21BDA41

PART-B

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

Naive Bayes classifier is a general term which refers to conditional independence of each of the features in the model, while Multinomial Naive Bayes classifier is a specific instance of a Naive Bayes classifier which uses a multinomial distribution for each of the features.

1. Gaussian Naive Bayes

This approach is built on the assumption of a normal distribution of probabilities. It means, that spam and not-spam classes of messages have frequencies of the words from vocabulary distributed by the Gaussian law:

$$P(x_i \mid y) = \frac{1}{\sqrt{2\pi\sigma_y^2}} \exp\left(-\frac{(x_i - \mu_y)^2}{2\sigma_y^2}\right)$$

The formula is based on the mean (**μ**) and Bessel corrected variance (**σ**) of the frequency of each word in the class of messages.

2. Multinomial Naive Bayes

Multinomial classification suits best for the discrete values like word counts. So, we expect it to show the best accuracy. In this case distribution of probabilities for each event bases on the formula:

$$\hat{\theta}_{yi} = \frac{N_{yi} + \alpha}{N_y + \alpha n}$$

Ny is the total number of features of the event **y** (total number of words in all spam messages), **Nyi** — count of each feature (summary number of repetitions of a word in all spam messages), **n** — the number of features (number of words in the vocabulary) and **α** is a smoothing Laplace parameter to discard the influence of words absent in the vocabulary. The same formula applies to the set of not-spam messages.

The formula is based on the mean (**μ**) and Bessel corrected variance (**σ**) of the frequency of each word in the class of messages.

3. Complement Naive Bayes

This approach is almost the same as the Multinomial, though now we count the occurrences of a word in the complement to the class. For example, for the spam message we will count the repetitions of each word in all the non-spam messages:

$$\hat{\theta}_{\tilde{c}i} = \frac{N_{\tilde{c}i} + \alpha_i}{N_{\tilde{c}} + \alpha}$$

Nc — total number of words in the opposite class (for the spam parameter — number of non-spam words), Nci — repetitions of a word in the opposite class (for a word from spam message — the number of repetitions in all non-spam messages). We also use the same smoothing parameters. After the calculation of basic values we start working with the real parameters:

$$\hat{w}_{ci} = \frac{\log \hat{\theta}_{ci}}{\sum_k |\log \hat{\theta}_{ck}|}$$

It is the weight for each word in the message of k words. The final decision is calculated by the formula:

$$\hat{c} = \arg\min_c \sum_i t_i w_{ci}$$

So, the classification result is the class with the minimum value of the sum of weights for each word in the message.

4. Bernoulli Naive Bayes

Bernoulli formula is close to the multinomial one, though the input is the set of boolean values (the word is present in the message or not) instead of the set of frequencies.

$$P(x_i \mid y) = P(i \mid y)x_i + (1 - P(i \mid y))(1 - x_i)$$

So, the algorithm explicitly penalizes the non-occurrence of a feature (word in the message is absent in the vocabulary) while the multinomial approach uses the smoothing parameter for the absent values. `sklearn` Bernoulli algorithm binarizes input values, so, no additional actions required.

5. Categorical Naive Bayes

Categorical Naive Bayes is suitable for the categorical values — if the example has the set of features or not. In our case, it means, that the vocabulary is treated as the set of features, and the occurrence of a word in the message is treated as the matching with the feature. All formulas are the same as for the multinomial approach but with the occurrences instead of repetitions.

Since the algorithm needs categorical values, we convert the frequencies of words to the presence of words: 1 — the message contains the word, 0 — the word is absent in the message.

6.Out-of-Core Naive Bayes — This classifier is used to handle cases of large scale classification problems for which the complete training dataset might not fit in the memory.

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.

Cosine Similarity

Cosine similarity measures the similarity between two vectors by taking the cosine of the angle the two vectors make in their dot product space. If the angle is zero, their similarity is one, the larger the angle is, the smaller their similarity. The measure is independent of vector length (the two vectors can even be of different length), which makes it a commonly used measure for high-dimensional spaces

Jaccard Similarity

Jaccard similarity measures the similarity between two nominal attributes by taking the intersection of both and divide it by their union. In terms of the above definitions this gives [14]; (2) A11 = total number of binary values where both vectors have the value 1. A01 = total number of binary values where first vector has value 1, other has value 0. A10 = total number of binary values where first vector has value 0, other has value 1. A00 = total number of binary values where both vectors have the value 0