

Лабораторна робота № 6

Наївний Байєс в Python

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

Хід роботи

Завдання 6.1. Ретельно опрацювати теоретичні відомості:

Завдання 6.2. Ретельно розібрати приклад: прогнозування з використанням теореми Байєса.

Завдання 6.3. Використовую данні з пункту 2 визначити відбудеться матч при наступних погодних умовах чи ні: Розрахунки провести з використанням Python.

Таблиця. 6.1. Вхідні умови

Варіант	Умова	
4,9,14	Outlook = Sunny Humidity = Normal Wind = Strong	Перспектива = Сонячно Вологість = Нормальна Вітер = Сильний

- Загальні ймовірності (Priors):
 - $P(\text{Yes}) = 9/14$
 - $P(\text{No}) = 5/14$
- Ймовірності для «Yes» (з відповідних таблиць):
 - $P(\text{Outlook} = \text{Sunny}|\text{Yes}) = 2/9$
 - $P(\text{Humidity} = \text{Normal}|\text{Yes}) = 6/9$
 - $P(\text{Wind} = \text{Strong}|\text{Yes}) = 3/9$
- Ймовірності для «No» (з відповідних таблиць):
 - $P(\text{Outlook} = \text{Sunny}|\text{No}) = 3/5$
 - $P(\text{Humidity} = \text{Normal}|\text{No}) = 1/5$
 - $P(\text{Wind} = \text{Strong}|\text{No}) = 3/5$

$$P(\text{Yes}|\text{Sunny, Normal, Strong}) = P(\text{Sunny}|\text{Yes}) * P(\text{Normal}|\text{Yes}) * P(\text{Strong}|\text{Yes}) * P(\text{Yes}) = 2/9 * 6/9 * 3/9 * 9/14 = 3/70 \approx 0.03174$$

					ДУ «Житомирська політехніка».25.121.19.000–Лр6							
Змн.	Арк.	№ докум.	Підпис	Дата								
Розроб.		Леус В.О.			Звіт з лабораторної роботи				Лім.	Арк.	Аркушів	
Перевір.		Маєвський О.В.									1	14
Керівник									ФІКТ Гр. ІПЗ-22-3			
Н. контр.												
Зав. каф.												

$$P(\text{No}|\text{Sunny, Normal, Strong}) = P(\text{Sunny}|\text{No}) * P(\text{Normal}|\text{No}) * P(\text{Strong}|\text{No}) *$$

$$P(\text{No}) = 3/5 * 1/5 * 3/5 * 5/14 = 3/140 \approx 0.02571$$

$$P(\text{Yes}) = 0.03174 / (0.03174 + 0.02571) = 0.5525$$

$$P(\text{No}) = 0.02571 / (0.021429 + 0.03174) = 0.4475$$

Ймовірність того, що матч відбудеться (55.25%), вища за ймовірність того, що він не відбудеться (44.75%).

Лістинг LR_6_task_3.py:

```
p_sunny_yes = 2/9
p_normal_yes = 6/9
p_strong_yes = 3/9
p_sunny_no = 3/5
p_normal_no = 1/5
p_strong_no = 3/5
p_yes = 9/14
p_no = 5/14
score_yes = p_sunny_yes * p_normal_yes * p_strong_yes * p_yes
score_no = p_sunny_no * p_normal_no * p_strong_no * p_no
print(f"Показник для 'Yes': {score_yes:.5f}")
print(f"Показник для 'No': {score_no:.5f}")
total_score = score_yes + score_no
final_p_yes = score_yes / total_score
final_p_no = score_no / total_score
print("\n--- Результати прогнозу ---")
print(f"Ймовірність, що матч відбудеться ('Yes'): {final_p_yes:.1%}")
print(f"Ймовірність, що матч не відбудеться ('No'): {final_p_no:.1%}")
if final_p_yes > final_p_no:
    print("\nВисновок: Модель прогнозує, що матч відбудеться. 🟢")
else:
    print("\nВисновок: Модель прогнозує, що матч НЕ відбудеться. 🟡")
```

```
Показник для 'Yes': 0.03175
Показник для 'No': 0.02571

--- Результати прогнозу ---
Ймовірність, що матч відбудеться ('Yes'): 55.2%
Ймовірність, що матч не відбудеться ('No'): 44.8%

Висновок: Модель прогнозує, що матч відбудеться. 🟢
```

Рис.6.1. Результат програми

Завдання 6.4. Застосуєте методи байєсівського аналізу до набору даних про ціни на квитки на іспанські високошвидкісні залізниці.

