Decision Tree

Rose Quilan
Decide whether to play outside or not
Three variables – outlook (sunny, rainy, overcast), Humidity and wind speed

From where to start the tree-

Some maths behind that – Entropy – politics (before election lot of exit polls – go to people Democrat/Republicans) Now given this data entropy tells uncertainty/randomness. You cannot predict it's uncertain, so it's high entropy. Now if we have two datasets – Neck/Neck, and one is high probable (highly probable – less entropy). Whenever entropy is low prediction is easier.

How does it relate to the dataset –

What is entropy, information gain and gini index in decision tree algorithm

If we have 2 red and 2 blue, that group is 100% impure.
If we have 3 red and 1 blue, that group is either 75% or 81% pure, if we use Gini or Entropy respectively.

Gini index

$$Gini = 1 - \sum_j p_j^2$$

$$Gini\ Index = 1\text{-}(probability\_red^2 + probability\_blue^2) = 1\text{-}(0.5^2 + 0.5^2) = 0.5$$

The impurity measurement is 0.5 because we would incorrectly label gumballs wrong about half the time. Because this index is used in binary target variables (0,1), a gini index of 0.5 is the least pure score possible. Half is one type and half is the other. Dividing gini scores by 0.5 can help intuitively understand what the score represents. 0.5/0.5 = 1, meaning the grouping is as impure as possible (in a group with just 2 outcomes).

3 red and 1 blue:

$$Gini\ Index = 1\text{-}(probability\_red^2 + probability\_blue^2) = 1\text{-}(0.75^2 + 0.25^2) = 0.375$$

The impurity measurement here is 0.375. If we divide this by 0.5 for more intuitive understanding we will get 0.75, which is the probability of incorrectly/correctly labeling.

Entropy

$$Entropy = -\sum_j p_j \log_2 p_j$$

2 red and 2 blue:

$$Entropy = [(probability\_red) * log2(probability\_red)]—[(probability\_blue) * log2(probability\ of\ blue)] = [(2/4) * log2(2/4)] -[(2/4) * log2(2/4)] = 1$$

The impurity measurement is 1 here, as it's the maximum impurity obtainable.

3 red and 1 blue:

$$Entropy = [(probability\_red) * log2(probability\_red)]—[(probability\_blue) * log2(probability\ of\ blue)] = [(3/4) * log2(3/4)] -[(1/4) * log2(1/4)] = 0.811$$

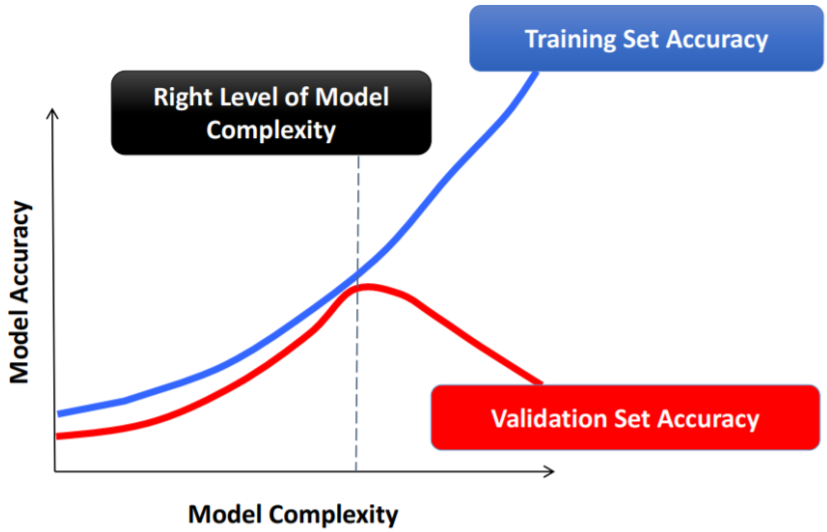
The purity/impurity measurement is 0.811 here, a bit worse than the gini score.

