ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ ЕКОНОМІЧНИЙ УНИВЕРСИТЕТ ІМЕНІ СЕМЕНА КУЗНЕЦЯ

ФАКУЛЬТЕТ ЕКОНОМІЧНОЇ ІНФОРМАТИКИ

КАФЕДРА ІНФОРМАЦІЙНИХ СИСТЕМ

Пояснювальна записка

до дипломної роботи

МАГІСТРА

на тему: «Підвищення ефективності розв’язування задач Machine Learning на основі вбудованих мовних засобів SQL Server»

Виконала:

студентка 2 року навчання

за освітнім ступенем «магістр»

зі спеціальності

126 «Інформаційні системи і технології»

Анна КОСКІНА

Керівник: к.ф-м.н., доц. кафедри ІС
Віктор ФЕДЬКО

Харків – 2020 рік

РЕФЕРАТ

Пояснювальна записка до магістерської роботи містить: 85 сторінок, 46 рисунів, 23 таблиці, 54 джерела, 11 формул.

Метою магістерської роботи є дослідження засобів вирішення задач в Machine Learning за допомогою SQL Server Machine Learning Services для визначення найбільш ефективного методу їх програмної реалізації: з використанням вбудованих мовних засобів SQL Server або класичного способу обробки даних.

Об'єкт дослідження – процеси вирішення задач в Machine Learning.

Предмет дослідження – методи та алгоритми вирішення задач машинного навчання з використанням можливостей SQL Server Machine Learning Services.

Методи дослідження – проведення експериментів щодо ефективності застосування засобів вирішення задач в Machine Learning за допомогою SQL Server Machine Learning Services та статистична обробка отриманих результатів.

У результаті дослідження проаналізовано існуючі способи обробки даних та визначити який з них є найкращим.

Для проведення дослідження були використані пакети Phyton – pandas, pyodbc та sklearn, для R – ODBC, DBI; програмні засоби – SQL Server Management Studio, Spyder, RStudio. Програмна реалізація дослідження виконана мовою Phyton.

Визначено спосіб, який дозволить збільшити ефективність розв'язування задач прогнозування даних у 2 – 4 рази, використовуючи алгоритми машинного навчання.

Отримані результати можуть бути впроваджені в багатьох сферах: виробництво, транспорті системи, медицина, освіта і т.д., що дозволить оптимізувати процес роботи з даними та працювати в режимі реального часу.

АВТОМАТИЗАЦІЯ, АЛГОРИТМ, АНАЛІЗ ДАНИХ, ДОСЛІДЖЕННЯ, ЕКСПЕРИМЕНТ, ЛІНІЙНА РЕГРЕСІЯ, МАШИННЕ НАВЧАННЯ, МЕТОДИ ПРОГРАМНОЇ РЕАЛІЗАЦІЇ МАШИННОГО НАВЧАННЯ, МОДЕЛЬ, НАВЧАННЯ МОДЕЛІ, ПРОГНОЗУВАННЯ.

ABSTRACT

Explanatory note to the master's thesis contains: 85 pages, 46 figures, 23 tables, 54 sources, 11 formulas.

The aim of this work is to study the means of solving problems in Machine Learning using SQL Server Machine Learning Services to determine the most effective method of their software implementation: using a stored procedure that contains a software implementation of work with data created by Python or classical data processing.

The object of research is the processes of solving problems in Machine Learning.

The subject of the study are methods and algorithms for solving Machine Learning problems using the capabilities of SQL Server Machine Learning Services.

Research methods are experiments on the effectiveness of the application of problem-solving tools in Machine Learning using SQL Server Machine Learning Services and statistical processing of the results.

As a result of the research, it is necessary to analyze the existing methods of data processing and determine which of them is the best.

Phyton packages - pandas, pyodbc and sklearn, software - SQL Server Management Studio, Spyder were used for the study. The software implementation of the study is performed in Phyton language.

A method that will increase the efficiency of solving data forecasting problems by 2 – 4 times using machine learning algorithms has been identified.

The results can be embedded in educational institutions for teachers whose disciplines are related to machine learning.

