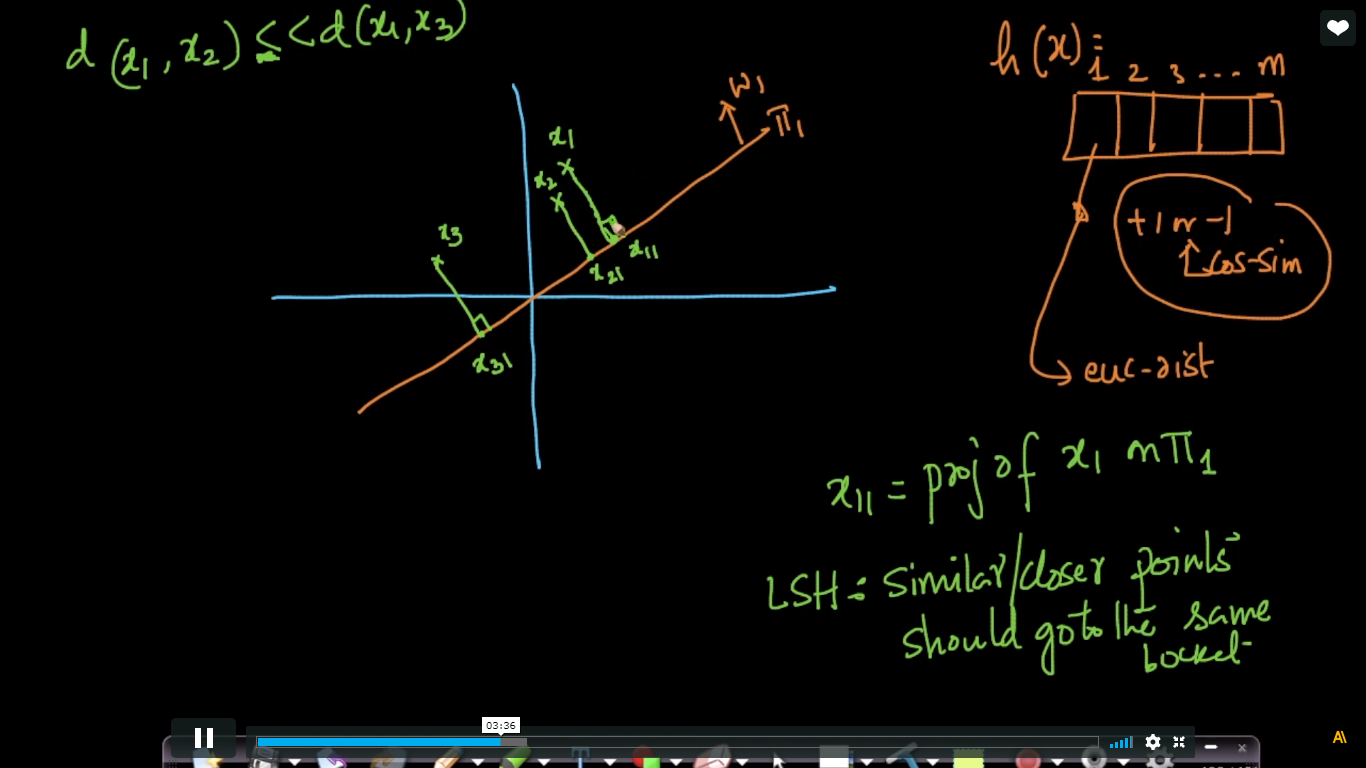
**LSH for euclidean distance**

Euclidean distance is a simplest extension LSH cosine similarity.

