

1. Investiga y explica con tus palabras cada uno de los siguientes conceptos (podés usar libros, videos, artículos, papers o apuntes de clase):

- ¿Qué es una red neuronal artificial?

Una red neuronal artificial es una red neuronal que es artificial porque no es real, es creada por nosotros, es decir es creada por el humano pero no de nacimiento sino que fue creado por nosotros (**Son nodos que se encargan de realizar una predicción a partir de datos que se le brindan a través de tensores que pueden estar más o menos abstractos dependiendo de la librería que se utiliza**)

- ¿Qué función cumplen las capas (entrada, ocultas, salida)?

Las funciones que cumplen las capas son:

Entrada: Reciben los datos que le mandamos (ej: pixeles de una imagen, números, etc).

Oculta: Se encargan de hacer los cálculos y las transformaciones inmediatas.

Salida: Devuelve la predicción o un mensaje final de los datos recibidos, como por ej: "gato", "spam", "positivo", etc.

- ¿Qué es un forward pass y por qué es importante?

Un "Forward Pass" (también conocido como "Forward Propagation") es la fase en la que una red neuronal procesa una entrada y genera una salida. Se lo puede entender como el "camino hacia adelante" desde la capa de entrada hasta la capa de salida. Es importante porque es el primer paso fundamental en el ciclo de entrenamiento de una red neuronal, después de que la red realiza una predicción mediante el forward pass, esta salida se compara con el valor real para calcular el error o "loss"

- ¿Qué es una función de activación? Nombra al menos tres y explica en qué se diferencian.

Una función de activación es una función matemática que va a decidir si una neurona está activa o no, es decir, cuanta cantidad de la información recibida debe pasar a la siguiente capa. Sin una función de activación, la red neuronal no podría aprender relaciones complejas ni resolver problemas como clasificación lineal.

**Relu:** Trabaja con datos de entrenamiento más complejo donde hay mucha información

**Sigmoid:** Trabaja con números binarios dando una respuesta como 1 o 0

**Softmax:** Se utiliza en la salida de una red neuronal y es multiclase haciendo que todos los resultados de analizar la información den como resultado 1.

- ¿Qué es una función de pérdida (loss function) y para qué se utiliza?

El loss o función de pérdida mide que tan mal está prediciendo una red neuronal. Se encarga de decir a la red en cuanto se está equivocando.

compara la salida predicha por la red ( $\hat{y}$ ) con el valor real (y) y devuelve un número:

- . Si el número es grande, la predicción es mala.
- . Si el número es cero o pequeño, la predicción es buena.
- ¿Qué es la backpropagation? ¿Cómo se relaciona con el cálculo de derivadas?

La Backpropagation es un algoritmo clave que le permite a una red neuronal aprender, funcionando como su sistema de corrección de errores, después de que la red hace una predicción y se calcula el error (o "loss"), la Backpropagation calcula cuánto contribuyó cada "peso" al error. Esto se relaciona con el cálculo de derivadas porque utiliza "gradientes" para saber la dirección y la intensidad en que deben ajustarse los pesos y sesgos.

## 2. Elegí un optimizador (por ejemplo: Adam o SGD) e investiga:

Elegimos el optimizador Adagrad

- ¿Quién lo propuso? (Nombre y año si está disponible)

Lo propusieron **John Duchi, Elad Hazan y Yoram Singer** en el año 2011

**"Adaptive Subgradient Methods for Online Learning and Stochastic Optimization"**

- ¿En qué tipo de problemas suele usarse?

Se utiliza principalmente en problemas de aprendizaje automático donde los datos son dispersos, como en el procesamiento de lenguaje natural (NLP), la clasificación de texto, la recomendación de sistemas y el aprendizaje en redes neuronales con características de alta dimensionalidad. Su principal ventaja es que adapta automáticamente la tasa de aprendizaje para cada parámetro, asignando tasas más pequeñas a parámetros con gradientes frecuentes y mayores a aquellos con gradientes menos frecuentes, lo que lo hace especialmente efectivo en conjuntos de datos con características esparsas o cuando los gradientes varían significativamente en magnitud.

- ¿Cuáles son sus ventajas y desventajas?

Como ventajas:

- tiene una tasa de aprendizaje adaptativa que significa que los parámetros que cambian mucho se actualizan menos y los que cambian poco se actualizan más
- Bueno para datos dispersos como por ejemplo para modelos de lenguaje natural

Como desventajas:

- No es recomendable usarlo para modelos grandes o para entrenamientos largos
- Una vez que deja de aprender se suele quedar atascado

## 3. Reflexión crítica:

¿Por qué pensás que el proceso de entrenamiento en redes neuronales necesita tantas etapas?

Creo que necesita varias etapas porque hay que asegurarse de que los datos de entrenamiento pasen por un camino más limpio para obtener una salida esperada, ya sea limpiando los sets de datos, ajustando tamaños de datos de entrenamiento como los de testing, también verificar que los datos de perdida sean los adecuados y evitar anomalías en el entrenamiento de la red neuronal.

¿Qué pasaría si no existiera el paso de retropropagación?

Quizás cuando el entrenamiento no salga como esperamos los datos de salida del modelo estén más sesgados por datos que no son beneficiosos, por lo tanto el modelo respondería pero de forma errónea

También el aprendizaje sería muy lento o quizás inutil porque se tendría que usar métodos básicos como prueba y error, lo que haría que el entrenamiento sea muy impreciso o directamente inviable en redes grandes.

La razón por la que debe de necesitar tantas etapas es porque aprende de poco y en cada etapa la red ajusta los pesos y sesgos para acercarse a la solución correcta. Esto se repite muchas veces (épocas) porque los datos al ser muchos o complejos se necesitan verlos varias veces para reconocer patrones, minimizar errores , entre otros.