**LAB-03**

**NAME: MAHIKA JADHAV REGNO-21BDA53**

A)

**Gaussian Naive Bayes** :Gaussian Naïve Bayes is used in classification and it assumes that the features follow a normal distribution.

**Multinomial Naive Bayes:**

Multinomial Naïve Bayes is used for discrete counts.

Eg: we have a text classification problem.

Here we can consider Bernoulli trials which is one step further and instead of “word occurring in the document”, we have “count how often word occurs in the document”, you can think of it as **“number of times outcome number x\_i is observed over the n trials”.**

**Complement Naive Bayes:**

 In complement Naive Bayes, instead of calculating the probability of an item belonging to a certain class, we calculate the probability of the item belonging to all the classes. This is the literal meaning of the word, complement and hence is called Complement Naive Bayes.

Bernoulli Naive Bayes:

The binomial model is useful if your feature vectors are binary (i.e. zeros and ones). One application would be text classification with ‘bag of words’ model where the 1s & 0s are “word occurs in the document” and “word does not occur in the document” respectively.

Categorical Naive Bayes:

The categorical Naive Bayes classifier is **suitable for classification with discrete features that are categorically distributed**. The categories of each feature are drawn from a categorical distribution.

Out-of-core naive Bayes model fitting:

Naive Bayes models can be used to tackle large scale classification problems for which the full training set might not fit in memory. To handle this case, [**MultinomialNB**](https://scikit-learn.org/stable/modules/generated/sklearn.naive_bayes.MultinomialNB.html#sklearn.naive_bayes.MultinomialNB) , **[BernoulliNB](https://scikit-learn.org/stable/modules/generated/sklearn.naive_bayes.BernoulliNB.html" \l "sklearn.naive_bayes.BernoulliNB" \o "sklearn.naive_bayes.BernoulliNB)**, and **[GaussianNB](https://scikit-learn.org/stable/modules/generated/sklearn.naive_bayes.GaussianNB.html" \l "sklearn.naive_bayes.GaussianNB" \o "sklearn.naive_bayes.GaussianNB)** expose a partial\_fit method that can be used incrementally as done with other classifiers as demonstrated in [Out-of-core classification of text documents](https://scikit-learn.org/stable/auto_examples/applications/plot_out_of_core_classification.html#sphx-glr-auto-examples-applications-plot-out-of-core-classification-py). All naive Bayes classifiers support sample weighting.

C) What is Jaccard and Cosine Similarity?

Jaccard similarity takes only unique set of words for each sentence / document while cosine similarity takes total length of the vectors.