$$Information\ Gain\ (Parent, Child)$$

$$= Entropy(parent) - [p1(c1) * entropy(c1) + p(c2) * entropy(c2) + \ldots]$$

Gini's maximum impurity is 0.5 and maximum purity is 0
Entropy's maximum impurity is 1 and maximum purity is 0

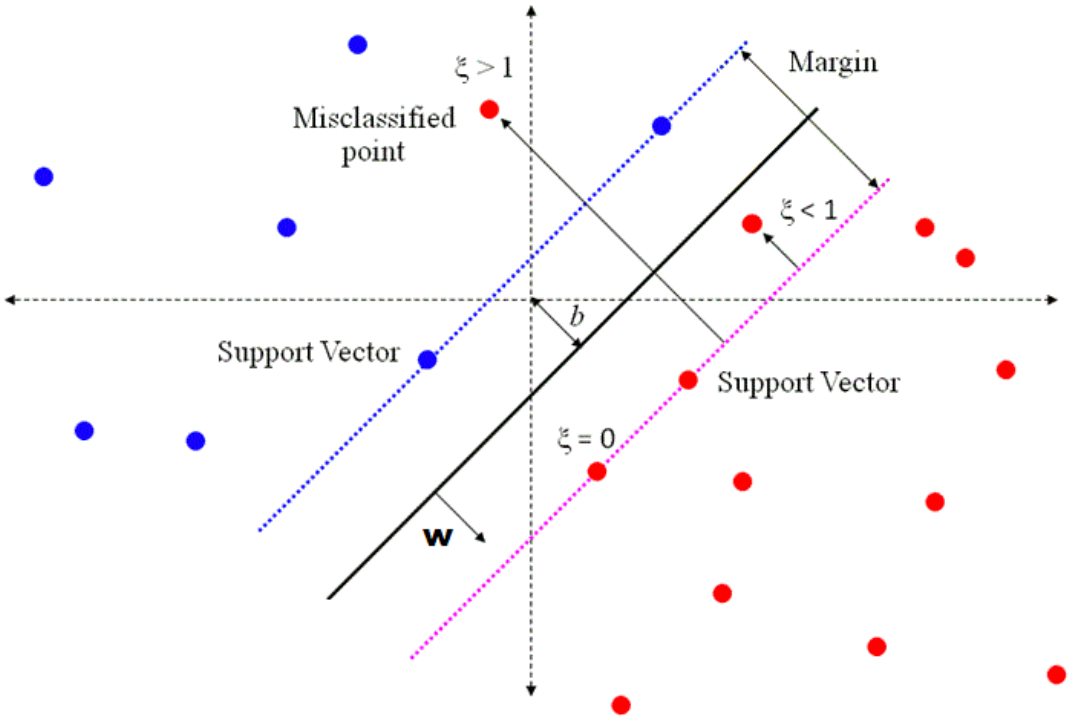
Generalize, don’t Memorize!

