

АНАЛІЗ ПОВЕДІНКОВИХ ДАНИХ ТА РОЗРОБКА АНАЛІТИЧНИХ МОДЕЛЕЙ ДЛЯ ПРИЙНЯТТЯ РІШЕНЬ

Апробація аналітичних моделей на предметній області
«Ідентифікація діамантів»

Дослідницька РГР з «Інтелектуального аналізу даних»

Виконавець: Бондаренко Олег, I26



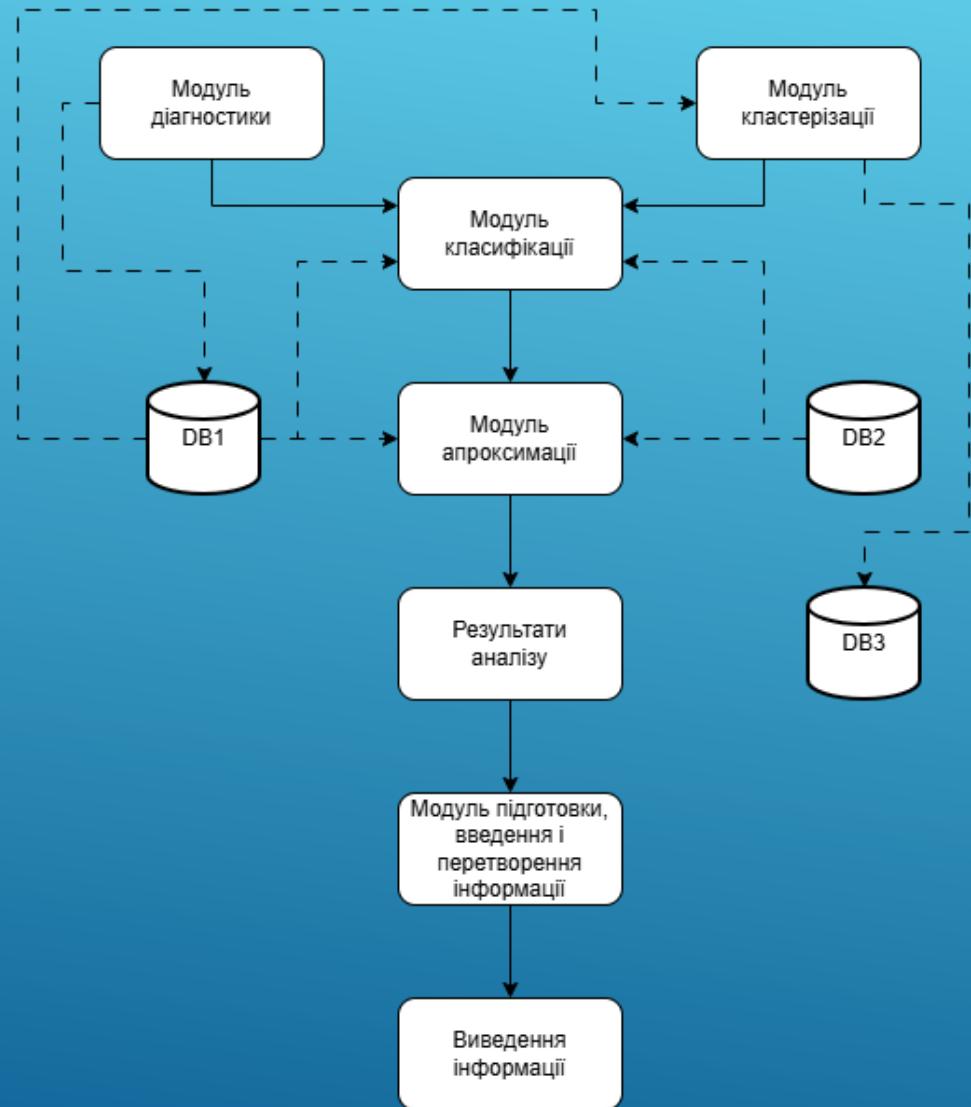
Мета

Формування системи інтелектуального аналізу поведінкових та предметно-орієнтованих даних з метою моделювання сценаріїв розвитку бізнес-рішень, виявлення трендів та оптимізації взаємодії.

