Dayananda Sagar University

School of Engineering, Hosur Main Road, Kudlu Gate, Bengaluru-560 068

ARTIFICIAL INTELLIGENCE Lab-II

Program 4

Naïve Bayes theorem

20AM3602

SEMESTER – VI

Course Code: 20AM3602

Prof. Pradeep Kumar K

Dept. of CS&E (AIML)

DSU, Bangalore

Naïve Bayes Theorem

- Bayes theorem is one of the earliest probabilistic inference algorithms developed by Reverend Bayes (which he used to try and infer the existence of God no less) and still performs extremely well for certain use cases.
- It is a classification technique based on Bayes' Theorem with an assumption of independence among predictors. In simple terms, a Naive Bayes classifier assumes that the presence of a particular feature in a class is unrelated to the presence of any other feature. For example, a fruit may be considered to be an apple if it is red, round, and about 3 inches in diameter. Even if these features depend on each other or upon the existence of the other features, all of these properties independently contribute to the probability that this fruit is an apple and that is why it is known as 'Naive'.

20AM3602

BAYES THEOREM IMPLEMENTATION

- Let's say we are trying to find the odds of an individual having diabetes, given that he or she was tested for it and got a positive result. In the medical field, such probabilies play a very important role as it usually deals with life and death situatuations.

- **P(D)** is the probability of a person having Diabetes. It's value is 0.01 or in other words, 1% of the general population has diabetes(Disclaimer: these values are assumptions and are not reflective of any medical study).

- **P(Pos)** is the probability of getting a positive test result.

- **P(Neg)** is the probability of getting a negative test result.

- **P(Pos|D)** is the probability of getting a positive result on a test done for detecting diabetes, given that you have diabetes. This has a value 0.9. In other words the test is correct 90% of the time. This is also called the Sensitivity or True Positive Rate.

- **P(Neg|~D)** is the probability of getting a negative result on a test done for detecting diabetes, given that you do not have diabetes. This also has a value of 0.9 and is therefore correct, 90% of the time. This is also called the Specificity or True Negative Rate.

GAUSSIAN NAIVE BAYES CLASSIFIER

"Gaussian" because this is a normal distribution

This is our prior belief

$$P(\text{class} \mid \text{data}) = \frac{P(\text{data} \mid \text{class}) \times P(\text{class})}{P(\text{data})}$$

We don't calculate this in naive bayes classifiers

The Bayes formula is as follows:

$$P(A|B) = \frac{P(B|A)\ P(A)}{P(B)}$$

- **P(A)** is the prior probability of A occuring independantly. In our example this is P(D). This value is given to us.
- **P(B)** is the prior probability of B occuring independantly. In our example this is P(Pos).
- **P(A|B)** is the posterior probability that A occurs given B. In our example this is P(D|Pos). That is, the probability of an individual having diabetes, given that, that individual got a positive test result. This is the value that we are looking to calculate.
- **P(B|A)** is the likelihood probability of B occuring, given A. In our example this is **P(Pos|D).** This value is given to us.

Putting our values into the formula for Bayes theorem we get:

P(D|Pos) = (P(D) * P(Pos|D) / P(Pos)

The probability of getting a positive test result P(Pos) can be calulated using the Sensitivity and Specificity as follows:

P(Pos) = [P(D) * Sensitivity] + [P(~D) * (1-Specificity))]

```
# P(D)
p_diabetes = 0.01

# P(~D)
p_no_diabetes = 0.99

# Sensitivity or P(Pos|D)
p_pos_diabetes = 0.9

# Specificity or P(Neg/~D)
p_neg_no_diabetes = 0.9

# P(Pos)
p_pos = (p_diabetes * p_pos_diabetes) + (p_no_diabetes * (1 - p_neg_no_diabetes))
print('The probability of getting a positive test result P(Pos) is: {}',format(p_pos))
```

```
The probability of getting a positive test result P(Pos) is: {} 0.10799999999999998
```

```
# P(D|Pos)
p_diabetes_pos = (p_diabetes * p_pos_diabetes) / p_pos
print('Probability of an individual having diabetes, given that that individual got a positive test result is:\
',format(p_diabetes_pos))
```

```
Probability of an individual having diabetes, given that that individual got a positive test result is: 0.08333333333
333336
```

Naive Bayes Theorem - Colabora

colab.research.google.com/drive/1KexjOyo59XW8qCsWBg5XVecZYpRbrmO9?authuser=1#scrollTo=GlPP6NoCkxEe

