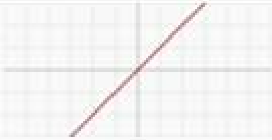
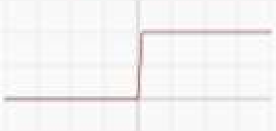
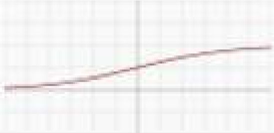
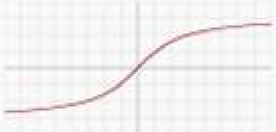
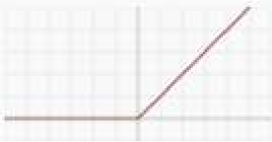
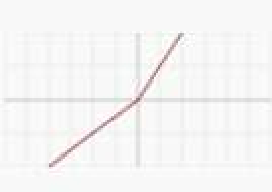
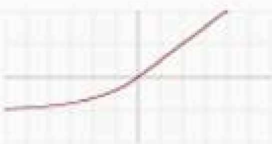
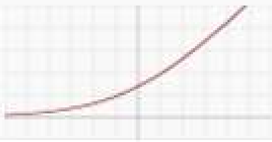
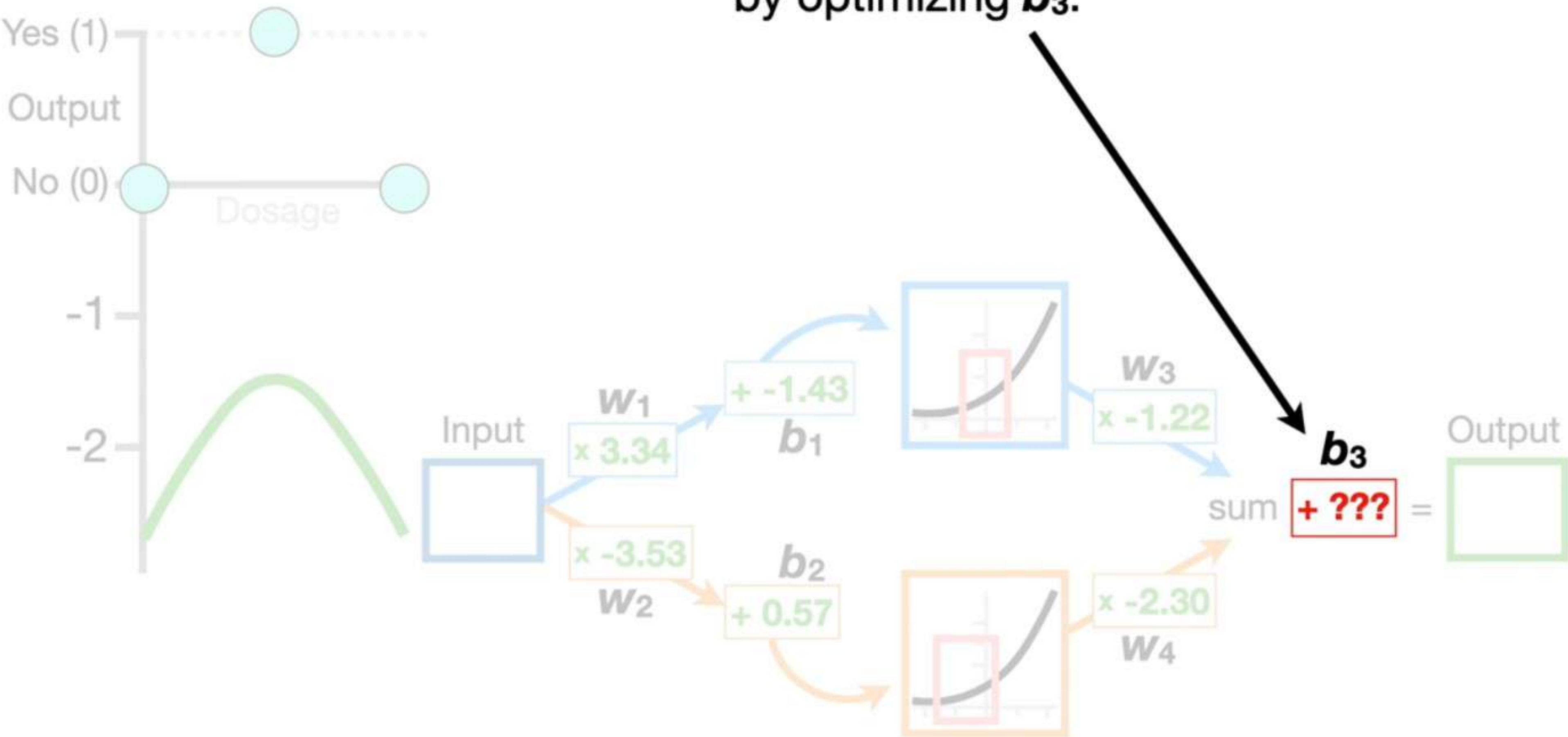
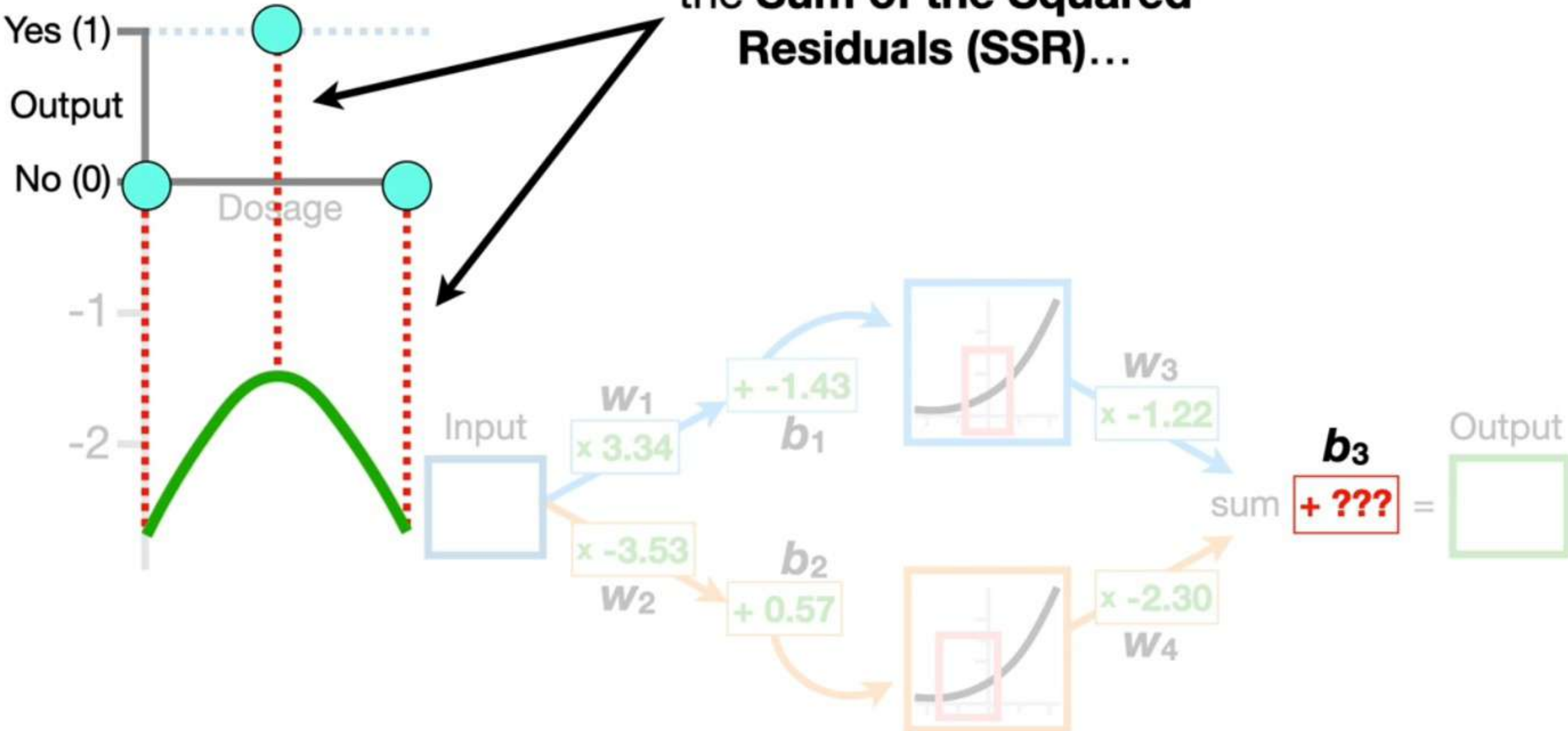
Activation functions with small ranges are usually used for solving classification problems.