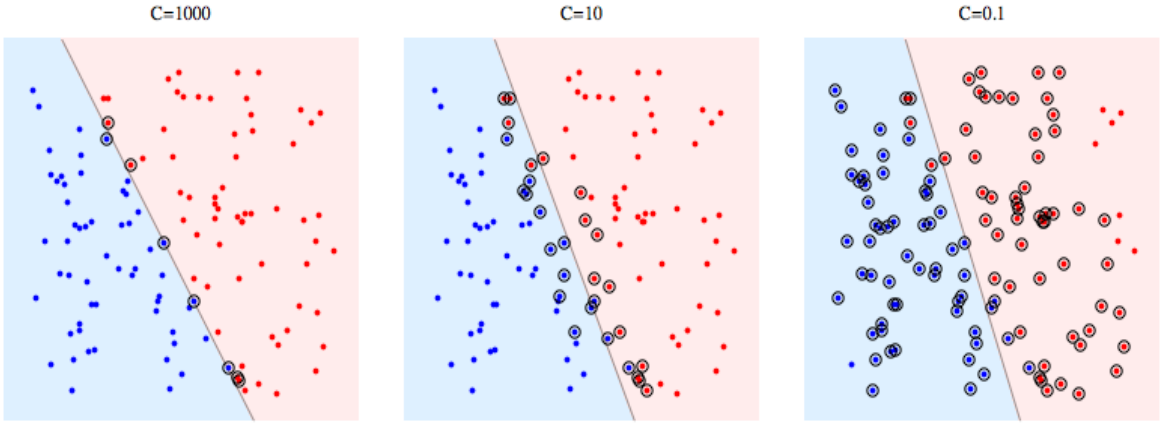


SVM

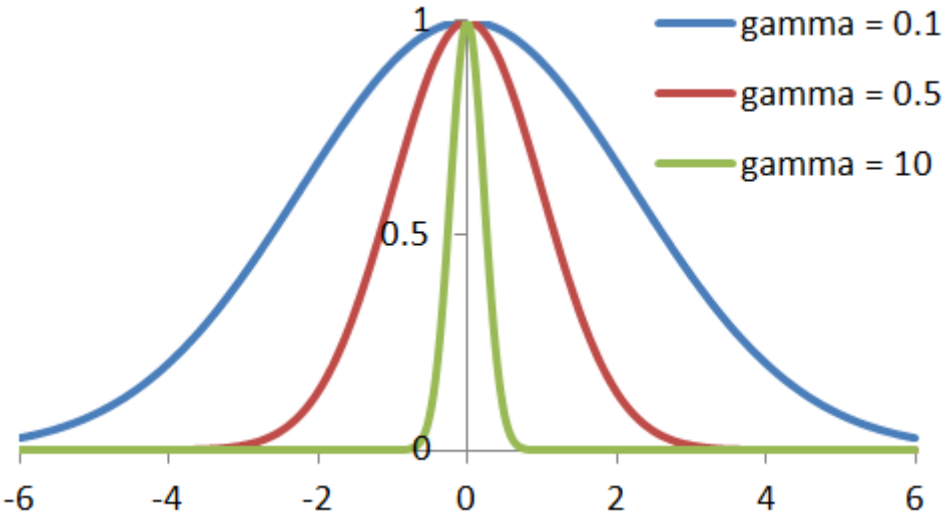
Hard margins and soft margin exist, support vector lies on margins
Three parameters that are important

