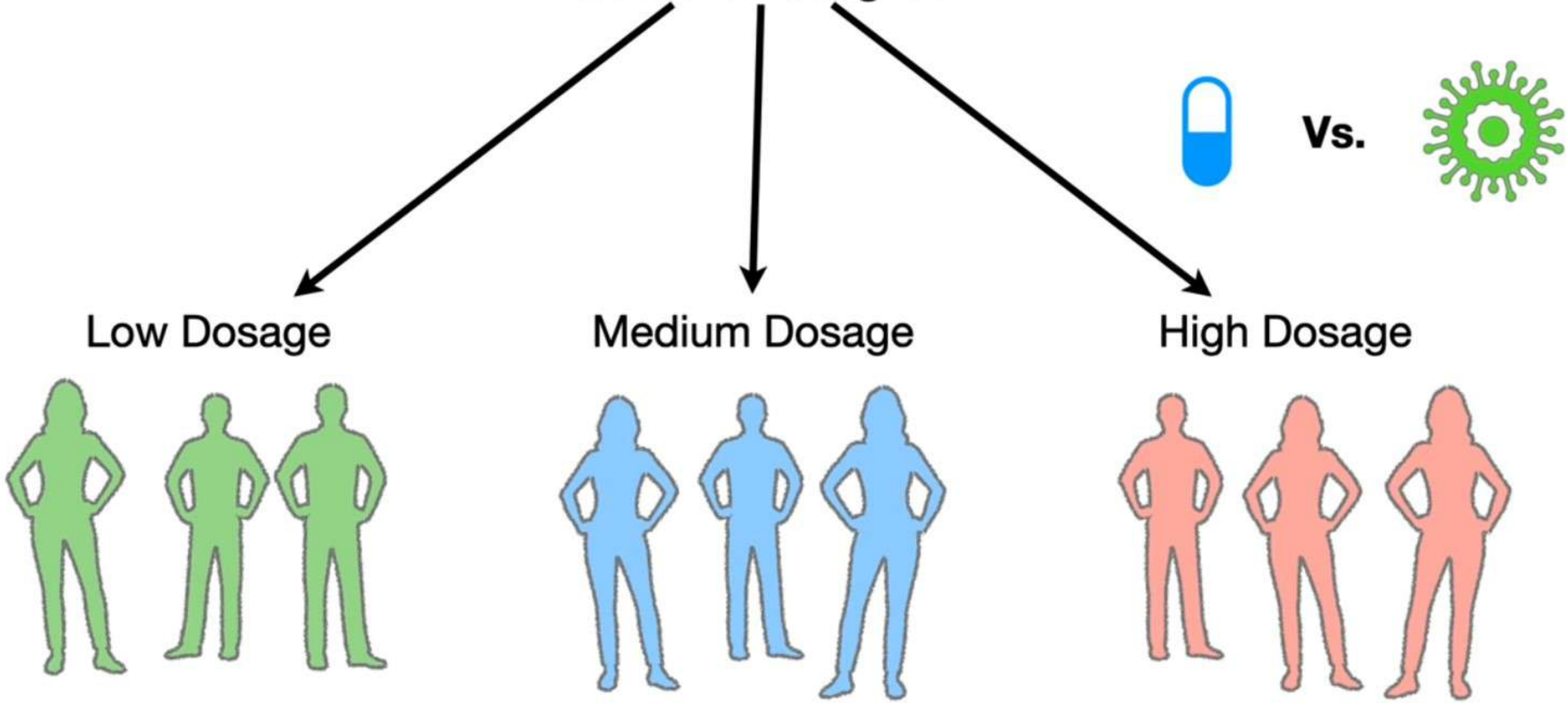
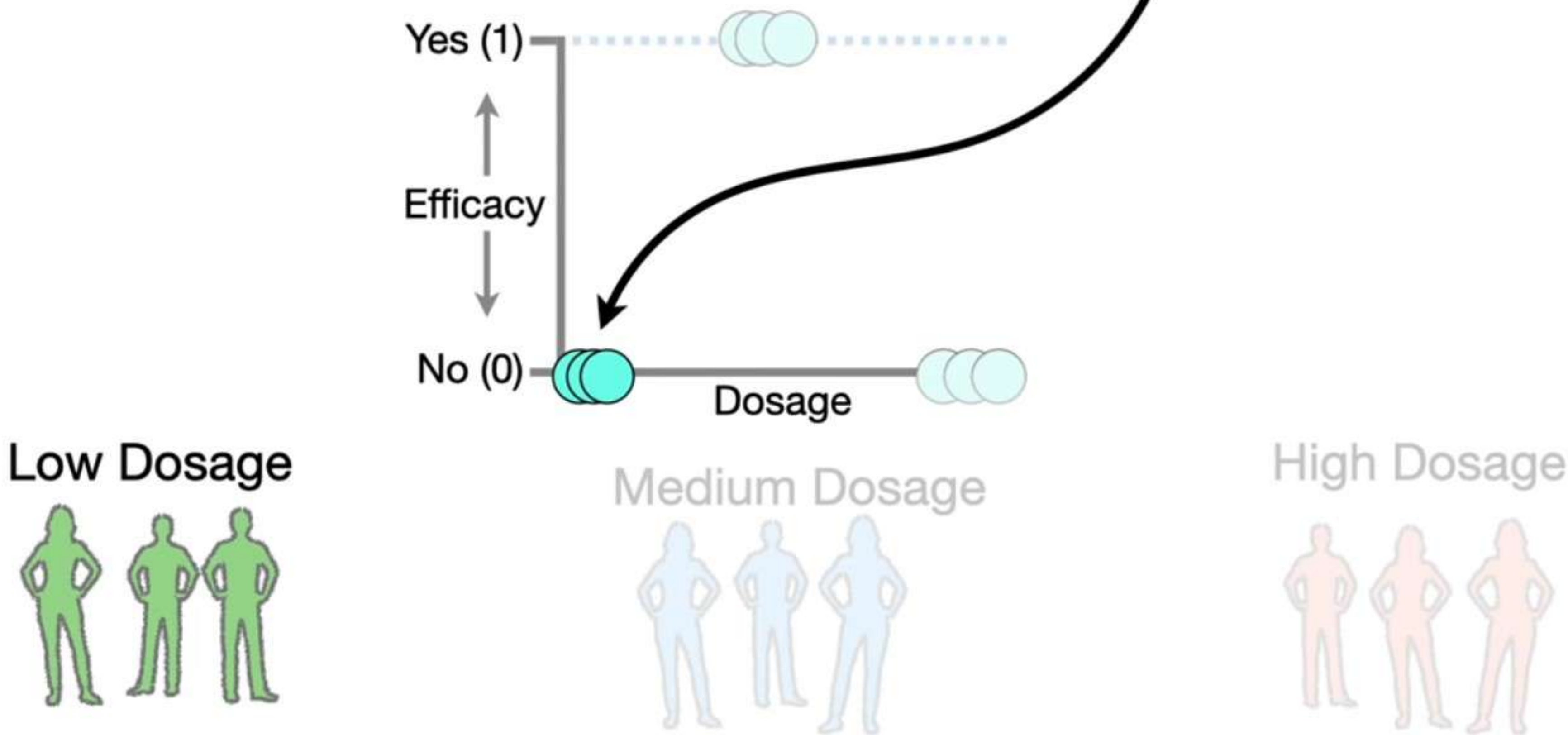
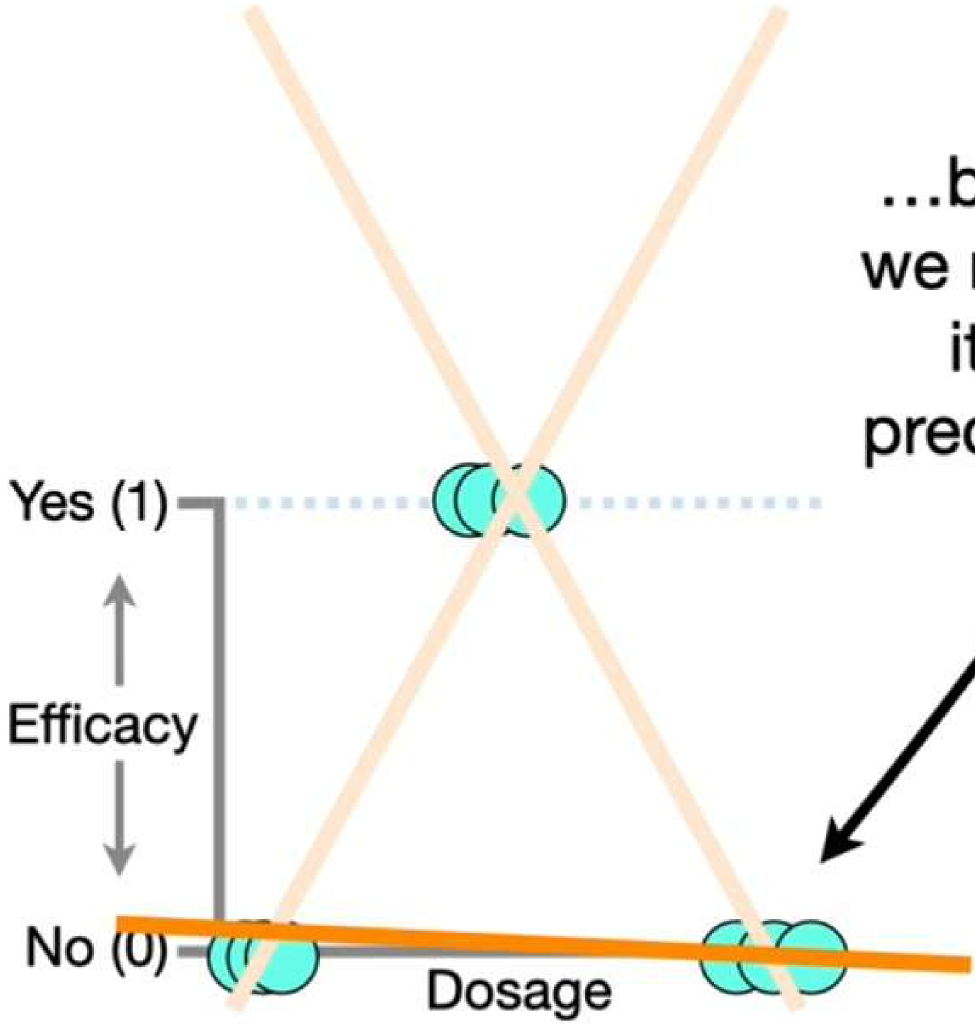
…and we gave the drug to **3** different groups of people with **3** different **Dosages.**

