Лабораторна робота №6

Наївний Байєс в Python

Виконав: ІПЗ-21-3 Осипчук Антон Олексійович

Github: https://github.com/AntonOsypchuk1/ai lab/tree/main/lab6

Завдання. Використовуючи дані з пункту 2 визначити відбудеться матч при наступних погодних умовах чи ні:

Варіант	Умова	
14	Outlook = Sunny Humidity = Normal Wind = Strong	Перспектива = Сонячно Вологість = Нормальна Вітер = Сильний

Лістинг програми:

```
# Вхідні дані
data = {
    "Outlook": {"Sunny": {"Yes": 3/10, "No": 2/4}},
    "Humidity": {"Normal": {"Yes": 6/9, "No": 1/5}},
    "Wind": {"Strong": {"Yes": 6/9, "No": 2/5}}
# Ймовірності для гри "Yes"
p_yes = (
    data["Outlook"]["Sunny"]["Yes"] *
    data["Humidity"]["Normal"]["Yes"] *
    data["Wind"]["Strong"]["Yes"]
# Ймовірності для гри "No"
p_no = (
    data["Outlook"]["Sunny"]["No"] *
    data["Humidity"]["Normal"]["No"] *
    data["Wind"]["Strong"]["No"]
# Нормалізація
p_yes_normalized = p_yes / (p_yes + p_no)
p_no_normalized = p_no / (p_yes + p_no)
print(f"Ймовірність гри 'Yes': {p yes normalized:.2f}")
```

```
print(f"Ймовірність гри 'No': {p_no_normalized:.2f}")

if p_yes_normalized > p_no_normalized:
    print("Прогноз: Гра відбудеться.")

else:
    print("Прогноз: Гра не відбудеться.")
```

Результат:

```
    antonosypchuk@MacBook-Air-Anton lab6 % python3 task3.py
Ймовірність гри 'Yes': 0.77
Ймовірність гри 'No': 0.23
Прогноз: Гра відбудеться.
```

Завдання. Застосувати методи байєсівського аналізу до набору даних про ціни на квитки на іспанські високошвидкісні залізниці.

Лістинг програми:

```
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.naive bayes import GaussianNB
from sklearn.preprocessing import LabelEncoder, KBinsDiscretizer
from sklearn.metrics import accuracy_score
url = "https://raw.githubusercontent.com/susanli2016/Machine-Learning-
with-Python/master/data/renfe_small.csv"
data = pd.read csv(url)
data = data.dropna()
le = LabelEncoder()
categorical_columns = ['origin', 'destination', 'train_type', 'fare']
for col in categorical_columns:
    data[col] = le.fit_transform(data[col])
discretizer = KBinsDiscretizer(n_bins=3, encode='ordinal',
strategy='uniform')
data['price_category'] = discretizer.fit_transform(data[['price']])
X = data[['origin', 'destination', 'train_type', 'fare']]
y = data['price_category']
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3,
random state=42)
```

```
model = GaussianNB()
model.fit(X_train, y_train)

y_pred = model.predict(X_test)

accuracy = accuracy_score(y_test, y_pred)
print(f"Точність моделі: {accuracy:.2f}")
```

Результат:

(venv) antonosypchuk@MacBook-Air-Anton lab6 % python3 task4.pyТочність моделі: 0.84