1 Forward path

$$\begin{aligned}
Z_1 &= W_1 \cdot X \\
A_1 &= ReLU(Z_1) && \equiv \text{'h' in code} \\
Z_2 &= W_2 \cdot A_1 && \equiv \text{'logp' in code} \\
A_2 &= sigmoid(Z_2) && \equiv \text{'p' in code}
\end{aligned}$$

2 Backward path

Since our final output of our forward calculations is a probability of sampling the action of going UP (=1), basically a coin toss, we can make use of the Bernoulli Distribution:

$$p(y, \theta) = \theta^y * (1 - \theta)^{1-y}$$

The log-likelihood function is:

$$logL(\theta) = \sum_{i=1}^{n} y_i * \log(\theta) + \sum_{i=1}^{n} (1 - y_i) * \log(1 - \theta)$$

Keep in mind that all our efforts during training focus on optimizing θ (represented by the 2-layer NN), in order to let us win as many games as possible. Our loss-function that we want to minimize is logL for n=1. θ is represented by A2 (or "p" in the code).

$$logL(\theta) = y * log(\theta) + (1 - y) * log(1 - \theta)$$

Calculate the partial derivate of logL wrt. W_2:

$$\begin{aligned}
\frac{\partial logL}{\partial W_2} &= \frac{\partial logL}{\partial A_2} * \frac{\partial A_2}{\partial W_2} \\
&= \frac{\partial logL}{\partial A_2} * \frac{\partial A_2}{\partial Z_2} * \frac{\partial Z_2}{\partial W_2} \\
&= \underbrace{(\frac{y}{A_2} - \frac{1-y}{1-A_2}) * (1 - A_2) * A_2}_{\text{'dlogps' in code}} * A_1
\end{aligned}$$

Calculate partial derivate of logL wrt. W_1:

$$\begin{aligned}
\frac{\partial logL}{\partial W_1} &= \frac{\partial logL}{\partial A_2} * \frac{\partial A_2}{\partial W_1} \\
&= \frac{\partial logL}{\partial A_2} * \frac{\partial A_2}{\partial Z_2} * \frac{\partial Z_2}{\partial W_1} \\
&= \frac{\partial logL}{\partial A_2} * \frac{\partial A_2}{\partial Z_2} * \frac{\partial Z_2}{\partial A_1} * \frac{\partial A_1}{\partial W_1} \\
&= \frac{\partial logL}{\partial A_2} * \frac{\partial A_2}{\partial Z_2} * \frac{\partial Z_2}{\partial A_1} * \frac{\partial A_1}{\partial Z_1} * \frac{\partial Z_1}{\partial W_1} \\
&= \underbrace{(\frac{y}{A_2} - \frac{1-y}{1-A_2}) * (1-A_2) * A_2}_{\text{'dlogps' in code}} * W_2 * \left\{ \begin{array}{ll} 0, & \text{for } Z_1 < 0 \\ 1, & \text{for } Z_1 > 0 \end{array} \right\} * X
\end{aligned}$$

For sampled action being y=1 (UP):

$$\begin{aligned}
\frac{\partial logL}{\partial W_2} &= (1-A_2) * A_1 \\
\frac{\partial logL}{\partial W_1} &= (1-A_2) * W_2 * \left\{ \begin{array}{ll} 0, & \text{for } Z_1 < 0 \\ 1, & \text{for } Z_1 > 0 \end{array} \right\} * X
\end{aligned}$$

For sampled action being down y=0 (DOWN):

$$\begin{aligned}
\frac{\partial logL}{\partial W_2} &= -A_2 * A_1 \\
\frac{\partial logL}{\partial W_1} &= -A_2 * W_2 * \left\{ \begin{array}{ll} 0, & \text{for } Z_1 < 0 \\ 1, & \text{for } Z_1 > 0 \end{array} \right\} * X
\end{aligned}$$