		Леус В.О.			ДУ «Житомирська політехніка».25.121.19.000 – Лр6	Арк.
		Масєвський О.В.				
Змн.	Арк.	№ докум.	Підпис	Дата		2

Лістинг LR_6_task_4.py:

```
import pandas as pd
import pymc as pm
import arviz as az
import matplotlib.pyplot as plt
import seaborn as sns
import numpy as np

df = pd.read_csv('renfe_small.csv')
df_clean = df.dropna(subset=['price', 'train_type', 'fare']).copy()
df_clean['train_type_code'] = pd.Categorical(df_clean['train_type']).codes
df_clean['fare_code'] = pd.Categorical(df_clean['fare']).codes
train_type_labels = pd.Categorical(df_clean['train_type']).categories
fare_labels = pd.Categorical(df_clean['fare']).categories
print("Дані успішно підготовлені.")
print(f"Категорії типів поїздів: {list(train_type_labels)}")
print(f"Категорії тарифів: {list(fare_labels)}")
plt.figure(figsize=(14, 6))
plt.subplot(1, 2, 1)
sns.histplot(df_clean['price'], kde=True)
plt.title('Розподіл цін на квитки')
plt.xlabel('Ціна (€)')
plt.ylabel('Частота')
plt.subplot(1, 2, 2)
sns.boxplot(x='train_type', y='price', data=df_clean)
plt.title('Ціна залежно від типу поїзда')
plt.xlabel('Тип поїзда')
plt.ylabel('Ціна (€)')
plt.xticks(rotation=45)
plt.tight_layout()
plt.show()

with pm.Model() as renfe_model:
    beta0 = pm.Normal('beta0', mu=0, sigma=10)
    beta_train_type = pm.Normal('beta_train_type', mu=0, sigma=5, shape=len(train_type_labels))
    beta_fare = pm.Normal('beta_fare', mu=0, sigma=5, shape=len(fare_labels))
    alpha = pm.Exponential('alpha', 1.0)
    mu = pm.math.exp(beta0 + beta_train_type[df_clean['train_type_code']] + beta_fare[df_clean['fare_code']])
    beta = alpha / mu
    price = pm.Gamma('price', alpha=alpha, beta=beta, observed=df_clean['price'])

with renfe_model:
    idata = pm.sample(2000, tune=1000, cores=1)
    idata.extend(pm.sample_posterior_predictive(idata))
print("Модель успішно навчена.")
summary = az.summary(idata, var_names=['beta0', 'beta_train_type', 'beta_fare', 'alpha'], hdi_prob=0.94)
print("\nЗведення параметрів моделі:")
print(summary)
summary_train_type = az.summary(idata, var_names=['beta_train_type'], hdi_prob=0.94)
summary_train_type.index = [f"beta_train_type[{label}]" for label in train_type_labels]
print("\nЗведення коефіцієнтів типу поїзда:")
print(summary_train_type)
summary_fare = az.summary(idata, var_names=['beta_fare'], hdi_prob=0.94)
summary_fare.index = [f"beta_fare[{label}]" for label in fare_labels]
print("\nЗведення коефіцієнтів тарифу:")
print(summary_fare)
mean_coefs_train = idata.posterior['beta_train_type'].mean(dim=('chain', 'draw')).values
print(f"\nБазовий тип поїзда: {train_type_labels[0]}")
for i, label in enumerate(train_type_labels[1:], 1):
    effect = np.exp(mean_coefs_train[i] - mean_coefs_train[0])
    print(f"Поїзд типу '{label}' у {effect:.2f} разів дорожчий/дешевший, ніж '{train_type_labels[0]}")
az.plot_ppc(idata, num_pp_samples=100)
plt.title('Апостеріорна предиктивна перевірка')
plt.xlabel('Ціна (€)')
plt.show()
az.plot_forest(idata, var_names=['beta_train_type', 'beta_fare'], combined=True, hdi_prob=0.94)
plt.title('Апостеріорні розподіли для коефіцієнтів')
plt.show()
```

Для тесту було взято 2500 строк, щоб робота програми не тривала більше 3 годин.