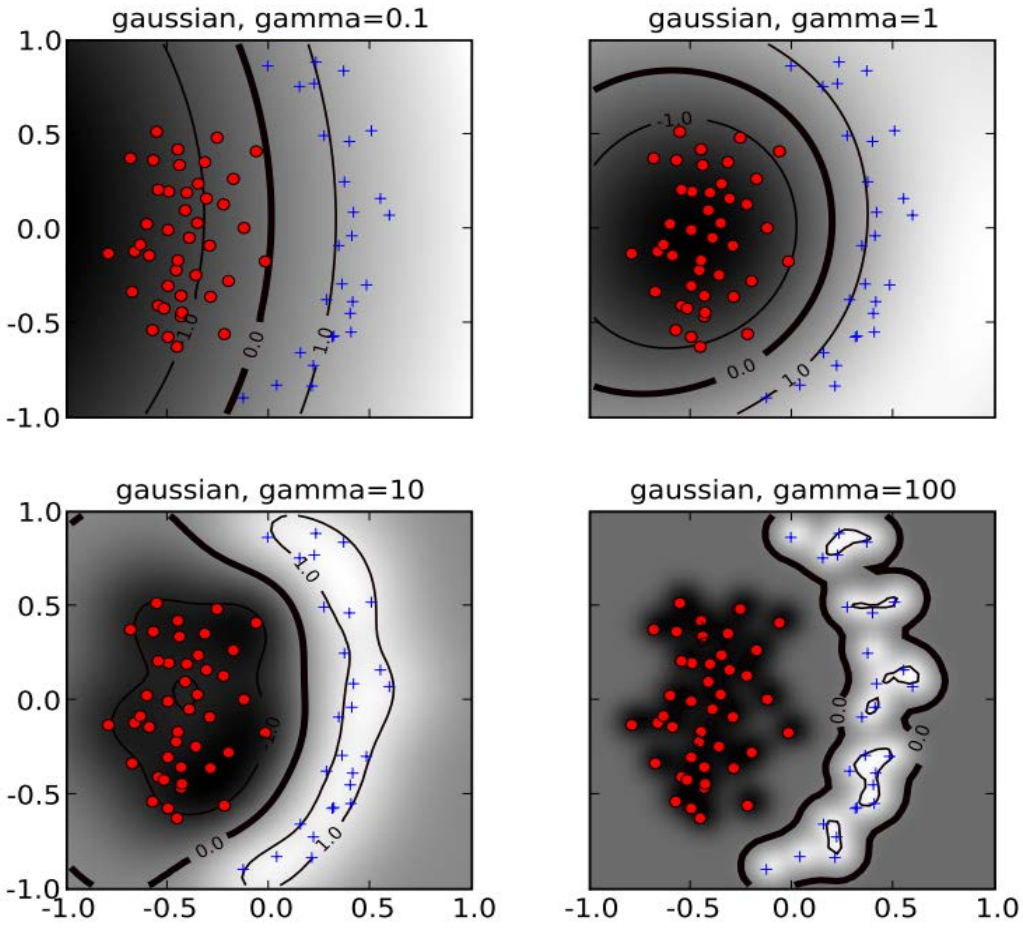
a) Cost – cost of misclassification, how much you would like to penalize the algo for misclassification. This value tells us how many points from the training set are allowed to be on the wrong side of the margin. SVM is always looking for the largest available margin between data groups. The margin is defined as a distance between decision hyperplane and the nearest point from a group, ie. the Support Vector. If all training data points are properly classified, the margin is usually relatively thin so it may be called a hard-margin. But we often deal with a set which is not linearly separable or have some outliers. Then, it's better to soften the margin and allow for some misclassification in exchange to the grouping generalization. Model might overfit if you take large c value– low bias and high variance. For low value – high bias and low variance.





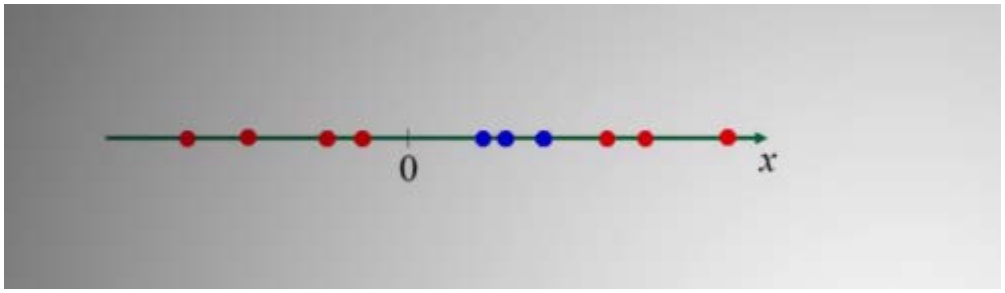
b) Gamma – higher dimensional space. 3D/4D space. In higher dimensional gamma will help to understand bias, small gamma (overfitting), larger gamma (underfitting)

