Національний університет «Полтавська політехніка імені Юрія Кондратюка»
(повне найменування вищого навчального закладу)
Навчально-науковий інститут інформаційних технологій та робототехніки
(повна назва інституту)
Кафедра комп'ютерних та інформаційних технологій і систем
(повна назва кафедри)

Пояснювальна записка
до дипломного проекту (роботи)
магістра
(рівень вищої освіти)
на тему
Інтелектуалізація аналізу та прогнозування попиту в оптовій та роздрібній торгівлі технічними оливами

Виконала: студентка 2 курсу, групи 601-ТН
спеціальності
122 Комп'ютерні науки
(шифр і назва спеціальності)
Рукас Т.С.
(прізвище та ініціали)
Керівник Ляхов О.Л.
(прізвище та ініціали)
Рецензент _____
(прізвище та ініціали)

Полтава – 2021 року

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

«ПОЛТАВСЬКА ПОЛІТЕХНІКА ІМЕНІ ЮРІЯ КОНДРАТЮКА»

НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ ТА РОБОТОТЕХНІКИ

КАФЕДРА КОМП'ЮТЕРНИХ ТА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ І СИСТЕМ

КВАЛІФІКАЦІЙНА РОБОТА МАГІСТРА

спеціальність 122 «Комп'ютерні науки»

на тему:

«Інтелектуалізація аналізу та прогнозування попиту в оптовій та роздрібній торгівлі технічними оливами»

Студентки групи 601-ТН Рукас Тетяни Сергіївни

Керівник роботи
доктор технічних наук,
професор Ляхов О.Л.

Завідувач кафедри
кандидат технічних наук,
доцент Головко Г. В.

Полтава – 2021

РЕФЕРАТ

Кваліфікаційна робота магістра: 70 с., 35 рисунків, 2 таблиці, 3 додатки, 30 джерел.

Об’єктом дослідження є торгівельні процеси в умовах нестабільного ринку та наявності великої кількості конкурентів.

Мета роботи: розробка методів для автоматизації процесу замовлення товарів, контролю складських запасів та підвищення прибутковості компанії.

Методи: аналіз продажу товарів компанії, проектування та розробка експертної системи, класифікація товарів та проведення машинного навчання у двох програмах.

Ключові слова: автоматизована система, експертна система, машинне навчання, база даних.

ANNOTATION

The explanatory note contains: 70 pages, 35 img.,2 tables, 30 sources, 3 additions.

The object of development: trading processes in an unstable market and the presence of a large number of competitors.

The purpose:development of methods for automating the process of ordering goods, control of inventory and increase the company's profitability.

Methods:analysis of sales of the company's products, design and development of an expert system, classification og goods and machine learning in two programs.

Keywords: automated system, expert system, machine learning, database.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СКОРОЧЕНЬ І ТЕРМІНІВ

БД – база даних.

ІС – інформаційна система.

ПЗ – програмне забезпечення.

АІС – автоматизована інформаційна система.

МН – машинне навчання.

KNN – метод k-найближчих сусідів.

МАП – методи аналізу продажу.

AI– штучний інтелект.

ВСТУП

Різні методи аналізу обсягу продажів компанії в оптовій та роздрібній торгівлі допомагають підвищити ефективність роботи будь-якого підприємства, якщо ними правильно користуватися. Це необхідна практика для кожного бізнесу, який планує розвиватися, а не стояти на місці. Коли підприємець аналізує отримані значення, він бачить перспективи розвитку. Але для контроля прибутку радиться проводити відразу кілька різних розрахунків.

В своїй роботі я проведу аналіз продажу товарів підприємства, на якому працюю. Зроблю це з метою визначення, який товар більше користується попитом, а який менше, а також товари, які є найпріоритетнішими в закупівлі – приносять найбільший прибуток підприємству. Визначення найрентабельніших товарів зорієнтує відділ закупівель щодо того, які позиції завжди потрібно мати в наявності та планувати їх замовлення у постачальника заздалегідь, і ні в якому разі не допускати дефіциту на складі. На відміну від звичайних аналізів продажу товарів, які я ще опишу детальніше у своїй роботі, моя підсумкова таблиця буде містити інформацію про основні характеристики товарів. Це зроблено з метою швидкої заміни необхідного продукту на аналогічний або дуже схожий у випадку форс-мажору чи припинення поставок від постачальника. Тобто, за допомогою програми, працівник, навіть без досвіду роботи, може легко визначити пріоритетність нової одиниці товару або швидко знайти на ринку необхідну одиницю товару за заданими в таблиці параметрами.

З іншого боку, треба враховувати, що запаси менш потрібних товарів заморожують значні обсяги оборотних коштів, ізолюють ланки системи руху товару та стадії бізнес-процесів один від одного, заважають оптимізації завантаженості складів. Тому особливу увагу треба звернути і на ту групу товарів, що користується найменшим попитом у покупців та прийняти

відповідне рішення щодо того, в якій кількості потрібно тримати запаси чи взагалі привозити деякі позиції лише за спеціальним замовленням від покупця. Працівники складської логістики можуть використати мій звіт для оптимізації робочих процесів: зручнішого розташування «популярних» товарів на території складу та пришвидшення вигрузки та погрузки товарів.

Отже, використовуючи алгоритми машинного навчання в своїй роботі, буде вирішено такі проблеми фірми:

- класифікація товарів по групам залежно від об’єму продаж та прибутку за певний період;
- збільшення прибутку компанії завдяки концентрації на топових позиціях;
- зменшення витрат шляхом зменшення запасів тих позицій, що продаються найменше або зовсім не продаються;
- здатність приймати правильне рішення в максимально короткий термін під час пошуку товару, який потрібно замінити на аналогічну позицію. Замінюючи швидко необхідну позицію, компанія отримає заплановані прибутки від продажів;
- можливість доповнювати асортимент новими товарами з відповідними вхідними даними, попередньо проаналізувавши, який вони матимуть попит серед покупців;
- покращення співпраці з клієнтами завдяки тому, що в наявності завжди будуть необхідні товари, або швидко замінені на аналогічні;
- оперативна доставка товарів через оптимізацію роботи складу.

РОЗДІЛ 1
ОГЛЯД ПРЕДМЕТНОЇ ГАЛУЗІ

1.1Товарні запаси: актуальність і стратегії управління

Питання управління запасами давно обговорюються в науці і практиці, активно опрацьовуються і використовуються як маркетинг-менеджментом, так і іншими науковими дисциплінами (наприклад, логістикою), і зараз не втратили своєї актуальності. Це обумовлено потужним і об'єктивним впливом даної категорії на результативність всієї господарської діяльності організації. По-перше, запаси необхідні для безперервності виробничого процесу і надійного забезпечення попиту через згладжування його непередбачених коливань і збоїв в поставках товарів. По-друге, вони надають надійність системі внутрішнього управління організацією через підвищення рівня її автономності і незалежності від контрагентів.

1.2 Огляд основнихметодів аналізу продажу товарів

Розглянути продажі зі сторони об’ємів, динаміки, структури та асортименту поможуть широко відомі методи аналізу продажів. Далі коротко опишу основні методи.

Метод аналізу динаміки продажів. Його метою є виявлення загального стану фактичних об’ємів продажів в порівнянні з попередніми періодами. За допомогою цього методу виявляється ріст та зниження продаж. Аналіз динаміки проводиться по показнику виручки, але можна використовувати і інші інструменти аналізу продажів: клієнтська база, зростання прибутку тощо.

Метод –аналіз рентабельності. Йогомета –визначення ефективності продажів з економічної точки зору. Для аналізу необхідно мати дані плану рентабельності, а також фактичні дані. Як правило, плани виставляються

відповідно до наявного бізнес-плану або на основі минулих періодів. Рентабельність продажів дасть розуміння того, скільки можна отримати прибутку з однієї гривні виручки. Даний показник повинен бути більше нуля. Визначається за формулою:

Рентабельність продажів = Прибуток від продажів / Виручка.

В результаті такого аналізу, виставляються плани на наступні періоди, а також здійснюються заходи, що сприяють збільшенню рентабельності продажів.

Метод – аналізклієнтськоїбази. За його допомогою можна виявлятитемпиприростуклієнтів,атакожступеня опрацюваннянаявноїбази. Обсягклієнтів,якізробили покупку(тобтокінцевихспоживачів),прямовпливаєнаобсягпродажівіотриманий прибуток. Клієнт–целюдина, якаплатитькомпаніїсвоїгроші.Вінхочеотриматиякіснийтоварабопослугузасправедливуплату.Уразі,якщоклієнтові непідходитьякістьтовару,цінаабосервіс,тоугода невідбудеться,продажне пройде. Саме тому дуже важливо відслідковувати стан клієнтської бази підприємства.

Метод рівномірності попиту (XYZ).Мета цього методу – визначити, на які товари попит буде стабільним. За допомогою аналізу продажів, цим методом можна заощадити бюджет і час, відмовившись від продажу товарів, на які не буде попиту. До речі, відмінно підходить для аналізу роздрібного продажу товарів. По темі: XYZ-аналіз: як рахувати. Етапи аналізу такі: складається список товарів і виручки, яку приносить товар. Дані заносяться в таблицю Ексель і за допомогою формул визначається коефіцієнт варіації. Потім товарам привласнюється категорія X, Y або Z.

Аналіз структури чека. Мета методу – виявити кількість певного товару на конкретній торговій площадці. При застосуванні цього методу досліджується кілька показників: Лист minimum must list (MML) – мінімально

необхідний асортимент, список товарів, що складається з декількох ключових SKU; Середнє Stock Keeping Unit (SKU) – одиниця товару, конкретна асортиментна позиція. За допомогою облікових систем можна отримати звіт, який покаже, скільки Stock Keeping Unit в середньому продається в торговій точці. Чим вище показник середнє SKU, тим більше представленість на ринку.

Аналіз по матриці BCG. Визначаютьсяпріоритетнітоварнігрупи,які подалі принесутьнайбільшийдохід. Данийметодзаснованийнарозрахункунаступнихпоказників:часткаринкутовару,темпизростанняринкудляцьоготоваруіобсягпродажів. Післярозрахунківтоваривзалежностівідчасткиринкуітемпівзростанняринкурозподіляютьсявматриці.Обсягпродажіввідображаєтьсяза допомогоюгуртків.Результатианалізуоформляютьсявматрицю BCG.

Контрольний аналіз обсягу продажів. За допомогою цього методу можна виявити відхилення між осягнутими фактовими показниками з продажу від запланованих. На кожну товарну групу виставляються план продажів на день, на тиждень, на місяць та рік, а потім проводиться оцінка виконання планів. Підходить як для роздрібних продажів товарів, так і для оптових. Базою для розрахунків при цьому методі аналізу продажів виступає виручка, прибуток, рентабельність і інші заплановані показники.

Факторнийаналізпродажів. Йогометоюєвиявлення факторів, щовпливаютьнаобсягпродажівівякій мірі. Дляпроведенняфакторногоаналізунеобхіднорозуміти,щотакевиручкаіщозалежитьвонавідцінинапропонованийтоварта відобсягузбуту.Цінавсвоючергузалежитьвідвитрат. Так,крокзакроком,виявляютьсяфактори,яківпливаютьнаобсягпродажів.Аналізвідбуваєтьсяшляхомпорівняннядвохперіодів(поточногодо минулого).

Експертний аналіз. Мета експертного аналізу – це експрес-оцінка аналізу продажів. Даний вид аналізу дає дуже суб'єктивні результати, особливо, коли він проводиться постійно з використанням одних і тих же експертів, не

зацікавлених у достовірності даних. Хороший ефект від використання цього методу аналізу продажів досягається, якщо проводити опитування клієнтської бази, тобто контрагентів зовнішнього середовища фірми. Для цього виявляються фактори, а потім опитуються експерти або клієнти. Відповідно до їх оцінки, кожному фактору виставляється оцінка, потім вони групуються, і у результаті отримується зведена таблиця факторів, на які потрібно звернути увагу.

Наступний метод є найефективнішим і найпопулярнішим та підходить компанії, у якій я працюю, зважаючи на специфіку діяльності, томуя використаю його у своїй роботі. Цей метод має назву – АВС-аналіз. Мета – виявити частку того чи іншого продукту в загальному обсязі продажів.Цей інструмент широко використовується в роздрібній торгівлі і дозволяє побачити, який торговельний напрямок генерує виручку, а які групи товарів зовсім погано продаються і не приносять вигоди бізнесу.Історично походження методу пов'язано з рішенням постачальницьких проблем, а саме з необхідністю концентрації зусиль на тих продуктах, які мають найбільшу вагу в загальній вартості сировини і матеріалів. В принципі АВС-аналіз має дуже широку сферу застосування, оскільки відповідно до досліджуваних величин (наприклад, товари, клієнти) класифікації можуть бути піддані різні галузі.

АВС-аналіз застосовують в логістиці з метою скорочення запасів, скорочення кількості переміщень на складі, загальне збільшення прибутку підприємства.Ідея аналізу полягає в тому, щоб з усім натовпом однотипних об'єктів, наприклад, асортименту товарів, виділити найбільш значущі з точки зору визначеної мети.АВС-метод використовується в постачанні, в збутовій діяльності, при класифікації споживачів і при управлінні запасами.

Класичний порядок АВС-класифікації включає в себе ряд етапів:

1. Вибір критерію класифікації;
2. Складання таблиці АВС-аналізу;
3. Виділення класифікаційних груп.

Перший етап. Вибір критерію класифікації залежить від стратегії підприємства і вимагає спільного обговорення цього питання службою логістики з керівниками підрозділів підприємства. В якості критеріїв класифікації можуть виступати: ціна закупівлі; прибуток від продажів; частка прибутку; дохід від продажів; частка в обороті; рентабельність продажів; середній рівень запасу; частка в створених запасах; період (швидкість) обороту запасу. Вибір ознаки визначає цільове використання результатів процедури диференціювання (класифікації, стратифікації, поділу, типізації або угруповання). Недоліком класичного АВС-аналізу є одиничність (унікальність) використовуваної ознаки. Наприклад, товари диференціюються тільки по маржі або тільки за ціною, що не дозволяє дати комплексну оцінку аналізованого елементу (товару). Пропонують використовувати, по-перше, механізм послідовного перебору релевантних ознак. Спочатку диференціювання номенклатури виконується за першою ознакою, потім отримані групи (А, В і С) диференціюються за другою ознакою і т.д. По-друге, можливе створення синтетичної ознаки диференціювання, використання якої дозволить дати комплексну оцінку отриманих груп.

Другий етап класифікації включає в себе основні розрахунки і сортування отриманих результатів. Фіксування абсолютних значень цільової ознаки в уже згадуваному асортименті (номенклатурі) товарів і розрахунок їх відносних значень (частки) в загальному обсязі аналізованої сукупності. В рамках даного етапу проводиться, по-перше, формування бази первинних даних, за якими буде виконуватися аналіз. Наприклад, фіксуються ціни на аналізовані товари, розраховується питома абсолютна маржа від продажу товару. По-друге, розраховується відносна частка (внесок) кожного елемента (товару) до загального цільового результату. Наприклад, розраховується частка витрат на закупівлю товару 1 (в%) в загальній вартості закуплених товарів; частка маржинальної прибутку по товару 1 (в%) в загальному обсязі маржинальної прибутку компанії.

Третій етап класифікації – виділення груп класифікації. Проводиться на основі закону Парето. Згідно з цим законом 20% об'єктів, дають, як правило, 80% результатів. Відповідно, решта 80% об'єктів дають 20% результатів. Закон Парето є універсальним і поширюється на багато об'єктів живої і неживої природи.

Кожна товарна позиція має для організації різну ступінь важливості за тими чи іншими параметрами. Наприклад, присутність в товарній номенклатурі суб'єкта господарювання певної групи товарів приносить йому найбільшу, в порівнянні з іншими товарами, маржу. У той же час продавець не може відмовитися від інших, менш результативних позицій (товарних груп і конкретних товарів), що становлять асортимент. Їх присутність є обов'язковою з наступних причин. По-перше, «інші» безпосередньо формують товарну пропозицію продавця. Неможливо скласти асортимент виключно з «ходових» позицій – перелік таких товарів знову буде розділений на лідерів і аутсайдерів. По-друге, «інші» товари необхідні суб'єкту для відповідності формату, статусу. Для цілей і завдань маркетингу релевантними об'єктами можуть бути матеріальні, людські, фінансові, тимчасові та інформаційні потоки.

Це означає, що, по-перше, керовані об'єкти повинні належати одному або подібному виду, класу, відділу. По-друге, адміністративні заходи повинні викликати схожий відгук, наприклад, зниження цін на весь асортимент товарів магазину може привести одночасно як до зростання обсягу продажів бюджетних повсякденних позицій, так і до скорочення збуту елітних категорій.

1.3 Створення експертної таблиці

Одним з найскладніших етапів моєї роботи є створення експертної таблиці для автоматичного визначення класу нової позиції за основними ознаками.Якщо є проблемачи задача, яку неможливо вирішити самостійно, - треба звертатися до фахівців в цій галузі. Дані – це інформація, що отримана в

результаті спостережень та вимірювань окремих властивостей (атрибутів), що характеризують об’єкти, процеси та явища предметної області. Знання виходять з даних, що отримані емпіричним шляхом. Тобто знання – це зв’язки та закономірності предметної області, що отримані в результаті практичної діяльності та професійного досвіду, що дозволяє спеціалістам ставити та вирішувати задачі в даній області.Ключовиметапомприроботізізнаннямиєформуванняполязнань,цянетривіальназадачавключаєвиявленняівизначенняоб'єктівіпонятьпредметноїобласті,їхвластивостейізв'язківміжними,атакожподанняїхвнаочнійтаінтуїтивнозрозумілій формі.

1.3.1 Моделі представлення знань. Вданийчасдлярізнихпредметнихобластей розроблено декількамоделейподання знань.Більшістьзнихзведенадо наступних класів:

- продукційнімоделі;
- семантичнімережі;
- фрейми;
- формальнілогічнімоделі.

Далі цебезлічкласівможнарозбитинадвівеликігрупи:модульні та мережеві.

Продукційнамодельзаснованананаправилах,дозволяєуявитизнанняу виглядіпропозиційтипу"Якщо(умова),то(дія)".
Під"умовою"розуміютьсядеякапропозиція-зразок,за якоюздійснюєтьсяпошуквбазізнань,апід"дією"–дії,що здійснюютьсяприуспішному результатіпошуку(вониможутьбутипроміжними,які виступаютьдалі якумови,ітермінальнимиабоцільовими, щозавершують роботусистеми).

Основною її перевагою моделі на основі семантичних мережєте,щовонабільше за іншихвідповідаєсьогоденнимуявленнямпро організаціюдовготривалоїпам'ятілюдини.

Фрейм–цеобраздля поданнястереотипупоняття, об'єктаабо ситуації. Структурафреймутрадиційно можебутипредставленаякпереліквластивостей:

(Ім'я фрейма: (Ім'я1-гослота:значення1-гослота),

(Ім'я2-гослота:значення2-гослота),

.....

(Ім'яN-гослота:значенняN-гослота)).

Формальнілогічнімоделі.

Традиційновподаннізнаньвиділяютьформальнілогічнімоделі,заснованінакласичномуобчисленні предикатів1-гопорядку,колиобласть, що вивчається,абозавданняописуєтьсяу виглядінаборуаксіом.Реальночислення предикатів1-гопорядкувпромисловихекспертнихсистемахпрактично невикористовується.Цялогічнамодельможе бути застосованавосновномувдослідницьких"іграшкових"системах,тому щопред'являєдужевисоківимогиіобмеженнядопредметноїобласті.

1.3.2 Етапи створення експертних систем. З огляду на вищесказане, технологію розробки ЕС можна представити схемою, що включає наступні етапи:

1. Попередній етап – цей етап включає діяльність, попередню до рішення про розробку нової ЕС. У рамках цього етапу здійснюються конкретизація завдання, підбір експертів в даній галузі для спільної роботи, вибір відповідних інструментальних засобів. Головною особливістю цього етапу є те, що може бути прийнято рішення про недоцільність розробки ЕС для обраного завдання.

2. Етап прототипування – в ході цього етапу створюється прототип ЕС, призначений перевірки правильності обраних засобів і методів розробки

нової ЕС. До прототипу системи не пред'являються високі вимоги. Основне його завдання полягає в ілюстрації можливостей майбутньої системи для фахівців, які безпосередньо беруть участь в розробці, а також для потенційних користувачів. На цьому етапі може бути здійснено коригування проекту, уточнені час, вартість і необхідні ресурси для завершення роботи.