		Леус В.О.			ДУ «Житомирська політехніка».25.121.19.000 – Лр6	Арк.
		Масевський О.В.				
Змн.	Арк.	№ докум.	Підпис	Дата		3

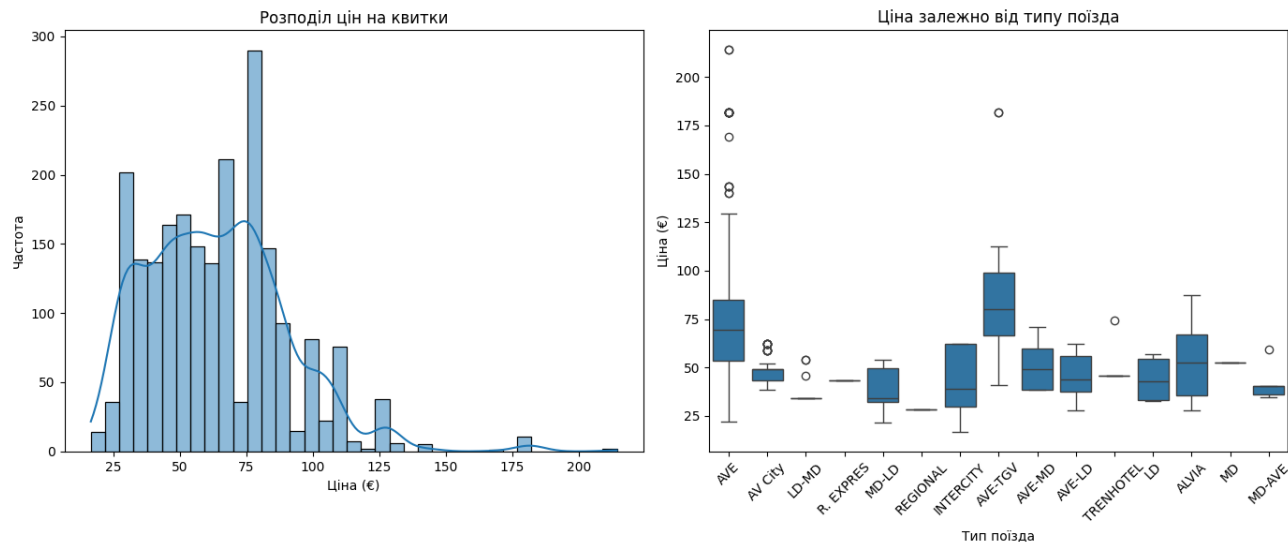


Рис.6.2. Гістограма цін на квитки + Діаграма розмаху цін за типами поїздів

```
Дані успішно підготовлені.
Категорії типів поїздів: ['ALVIA', 'AV City', 'AVE', 'AVE-LD', 'AVE-MD', 'AVE-TGV', 'INTERCITY', 'LD', 'LD-MD', 'MD', 'MD-AVE', 'MD-LD', 'R. EXPRES', 'REGIONAL', 'TRENHOTEL']
Категорії тарифів: ['Adulto ida', 'Flexible', 'Promo', 'Promo +']
Initializing NUTS using jitter+adapt diag...
Sequential sampling (2 chains in 1 job)
NUTS: [beta0, beta_train_type, beta_fare, alpha]
```