Name	Plot	Equation	Derivative
Identity		$f(x) = x$	$f'(x) = 1$
Binary step		$f(x) = \begin{cases} 0 & \text{for} \quad x < 0 \\ 1 & \text{for} \quad x \geq 0 \end{cases}$	$f'(x) = \begin{cases} 0 & \text{for} \quad x \neq 0 \\ ? & \text{for} \quad x = 0 \end{cases}$
Logistic (a.k.a Soft step)		$f(x) = \frac{1}{1+e^{-x}}$	$f'(x) = f(x)(1-f(x))$
TanH		$f(x) = \tanh(x) = \frac{2}{1+e^{-2x}} - 1$	$f'(x) = 1 - f(x)^2$
ArcTan		$f(x) = \tan^{-1}(x)$	$f'(x) = \frac{1}{x^2+1}$
Rectified Linear Unit (ReLU)		$f(x) = \begin{cases} 0 & \text{for} \quad x < 0 \\ x & \text{for} \quad x \geq 0 \end{cases}$	$f'(x) = \begin{cases} 0 & \text{for} \quad x < 0 \\ 1 & \text{for} \quad x \geq 0 \end{cases}$
Parameteric Rectified Linear Unit (PReLU)[2]		$f(x) = \begin{cases} \alpha x & \text{for} \quad x < 0 \\ x & \text{for} \quad x \geq 0 \end{cases}$	$f'(x) = \begin{cases} \alpha & \text{for} \quad x < 0 \\ 1 & \text{for} \quad x \geq 0 \end{cases}$
Exponential Linear Unit (ELU)[3]		$f(x) = \begin{cases} \alpha(e^x - 1) & \text{for} \quad x < 0 \\ x & \text{for} \quad x \geq 0 \end{cases}$	$f'(x) = \begin{cases} f(x) + \alpha & \text{for} \quad x < 0 \\ 1 & \text{for} \quad x \geq 0 \end{cases}$
SoftPlus		$f(x) = \log_e(1+e^x)$	$f'(x) = \frac{1}{1+e^{-x}}$

Then we demonstrated the **Main Ideas** behind **Backpropgation** by optimizing ***b*₃**.

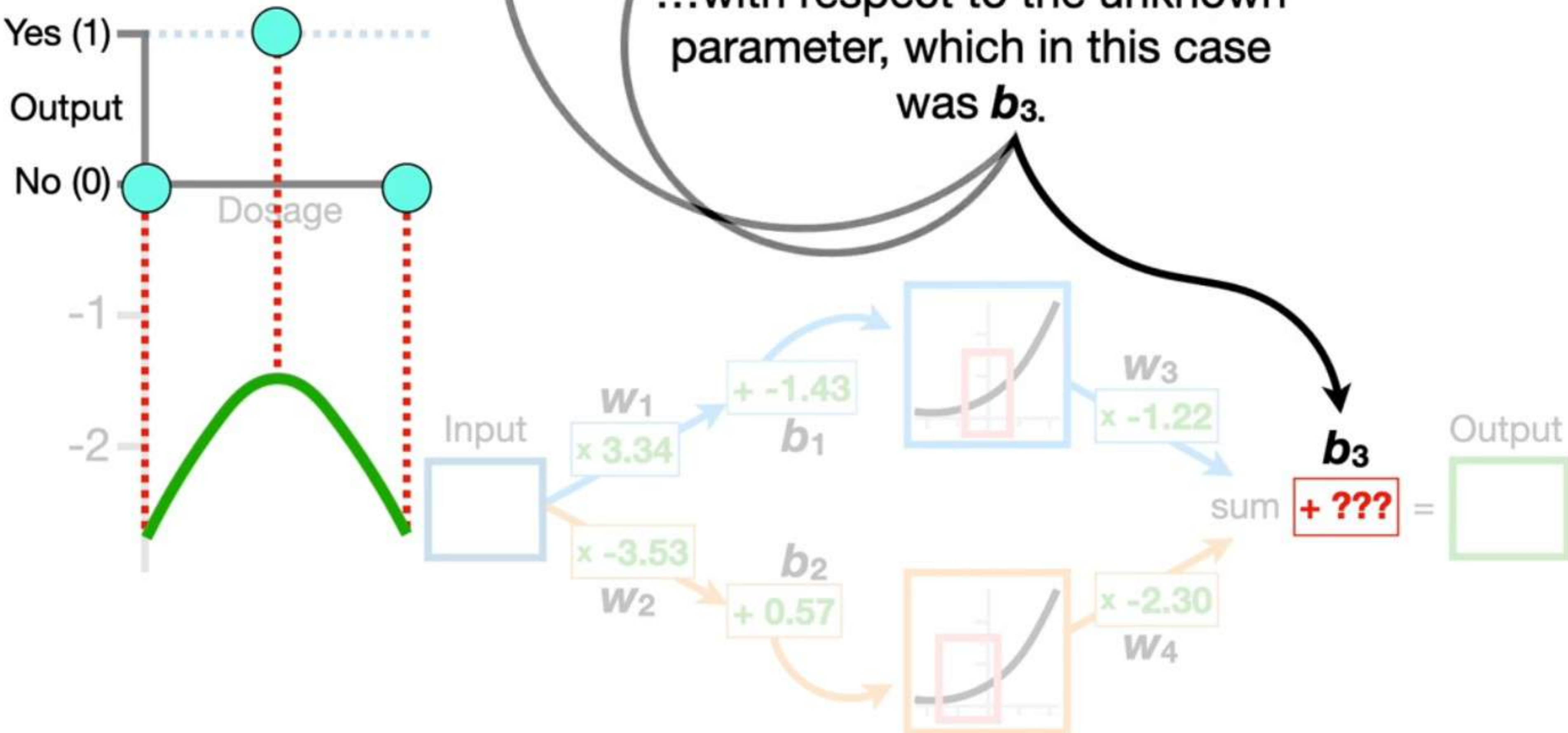


We first used **The Chain Rule** to calculate the derivative of the **Sum of the Squared Residuals (SSR)**…



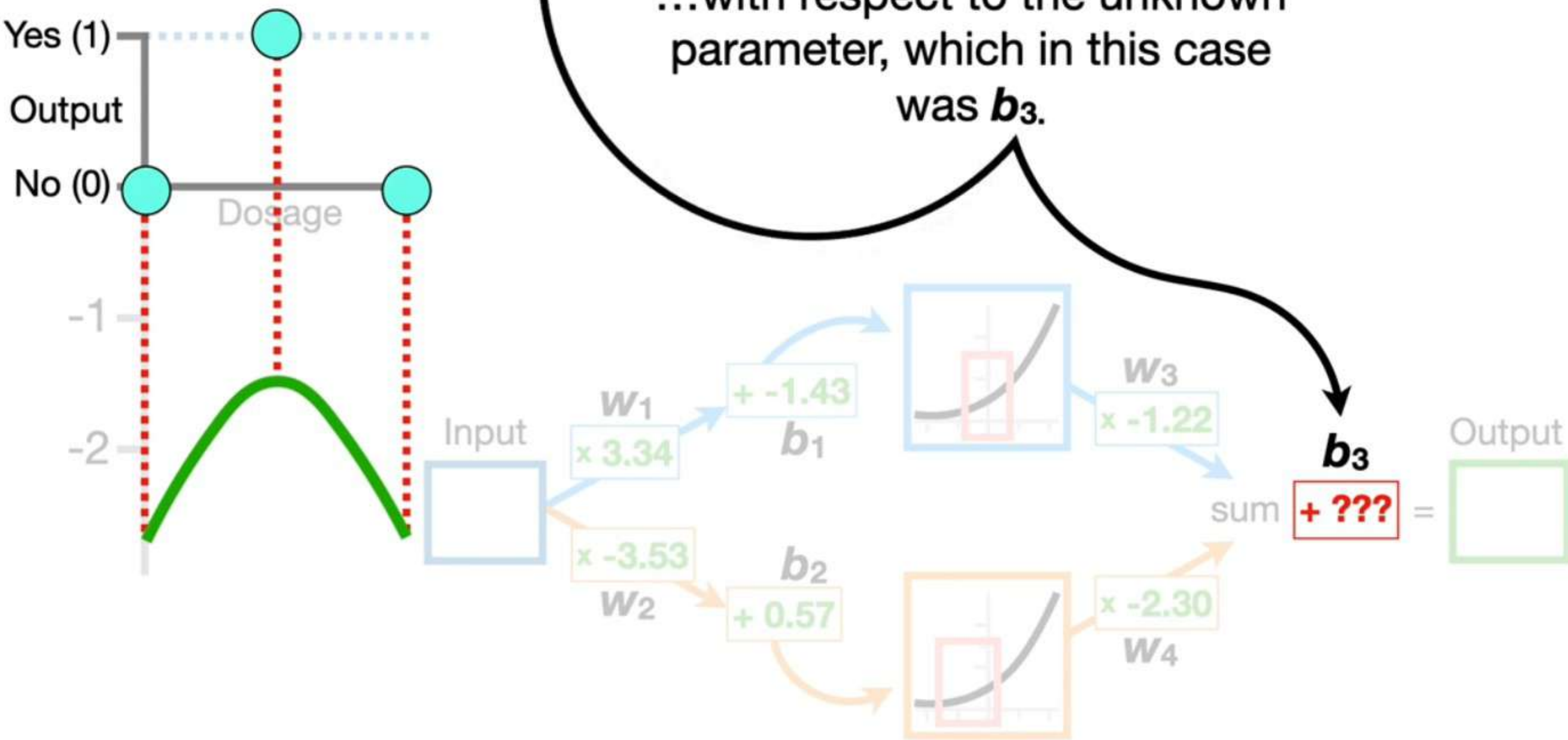
$$\frac{d\,SSR}{d\,b_3} = \frac{d\,SSR}{d\,Predicted} \times \frac{d\,Predicted}{d\,b_3}$$

...with respect to the unknown parameter, which in this case was b_3.