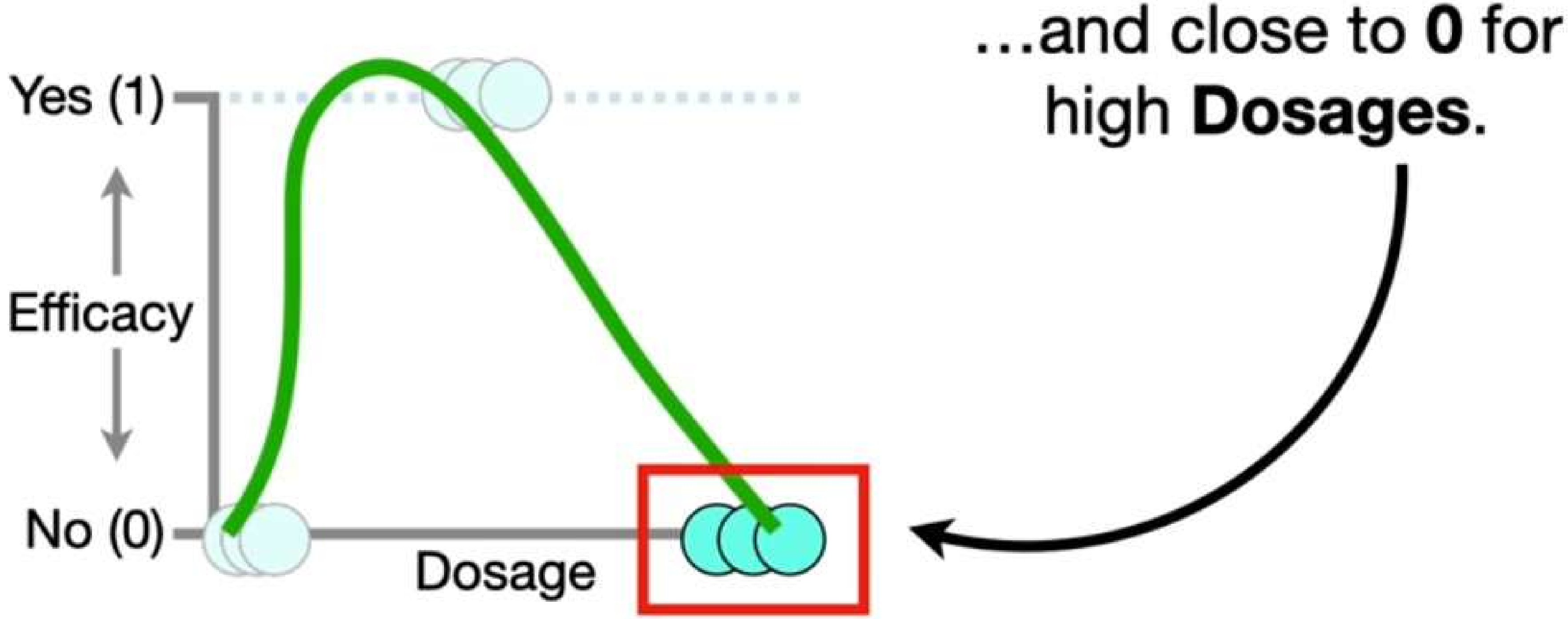


The low **Dosages** were ***not Effective***, so we set them to **0** on this graph.