AUTOMATION, ALGORITHM, DATA ANALYSIS, RESEARCH, EXPERIMENT, LINEAR REGRESSION, MACHINE LEARNING, PROGRAM IMPLEMENTATION METHODS, MODEL, MODEL TRAINING, FORECASTING.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ

AI – Artificial intelligence (штучний інтелект)

IT – Information Technologies (інформаційні технології)

ML – Machine Learning (машинне навчання)

SSMS – SQL Server Management Studio (програма з Microsoft SQL Server для конфігурації, управління і адміністрування всіх компонентів Microsoft SQL Server)

SQL – Structured Query Language (мова структурованих запитів)

БД – база даних

ВСТУП

В наш час важко уявити життя людства без використання інформаційних технологій. Вони посідають важливе місце майже у всіх сферах життя: в медицині, освіті, виробництві та багатьох інших. Завдяки їм можна значно скоротити час на пошук, аналіз та обробку необхідної інформації, полегшити вирішення певних проблем, здійснення складних математичних обчислень, забезпечити швидкий обмін даними.

Також в останні роки стрімкого розвитку набуває машинне навчання. Це великий підрозділ штучного інтелекту, що вивчає методи побудови алгоритмів, здатних навчатися. Основа мета – на основі вхідних даних передбачити можливу поведінку системи. Чим різноманітнішими будуть вхідні дані, тим простіше машині знайти закономірності між ними і тим точніший результат.

Таким чином, інформаційні технології в своєму розвитку вийшли на більш якісний рівень. Інформаційні технології на основі новітньої комп'ютерної техніки сприяють високоефективній організації управління на підприємствах, в навчальних закладах; допомагають знизити витрати на різні операції.

Дослідження методів програмної реалізації машинного навчання та ефективної обробки даних наразі є досить актуальною темою, оскільки потреба у використанні комп’ютерної техніки, програмних продуктів високої якості з кожним днем зростає все більше. Застосування машинного навчання може значно пришвидшити та підвищити ефективність прийняття рішень, прогнозувати та аналізувати поведінку системи, завдяки чому можна буде працювати в режимі реального часу та уникнути небажаних ситуацій.

Метою роботи є дослідження засобів вирішення задач в Machine Learning за допомогою SQL Server Machine Learning Services для визначення найбільш ефективного методу їх програмної реалізації: з використанням вбудованих мовних засобів SQL Server або класичного способу обробки даних.

Об'єкт дослідження – процеси вирішення задач в Machine Learning.

Предмет дослідження – методи та алгоритми вирішення задач машинного навчання з використанням можливостей SQL Server Machine Learning Services.

Методи дослідження – проведення експериментів щодо ефективності застосування засобів вирішення задач в Machine Learning за допомогою SQL Server Machine Learning Services та статистична обробка отриманих результатів.

Задачі дослідження:

1) Проаналізувати стан проблеми.
2) Провести огляд об’єкта управління.

3) Провести огляд літературних джерел.

4) Ознайомитися з існуючими програмними засобами, які використовуються для машинного навчання.

5) Підготувати комп'ютерну систему, на якій буде проводитись дослідження.

6) Підготувати дані, що будуть основою для навчання моделі, яка в подальшому буде використовуватися для отримання прогнозованих даних.

7) Провести обробку даних з використанням методів Machine Learning та визначити найбільш ефективний спосіб роботи з даними.

8) Проаналізувати отримані результати.

Новизна дослідження – визначення кількісної оцінки ефективності застосування вбудованих мовних засобів SQL Server Machine Learning Services. Шляхом проведення експериментів було визначено, що застосування SQL Server ML Services дозволяє пришвидшити процес обробки даних в 2 – 4 рази для локально розташованого сервера та в 2.5 рази для сервера, що знаходиться в хмарній платформі. Виявлено, що зі збільшенням кількості даних ефективність використання вбудованих мовних засобів SQL Server ставала більш помітною.

Визначено ефективність застосування сервера БД для обробки даних та переваги даного способу.

Практична цінність роботи – результати досліджень можуть бути впроваджені в багатьох сферах: виробництво, транспорті системи, медицина, освіта і т.д.

Було виявлено, що задачі дослідження вирішуються в 2 – 4 рази швидше, ніж за класичного способу обробки даних, що дозволяє працювати в режимі реального часу. Завдяки цьому є змога миттєво реагувати на зміни та уникати небажаних наслідків роботи системи.

