Quiz 2: Neural Networks (part 1) — Answer Key

E190BO *Instructor copy — 16 points total*

1. (4 pts, 1 pt each) *Linear-operation shapes,* $s_0 = 4$*,* $s_1 = 5$*,* $s_2 = 3$*.*

- $W^{(1)}$: $\mathbb{R}^{4\times5}$
- $\bar{b}^{(1)}$: $\mathbb{R}^{1\times5}$
- $W^{(2)}$: $\mathbb{R}^{5\times3}$
- $\bar{b}^{(2)}$: $\mathbb{R}^{1\times3}$

2. (3 pts) *Batched shapes,* $N = 32$*.*

(a) $A^{(0)}$: $\mathbb{R}^{32\times4}$

(b) $Z^{(1)}$: $\mathbb{R}^{32\times5}$

(c) The same bias vector is added to every example in the batch — it is *broadcast* across the N rows — so there is still only one bias per hidden unit, giving $\mathbb{R}^{1\times5}$ regardless of N.

3. (2 pts) *Nonlinearity.*

Blank: a **nonlinear activation function** (e.g. ReLU).

Why: without it, stacking linear layers just gives another linear function, so the whole network would collapse to a single linear model and could not learn nonlinear decision boundaries.

4. (3 pts) *Softmax.*

Softmax maps C real numbers to a **valid probability distribution**: all outputs are between 0 and 1 and they sum to 1, with larger inputs mapping to larger probabilities.

That is exactly what a classifier's last layer should produce — a predicted probability for each of the C classes — which can then be compared against the one-hot label with cross-entropy loss.

5. (4 pts) *Chain rule / computation graph for* $y = \log(x^2 + x + 3)$*.*

(a) $f(x) = x^2 + x + 3$, $g(u) = \log u$, so $y = g(f(x))$.

(b) Computation graph:

$$x \longrightarrow \boxed{f(\cdot)} \xrightarrow{z=f(x)} \boxed{g(\cdot)} \longrightarrow y$$

with local derivatives $\dfrac{dz}{dx} = f'(x)$ and $\dfrac{dy}{dz} = g'(z)$.

(c) By the chain rule, multiply the local derivatives along the graph:

$$\frac{dy}{dx} = \frac{dy}{dz} \cdot \frac{dz}{dx} = g'(z) \cdot f'(x) = \frac{1}{x^2 + x + 3} \cdot (2x + 1)$$