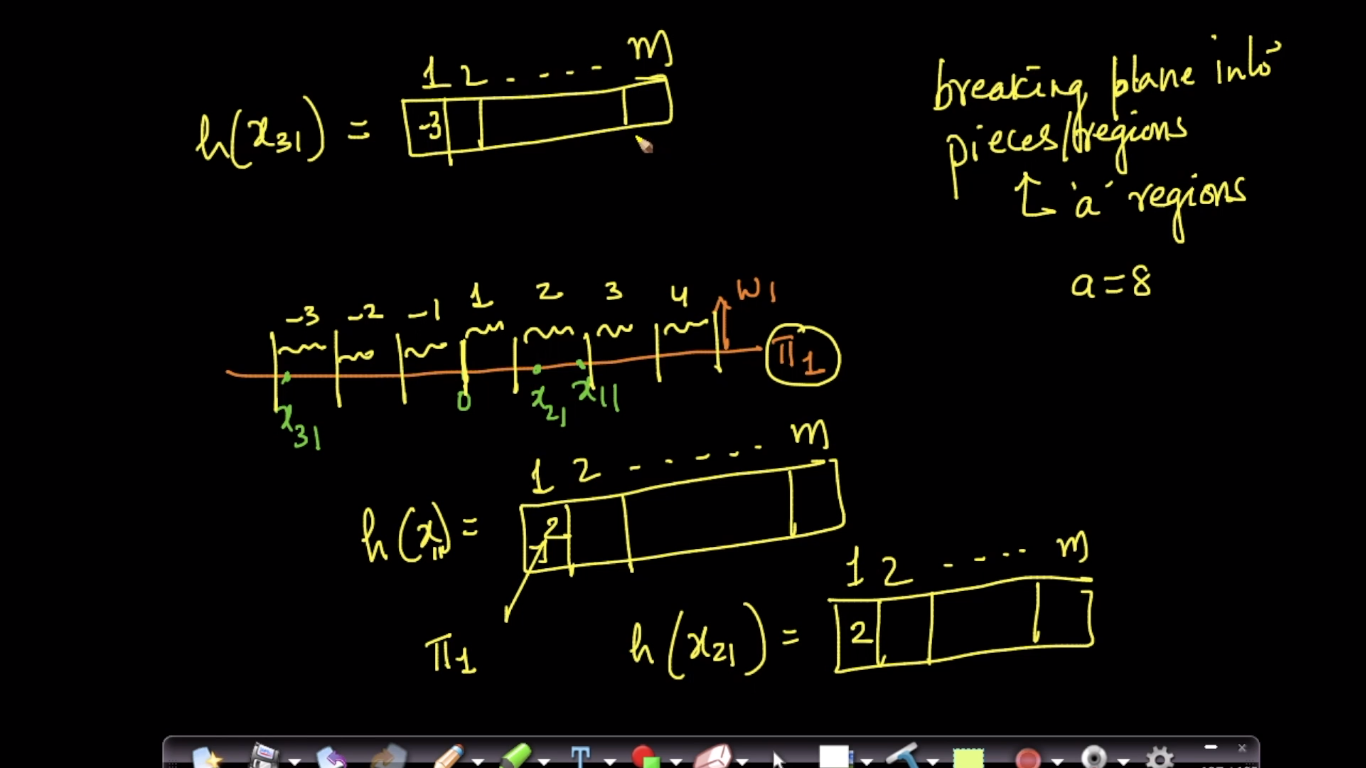


In this we project all points in the hyperplane. and as in lsh cosine similarity hash function store +1 or -1 , in Euclidean distance it store region number where point project on hyperplane.

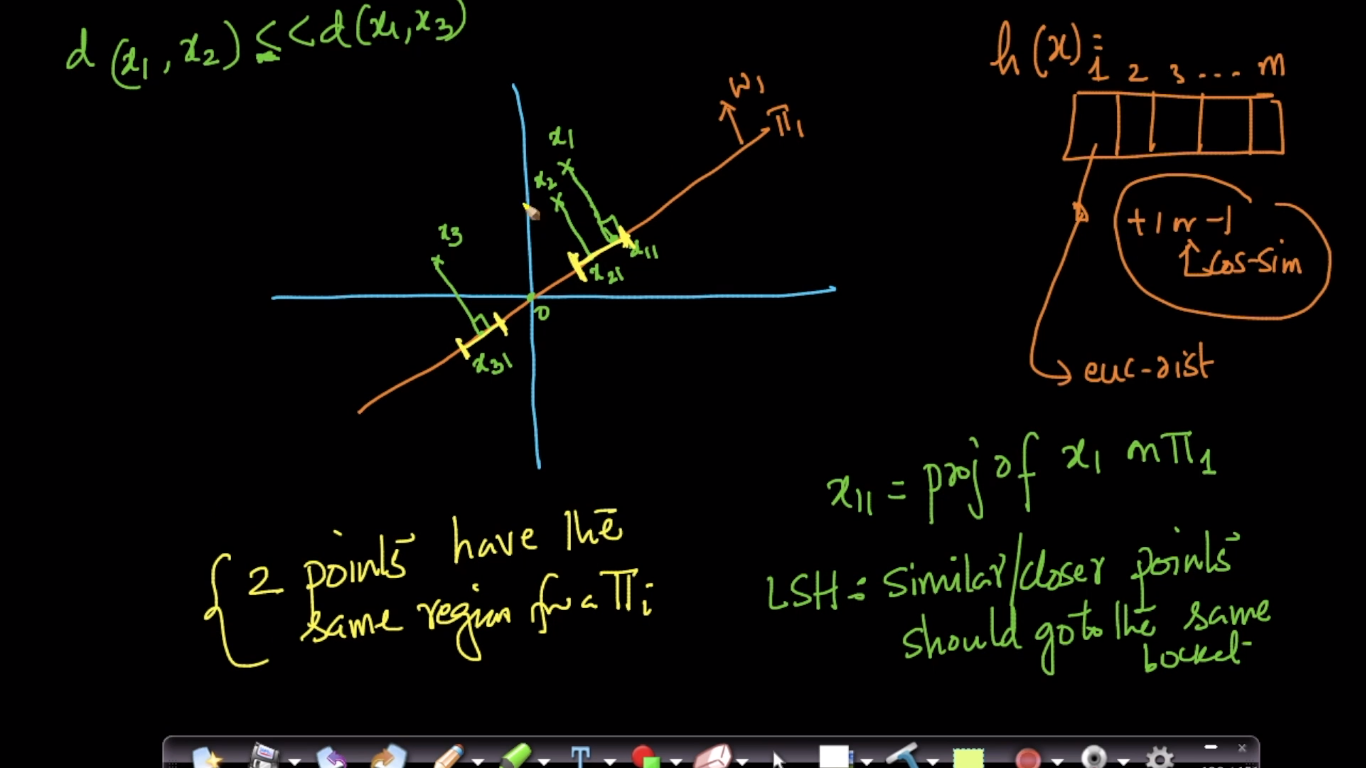
As shown below we break the hyperplane in a region here we take a=8.