Рис.6.3. Вивід в консоль на цьому етапі

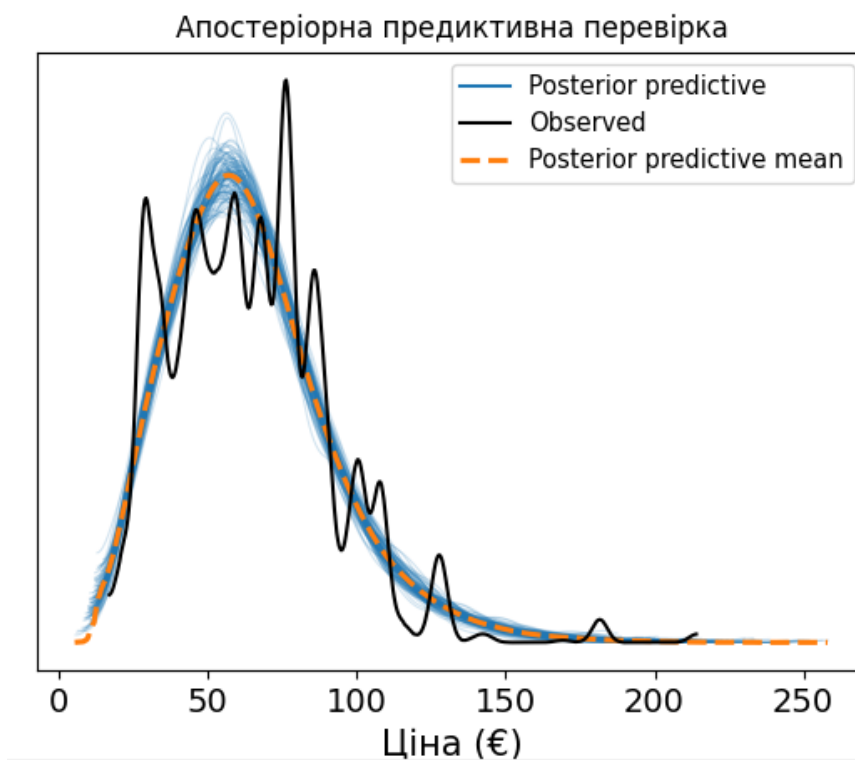


Рис.6.4. Графік апостеріорної предиктивної перевірки (PPC)

		Леус В.О.			ДУ «Житомирська політехніка».25.121.19.000 – Лр6	Арк.
		Масвський О.В.				
Змн.	Арк.	№ докум.	Підпис	Дата		4

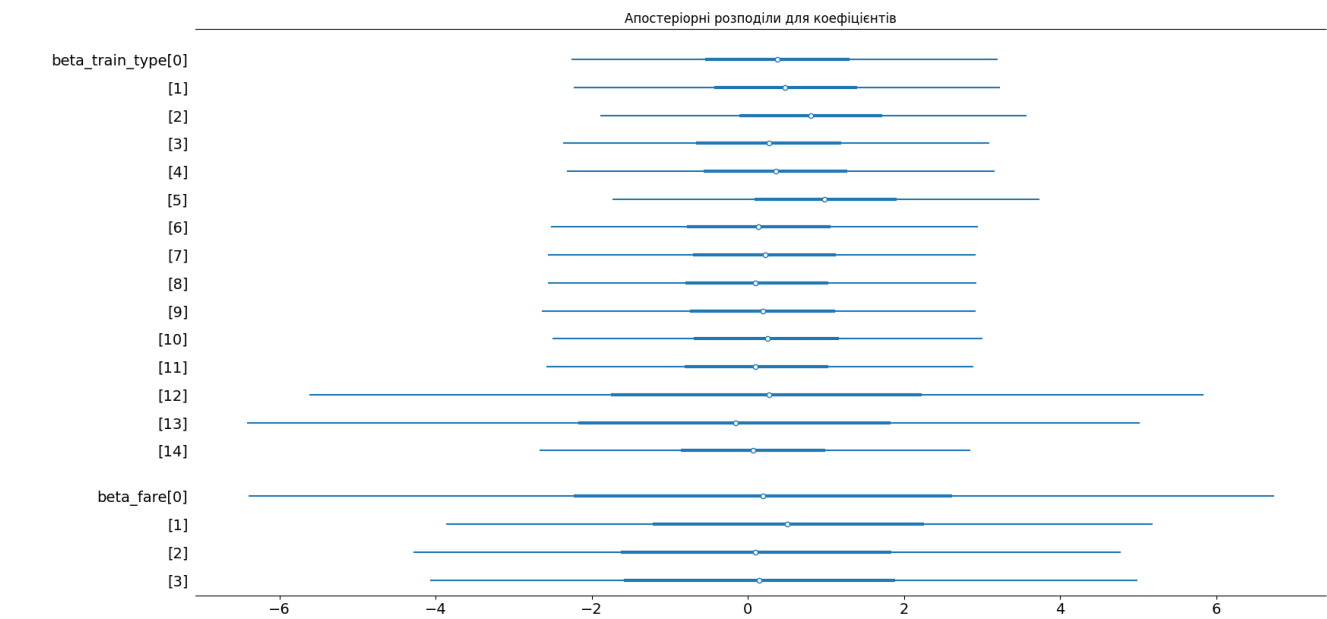


Рис.6.5. «Лісовий» графік коефіцієнтів моделі

Progress	Draws	Divergences	Step size	Grad evals	Sampling Speed	Elapsed	Remaining
	3000	0	0.00	1023	5.65 draws/s	0:08:51	0:00:00
	3000	0	0.00	127	3.03 draws/s	0:16:30	0:00:00

