(*) The curse of Dimensionality

→ In the real world datasets, we often encounter data with hundreds or even thousands of features. Problems with more features/dimensions.

① As the majority of the machine learning algorithms rely on the calculation of distance for model building, and as the number of dimensions increases, it becomes more & more computationally intensive to create a model out of it.

Eg:

→ To calculate the distance b/w two points in just one dimension, we can just subtract the co-ordinates of one point from another.

→ For 2D, $= \sqrt{(x_1-x_2)^2+(y_1-y_2)^2}$

→ And for ND, it becomes: $\sqrt{(a_1-a_2)^2+(b_1-b_2)^2+\cdots+(n_1-n_2)^2}$

→ This is the effort for just two points, just imagine the no. of calculations involved for all datapoints.

② Hard to visualize the relationship b/w features: If we have an n-Dimensional dataset, the only solution left is to create either a 2D/3D graph out of it. Suppose we have 1000 features in the dataset, if we plan to create 2D graphs, we could need $(1000 \times 999)/2 = 499500$ combinations!! And we know it is not possible to spend time to analyze that many graphs to understand the relationship b/w the variables.

→ And like this, when we have huge no. of features, we need to ask the questions like -

① Are all the features really contributing to decision making.

② Is there a way to come to the same conclusion using lesser no. of features.

③ Is there a way to combine features to create a new feature and drop the old ones.

④ Is there a way to remodel features in a way to make them visually comprehensible.

The answer to all the above questions is :-

⊙ Dimensionality Reduction Technique

→ Dimensionality reduction is a feature selection technique using which we reduce the no. of features to be used for making a model without losing significant amount of information compared to original dataset.

→ In other words, a dimensionality reduction technique projects a data of higher dimension to a lower dimension subspace.

→ DR technique should be used before feeding the data to a machine learning algorithm. It reduces the space in which the distances are calculated, thereby improving ML algorithm performance