Design of our Model

INPUT: data for all 7 parameters of each of the top 100 websites:

Speed index
Load time
First byte
Start render
DOM elements

Algorithm:

Step1: Find min and max of each of the 7 parameters.

Step2: Normalize each parameter. After normalization, each parameter will have a value in between 0 and 1.

Step 3: iterate through each parameter and assign it points (from 1 to 100) on the basis of its normalized scale.

For example, after normalization speed index is 0.2 for one website. Thinking intuitively, speed value should be more for good ranking and good performance. Now, we design our scale from 0 to 1 in such a way that high speed = high points.

Example 1: calculating speed index

```
If (speed > =0 and speed <= .1) {
   //assign lowest points to the speed parameter variable. Eg = assign 10 points

If(speed >= 0.2 and speed =< .3){
  //Assign 20 points

If(speed > = .3 and speed =<.4
 //Assign 30 points
```

And so on – for speed index - the points we assign increases as the value of the parameter lies in the range closer to 1.

Example 2: calculating load time

```
If (load time > =0 and speed <= .1) {
```

//assign highest points to the speed parameter variable. Eg = assign 90 points

If(speed >= 0.1 and speed =< .2){
// Assign 80 points

If(speed > = .2 and speed =<.3)
//Assign 70 points

And so on .. for load time - the points decreases as the value of the parameter lying in the range is closer to 0.

Likewise, we do this step for each parameter for each website.

At the end of this step, we get **total optimal points for each website** having: speed points + load time points + …all other parameter points.

Step 4:

Now, we have the total points depending on the scale of each of the 100 websites calculated.

Step 5:

Compare the optimal points calculated now to get the ranking. As we have normalized the scale, and have the model having total optimal points for each website.

Step 6:

Sort the total points of each website and get the ranking info depending on the sort. One with the maximum points calculated is the ranked first and so on..

Step 7 : Once we have the ranking, if user wants to improve the ranking of a given website, so basically increase the points for that website.

//Calculate increase of points required to make a website No.1
Points_rank1_website = p1;
Current_ Points_of_website_user_wants_to_improve = p2;

Points_needed = (points of rank 1 website – point of website user wants to improve to rank 1)

//Then increase those many points

Points_of_website_user_wants_to_improve = Points_of_website_user_wants_to_improve + points_needed;

Step 7:

Depending on the new points, again sort the list and display ranking. Now user has the provision to improve the ranking of any website and display it.