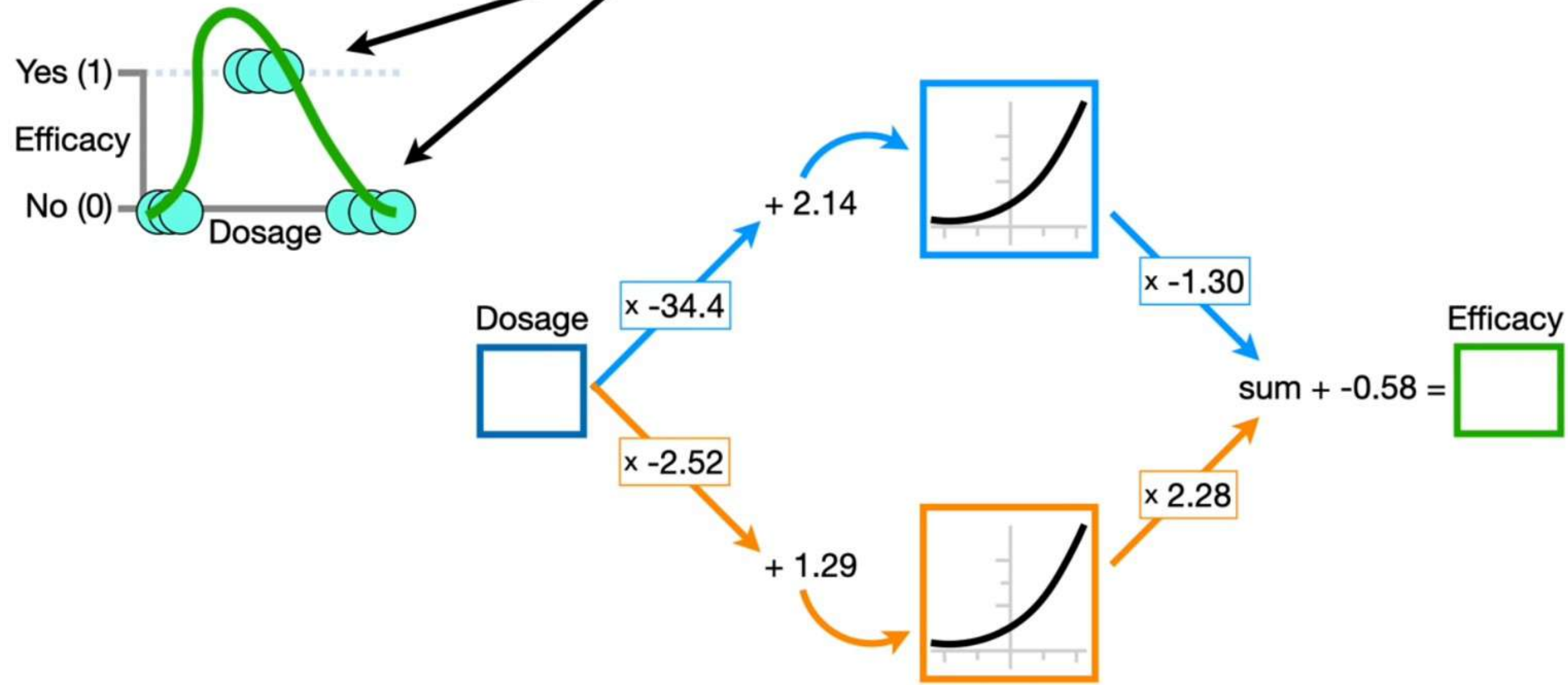




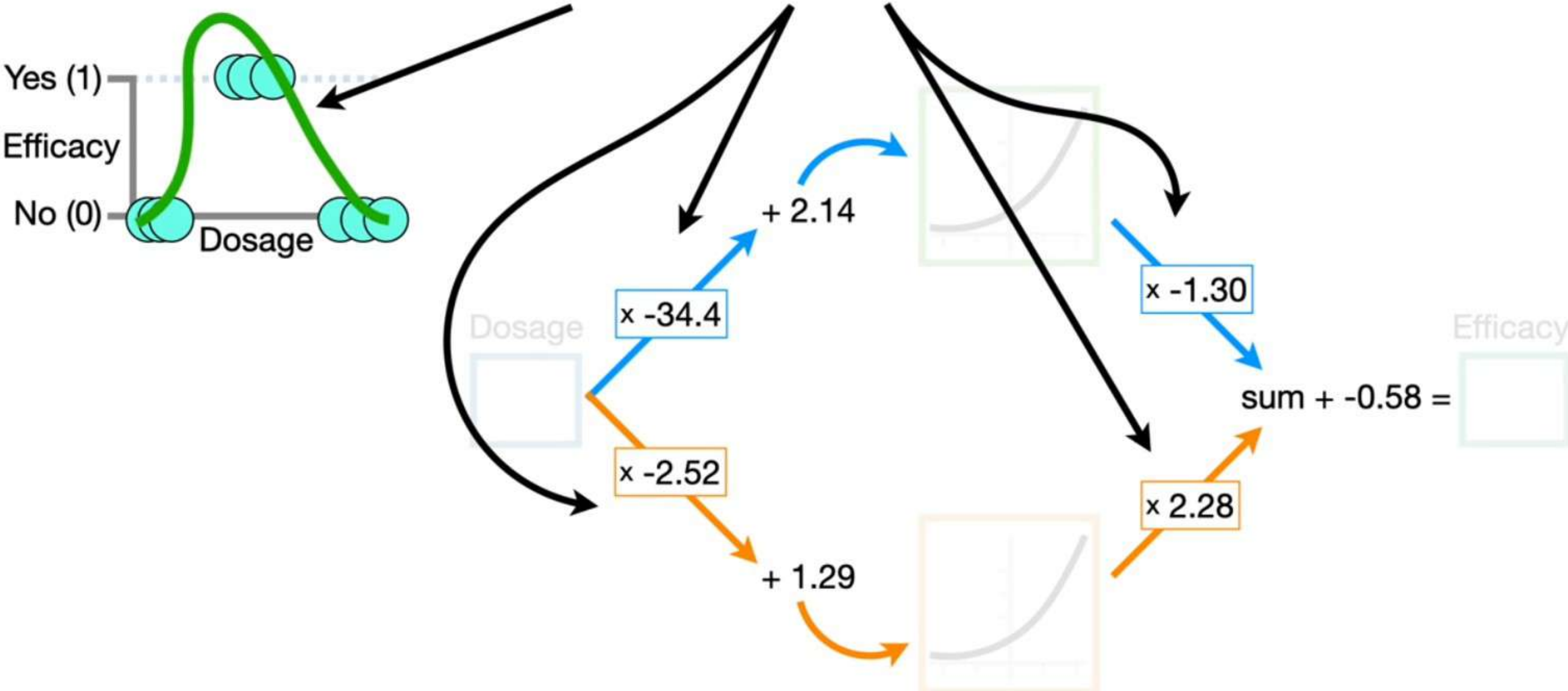

...because no matter how we rotate the **straight line**, it can only accurately predict **2** of the **3** dosages.


Yes (1)
Efficacy
No (0)
Dosage
…and close to **0** for
high **Dosages**.

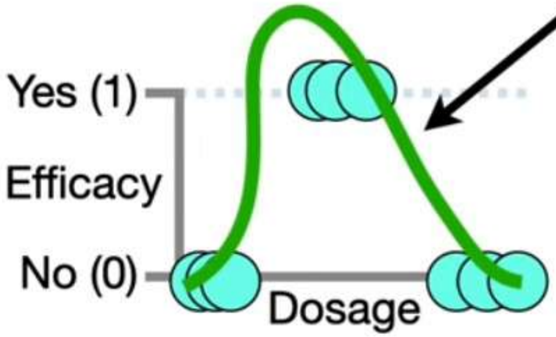
…creates this **green squiggle**.

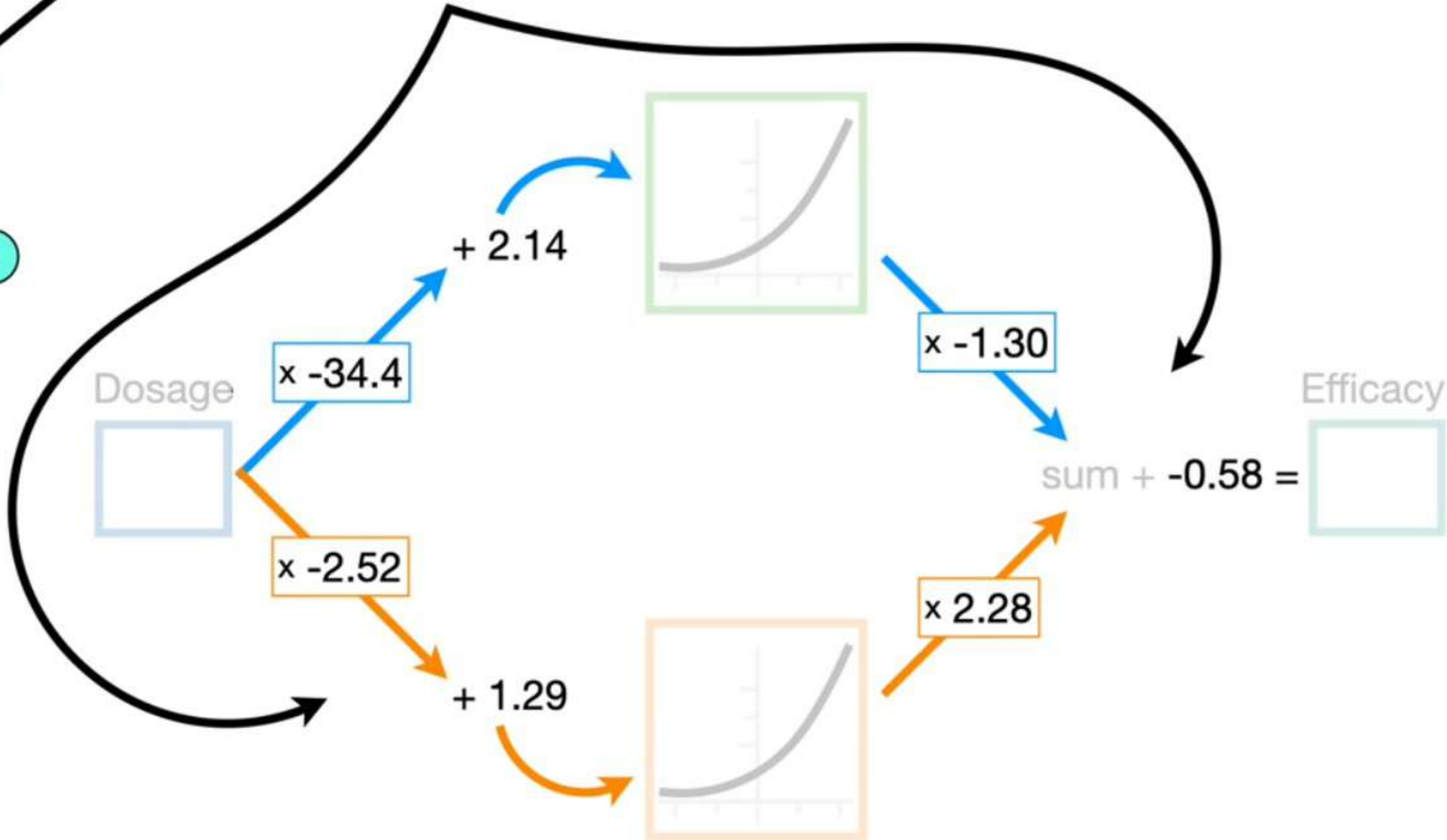


**NOTE:** The numbers along each connection represent parameter values that were estimated when this **Neural Network** was fit to the data.

