

## Лабораторна робота №6

### Наївний Байєс в Python

**Мета роботи:** набути навичок працювати з даними і опонувати роботу у Python з використанням теореми Байєса.

#### Література

Supervised learning - [https://scikit-learn.org/stable/user\\_guide.html](https://scikit-learn.org/stable/user_guide.html)

Naive Bayes Tutorial: Naive Bayes Classifier in Python - <https://dzone.com/articles/naive-bayes-tutorial-naive-bayes-classifier-in-pyt>

Наивный байесовский классификатор - <http://datascientist.one/naive-bayes/>

Хід роботи:

**Завдання 1.** Ретельно опрацювати теоретичні відомості:

- теорему Байєса;
- які типи наївного байєсівського класифікатора є;
- де використовується Наївний Байєс.

**Завдання 2.** Ретельно розібрати приклад: прогнозування з використанням теореми Байєса.

В ході розбору прикладу в ньому було виявлено декілька неточностей в обчисленнях та значеннях.

Likelihood Table		Play		
		Yes	No	
Outlook	Sunny	3/10	2/4	5/14
	Overcast	4/10	0/4	4/14
	Rainy	3/10	2/4	5/14
		10/14	4/14	

$P(x|c) = P(\text{Sunny} | \text{Yes}) = 3/10 = 0.3$

$P(x) = P(\text{Sunny}) = 5/14 = 0.36$

$P(c) = P(\text{Yes}) = 10/14 = 0.71$

в таблиці тотал єс та тотал ні було вказано 10 та 4, в прикладі ж ці значення були 9 та 5, я виправив власну таблицю і вона має ось такий вигляд:

```
data = {
  "Outlook": {
    "rainy": {"Yes": 2, "No": 3},
    "sunny": {"Yes": 3, "No": 2},
    "overcast": {"Yes": 4, "No": 0}
  },
  "Humidity": {
    "high": {"Yes": 3, "No": 4},
    "normal": {"Yes": 6, "No": 1}
  },
  "Wind": {
    "weak": {"Yes": 6, "No": 2},
    "strong": {"Yes": 3, "No": 3}
  }
}
```

Тепер значення порівнялись а таблиця стала навіть логічнішою (при дощі ні має більше значення).

Крім того, в прикладі вказаному в ЛР:

Ймовірність «Yes» в цей день =  $P(\text{Outlook} = \text{Rain}|\text{Yes}) * P(\text{Humidity} = \text{High}|\text{Yes}) * P(\text{Wind} = \text{Weak}|\text{Yes}) * P(\text{Yes}) = 2/9 * 3/9 * 6/9 * 9/14 = 0,0199$

Ймовірність Так в цей день порахована не вірно:

$$((((((2 / 9) * 3) / 9) * 6) / 9) * 9) / 14 =$$

$$0.03174603174$$

Як і ось тут:

Ймовірність негативної відповіді «No» в цей день =  $P(\text{Outlook} = \text{Rain}|\text{No}) * P(\text{Humidity} = \text{High}|\text{No}) * P(\text{Wind} = \text{Weak}|\text{No}) * P(\text{No}) = 2/5 * 4/5 * 2/5 * 5/14 = 0,0166$

Google

2/5 \* 4/5 \* 2/5 \* 5/14

Все Картинки Покупки Видео Новости Веб-версия Книги : Ещё Инструменты

0.04571428571

Плюс до того, мої значення будуть відрізнятись, бо за таблицею в мене при дощі коефіцієнт на Ні  $\frac{3}{5}$  а не  $\frac{2}{5}$  тому:

0.06857142857

Тому ймовірність на Так в цей день:

$$\frac{(((((((2/9) * 3) / 9) * 6) / 9) * 9) / 14) / (((((((2/9) * 3) / 9) * 6) / 9) * 9) / 14) + (((((((3/5) * 4) / 5) * 2) / 5) * 5) / 14)}{(((((((2/9) * 3) / 9) * 6) / 9) * 9) / 14) + (((((((3/5) * 4) / 5) * 2) / 5) * 5) / 14)}$$

0.3164556962

Та на Ні:

$$\frac{((((((((3/5) * 4) / 5) * 2) / 5) * 5) / 14) / (((((((2/9) * 3) / 9) * 6) / 9) * 9) / 14) + (((((((3/5) * 4) / 5) * 2) / 5) * 5) / 14)}{((((((((2/9) * 3) / 9) * 6) / 9) * 9) / 14) + (((((((3/5) * 4) / 5) * 2) / 5) * 5) / 14)}$$

0.68354430379

Перейдемо до 3 завдання:

**Завдання 3. Використовую данні з пункту 2 визначити відбудеться матч при наступних погодних умовах чи ні: Розрахунки провести з використанням Python.**

Ще в ході виконання завдання два було реалізовано програмний код на мові Python, який обчислює це все самостійно:

Лістинг:

```
data = {
    "Outlook": {
        "rainy": {"Yes": 2, "No": 3},
        "sunny": {"Yes": 3, "No": 2},
        "overcast": {"Yes": 4, "No": 0}
    },
    "Humidity": {
        "high": {"Yes": 3, "No": 4},
        "normal": {"Yes": 6, "No": 1}
    },
    "Wind": {
        "weak": {"Yes": 6, "No": 2},
        "strong": {"Yes": 3, "No": 3}
    }
}

total_yes = 9
```

```
total_no = 5
total = total_yes + total_no

p_yes = total_yes / total
p_no = total_no / total

conditions = {
    "Outlook": input("Enter outlook (sunny, overcast, rainy): ").lower(),
    "Humidity": input("Enter humidity level (high, normal): ").lower(),
    "Wind": input("Enter wind strength (weak, strong): ").lower()
}

p_rain_yes = data["Outlook"][conditions["Outlook"]]["Yes"] / total_yes
p_rain_no = data["Outlook"][conditions["Outlook"]]["No"] / total_no

p_humidity_yes = data["Humidity"][conditions["Humidity"]]["Yes"] / total_yes
p_humidity_no = data["Humidity"][conditions["Humidity"]]["No"] / total_no

p_wind_yes = data["Wind"][conditions["Wind"]]["Yes"] / total_yes
p_wind_no = data["Wind"][conditions["Wind"]]["No"] / total_no

p_yes_given_conditions = p_rain_yes * p_humidity_yes * p_wind_yes * p_yes
p_no_given_conditions = p_rain_no * p_humidity_no * p_wind_no * p_no

total_probability = p_yes_given_conditions + p_no_given_conditions
p_yes_final = p_yes_given_conditions / total_probability
p_no_final = p_no_given_conditions / total_probability

print(f"Ймовірність, що матч відбудеться (Yes): {p_yes_final:.2f}")
print(f"Ймовірність, що матч не відбудеться (No): {p_no_final:.2f}")
```

Також звіримо його обчислення з тим що в нас вийшло в минулому завданні:

Тому ймовірність на Так в цей день:

$$\frac{(((((((2/9) * 3) / 9) * 6) / 9) * 9) / 14) / (((((((((2/9) * 3) / 9) * 6) / 9) * 9) / 14) + (((((((((3/5) * 4) / 5) * 2) / 5) * 5) / 14) / (((((((((2/9) * 3) / 9) * 6) / 9) * 9) / 14) + (((((((((3/5) * 4) / 5) * 2) / 5) * 5) / 14))}}}}}}}}}}{0.3164556962}$$

Та на Ні:

$$\frac{(((((((3/5) * 4) / 5) * 2) / 5) * 5) / 14) / (((((((((2/9) * 3) / 9) * 6) / 9) * 9) / 14) + (((((((((3/5) * 4) / 5) * 2) / 5) * 5) / 14))}}}}}}}}{0.68354430379}$$

```
root@dmytro-X550CL:/home/alhim/Завантаження/ШІ/lab6# /bin/python3 /home/alhim/Завантаження/ШІ/lab6/lab6.py
Enter outlook (sunny, overcast, rainy): Rainy
Enter humidity level (high, normal): High
Enter wind strength (weak, strong): weak
Ймовірність, що матч відбудеться (Yes): 0.32
Ймовірність, що матч не відбудеться (No): 0.68
root@dmytro-X550CL:/home/alhim/Завантаження/ШІ/lab6#
```

Загалом програма дозволяє дослідити всі варіанти, але перейдемо саме до нашого та дослідимо інші:

Мій варіант 27, їх всього 15 тому виконуємо 12:

2, 7, 12	Outlook = Overcast Humidity = High Wind = Strong	Перспектива = Похмуро Вологість = Висока Вітер = Сильний
----------	--	--

```
Enter outlook (sunny, overcast, rainy): overcast
Enter humidity level (high, normal): high
Enter wind strength (weak, strong): strong
Ймовірність, що матч відбудеться (Yes): 1.00
Ймовірність, що матч не відбудеться (No): 0.00
```

Також виконаємо решту:

Варіант	Умова	
1, 6, 11	Outlook = Overcast Humidity = High Wind = Weak	Перспектива = Похмуро Вологість = Висока Вітер = Слабкий
3, 8, 13	Outlook = Sunny Humidity = High Wind = Weak	Перспектива = Сонячно Вологість = Висока Вітер = Слабкий
4, 9, 14	Outlook = Sunny Humidity = Normal Wind = Strong	Перспектива = Сонячно Вологість = Нормальна Вітер = Сильний
5, 10, 15	Outlook = Rain Humidity = High Wind = Strong	Outlook = Дощ Вологість = Висока Вітер = Сильний

```

root@dmytro-X550CL:/home/alhim/Завантаження/ШІ/lab6# /bin/python3 /home/alhim/Завантаження/ШІ/lab6/lab6.py
Enter outlook (sunny, overcast, rainy): overcast
Enter humidity level (high, normal): high
Enter wind strength (weak, strong): weak
Ймовірність, що матч відбудеться (Yes): 1.00
Ймовірність, що матч не відбудеться (No): 0.00
root@dmytro-X550CL:/home/alhim/Завантаження/ШІ/lab6# /bin/python3 /home/alhim/Завантаження/ШІ/lab6/lab6.py
Enter outlook (sunny, overcast, rainy): Sunny
Enter humidity level (high, normal): high
Enter wind strength (weak, strong): weak
Ймовірність, що матч відбудеться (Yes): 0.51
Ймовірність, що матч не відбудеться (No): 0.49
root@dmytro-X550CL:/home/alhim/Завантаження/ШІ/lab6# /bin/python3 /home/alhim/Завантаження/ШІ/lab6/lab6.py
Enter outlook (sunny, overcast, rainy): sunny
Enter humidity level (high, normal): normal
Enter wind strength (weak, strong): strong
Ймовірність, що матч відбудеться (Yes): 0.74
Ймовірність, що матч не відбудеться (No): 0.26

```

```

root@dmytro-X550CL:/home/alhim/Завантаження/ШІ/lab6# /bin/python3 /home/alhim/Завантаження/ШІ/lab6/lab6.py
Enter outlook (sunny, overcast, rainy): rainy
Enter humidity level (high, normal): high
Enter wind strength (weak, strong): strong
Ймовірність, що матч відбудеться (Yes): 0.13
Ймовірність, що матч не відбудеться (No): 0.87

```

**Завдання 4.** Застосуєте методи байєсівського аналізу до набору даних про ціни на квитки на іспанські високошвидкісні залізниці.

– Вхідні дані: [https://raw.githubusercontent.com/susanli2016/Machine-Learning-with-Python/master/data/renfe\\_small.csv](https://raw.githubusercontent.com/susanli2016/Machine-Learning-with-Python/master/data/renfe_small.csv)

Лістинг:

```

import pandas as pd
from sklearn.preprocessing import LabelEncoder
from sklearn.model_selection import train_test_split
from sklearn.linear_model import BayesianRidge
from sklearn.metrics import mean_squared_error

data = pd.read_csv('data.txt')

```

```

data = data.dropna(subset=['price'])

data['origin_enc'] = LabelEncoder().fit_transform(data['origin'])
data['destination_enc'] =
LabelEncoder().fit_transform(data['destination'])
data['train_type_enc'] =
LabelEncoder().fit_transform(data['train_type'])
data['train_class_enc'] =
LabelEncoder().fit_transform(data['train_class'])
data['fare_enc'] = LabelEncoder().fit_transform(data['fare'])

X = data[['origin_enc', 'destination_enc', 'train_type_enc',
'train_class_enc', 'fare_enc']]
y = data['price']

X_train, X_test, y_train, y_test = train_test_split(X, y,
test_size=0.2, random_state=42)

model = BayesianRidge()
model.fit(X_train, y_train)

y_pred = model.predict(X_test)
mse = mean_squared_error(y_test, y_pred)
print(f"Середньоквадратична похибка на тестовому наборі: {mse}")

def predict_price(origin, destination, train_type, train_class, fare):
    origin_enc =
LabelEncoder().fit(data['origin']).transform([origin])[0]
    destination_enc =
LabelEncoder().fit(data['destination']).transform([destination])[0]
    train_type_enc =
LabelEncoder().fit(data['train_type']).transform([train_type])[0]
    train_class_enc =
LabelEncoder().fit(data['train_class']).transform([train_class])[0]
    fare_enc = LabelEncoder().fit(data['fare']).transform([fare])[0]

    input_data = pd.DataFrame([origin_enc, destination_enc,
train_type_enc, train_class_enc, fare_enc]],
                                columns=['origin_enc', 'destination_enc',
'train_type_enc', 'train_class_enc', 'fare_enc'])

    predicted_price = model.predict(input_data)[0]
    return predicted_price

```

```
origin = input("Enter the origin station: ")
destination = input("Enter the destination station: ")
train_type = input("Enter the train type (e.g., AVE): ")
train_class = input("Enter the train class (e.g., Turista): ")
fare = input("Enter the fare type (e.g., Promo): ")

predicted_price = predict_price(origin, destination, train_type,
train_class, fare)
print(f"The predicted price is: {predicted_price}")
```

```
Середньоквадратична похибка на тестовому наборі: 218.33846291905388
Enter the origin station: MADRID
Enter the destination station: VALENCIA
Enter the train type (e.g., AVE): AVE
Enter the train class (e.g., Turista): Turista
Enter the fare type (e.g., Promo): Promo
The predicted price is: 44.10358351408544
```

GitHub: [https://github.com/Alhim616/AI\\_Labs\\_Yanushevykh](https://github.com/Alhim616/AI_Labs_Yanushevykh)