3. Етап доопрацювання – це по суті основний, найбільш рутинний і тривалий етап роботи над ЕС. Всі компоненти багаторазово тестуються і доводяться до відповідності вимогам проекту. Найбільшу складність викликає доробка і доказ адекватності та ефективності БЗ, так як кількість записів в ній може бути на порядок більше, ніж в прототипі. На практиці межа між етапами може бути розмита, а сам процес проектування є досить неформальним, так як пов'язаний з дослідженням і спробою копіювання діяльності людини. Велика кількість застосовуваних евристик, інтуїтивний підхід до вирішення завдань експертами роблять процес створення ЕС творчим. Втім, формалізація технології ЕС, розробка в її рамках математичних методів і алгоритмів формування і обробки знань – це і є суть сучасної теорії ЕС.

1.3.3 Приклади створення таблиць для вирішення задач машинного навчання. Вирішуючи завдання машинного навчання з учителем і без, важливо представити свої вхідні дані в форматі, зрозумілому для комп'ютера. Форма знайдених даних зазвичай описується одним із двох способів: структурованим чи неструктурованим. Структуровані дані складаються з рядків і стовпців: від скромної електронної таблиці до складних систем реляційних баз даних, структуровані дані – це інтуїтивно зрозумілий спосіб зберігання інформації. Можна описати користувачів за віком, статтю, датою створення облікового запису і частоті покупок в інтернет-магазині. Можна описати зображення пухлини за допомогою градацій сірого кольору для кожного пікселя або за допомогою розміру, форми і кольору пухлини.

У машинному навчанні кожен об'єкт або рядок називаються прикладом (sample) або точкою даних (data point), а стовпці-властивості, які описують ці приклади, називаються характеристиками або ознаками (features). Треба мати на увазі, що жоден алгоритм машинного навчання не зможе зробити прогноз за даними, які не містять ніякої корисної інформації. Наприклад, якщо єдина ознака пацієнта – це його прізвище, алгоритм не зможе передбачити його стать. Цієї інформації просто немає в даних. Якщо додати ще одну ознаку – ім'я пацієнта, то справа вже складеться краще, оскільки часто, знаючи ім'я людини, можна судити про її стать.

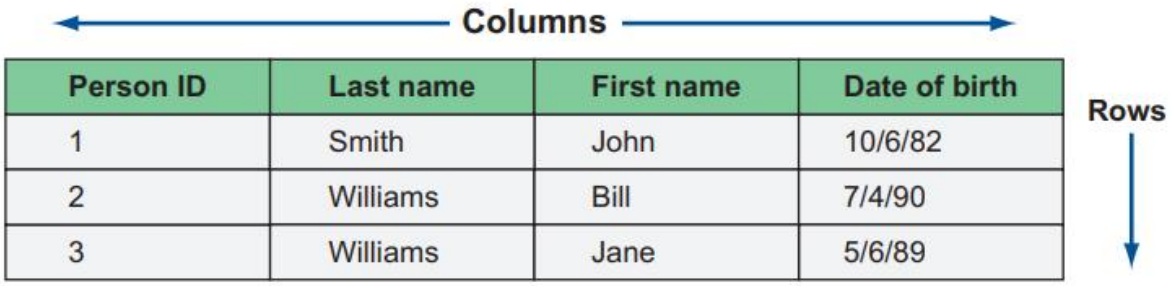

Рисунок. 1.1 – Приклад структурованого набору даних

На Рисунку 1.1 показаний приклад структурованого набору даних із рядками та стовпцями. Природно, щоб тяжіти до цього формату, коли думаєте про дані, оскільки ця структура допомагає утримувати пов’язані фрагменти інформації в одному візуальному просторі. Рядок представляє логічну сутність: у таблиці кожен рядок представляє людину. Ряди складаються з одного або декількох стовпців, які представляють те, що ми знаємо про кожну сутність. В таблиці ми записали прізвище, ім’я, дату народження та унікальний ідентифікатор кожної людини.

1.4Роль машинного навчання в аналітиці

Штучний інтелект (AI) і машинне навчання зараз вважаються одними з найбільших інновацій з часів мікрочіпа.Раніше AI був фантастичним поняттям з наукової фантастики, але тепер він стає повсякденною реальністю.Нейронні мережі прокладають шлях до проривів у машинному навчанні, які називають «глибоким навчанням». Машинне навчання може допомогти нам жити щасливішим, здоровішим і продуктивнішими, якщо ми знаємо, як використовувати його силу.

Деякі кажуть, що AI відкриває чергову «промислову революцію».У той час як попередня промислова революція використовувала фізичну та механічну силу, ця нова революція буде використовувати розумові та когнітивні здібності.Колись комп’ютери замінять не лише ручну, а й розумовудіяльність.

Машинне навчання – це область штучного інтелекту, яка розглядає спосіб розробки алгоритмів, з можливістю навчання. Машинне навчання лежить на стику традиційної математики і математичної статистики. Але має декілька важливих відмінностей від математичної статистики – машинне навчання зазвичай має справу з з великими та важкими наборами даних (наприклад, таких, що складаються з мільйонів фотографій, кожна з яких складається з десятків тисяч пікселів), до яких практично неможливо застосувати класичні

методи статистичного аналізу. Крім того вирішує проблеми, пов'язані з перенавчанням та точності та обчислень. У машинному навчанні система навчається, а не програмується явно. Їй передаються численні приклади, що мають відношення до задачі, що вирішується, а вона знаходить в цих прикладах статистичну структуру, яка дозволяє системівиробити правила для автоматичного вирішення задачі.Для того, щоб машина могла вчитися та тренуватися самостійно, ій потрібний великий об’єм даних – bigdata. Надійний набор для навчання містить структуровані, напівструктуровані та неструктуровані дані, допоможе машині розробити багатовимірне уявлення про реальний світ та проблему, для рішення якої вона призначена. Крім цього, необхідно порівнювати також ефективність побудованих моделей. Так як пошук і розмітка даних складне завдання, що вимагає великих трудовитрат, для тестування моделей використовують вже зібрані набори даних, наведені в зручний формат.

1.4.1 Навчання з учителем. Задачі класифікації та регресії. Однією з основних здібностей, якими володіли люди з первісних часів, є поділ відомого світу на окремі класи, де окремі об’єкти мають спільні риси, які класифікатор вважає важливими. Починаючи з того, що первісні мешканці печер класифікували природний світ, у якому вони жили, розрізняючи рослини та тварин, корисні чи небезпечні для їх виживання, ми приходимо до сучасності, коли відділи маркетингу класифікують споживачів за цільовими сегментами, а потім діють з належними маркетинговими планами.

Найбільш успішні алгоритми машинного навчання – це ті, що автоматизують процеси прийняття рішень шляхом узагальнення відомих прикладів. У цих методах, відомих як контрольоване навчання або навчання з учителем (*supervised learning*), користувач надає алгоритму пари об'єкт-відповідь, а алгоритм знаходить спосіб отримання відповіді по об'єкту. Зокрема, алгоритм здатний видати відповідь для об'єкта без будь-якої

допомоги людини, якого він ніколи не бачив раніше. Алгоритми машинного навчання, які навчаються на парах об'єкт-відповідь, називаються алгоритмами навчання з учителем, так як «вчитель» показує алгоритму відповідь в кожному спостереженні, за яким відбувається навчання. Незважаючи на те, що створення набору з об'єктами та відповідями – це часто трудомісткий процес, який здійснюється вручну, алгоритми навчання з учителем є інтерпретованими та якість їх роботи легко виміряти. Якщо завдання можна сформулювати у вигляді завдання навчання з учителем, і можна створити набір даних, який включає в себе відповіді, ймовірно, машинне навчання вирішить цю проблему.Робота з великими потоками даних сьогодні вимагає тієї ж класифікаційної здатності, але в іншому масштабі. Щоб виявити невідомі групи сигналів, що присутні у даних, нам потрібні спеціалізовані алгоритми, які можуть навчитися призначати приклади певним даним класам (підхід під наглядом) і виявити нові цікаві заняття про які ми не знали (навчання без нагляду).

Є два основні завдання машинного навчання з учителем: класифікація (classification) та регресія (regression).

Мета класифікації полягає в тому, щоб спрогнозувати мітку класу (class label), яка являє собою вибір зі заздалегідь певного списку можливих варіантів. Тобто, класифікація – це підкатегорія контрольованого навчання, метою якої є передбачення категорійних міток класів нових інстанцій на основі минулих спостережень. Ці мітки класів є дискретними, невпорядкованими значеннями, які можна розуміти як членство в групах екземплярів. Класифікація іноді розділяється на бінарну класифікацію (binary classification), яка є окремим випадком поділу на два класи та мультикласову класифікацію (multiclass classification), за якої в класифікації бере участь більше двох класів. Бінарну класифікацію можна представити як спробу відповісти на поставлене запитання в форматі «так/ні». Класифікація електронних листів на спам і неспам є прикладом бінарної класифікації. Класифікація має вирішальне

значення для процесу побудови нових знань, оскільки, збираючи подібні об’єкти, можна:

- згадати всі предмети в класі одного і того ж номіналу;
- узагальнити відповідні ознаки за зразковим типом класу;
- пов’язати певні дії або автоматично згадати певні знання.

Навіть, незважаючи на те, що основна рутина як науковця з даних – втілювати на практиці навички прогнозування, також доведеться дати корисне уявлення про можливу нову інформацію, наявну у ваших даних. Наприклад, досліднику часто доведеться знаходити нові функції, щоб посилити спрогнозування моделей, знайти простий спосіб зробити складні порівняння всередині даних та відкрити спільноти в соціальних мережах.

Підхід до класифікації, орієнтований на дані, що називається кластеризацією, виявиться великою допомогою у досягненні успіху у проекті даних, коли потрібно надати нову інформацію з нуля, та не вистачає позначених даних або хочеться створити для них нові мітки.

Мета регресії полягає в тому, щоб спрогнозувати число з плаваючою точкою (floating-point number) або безперервне число, якщо висловлюватись у термінах програмування, або дійсне число (real number), якщо говорити мовою математичних термінів. Прогнозування річного доходу людини в залежності від її освіти, віку і місця проживання є прикладом завдання регресії. Прогнозоване значення доходу являє собою суму (amount) і може бути будь-яким числом в заданому діапазоні. Найпростіший спосіб відрізнити класифікацію від регресії – запитати, чи закладена в отриманій відповіді певна безперервність (спадкоємність). Якщо отримані результати безперервно пов'язані один з одним, то дана задача є задачею регресії.

1.4.2 Навчання без учителя. Кластеризація. У навчанні без учителя виявляються внутрішні взаємозв'язки і закономірності об'єктів, так як за допомогою нього вирішуються завдання обробки даних, в яких відома лише навчальна вибірка (з описом численних об'єктів). Тобто тренувальні дані доступні всі відразу, але відповіді для поставленого завдання невідомі.

Основним завданням кластеризації є поділ задану представлених даних на незалежні групи, які іменуються кластерами. Дані кластери складаються з різних об'єктів, об'єднаних за певною ознакою, що істотно відрізняють об'єкт, від інших груп.

1.5 Вирішення задачі класифікації за допомогою програми Python 3.9

Згідно з останнім індексом спільноти програмістів TIOBE, Python є однією з 10 найпопулярніших мов програмування 2017 року. Python — мова програмування загального призначення та високого рівня. Ви можете використовувати Python для розробки настільних програм із графічним інтерфейсом користувача, веб-сайтів та веб-додатків. Крім того, Python, як мова програмування високого рівня, дозволяє зосередитися на основній функціональності програми, виконуючи загальні завдання програмування. Прості правила синтаксису мови програмування також полегшують вам підтримку читання бази коду та обслуговування програми. Існує також ряд причин, чому ви повинні віддати перевагу Python перед іншими мовами програмування.

Отже, ви можете використовувати Python для створення користувацьких програм без написання додаткового коду. Читабельна та чиста кодова база допоможе вам підтримувати та оновлювати програмне забезпечення без зайвих витрат часу та зусиль. Він повністю підтримує об'єктно-орієнтоване та структуроване програмування. Крім того, його мовні особливості підтримують різні концепції функціонального та аспектно-орієнтованого програмування. На

даний момент Python підтримує багато операційних систем. Ви навіть можете використовувати інтерпретатори Python для запуску коду на певних платформах та інструментах. Крім того, Python є інтерпретованою мовою програмування. Це дозволяє вам запускати той самий код на кількох платформах без перекомпіляції. Його велика та надійна стандартна бібліотека робить Python оцінкою над іншими мовами програмування. Стандартна бібліотека дозволяє вибирати з широкого спектру модулів відповідно до ваших потреб. Ви навіть можете використовувати кілька фреймворків Python з відкритим кодом, бібліотек та інструментів розробки, щоб скоротити час розробки без збільшення вартості розробки.

Отже, ви можете використовувати мову програмування для розробки як настільних, так і веб-додатків. Крім того, ви можете використовувати Python для розробки складних наукових і числових додатків. Python розроблено з функціями для полегшення аналізу та візуалізації даних. Ви можете скористатися можливостями аналізу даних Python, щоб створювати власні рішення для великих даних, не витрачаючи зайвого часу та зусиль.

Об’єкт Dataframe являє собою табличну структуру даних, що складається з упорядкованої колекції стовпців, причому типи значень в різних стовпцях можуть відрізнятися (числовий, рядки, булевий тощо). Є багато способів сконструювати об’єкт Dataframe у Python, найпопулярнішим серед них є – на основі словника списків однакової довжини або масивів NumPy:

```
In:
data = { ‘State’ :[ ‘Ohio’ ,  ‘Stat’ ],
              ‘year’ : [2000, 2002],
              ‘pop’ :[1.5, 1.8]}
frame = DataFrame(data)
print(frame)
Out:
```

```
pop    State  year
0      1.5      Ohio   2000
1      1.8      Stat2002
```

Також можна задавати послідовність стовпчиків в певному порядку, вивести на друк окремі стовпчики та безліч інших перетворень.

Python став загальноприйнятою мовою для багатьох сфер застосування науки про дані (data science). Він поєднує в собі міць мов програмування з простотою використання предметно-орієнтованих скриптових мов типу MATLAB або R. В Python є бібліотеки для завантаження даних,обробки природної мови та зображень, візуалізації, статистичних обчислень та багато чого іншого. Це пропонує фахівцям з роботи з даними (data scientists) великий набір інструментів загального і спеціального призначення. Вкрай важливо для цих процесів мати інструменти, які дозволяють оперативно і легко працювати. Машинне навчання та аналіз даних – це в основному ітераційні процеси, в яких дані задають хід аналізу.

1.5.1 Опис основних бібліотек Python для вирішення задач машинного навчання. Розглянемо основні бібліотеки, які необхідно встановити для виконання моєї роботи:

NumPy: Ця аналітична бібліотека надає користувачеві підтримку для багатовимірних масивів, включаючи математичні алгоритми, необхідні для їх оперування. Масиви використовуються для зберігання даних, а також для швидких матричних операцій, які дуже потрібні для вирішення багатьох проблем науки про дані. Python не призначений для здійснення обчислень, тому кожному вченому з даних потрібен такий пакет. NumPy розширює мову програмування, включаючи використання багатьох математичних функцій високого рівня. Щоб встановити цей інструмент, потрібно набрати наступну команду: pip install numpy.

SciPy: Комплекс пакетів, що призначені для вирішення різних стандартних обчислювальних задач. Разом з NumPy утворюють достатньо повну заміну значної частини системи Matlab та багатьох доповнень до неї. SciPy дає змогу використовувати алгоритми для обробки зображень, лінійної алгебри, матриці та інше. Команда для установки: pip install scipy.

Pandas: Є дуже популярною і потужною основою для аналізу структурованих даних. Ця бібліотека потрібна в основному для обробки різноманітних даних таблиці. Встановивши її користувач матиме можливість завантажувати дані з будь-якого джерела та маніпулювати в міру необхідності. Вона представляє розвинені засоби індексування, що дозволяють просто змінювати форму наборів даних, формувати поздовжні та поперечні зрізи, виконувати агрегування та обирати підмножини. Для розробки фінансових додатків бібліотека пропонує багатий набір високопродуктивних засобів аналізу часових рядів, спеціально орієнтованих на фінансові дані. Назва Pandas походить від "панельних даних", економетричного терміну з багатовимірних даних. Нижче наведено основні особливості бібліотеки Pandas:

- забезпечує механізм завантаження об’єктів даних із різних форматів;
- створює ефективні об'єкти фрейму даних за замовчуванням та індивідуальне індексування;
- змінює форму та повертає набори дат;
- забезпечує ефективні механізми обробки відсутніх даних;
- об’єднує, групує, агрегує та трансформує дані;
- маніпулює великими наборами даних шляхом реалізації різних функцій, таких як нарізка, індексування, підмножина, видалення та вставка;
- забезпечує ефективну функціональність часових рядів.

Іноді доводиться імпортувати пакет Pandas, оскільки стандартний дистрибутив Python не поставляється в комплекті з модулем Pandas.

Легкою альтернативою є установка Numpy за допомогою популярного Python інсталятор пакетів pip. Бібліотека Pandas використовується для створення та обробки серій, фреймів даних та панелей. Команда установки: pip install pandas.

Matplotlib: Найпопулярніший інструмент для створення графіків та інших способів візуалізації даних. Цей пакет містить все необхідне для побудови ділянок з масиву. Ви також маєте можливість їх візуалізувати інтерактивно. Графік використовується для статистики та аналізу даних для відображення співвідношення між змінними. Це робить Matplotlib незамінною бібліотекою для Python. Команда для установки: pip install matplotlib.

Scikit-learn: Дуже необхідний інструмент для науки про дані та машин вивчення, Scikit, мабуть, найважливіший пакет у вашому інструментарії. Це потрібно для попередньої обробки даних, показників помилок, контролю навчання без учителя та багато іншого. Установка: pip install scikit-learn. Scikit-learn– проектом з відкритим вихідним кодом, тобто його можна вільно використовувати і поширювати, і будь-яка людина може легко отримати вихідний код, щоб побачити, що відбувається «за лаштунками ». Проект постійно розвивається і вдосконалюється, і у нього дуже активна спільнота користувачів. Він містить ряд сучасних алгоритмів машинного навчання, а також повну документацію по кожному алгоритму. Це дуже популярний інструмент і найвідоміша «пітонівська» бібліотека для машинного навчання. Вона широко використовується в промисловості та науці, а в інтернеті є багатий вибір навчальних матеріалів і прикладів програмного коду. Scikit-learn прекрасно працює з низкою інших наукових інструментів Python.

1.5.2 Алгоритм k-найближчих сусідів. Алгоритм k-найближчих сусідів, можливо, є найпростішим алгоритмом машинного навчання. Класифікація k-найближчого сусіда (KNN) була розроблена на основі необхідності проводити дискримінантний аналіз, коли достовірні параметричні оцінки щільності ймовірності невідомі або важко визначити.

Фікс і Ходжес в 1951 році ввели непараметричний метод класифікації шаблонів, який з тих пір став відомий як правило k-найближчих сусідів. Як випливає з назви, алгоритм працює на основі більшості голосів свого класу k-найближчих сусідів. K-найближчі сусіди (KNN) не стосуються побудови правил на основі даних, коефіцієнтів або ймовірності. KNN працює на основі подібності.

Коли вам потрібно передбачити щось на зразок класу, найкраще знайти спостереження, найбільш подібні до тих, які ви хочете класифікувати чи оцінити. Тоді ви можете отримати необхідну відповідь із подібних випадків. Спостереження за кількістю подібних спостережень не означає вивчення чогось, а скоріше вимірювання. Оскільки KNN нічого не вивчає, його вважають ледачим, і ви почуєте, як його називають ледачим учнем або учнем на основі екземплярів. Ідея полягає в тому, що подібні умови зазвичай дають подібні результати.

Алгоритм швидкий під час навчання, тому що йому потрібно лише запам'ятати дані про спостереження. Він насправді розраховує більше під час прогнозів. Якщо спостережень занадто багато, алгоритм може стати повільним і зайняти багато пам'яті. Краще не використовувати його з великими даними, інакше прогнозування може зайняти майже вічність. Більш того, цей простий та ефективний алгоритм працює краще, якщо у вас є окремі групи даних без надто великої кількості змінних, оскільки алгоритм також чутливий до прокляття розмірності. Прокляття розмірності відбувається зі збільшенням кількості змінних.

Розглянемо ситуацію, коли вимірюється відстань між спостереженнями, і, оскільки простір стає все більшим і більшим, стає важче знайти справжніх сусідів – проблема для KNN, яка іноді помилково сприймає далеке спостереження за близьке. Надання ідеї так само, як гра в шахи на багатовимірному рівні – шахова дошка. Під час гри на класичній 2-D дошці більшість фігур знаходяться поруч, і ви зможете легше виявити можливості та загрози для ваших пішаків, коли у вас є 32 фігури та 64 позиції. Однак, якщо почати грати на тривимірній дошці, наприклад, у тих, що є у деяких науково-фантастичних фільмах, 32 штуки можуть загубитися на 512 можливих позиціях. Тепер можна уявити, як грати з 12-D шаховою дошкою. Можна легко неправильно зрозуміти, що поблизу, а що далеко. Це і відбувається з KNN. Все ще можна зробити KNN розумним у виявленні подібності між спостереженнями, видаливши зайву інформацію та спростивши розмірність даних, використовуючи методи зменшення даних. Отже, основним недоліком KNN є складність пошуку найближчих сусідів для кожної вибірки.