Визначено найбільш ефективний спосіб обробки даних, використання якого дозволяє зменшити час, необхідний на отримання кінцевого результату вирішення задачі засобами машинного навчання.

За темою дослідження були опубліковані тези [14].

1. АНАЛІЗ ПРОБЛЕМАТИКИ ТА ПОСТАНОВКА ЗАВДАНЬ ДОСЛІДЖЕННЯ «ЕФЕКТИВНІСТЬ РОЗВ'ЯЗУВАННЯ ЗАДАЧ MACHINE LEARNING НА ОСНОВІ ВБУДОВАНИХ МОВНИХ ЗАСОБІВ SQL SERVER»

1.1. Коротка характеристика об'єкта управління

Кафедра інформаційних систем ХНЕУ ім. С. Кузнеця у співпраці з провідними IT-компаніями, спираючись на новітні підходи в освіті, досвід участі у міжнародних проектах, готує креативних та висококваліфікованих фахівців:

- бакалаврів за спеціальностями 121 «Інженерія програмного забезпечення», 122 «Комп'ютерні науки» та 126 «Інформаційні системи та технології»;
- магістрів за спеціальностями 122 «Комп'ютерні науки» та 126 «Інформаційні системи та технології».

Навчальний процес здійснюють 31 викладач (з них 7 докторів наук та 20 кандидатів наук), більшість з яких мають професійні сертифікати від компаній Microsoft, IBM та інших.

У курсах, які викладаються на кафедрі, широко використовуються матеріали, отримані в процесі стажування викладачів на базі провідних IT-компаній міста, таких як, EPAM Systems, NIX Solutions. Також проводяться відкриті заняття та тренінги за участю фахівців таких компаній. Це дозволяє значно підвищити якість курсових і дипломних проектів, їхню практичну значимість для підприємств, компаній IT-галузі, на матеріалах яких вони виконувалися, та підвищити успішність студентів.

На кафедрі існують студентські гуртки, діяльність яких пов'язана з проектуванням, розробкою, впровадженням та експлуатацією інформаційних систем. Працює Microsoft IT Academy. Кафедра постійно приймає участь у програмі «Академічна ініціатива» компанії IBM.

Кафедра бере участь у міжнародних програмах підготовки висококваліфікованих фахівців, а саме: спільній франко-українській програмі підготовки магістрів за фахом «Бізнес-інформатика» з університетом Ліон 2 (Ліон, Франція); спільній магістерській програмі подвійного диплому «Створення інноваційних підприємств» з університетом Монпельє (Монпельє, Франція); спільній словацько-українській програмі «Бізнес-аналітика» та інформаційні системи у підприємництві» (Братислава, Словаччина) Також, продовжується робота над такими міжнародними проектами:

1) HORIZON 2020 «Gender Equality Plans for Information Sciences and Technology Research Institutions» (EQUAL-IST) (Горизонт 2020 «Планування гендерної рівності в наукових дослідженнях у галузі комп'ютерних наук та інформаційних технологій», грантова угода № 710549).

2) ERASMUS+ Establishing Modern Master-level Studies in Information Systems (Еразмус+ «Створення сучасної магістерської програми в галузі інформаційних систем», проект № 561592-EPP-1-2015-1- FR-EPPKA2-CBHE-JP).

3) ERASMUS+ Development of a network infrastructure for youth innovation entrepreneurship support on fablab platforms (Еразмус+ «Створення мережі та інфраструктури підтримки молодіжного інноваційного підприємництва на платформі фаблабів», проект № 561536-EPP-1-2015-1-UK-EPPKA2-CBHE-JP).

4) ERASMUS+ Structuring cooperation in doctoral research, transferrable skills training, and academic writing instruction in Ukraine's regions (Еразмус+ «Структуризація співпраці щодо аспірантських досліджень, навчання універсальних навичок та академічного письма на регіональному рівні України», проект № 574064-EPP-1-2016-1-LT-EPPKA2- CBHE-SP).

5) ERASMUS+ Promoting internationalization of research through establishment and operationalization of Cycle 3 Quality Assurance System in line with the European Integration (Еразмус+ «Стимулювання інтернаціоналізації досліджень шляхом запровадження системи забезпечення якості третього рівня вищої освіти у відповідності до європейських вимог», проект № 574273-EPP-1-2016-1-AM-EPPKA2-CBHE-SP).