Sampling 2 chains for 1,000 tune and 2,000 draw iterations (2,000 + 4,000 draws total) took 991 seconds.
Chain 0 reached the maximum tree depth. Increase 'max_treedepth', increase 'target_accept' or reparameterize.
Chain 1 reached the maximum tree depth. Increase 'max_treedepth', increase 'target_accept' or reparameterize.
We recommend running at least 4 chains for robust computation of convergence diagnostics.
The rhat statistic is larger than 1.01 for some parameters. This indicates problems during sampling. See <https://arxiv.org/abs/1903.08008> for details.
The effective sample size per chain is smaller than 100 for some parameters. A higher number is needed for reliable rhat and ess computation. See <https://arxiv.org/abs/1903.08008> for details.
Sampling: [price]
Sampling ... 100% 0:00:00 / 0:00:00
Модель успішно навчена.

Зведення параметрів моделі:

	mean	sd	hdi_3%	hdi_97%	mcse_mean	mcse_sd	ess_bulk	ess_tail	r_hat
beta0	3.340	2.729	-1.813	8.268	0.186	0.080	209.0	600.0	1.02
beta_train_type[0]	0.284	1.491	-2.265	3.197	0.362	0.178	13.0	49.0	1.16
beta_train_type[1]	0.393	1.493	-2.229	3.230	0.361	0.179	13.0	50.0	1.16
beta_train_type[2]	0.713	1.492	-1.895	3.571	0.362	0.179	13.0	50.0	1.16
beta_train_type[3]	0.173	1.493	-2.365	3.095	0.362	0.178	13.0	50.0	1.16
beta_train_type[4]	0.270	1.495	-2.321	3.164	0.364	0.179	13.0	51.0	1.16
beta_train_type[5]	0.897	1.492	-1.738	3.730	0.360	0.179	13.0	50.0	1.16
beta_train_type[6]	0.049	1.493	-2.524	2.941	0.360	0.179	13.0	50.0	1.16
beta_train_type[7]	0.119	1.494	-2.558	2.918	0.362	0.179	13.0	48.0	1.16
beta_train_type[8]	0.026	1.493	-2.562	2.923	0.363	0.179	13.0	49.0	1.16
beta_train_type[9]	0.099	1.498	-2.644	2.916	0.360	0.176	13.0	49.0	1.16
beta_train_type[10]	0.144	1.495	-2.509	3.000	0.360	0.180	13.0	48.0	1.16
beta_train_type[11]	0.012	1.492	-2.584	2.883	0.363	0.178	13.0	49.0	1.16
beta_train_type[12]	0.250	3.046	-5.618	5.836	0.147	0.078	432.0	762.0	1.00
beta_train_type[13]	-0.173	3.046	-6.419	5.022	0.147	0.078	432.0	758.0	1.00
beta_train_type[14]	-0.025	1.493	-2.670	2.845	0.362	0.178	13.0	49.0	1.16
beta_fare[0]	0.178	3.523	-6.395	6.744	0.160	0.092	485.0	722.0	1.00
beta_fare[1]	0.536	2.488	-3.870	5.181	0.119	0.067	439.0	773.0	1.01
beta_fare[2]	0.120	2.488	-4.288	4.771	0.119	0.067	439.0	773.0	1.01
beta_fare[3]	0.178	2.488	-4.067	4.993	0.119	0.067	440.0	805.0	1.01
alpha	11.152	0.334	10.540	11.777	0.011	0.007	968.0	1335.0	1.00