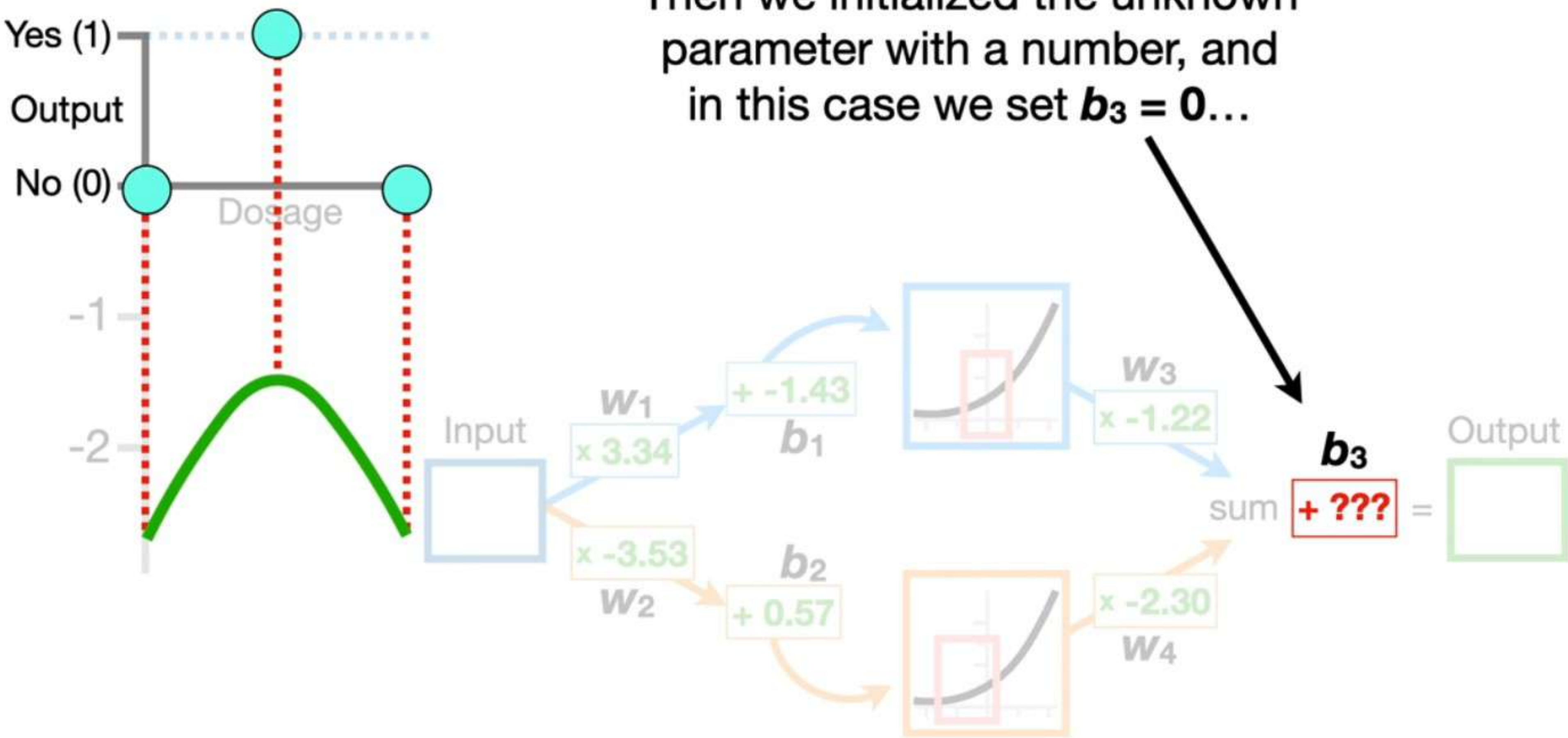
$$\frac{d\ SSR}{d\ b_3} = \sum_{i=1}^{n=3} -2 \times (\text{Observed}_i - \text{Predicted}_i) \times \mathbf{1}$$

…with respect to the unknown parameter, which in this case was $\boldsymbol{b_3}$.

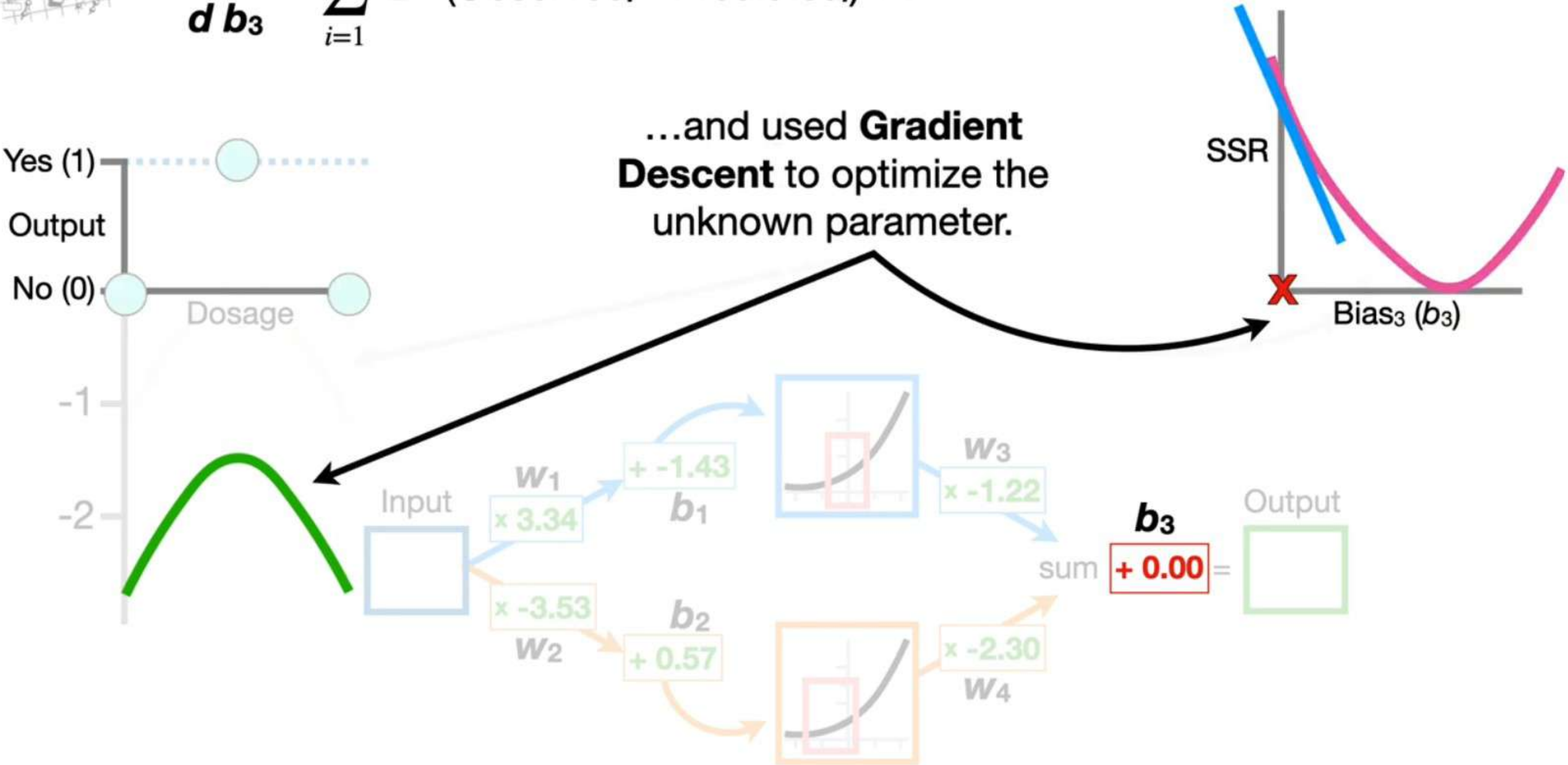


$$\frac{d\ SSR}{d\ b_3} = \sum_{i=1}^{n=3} -2 \times (\text{Observed}_i - \text{Predicted}_i) \times \mathbf{1}$$

Then we initialized the unknown parameter with a number, and in this case we set $b_3 = 0$…