Now after projecting x1 on plane it comes on 2nd region therefore in hash of x11 i.e h(x11) we store 2 in the 1st index(i.e of first hyperplane)

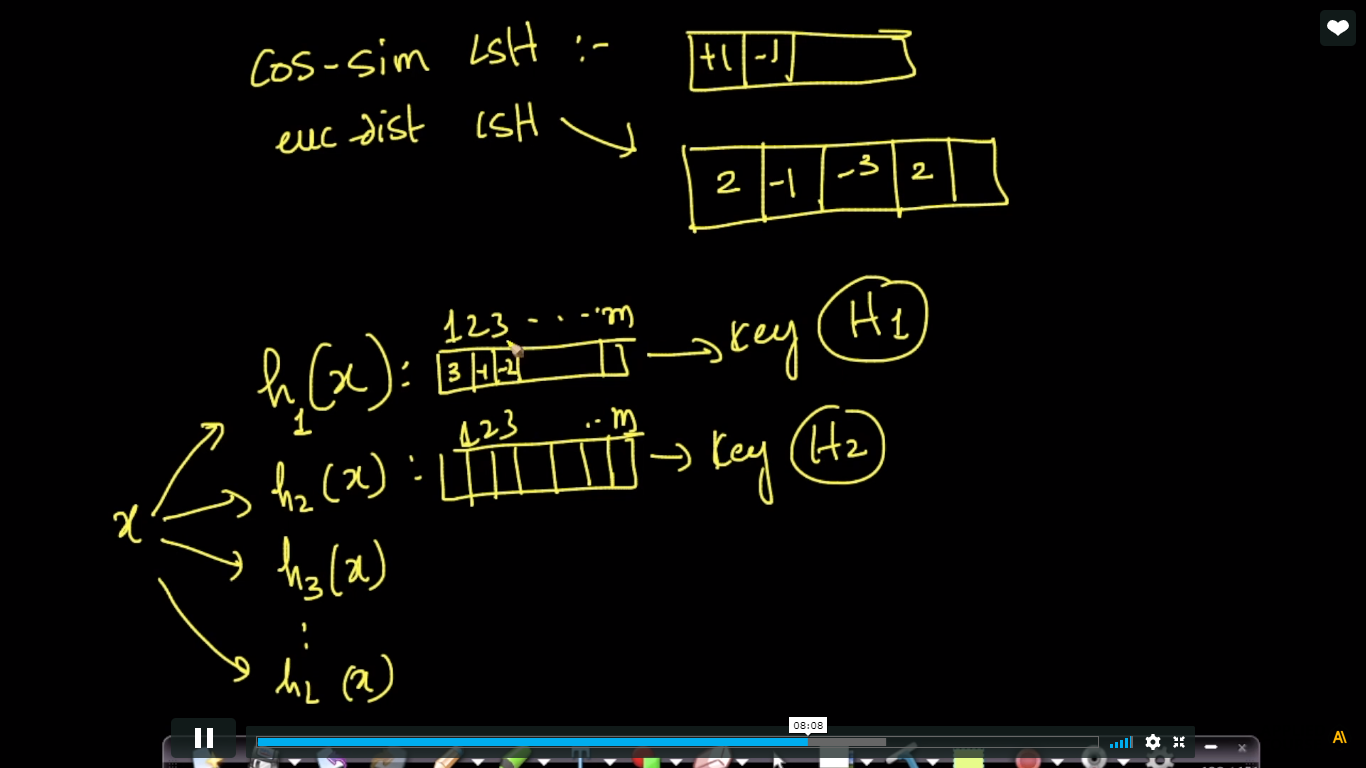
Similarly we do for remaining points.