Рис.6.6. Вивід в консоль

		Леус В.О.			ДУ «Житомирська політехніка».25.121.19.000 – Лр6	Арк.
		Масевський О.В.				5
Змн.	Арк.	№ докум.	Підпис	Дата		

```

Зведення коефіцієнтів типу поїзда:
      mean      sd  hdi_3%  hdi_97%  mcse_mean  mcse_sd  ess_bulk  ess_tail  r_hat
beta_train_type[ALVIA]    0.284  1.491  -2.265   3.197    0.362   0.178    13.0    49.0   1.16
beta_train_type[AV City]  0.393  1.493  -2.229   3.230    0.361   0.179    13.0    50.0   1.16
beta_train_type[AVE]      0.713  1.492  -1.895   3.571    0.362   0.179    13.0    50.0   1.16
beta_train_type[AVE-LD]   0.173  1.493  -2.365   3.095    0.362   0.178    13.0    50.0   1.16
beta_train_type[AVE-MD]   0.270  1.495  -2.321   3.164    0.364   0.179    13.0    51.0   1.16
beta_train_type[AVE-TGV]  0.897  1.492  -1.738   3.730    0.360   0.179    13.0    50.0   1.16
beta_train_type[INTERCITY] 0.049  1.493  -2.524   2.941    0.360   0.179    13.0    50.0   1.16
beta_train_type[LD]       0.119  1.494  -2.558   2.918    0.362   0.179    13.0    48.0   1.16
beta_train_type[LD-MD]    0.026  1.493  -2.562   2.923    0.363   0.179    13.0    49.0   1.16
beta_train_type[MD]       0.099  1.498  -2.644   2.916    0.360   0.176    13.0    49.0   1.16
beta_train_type[MD-AVE]   0.144  1.495  -2.509   3.000    0.360   0.180    13.0    48.0   1.16
beta_train_type[MD-LD]    0.012  1.492  -2.584   2.883    0.363   0.178    13.0    49.0   1.16
beta_train_type[R. EXPRES] 0.250  3.046  -5.618   5.836    0.147   0.078   432.0   762.0   1.00
beta_train_type[REGIONAL] -0.173  3.046  -6.419   5.022    0.147   0.078   432.0   758.0   1.00
beta_train_type[TREHOTEL] -0.025  1.493  -2.670   2.845    0.362   0.178    13.0    49.0   1.16

Зведення коефіцієнтів тарифу:
      mean      sd  hdi_3%  hdi_97%  mcse_mean  mcse_sd  ess_bulk  ess_tail  r_hat
beta_fare[Adulto ida]    0.178  3.523  -6.395   6.744    0.160   0.092   485.0   722.0   1.00
beta_fare[Flexible]      0.536  2.488  -3.870   5.181    0.119   0.067   439.0   773.0   1.01
beta_fare[Promo]         0.120  2.488  -4.288   4.771    0.119   0.067   439.0   773.0   1.01
beta_fare[Promo +]       0.178  2.488  -4.067   4.993    0.119   0.067   440.0   805.0   1.01

Базовий тип поїзда: ALVIA
Поїзд типу 'AV City' у 1.12 разів дорожчий/дешевший, ніж 'ALVIA'
Поїзд типу 'AVE' у 1.54 разів дорожчий/дешевший, ніж 'ALVIA'
Поїзд типу 'AVE-LD' у 0.90 разів дорожчий/дешевший, ніж 'ALVIA'
Поїзд типу 'AVE-MD' у 0.99 разів дорожчий/дешевший, ніж 'ALVIA'
Поїзд типу 'AVE-TGV' у 1.85 разів дорожчий/дешевший, ніж 'ALVIA'
Поїзд типу 'INTERCITY' у 0.79 разів дорожчий/дешевший, ніж 'ALVIA'
Поїзд типу 'LD' у 0.85 разів дорожчий/дешевший, ніж 'ALVIA'
Поїзд типу 'LD-MD' у 0.77 разів дорожчий/дешевший, ніж 'ALVIA'
Поїзд типу 'MD' у 0.83 разів дорожчий/дешевший, ніж 'ALVIA'
Поїзд типу 'MD-AVE' у 0.87 разів дорожчий/дешевший, ніж 'ALVIA'
Поїзд типу 'MD-LD' у 0.76 разів дорожчий/дешевший, ніж 'ALVIA'
Поїзд типу 'R. EXPRES' у 0.97 разів дорожчий/дешевший, ніж 'ALVIA'
Поїзд типу 'REGIONAL' у 0.63 разів дорожчий/дешевший, ніж 'ALVIA'
Поїзд типу 'TREHOTEL' у 0.73 разів дорожчий/дешевший, ніж 'ALVIA'

```

Рис.6.7. Вивід в консоль

Посилання на гіт: <https://github.com/VadymLeus/Y4S1-AIS>

Висновок: в ході виконання лабораторної роботи були набуті навички працювати з даними і опонувано роботу у Python з використанням теореми Байєса.