**Міністерство освіти і науки України**

**Національний технічний університет України «КПІ» імені Ігоря Сікорського**

**Кафедра обчислювальної техніки ФІОТ**

**ЗВІТ**

**з лабораторної роботи №8**

**з навчальної дисципліни «Вступ до технології Data Science»**

**Тема:**

**ДОСЛІДЖЕННЯ ТЕХНОЛОГІЇ ВИЗНАЧЕННЯ КРЕДИТНИХ РИЗИКІВ ДЛЯ БАНКІВСЬКИХ CRM СИСТЕМ**

**Виконав:**

Студент X курсу кафедри ОТ ФІОТ,

Навчальної групи ІА-73

Петренко П.П.

**Перевірив:**

Професор кафедри ОТ ФІОТ

Писарчук О.О.

**Київ 2023**

**І. Мета:**

виявити дослідити та узагальнити особливості застосування методів визначення кредитних ризиків для банківських CRM, ERP систем з використанням спеціалізованих пакетів мови програмування Python.

**ІІ. Завдання:**

| 3, 18 | Розробити програмний скрипт, що реалізує:  1. Парсінг файлів параметрів: Pr15\_data\_description.xlsx, Pr15\_sample\_data.xlsx;  2. Вибір індикаторів скорингової таблиці (25 шт.);  3. Очищення даних;  4. Розрахунок інтегрованої оцінки Scor за самостійно обраною моделлю;  5. Кластеризацію позичальників за бінарною характеристикою надання / відмова у кредитування;  6. Візуалізація результатів розрахунків у формі графіку, файлів (таблиці). |
| --- | --- |

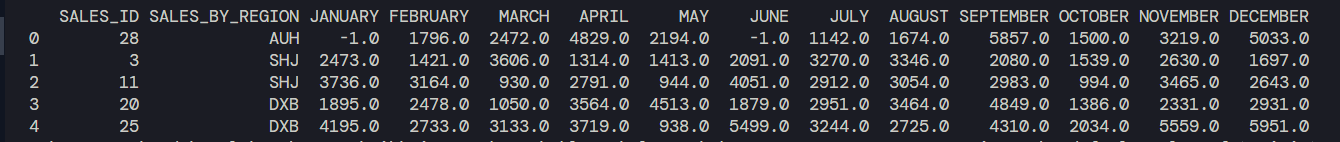


Рисунок 1. Очищені дані

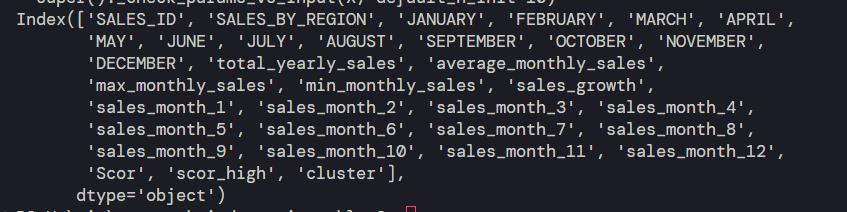


Рисунок 2. Індикатори ( колонки )