Що слід пам’ятати:

- обирати непарне значення k для двокласної задачі;
- k не повинно бути кратним кількості класів.

На Рисунку 1.2 показано рішення задачі класифікації для деякого набору даних.

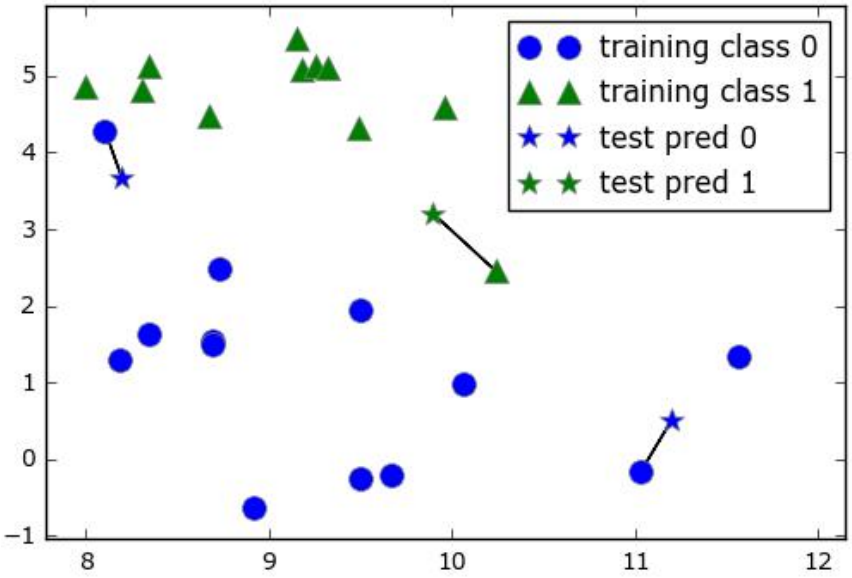

Рисунок 1.2 – Прогнози, отримані для набору даних за допомогою моделі одного найближчого сусіда

Приклад коду із вибором одного найближчого сусіда:

In:

```
mglearn.plots.plot_knn_classification (n_neighbors = 1)
```

Тут додано три нові точки даних, показані у вигляді зірочок. Для кожної відзначено найближчу точку навчального набору. Прогноз, який дає алгоритм одного найближчого сусіда – мітка цієї точки (показана кольором маркера). Замість того, щоб враховувати лише одного найближчого сусіда, можна розглянути будь-яку кількість (k) сусідів. Звідси і походить назва алгоритму k-найближчих сусідів. Коли розглядається більше одного сусіда, для присвоєння мітки використовується голосування (voting). Це означає, що для кожної точки тестового набору підраховується кількість сусідів, що відносяться до класу 0, і кількість сусідів, що відносяться до класу 1. Потім треба присвоїти точці тестового набору клас, що найчастіше зустрічається: іншими словами, вибирається клас, який набрав більшість серед k-найближчих сусідів. У прикладі, наведеному нижче (Рисунок 1.3), використовуються три найближчих сусіда:

In:

```
mglearn.plots.plot_knn_classification (n_neighbors = 3)
```

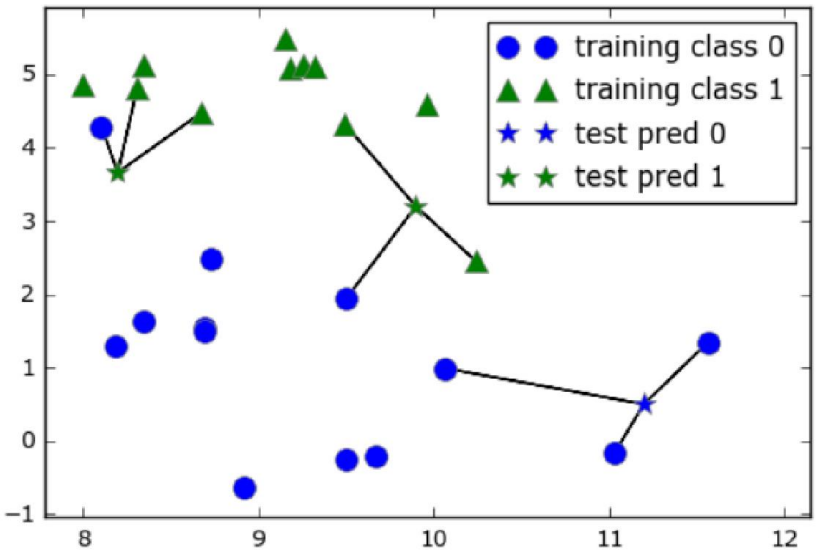

Рисунок 1.3 – Прогнози, отримані для набору даних за допомогою моделі трьох найближчих сусідів

І знову прогнози передані кольором маркера. Видно, що прогноз для нової точки даних у верхньому лівому кутку відрізняється від прогнозу, отриманого при використанні одного найближчого сусіда. Хоча даний малюнок ілюструє завдання бінарної класифікації, цей метод можна застосувати до наборів даних з будь-якою кількістю класів. У разі мультикласової класифікації підраховується кількість сусідів, що належать до кожного класу, і знову прогнозується клас, що зустрічається найчастіше. Далі треба подивитися, як можна застосувати алгоритм k-найближчих сусідів, використовуючи scikit-learn. По-перше, треба розділити вхідні дані на навчальний і тестовий набори, щоб оцінити узагальнюючу здатність моделі:

In:

```
from sklearn.model_selection import train_test_split
```

```
X, y = mglearn.datasets.make_forge ()
X_train, X_test, y_train, y_test = train_test_split (X, y, random_state = 0)
```

Далі виконується імпорт і створюється об'єкт-екземпляр класу, задаючи параметри, наприклад, кількість сусідів, що буде використовуватися для класифікації. В даному випадку він рівний 3.

```
In:
from sklearn.neighbors import KNeighborsClassifier
clf = KNeighborsClassifier(n_neighbors=3)
clf.fit(X_train, y_train)
```

Щоб отримати прогнози для тестових даних, викликається метод *predict.* Для кожної точки тестового набору він обчислює її найближчих сусідів у навчальному наборі ті знаходить серед них клас, що найчастіше зустрічається.

```
In:
print("Прогнозинатестовомунаборі:
{}".format(clf.predict(X_test)))
```

Для оцінки узагальнюючої здатності моделі викликається метод *score* з тестовими даними й тестовими мітками:

```
In:
print ("Правильністьнатестовомунаборі: {: .2f}". format (clf.score (X_test, y_test)))
Out [16]:
Правильністьнатестовомунаборі: 0.86
```

Як видно, модель з прикладу має правильність 86%, тобто модель правильно передбачила клас для 86% прикладів тестового набору.

1.5.3 Візуалізація даних за допомогою Matplotlib. Технології науки про дані накладають необхідність переходу від відносно простих графіків до багатогранних реляційних карт.

Візуалізація даних грає важливу роль зараз і в майбутньому у застосуванні багатьох концепцій. Побудова графіків, а також статична чи інтерактивна візуалізація – одні з найважливіших задач аналізу даних. Вони можуть бути частиною процесу дослідження, наприклад, застосовуватися для виявлення викидів, визначення необхідних перетворень даних чи пошуку ідей для побудови моделей. Більшість людей краще візуалізують інформацію, коли бачать її у графічному, а не текстовому форматі. Графіка допомагає людям бачити відношення та з легкістю порівнювати. Навіть, якщо ви можете з легкістю впоратися з абстрагуванням текстових даних, виконання аналізу даних – це лише спілкування. Якщо ви не можете передати свої ідеї іншим людям, акт отримання, формування та аналізу даних має невелику цінність, окрім ваших особистих потреб. На щастя, Python робить завдання перетворення ваших текстових даних у графіку відносно легкою за допомогою MatPlotLib, який насправді є імітацією програми MATLAB. Вони обидва використовують одні й ті ж завдання, і вони мають подібний метод визначення графічних елементів. *Matplotlib* – це пакет для побудови графіків поліграфічної якості. Для цього пакета є цілий ряд додаткових бібліотек, наприклад *mplot3d*для побудови трьохвимірних графіків та *basemap*для побудови карт та проекцій. Графіки у цьому пакеті знаходяться всередині об’єкта малюнка *Figure.* Створити новий малюнок можна методом *plt.figure*:

```
In:
fig = plt.figure()
```

Головна функція *matplotlibplot*– приймає масиви координат X та Y, а також необов’язкову, в якій закодовані колір та стиль лінії. Повний перелік типів графіків є в документації по *matplotlib.*

1.6 Вирішення задачі класифікації за допомогою STATISTICA10.0

В задачі класифікації мережа повинна віднести кожне спостереження до одного з декількох класів. У пакеті STNeuralNetworksдля класифікації використовується номінальна вихідна змінна – різні її значення відповідають різним класам. У STNeuralNetworks класифікацію можна здійснювати за допомогою мереж наступних типів: багатошарового персептрона, радіальної базисної функції, мережі Кохонена, ймовірностей нейронної мережі та лінійної мережі. У цій програмі функції класифікації обчислюються для кожної сукупності та можуть безпосередньо брати участь у класифікації об’єктів.

Спостереження слід відносити до тієї сукупності, для якої обчислено найбільшу класифікаційну вагу. Можна використовувати ці функції для того, щоб визначити перетворення для нових змінних. Коли введено нове спостереження, програма автоматично обчислює класифікаційну вагу для кожної сукупності.

У програмі STATISTICA10.0 та інших версіях зручно аналізувати табличні дані. Наприклад,одниміз способівперевіркиєпорівнянняспостережуванихзначеньіпередбаченихрезультатів.Порівнянняспостережуванихіпередбачених значеньдляобраноїмережі,наприклад,длянавчальноїтатестовоївибірок.

По своїй задумці система STATISTICAволодіє широкими графічними можливостями, включаючи створення наукових, ділових, трьохвимірних та двовимірних графіків в різних системах координат. В системі вбудований язик програмування STATISTICABASIC. Він дає можливість генерувати різні часові ряди для подальшого аналізу та перевірки точності алгоритмів. Програма організована за модульним принципом, які містять описові статистики, методи статистичного аналізу різних таблиць, різносторонній інструментарій для проведення дослідного аналізу даних. Процедура прогнозування дозволяє

проводити аналіз типу «що-якщо» та інтерактивно обчислювати передбачені значення по заданим з клавіатури значенням предикторів.

РОЗДІЛ 2
ПРОЕКТНІ РІШЕННЯ

На сьогоднішній день існує багато готових інструментів для вирішення задач, подібних до тих, що я спробую вирішити у своїй роботі. Та ці способи вирішення завдань зазвичайабо єуніверсальними, або кожна організація підлаштовує під себе або комбінує різні способи аналізу продажів товарів з метою збільшення прибутку та оптимізації своєї роботи. Можливо, якісь фірми використовують і машинне навчання. Тому, порадившись з керівництвом та фахівцями в цій області, я запропонувала їм спробувати вирішити проблеми компанії. Ми поєднаємо відомі методи аналізу продажу, детальний опис кожної одиниці товару та алгоритми машинного навчання.

На перший погляд може здатися, що вирішення даної задачі підходить лише для моєї організації, але воно однозначно підходить і для подібних компаній, потрібно лише, згідно моєї готової роботи, поетапно відтворити такі самі дії.

2.1 Створення звіту продажу товарів

Для того, щоб виконати практичну частину роботи, я спочатку зробила у програмі своєї компанії звіт «продаж товарів» за останні три роки (липень 2018р. – липень 2021р.). Звіт складається з таких стовпців:

- артикул;
- найменування;
- бренд (Brand);
- вага (л) (Weight);
- кількість одиниць (шт);
- ціна з ПДВ (грн);
- сума з ПДВ (грн);

- валовий прибуток (грн);
- рентабельність (%).

2.2 Формування команди фахівців та створення експертної таблиці

Потім я зібрала команду фахівців, що працюють в нашій компанії, для подальшої роботи. Окрім мене до цієї групи ввійшли такі учасники: представник відділу закупівель, бізнес-аналітик, експерт відділу виробництва.Разом ми провели змістовний аналіз проблемної області.Ми вирішили, що визначення характеристик кожної позиції товару зменшить надалі проблеми із закупівлями та підбором подібного товару. Це дасть змогу глибше оцінити характеристики позицій, що мають вплив на попит покупців. Ми обрали властивості олив, які показують якість, специфіку застосування, технічні особливості тощо. Це дозволяє відрізнити майже пів тисячі найменувань одне від одногозначно краще, ніж просто інформація, що міситься в короткій назві товару. До цих характеристик олив ми вирішили додати такі стовпці:

- температура застигання (Pour_Point);
- точказаймання (OC_Flash_point);
- індекс в'язкості (Viscosity_Index);
- лужне число (Alkaline_Number);
- кінематична в'язкість при 100 ºС (KinematicViscosity);
- застосування (легкові / грузові авто) (Applicability);
- рівень допуску за класифікацією ACEA (Tolerance);
- застосування (двигун бензиновий/дизельний) (Petrol);
- в'язкість(Viscosity_Grade_1 та Viscosity_Grade_2);
- хімічний склад оливи (Composition).

На даному етапі зі всієї кількості стовпців, що мали в таблиці, ми видалили артикул, найменування та рентабельність, щоб зручніше було

проводити подальшу обробку. Детально вивчивши методи аналізу продажу товарів, ми обрали такий, що найбільше підходить до специфіки роботи нашої фірми. Використовуючи ABC-аналіз,разом з фахівцем з відділу закупівель, зробили класифікацію даних у таблиці згідно трьох етапів, що описані у теоретичній частині.Зважаючи на те, що при виборі одного будь-якого критерію для класифікації не можливо дати комплексну оцінку аналізованого елементу (товару), ми скористались механізмом послідовного перебору релевантних ознак. Спочатку диференціювання номенклатури виконали за першою ознакою – валовий прибуток (грн), тому що прибуток компанії – це одна з найголовніших цілей. Для цього відсортували таблицю у порядку спадання за цією ознакою, а потім отримані групи (A, B і C) диференціювали за другою ознакою – кількість проданих одиниць. В результаті отримано таблицю, в якій кожен товар має свій клас – де 1-й клас –це найрентабільніший для компанії, 2-й – менш важливіший, а 3-й клас відповідно має найменші і прибуток, і кількість проданих одиниць.

Наступним етапом нашої роботи став збір інформації обраних нами характеристик технічних олив. Для цього ми взяли дані з сайтів продажу олив. Ми використовували переважно офіційні сайти та технічні експертизи від постачальників, що маємо в наявності.

На етапі формалізації було обрано інструментальнимизасобами програмні комплекси STATISTICA 10.0 та Python 3.9. По черзі проведемо навчання в обох програмах та порівняємо результати.

В ході етапу тестування провели оцінку обраного способу представлення знань в своїй експертній системі в цілому. Для цього підібрали приклади, що забезпечують перевірку всіх можливостей нової ЕС, тестували якість та коректність інформації, яка видається ЕС на виході, повноту бази знань та інші фактори роботи системи.Етап пройшов успішно – всі дані зібрані в повному обсязі та коректно.

2.3Проведення навчання у програмі Python 3.9

Перед початком роботи з класифікованим файлом, ми видалили з нього зайві позиції, які потрібні були лише для проведення класифікації, а для подальшого навчання вони будуть лише заважати. Це – ціна (з ПДВ), сума (з ПДВ), валовий прибуток (грн), рентабельність (%). Варто зауважити, що спочатку ми видалили ще й стовпці – бренд та об'єм, помилково вважаючи, що це не впливає на популярність товарів. Та, провівши попереднє тестове навчання у програмі Statistica 10.0, побачили результат навчання близько 63%, що є зовсім неприйнятним. Проаналізувавши це, повернули до таблиці стовпці – бренд та об'єм та отримали майже ідеальний попередній результат, про який буде описано далі у моїй роботі.

2.3.1 Завантаження файлу з даними. Виведення на екран даних з цього файлу. Щоб розпочати процес навчання, треба завантажитинаш файл у Python 3.9.1. Це можна зробити за допомогою методу бібліотеки *pandas* зчитування файлів формату .xlsx – pd.read_excel(‘шлях до файлу\назва_файлу.xlsx’) аналогічно для зчитування файлів формату .csv– pd.read_csv(‘шлях до файлу\назва_файлу.csv’). Надалі в роботі буде показано,як завантажувати файли формату .txt. Потім,за допомогою програмного коду, що на Рисунок 2.1, вивела на екран значення ключів та зміст деяких з них(Рисунки 2.2, 2.3 та 2.4).