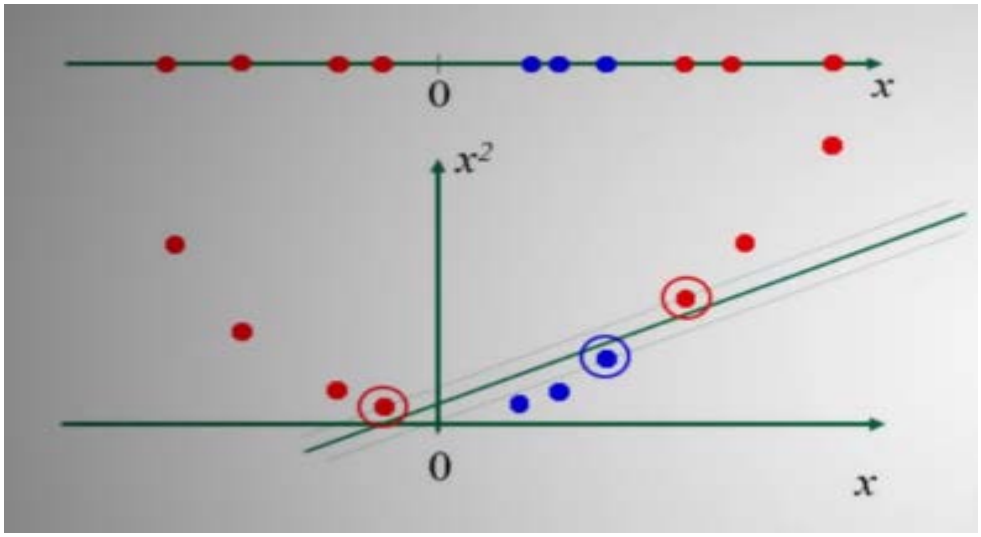


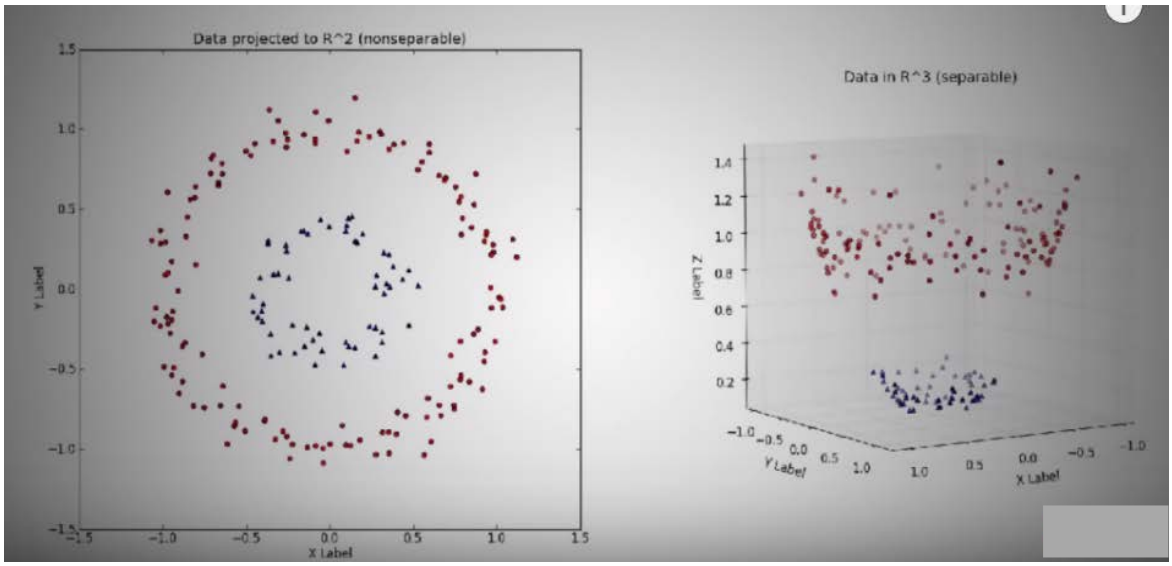


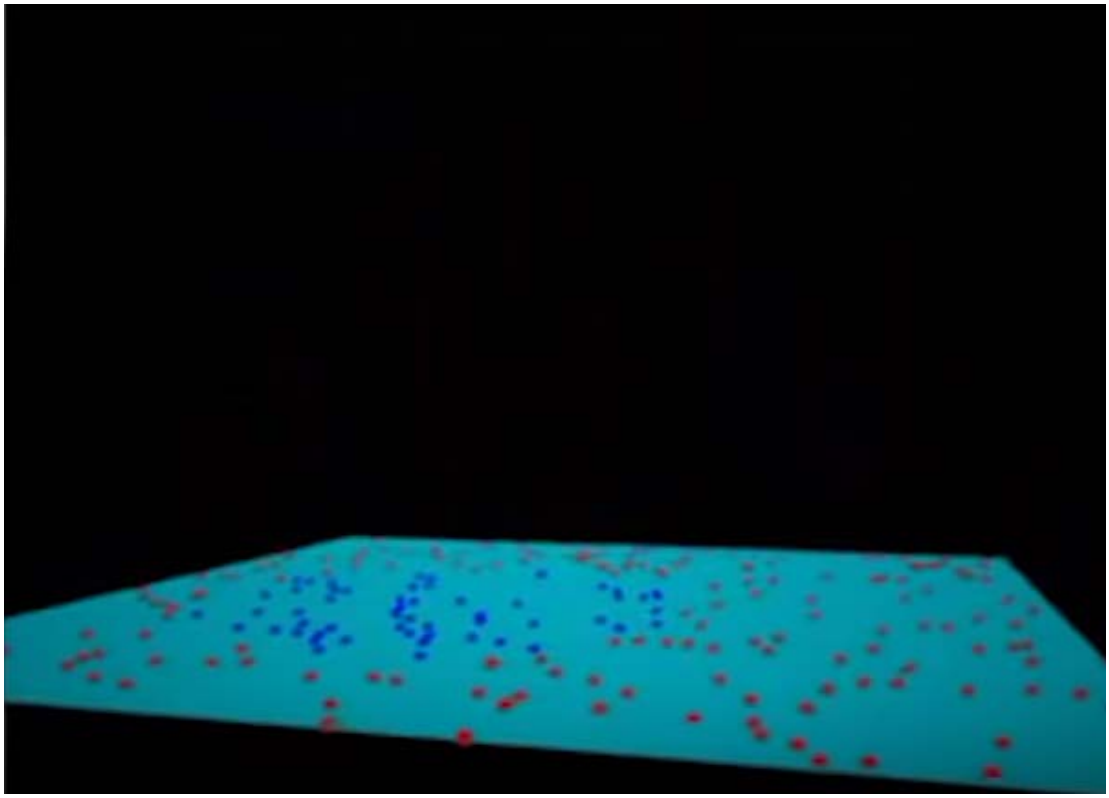
So lesser the gamma better it is as it avoids overfitting

