I would approach this by writing some qualifications:

1. 0 events trigger a score of 0.
2. Non edge event count observations are where the score – 100-threshold would live.
3. Any score after the threshold will be 100.

If so, here’s a simple example:

# Stage Data:

userid <- c("a1","a2","a3","a4","a11","a12","a13","a14","u2","wtf42","ub40","foo","bar","baz","blue","bop","bob","boop","beep","mee","r")

events <- c(0,0,0,0,0,0,0,0,0,0,0,0,1,2,3,2,3,6,122,13,1)

df1 <- data.frame(userid,events)

Optional: Normalize events to be in [1,2]. This might be helpful for logarithmic properties. This will allow you to control sensitivity, but it must be done right as we are dealing with exponents and logarithms. I am not using normalization in the sample:

normevents <- (events-mean(events))/((max(events)-min(events))\*2)+1.5

Set the threshold:

MaxScoreThreshold=0.25

Get the non edge quintiles of the events distribution:

qts <- quantile(events[events>min(events) & events<max(events)], c(seq(from=0, to=100,by=5)/100))

Find the Events quantity that give a score of 100 using the set threshold.

MaxScoreEvents <- quantile(qts,MaxScoreThreshold)

Find the exponent of your exponential function, given that:

Score = events ^ exponent, for events being a Natural number (integer >0, we took care of it by omitting the edges) and exponent > 1.

exponent <- log(100)/log(MaxScoreEvents)

Generate the scores:

df1$Score <- apply(as.matrix(events^exponent),1,FUN = function(x) {

if (x > 100) {

result <- 100

}

else if (x < 0) {

result <- 0

}

else {

result <- x

}

return(ceiling(result))

})

df1

userid events Score

1 a1 0 0

2 a2 0 0

3 a3 0 0

4 a4 0 0

5 a11 0 0

6 a12 0 0

7 a13 0 0

8 a14 0 0

9 u2 0 0

10 wtf42 0 0

11 ub40 0 0

12 foo 0 0

13 bar 1 1

14 baz 2 100

15 blue 3 100

16 bop 2 100

17 bob 3 100

18 boop 6 100

19 beep 122 100

20 mee 13 100

21 r 1 1

I would rely more on the data to define the parameters, threshold in this case. If you have prior data as to what users really did whatever it is your score assess you can perform supervised learning , wherever the ratio is over 50%, or something like this. Or If the graph of events to probability of ‘success’ looks like the cumulative probability function of a normal distribution, I’d set threshold @ wherever it hits 45 degrees.

You could also use logistic regression if you have prior data but instead of a Logit function ingesting the output of regression, use the number as your score. You can normalize it to be witin 0-100.

It’s not always easy to write a Data Science question. I made many assumptions as to what you are looking for, hope this is the general direction.