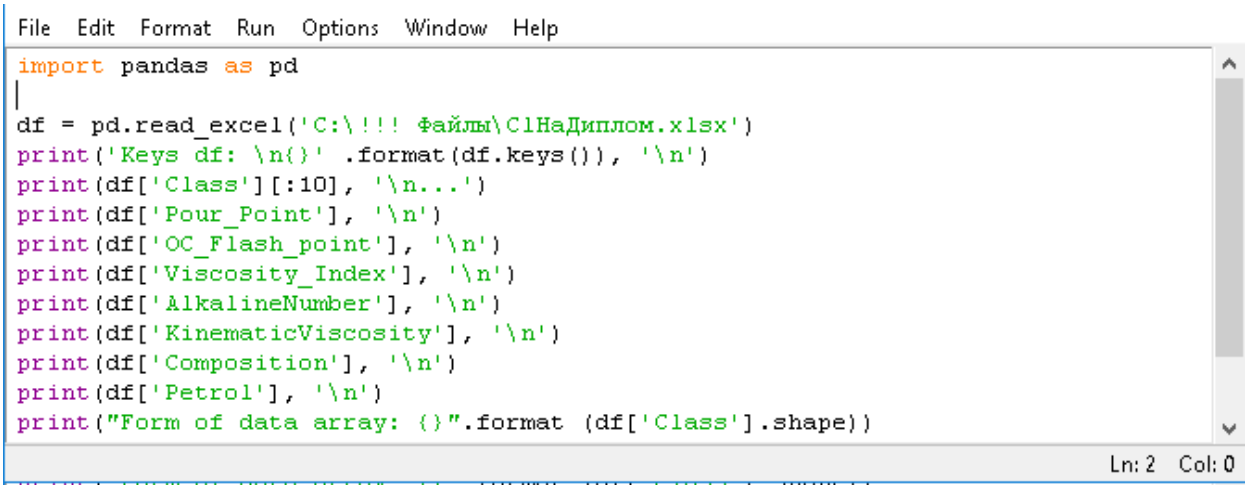

Рисунок.2.1 – Програмний код виведення даних файлу на екран

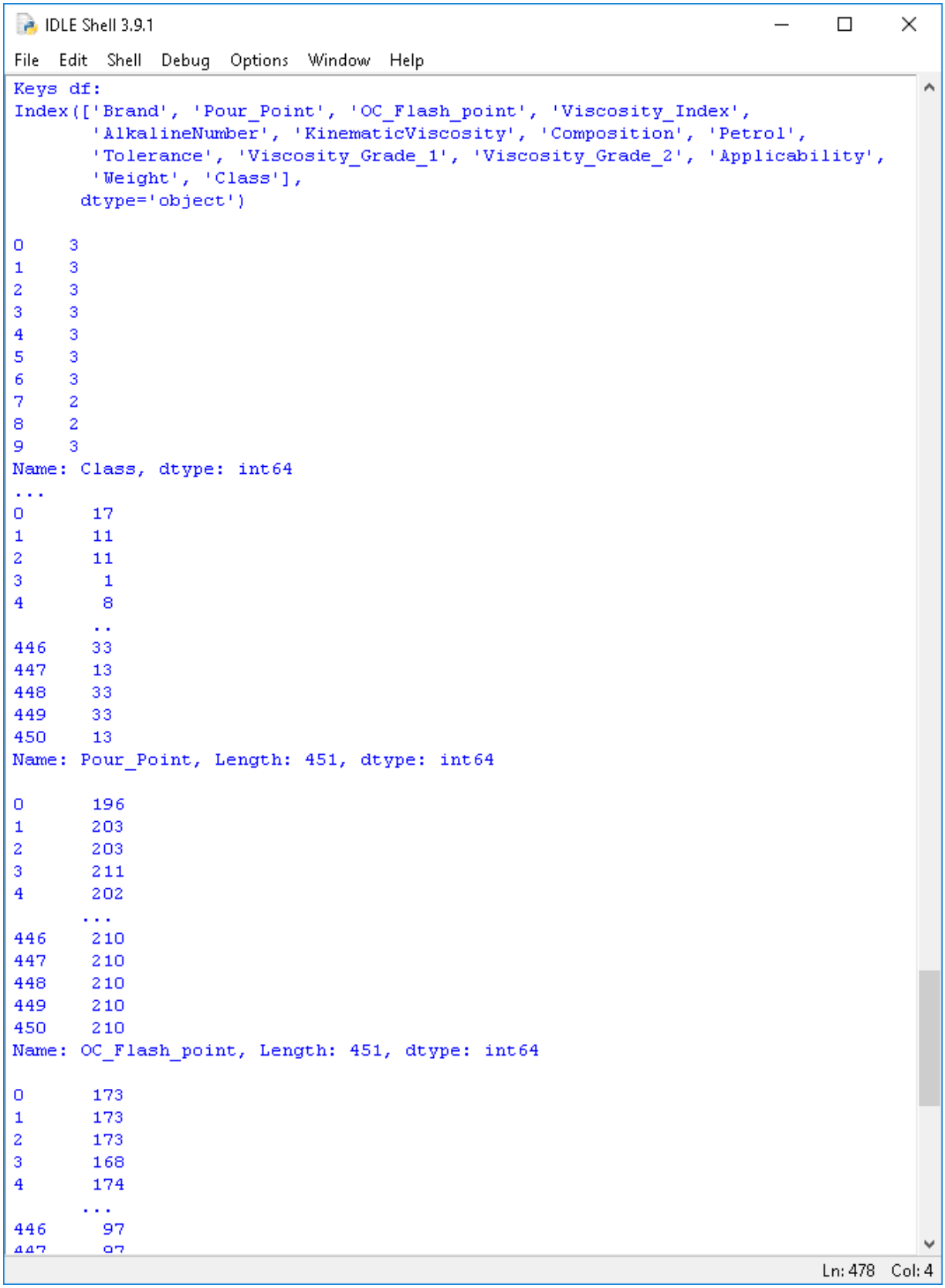

Рисунок 2.2 – Виведення на екран значення ключів та уривки зі стовпців з даними

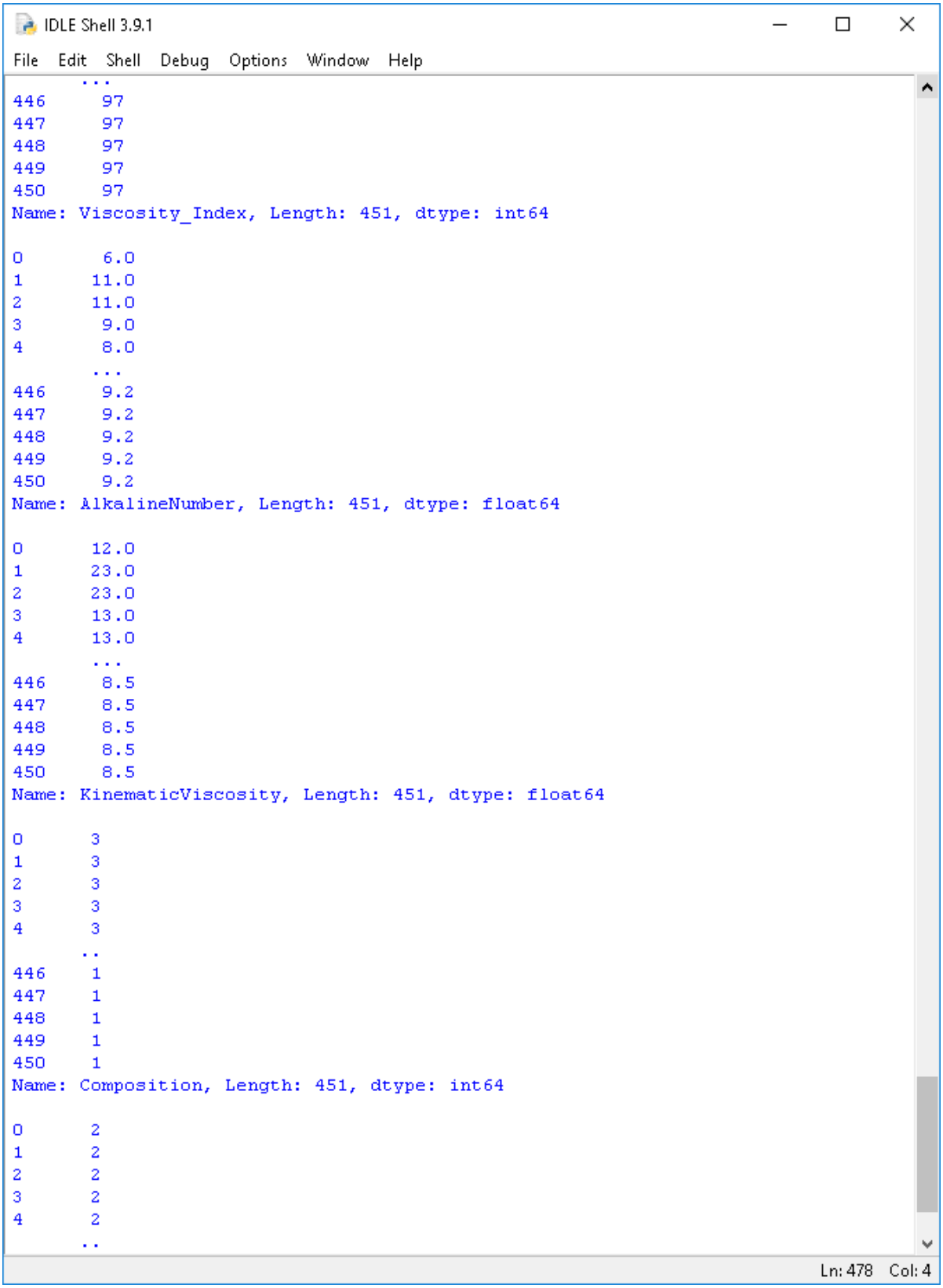

Рисунок 2.3 – Виведення на екран уривків зі стовпців з даними

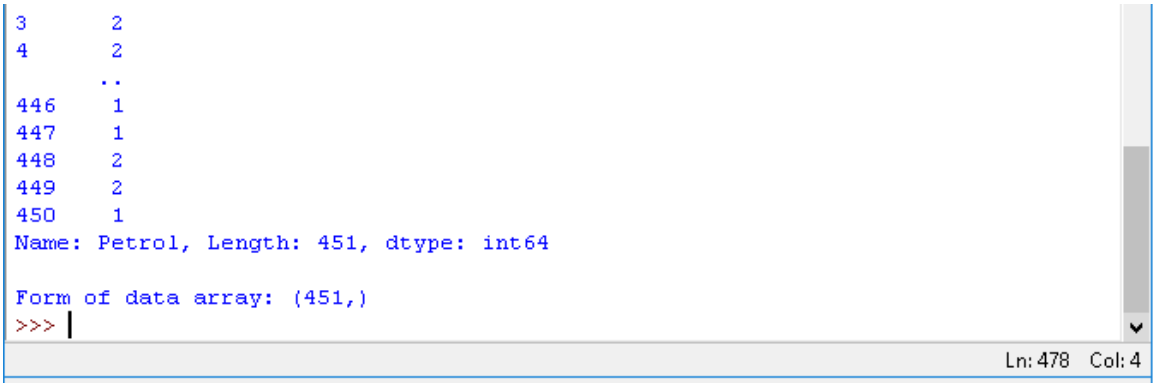

Рисунок 2.4 – Виведення на екран уривків зі стовпців з даними

Наступним кроком роботи є розділення робочої таблиці на 2 частини: основну та тестову. Зроблено це для того, щоб провести у програмі навчання з основною частиною, а потім перевірити, чи працює алгоритм із тестовою частиною.

Проводимо навчання за допомогою методу k-найближчих сусідів (Рисунок 2.5), змінюючи їх кількість та спостерігаючи, як змінюється результат навчання на Рисунках 2.6– 2.8. Як видно з рисунків – найкращий результат Samples≈ 0,83157отримано при knn= 1, а при збільшенні кількості сусідів –knn = 2 маємо результат трохи менший за попередній Samples ≈ 0,82105, далі при knn = 4Samples ≈ 0,8.

Ті самі дії проведені й для другого набору даних – тестовою частиною. Проаналізуємо отримані результати навчання. Так само проведено декілька разів навчання з різною кількістю сусідів (Рисунки 2.9– 2.12). При knn = 1та при knn = 2 отримано однакове значення Samples ≈ 0,90909, при knn = 3 результат навчання ≈ 0,81818, а надалі ще менше ≈ 0,77272. Можна зробити висновки, що не потрібно збільшувати кількість сусідів, адже маємо відмінний результат і при малій кількості сусідів так само, як і з основною частиною нашої вибірки. Звернемо увагу на матрицю ConfusionMatrix – вона показує зовсім незначні відхилення, а з тестовим набором даних взагалі майже ідеальний вигляд цієї матриці з рідкисними випадками відхилення. Аналізуючи

показникClassificationReport (стовпець precision), бачимо, що прикількості сусідів = 2 у тестовому наборі, найкраще передбачаються значення, а для 2 та 3 класу взагалі дорівнюють 1. Всі інші показники також задовільняють нас.

File Edit Format Run Options Window Help

```
import numpy as np
import pandas as pd
import mglearn

input_file = 'DIPLOM17test.txt'
data = np.loadtxt(input_file)
X, y = data [:, : -1], data[:, -1]
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, y, random_state=0)

print('Fragment of learning sample', end='\n')
for i in range(6):
    print(X_train[i], y_train[i], end='\n')

from sklearn.neighbors import KNeighborsClassifier
knn=KNeighborsClassifier(n_neighbors = 4)
knn.fit(X_train, y_train)

X_new=np.array(X_test[:2])
X_new

prediction = np.array (knn.predict(X_new))
print ('Method k-mean for X_test[0] and X-test[1]', prediction)
print ('Label from test simple', y_test[:2])

y_pred = knn.predict(X_test)

from sklearn.metrics import classification_report, confusion_matrix, accuracy_score
result = confusion_matrix(y_test, y_pred)
print("Confusion Matrix:")
print(result)
result1 = classification_report(y_test, y_pred)
print("Classification Report:",)
print (result1)
result2 = accuracy_score(y_test,y_pred)
print("Samples:",result2)
```

Ln: 1 Col: 0

Рисунок 2.5 – Програмний код алгоритму k-найближчих сусідів

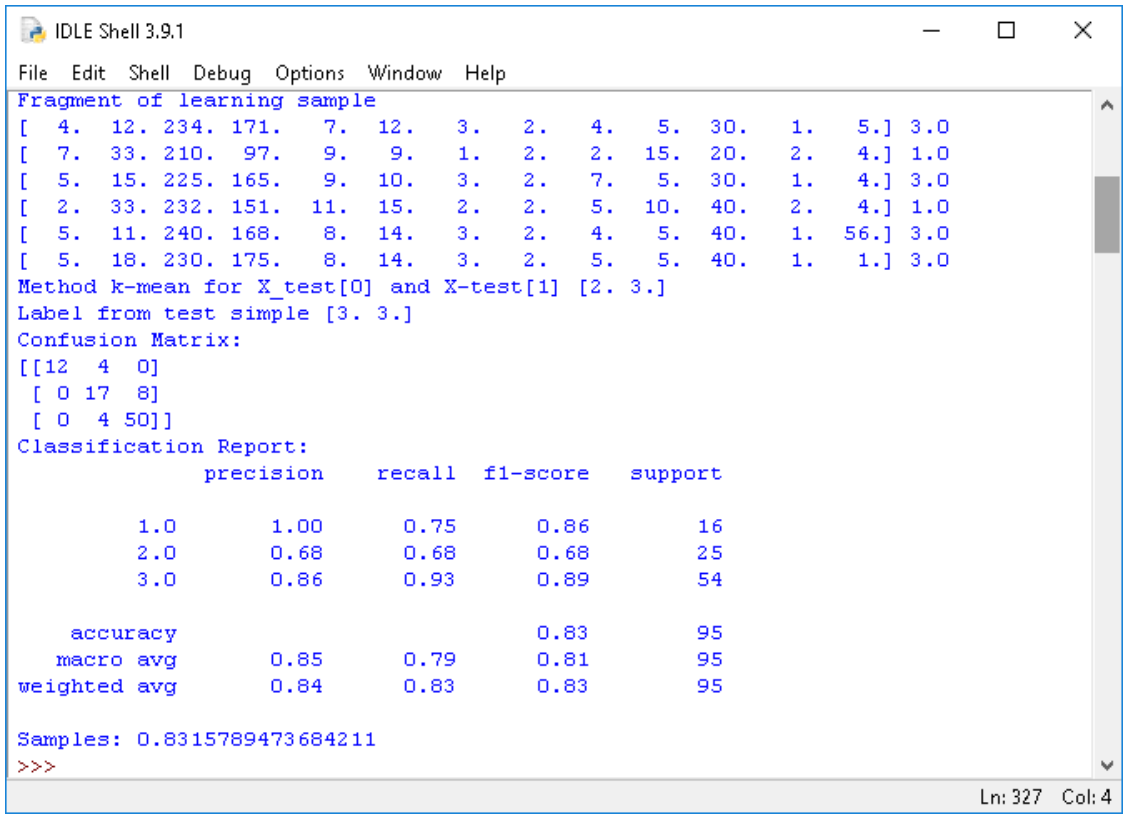

Рисунок 2.6 – Результат роботи алгоритму при knn=1

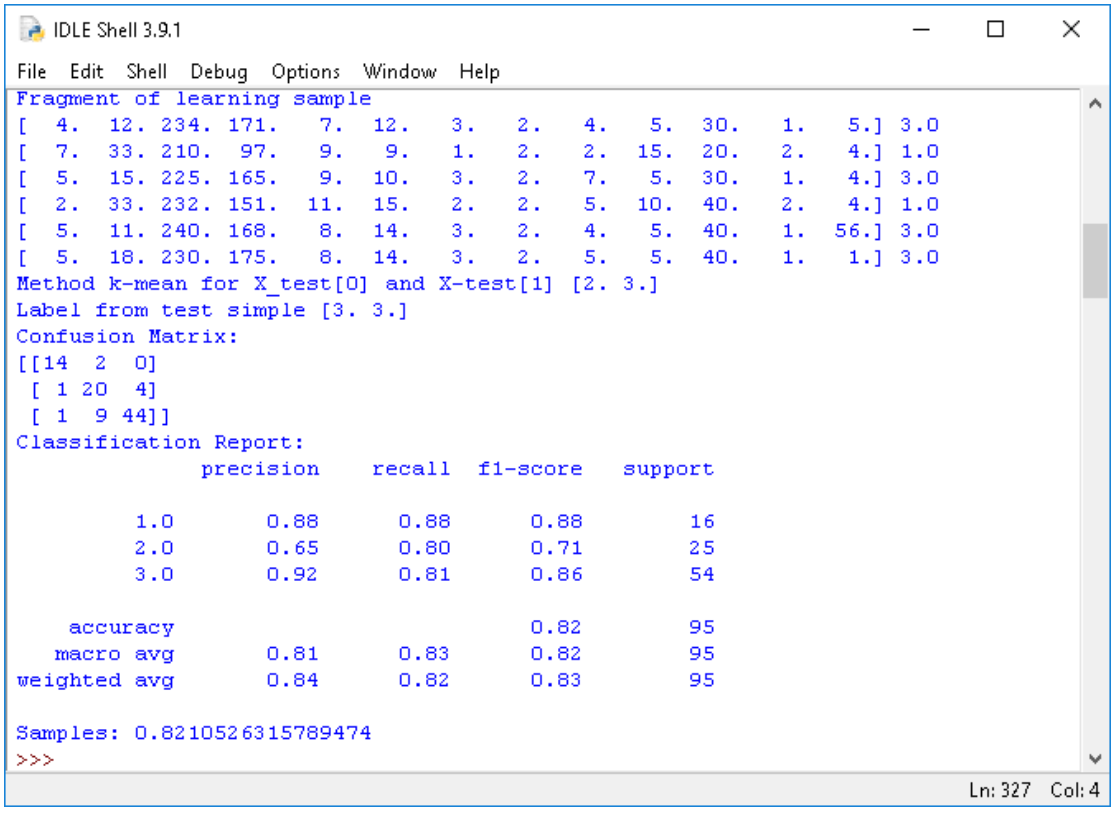

Рисунок 2.7 – Результат роботи алгоритму при knn=2

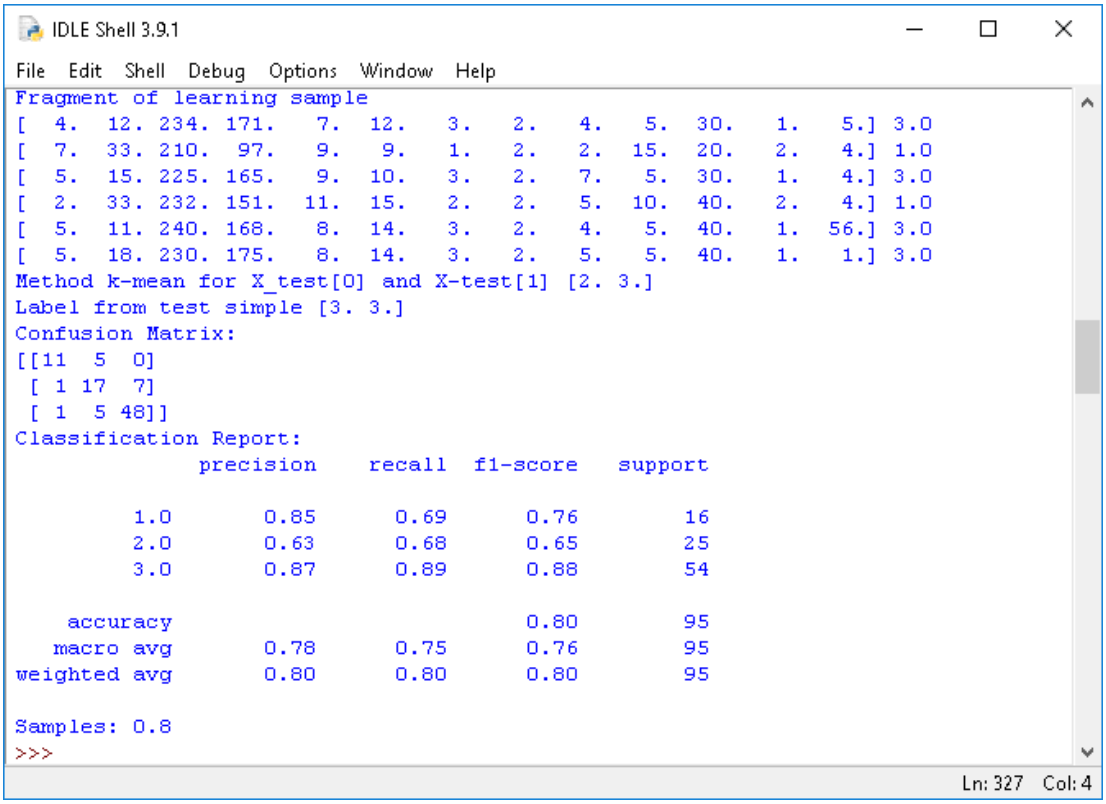

Рисунок 2.8 – Результат роботи алгоритму при knn=4

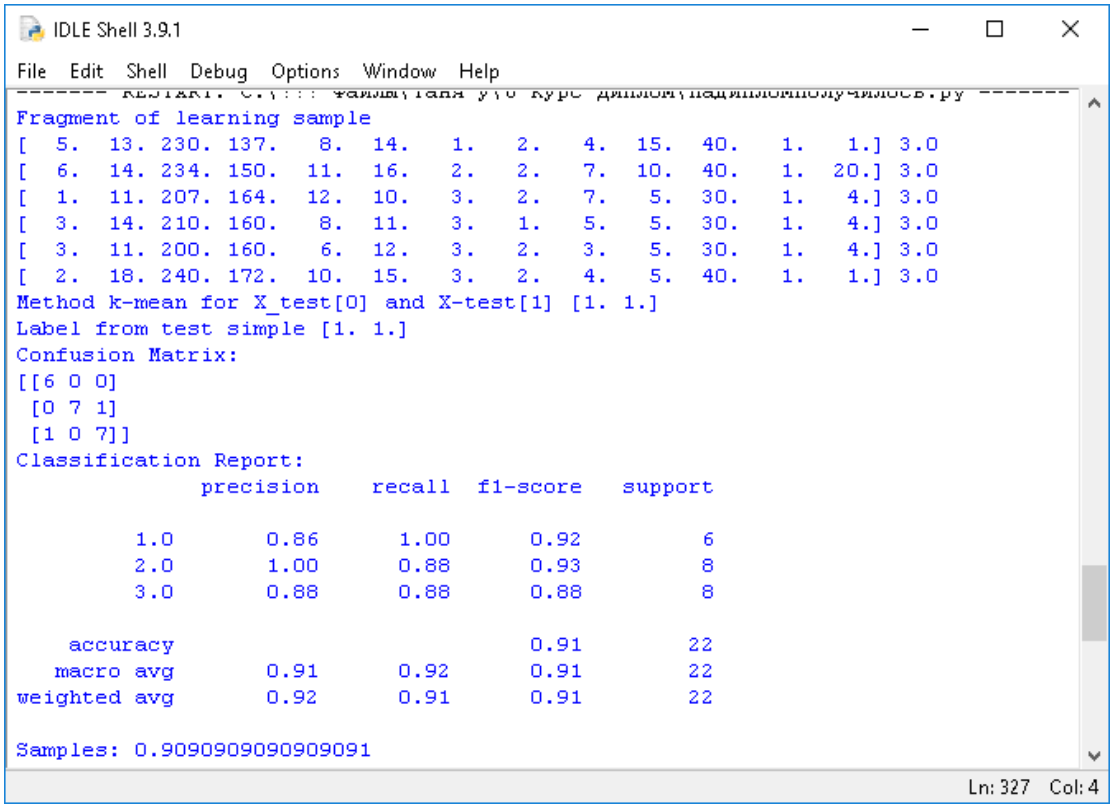

Рисунок 2.9 – Результат роботи алгоритму при knn=1 (з тестовою частиною)