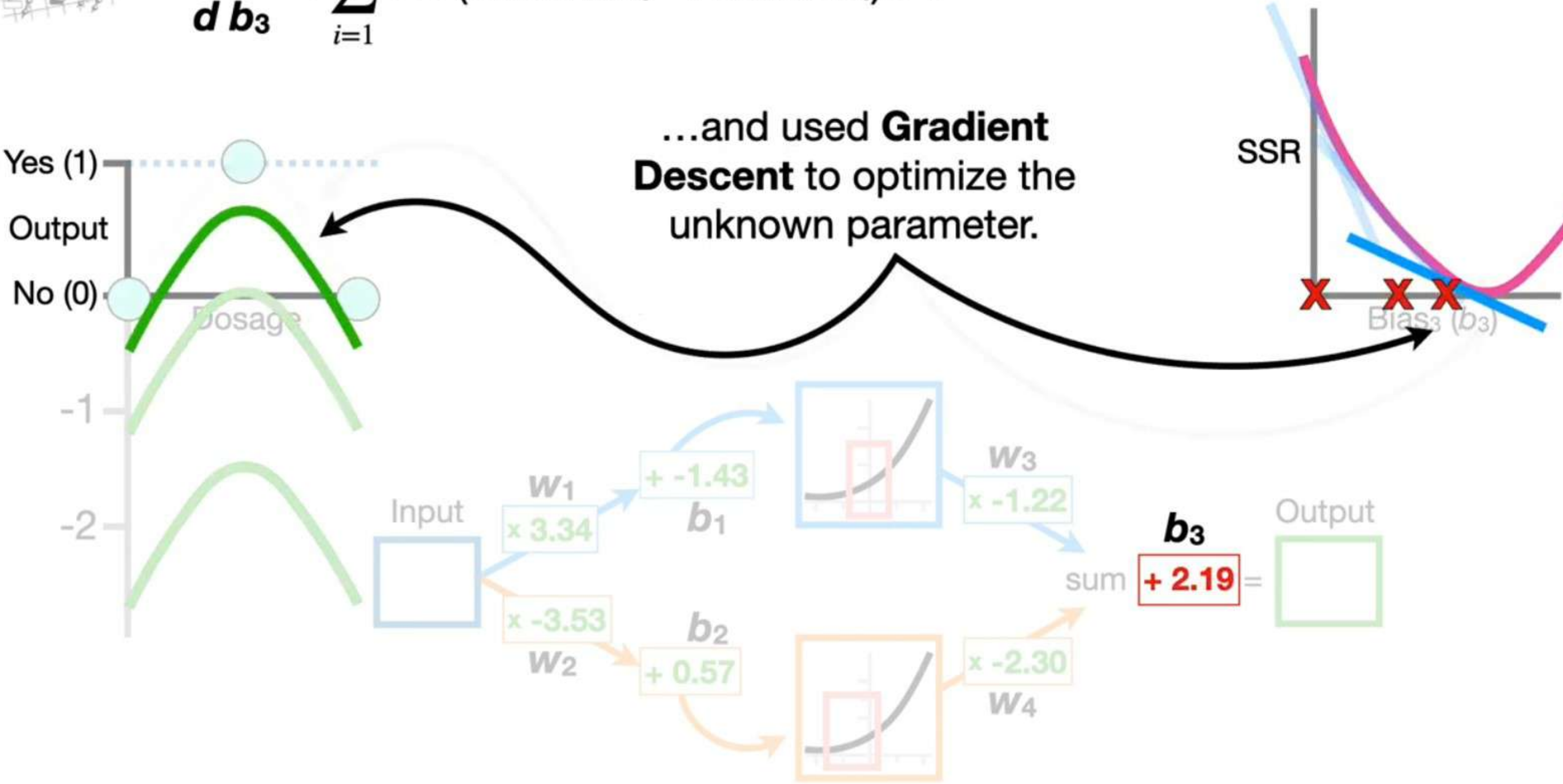


$$\frac{d\ SSR}{d\ b_3} = \sum_{i=1}^{n=3} -2 \times (\text{Observed}_i - \text{Predicted}_i) \times \mathbf{1}$$

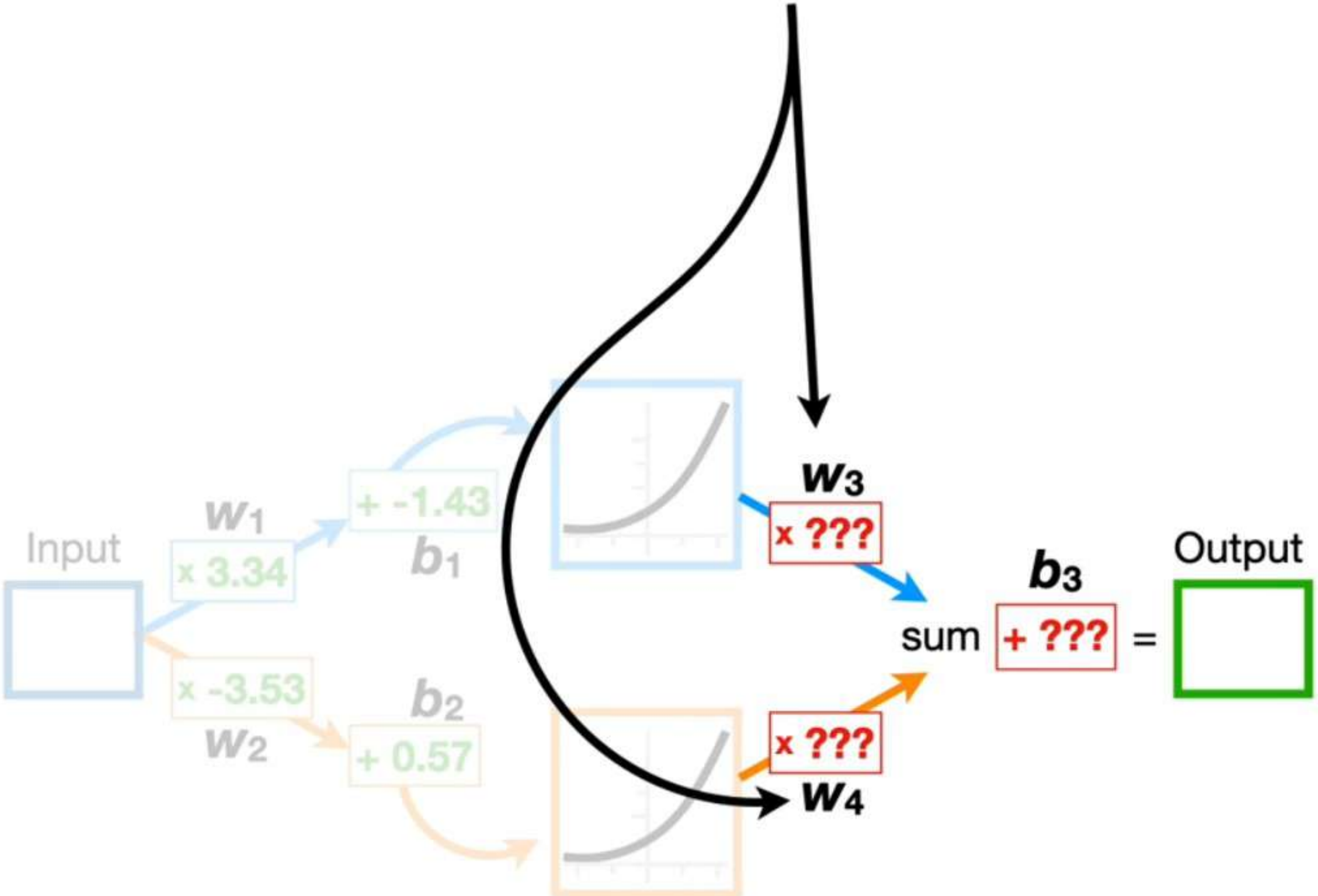


$$\frac{d\ SSR}{d\ b_3} = \sum_{i=1}^{n=3} -2 \times (\text{Observed}_i - \text{Predicted}_i) \times 1$$

