

# **Dudas y Problematicas**

# Tensorflow 2 vs Keras

Problemática general el entrenamiento de TF2 difiere del realizado en Keras

- ▶ Esta diferencia existe cuando se usa Gradient tape Vs model.fit()

# Gradient tape Vs model.fit()

- ▶ Problema en el cálculo, keras omite el cálculo del último batch si este no esta completo o de los mismos elementos  
<https://github.com/tensorflow/tensorflow/issues/38596>
- ▶ Problema de desvanecimiento de gradientes la precisión manejada por keras es superior a la precisión manejada por Gradient tape

## Gradient tape Vs model.fit()

Hay que tomar en cuenta que la implementación en keras complementa detalles que no se autoajustan en TF2 y muchos problemas se resuelven especificando detalladamente las funciones

## Gradient tape Vs model.fit()

Al parecer el problema en errores de cómputo en gradient tape y keras aparecen cuando se utiliza la función **softmax** esto solo pasa cuando se ocupan funciones decoradas es decir que se tiene `@tf.function`

<https://github.com/tensorflow/tensorflow/issues/32895>

## Gradient tape Vs model.fit()

Para solucionar el problema de softmax es recomendable realizar el cálculo de probabilidades por una función de activación softmax y realizar el cálculo de la entropía especificando que se hace entre lógicos esto lo calcula activamente keras aun cuando no se especifique

```
tf.keras.losses.SparseCategoricalCrossentropy(  
    from_logits=True)
```

## Gradient tape Vs model.fit()

Este error cambia debido a las funciones del error y en algunas funciones dará mejores resultados keras y en otras tensorflow 2 en este momento no cuento con una lista especifica de cual función le da preferencia a que implementacion.

<https://github.com/tensorflow/tensorflow/issues/35585>

# Keras entendiendo que hacen las funciones

En gran medida keras si es una caja negra y para entenderlo lo único que se puede hacer es recorrer las implementaciones en github sin embargo hay algunos códigos que les podrán ayudar como referencia



# Adaptacion del learning rate

Tanto keras como TF puede adaptar el learning rate por medio del optimizador como es el caso de adams, sin embargo keras proporciona otros métodos para reducir el learning rate

[Ver link](#)

## **Dudas específicas: revolver el data set**

Revolver el dataset cada epoca. Si el dataset es de suficiente tamaño no debería generar un cambio ni beneficio sin embargo en dataset pequeños esto ayuda a que se reduzca el sobre entrenamiento.

## Dudas específicas: vectores de altas dimensiones

- ▶ Mucha más información muy detallada
- ▶ Tienden al sobre aprendizaje ya que permiten más memoria a la red y con esto aprender casos de memoria