Naive Bayes Theorem

File Edit View Insert Runtime Tools Help All changes saved

Comment Share

+ Code + Text

RAM Disk

[4] 0s
```
p_neg_no_diabetes = 0.9
```

[14] 0s
```
#p(pos)
p_pos = (p_diabetes *p_pos_diabetes) + (p_no_diabetes *(1-p_neg_no_diabetes))
print('The probability of getting a positive test result p(pos) is: {}', format(p_pos))
```

```
The probability of getting a positive test result p(pos) is: {} 0.10799999999999998
```

[20] 0s
```
in calculate p(D|Pos)

_diabetes * p_pos_diabetes) / p_pos
of an individual having diabetes:given that individual got a positive test result is :', format(p_diabetes_pos))
```

```
Probability of an individual having diabetes:given that individual got a positive test result is : 0.0833333333333333
```

0s completed at 11:10 AM

27°C Mostly cloudy ENG 11:11 AM 09-03-2023

sh text

When to Use Naive Bayes

1. High-speed training and prediction
2. Probability prediction based on data
3. Easy to interpret
4. Non-tunable parameters

20AM3602

Implement naïve bayes theorem to classify the English text

- Text Analysis is a major application field for machine learning algorithms. However the raw data, a sequence of symbols (i.e. strings) cannot be fed directly to the algorithms themselves as most of them expect numerical feature vectors with a fixed size rather than the raw text documents with variable length.

Naive Bayes algorithm

- Naive Bayes classifiers are a collection of classification algorithms based on Bayes' Theorem. It is not a single algorithm but a family of algorithms where all of them share a common principle, i.e. every pair of features being classified is independent of each other.
- The dataset is divided into two parts, namely, feature matrix and the response/target vector.
- The Feature matrix (X) contains all the vectors(rows) of the dataset in which each vector consists of the value of dependent features. The number of features is d i.e. X = (x1,x2,x2, xd).
- The Response/target vector (y) contains the value of class/group variable for each row of feature matrix.

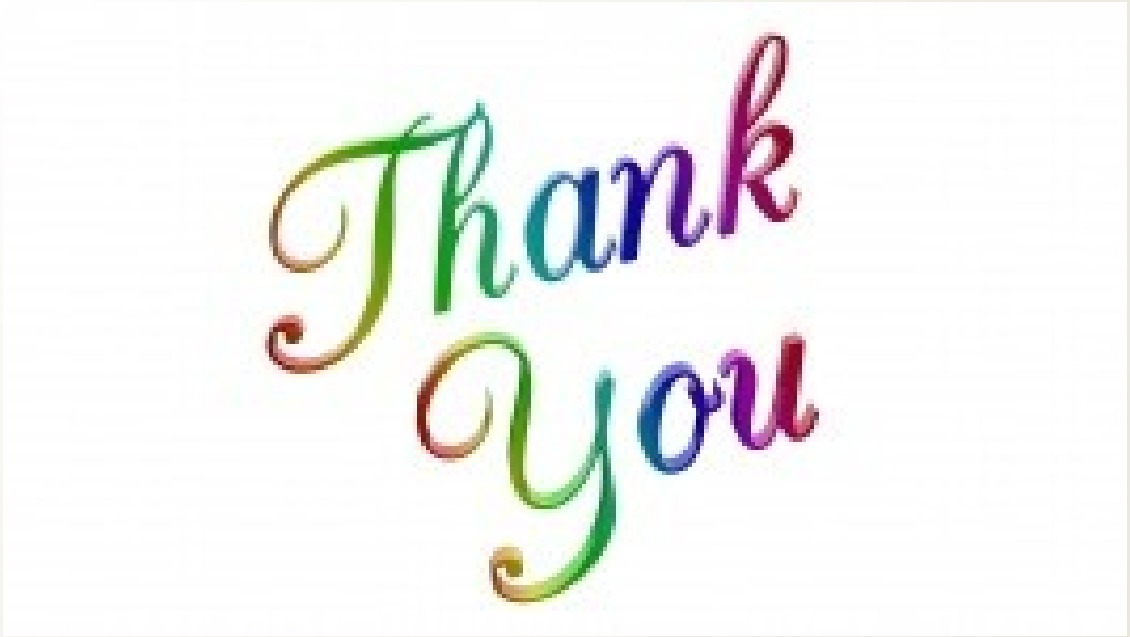
20AM3602

Basic of NBT

- https://colab.research.google.com/drive/1KexjOyo59XW8qCsWBg5XVecZYpRbrmO9?authuser=1

Implement naïve bayes theorem to classify the English text

https://colab.research.google.com/drive/1EFZ0nOOdqIo1MOfVYZVZZjO-EDpK2LPZ?authuser=1


Thank
You