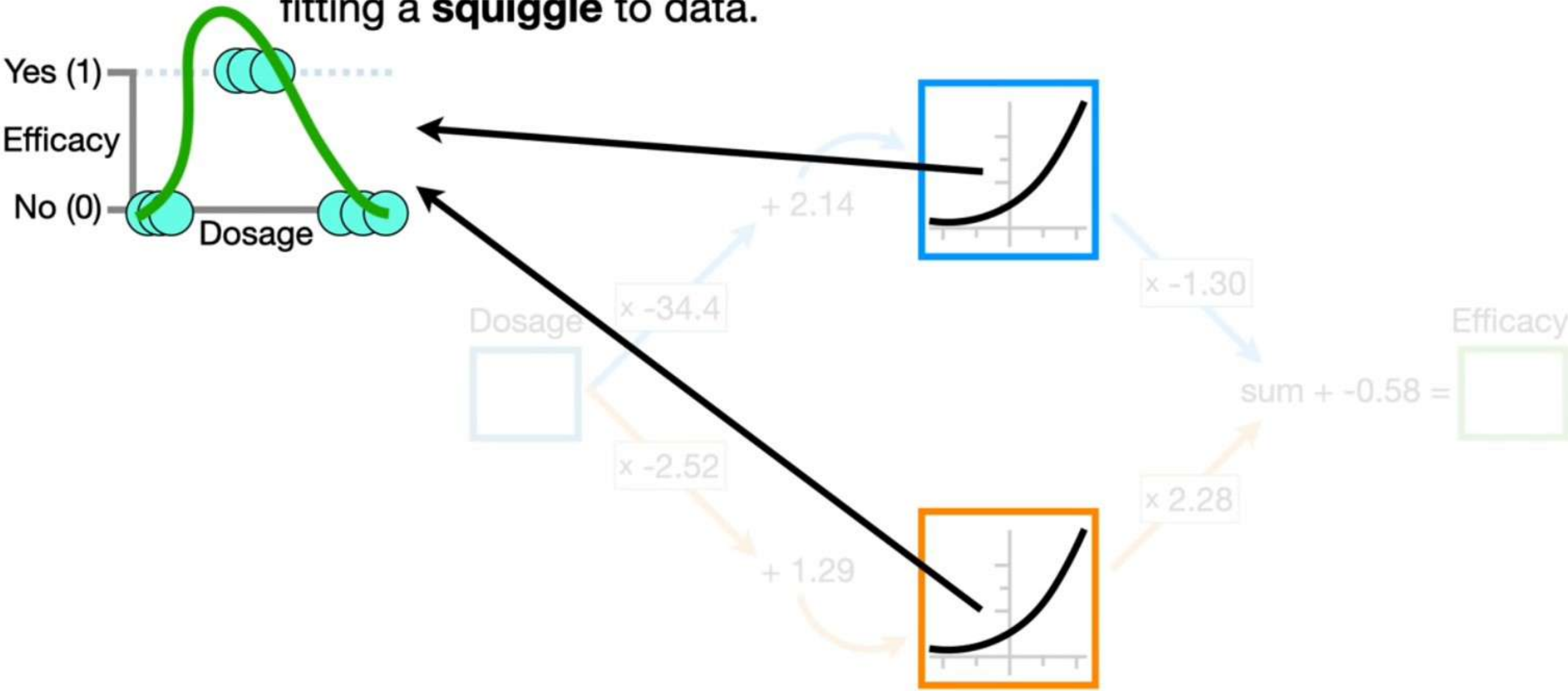


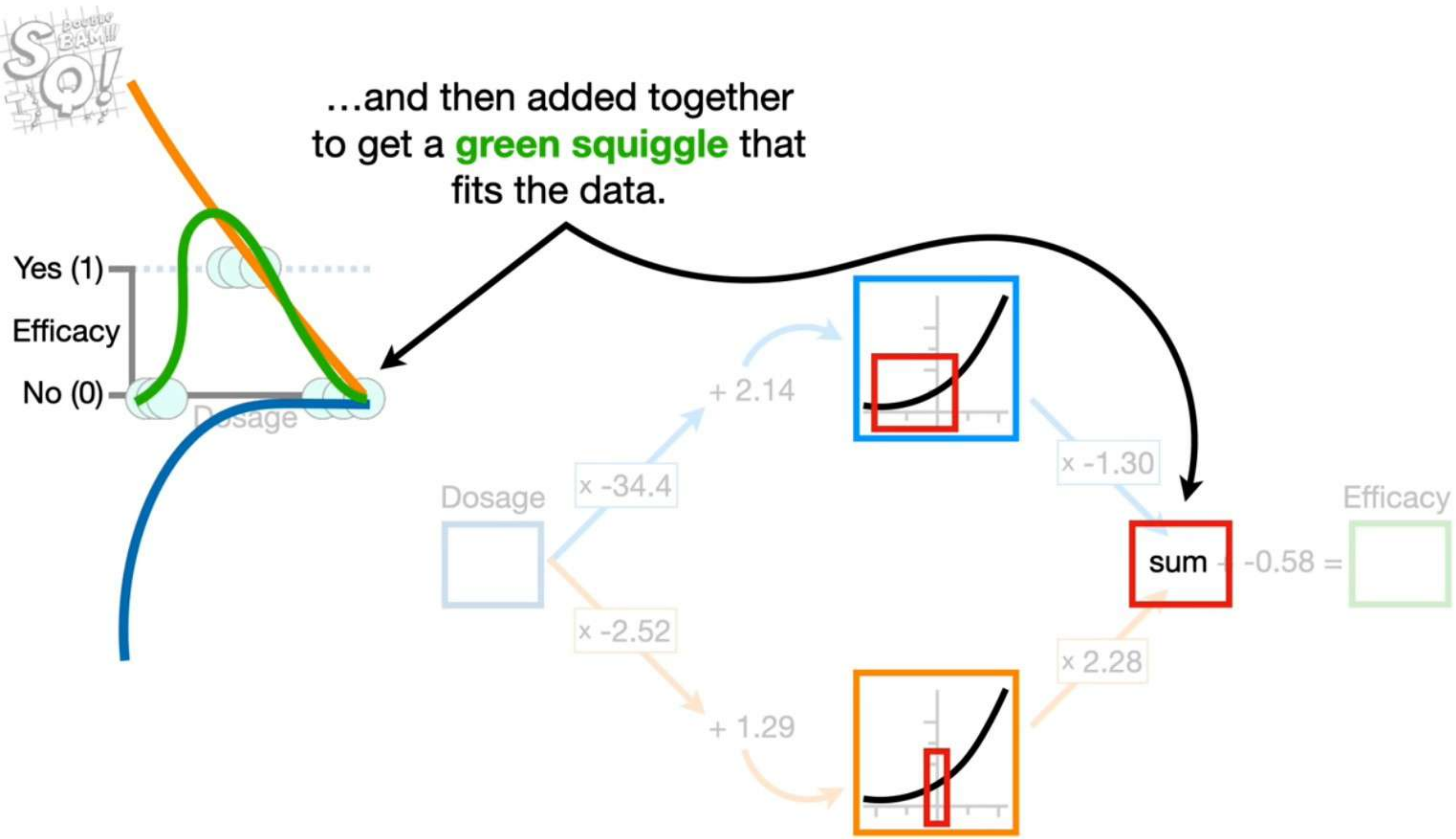
...that are estimated when we fit the **Neural Network** to a dataset using a method called **Backpropagation**.



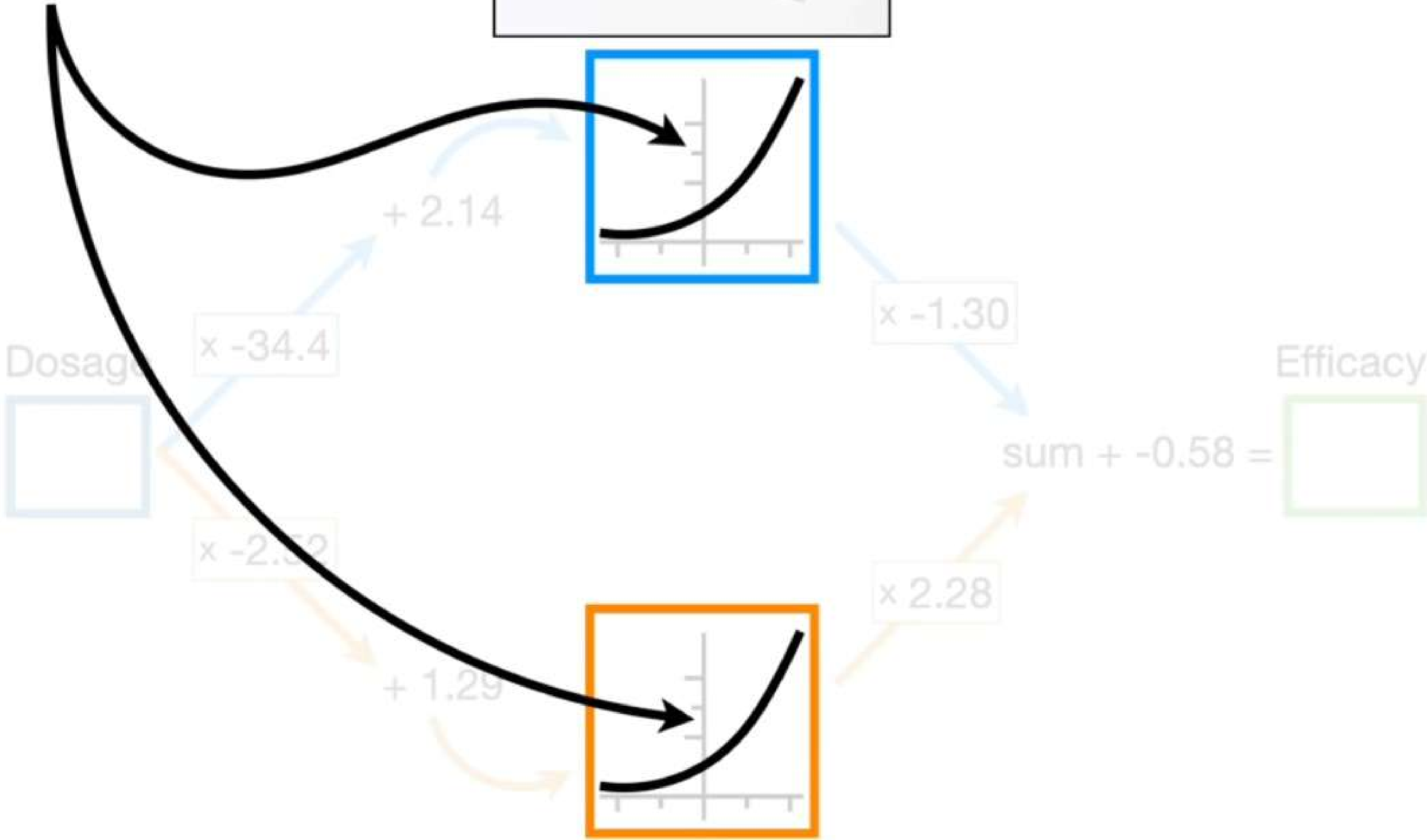
These **bent** or **curved lines** are the building blocks for fitting a **squiggle** to data.



