**Model Architecture**

In machine learning, Naïve Bayes classification is a straightforward and powerful algorithm for the classification task so, I implemented Naive Bayes Classification algorithm with Python and Scikit-Learn in this model. I built a Naive Bayes Classifier to predict whether a student will succeed in or not based in two stages first stage after completion of 20 percent of the course and after 50 percent.

Naïve Bayes Classifier uses the Bayes’ theorem to predict membership probabilities for each class such as the probability that given record or data point belongs to a particular class. The class with the highest probability is considered as the most likely class.

There are 3 types of Naïve Bayes algorithm. The 3 types are listed below:-

1. Gaussian Naïve Bayes
2. Multinomial Naïve Bayes
3. Bernoulli Naïve Bayes

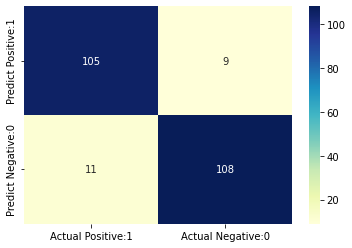
In this model I used Gaussian Naïve Bayes classifier.

When we have continuous attribute values, we made an assumption that the values associated with each class are distributed according to Gaussian or Normal distribution.

**Results**

In this project, I build a Gaussian Naïve Bayes Classifier model to predict whether a student will succeed in or not based in two stages first stage after completion of 20% of the course and after 50%. The model yields a very good performance as indicated by the model accuracy which was found to be 0.8069 at 20% of the course and 0.9142 at 50% of the course.

Confusion Matrix at 50%



A confusion matrix is a tool for summarizing the performance of a classification algorithm. A confusion matrix will give us a clear picture of classification model performance and the types of errors produced by the model. It gives us a summary of correct and incorrect predictions broken down by each category. The summary is represented in a tabular form.

Four types of outcomes are possible while evaluating a classification model performance. These four outcomes are described below:-

True Positives (TP) – True Positives occur when we predict an observation belongs to a certain class and the observation actually belongs to that class and the number of TP in this model 105.

True Negatives (TN) – True Negatives occur when we predict an observation does not belong to a certain class and the observation actually does not belong to that class and the number of TN in this model 108.

False Positives (FP) – False Positives occur when we predict an observation belongs to a certain class but the observation actually does not belong to that class. This type of error is called Type I error and the number of FP in this model 11.

False Negatives (FN) – False Negatives occur when we predict an observation does not belong to a certain class but the observation actually belongs to that class. This is a very serious error and it is called Type II error and the number of FN in this model 9.

The accuracy calculated as follows:

(TP+TN)/(TP+TN+FP+FN) = (105+108)/(105+108+11+9)= 0.9142