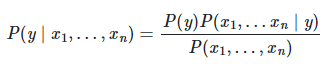
**ML LAB- 3**

**AYESHA SHARIFF – 21BDA18**

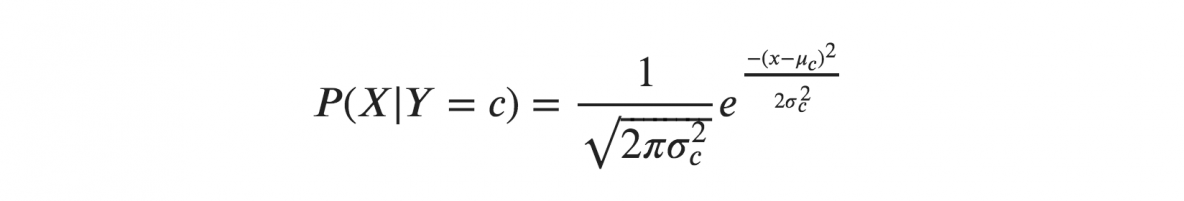
Naive Bayes algorithm is one of the well-known supervised classification algorithms. It bases on the Bayes theorem, it is very fast and good enough for text classification. Naive Bayes methods are a set of supervised learning algorithms based on applying Bayes’ theorem with the “naive” assumption of conditional independence between every pair of features given the value of the class variable. Bayes’ theorem states the following relationship, given class variable y and dependent feature vector x1 through xn.



1. Gaussian Naive Bayes

Gaussian Naive Bayes is a probabilistic classification algorithm based on applying Bayes' theorem with strong independence assumptions. In the context of classification, independence refers to the idea that the presence of one value of a feature does not influence the presence of another (unlike independence in probability theory). Naive refers to the use of an assumption that the features of an object are independent of one another. In the context of machine learning, naive Bayes classifiers are known to be highly expressive, scalable, and reasonably accurate, but their performance deteriorates rapidly with the growth of the training set.

we assume that X’s follow a Gaussian or normal distribution, we must substitute the probability density of the normal distribution and name it Gaussian Naïve Bayes. To compute this formula, you need the mean and variance of X.



In the above formulae, sigma and mu is the variance and mean of the continuous

1. Multinomial Naïve Bayes

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.

1. Complement Naïve Bayes

Complement Naive Bayes is somewhat a modification of the standard Multinomial Naive Bayes algorithm. Multinomial Naive Bayes is not able to do very well with unstable data. Imbalanced data sets are instances where the number of instances belonging to a particular class is greater than the number of instances belonging to different classes. This implies the spread of the examples is not even. This kind of data can be difficult to analyse as models can easily overfit this data to benefit a class with a larger instance.

1. Bernoulli Naïve Bayes

This is used for discrete data and it works on Bernoulli distribution. The main feature of Bernoulli Naive Bayes is that it accepts features only as binary values like true or false, yes or no, success or failure, 0 or 1 and so on. So when the feature values are binary we know that we have to use Bernoulli Naive Bayes classifier.

Bernoulli Naive Bayes Classifier is based on the following rule:

P ( x i ∣ y ) = P ( i ∣ y ) x i + ( 1 − P ( i ∣ y ) ) ( 1 − x i )

1. Categorical Naïve Bayes

It is suitable for classification with discrete features which assumes categorically distribution for each feature. The features should to encoded using label encoding techniques such that each category would be mapped to a unique number.

1. Out-of-core naive Bayes model fittingThis classifier is used to handle cases of large scale classification problems for which the complete training dataset might not fit in the memory.

What is Jaccard and Cosine Similarity?

Jaccard Similarity:-

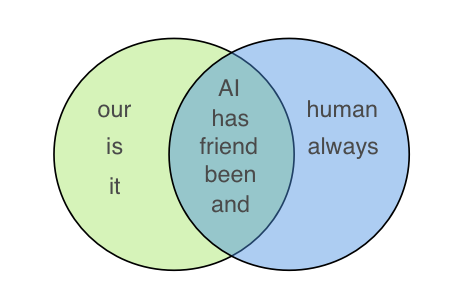
Jaccard similarity or intersection over union is defined as size of intersection divided by size of union of two sets. Let’s take example of two sentences:

Sentence 1: AI is our friend and it has been friendly

Sentence 2: AI and humans have always been friendly

In order to calculate similarity using Jaccard similarity, we will first perform lemmatization to reduce words to the same root word. In our case, “friend” and “friendly” will both become “friend”, “has” and “have” will both become “has”. Drawing a Venn diagram of the two sentences we get:

Venn Diagram for calculations

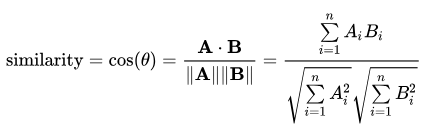


Venn Diagram of the two sentences for Jaccard similarity

For the above two sentences, we get Jaccard similarity of 5/(5+3+2) = 0.5 which is size of intersection of the set divided by total size of set. One thing to note here is that since we use sets, “friend” appeared twice in Sentence 1 but it did not affect our calculations

Cosine Similarity:-

The cosine similarity is measure the cosine angle between the two vectors. For cosien we have to convert all sentences to vectors. For converting to vector we can use TF-IDF, Word2Vec. The formula for cosine similarity is:-



**Major difference between jaccard and cosine similarity:-**

1. Jaccard Similarity takes set of unique length of words instead cosine similarity takes whole sentence vector
2. If data duplication is not matter then its better to use jaccard similarity else cosine similarity is good for measuring the similarity between two vectors even if the data duplication is there.