Розпочато роботу над новим проектом: ERASMUS+ Implementation of Education Quality Assurance system via cooperation of University-Business-Government in HEIs (Еразмус+ «Імплементація системи забезпечення якості освіти через співробітництво університету-бізнесу-уряду в ЗВО», проект № 586109-EPP-1-2017-1-RO-EPPKA2-CBHE-SP) [9].

Співробітниками кафедри підготовлено та видано 3 підручника, більш ніж 94 навчальних посібників, з яких 15 мають гриф Міністерства освіти і науки України. Кафедрою розроблено два мультимедійних інтерактивних електронних начальних посібника з інформатики для самостійної роботи студентів.

На кафедрі ведеться постійна робота зі студентами молодших курсів по залученню їх до науково-дослідницької роботи, для цього на кафедрі організовано роботу студентського наукового гуртка.

З переходом на кредитно-модульну систему навчання усі викладачі кафедри впроваджують у навчальний процес нові методи викладання з використанням сучасних технічних засобів і інформаційних технологій. Викладачами кафедри проводиться активна робота з впровадження системи дистанційної освіти для студентів очної та заочної форм навчання [22].

Схема організаційної структури кафедри інформаційних систем, яка була базою практики та для якої будуть корисними результати даного дослідження, детальніше розглянута на рис. 1.1.

Працівники кафедри, які викладають дисципліни, пов’язані з процесами машинного навчання (такі як «Системи штучного інтелекту», «Управлінські ІС та сховища даних»), зможуть використовувати інформацію, представлену у даній дипломній роботі для покращення прочесу навчання. Матеріали будуть корисними як для студентів, так і для викладачів, оскільки після ознайомлення з даними, представленими в роботі, буде чітко зрозуміло основні переваги та недоліки різних способів обробки даних, проблеми, які можуть виникнути в роботі з ними та можливі засоби їх вирішення. Буде можливість розв’язати більший спектр різноманітних задач за допомогою методів машинного навчання за рахунок зменшення часу, необхідного на отримання результатів.

Оскільки в роботі представлено багато графіків та таблиць, це дає змогу наочно оцінити всі результати досліджень та визначити для себе переваги того чи іншого способу роботи з даними в залежності від того, який результат він бажає отримати.

Окрім того, результати даного дослідження можуть бути корисними під час автоматизації деяких процесів на кафедрі. Наприклад, прогнозування кількості абітурієнтів, які будуть навчатися на кафедрі, на основі статистичних даних попередніх років.