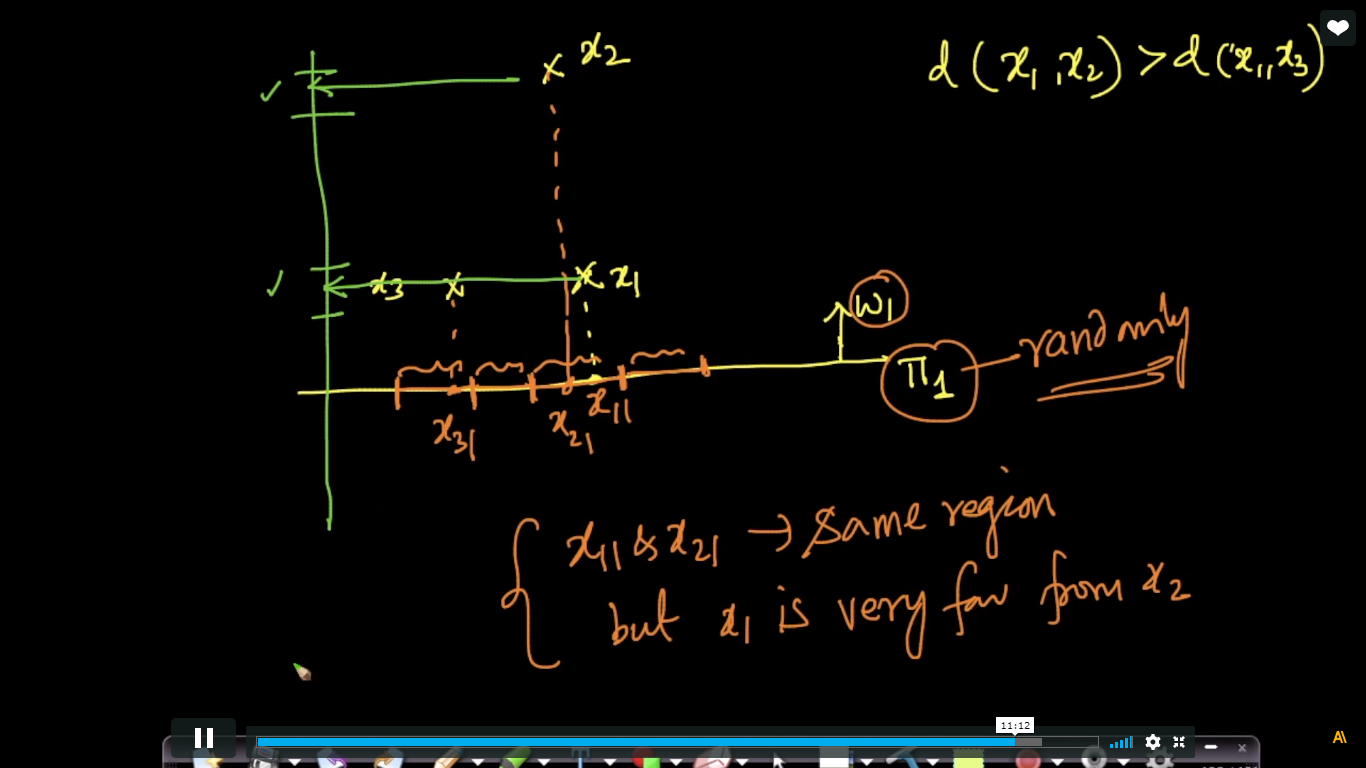
As shown below as x1 and x2 are closer therefore they are projected on same region.



As we generate hash of x1 similarly we generate hash of x2 where hash function are of m-dim. i.e h1(x) stores regions where x1 project on each hyperplane(we take m hyperplanes).



Below image show as x1 and x2 is far away still it falls on same region but as we take many hyperplanes randomly therefore they falls on different on other plane.



As lsh is probabilistic/randomized algorithm therefore it is not perfect.