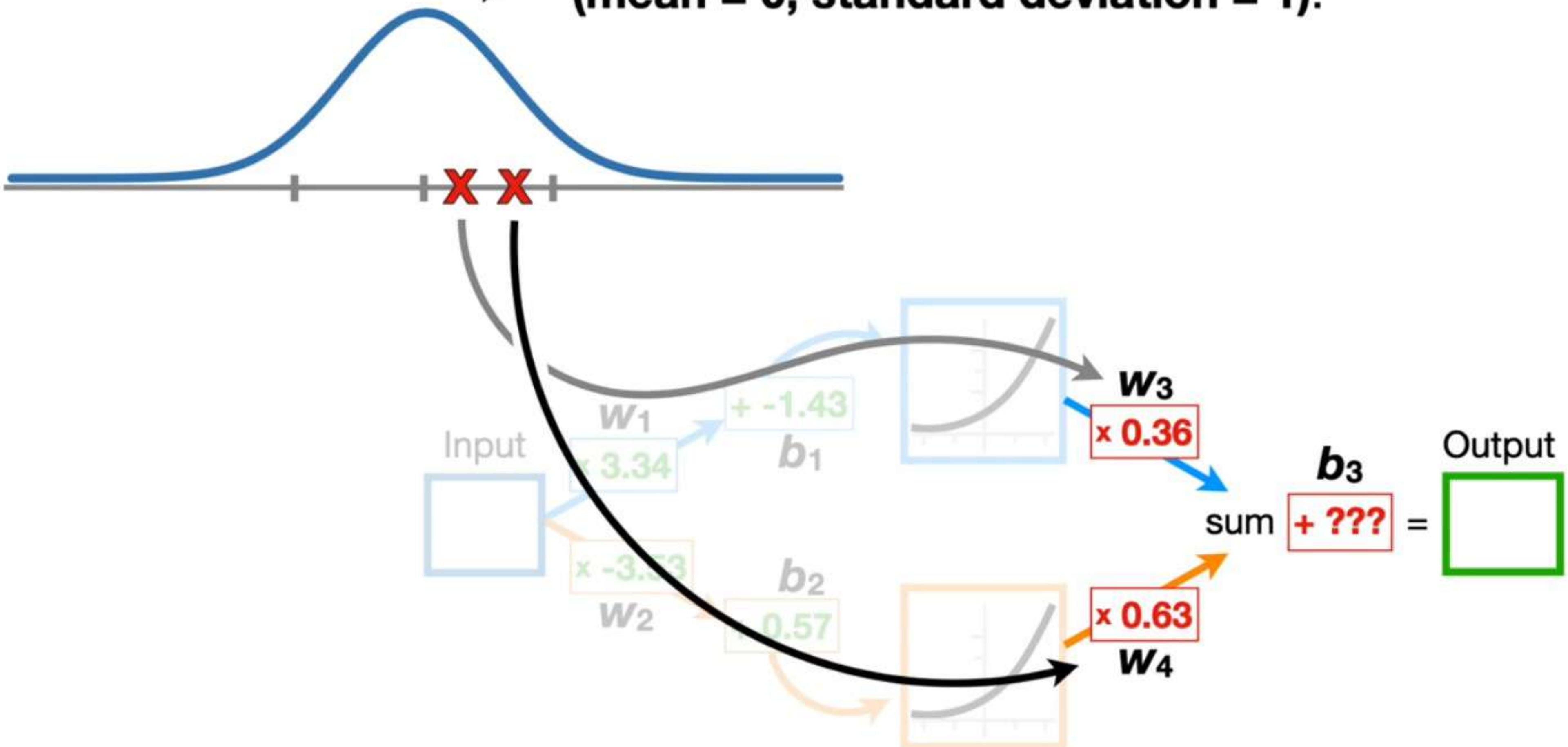
...and used **Gradient Descent** to optimize the unknown parameter.



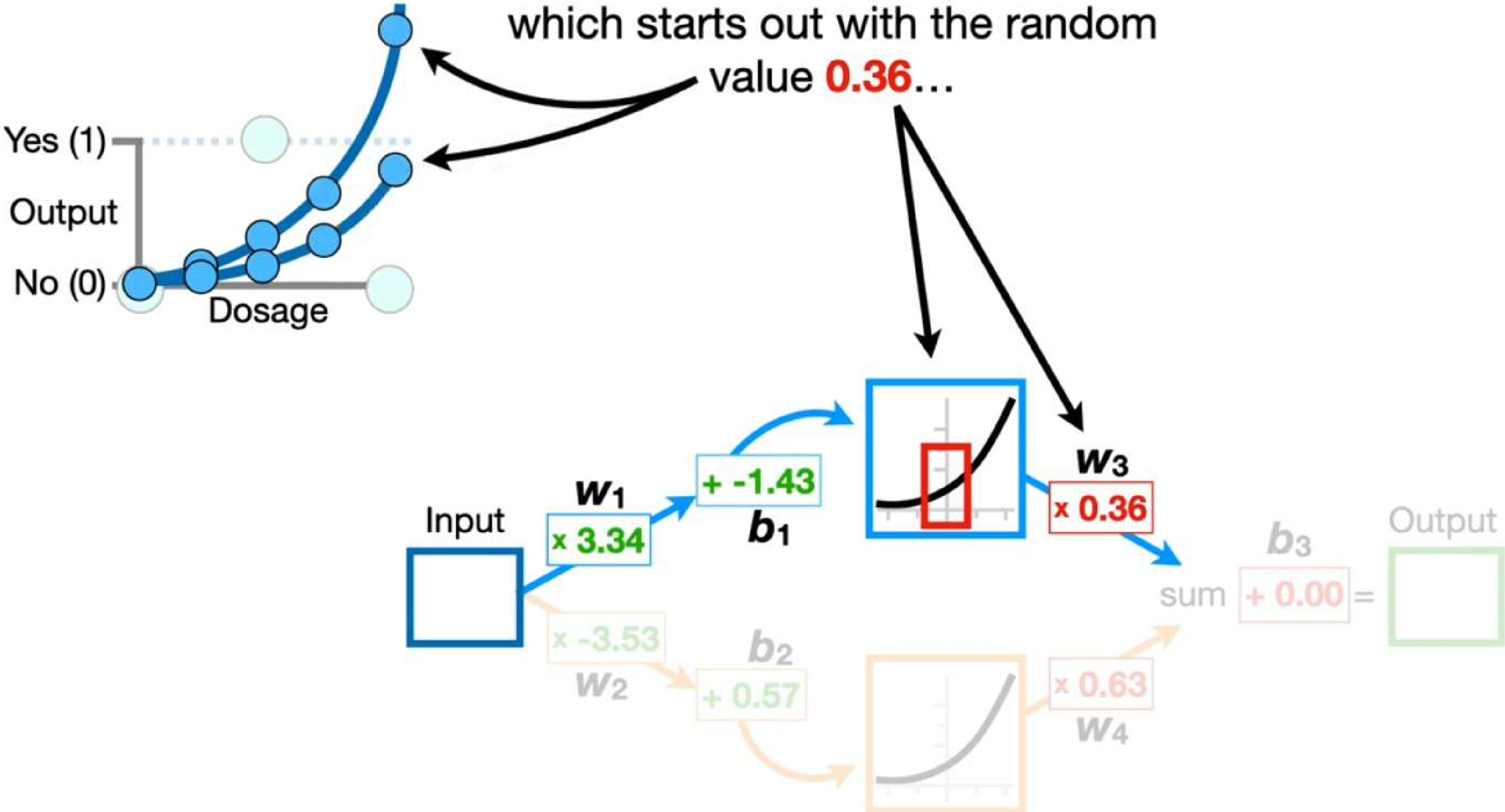
The first thing we do is initialize the **Weights**, ***w***$_3$ and ***w***$_4$, with random starting values…



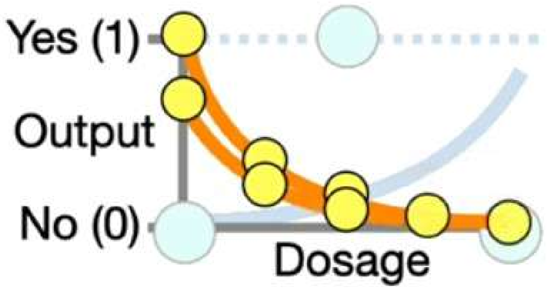
...and, in this example, that means we randomly select **2** values from a **Standard Normal Distribution (mean = 0, standard deviation = 1)**.

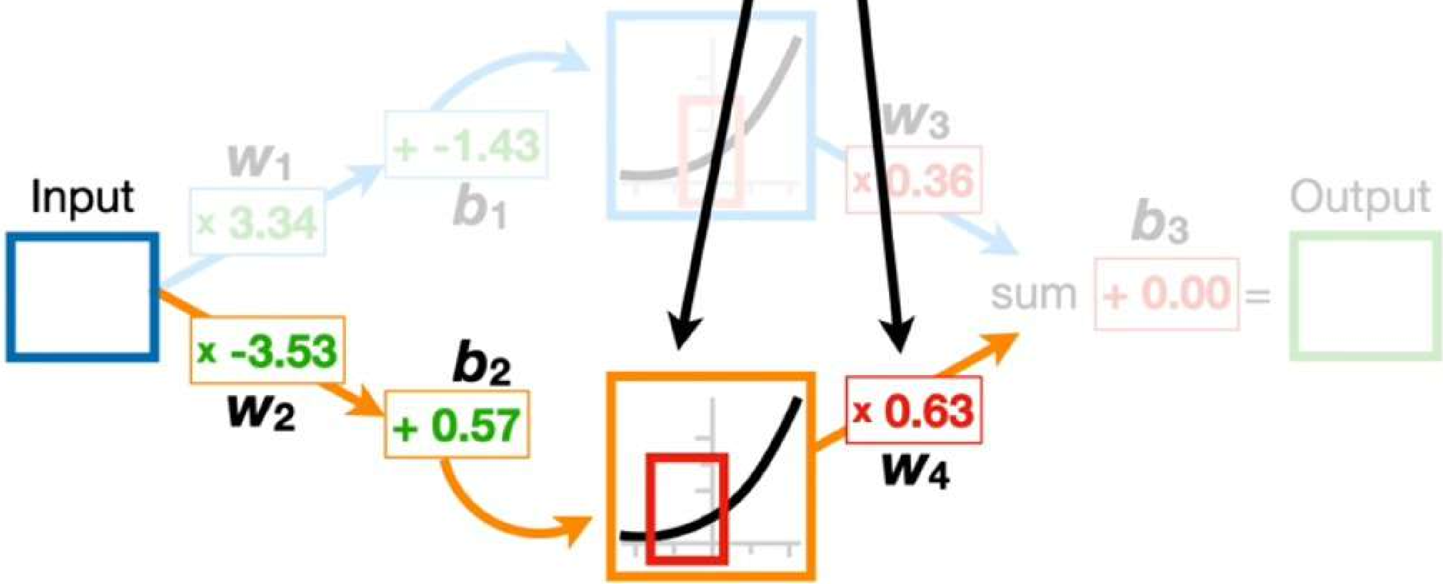


Now we multiply the y-axis coordinates on the **blue curve** by w_3, which starts out with the random value **0.36**…