Завдання

1. Отримати вибіркові дані поведінкових метрик користувачів веб-проєкту з різних джерел (первинна предметна область) та провести їх первинну обробку.
2. Здійснити регресійний та survival-аналіз для виявлення впливу поведінкових ознак на ймовірність покупки або конверсії.
3. Провести розкладання часових рядів поведінкових метрик на складові (тренд, сезонність).
4. **Провести апробацію розробленого комплексу методів на новій предметній області – ідентифікація діамантів.**
5. Сформувати висновки та рекомендації щодо оптимізації бізнес-рішень на основі виявлених закономірностей, трендів та результатів моделювання.

МЕТА ТА ЗАВДАННЯ

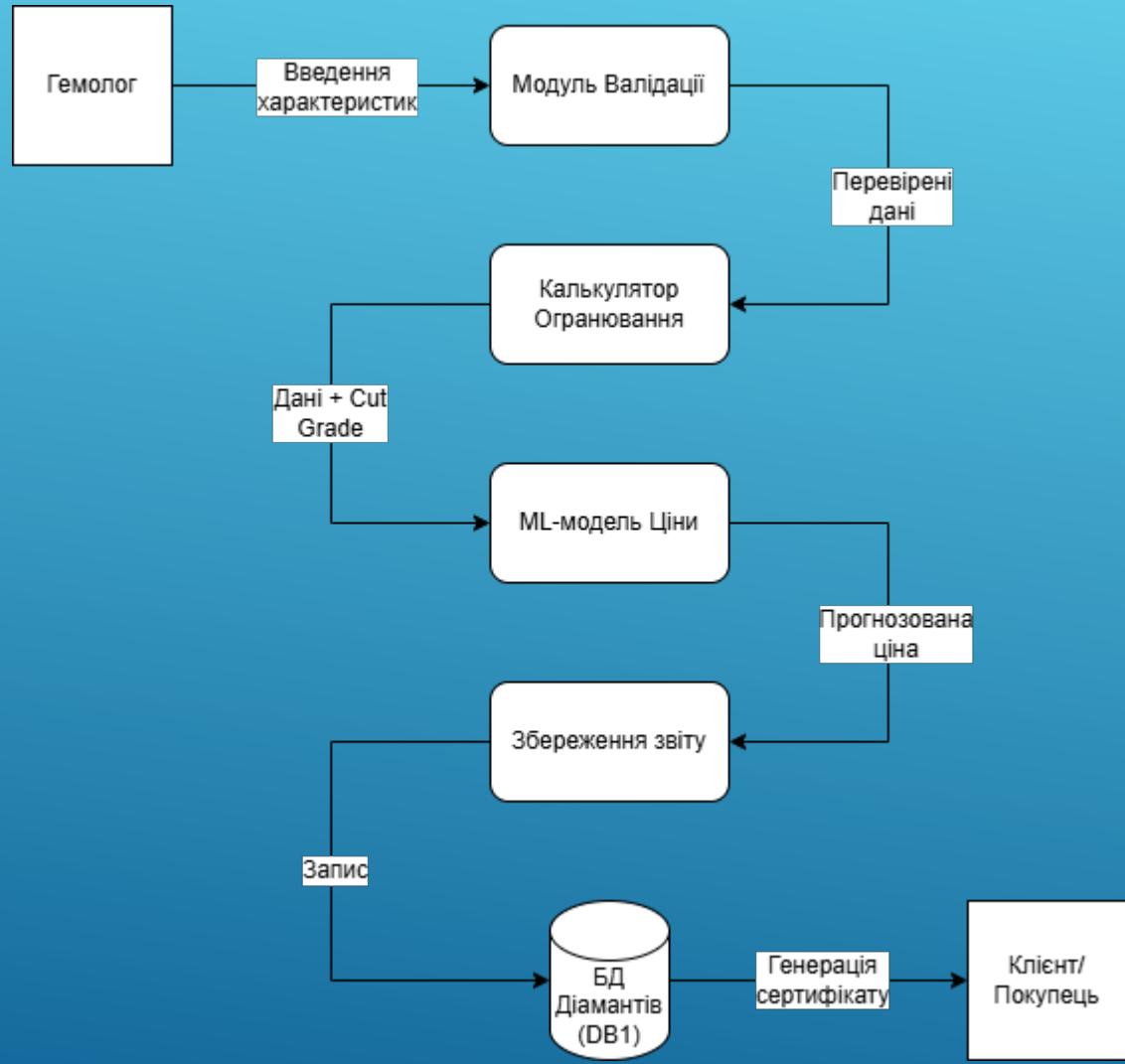


КОНЦЕПТ

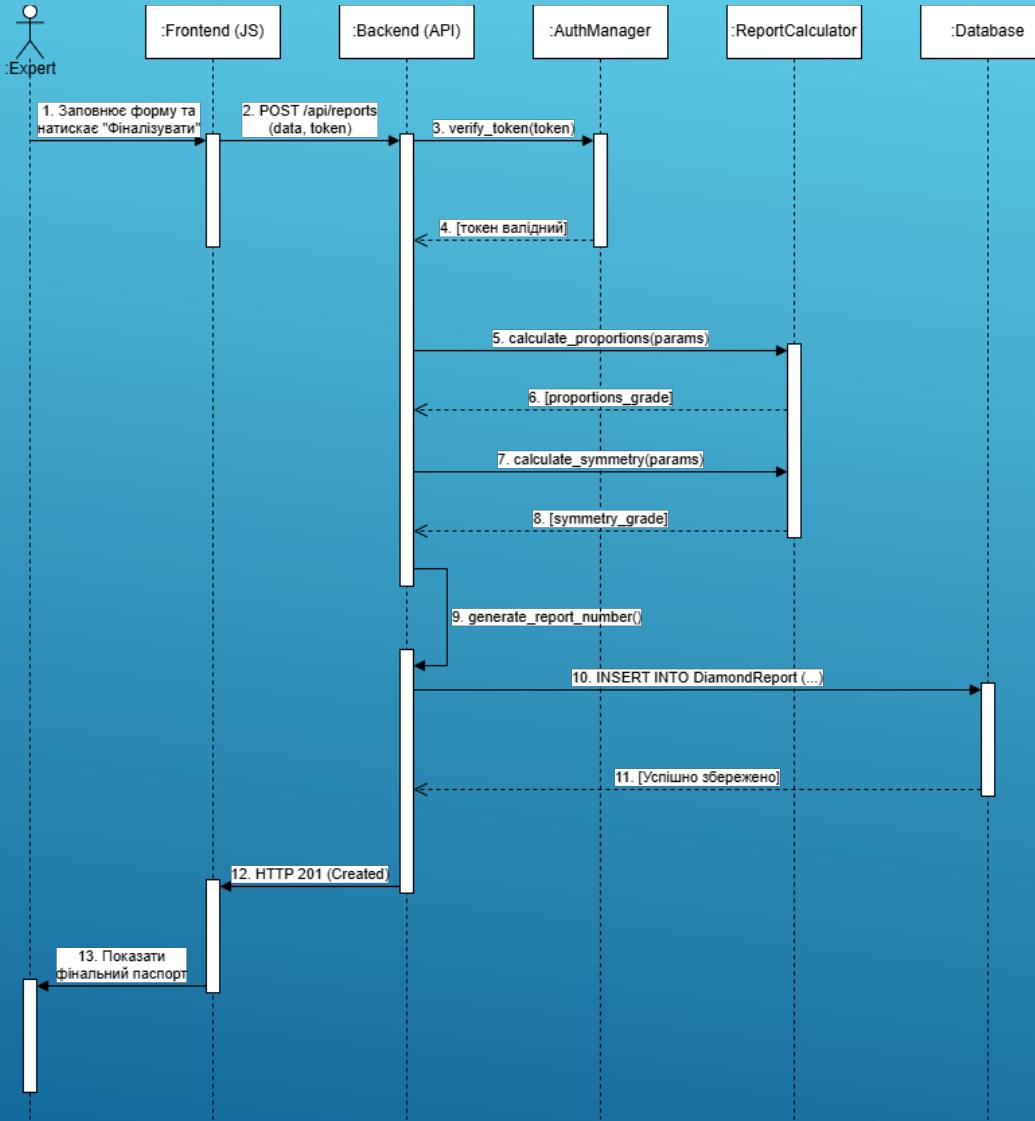
Модуль діагностики (інформаційна система) слугує для збору первинних транзакційних даних (OLTP) та їх збереження у DB1 (базі даних діамантів).