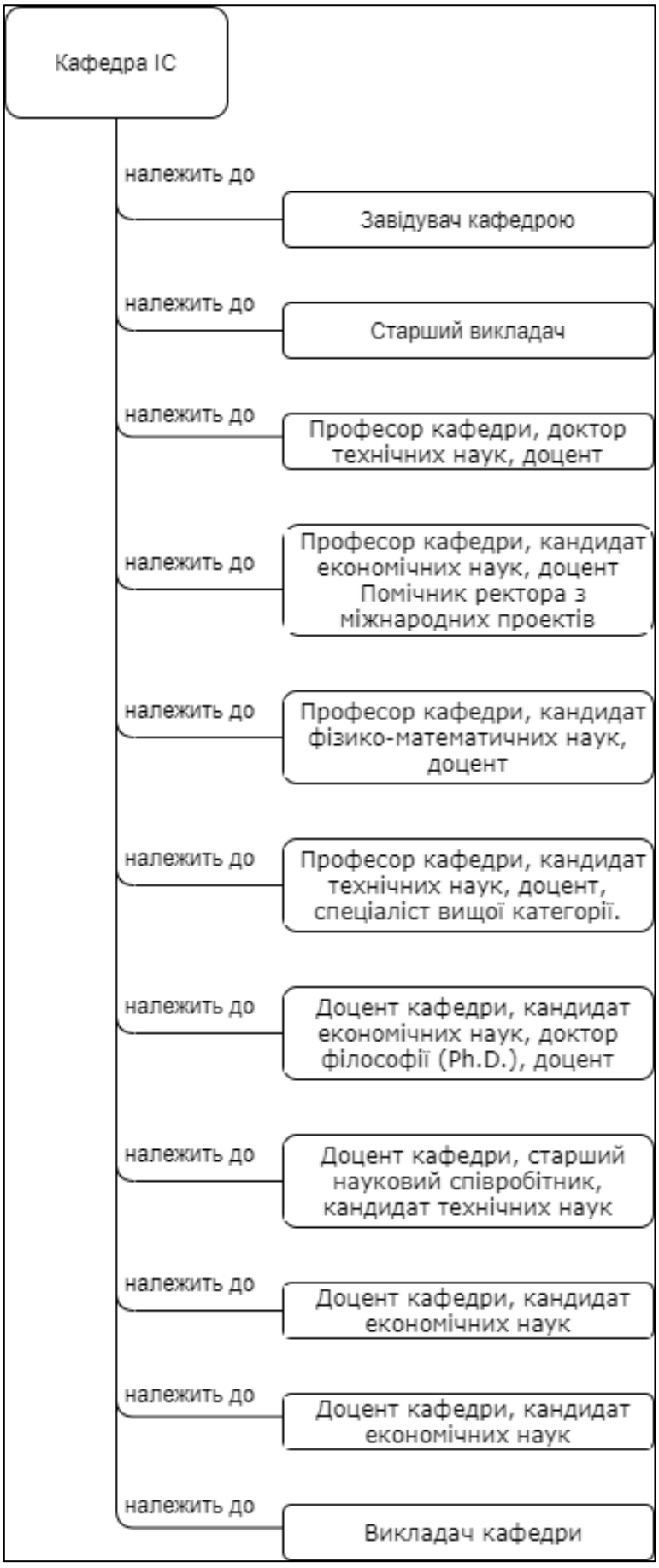

Рис. 1.1. Схема кафедри інформаційних систем

1.2. Огляд літературних джерел

Огляд літератури виконувався на основі статтей з інтернет ресурсів, наукових журналів та конференцій.

Основними джерелами, які найкращим чином розкривають тему даного дослідження є: What is SQL Server Machine Learning Services (Python and R)?, Brien Posey – A closer look at Python-SQL Server 2017 integration, Alison DeNisco Rayome – R vs. Python: Which is a better programming language for data science?, Орельен Ж. – Прикладное машинное обучение с помощью Scikit-Learn и TensorFlow, Федько В. В. – Аналіз даних в SQL Server засобами Python та Peter Zaitsev – MySQL Query Performance Troubleshooting: Resource-Based Approach.

Опрацьований матеріал можна поділити на 2 блоки: загальна інформація про використання машинного навчання в SQL Server та інформація щодо проведення аналізу продуктивності роботи сервера БД.

Ознайомлення із машинним навчанням:

1) What is SQL Server Machine Learning Services (Python and R) [53]?

В статті описується функція SQL Server, яка дає можливість запускати сценарії Python та R з реляційними даними. Наведено приклади, за допомогою яких можна більш детально ознайомитися з можливостями даної функції, перераховано пакети кожної з мов програмування, які можуть бути використанні під час проведення машинного навчання.

2) Brien Posey – A closer look at Python-SQL Server 2017 integration [36].

В статті пояснено з якою метою компанія Microsoft вирішила здійснити інтеграцію мов програмування R, Python та SQL Server; представлені відповіді на питання, які можуть виникнути:

a) яку вигоду користувачі SQL Server отримують від інтеграції Python?

b) яким чином Microsoft здійснює захист баз даних SQL Server від шкідливих сценаріїв Python?

3) Georg Thomas – AI and ML: Why have machine learning in SQL Server at all [38].

У статті автор висловив свою думку з приводу необхідності використання машинного навчання в SQL Server та також визначив переваги такого підходу:

«Із збільшенням обсягу даних вам не потрібно переміщувати (а потім оновлювати) дані. Ви можете обробити їх там, де вони зберігається (в SQL Server). Окрім того ви можете скористатися багатопотоковою архітектурою SQL Server».

4) Alison DeNisco Rayome – R vs. Python: Which is a better programming language for data science [34]?

В статті наведена порівняльна характеристика мов програмування, які найчастіше використовуються при вирішення задач за допомогою машинного навчання. В ході дослідження було виявлено, що обидві мови програмування мають великий набір вбудованих бібліотек та пакетів, які забезпечують точність та якість отриманих результатів.

5) Reena Shaw – The 10 Best Machine Learning Algorithms for Data Science Beginners [47].

