We know how to place linear functions with linear regression using

**y = xm +b (1 dimension)**

**or**

**y = x1m1 + x2m2 + x3m3 + … xnmn + b (n dimensions)**

Where x is our input and m represents the parameters or ***Ɵ***s

But what about when the data looks like this:

A straight line wouldn’t really fit would it?

Or this:

Step 1 is always, ALWAYS look at the data.

Use a previous function that you know looks similar to the data, and from that base then run logistic regression.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| X | y=x\*x | X | y= sin(x) | X | y=sqrt(x) |
| -6 | 36 | -6 | 0.279415 | -6 | #NUM! |
| -5 | 25 | -5 | 0.958924 | -5 | #NUM! |
| -4 | 16 | -4 | 0.756802 | -4 | #NUM! |
| -3 | 9 | -3 | -0.14112 | -3 | #NUM! |
| -2 | 4 | -2 | -0.909297 | -2 | #NUM! |
| -1 | 1 | -1 | -0.841471 | -1 | #NUM! |
| 0 | 0 | 0 | 0 | 0 | 0 |
| 1 | 1 | 1 | 0.841471 | 1 | 1 |
| 2 | 4 | 2 | 0.909297 | 2 | 1.414214 |
| 3 | 9 | 3 | 0.14112 | 3 | 1.732051 |
| 4 | 16 | 4 | -0.756802 | 4 | 2 |
| 5 | 25 | 5 | -0.958924 | 5 | 2.236068 |
| 6 | 36 | 6 | -0.279415 | 6 | 2.44949 |
| 7 | 49 | 7 | 0.656987 | 7 | 2.645751 |
| 8 | 64 | 8 | 0.989358 | 8 | 2.828427 |

So instead of using y = mx+b we can use y= x\*x +b, or y = sin(x) + b or y = sqrt(x) + b or combinations of them y = m\*(x\*x) + sqrt(x) + b, and for those hypothesis do gradient decent.

There is a way we can model these more complex functions still as a linear regression. The trick is to treat the other polynomial terms in the functions as an extra feature in y = mx+b for example:

y= x\*x + b

X2 = x\*x, add squared x as the second input of the model.

y = Ɵ0 + Ɵ1x1 + Ɵ2x2

To do this we must update our dataset to include the second column.

|  |  |  |
| --- | --- | --- |
| X1 | X2 | Y |
| -6 | 36 | 36 |
| -5 | 25 | 25 |
| -4 | 16 | 16 |
| -3 | 9 | 9 |
| -2 | 4 | 4 |
| -1 | 1 | 1 |
| 0 | 0 | 0 |
| 1 | 1 | 1 |
| 2 | 4 | 4 |
| 3 | 9 | 9 |
| 4 | 16 | 16 |
| 5 | 25 | 25 |

So now we have a simple linear regression and can use the same algorithm we have used before. This trick grows more expensive exponentially (with each new parameter) so you only want to do it for not too complex functions (more than a hundred). Though if need be, it can be done.