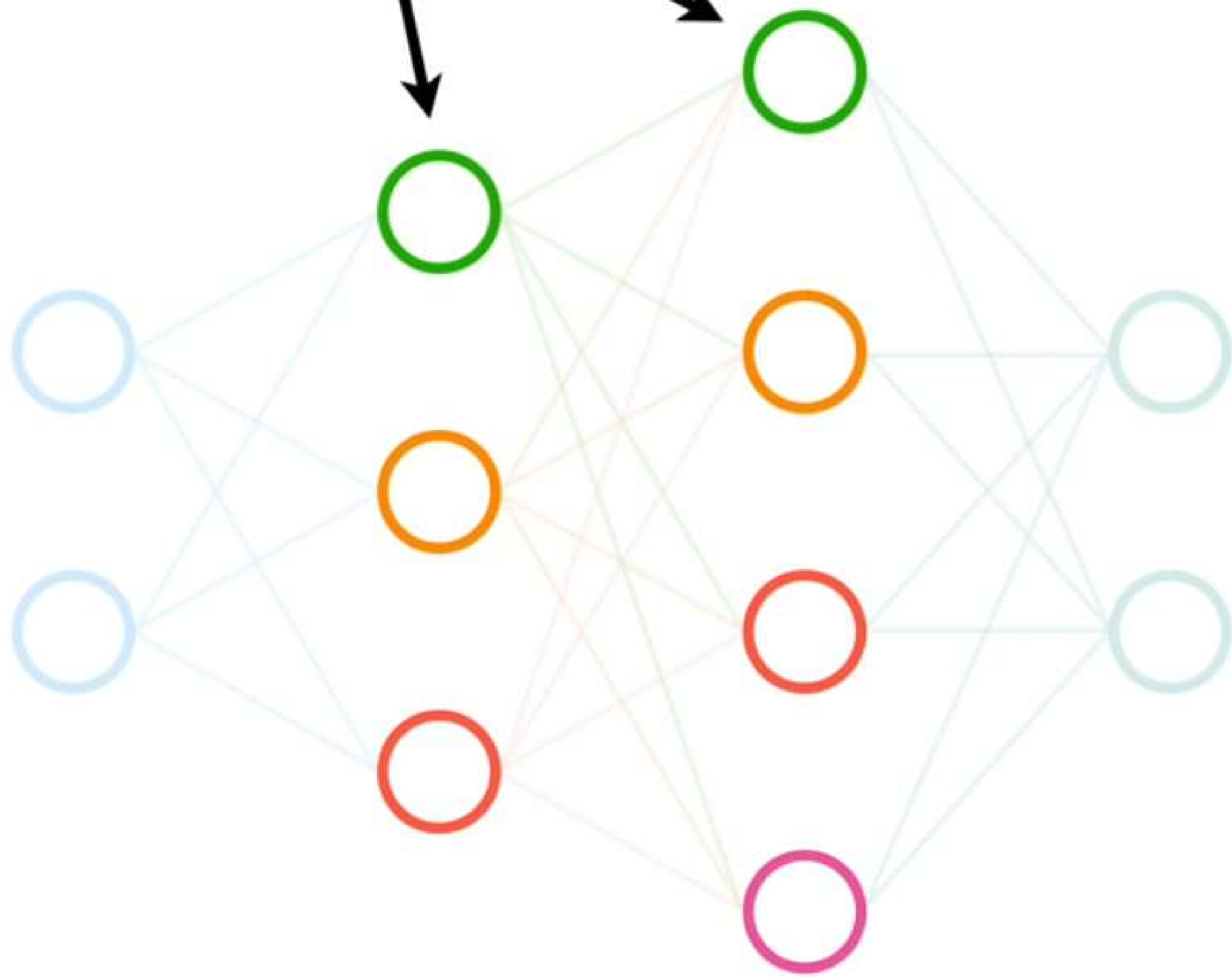


…and then added together to get a **green squiggle** that fits the data.
Yes (1)
Efficacy
No (0)
Dosage
+ 2.14
x -34.4
Dosage
x -1.30
sum
-0.58 =
Efficacy
x -2.52
+ 1.29
x 2.28

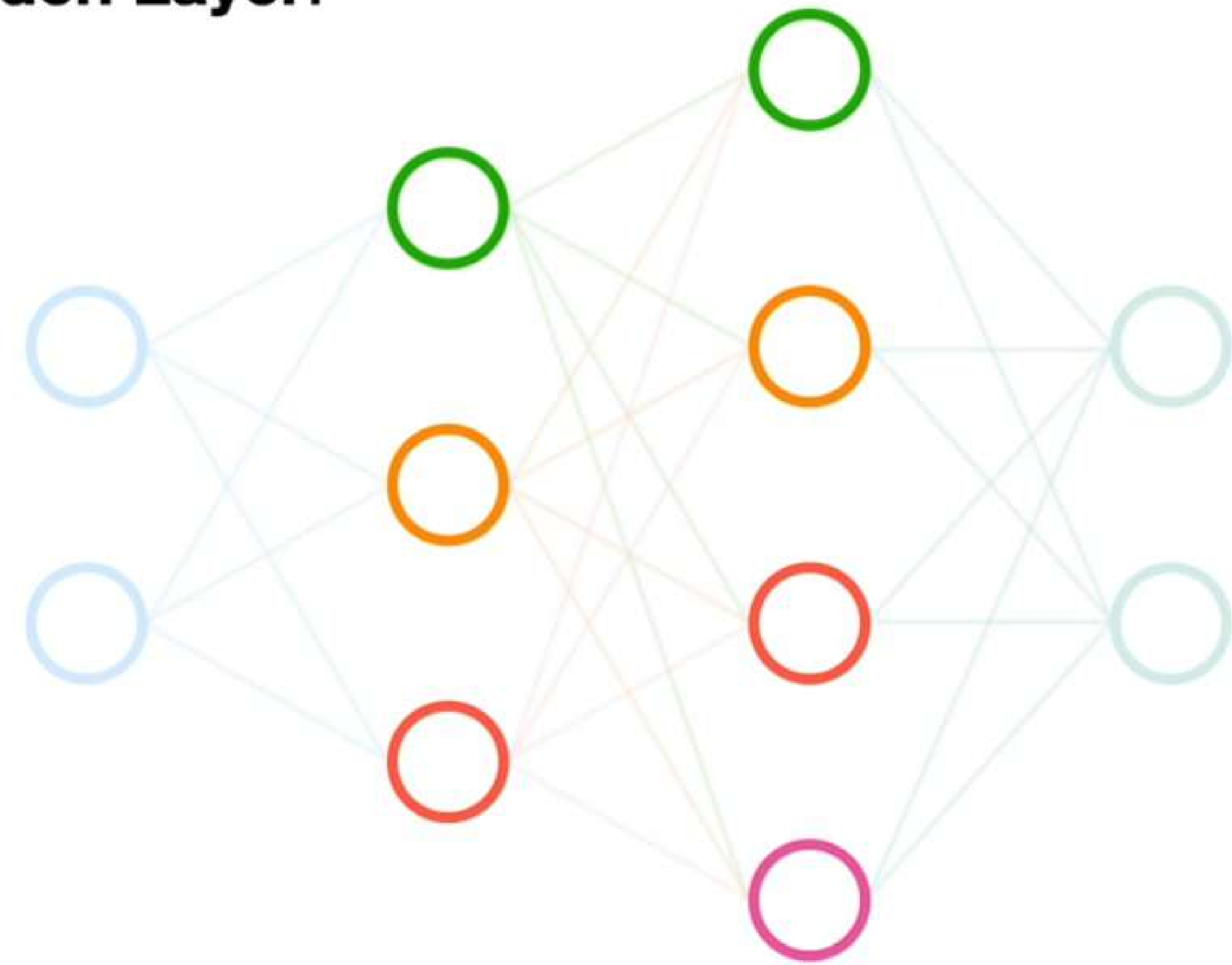
This specific **curved line** is called **softplus**, which sounds like a brand of toilet paper.

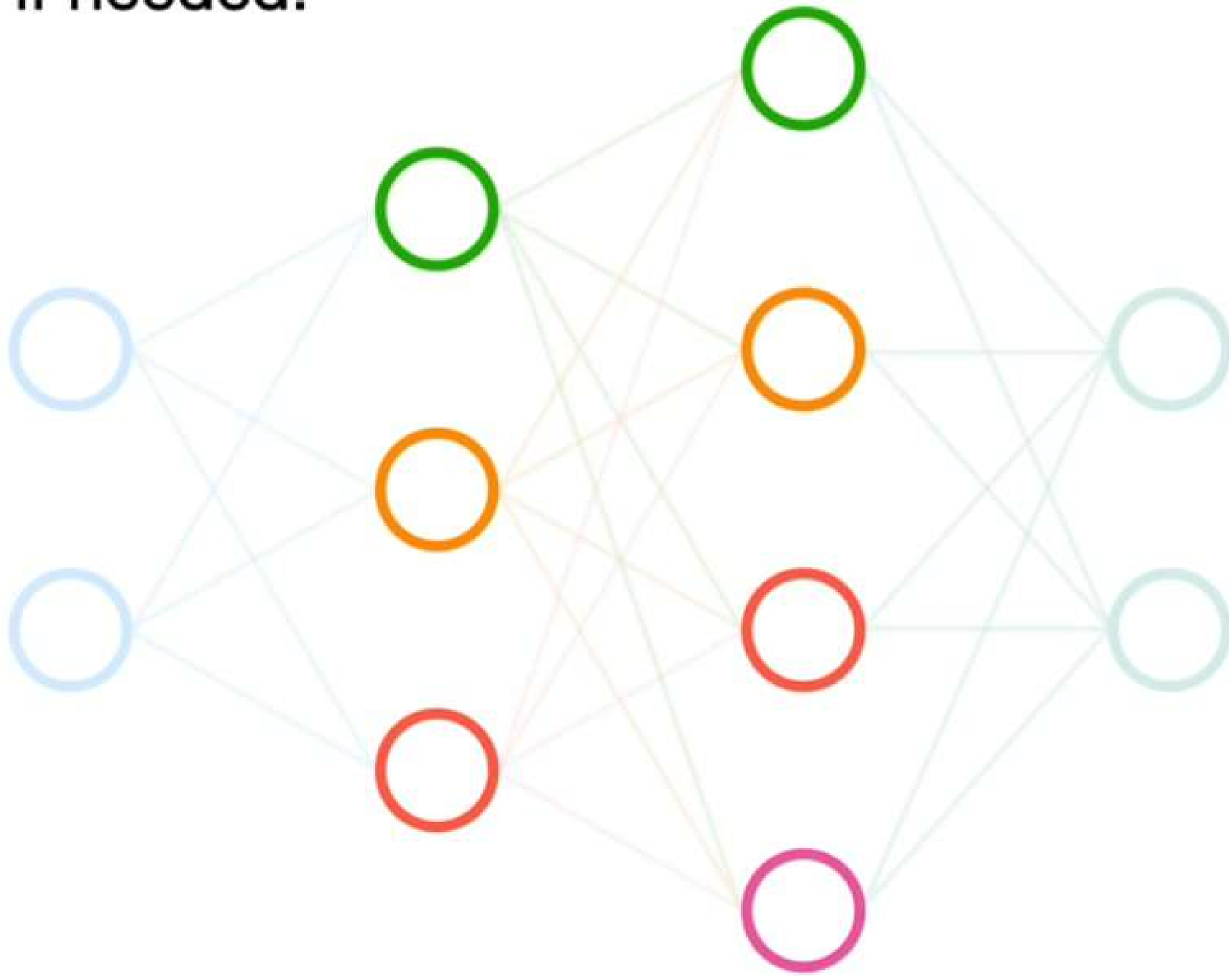
The layers of **Nodes** between the **Input** and **Output Nodes** are called **Hidden Layers.**