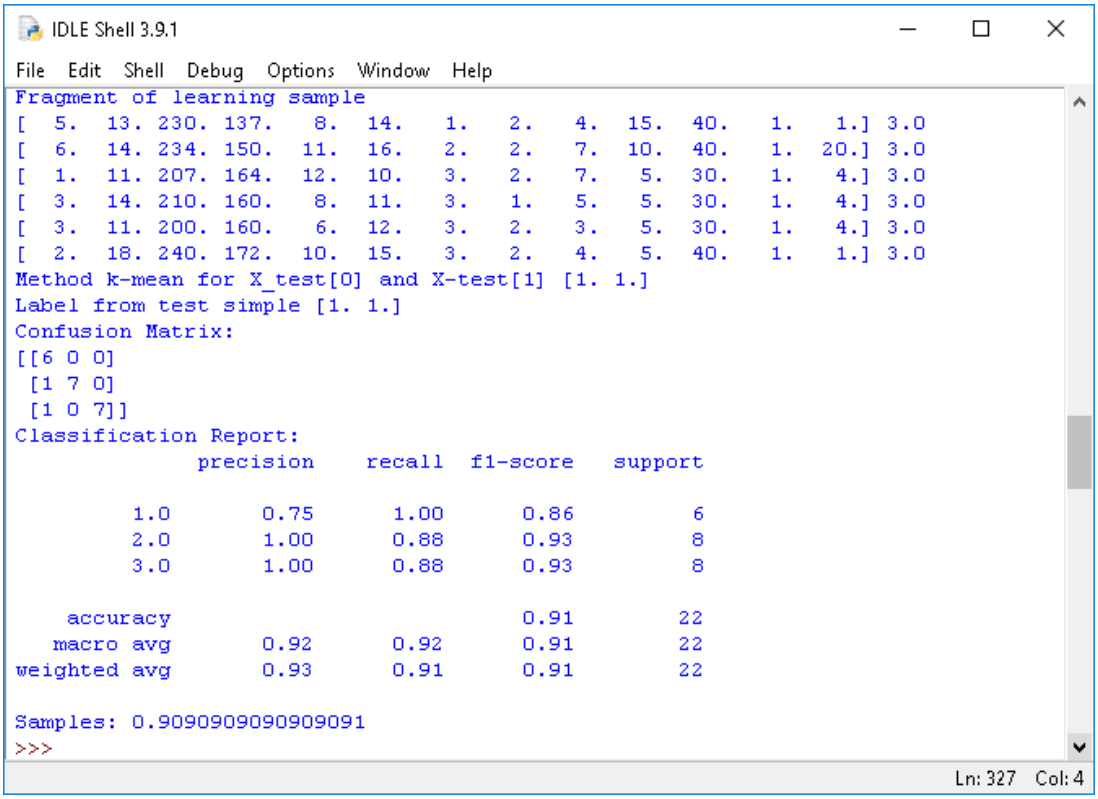

Рисунок 2.10 – Результат роботи алгоритму при knn=2 (з тестовою частиною)

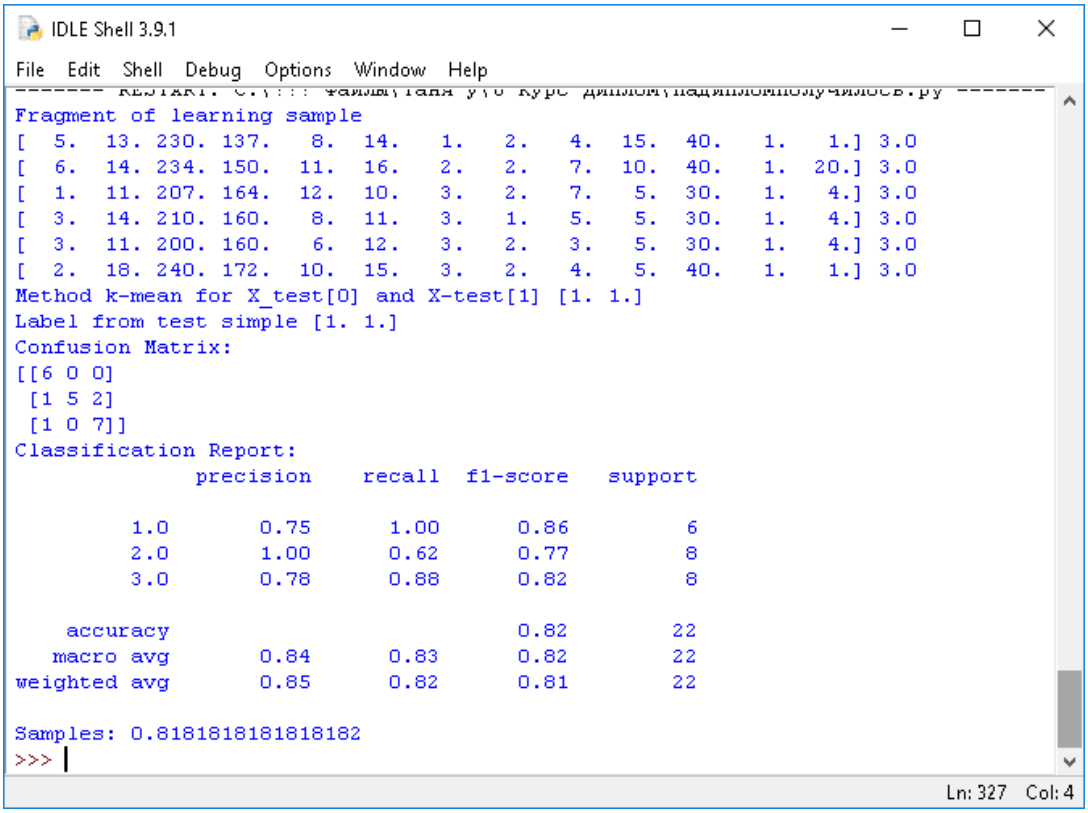

Рисунок 2.11 – Результат роботи алгоритму при knn=3(з тестовою частиною)

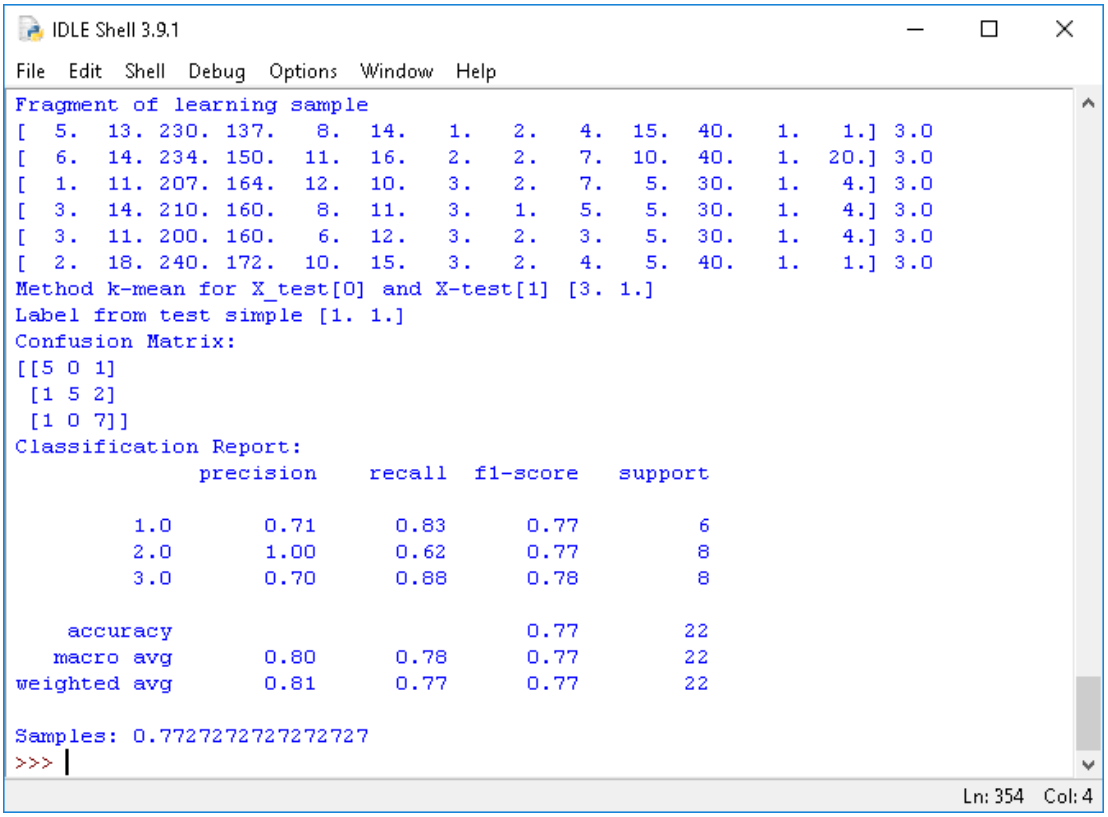

Рисунок 2.12 – Результат роботи алгоритму при knn=4 (з тестовою частиною)

На основі зображених малюнків можна зробити висновки, що алгоритм дає достатньо відмінний результат в обох таблицях при невеликому числі сусідів knn =1 та knn = 2, а при збільшенні кількості сусідів knn =3 та knn = 4 результат стає гіршим. Це є відмінним результатом, що при меншій кількості сусідів маємо кращий результат, ніж при їх збільшенні.

2.3.2 Візуалізація даних за допомогою пакетуmatplotlib. Отже, як вже сказано в теоретичній частині, – візуалізація є невід'ємною частиною аналізу вхідних даних. Це напростіший та зручний спосіб швидко оцінити таблицю вхідних даних, особливо якщо вона є великою за обсягом. На Рисунку 2.13 зображено програмний код для візуалізації всіх даних одночасно. Хоч таблиця є немалою, візуалізація досить добре відображена (Рисунки 2.14 – 2.15).

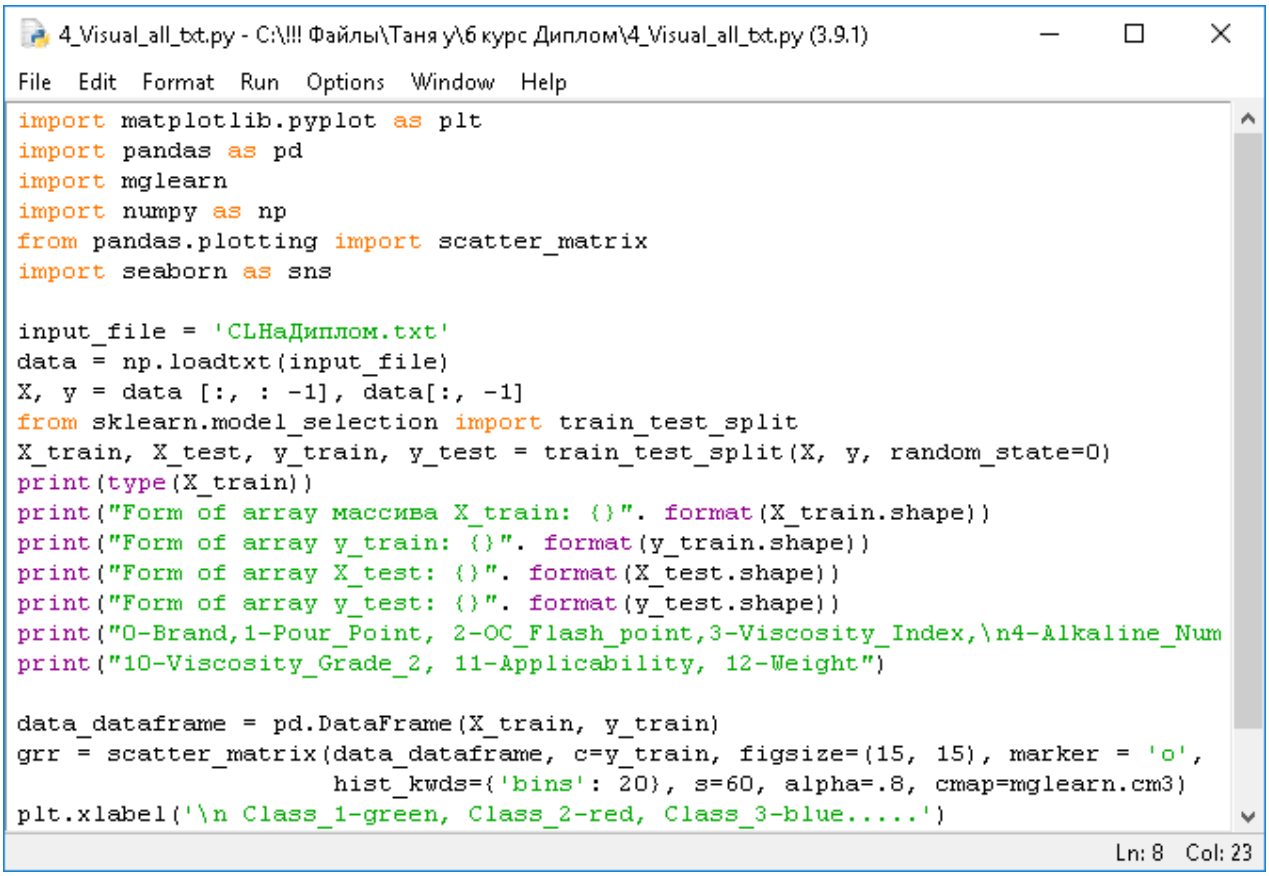

Рисунок 2.13 – Програмний код візуалізації даних (загальний вигляд)

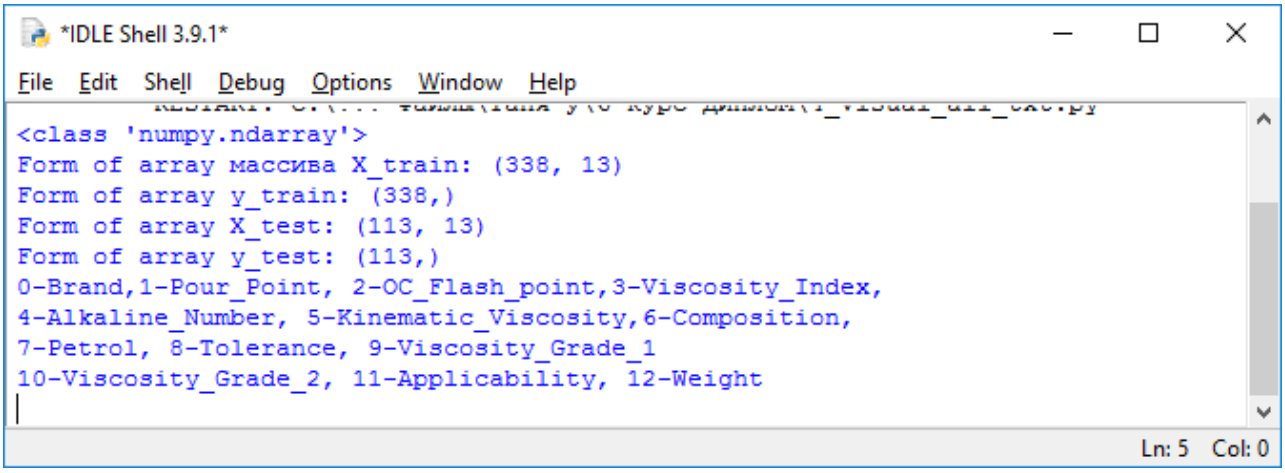

Рисунок 2.14 – Коментарі до візуалізації даних

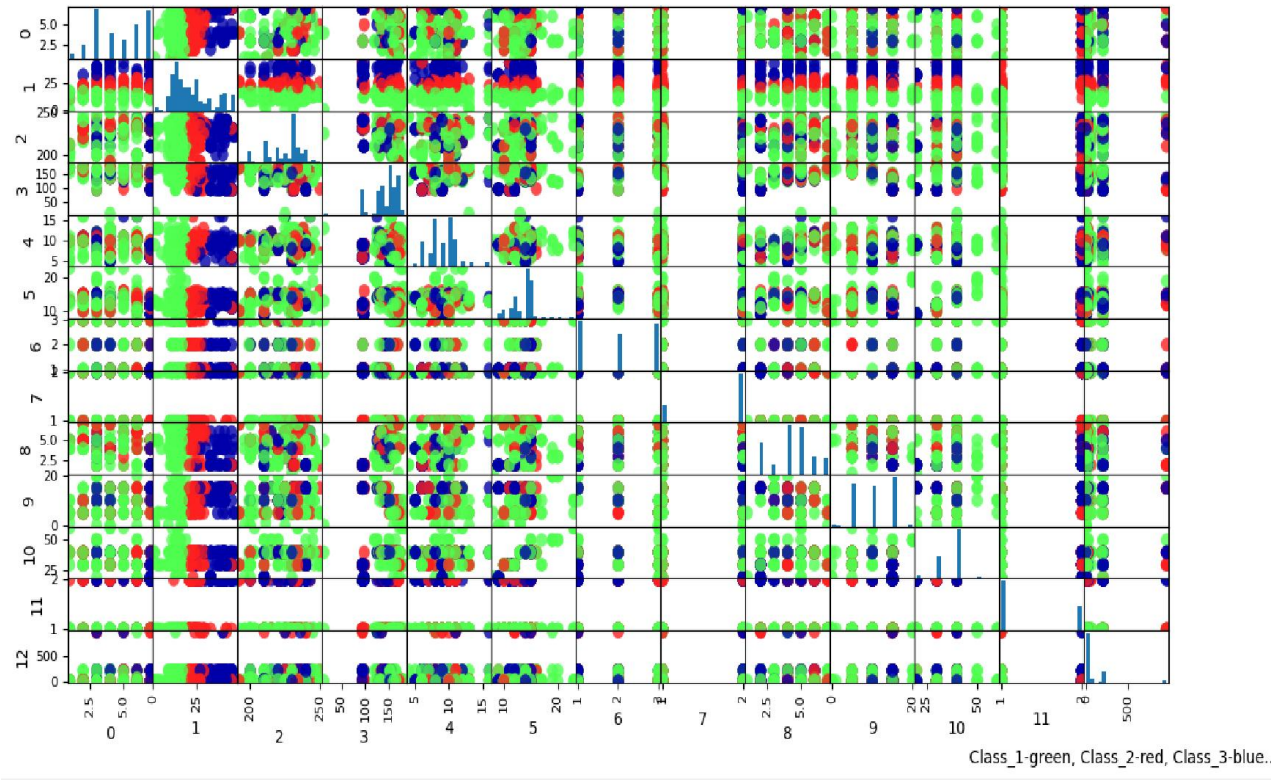

Рисунок 2.15 – Візуалізація даних (загальний вигляд)

Щоб детальніше роздивитися,як розподілилися класи за кожною ознакою, скористаємось ще однією візуалізацією, що зображена на Рисунках 2.16. А результат, отриманий в результаті роботи коду, зображено на Рисунки 2.17 – 2.22.Це значно зручніше, якщо потрібно детально роздивитися, як розподіляється кожна ознака по класам та, особливо, які ознаки впливають на результат навчання – тобто візуально видно класифікацію. Можна відзначити, що, згідно цих Рисунків – 2.17 – 2.22, по показниковіComposition, значення майже не відрізняються, тобто ці характеристики майже не впливають на розподіл за класами. Краще візуально видно розподіл за наступними характеристиками: OC_Flash_point, Viscosity_Index, Alkaline_Number, Brand, Tolerance, Viscosity_Grade_1, Weight. В деяких значеннях по цим характеристикам можна відслідкувати певну закономірність класифікації. Найкраще з усіх стовпців даних класифікуються стовпці, що залишилися: Pour_Point, Viscosity_Grade_2 та Kinematic_Viscosity.