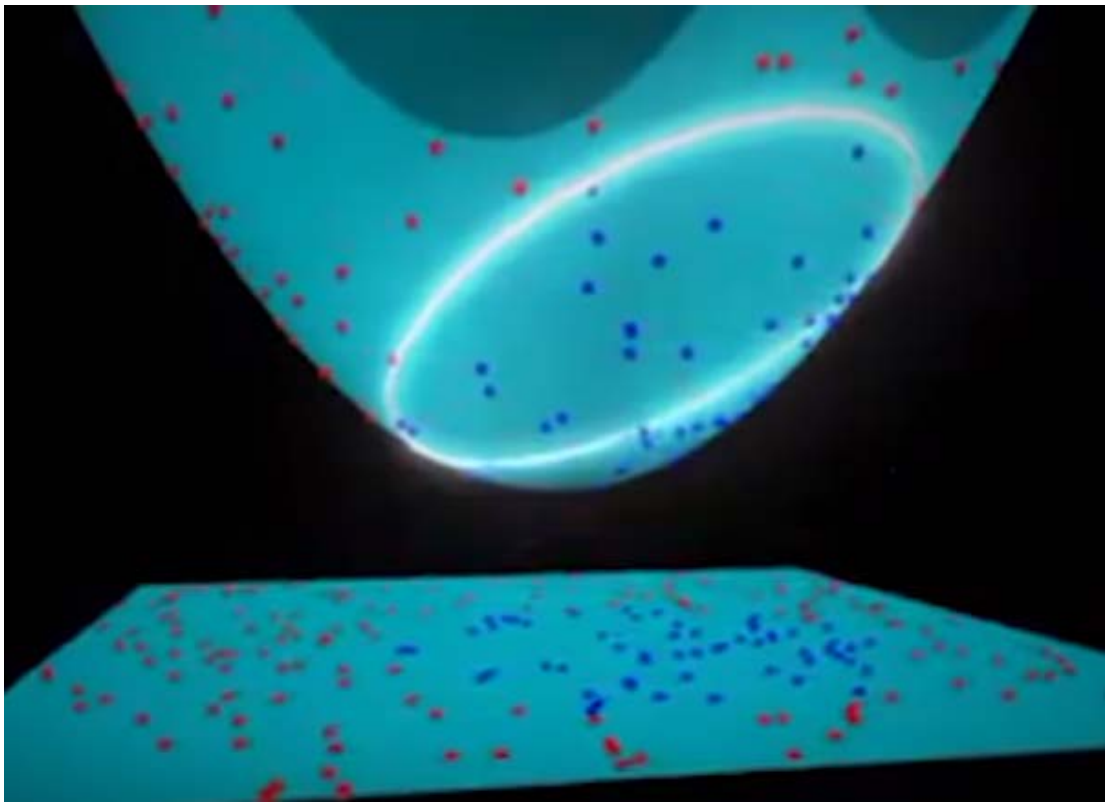
c) Kernel – function to represent data in high dimensional space. Which kernel to use comes from domain knowledge.

