**LAB-03**

(B) In a word document,

a. Write the difference between the following:

i. Gaussian Naive Bayes,

ii. Multinomial Naive Bayes,

iii. Complement Naive Bayes,

iv. Bernoulli Naive Bayes,

v. Categorical Naive Bayes,

vi. Out-of-core naive Bayes model fitting

b. Define which text preprocessing and text transformation steps did you use for the above.

c. What is Jaccard and Cosine Similarity?

ANSWERS

* + 1. Gaussian Naive Bayes supports continuous valued features and models each as conforming to a Gaussian (normal) distribution. An approach to create a simple model is to assume that the data is described by a Gaussian distribution with no co-variance (independent dimensions) between dimensions.
    2. The Multinomial Naive Bayes algorithm is a Bayesian learning approach popular in Natural Language Processing (NLP). The program guesses the tag of a text, such as an email or a newspaper story, using the Bayes theorem. It calculates each tag's likelihood for a given sample and outputs the tag with the greatest chance.
    3. 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.
    4. Bernoulli Naive Bayes. Bernoulli NB implements the naive Bayes training and classification algorithms for data that is distributed according to multivariate Bernoulli distributions; i.e., there may be multiple features but each one is assumed to be a binary-valued (Bernoulli, Boolean) variable.
    5. 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.
    6. 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, [**Multinomial NB**](https://scikit-learn.org/stable/modules/generated/sklearn.naive_bayes.MultinomialNB.html#sklearn.naive_bayes.MultinomialNB), [**Bernoulli NB**](https://scikit-learn.org/stable/modules/generated/sklearn.naive_bayes.BernoulliNB.html#sklearn.naive_bayes.BernoulliNB), and [**Gaussian NB**](https://scikit-learn.org/stable/modules/generated/sklearn.naive_bayes.GaussianNB.html#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.
  1. (doubt)
  2. Jaccard similarity takes only unique set of words for each sentence / document while cosine similarity takes total length of the vectors.

Jaccard Similarity is a common proximity measurement used to compute the similarity between two objects, such as two text documents. Jaccard similarity can be used to find the similarity between two asymmetric binary vectors or to find the similarity between two sets.

Cosine similarity measures the similarity between two vectors of an inner product space. It is measured by the cosine of the angle between two vectors and determines whether two vectors are pointing in roughly the same direction. It is often used to measure document similarity in text analysis.