Отже, завдяки візуалізації можна легко зрозуміти, чи достатньо даних для проведення класифікації, як вони розподіляються на графіку та які стовпці можна взагалі видалити через те, що вони не впливають кінцевий результат. Візуалізація є невід'ємною частиною процесу дослідження і для виявлення викидів, визначення необхідних перетворень даних тощо.

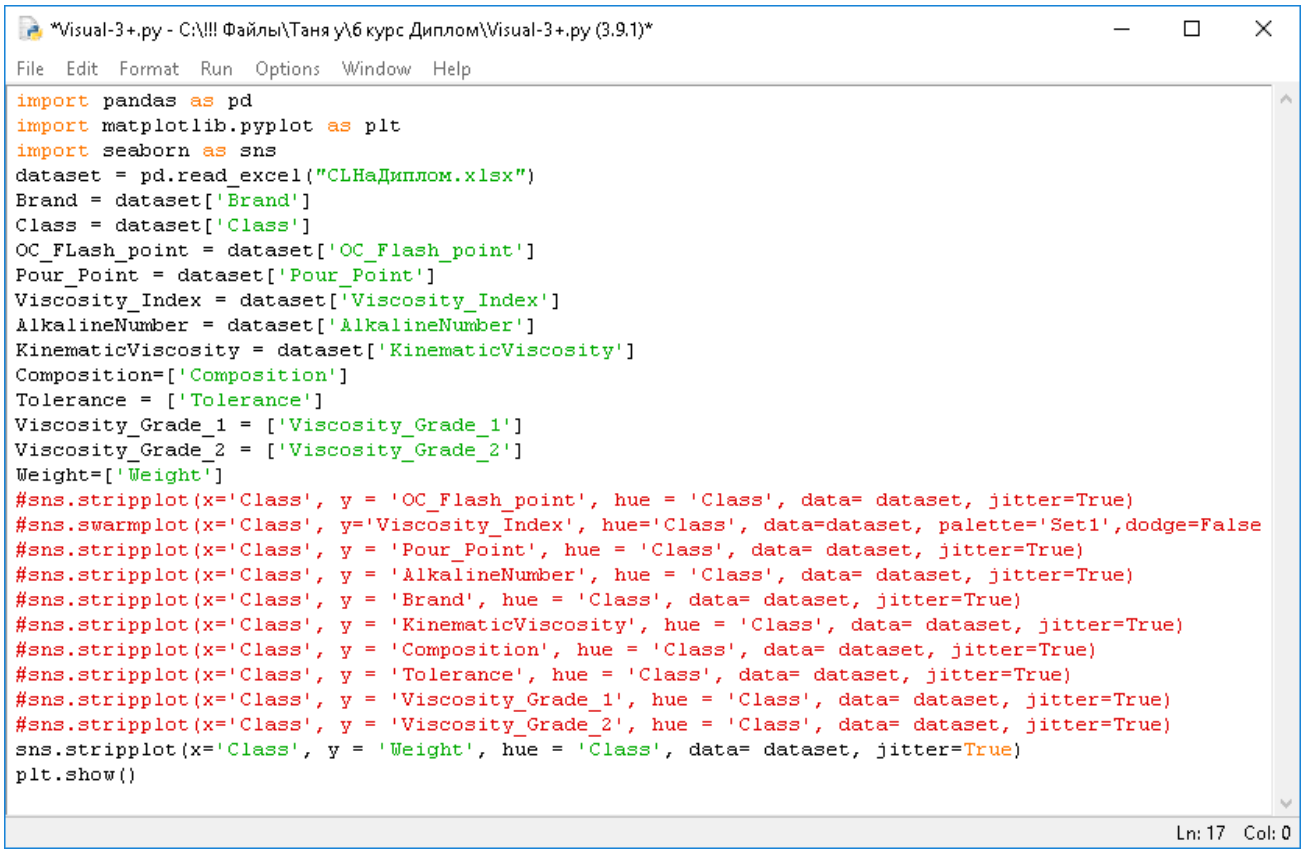

Рисунок 2.16 – Програмний код візуалізації (2)

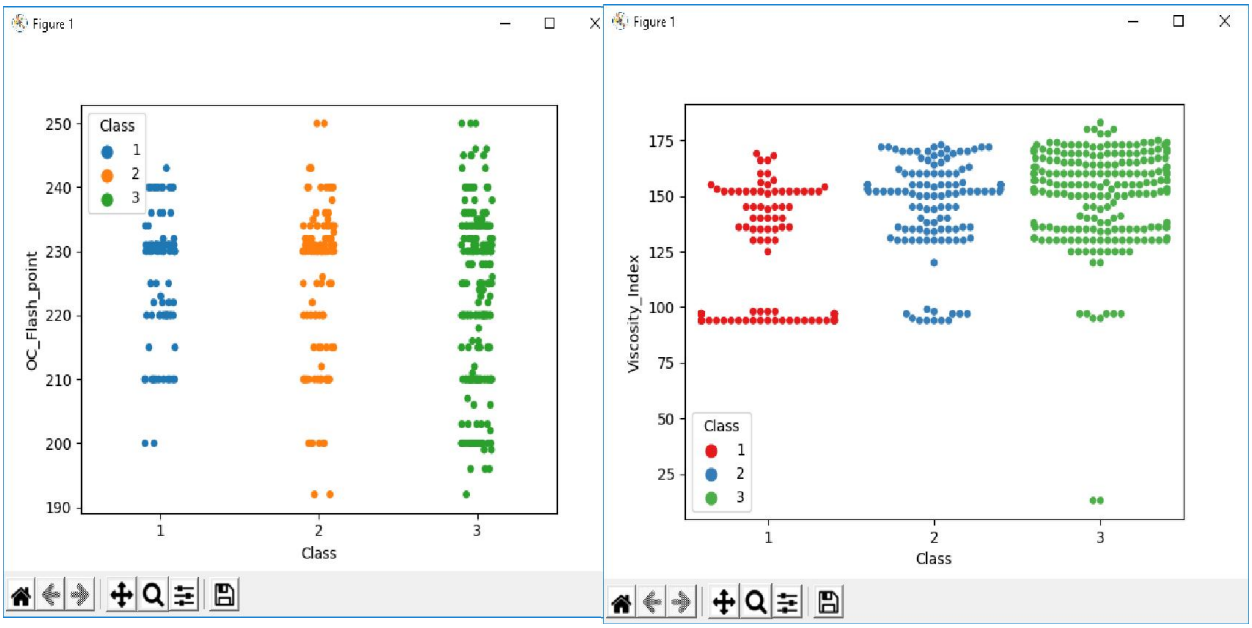

Рисунок 2.17 – Результат візуалізації OC-Flash_Pointта Viscosity_index

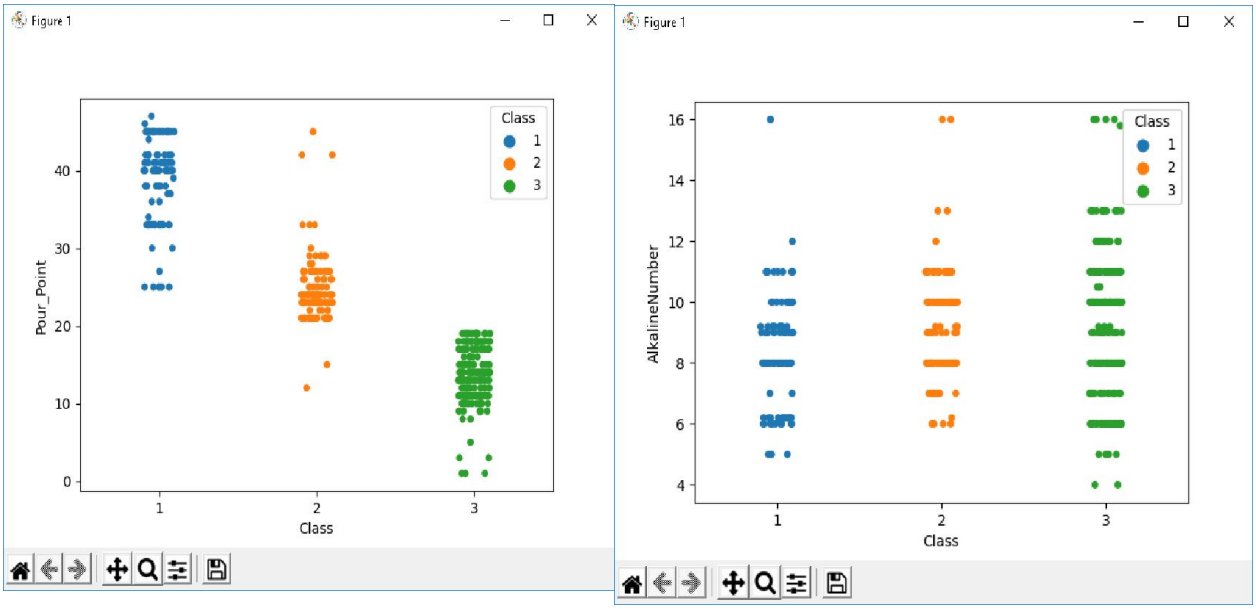

Рисунок 2.18 – Результат візуалізації Pour_Pointта Alkaline_Number

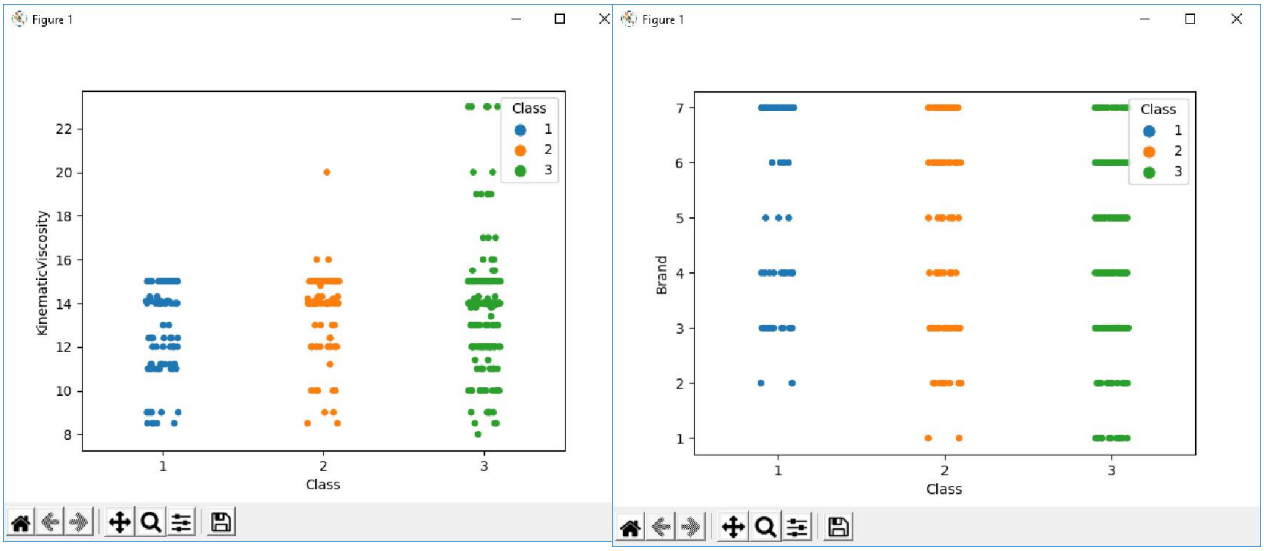

Рисунок 2.19 – Результат візуалізації Kinematic_Viscosity та Brand

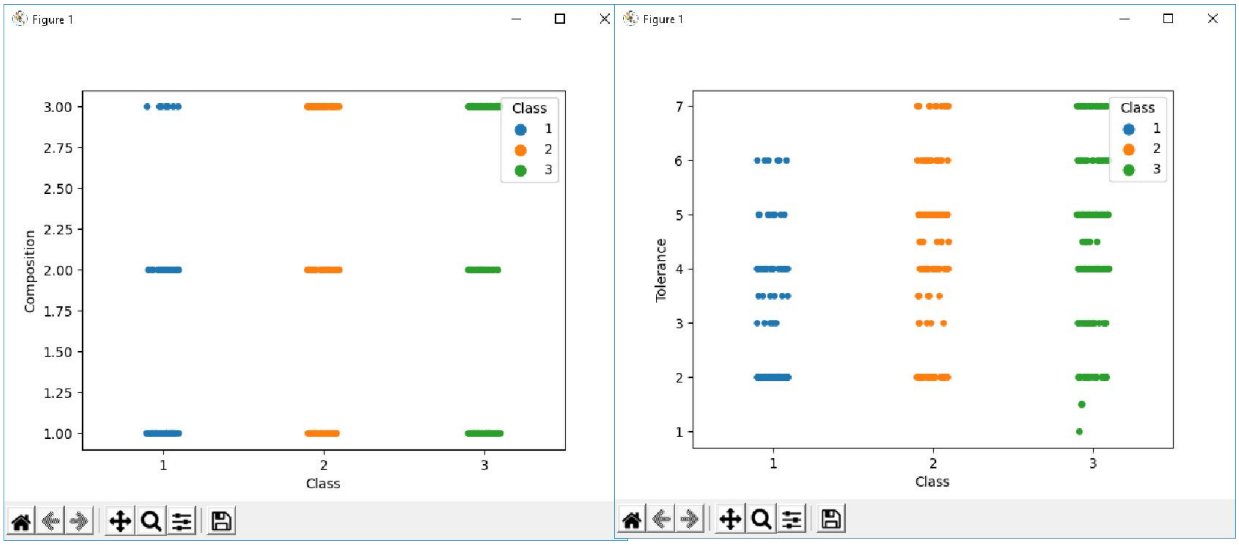

Рисунок 2.20 – Результат візуалізації Composition та Tolerance

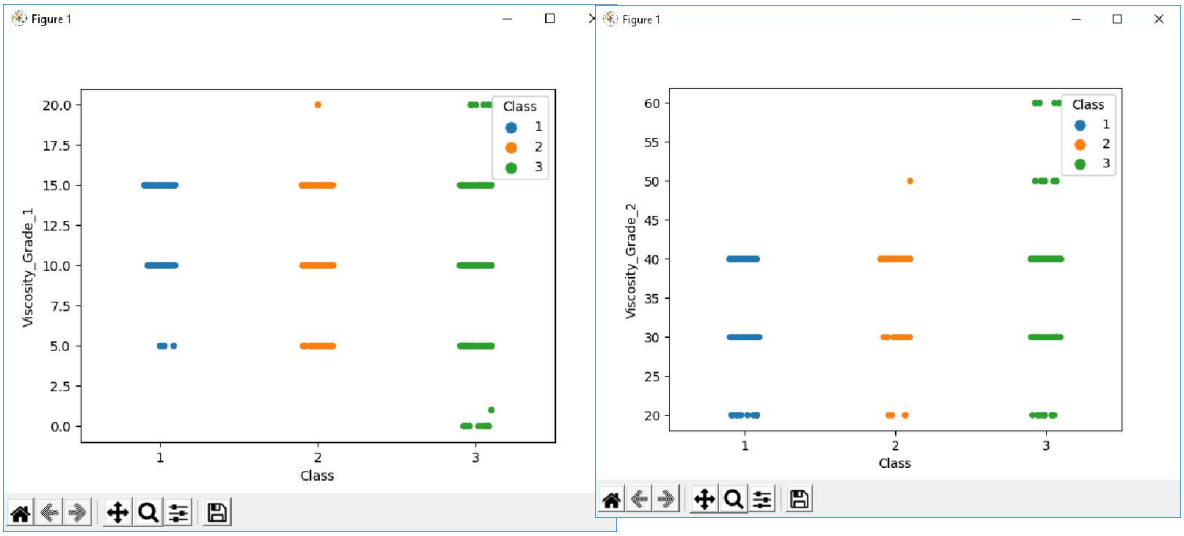

Рисунок 2.21 – Результат візуалізації Viscosity_Grade_1 та Viscosity_Grade_2

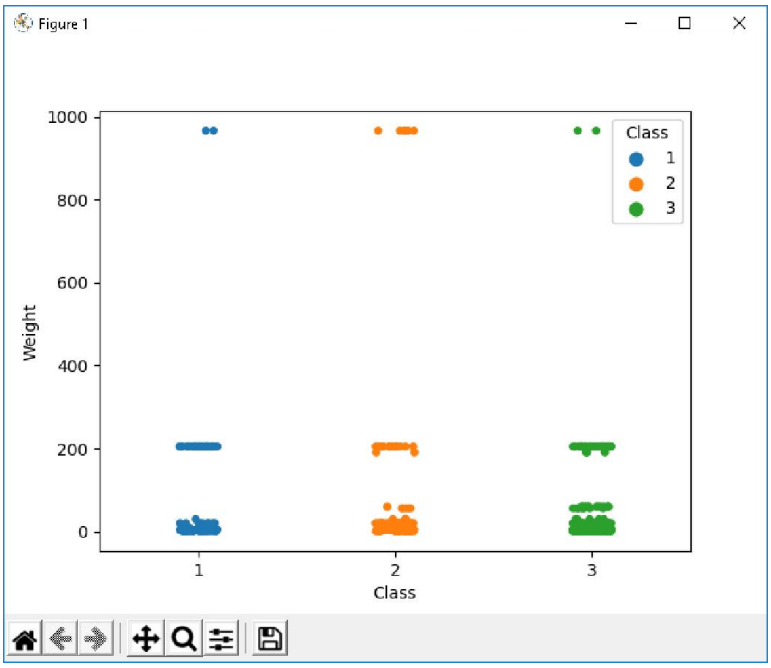

Рисунок 2.22 – Результатвізуалізації Weight

2.4 Проведення навчання у програмі Statistica 10.0

Наступним етапом моєї роботи буде проведення навчання мережі у програмі Statistica 10.0. Для цього спочатку розділила свою вибірку на дві частини: перша частина – основна для навчання мережі, а друга частина – таблиця з декількох позицій для прогнозування класів, щоб перевірити

правильність навчання мережі на раніше невідомих зразках. Фрагмент першої вибірки для навчання зображено на Рисунок 2.23. Потім треба провести наступні дії: запустити модуль Нейронні мережі STATISTICA. Обрати у панелі інструментів «Анализ» – «Нейронные сети» і тип задачі «Классификация», натиснути «ОК» - Рисунок 2.24. Далі, переходимо до задання змінних. Для цього натискаємо кнопку «Переменные». У діалоговому вікні вибрати змінні. В даному прикладі в першій колонці обрати «Категориальная целевая» – *Class* – під номером 14. Потім в третій колонці «Категориальная входная» обрати змінні – з 1 по 13 (Рисунок 2.25).