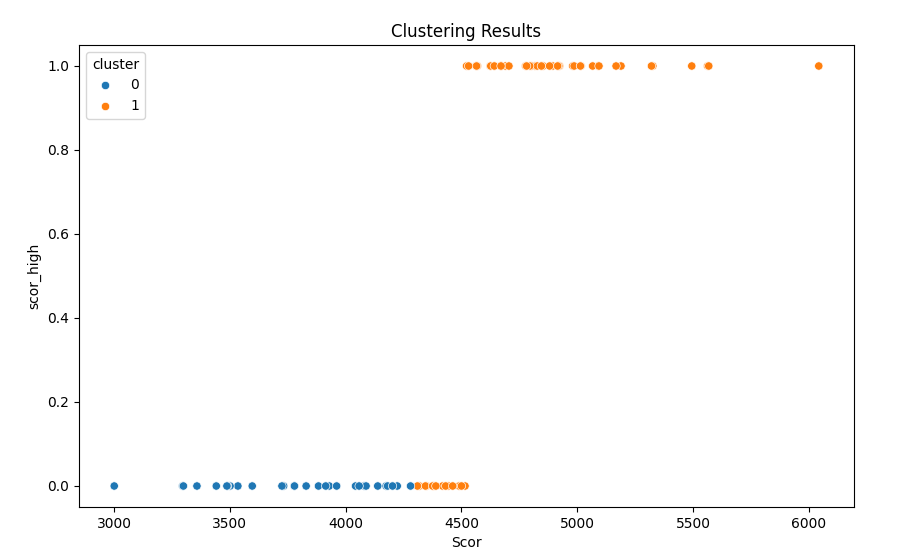


Рисунок 3. кластеризація

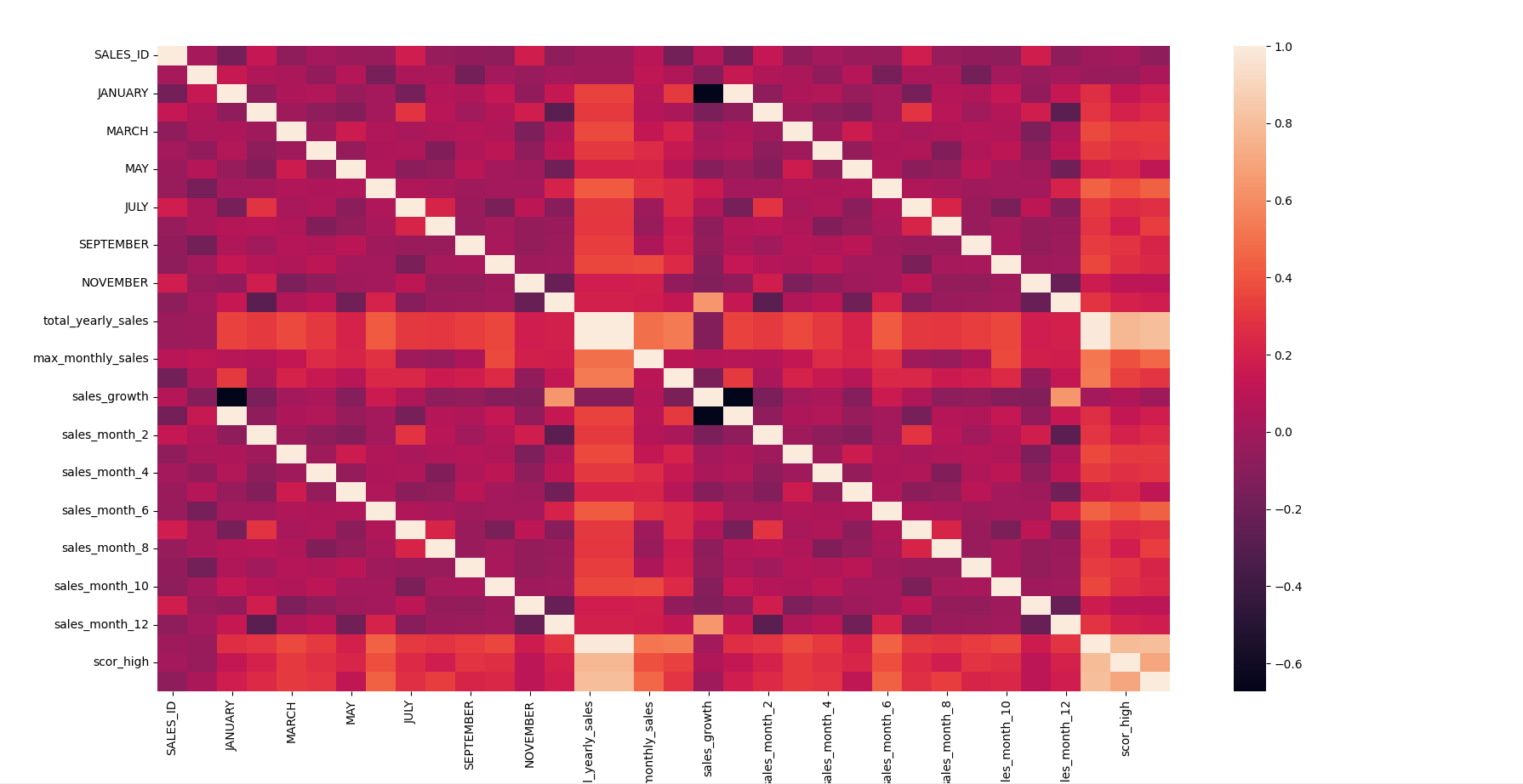


Рисунок 4. корреляція колонок

**ІІІ. Довести адекватність сформованих моделей та працездатність розробленого скріпта.**

**Розроблений код повинен бути раціональним та відповідати вимогам до чистого коду.**

**import pandas as pd**

**import numpy as np**

**import matplotlib.pyplot as plt**

**from sklearn.cluster import KMeans**

**import matplotlib.pyplot as plt**

**import seaborn as sns**

**df = pd.read\_csv('dataset.csv')**

***# Strip leading and trailing whitespace***

**df = df.applymap(*lambda* *x*: x.strip() if isinstance(x, str) else x)**

***# Replace 'n.a.' and 'not avilable' with NaN***

**df = df.replace(['n.a.', 'not avilable', 'not available'], np.nan)**

***# Remove commas from numeric columns***

**df.iloc[:, 2:] = df.iloc[:, 2:].replace({',': ''}, *regex*=True)**

***# Convert numeric columns to numeric type***

**df.iloc[:, 2:] = df.iloc[:, 2:].apply(pd.to\_numeric, *errors*='coerce')**

**df = df.dropna()**

**df = df.reset\_index(*drop*=True)**

**print(df.info())**

**print(df.describe())**

**print(df.head(5))**

***# Total yearly sales***

**df['total\_yearly\_sales'] = df.iloc[:, 2:].sum(*axis*=1)**

***# Average monthly sales***

**df['average\_monthly\_sales'] = df.iloc[:, 2:-1].mean(*axis*=1)**

***# Maximum monthly sales***

**df['max\_monthly\_sales'] = df.iloc[:, 2:-2].max(*axis*=1)**

***# Minimum monthly sales***

**df['min\_monthly\_sales'] = df.iloc[:, 2:-3].min(*axis*=1)**

***# Sales growth from January to December***

**df['sales\_growth'] = df['DECEMBER'] - df['JANUARY']**

