1. 1 punto

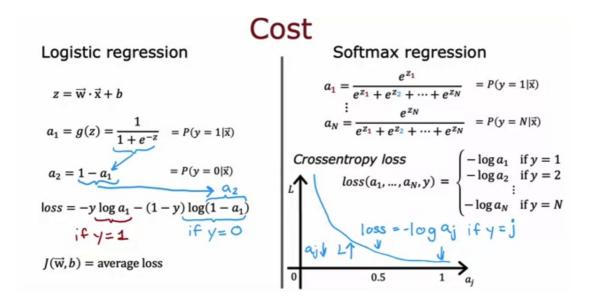
## Softmax regression (4 possible outputs)

Para una tarea de clasificación multiclase que tiene 4 salidas posibles, la suma de todas las activaciones suma 1. Para una tarea de clasificación multiclase que tiene 3 salidas posibles, la suma de todas las activaciones debe sumar....

- Menos que 1
- Más de 1
- ·
- O Variará, dependiendo de la entrada x.
- Correcto ¡Sí! La suma de todas las activaciones de softmax debe sumar 1. Una forma de ver esto es que si $y^{Con_1}=10, y^{Con_2}=20, y^{Con_3}=30,$  entonces la suma de $a_1+a_2+a_3$ es igual a  $\frac{y^{Con_1}+y^{Con_2}+y^{Con_3}}{y^{Con_1}+y^{Con_2}+y^{Con_3}}$  que es 1

1/1 punto

2.



Para la clasificación multiclase, la pérdida de entropía cruzada se usa para entrenar el modelo. Si hay 4 clases posibles para la salida, y para un ejemplo de entrenamiento en particular, la verdadera clase del ejemplo es la clase 3 (y=3), entonces, ¿a qué se simplifica la pérdida de entropía cruzada? [Sugerencia: esta pérdida debería disminuir cuando $a_3$ se hace más grande.]

$$\bigcirc$$
 -  $log(un_3)$ 

$$-log(un_1) + -logramo(un_2) + -logramo(un_3) + -logramo(un_4)$$

## ✓ Correcto

Correcto. Cuando la etiqueta verdadera es 3, entonces la pérdida de entropía cruzada para ese ejemplo de entrenamiento es solo el negativo del logaritmo de la activación de la tercera neurona del softmax. Todos los demás términos de la ecuación de pérdida de entropía cruzada ( $-log(un_1), -log(un_2), y_{--}$   $log(un_1)$ ) son ignorados

3. 1 / 1 punto

## MNIST (more numerically accurate)

```
import tensorflow as tf
from tensorflow.keras import Sequential
from tensorflow.keras.layers import Dense
model = Sequential([
    Dense(units=25, activation='relu')
    Dense(units=15, activation='relu')
    Dense(units=10, activation='linear'))]

loss
from tensorflow.keras.losses import
    SparseCategoricalCrossentropy
model.compile(...,loss=SparseCategoricalCrossentropy(from_logits=True))

fit
model.fit(X,Y,epochs=100)

predict
logits = model(X)
    f_x = tf.nn.softmax(logits)
```

Para la clasificación multiclase, la forma recomendada de implementar la regresión softmax es establecer from\_logits=True en la función de pérdida, y también definir la capa de salida del modelo con...

- una activación 'lineal'
- una activación 'softmax'
  - ✓ Correcto

¡Sí! Establezca la salida como lineal, porque la función de pérdida maneja el cálculo del softmax con un método más estable numéricamente.