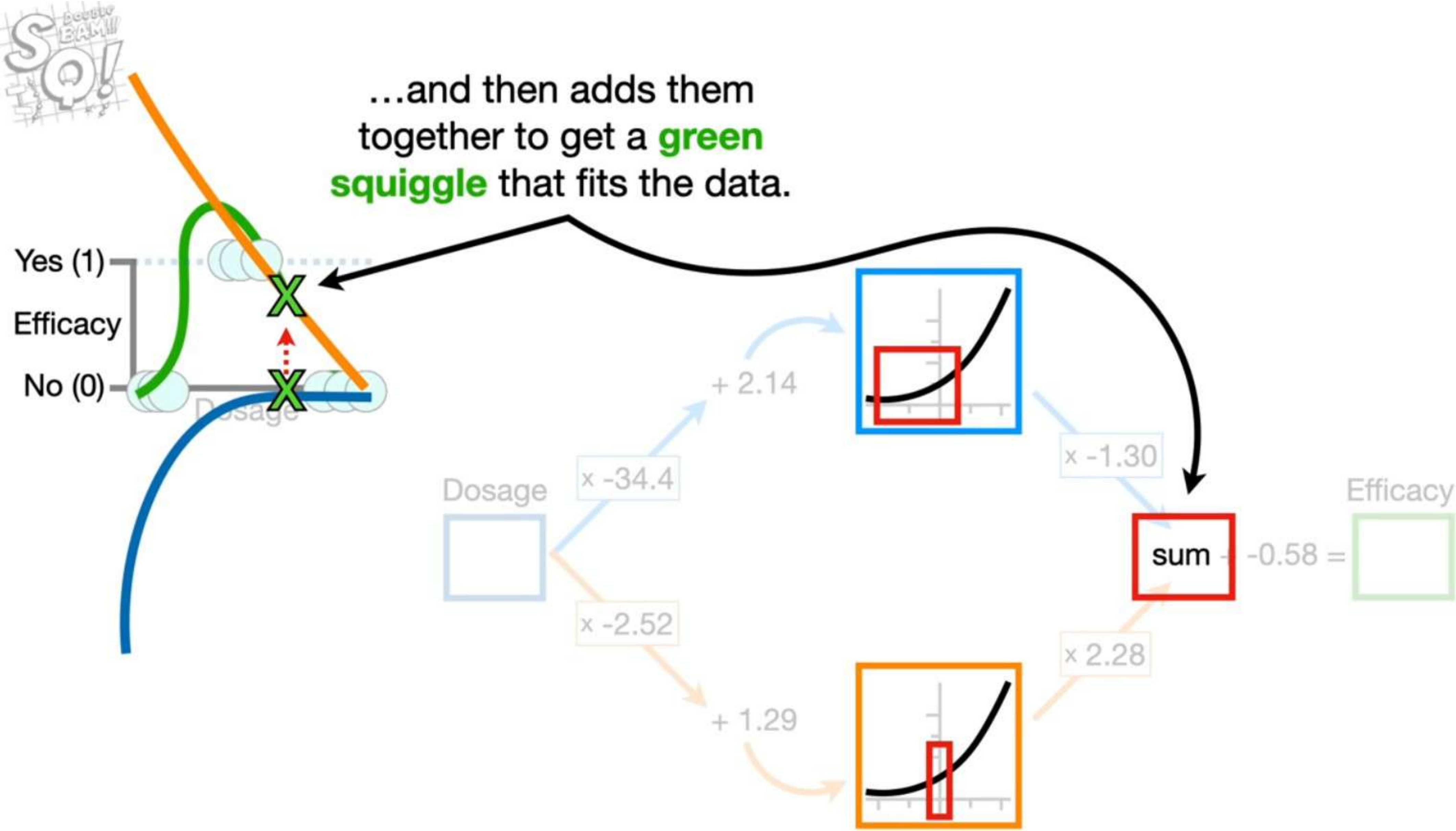


When you build a **Neural Network**, one of the first things you do is decide how many **Hidden Layers** you want, and how many **Nodes** go into each **Hidden Layer**.



…you essentially make a guess and see how well the **Neural Network** performs, adding more layers and nodes if needed.

