

# Package ‘GoatBayes’

**Wersja:** 1.0.0

**Przedmiot:** Metody Odkrywania Wiedzy (MOW); WEiTI PW

**Opis:** Implementacja naiwnego klasyfikatora Bayesa w wersji binarnej oraz częstościowej

**Autorzy:** Piotr Chodyko, Damian Portasiński

---

trainB.....	1
trainM.....	2
predictB.....	2
predictM.....	3

---

---

trainB	<i>Naiwny klasyfikator Bayesa w wersji binarnej</i>
--------	-----------------------------------------------------

---

## Opis

Oblicza warunkowe prawdopodobieństwa występowania danych słów pod warunkiem danej klasy, zakładając niezależność słów od siebie.

## Użycie

```
bayes <- trainB(dataset, classes)
# bayes będzie składać się z dwóch pól:
# p_class - lista z prawdopodobieństwami wystąpienia klas w zbiorze
danych
# warunkowe prawdopodobieństwa wystąpienia słów
```

## Argumenty

- **dataset** - data frame zawierający binarne wartości odpowiadające występowaniu danej zmiennej w danym przykładzie, wymiary [n,m]
- **classes** - data frame o wymiarach [n,1], zawierający klasy przykładów w dataset

## Szczegóły

Implementacja klasyfikatora bayesa w wersji binarnej zakłada niezależność zmiennych oraz wygładzanie wartości prawdopodobieństwa w celu zabezpieczenia się przed zerowym prawdopodobieństwem wystąpienia zmiennej pod warunkiem danej klasy (ponieważ zbiór danych zazwyczaj jest niewielkim podzbiorem pewnej dziedziny)

---

trainM      *Naiwny klasyfikator Bayesa w wersji częstościowej*

---

### Opis

Oblicza warunkowe prawdopodobieństwa występowania danych słów pod warunkiem danej klasy, zakładając niezależność słów od siebie.

### Użycie

```
bayes <- trainM(dataset, classes)
# bayes będzie składać się z dwóch pól:
# p_class - lista z prawdopodobieństwami wystąpienia klas w zbiorze
danych
# probabilities - warunkowe prawdopodobieństwa wystąpienia słów
```

### Argumenty

- dataset - data frame zawierający m kolumn odpowiadającym wszystkim słowom występującym w przykładach, oraz n wierszy odpowiadających ilości przykładów; w data frame na pozycji [p,q] - ile razy słowo q wystąpiło w przykładzie p
- classes - data frame o wymiarach [n,1], zawierający klasy przykładów w dataset

### Szczegóły

Implementacja klasyfikatora bayesa w wersji częstościowej zakłada niezależność zmiennych oraz wygładzanie wartości prawdopodobieństwa w celu zabezpieczenia się przed zerowym prawdopodobieństwem wystąpienia zmiennej pod warunkiem danej klasy (ponieważ zbiór danych zazwyczaj jest niewielkim podzbiorem pewnej dziedziny)

---

predictB      *Metoda predykcji dla naiwnego klasyfikatora Bayesa w wersji binarnej*

---

### Opis

Przewiduje wartości przynależności do klas na podstawie modelu wytrenowanego metodą trainB

### Użycie

```
bayes <- trainB(dataset, classes)
results <- predictB(bayes, dataset) #przewidywanie całego zbioru
trenującego
#results jest dataframe'em o wymiarach [ilosc_wierszy(dataset), 2]
gdzie w komórce [i,j] określono logarytm prawdopodobieństwa, że
przykład i należy do klasy j
```

### Argumenty

- classifier- obiekt zwracany przez funkcję trainB
- dataset - data frame przykładów do predykcji
- type - jeśli podany argument "raw" to zwracane są wartości prawdopodobieństw dla każdej z klas, jeśli nie to zwracana jest tylko przewidziana klasa

### Szczegóły

Funkcja wylicza sumę logarytmów prawdopodobieństw warunkowych występowania danych słów w przykładzie (pod warunkiem danej klasy).

---

predictM     *Metoda predykcji dla naiwnego klasyfikatora Bayesa w wersji częstościowej*

---

## Opis

Przewiduje wartości przynależności do klas na podstawie modelu wytrenowanego metodą trainM

## Użycie

```
bayes <- trainM(dataset, classes)
results <- predictM(bayes, dataset) #przewidywanie całego zbioru
trenującego
#results jest dataframe'em o wymiarach [ilosc_wierszy(dataset),
liczba_klas] gdzie w komórce [i,j] określono wartość proporcjonalną do
prawdopodobieństwa że przykład i należy do klasy j
```

## Argumenty

- classifier- obiekt zwracany przez funkcję trainM
- dataset - data frame przykładów do predykcji
- type - jeśli podany argument "raw" to zwracane są wartości prawdopodobieństw dla każdej z klas, jeśli nie to zwracana jest tylko przewidziana klasa

## Szczegóły

Funkcja wylicza wartość proporcjonalną do wartości prawdopodobieństwa, że dany przykład należy do danej klasy - korzystając z zależności

$$P(c = d \mid a_i = v_i, \dots, a_n = v_n) \sim P(c = d) * \prod_{i=1}^n P(a_i \mid c = d)^{v_i}$$