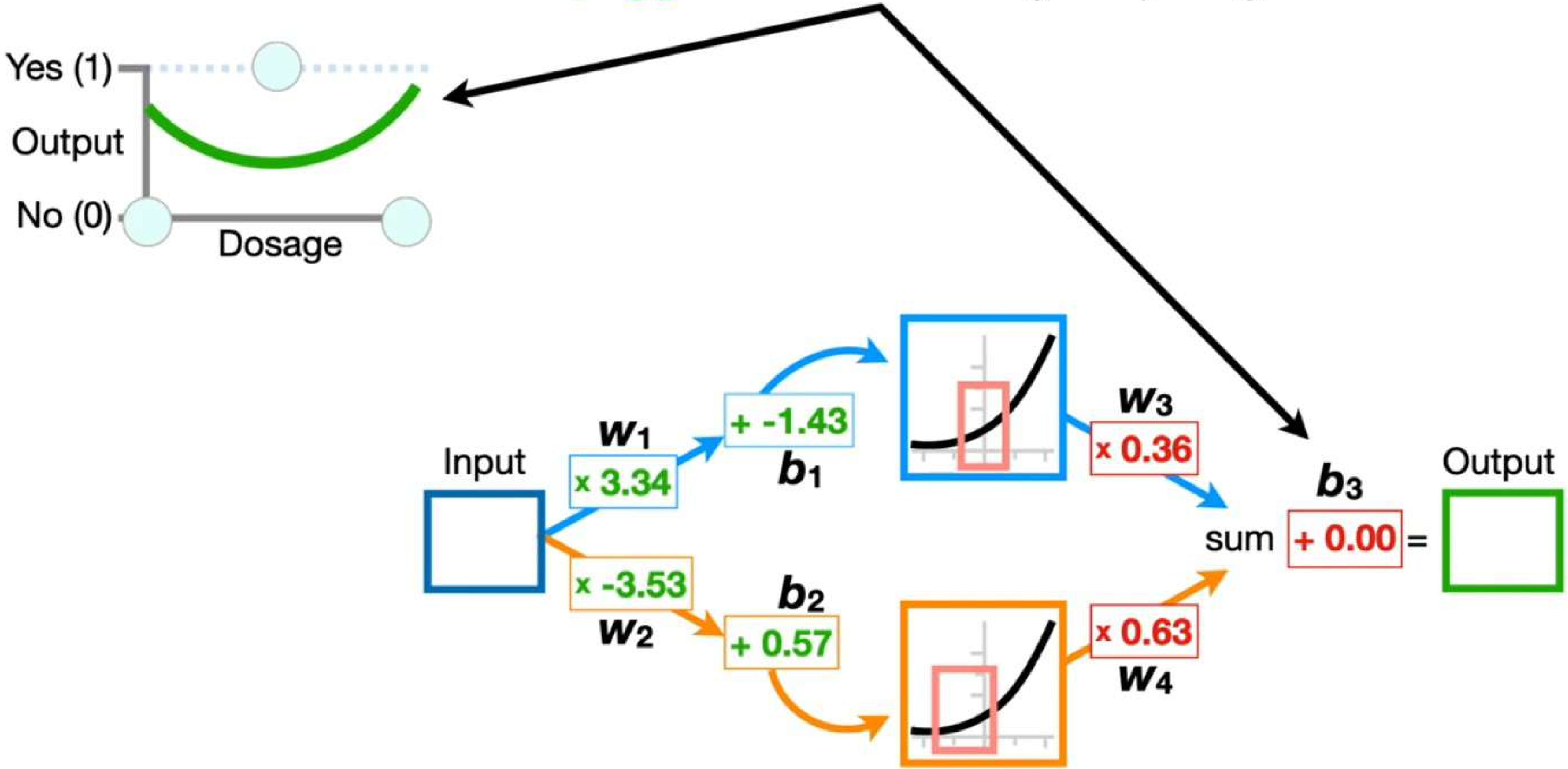


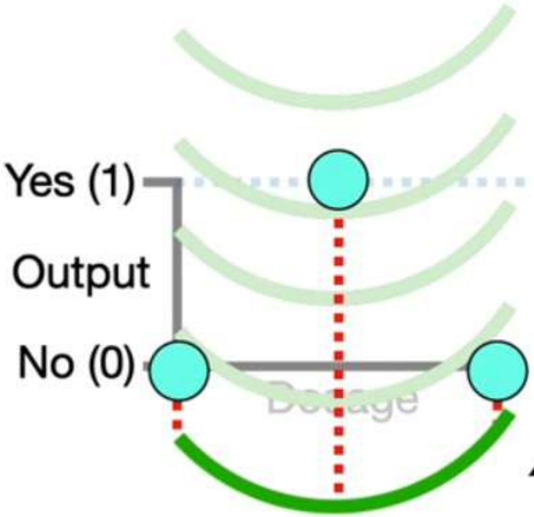
Now we multiply the y-axis coordinates on the orange curve by w_4, which starts with the random value 0.63…



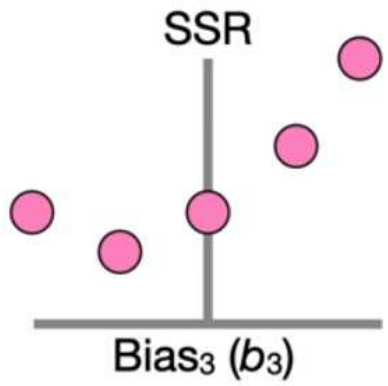


Lastly, since the initial value for b_3 is **0**, adding it to the y-axis values on the **green squiggle** does not change anything.



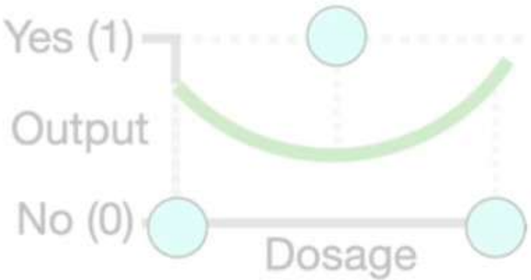


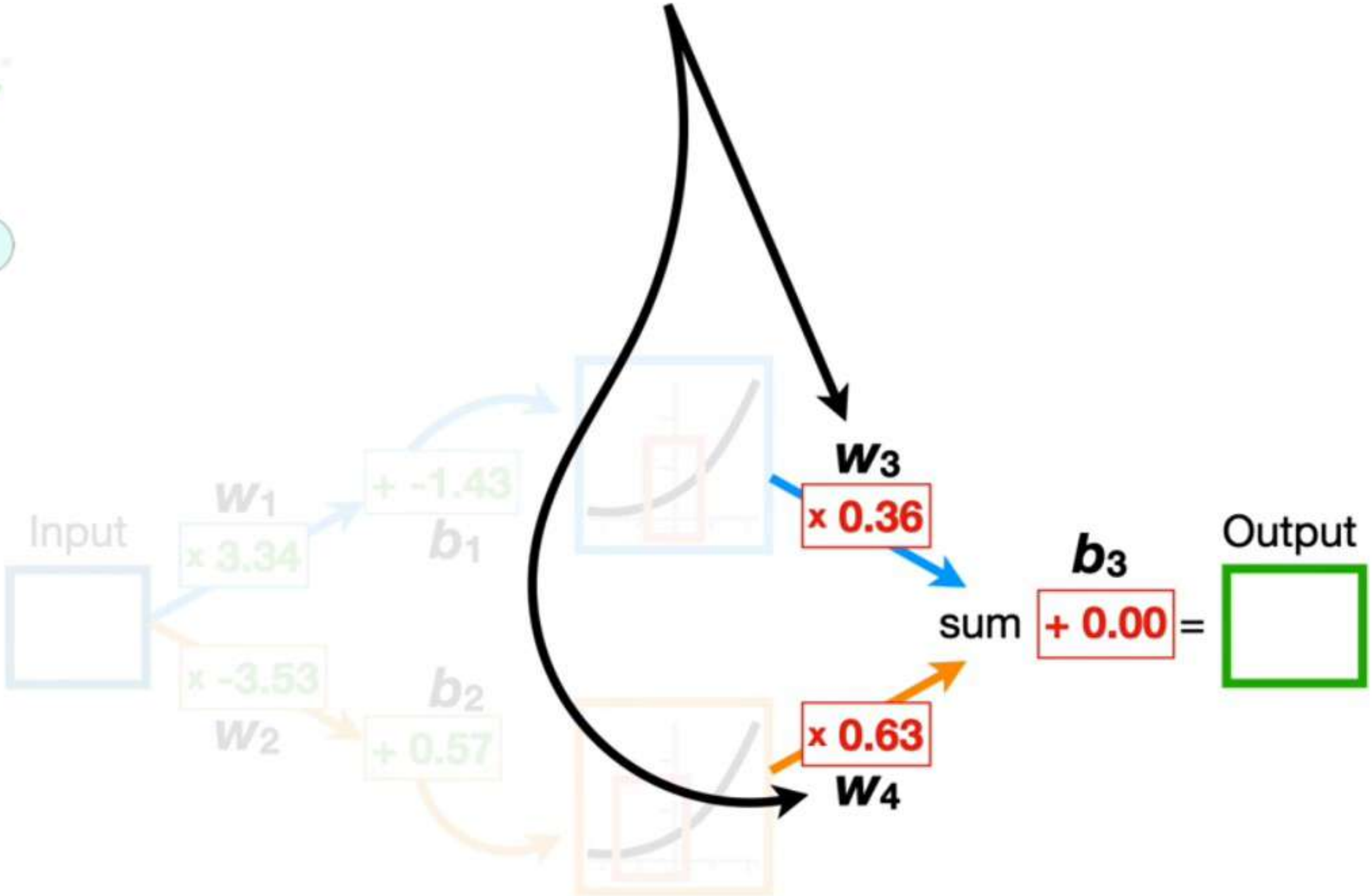
And just like before, if we change b_3 then we will change the **SSR**…