Модулі аналітики (ML) (кластеризація, класифікація, апроксимація) працюють «за лаштунками», використовуючи дані з DB1 (характеристики) та DB2 (ринкові показники) для навчання моделей.

Результати аналізу зберігаються у DB3 (базі груп/кластерів) або повертаються користувачу через Модуль виведення інформації.



ФРАГМЕНТ ДІАГРАМИ ПОТОКІВ ДАНИХ (DATA FLOW DIAGRAM)



ПРОЕКТУВАННЯ ЛОГІКИ

ПРОТОТИП UI: МОДУЛЬ ДІАГНОСТИКИ

Експерт вводить crown_angle (ліворуч), і цей код (праворуч) перетворює його на proportions_grade («Excellent»)

Виклик методу

```
1 # ReportCalculator
2 #
3 # /modules/report_calculator.py
4 #
5 class ReportCalculator:
6
7     def calculate_proportions_grade(
8         self,
9             table: float,
10            crown_angle: float,
11            pavilion_angle: float
12        ) -> str:
13
14     """
15     Розраховує оцінку пропорцій
16     на основі правил IDC (стор. 34)
17     """
18
19     # Правила IDC "зашиті" в коді
20     # (Це спрощена логіка для прикладу)
21
22     if (54 <= table <= 62 and
23         32.0 <= crown_angle <= 36.0 and
24         40.6 <= pavilion_angle <= 41.8):
25
26         return "Excellent"
27
28     elif (52 <= table <= 53 or
29           63 <= table <= 66):
30
31         return "Very Good"
32
33     else:
34
35         return "Good"
36
37     def calculate_symmetry_grade( ... ):
38         # ... (тут логіка для симетрії) ...
```

ПРОТОТИП UI: АНАЛІТИЧНА ПАНЕЛЬ

Інформаційна система не просто «записник». Вона бере збережені дані (ліворуч) і збагачує їх за допомогою ML-моделей (праворуч)

Diamond IS Dashboard Reports Analytics

Expert: O. Bondarenko ▾

Analytics Dashboard

Market Segments Map (SOM)

A heatmap representing Market Segments Map (SOM). The grid is composed of small squares colored blue, green, red, or yellow. A single yellow star is placed on one of the red squares in the lower-right quadrant of the map.

Legend

- Cluster 1 (Technical)
- Cluster 2 (Mass market)
- Cluster 5 (Investment)
- ★ Diamond

Analysis for Report: D-12345

Predicted Class
Investment Grade
(90% Confidence)

Predicted Market Value
\$10,500
(vs. List Price: \$10,200)

Market Segment
Cluster 5: High Value
Belongs to the top 15% of investment-grade stones.

