

Examen intermedio
Redes neuronales, período 2018–2.
Profesor: Julio Waissman Vilanova.

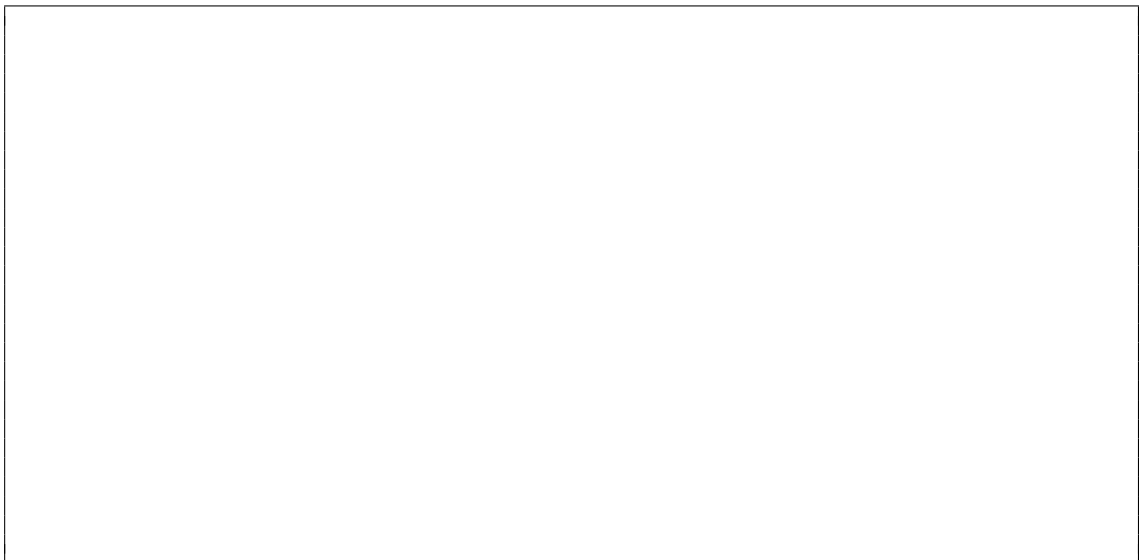
Nombre: _____

1. (10 puntos) Responde falso o verdadero

- (a) ____ Una red neuronal es un aproximador universal.
- (b) ____ Si utilizamos el método de inercia (o momento) en el algoritmo de descenso de gradiente, y la tasa de aprendizaje es muy pequeña, el valor de la función de pérdida siempre disminuye en cada paso de entrenamiento.
- (c) ____ Cuando la cantidad de datos no es muy grande, la estrategia de entrenamiento con *minibatch* es mejor que utilizar el entrenamiento por lotes (*batch*).
- (d) ____ La técnica de *Dropout* tiene como objetivo evitar el sobreaprendizaje.
- (e) ____ Es posible ajustar la función $f(x) = 1/y$ usando una red neuronal.
- (f) ____ El método de entrenamiento *Adam* modifica la tasa de aprendizaje de cada neurona en forma independiente.
- (g) ____ Si utilizamos una capa *max pool* en una CNN, entonces ya no es posible utilizar el método de entrenamiento de *backpropagation*.
- (h) ____ Siempre es preferible utilizar la función de activación logística para las capas ocultas, a menos que la cantidad de neuronas lo hagan prohibitivo computacionalmente.
- (i) ____ Una de las características más importantes de las arquitecturas de aprendizaje profundo es el hecho que se comparten parámetros ya sea en forma espacial o en forma temporal.

2. (20 puntos) Responde a las preguntas.

- (a) Dibuja la forma de la función de activación de una unidad ReLU.



- (b) ¿Porque el método de parada temprana previene el sobreaprendizaje? ¿Funcionaría el método en una regresión lineal?

- (c) ¿Que pasaría con el aprendizaje si se inicializan todos los pesos de la red neuronal con el mismo valor?

- (d) ¿Porque en las arquitecturas profundas las capas (o las unidades) deben de contener siempre una función no lineal?