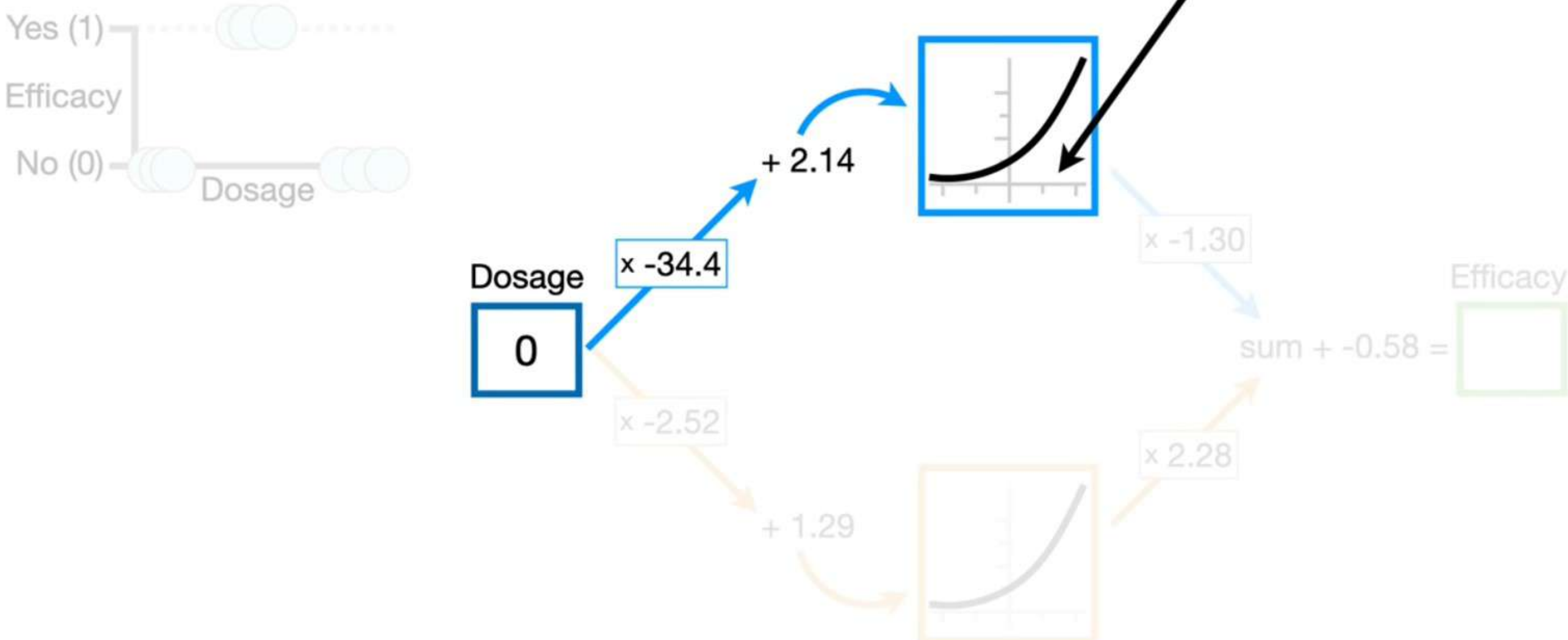


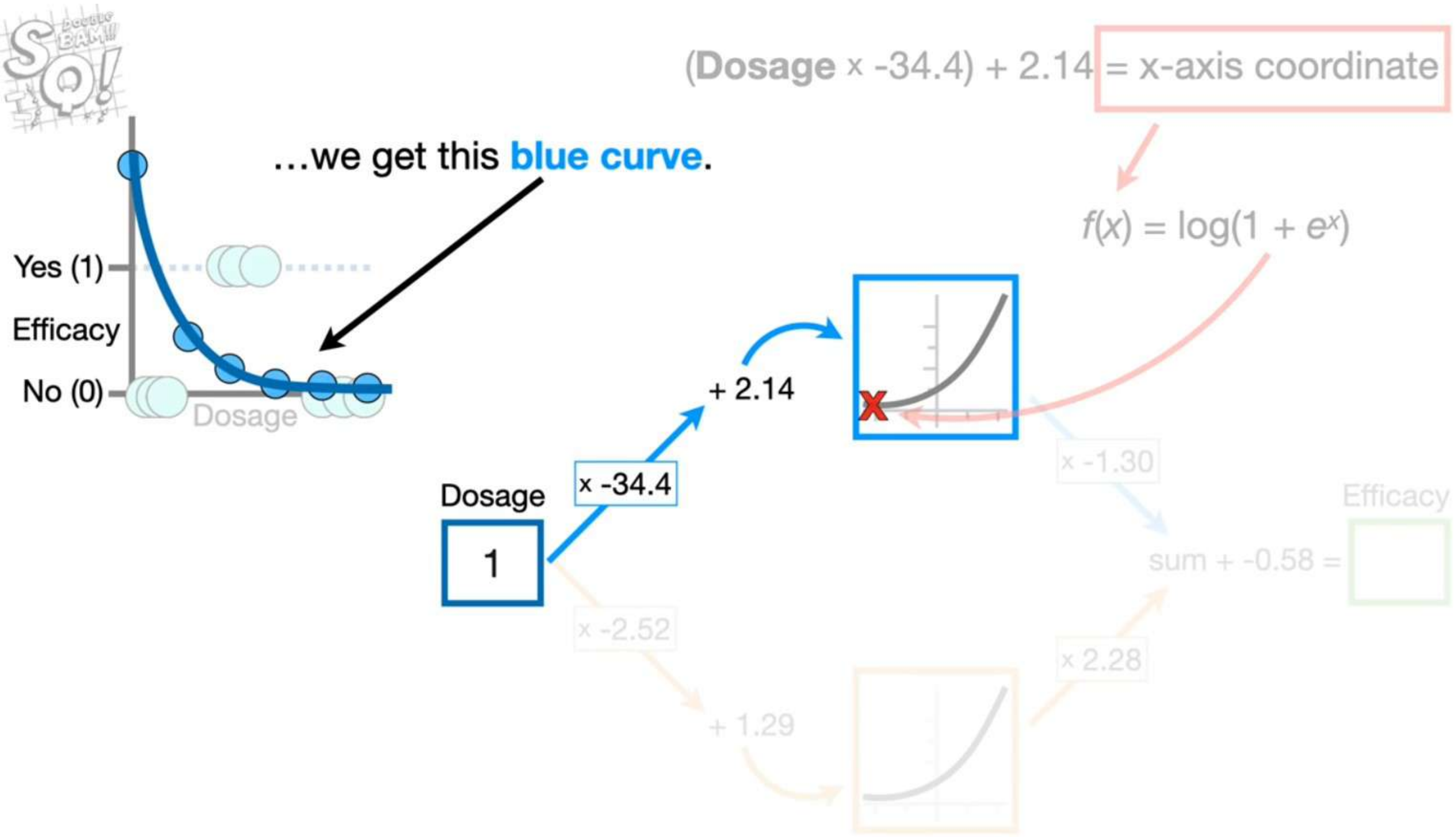


...and then adds them together to get a green squiggle that fits the data.
Yes (1)
Efficacy
No (0)
Dosage
+ 2.14
x -34.4
x -1.30
Dosage
Efficacy
sum
-0.58 =
x -2.52
x 2.28
+ 1.29

(**Dosage** x -34.4) + 2.14 $\boxed{\text{= x-axis coordinate}}$

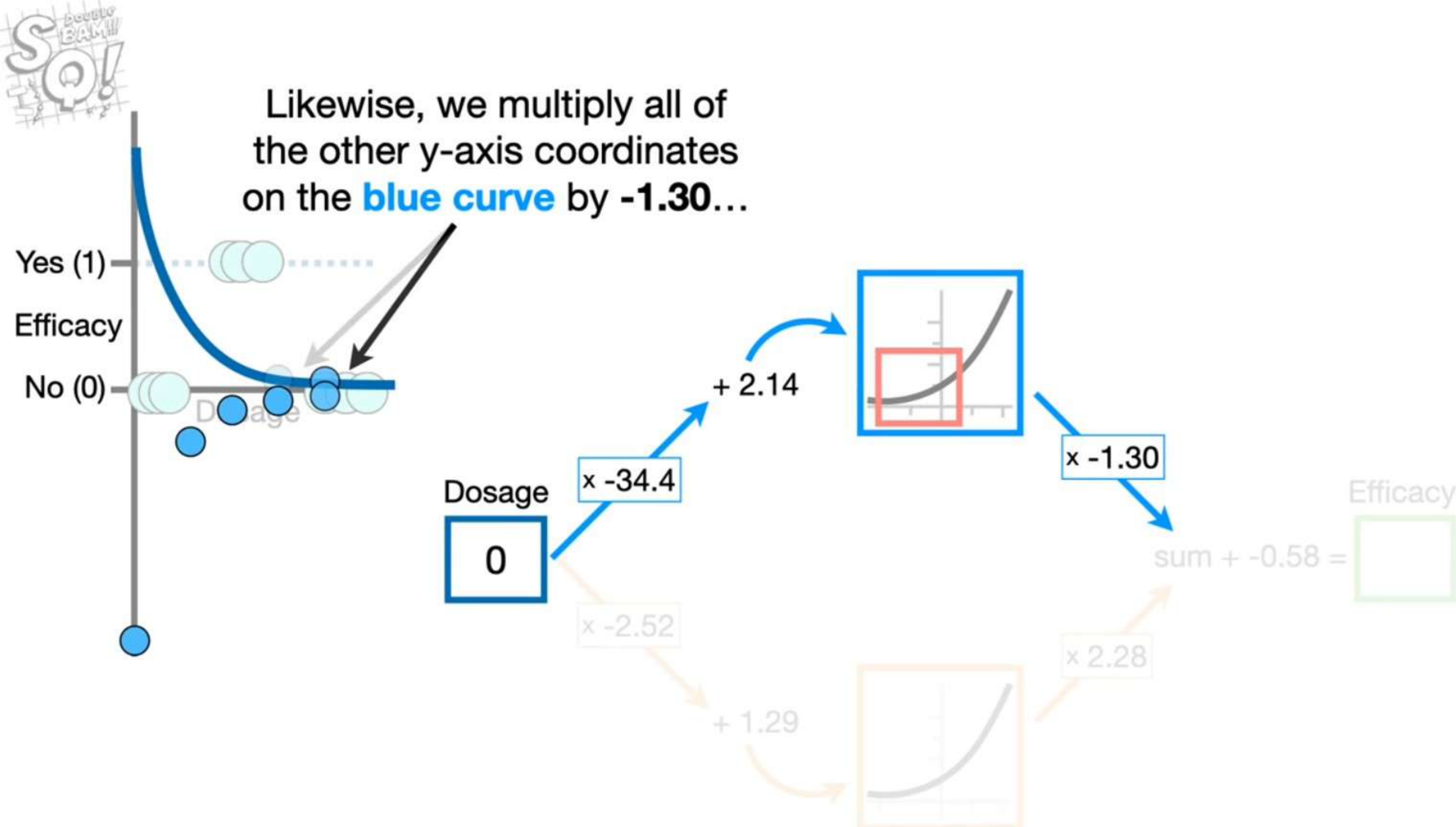
…and the result is an x-axis coordinate for the **Activation Function**.

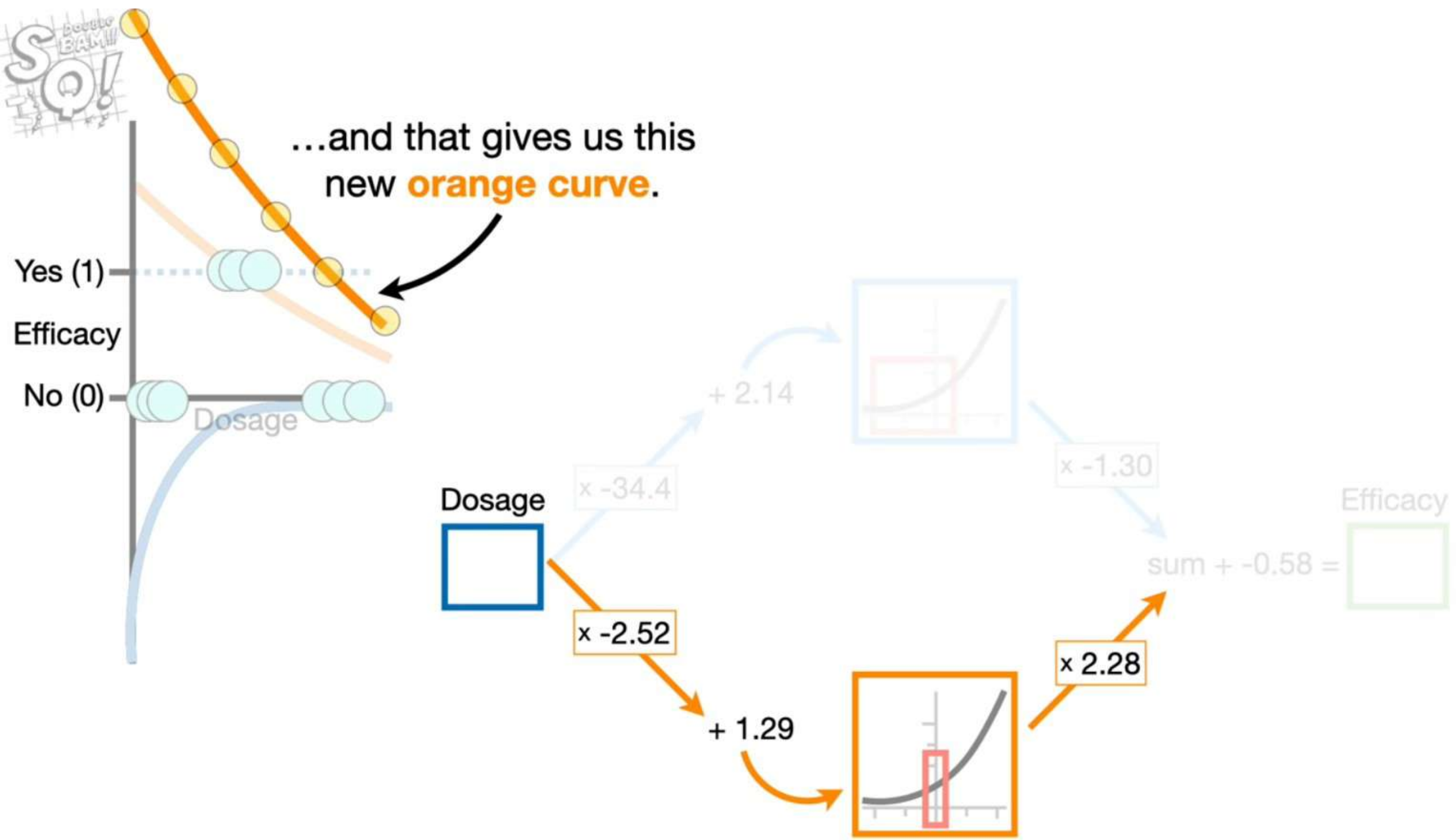




(Dosage x -34.4) + 2.14 = x-axis coordinate
$f(x) = \log(1 + e^x)$
...we get this blue curve.
Yes (1)
Efficacy
No (0)
Dosage
+ 2.14
x -34.4
Dosage
1
x -1.30
Efficacy
sum + -0.58 =
x -2.52
x 2.28
+ 1.29