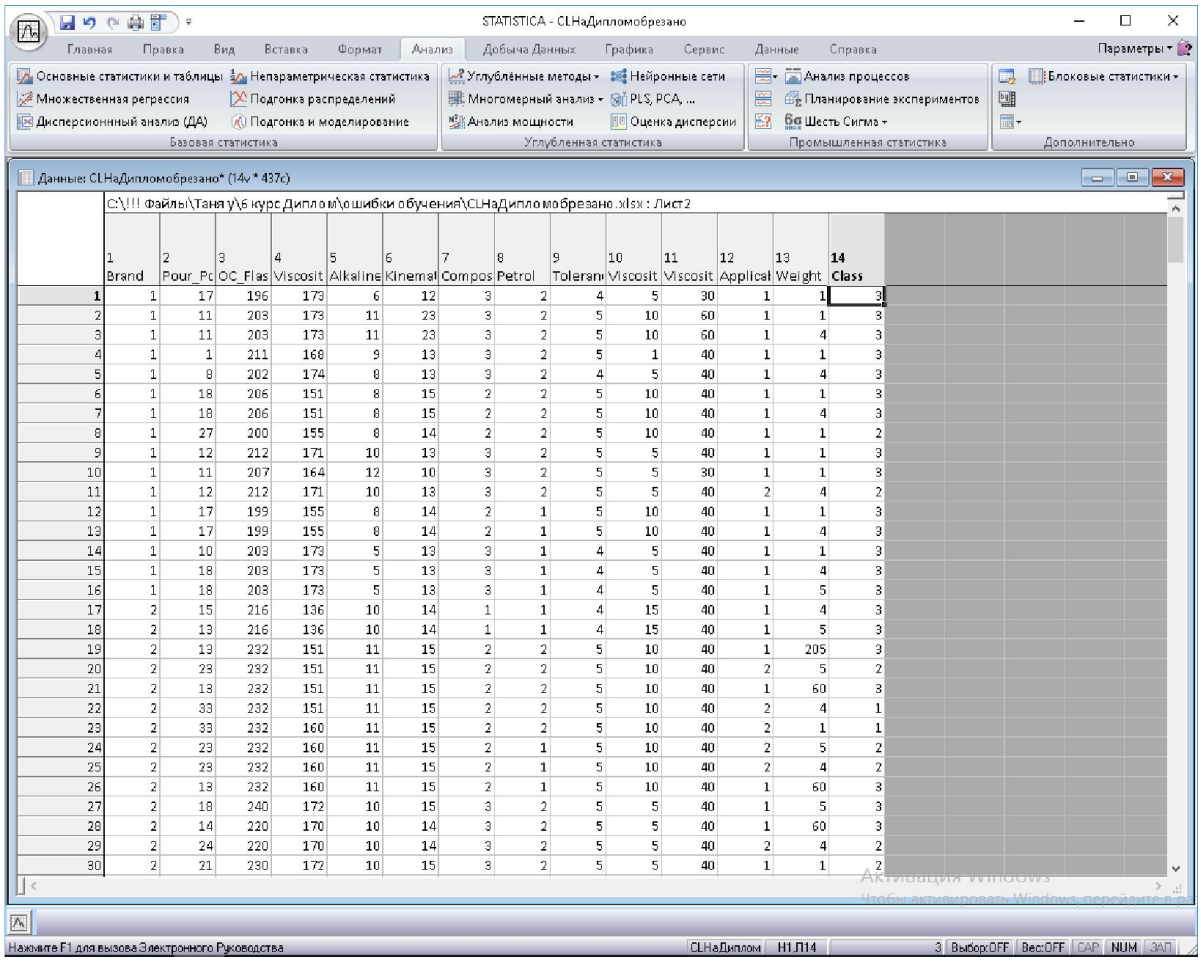

Рисунок 2.23 – Фрагмент навчальної вибірки

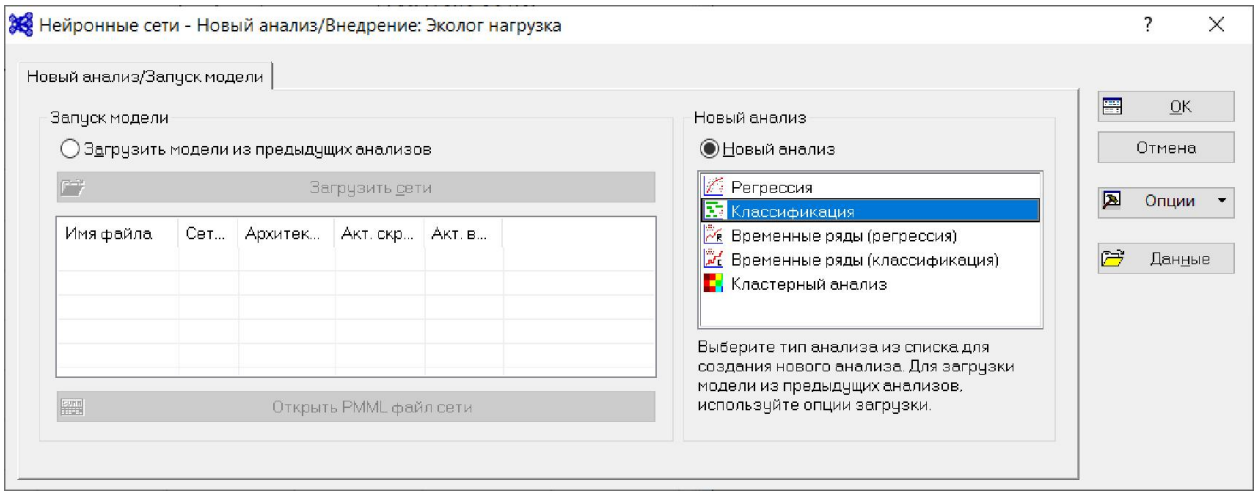

Рисунок 2.24 – Вибір методу аналізу нейронної мережі

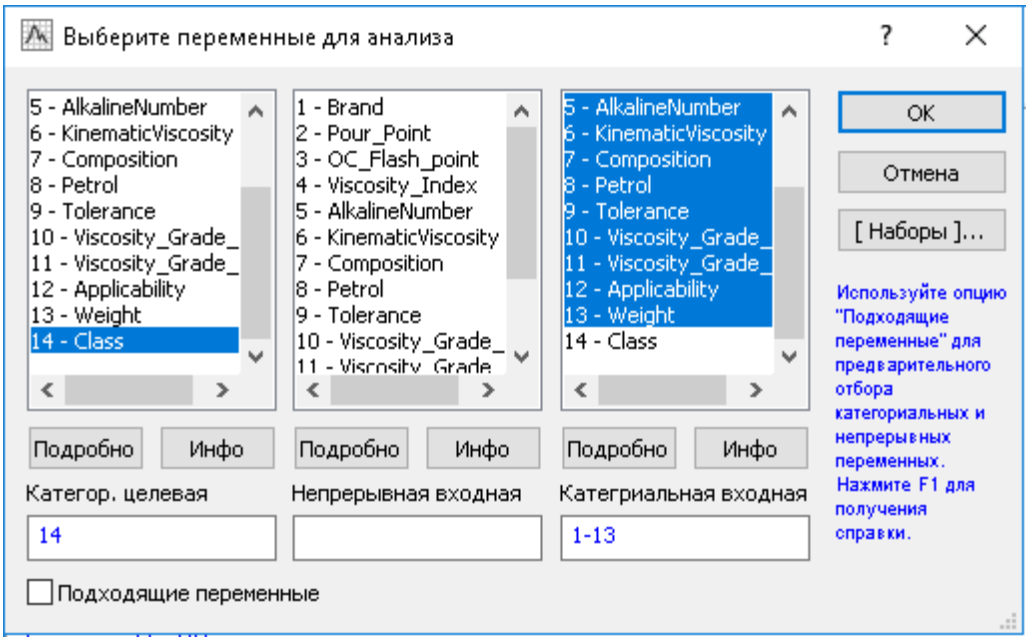

Рисунок 2.25 – Вікно задання змінних

Далі на вкладці «Быстрый» обрати: «Сетей для обучения» – 20, «Сохранять сетей» – 8. Поставити галочки на типі мереж: МЛП. Далі натиснути «Обучить», чекати завершення процесу навчання.

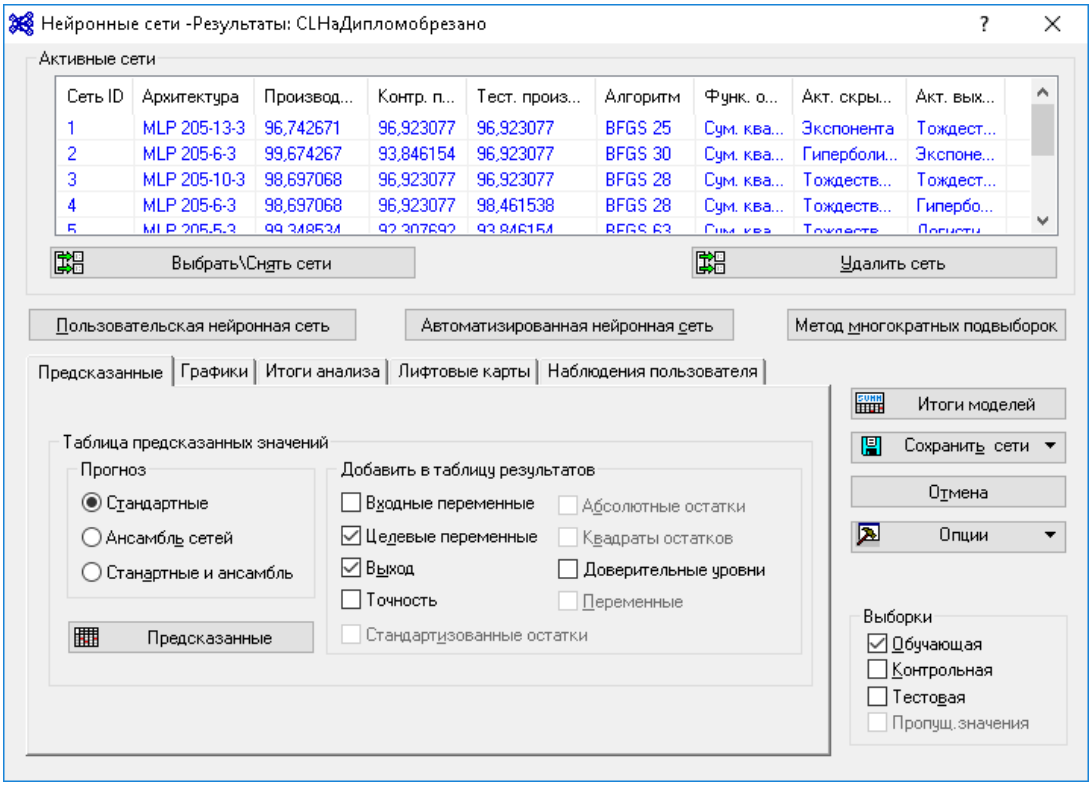

Рисунок 2.26 – Результати навчання мережі

Отже, бачимо результат – мережі навчились майже ідеально. Натиснути «Итоги моделей»(Рисунок 2.27) та «Итоги классификации»(Рисунок2.28). Обираю з них три найкращі – натиснути «Выбрать/Снять сети» (Рисунок 2.29).

Итоги моделей (CLНаДипломобрезано)

N	Архитектура	Производительность обуч.	Контр. производительность.	Тест. производительность.	Алгоритм обучения	Функция ошибки	Ф-я актив. скрытых нейр.	Ф-я актив. выходных нейр.
1	MLP 205-13-3	96,74267	96,92308	96,92308	BFGS 25	Сум. квадр.	Экспонента	Тождественная
2	MLP 205-6-3	99,67427	93,84615	96,92308	BFGS 30	Сум. квадр.	Гиперболическая	Экспонента
3	MLP 205-10-3	98,69707	96,92308	96,92308	BFGS 28	Сум. квадр.	Тождественная	Тождественная
4	MLP 205-6-3	98,69707	96,92308	98,46154	BFGS 28	Сум. квадр.	Тождественная	Гиперболическая
5	MLP 205-5-3	99,34853	92,30769	93,84615	BFGS 63	Сум. квадр.	Тождественная	Логистическая
6	MLP 205-13-3	96,09121	87,69231	96,92308	BFGS 11	Сум. квадр.	Экспонента	Логистическая
7	MLP 205-8-3	97,71987	92,30769	93,84615	BFGS 30	Сум. квадр.	Логистическая	Экспонента
8	MLP 205-14-3	99,34853	92,30769	98,46154	BFGS 53	Сум. квадр.	Экспонента	Гиперболическая

Рисунок 2.27 – Підсумки моделей

	Class (Итоги классификации) (CLНаДипломобрезано) Выборки: Обучающая				
		Class-1	Class-2	Class-3	**Class-Все**
1.MLP 205-13-3	Все	59,0000	82,0000	166,0000	307,0000
	Правильно	50,0000	82,0000	165,0000	297,0000
	Неправильно	9,0000	0,0000	1,0000	10,0000
	Правильно (%)	84,7458	100,0000	99,3976	96,7427
	Неправильно (%)	15,2542	0,0000	0,6024	3,2573
2.MLP 205-6-3	Все	59,0000	82,0000	166,0000	307,0000
	Правильно	59,0000	82,0000	165,0000	306,0000
	Неправильно	0,0000	0,0000	1,0000	1,0000
	Правильно (%)	100,0000	100,0000	99,3976	99,6743
	Неправильно (%)	0,0000	0,0000	0,6024	0,3257
3.MLP 205-10-3	Все	59,0000	82,0000	166,0000	307,0000
	Правильно	58,0000	79,0000	166,0000	303,0000
	Неправильно	1,0000	3,0000	0,0000	4,0000
	Правильно (%)	98,3051	96,3415	100,0000	98,6971
	Неправильно (%)	1,6949	3,6585	0,0000	1,3029
4.MLP 205-6-3	Все	59,0000	82,0000	166,0000	307,0000
	Правильно	59,0000	78,0000	166,0000	303,0000
	Неправильно	0,0000	4,0000	0,0000	4,0000
	Правильно (%)	100,0000	95,1220	100,0000	98,6971
	Неправильно (%)	0,0000	4,8780	0,0000	1,3029
5.MLP 205-5-3	Все	59,0000	82,0000	166,0000	307,0000
	Правильно	59,0000	82,0000	165,0000	306,0000
	Неправильно	0,0000	0,0000	1,0000	1,0000
	Правильно (%)	100,0000	100,0000	99,3976	99,6743
	Неправильно (%)	0,0000	0,0000	0,6024	0,3257
6.MLP 205-13-3	Все	59,0000	82,0000	166,0000	307,0000
	Правильно	55,0000	78,0000	162,0000	295,0000
	Неправильно	4,0000	4,0000	4,0000	12,0000
	Правильно (%)	93,2203	95,1220	97,5904	96,0912
	Неправильно (%)	6,7797	4,8780	2,4096	3,9088
7.MLP 205-8-3	Все	59,0000	82,0000	166,0000	307,0000
	Правильно	54,0000	81,0000	165,0000	300,0000
	Неправильно	5,0000	1,0000	1,0000	7,0000
	Правильно (%)	91,5254	98,7805	99,3976	97,7199
	Неправильно (%)	8,4746	1,2195	0,6024	2,2801

Рисунок2.28 – Підсумки класифікації

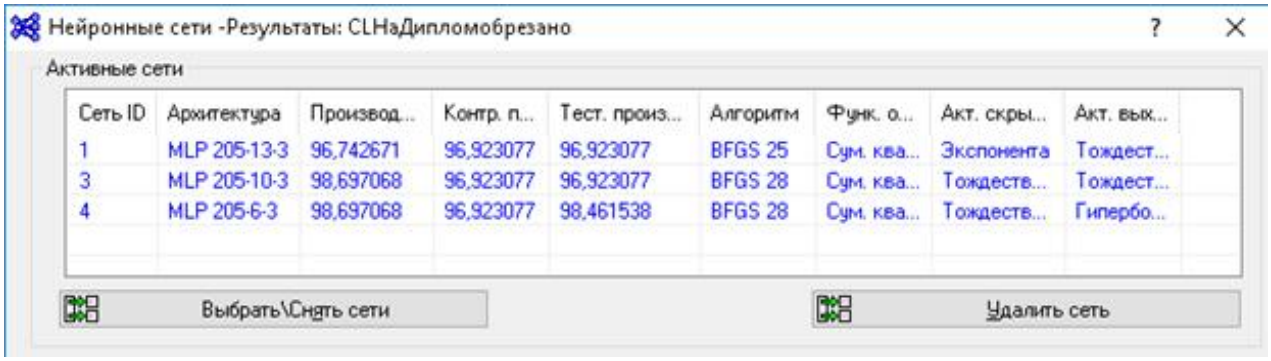

Рисунок2.29 – Обрані мережі.

З цієї таблиці бачимо, що кращі мережі MLP 205-13-3, MLP 205-10-3, MLP 205-6-3 мають такі показники:

- продуктивність – 96,7 - 98,7% ;
- контрольна продуктивність – 96,9%;
- тестова продуктивність – 96,9-98,5%.

Варто зауважити, що й всі інші навчені мережі мають результати продуктивності більше 92%, крім одного випадку – 87%, натиснути «Матрица ошибок» (Рисунок 2.30), за допомогою цього звіту можна побачити скільки маємо помилок навчання.

Предсказанные категор.	Class (Матрица ошибок) (CLНаДипломобрезано) Выборки: Обучающая		
	Class-1	Class-2	**Class-3**
1.MLP 205-13-3-1	50	0	0
1.MLP 205-13-3-2	9	82	1
1.MLP 205-13-3-3	0	0	165
3.MLP 205-10-3-1	58	1	0
3.MLP 205-10-3-2	1	79	0
3.MLP 205-10-3-3	0	2	166
4.MLP 205-6-3-1	59	2	0
4.MLP 205-6-3-2	0	78	0
4.MLP 205-6-3-3	0	2	166

Рисунок 2.30 – Матриця помилок для трьох обраних моделей

Наступним кроком відкрити вкладку «Наблюдения пользователя» та натиснути «Значения пользователя». Скопіювати зі своєї«Таблиця для передбачення значень» (Таблиця 2.1), окрім колонки Class у таблицю «Значения пользователя», попередньо поставивши цифру 13 у пункті «Количество наблюдений для прогноза» – для дослідження я обрала рандомно 13 зразків зі своєї таблиці, які не приймали участь у навчанні (Рисунок 2.31).

Таблиця 2.1 – Таблиця для передбачення значень

	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1	Brand	Pour_ Point	OC_Flash _point	Viscosity _Index	Alkaline Number	Kinematic Viscosity	Comp osition	Petrol	Tolerance	Viscosity _Grade_ 1	Viscosity _Grade_ 2	Applicability	Weight	Class
2	**3**	27	200	155	8	14	2	2	5	10	40	1	4	2
3	2	24	220	170	10	14	3	2	5	5	40	2	1	2
4	3	33	220	130	8	15	1	2	3	15	40	2	1	1
5	3	13	220	130	8	15	1	2	3	15	40	1	30	3
6	**3**	24	210	145	8	15	2	1	4	10	40	1	205	2
7	**3**	40	236	135	11	15	1	2	4	15	40	2	966	1
8	**4**	14	234	178	12	23	3	2	5	10	60	1	4	3
9	**5**	27	235	153	10	15	3	1	7	10	40	2	191	2
10	**5**	10	225	134	11	15	1	1	6	15	40	1	56	3
11	**6**	30	236	136	11	13	1	2	6	15	40	2	205	1
12	7	41	230	152	8	14	2	2	4	10	40	2	1	1
13	**6**	10	234	138	10	16	1	1	7	15	40	1	205	3
14	7	24	236	135	7	14	1	1	4	15	40	1	20	2
15	**4**	14	225	136	9	15	1	1	3	15	40	1	20	3

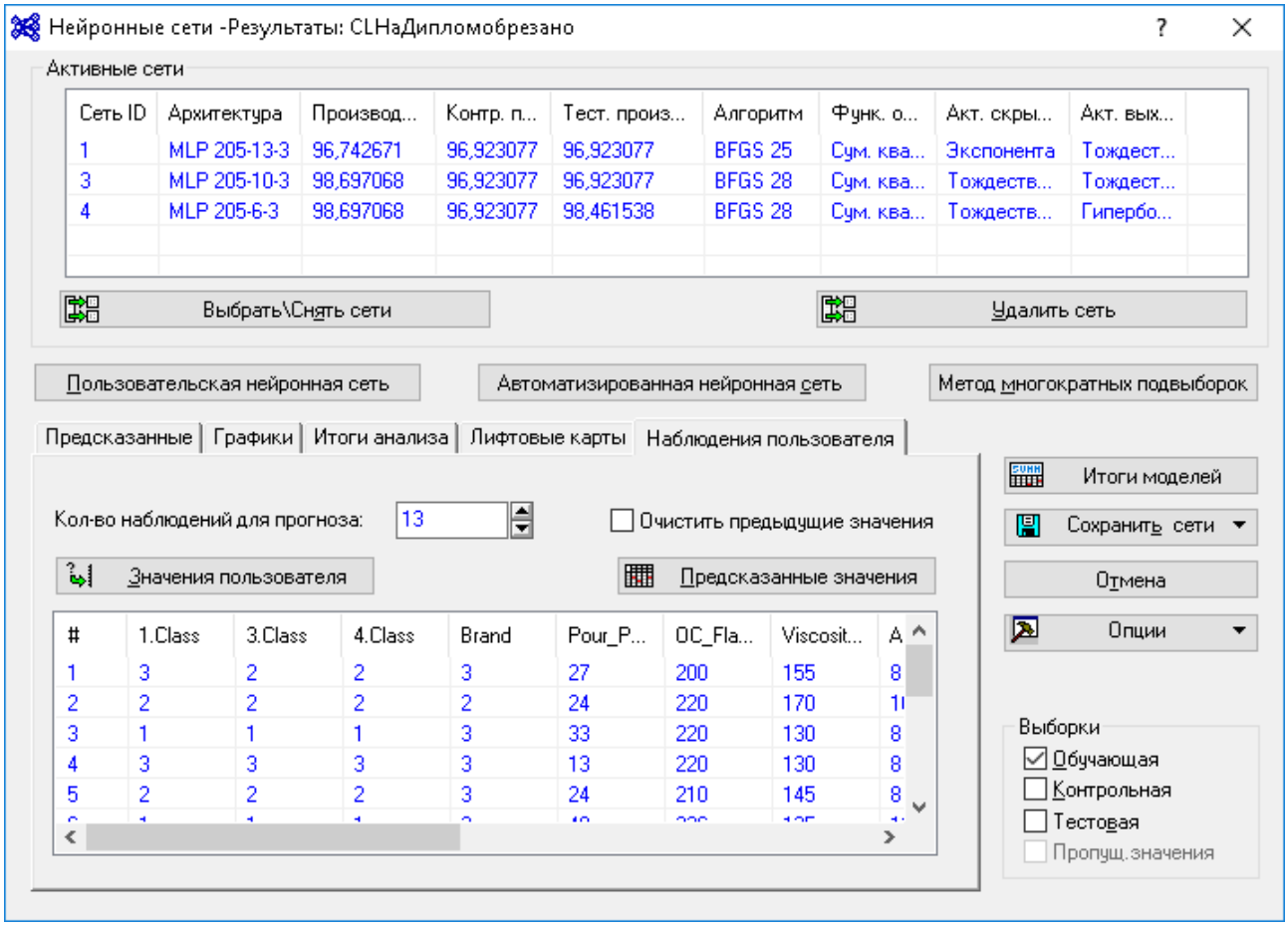

Рисунок 2.31 – Заповнення таблиці значеннями користувача

Натиснути кнопку «Предсказанные значения»і отримати таблицю значень користувача, що зображена на Рисунок 2.32. Потім до своєї таблиці для передбачення значень додала для порівняння три стовпчики з результатом зі Statistica 10.0 (Таблиця 2.2).