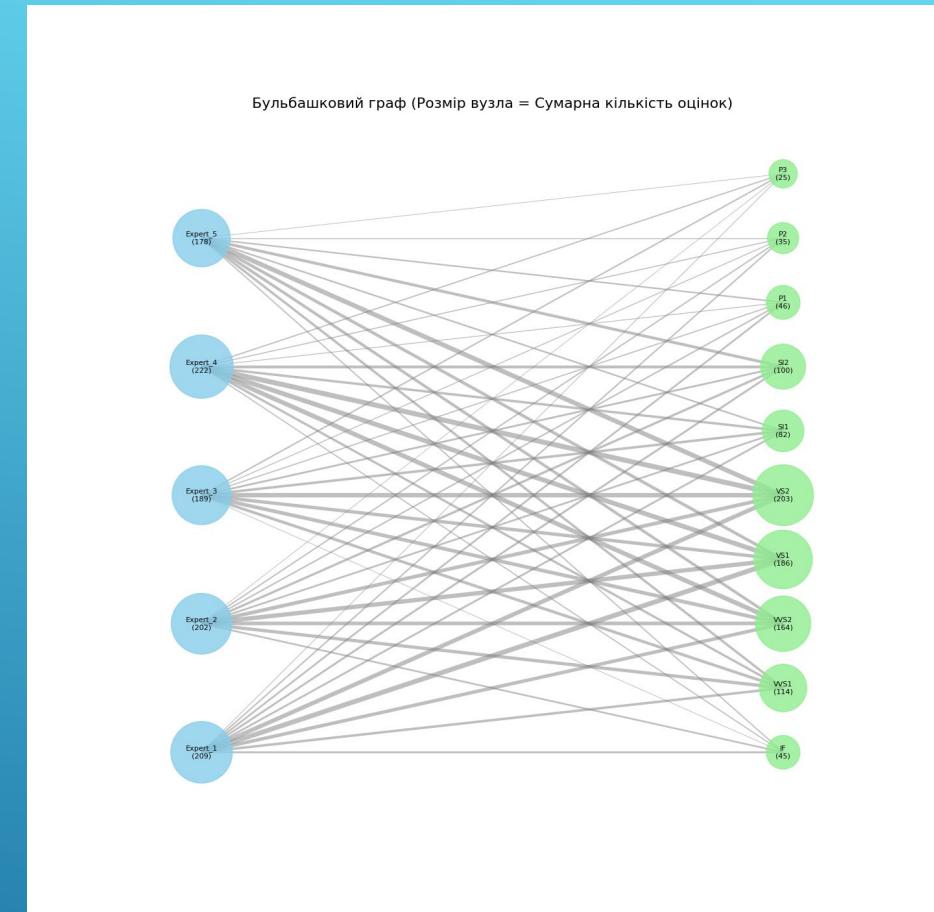
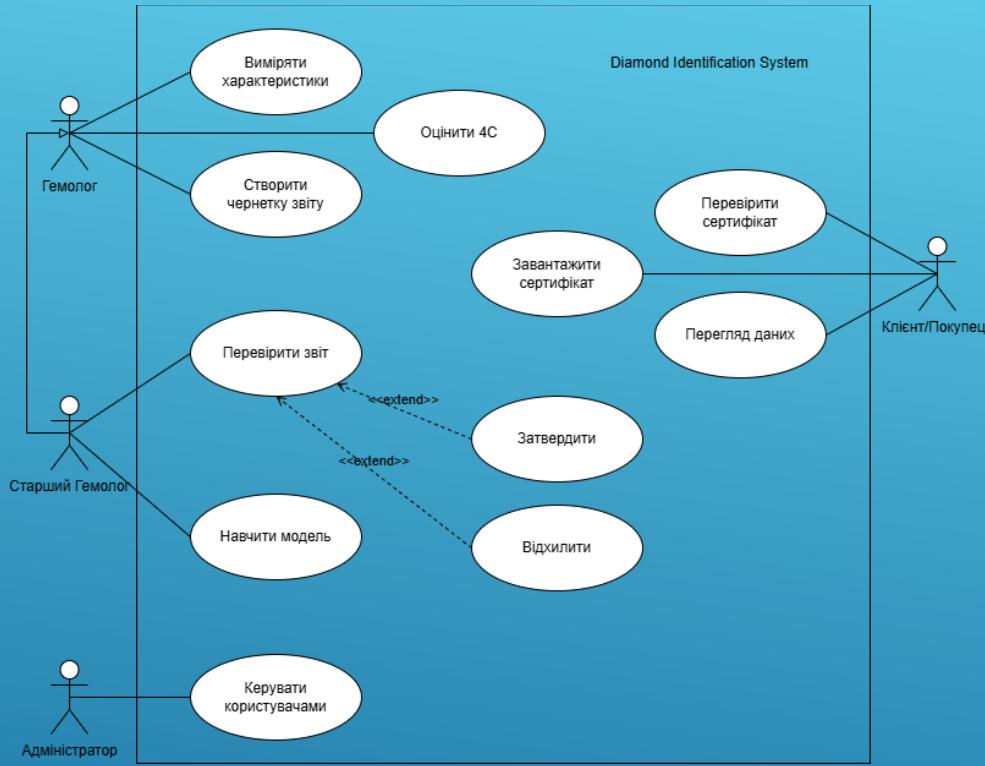
```
1 # FastAPI-ендпоінт
2 #
3 # /main.py (FastAPI роутер)
4 #
5 @app.get("/api/reports/{report_id}/analysis")
6 def get_report_analysis(report_id: int):
7
8     report = db.get_report(report_id) # Отримуємо звіт з DB1
9
10    # Готуємо дані для моделей
11    features = [report.carat_weight, report.crown_angle, ...]
12
13    # Викликаємо навченні ML-моделі
14
15    # 1. Апроксимація (Ціна)
16    predicted_price = price_model.predict(features)
17
18    # 2. Класифікація (Клас)
19    predicted_class = class_model.predict(features)
20
21    # 3. Кластеризація (Позиція на карті)
22    # (som_model.predict() повертає координати (x,y) на карті)
23    cluster_coords, cluster_name = som_model.predict(features)
24
25
26    return {
27        "price": predicted_price[0],
28        "class": predicted_class[0],
29        "cluster_info": {
30            "name": cluster_name,
31            "coords": cluster_coords
32        }
33    }
```



