

ЛАБОРАТОРНА РОБОТА № 6

Наївний Байєс в Python

Мета: використовуючи спеціалізовані бібліотеки та мову програмування Python дослідити методи ансамблів у машинному навчанні.

Варіант 15

Хід роботи:

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

```
dataset = [
    {"Weather": "Sunny", "Moisture": "High", "Breeze": "Weak", "Game": "No"},
    {"Weather": "Sunny", "Moisture": "High", "Breeze": "Strong", "Game": "No"},
    {"Weather": "Overcast", "Moisture": "High", "Breeze": "Weak", "Game": "Yes"},
    {"Weather": "Rain", "Moisture": "High", "Breeze": "Weak", "Game": "Yes"},
    {"Weather": "Rain", "Moisture": "Normal", "Breeze": "Weak", "Game": "Yes"},
    {"Weather": "Rain", "Moisture": "Normal", "Breeze": "Strong", "Game": "No"},
    {"Weather": "Overcast", "Moisture": "Normal", "Breeze": "Strong", "Game": "Yes"},
    {"Weather": "Sunny", "Moisture": "Normal", "Breeze": "Weak", "Game": "Yes"},
    {"Weather": "Sunny", "Moisture": "Normal", "Breeze": "Strong", "Game": "Yes"},
    {"Weather": "Rain", "Moisture": "Normal", "Breeze": "Weak", "Game": "Yes"},
    {"Weather": "Sunny", "Moisture": "High", "Breeze": "Strong", "Game": "No"},
    {"Weather": "Overcast", "Moisture": "High", "Breeze": "Strong", "Game": "Yes"},
    {"Weather": "Overcast", "Moisture": "Normal", "Breeze": "Weak", "Game": "Yes"},
    {"Weather": "Rain", "Moisture": "High", "Breeze": "Strong", "Game": "No"},
]

yes_count = sum(1 for d in dataset if d["Game"] == "Yes")
no_count = sum(1 for d in dataset if d["Game"] == "No")
total_count = len(dataset)

prob_yes = yes_count / total_count
prob_no = no_count / total_count

def find_probability(feature, value, outcome):
    filtered = [d for d in dataset if d[feature] == value and d["Game"] == outcome]
    return len(filtered) / (yes_count if outcome == "Yes" else no_count)

weather_yes = find_probability("Weather", "Rain", "Yes")
moisture_yes = find_probability("Moisture", "High", "Yes")
breeze_yes = find_probability("Breeze", "Strong", "Yes")

weather_no = find_probability("Weather", "Rain", "No")
moisture_no = find_probability("Moisture", "High", "No")
```

Зм.	Роботу виконав	Перевірив	Голенко М.Ю.					1	19
Роботу виконав		Керівник						ФІКТ Гр. ІПЗ-21-3	
		Н. контр.							
		Зав. каф.							

Звіт з
лабораторної роботи

```

breeze_no = find_probability("Breeze", "Strong", "No")

yes_data_prob = weather_yes * moisture_yes * breeze_yes * prob_yes
no_data_prob = weather_no * moisture_no * breeze_no * prob_no

final_prob = yes_data_prob + no_data_prob
result_yes = yes_data_prob / final_prob
result_no = no_data_prob / final_prob

print(f"P(Yes) = {result_yes:.2f}")
print(f"P(No) = {result_no:.2f}")

if result_yes > result_no:
    print("Гра відбудеться.")
else:
    print("Гра не відбудеться.")

```

```

LR_6_task1 : [Errno 2] No such file or directory
sh-3.2# python3 LR_6_task1.py
P(Yes) = 0.21
P(No) = 0.79
Гра не відбудеться.
sh-3.2# █

```

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

```

import pandas as pd
from collections import defaultdict
import ssl

ssl._create_default_https_context = ssl._create_unverified_context

url_data = "https://raw.githubusercontent.com/susanli2016/Machine-Learning-with-Python/master/data/renfe_small.csv"
dataset = pd.read_csv(url_data)

dataset = dataset[["price", "train_type", "origin", "destination",
"train_class"]].dropna()

price_ranges = [
    dataset["price"].min(),
    dataset["price"].quantile(0.33),
    dataset["price"].quantile(0.66),
    dataset["price"].max()
]

dataset["price_segment"] = pd.cut(dataset["price"], bins=price_ranges, labels=["low",
"medium", "high"])

```

		Паламарчук В.В.			ДУ «Житомирська політехніка».24.121.15.000 – Лр6	Арк.
		Голенко М.Ю.				2
Змн.	Арк.	№ докум.	Підпис	Дата		

```

feature_distribution = defaultdict(lambda: defaultdict(int))
category_totals = defaultdict(int)

for _, entry in dataset.iterrows():
    price_segment = entry["price_segment"]
    category_totals[price_segment] += 1
    for attribute in ["train_type", "origin", "destination", "train_class"]:
        feature_distribution[attribute][(entry[attribute], price_segment)] += 1

def probability_given_category(attribute, value, category):
    count = feature_distribution[attribute].get((value, category), 0)
    total = category_totals[category]
    return count / total if total else 0

def compute_category_probability(category, conditions):
    prior_prob = category_totals[category] / len(dataset)
    likelihood = 1
    for attribute, value in conditions.items():
        likelihood *= probability_given_category(attribute, value, category)
    return prior_prob * likelihood

ticket_params = {
    "train_type": "AVE",
    "origin": "MADRID",
    "destination": "SEVILLA",
    "train_class": "Turista"
}

posterior_results = {}
for segment in category_totals.keys():
    posterior_results[segment] = compute_category_probability(segment, ticket_params)

total_posterior = sum(posterior_results.values())
normalized_results = {seg: prob / total_posterior for seg, prob in
    posterior_results.items()}

print("Ймовірності для кожної категорії ціни квитка -->")
for segment, prob in normalized_results.items():
    print(f"{str(segment).capitalize()}: {prob:.2f}")

```

```

sh-3.2# python3 LR_6_task2.py
Ймовірності для кожної категорії ціни квитка -->
Low: 0.13
High: 0.37
Medium: 0.50
Nan: 0.00
sh-3.2# █

```

Посилання на Github: https://github.com/vladpalamar/Lab6_Ai.git

		Паламарчук В.В.			ДУ «Житомирська політехніка».24.121.15.000 – Лр6	Арк.
		Голенко М.Ю.				3
Змн.	Арк.	№ докум.	Підпис	Дата		

Висновки: використав спеціалізовані бібліотеки та мову програмування Python, та дослідив методи ансамблів у машинному навчанні.

		Паламарчук В.В.			ДУ «Житомирська політехніка».24.121.15.000 – Лр6	Арк.
		Голенко М.Ю.				4
Змн.	Арк.	№ докум.	Підпис	Дата		