3. (20 puntos puntos) Consideremos una red neuronal con 2 neuronas en la capa de entrada, dos neuronas en la primer capa oculta, dos neuronas en la segunda capa oculta y una neurona en la capa de salida. Todas las neuronas utilizan una función de activación logística. Las matrices de peso de la red neuronal están dadas por:

$$W^{(1)} = \begin{bmatrix} 2.0 & -1.0 \\ 0.5 & 1.0 \end{bmatrix}, \quad W^{(2)} = \begin{bmatrix} -2.0 & 4.0 \\ 0.2 & 0.3 \end{bmatrix}, \quad W^{(3)} = \begin{bmatrix} 1.0 & 1.0 \end{bmatrix},$$

y los sesgos por:

$$b^{(1)} = [-1.0, 1.0]^T, \quad b^{(2)} = [-1.0, 0.1]^T, \quad b^{(3)} = 1.0$$

(a) La salida para una entrada $x = [1, 1]^T$ es

.

(b) El criterio de pérdida para este tipo de red neuronal es:

.

(c) El costo para este unico dato de entrada es

.

(d) Si para $x = [1, 1]^T$ sabemos que le corresponde la clase 1, entonces calcula, utilizando *backpropagation*, las matrices de gradientes $\nabla_{W^l} J(W)$ para $l = 1, 2, 3$.

4. (30 puntos) Consideremos una red neuronal hacia adelante densa, en la cual todas las neuronas de la capa anterior se encuentran completamente conectadas con las neuronas de la capa siguiente.

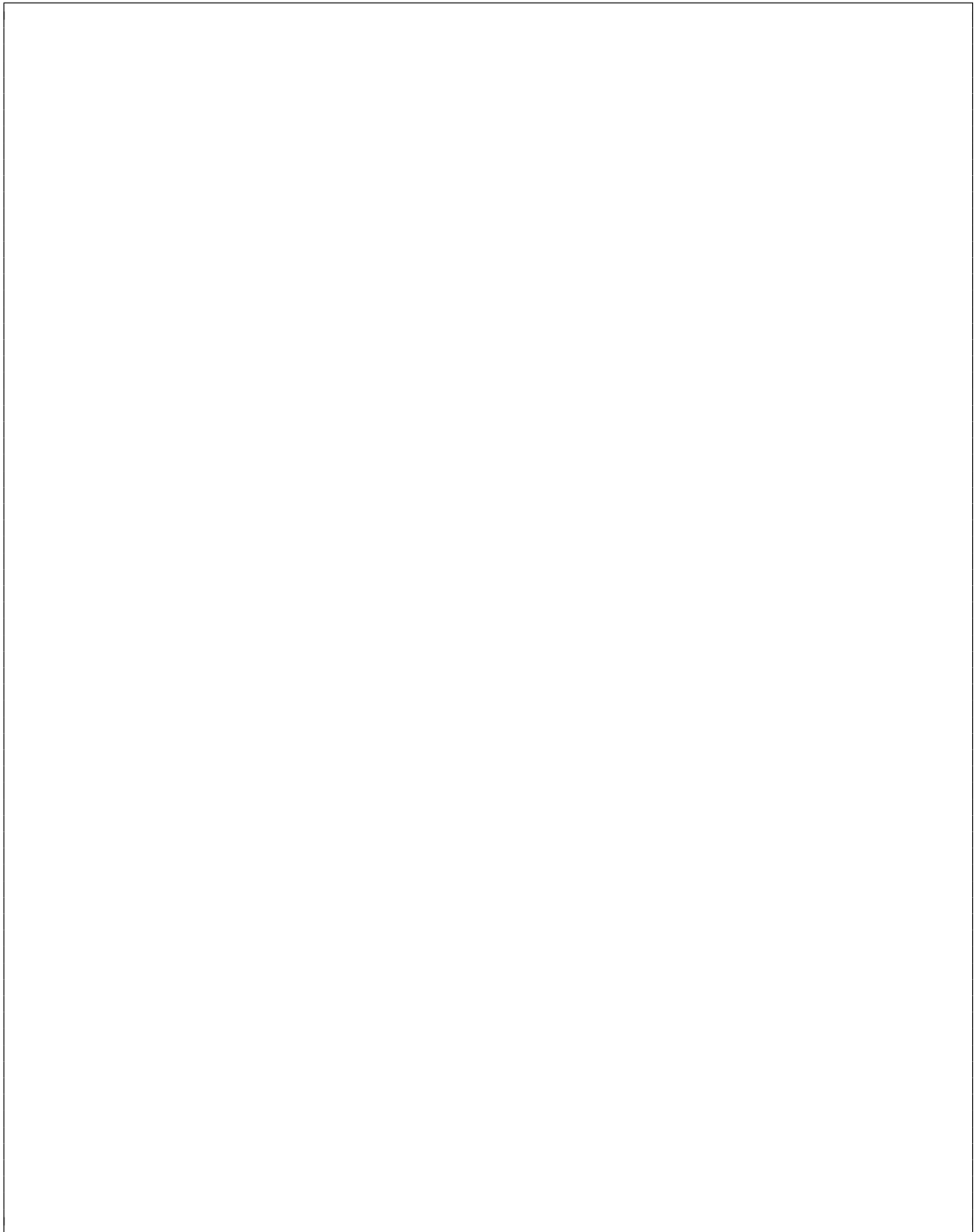
- (a) Considera una red neuronal con una sola capa oculta, con 5 neuronas de entrada, 3 neuronas en la capa oculta, y una neurona en la capa de salida. ¿Cual es el número total de operaciones requeridas para realizar un solo *epoch* de una operación del algoritmo de *backpropagation*, si contamos con un solo dato en el conjunto de aprendizaje?