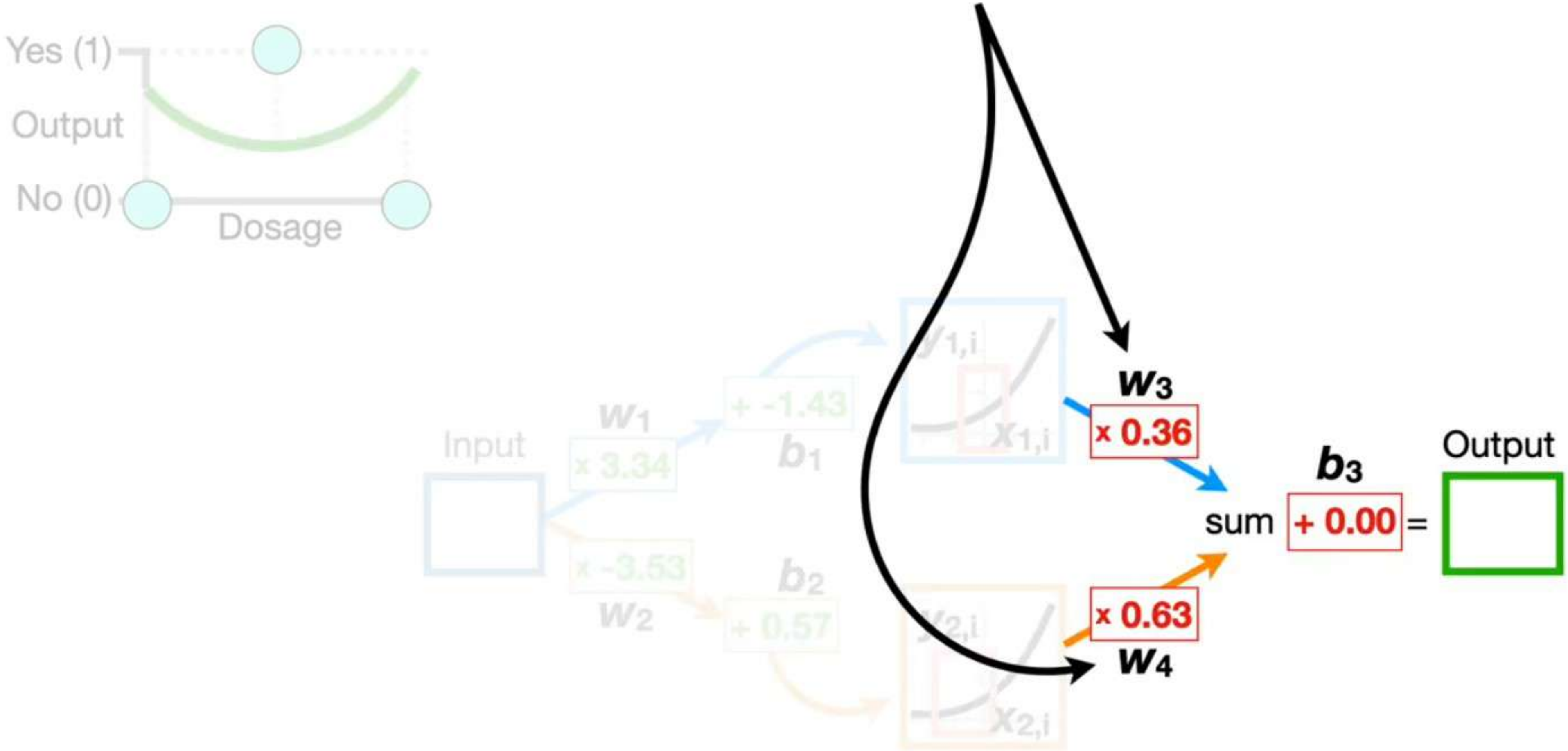
$$SSR = (0 - -0.28)^2 + (1 - -0.54)^2 + (0 - -0.22)^2 = 2.5$$

Now let's talk about how to calculate the derivatives of the **SSR** with respect to the **Weights** w_3 and w_4.

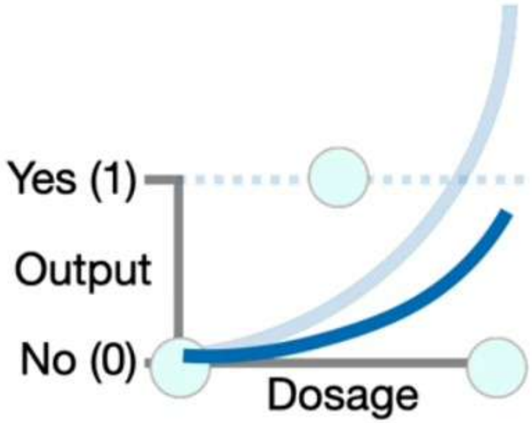




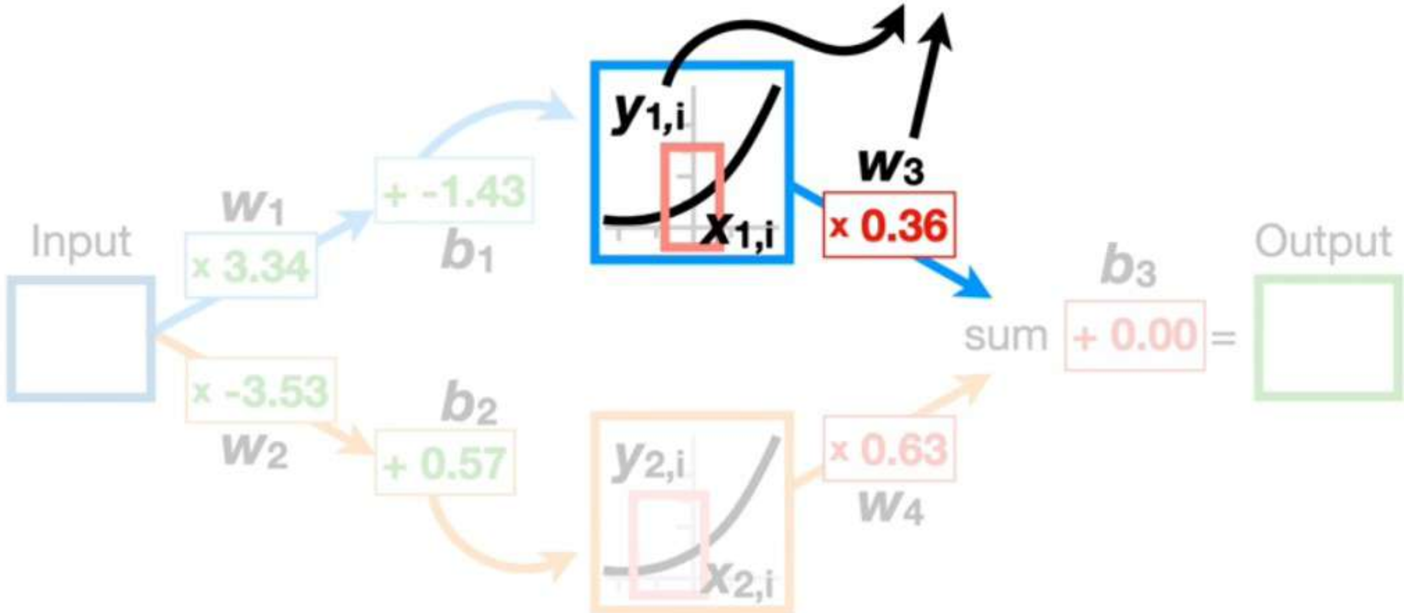
...we can talk about how to calculate the derivatives of the **SSR** with respect to the **Weights** w_3 and w_4.