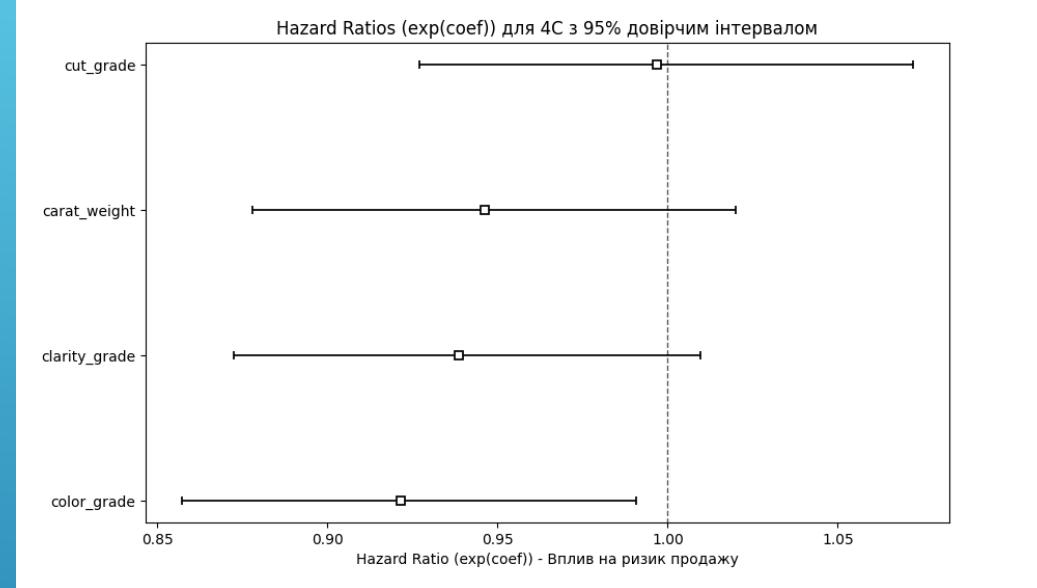
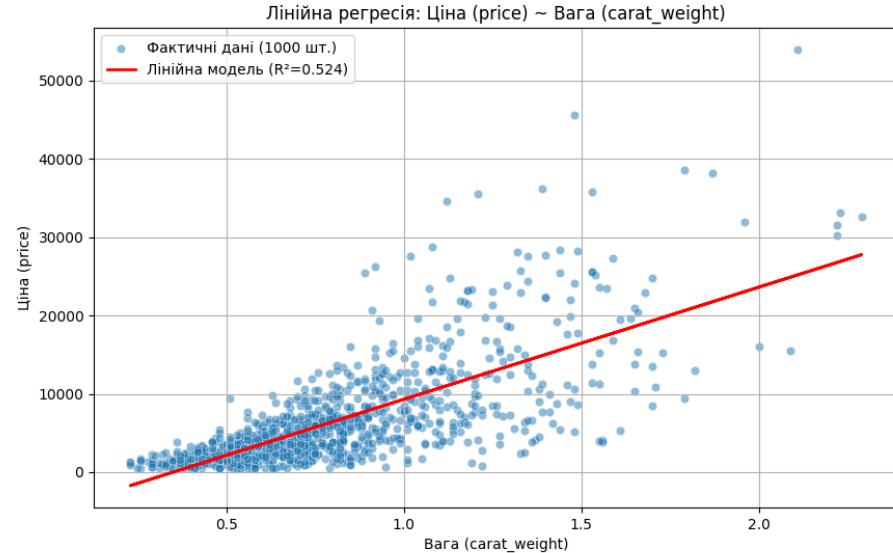
ПРАКТИЧНИЙ АНАЛІЗ 1: OLAP (ЗВЕДЕНІ ТАБЛИЦІ)

