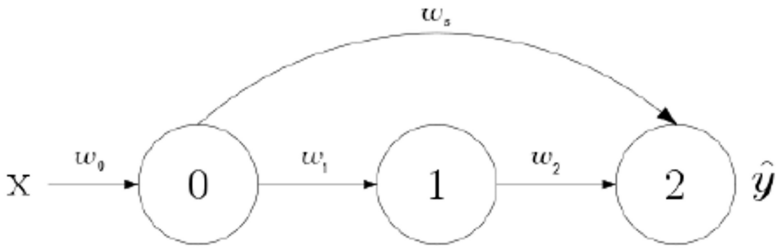
Backpropagation

Perform a forward and backward pass to calculate the gradients for the weights w_0, w_1, w_2, w_s in the following MLP. Each node represents one unit with a weight w_i, $i \in \{0, 1, 2\}$ connecting it to the previous node. The connection from unit 0 to unit 2 is called a **skip connection**, which means unit 2 receives input from two sources and thus has an additional weight w_s. The weighted inputs are added before the nonlinearity is applied. You can refer to this blog post for a more intuitive understanding of backpropagation. In case of multiple outgoing connections from a neuron (for eg, forward and skip connections), the final gradient can be computed by accumulating the incoming gradients via these multiple connections as a direct consequence of the chain rule.



We assume that we want to solve a regression task. We use an L1-loss $L(\hat{y}, y) = |y - \hat{y}|$ The nonlinearities for the first two units are rectified linear functions/units (ReLU): $g_0(z) = g_1(z) = \begin{cases} 0, z < 0 \\ z, else \end{cases}$.

We do not use a nonlinearity for the second unit: $g_2(z_2) = z_2$ and there are **no biases**.

Question Perform the backpropagation algorithm for the above network.