***# Minimum monthly sales***

**df['min\_monthly\_sales'] = df.iloc[:, 2:-3].min(*axis*=1)**

***# Sales growth from January to December***

**df['sales\_growth'] = df['DECEMBER'] - df['JANUARY']**

***# Monthly sales for each month***

**months = ['JANUARY', 'FEBRUARY', 'MARCH', 'APRIL', 'MAY', 'JUNE', 'JULY', 'AUGUST', 'SEPTEMBER', 'OCTOBER', 'NOVEMBER', 'DECEMBER']**

**for i, month in enumerate(months):**

**df[*f*'sales\_month\_{i+1}'] = df[month]**

**months = ['JANUARY', 'FEBRUARY', 'MARCH', 'APRIL', 'MAY', 'JUNE', 'JULY', 'AUGUST', 'SEPTEMBER', 'OCTOBER', 'NOVEMBER', 'DECEMBER']**

**for i, month in enumerate(months):**

**df[*f*'sales\_month\_{i+1}'] = df[month]**

***def* calculate\_integrated\_score(*df*):**

***# This will depend on your specific scoring model***

**df['Scor'] = df.iloc[:, 2:].mean(*axis*=1)**

**return df**

**df = calculate\_integrated\_score(df)**

**kmeans = KMeans(*n\_clusters*=2)**

**df['scor\_high'] = (df['Scor'] > df['Scor'].median()).astype(int)**

**df['cluster'] = kmeans.fit\_predict(df[['Scor', 'scor\_high']])**

**plt.figure(*figsize*=(10, 6))**

**sns.scatterplot(*data*=df, *x*='Scor', *y*='scor\_high', *hue*='cluster')**

**plt.title('Clustering Results')**

**plt.show()**

**df['SALES\_BY\_REGION'] = pd.factorize(df['SALES\_BY\_REGION'])[0]**

**print(df.columns)**

**sns.heatmap(df.corr(), *annot*=False)**

**plt.show()**

**III. Висновок:**

виявив дослідив та узагальнив особливості застосування методів визначення кредитних ризиків для банківських CRM, ERP систем з використанням спеціалізованих пакетів мови програмування Python.

Виконав: студент Петренко П.П.