expert_name	2023	2024	2025
Expert_1	7 101	5 557	7 000
Expert_2	8 163	6 686	7 263
Expert_3	6 042	6 143	5 124
Expert_4	6 356	5 747	6 351
Expert_5	6 597	8 922	7 101

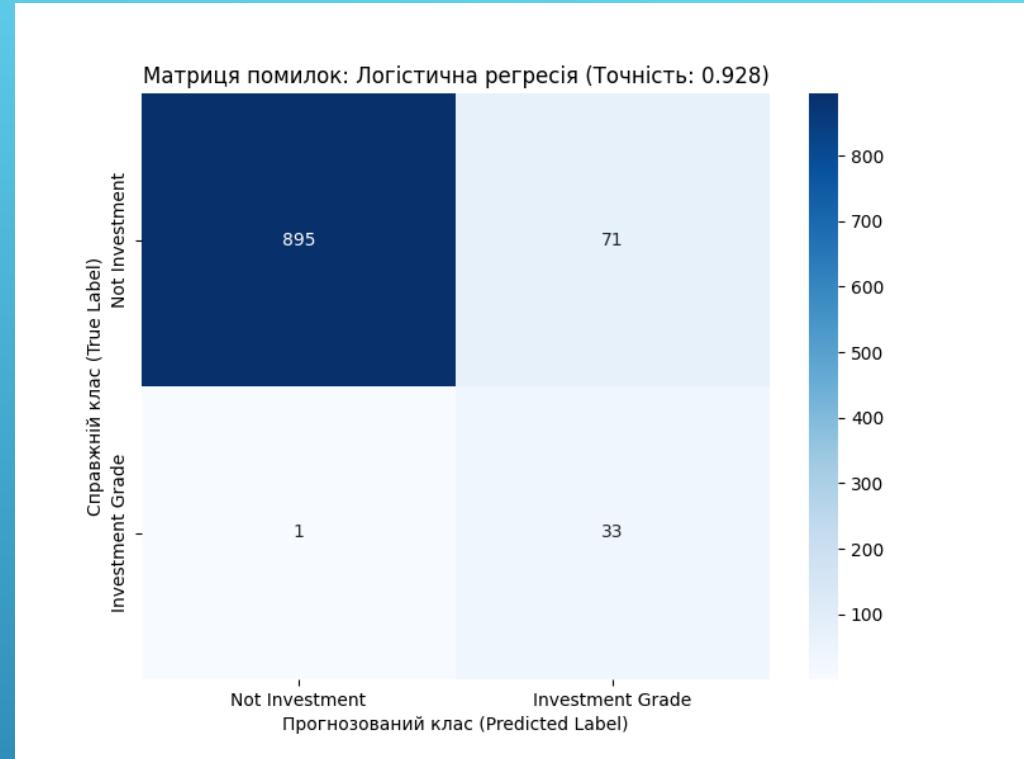
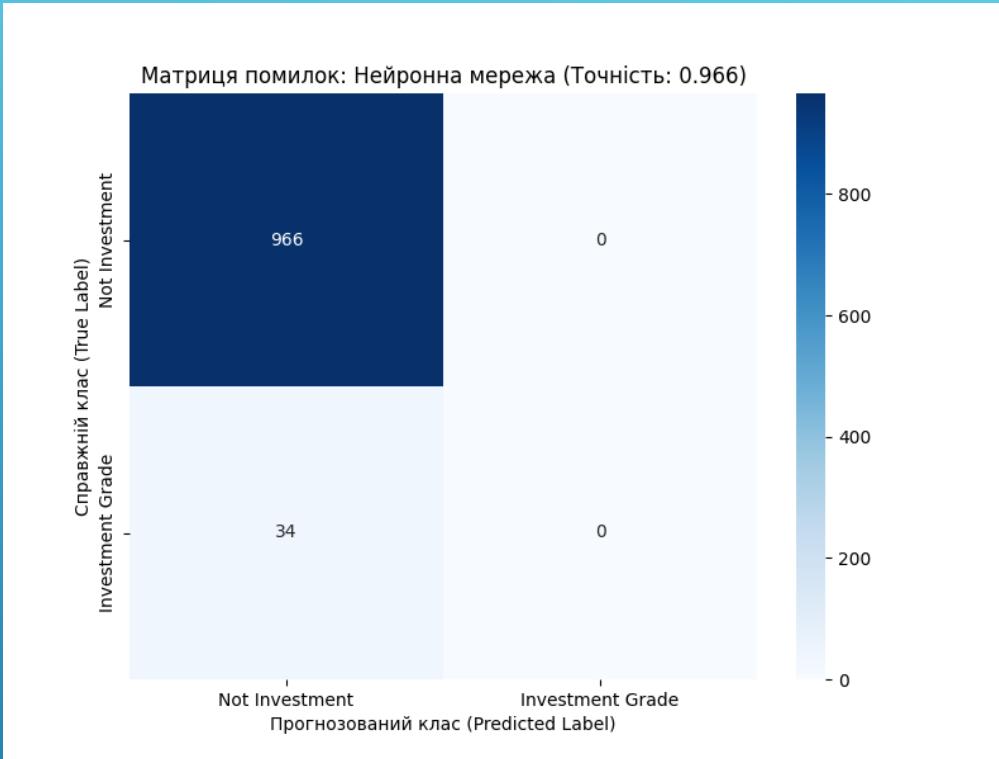
Таблиця дозволяє швидко оцінити динаміку середньої вартості роботи кожного експерта. Наприклад, ми можемо побачити, чи зростає середня вартість каменів, що проходять через Expert_1, з року в рік, чи, можливо, Expert_3 спеціалізується на дорожчих каменях.



ПРАКТИЧНИЙ АНАЛІЗ 2: МЕРЕЖЕВИЙ АНАЛІЗ (NETWORKX)



ПРАКТИЧНИЙ АНАЛІЗ 3: РЕГРЕСІЯ ТА SURVIVAL-АНАЛІЗ



ПРАКТИЧНИЙ АНАЛІЗ 4: КЛАСИФІКАЦІЯ ТА «ПАРАДОКС ТОЧНОСТІ»



Висновки

- ▶ Методологія РГР була успішно апробована на новій предметній області
- ▶ Спроєктовано повну архітектуру IC, що поєднує OLTP, OLAP та ML-модулі
- ▶ Практичний аналіз (OLAP, NetworkX, Регресія, Класифікація) на вітрині даних diamonds_dataset.csv довів свою ефективність

Подальші кроки

- ▶ Реалізація модуля Кластеризації (K-Means та візуалізація SOM)
- ▶ Реалізація модуля Аналізу часових рядів (для report_date)
- ▶ Імплементація Fuzzy Logic (Мамдані та Сугено) для покращення ReportCalculator

ВИСНОВКИ ТА ПОДАЛЬШІ КРОКИ