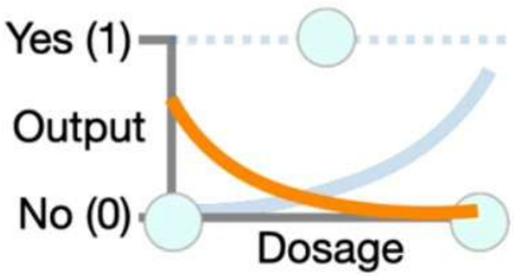
And that means we can plug $y_{1,i}$ times w_3 into the equation for the **Predicted** values.



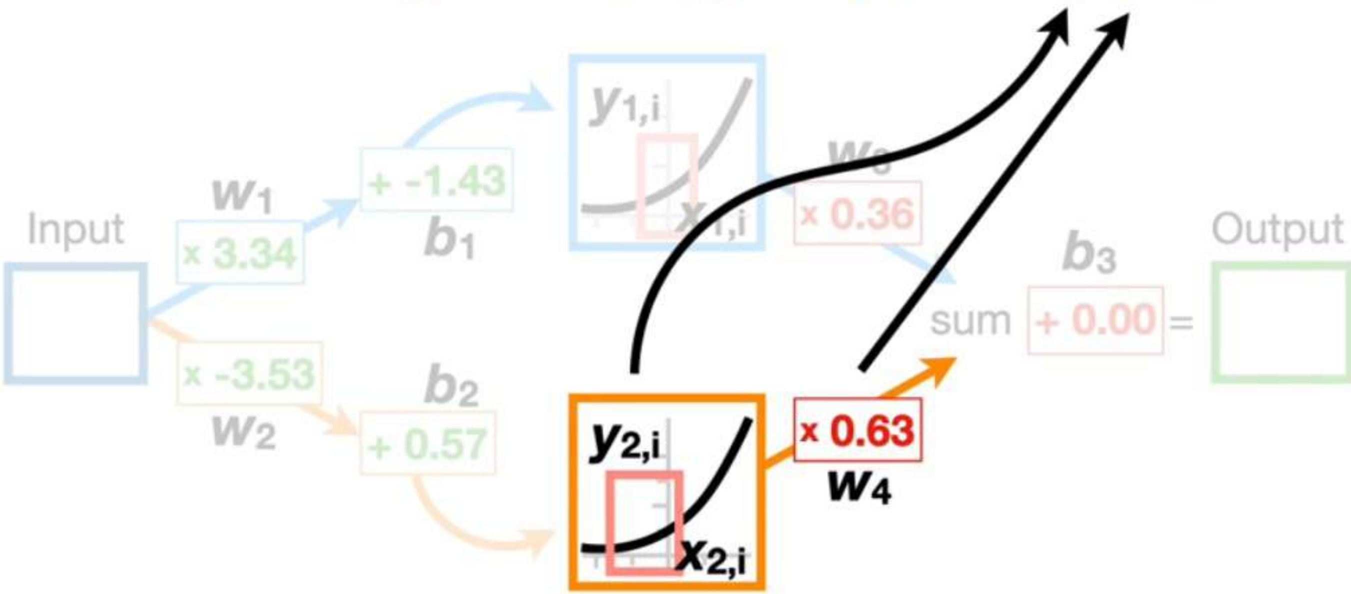
$$\text{Predicted}_i = \textbf{green squiggle}_i = y_{1,i}w_3 + \textbf{orange} + b_3$$



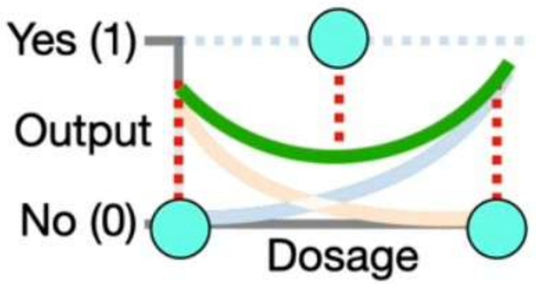
And that means we can plug $y_{2,i}$ times w_4 into the equation for the **Predicted** values.



$$\text{Predicted}_i = \textbf{green squiggle}_i = y_{1,i} w_3 + y_{2,i} w_4 + b_3$$



…then the **SSR** are linked to $\boldsymbol{w_3}$ and $\boldsymbol{w_4}$…



$$\boxed{\text{SSR}} = \sum_{i=1}^{n=3} (\text{Observed}_i - \text{Predicted}_i)^2$$

$$\text{Predicted}_i = \textbf{green squiggle}_i = y_{1,}\boxed{w_3} + y_{2,}\boxed{w_4} + \boldsymbol{b_3}$$

$$\frac{d\,SSR}{d\,w_3} = \frac{d\,SSR}{d\,Predicted} \times \frac{d\,Predicted}{d\,w_3}$$

$$\frac{d\,SSR}{d\,w_4} = \frac{d\,SSR}{d\,Predicted} \times \frac{d\,Predicted}{d\,w_4}$$

…times the derivative of the **Predicted** values with respect to $\boldsymbol{w_4}$.

$$\text{SSR} = \sum_{i=1}^{n=3}(\text{Observed}_i - \boxed{\text{Predicted}_i})^2$$

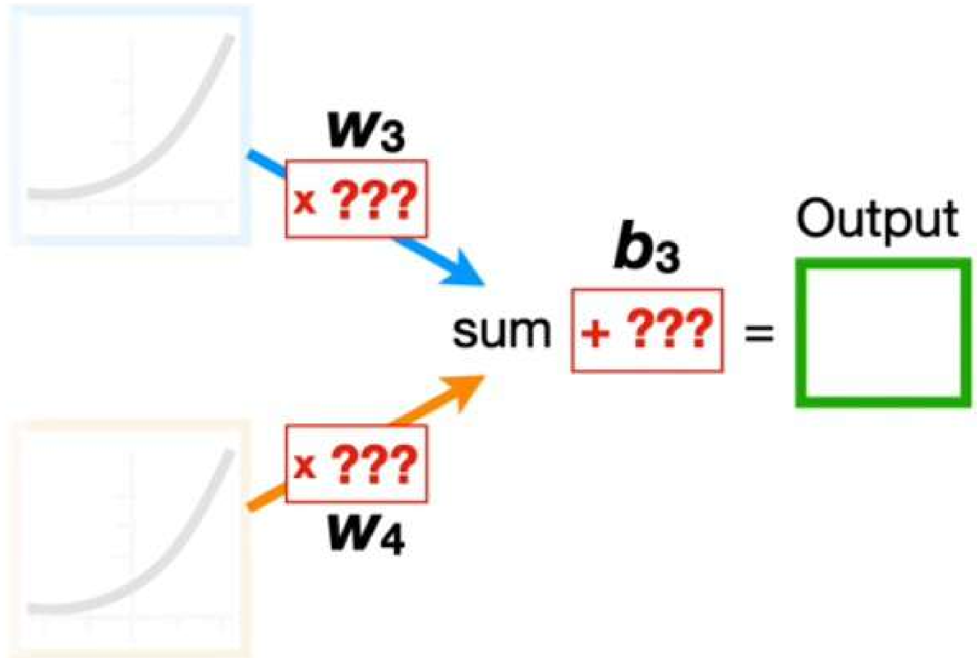
$$\boxed{\text{Predicted}_i} = \textbf{green squiggle}_i = y_{1,i}w_3 + y_{2,i}\boxed{w_4} + b_3$$

$$\frac{d\,SSR}{d\,w_3} = \sum_{i=1}^{n=3} -2 \times (\text{Observed}_i - \text{Predicted}_i) \times y_{1,i}$$

$$\frac{d\,SSR}{d\,w_4} = \sum_{i=1}^{n=3} -2 \times (\text{Observed}_i - \text{Predicted}_i) \times y_{2,i}$$

$$\frac{d\,SSR}{d\,b_3} = \sum_{i=1}^{n=3} -2 \times (\text{Observed}_i - \text{Predicted}_i) \times \mathbf{1}$$

…we can plug them into **Gradient Descent** to optimize w_3, w_4 and b_3.



$$\frac{d\,SSR}{d\,w_3} = \sum_{i=1}^{n=3} -2 \times (\text{Observed}_i - \text{Predicted}_i) \times y_{1,i}$$

$$\frac{d\,SSR}{d\,w_4} = \sum_{i=1}^{n=3} -2 \times (\text{Observed}_i - \text{Predicted}_i) \times y_{2,i}$$

$$\frac{d\,SSR}{d\,b_3} = \sum_{i=1}^{n=3} -2 \times (\text{Observed}_i - \text{Predicted}_i) \times \mathbf{1}$$

Now we repeat that process until the **Predictions** no longer improve very much, or we reach a maximum number of steps or we meet some other criteria.