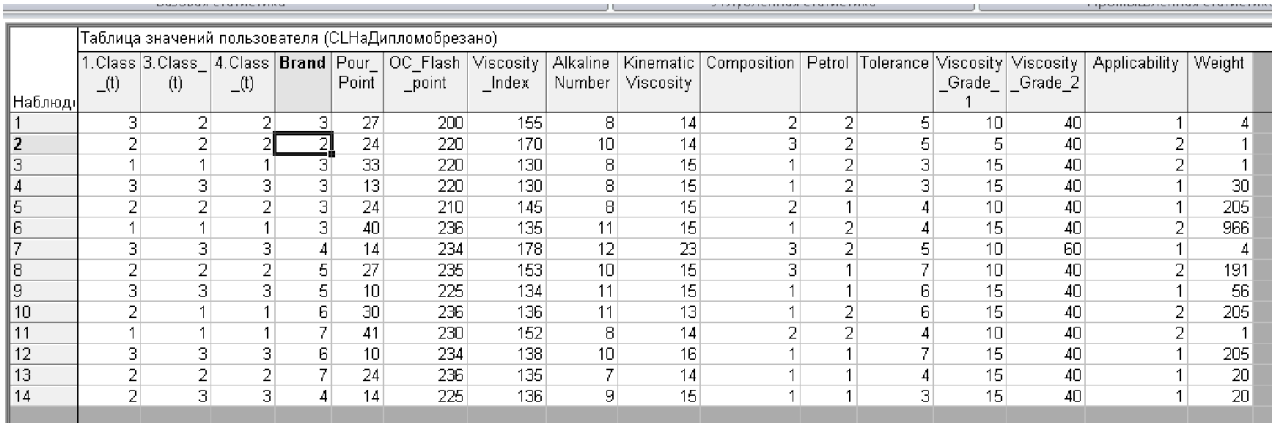

Таблица значений пользователя (CLНаДипломобрезано)

Наблюд	1.Class_(t)	3.Class_(t)	4.Class_(t)	Brand	Pour_Point	OC_Flash_point	Viscosity_Index	Alkaline Number	Kinematic Viscosity	Composition	Petrol	Tolerance	Viscosity_Grade_1	Viscosity_Grade_2	Applicability	Weight
1	3	2	2	3	27	200	155	8	14	2	2	5	10	40	1	4
2	2	2	2	2	24	220	170	10	14	3	2	5	5	40	2	1
3	1	1	1	3	33	220	130	8	15	1	2	3	15	40	2	1
4	3	3	3	3	13	220	130	8	15	1	2	3	15	40	1	30
5	2	2	2	3	24	210	145	8	15	2	1	4	10	40	1	205
6	1	1	1	3	40	236	135	11	15	1	2	4	15	40	2	966
7	3	3	3	4	14	234	178	12	23	3	2	5	10	60	1	4
8	2	2	2	5	27	235	153	10	15	3	1	7	10	40	2	191
9	3	3	3	5	10	225	134	11	15	1	1	6	15	40	1	56
10	2	1	1	6	30	236	136	11	13	1	2	6	15	40	2	205
11	1	1	1	7	41	230	152	8	14	2	2	4	10	40	2	1
12	3	3	3	6	10	234	138	10	16	1	1	7	15	40	1	205
13	2	2	2	7	24	236	135	7	14	1	1	4	15	40	1	20
14	2	3	3	4	14	225	136	9	15	1	1	3	15	40	1	20

Рисунок 2.32 – Таблиця значень користувача

Таблиця 2.2 – Передбачення значень в порівнянні з результатом з програми

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
1	Brand	Pour_Point	OC_Flash_point	Viscosity_Index	Alkaline Number	Kinematic Viscosity	Composition	Petrol	Tolerance	Viscosity_Grade_1	Viscosity_Grade_2	Applicability	Weight	Class	Class1 (Statistica)	Class2 (Statistica)	Class3 (Statistica)	№ п/п
2	**3**	27	200	155	8	14	2	2	5	10	40	1	4	2	3	2	2	1
3	2	24	220	170	10	14	3	2	5	5	40	2	1	2	2	2	2	2
4	3	33	220	130	8	15	1	2	3	15	40	2	1	1	1	1	1	3
5	3	13	220	130	8	15	1	2	3	15	40	1	30	3	3	3	3	4
6	**3**	24	210	145	8	15	2	1	4	10	40	1	205	2	2	2	2	5
7	**3**	40	236	135	11	15	1	2	4	15	40	2	966	1	1	1	1	6
8	**4**	14	234	178	12	23	3	2	5	10	60	1	4	3	3	3	3	7
9	**5**	27	235	153	10	15	3	1	7	10	40	2	191	2	2	2	2	8
10	**5**	10	225	134	11	15	1	1	6	15	40	1	56	3	3	3	3	9
11	**6**	30	236	136	11	13	1	2	6	15	40	2	205	1	2	1	1	10
12	7	41	230	152	8	14	2	2	4	10	40	2	1	1	1	1	1	11
13	**6**	10	234	138	10	16	1	1	7	15	40	1	205	3	3	3	3	12
14	7	24	236	135	7	14	1	1	4	15	40	1	20	2	2	2	2	13
15	**4**	14	225	136	9	15	1	1	3	15	40	1	20	3	2	3	3	14

З Таблиці 2.2 бачимо, що із трьох обраних навчених нейронних мереж дві: MLP-205-10-3 та MLP-205-6-3 –безпомилково в усіх випадках передбачили клас. А одна мережа MLP-205-13-3 в трьох випадках дала помилку, що виділено червоним кольором в таблиці. З цього можна зробити висновки, що мережі MLP-205-10-3 та MLP-205-6-3 можна впевнено використовувати для

визначення класу нової позиції для нашої мережі, а мережу MLP-205-13-3 краще не використовувати для подальшої роботи.

2.5 Порівняння результатів навчання у програмах Statistica 10.0 та Python 3.9

Порівняю результати навчання, що отримала в обох програмах Pythonта Statistica.

У програміPythonза допомогою методу KNNя маю результати навчання: 0,83; 0,82 та 0,8 при кількості сусідів knn = 1; 2 та 4. На тестовій вибірці результат виявився ще кращим 0,91 при knn = 1 та 2, а це є доказом того, що нові зразки, які треба буде класифікувати користувачу, будуть віднесені до правильного класу.

У програмі Statisticaотимали результат тестової продуктивності – 96,92%, що є практично ідеальним для передбачення.А всі нові зразки у двох мережах визначили клас на 100% правильно.

Отже, навчання мереж пройшло успішно в обох програмах і цей процес можна впевнено проводити не тільки в тій організації, в якій я працюю, а й в інших подібних компаніях.

ВИСНОВКИ

Порівнюючи результати роботи в обох програмах, можна зробити висновки, що навчання мережі пройшло вдало. Отже, проведений процес можна успішно використовувати в аналізі продажу товарів компанії, оптимізації роботи та автоматизації процесів. Це значно заощадить час працівників на обробку нових даних та замовлення товару, збільшить прибутки підприємства та спростить роботу, істотно зменшуючи ризик людської помилки чи некомпетентності.

Спроба вдосконалити бізнес-процеси компанії пройшла успішно – всі поставленні цілі та завдання виконані. Алгоритми машинного навчання значно полегшили роботу багатьох відділів та вдосконалили процеси. Основна мета – збільшення прибутку підприємства та зменшення витрат досягнута.

Отже, вирішено,поставлені на початку роботи, проблеми:

- класифікація товарів по групам;
- концентрація на топових позиціях і в результаті маємо збільшення прибутку компанії;
- зменшення запасів «неходових» позицій і відповідно зменшення витрат на їх зберігання на складах;
- швидкий пошук товарів, які потрібно замінити на аналогічну позицію, користуючись навченими мережами;
- доповнення асортименту новими товарами з відповідними вхідними даними, попередньо проаналізувавши за допомогою навчених мереж, який вони матимуть попит серед покупців;
- покращення співпраці з клієнтами завдяки тому, що в наявності завжди є необхідні товари;
- оперативна доставка товарів через оптимізацію роботи складу.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Manohar Swamynathan Bangalore. Mastering Machine Learning with Python in Six Steps., Karnataka, India, 2019. – 771c.

2. Jesse C.Daniel. Data Science with Python and Dask. USA, 2019. – 296 c.

3. Aaron Khan. Python. Data Science. 2019. - 112c.

4. John Paul Mueller and Luka Massaron. Python for Data Science for Dummies, 2nd Edition, 2019. – 498 c.

5. Sayan Mukhopadhyay. Advanced Data Analytics Using Python, 2018. – 195 c.

6. Dr. Ossama Embarak. Data Analysis and Visualization Using Python. Higher Colleges of Technology, Abu Dhabi, United Arab Emirates, 2018. – 390c.

7. Пилипчук В.П., Данніков О.В. Управління продажем: навч.посібник. – К.: КНЕУ, 2011. – 627 [5] с.

8. С.П. Альошин, О.О. Бородіна. Навчальний посібник «Інтелектуальний аналіз даних» для студентів денної та заочної форми навчання: 6.050101 — «Комп'ютерні науки» — Полтава: ПолтНТУ, 2014. — 187 с.

9. Додаткові методи управління запасами.[Електронний ресурс]: [Веб-сайт] – Електроні дані. – Режим доступу: https://studme.org/10560412/logistika/dopolnitelnye_metody_upravleniya_zapasami

10. Менеджмент. Експернті системи: особливості застосування. Реферат. [Електронний ресурс]: [Веб-сайт] – Електроні дані. – Режим доступу: https://osvita.ua/vnz/reports/management/13574/

11. Прогнозирование в системе STATISTICA в среде Windows. Основы теории и интенсивная практика на компьютере. Боровиков В.П., Ивченко Г.И. – М.: Финансы и статистика, 2000. – 380с., ил.

12. Machine Learning for Time Series Forecasting with Python. Francesca Lazzeri, PhD, 2021.–206 с.

13. Етапи розробки експертних систем. [Електронний ресурс]: [Веб-сайт] – Електроні дані. – Режим доступу:http://um.co.ua/2/2-6/2-65179.html

14. Oliver R.Simpson. Python for data analysis., 2019.– 145с.

15. Suresh Kumar Mukhiya, Usman Ahmed. Hands – On Exploratory Data Analysis with Python. Packt Publishing, 2020. - 342с.

16. Mark Solomon Brown.Python Data Science.The Bible. – 2019. – 89 с.

17. Naomi Ceder. The Quick Python Book. Third Edition. Manning Publications.USA, 2020.– 473 с.

18. Michael Heydt. Mastering Pandas for Finance. 2015.– 298 с.

19. Автомобілі. [Електронний ресурс]: [Веб-сайт] – Електроні дані. – Режим доступу:http://www.lotosoil.pl/pl/home/produkty/samochody-osobowe

20. Моторні оливи Castrol. [Електронний ресурс]: [Веб-сайт] – Електроні дані. – Режим доступу: https://castroloil.com.ua/motor.html

21. Автомобільні оливи. [Електронний ресурс]: [Веб-сайт] – Електроні дані. – Режим доступу:https://oil-expert.com.ua/catalog/avto-masla

22. Центр знань. Характеристика. Класифікація ACEA [Електронний ресурс]: [Веб-сайт] – Електроні дані. – Режим доступу:https://penriteoil.com.au/knowledge-centre/Specifications/194/acea-service-classifications/364

23. Каталог мастильних матеріалів. [Електронний ресурс]: [Веб-сайт] – Електроні дані. – Режим доступу: https://www.total-oil.ru/katalog/

24. Огляди. [Електронний ресурс]: [Веб-сайт] – Електроні дані. – Режим доступу: https://maslo.expert/marki-masel

25. Totalenergies. Автомобільні оливи.[Електронний ресурс]: [Веб-сайт] – Електроні дані. – Режим доступу: https://cataloguk.total.ua/russian-ua/automotive/cars?bn=TOTAL

26. Wei-Meng Lee. Python Machine Learning. USA.: Wiley, 2019. – 296с.

27. ABCта XYZ аналіз у Торгсофт. ABC аналіз.[Електронний ресурс]: [Веб-сайт] – Електроні дані. – Режим доступу: https://torgsoft.ua/ua/articles/gid-po-torgsoft/abc-i-xyz-analiz-v-torgsoft/

28. Elf. Автомобільна моторна олива [Електронний ресурс]: [Веб-сайт] – Електроні дані. – Режим доступу:https://www.elf.ua/uk/automotive-lubricants/car-engine-oils

29. Кубишина Н. С. Методика розробки стратегічного набору товарів на промисловому ринку / Н. С. Кубишина // Економічний вісник НТТУ «КПІ». – 2010. – № 7. – С. 171-178. Кубишина Н. С. Методика розробки стратегічного набору товарів на промисловому ринку / Н. С. Кубишина // Економічний вісник НТТУ «КПІ». – 2010. – № 7. – С. 171-178.

30. Урядовий портал [Електронний ресурс]: [Веб-сайт] – Електроні дані. – Режим доступу:https://www.kmu.gov.ua/news/utochneni-dani-vvp-za-2020-rik-vid-derzhstatu-viyavilis-krashchimi-nizh-prognoz

31. Python: 7 importantreasonswhyyoushoulduse Python[Електронний ресурс]: [Веб-сайт] – Електроні дані. – Режим доступу:https://medium.com/@mindfiresolutions.usa/python-7-important-reasons-why-you-should-use-python-5801a98a0d0b

32. 15 WaysMachinelearningwillimpactyoureverydaylife[Електронний ресурс]: [Веб-сайт] – Електроні дані. – Режим доступу:https://elitedatascience.com/machine-learning-impact

ДОДАТОК А
ВИХІДНИЙ ПРОГРАМНИЙ КОД – ВІЗУАЛІЗАЦІЯ

(2_Visual)

```
importpandasaspd

importmatplotlib.pyplotasplt

importseabornassns

dataset= pd.read_excel("CLНаДиплом.xlsx")

Brand = dataset['Brand']

Class = dataset['Class']

OC_FLash_point = dataset['OC_Flash_point']

Pour_Point = dataset['Pour_Point']

Viscosity_Index = dataset['Viscosity_Index']

AlkalineNumber = dataset['AlkalineNumber']

KinematicViscosity = dataset['KinematicViscosity']

Composition=['Composition']

Tolerance = ['Tolerance']

Viscosity_Grade_1 = ['Viscosity_Grade_1']

Viscosity_Grade_2 = ['Viscosity_Grade_2']

Weight=['Weight']

#sns.stripplot(x='Class', y = 'OC_Flash_point', hue = 'Class', data= dataset, jitter=True)

#sns.swarmplot(x='Class', y='Viscosity_Index', hue='Class', data=dataset, palette='Set1',dodge=False)

#sns.stripplot(x='Class', y = 'Pour_Point', hue = 'Class', data= dataset, jitter=True)

#sns.stripplot(x='Class', y = 'AlkalineNumber', hue = 'Class', data= dataset, jitter=True)

#sns.stripplot(x='Class', y = 'Brand', hue = 'Class', data= dataset, jitter=True)
```

```
#sns.stripplot(x='Class', y = 'KinematicViscosity', hue = 'Class', data= dataset, jitter=True)

#sns.stripplot(x='Class', y = 'Composition', hue = 'Class', data= dataset, jitter=True)

#sns.stripplot(x='Class', y = 'Tolerance', hue = 'Class', data= dataset, jitter=True)

#sns.stripplot(x='Class', y = 'Viscosity_Grade_1', hue = 'Class', data= dataset, jitter=True)

#sns.stripplot(x='Class', y = 'Viscosity_Grade_2', hue = 'Class', data= dataset, jitter=True)

sns.stripplot(x='Class', y = 'Weight', hue = 'Class', data= dataset, jitter=True)

plt.show()
```

3_Visual_all

```
import matplotlib.pyplot as plt

import matplotlib as mpl

import pandas as pd

import mglearn

import seaborn as sns

import numpy as np

df = pd.read_excel('ClНаДиплом1.xlsx')

plt.figure(figsize=(2,3), dpi= 50)

sns.pairplot(df, hue="Class")

plt.show()
```

4_Visual_all_txt

```
import matplotlib.pyplot as plt

import pandas as pd

import mglearn

import numpy as np
```

```
from pandas.plotting import scatter_matrix
import seaborn as sns
input_file = 'CLНаДиплом.txt'
data = np.loadtxt(input_file)
X, y = data [:, : -1], data[:, -1]
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, y, random_state=0)
print(type(X_train))
print("Form of array массива X_train: {}". format(X_train.shape))
print("Form of array y_train: {}". format(y_train.shape))
print("Form of array X_test: {}". format(X_test.shape))
print("Form of array y_test: {}". format(y_test.shape))
print("0-Brand,1-Pour_Point, 2-OC_Flash_point,3-Viscosity_Index,\n4-Alkaline_Number, 5-Kinematic_Viscosity,6-Composition,\n7-Petrol, 8-Tolerance, 9-Viscosity,_Grade_1")
print("10-Viscosity_Grade_2, 11-Applicability, 12-Weight")
data_dataframe = pd.DataFrame(X_train, y_train)
grr = scatter_matrix(data_dataframe, c=y_train, figsize=(15, 15), marker = 'o',
                     hist_kwds={'bins': 20}, s=60, alpha=.8, cmap=mglearn.cm3)
plt.xlabel('\n Class_1-green, Class_2-red, Class_3-blue.....')
plt.show()
```

ДОДАТОК Б
ВИХІДНИЙ ПРОГРАМНИЙ КОД –ВИВЕДЕННЯ КЛЮЧІВ

(1_на_диплом)

```
import pandas as pd
df = pd.read_excel('C:\!!! Файлы\ClНаДиплом.xlsx')
print('Keys df: \n{}' .format(df.keys()), '\n')
print(df['Class'][:10], '\n...')
print(df['Pour_Point'], '\n')
print(df['OC_Flash_point'], '\n')
print(df['Viscosity_Index'], '\n')
print(df['AlkalineNumber'], '\n')
print(df['KinematicViscosity'], '\n')
print(df['Composition'], '\n')
print(df['Petrol'], '\n')
print("Form of data array: {}".format (df['Class'].shape))
```

ДОДАТОК В
ВИХІДНИЙ ПРОГРАМНИЙ КОД –МЕТОД К-НАЙБЛИЖЧИХ СУСІДІВ

```
import numpy as np
import pandas as pd
import mglearn
input_file = 'DIPLOM17.txt'
data = np.loadtxt(input_file)
X, y = data [:, : -1], data[:, -1]
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, y, random_state=0)

print('Fragment of learning sample', end='\n')
for i in range(5):
    print(X_train[i], y_train[i], end='\n')
from sklearn.neighbors import KNeighborsClassifier
knn=KNeighborsClassifier(n_neighbors = 3)
knn.fit(X_train, y_tra
X_new=np.array(X_test[:2])
X_new
prediction = np.array (knn.predict(X_new))
print ('Method k-mean for X_test[0] and X-test[1]', prediction)
print ('Label from test simple', y_test[:2])
y_pred = knn.predict(X_test)
print("predictions for all test sampes X_test:\n()".format(y_pred))
```

```
print('Estmation result of correct classification for test samples:'+
    '(y_pred=y_test)/ samples = {:.2f}'. format(np.mean(y_pred==y_test))
```