Solo vamos a contar como operaciones los productos de la forma $w_{i,j}^{(l)} a_j^{(l-1)}$, $w_{i,j}^{(l)} a_i^{(l)}$ y $a_i^{(l-1)} \delta_j^{(l)}$ manteniendo la notación utilizada en el curso.

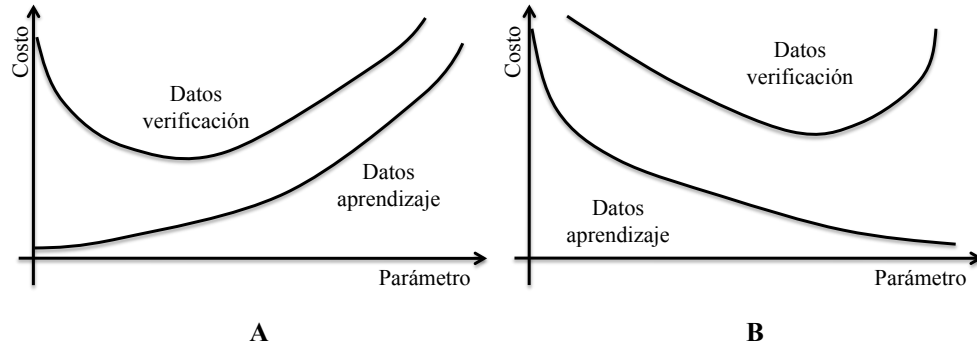
- (b) Vamos a llamar *nodo* a cualquier unidad de una red neuronal, independientemente si es entrada, salida, o neuronas de la capa oculta. Consideremos una red con 10 nodos de entrada, un solo nodo de salida, y 36 nodos en capas ocultas. Los nodos de las capas ocultas se pueden distribuir como mejor convenga, y en tantas capas ocultas como se quiera, siempre que los nodos de la capa $l - 1$ se encuentren completamente conectados con los nodos de la capa l .

1. ¿Cuál es la topología de la red (capas y número de nodos por capa) que genere el menor número de parámetros de aprendizaje (pesos y sesgos) posibles?
2. ¿Cuál es la topología de la red que implica tener el mayor número de parámetros de aprendizaje posibles?

- (c) Supongamos ahora que tenemos una red con 10 neuronas de entrada, 20 neuronas en la primer capa oculta, 20 neuronas en una segunda capa oculta y 1 neurona de salida, donde todas las neuronas tienen una función de activación lineal. Demuestre que esta red neuronal es equivalente a otra red que solamente tenga las 10 neuronas de la capa de entrada y la neurona de la capa de salida directamente.



5. (20 puntos) Considera las siguientes figuras A y B donde se presenta el costo en los datos de validación y verificación respecto a el valor de un parámetro. Cada punto de las curvas representa el valores el costo después de haber entrenado a una red neuronal completamente con el valor dado de dicho parámetro.



Asigna cual es la curva (A o B) que podría corresponder al variar cada uno de los siguientes parámetros:

- (a) ___ Número de neuronas en la capa oculta (asumiendo una sola capa oculta).
- (b) ___ Número de capas ocultas en una red neuronal.
- (c) ___ Valor de la tasa de aprendizaje α .
- (d) ___ Umbral θ entre 0 y 1 por el cual se considera que un objeto pertenece a la clase 1, si la salida de la red neuronal es logística (por default $\gamma = 0.5$).
- (e) ___ Número de epochs utilizados en el aprendizaje.