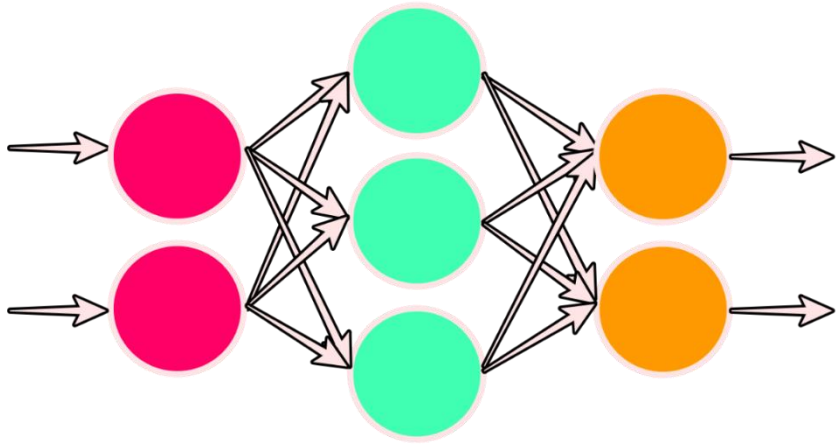
Siddhardhan

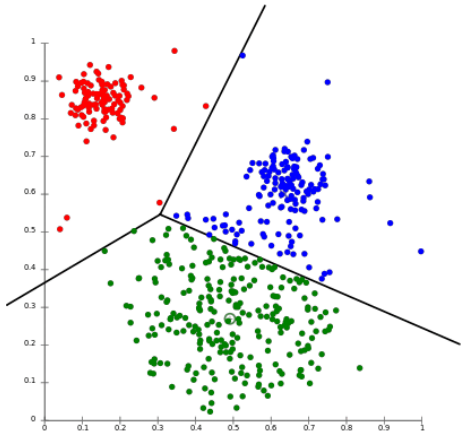
How to choose the right Machine Learning Model? (Model Selection)



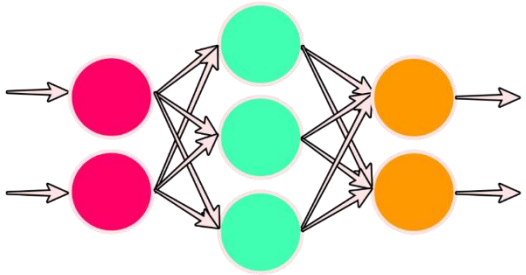
Model Selection

Model Selection in Machine Learning is the process of choosing the best suited model for a particular problem. Selecting a model depends on various factors such as the dataset, task, nature of the model, etc.

Logistic Regression



K-Means Clustering



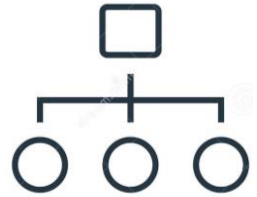
Neural Network

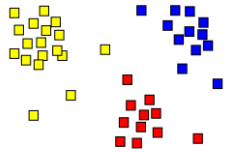
Model Selection

Models can be selected based on :

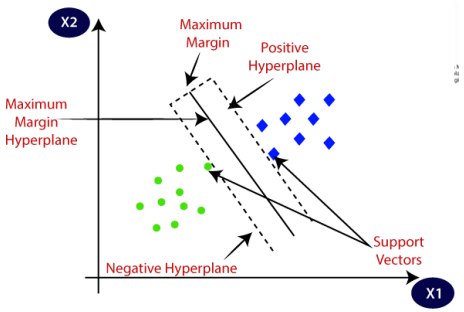
1. Type of Data available:
   a. Images & Videos – CNN
   b. Text data or Speech data – RNN
   c. Numerical data – SVM, Logistic Regression, Decision trees, etc.

2. Based on the task we need to carry out:
   a. Classification tasks – SVM, Logistic Regression, Decision trees, etc.
   b. Regression tasks – Linear regression, Random Forest, Polynomial regression, etc.
   c. Clustering tasks – K-Means Clustering, Hierarchical Clustering





Cross Validation

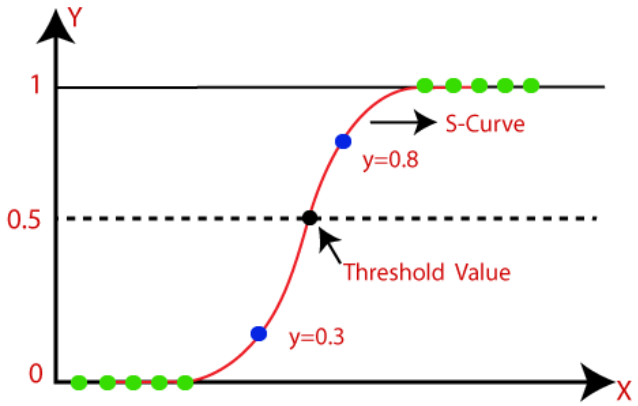


Support Vector Machine

	Dataset					*Accuracy*
Iteration 1	Train	Train	Train	Train	Test	88%
Iteration 2	Train	Train	Train	Test	Train	83%
Iteration 3	Train	Train	Test	Train	Train	86%
Iteration 4	Train	Test	Train	Train	Train	81%
Iteration 5	Test	Train	Train	Train	Train	84%

$$\text{Mean Accuracy} = \frac{88 + 83 + 86 + 81 + 84}{5} = 84.4\ \%$$

Cross Validation



Logistic Regression

	Dataset					*Accuracy*
Iteration 1	Train	Train	Train	Train	Test	90%
Iteration 2	Train	Train	Train	Test	Train	88%
Iteration 3	Train	Train	Test	Train	Train	86%
Iteration 4	Train	Test	Train	Train	Train	91%
Iteration 5	Test	Train	Train	Train	Train	85%

$$\text{Mean Accuracy} = \frac{90 + 88 + 86 + 91 + 85}{5} = 88\ \%$$

Cross Validation

✓ *Accuracy score for SVM = 84.4 %*

✓ *Accuracy score for Logistic Regression = 88 %*

Cross Validation Implementation:

```
>>> from sklearn import datasets, linear_model
>>> from sklearn.model_selection import cross_val_score
>>> diabetes = datasets.load_diabetes()
>>> X = diabetes.data[:150]
>>> y = diabetes.target[:150]
>>> lasso = linear_model.Lasso()
>>> print(cross_val_score(lasso, X, y, cv=3))
[0.33150734 0.08022311 0.03531764]
```