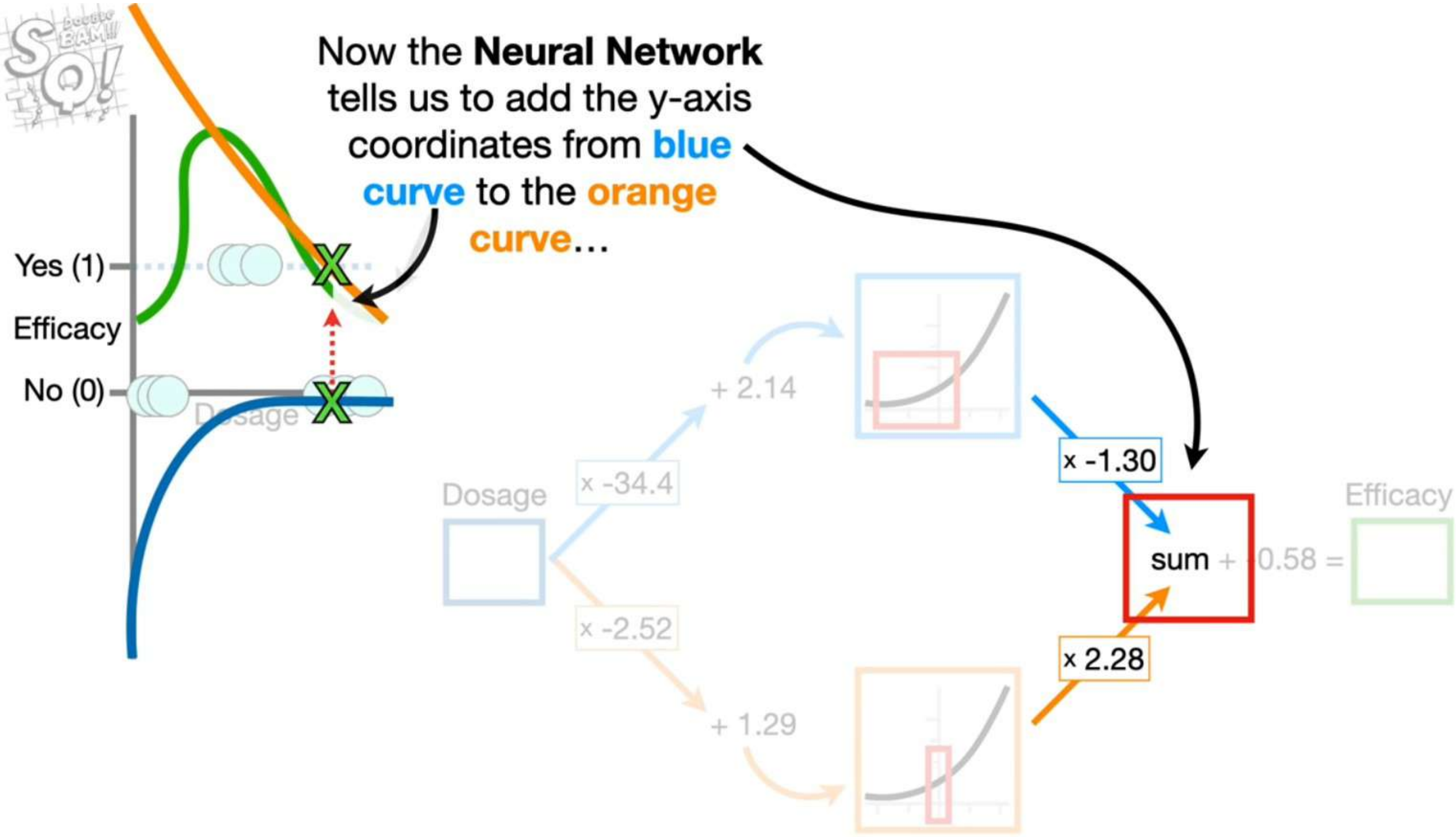
Likewise, we multiply all of the other y-axis coordinates on the **blue curve** by **-1.30**…

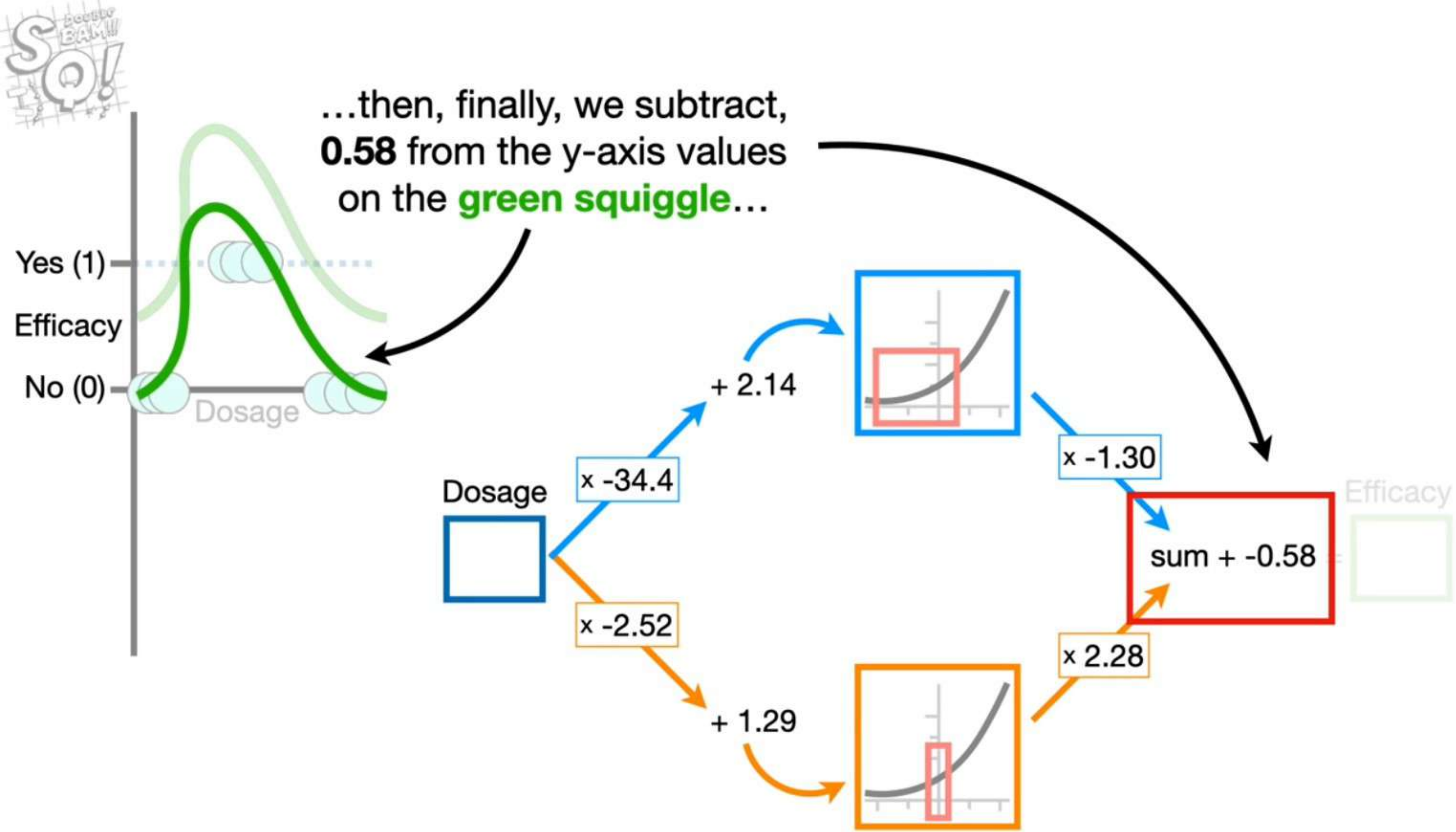




Double BAM!!!
...and that gives us this new orange curve.
Yes (1)
Efficacy
No (0)
Dosage
Dosage
x -34.4
+ 2.14
x -1.30
Efficacy
sum + -0.58 =
x -2.52
+ 1.29
x 2.28

Now the **Neural Network** tells us to add the y-axis coordinates from **blue curve** to the **orange curve**…




...then, finally, we subtract,
**0.58** from the y-axis values
on the **green squiggle**...
Yes (1)
Efficacy
No (0)
Dosage
Dosage
x -34.4
+ 2.14
x -1.30
x -2.52
+ 1.29
x 2.28
sum + -0.58
Efficacy