L2
- Agrega más datos.
- Data augmentation.
- Dropout.
- Early stopping.
- Callbacks.



- Funciones de activación:



- Multi Clases: Softmax
- Binarios: Sigmoidal
- Regresiones: Función lineal.
- Clase predicciones mayor que 0: **RelU.**



- Configuración de red:



- Aplica capas convoluciones y poolings (1 layers).
- Tómate el tiempo de encontrar el learning rate óptimo.
- Ve almacenando en cada epoch de entrenamiento.



Recomendaciones de la red

- Entre más capas y neuronas, no significa que es mejor la red.
- La solución no es siempre redes neuronales.
- Aliado aprendizaje por transferencia.



Metricas para medir la eficiencia del modelo por medio de Callbacks

Cuando entrenamos nuestros modelos requerimos de monitorear su desempeño durante el entrenamiento, esto nos dará tiempo de reacción para entender por qué rinden como rinden. En esta ocasión manejaremos callbacks personalizados para monitorear el desempeño de nuestra red en términos de precisión y pérdida.

Keras Autotuner



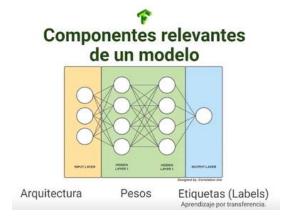
Historia del Tunner de Keras



Almacenamiento y carga de modelos

Componentes de un modelo:

- Arquitectura: cantidad de neuronas, capas, parámetros.
- Pesos:
- Etiquetas: clases del modelo



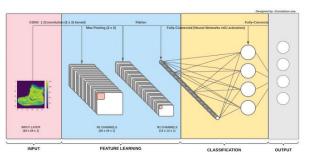
Aprendizaje por transferencia

Usar la misma arquitectura de un modelo anterior y usarla para un nuevo modelo, ajustando en este ultimo solamente los pesos. Eliminamos la ultima capa y la reemplazamos por la ultima capa del problema que estemos resolviendo.

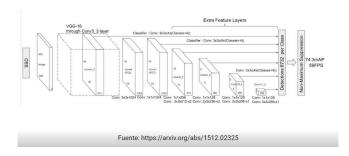








Ejemplo de infraestructuras de una red convolucional



InceptionV3 es el modelo que se está cargando

Cuando hacer uso del aprendizaje por transferencia?

- NLP: tenemos palabras. Cada palabra se ve como un byte que representa información.
- Vision computarizada: usa pixeles, con grandes volúmenes de imágenes.

Cuándo utilizar modelos pre-entrenados



Procesamiento de lenguaje natural



Visión computarizada



¿Por qué utilizarlos?

- Se necesita poca data.
- Te permite generar iteraciones muy rápidas.
- Generaliza muy bien los modelos.
- Transferir extracción de características principales.







The second of th

YoloV3

AlexNet

Estos son modelos pre-entrenados.

AlexNet: https://arxiv.org/ftp/arxiv/papers/1803/1803.01164.pdf & https://cvml.ista.ac.at/courses/DLWT_W17/material/AlexNet.pdf

YOLOV3: https://pjreddie.com/media/files/papers/YOLOv3.pdf



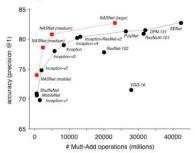




Accuracy (precisión)

Coste computacional

¿Cómo seleccionar cual modelo pre-entrenado usar?



Fuente: https://ai.googleblog.com/2017/11/automl-for-large-scale-image.html

Los que tienen mejor relación coste computacional – accuracy son: inception v1, nasnet, mobilenet.

API Funcional de Keras

Permite hacer redes neuronales no secuenciales.