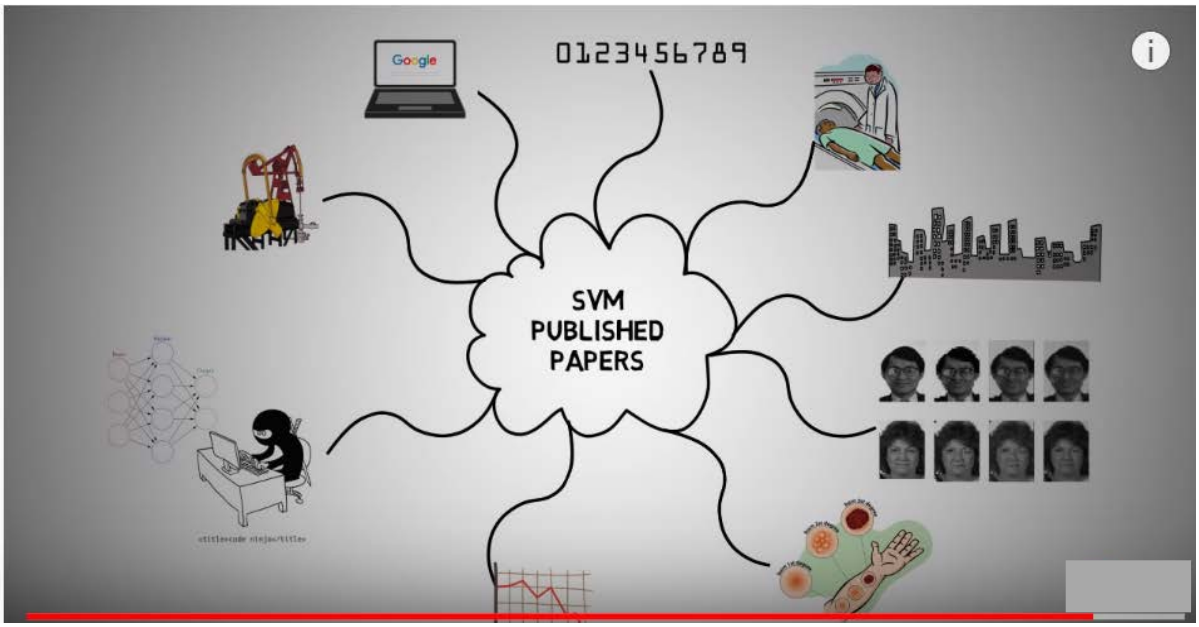











0123456789
SVM
PUBLISHED
PAPERS