В статті розглядаються основні алгоритми машинного навчання, особливості та переваги кожного з них, принципи роботи з ними.

6) Кондрашов Ю. Н. – Анализ данных и машинное обучение на платформе MS SQL Server [11].

Автор розглядає сучасні технології аналізу даних і машинного навчання і способи їх реалізації засобами MS SQL Server. Наводяться передумови появи аналітичних технологій та теоретичні і практичні аспекти використання сховищ даних. Докладно розглядаються інструментальні засоби підготовки даних і алгоритми розв'язання типових задач Data Mining (регресія, класифікація, кластеризація, пошук асоціативних зв'язків, аналіз послідовностей, прогнозування, нейронні мережі) на платформі MS SQL Server з використання MS Excel.

7) How to select algorithms for Azure Machine Learning [40].

У статті наведена порівняльна характеристика алгоритмів машинного навчання. Були визначені наступні показники, які є ключовими під час вибору певного алгоритму: точність, час навчання, лінійність, кількість параметрів, кількість функцій. Необхідно лише чітко встановити цілі, які повинні бути досягнуті в результаті машинного навчання, що призведе до полегшення вибору алгоритму.

8) Шибайкин С. Д., Никулин В. В., Аббакумов А. А. – Анализ применения методов машинного обучения компьютерных систем для повышения защищенности от мошеннических текстов [32].

В статті розглядаються методи машинного навчання, які можуть використовуватися для визначення шахрайських текстів. Для підвищення ефективності роботи даних методів було запропоновано об’єднати їх в ансамблі. Даний спосіб є досить розповсюдженим явищем при вирішення задач машинного навчання. Було проведено аналіз отриманих результатів та зроблено висновок про те, що використання ансамблів для визначення шахрайських

текстів дає більш точні результати у порівнянні з роботою окремих аналізаторів.

9) Орельен Ж. – Прикладное машинное обучение с помощью Scikit-Learn и TensorFlow [20].

В книзі описуються концепції, методи і інструменти, необхідні для реалізації програм, які допоможуть вивчати масиви даних. На основі бібліотек Python Scikit-Learn та TensorFlow побудовано багато прикладів навчання моделі, що дозволяє з легкістю освоїти основні принципи вирішення задач за допомогою машинного навчання.

10) Франсуа Шолле – Глубокое обучение на Python [28].

Автор пропонує більше 30 прикладів програмного коду з детальними коментарями і рекомендаціями, що робить книгу орієнтованою на вирішення практичних задач. У прикладах використовуються фреймворк глибокого навчання Keras, написаний на Python, і бібліотека TensorFlow. Книга складається з двох частин – теоретичної та практичної, що сприяє кращому розумінню нового матеріалу.

11) Хомутов Н. Ю. – Методы повышения эффективности моделей машинного обучения, основанные на различных принципах снижения размерности [29].

У даній роботі описується методика побудови моделей, що використовують вибрані оптимальні комбінації ключових для прогнозування параметрів, побудованих за окремими ознаками досліджуваного об'єкта. Мета даної методики – зниження розмірності нового простору ознак і збільшення підсумкової узагальнюючої здатності моделі. Автор наголошує на тому, що використання ансамблів дозволяє моделям компенсувати помилки один одного, даючи більш якісну підсумкову модель.

12) Коротеев М.В. – Обзор некоторых современных тенденций в технологии машинного обучения [12].

В статті автор розглядає новації в сфері машинного навчання, які можуть мати вплив на розвиток даної галузі. На основі аналізу наукової літератури були висунуті гіпотези щодо тенденцій розвитку машинного навчання та визначені найбільш перспективні напрями дослідження. Було розглянуто такі сучасні технології в машинному навчанні, як використання попередньо навчених моделей, побудова мультизадачних систем, нейроеволюція, проблема створення інтерпретованих моделей.

13) Кафтанников И.Л., Парасич А.В. – Проблемы формирования обучающей выборки в задачах машинного обучения [10].

Автори даної статті наголошують на тому, що правильне формування навчальної вибірки є одним з найголовніших підготовчих етапів вирішення задачі за допомогою машинного навчання, оскільки від даних, на основі яких буде проводить навчання, багато в чому залежить достовірність отриманого результату.

У статті досліджуються можливі проблеми і помилки при формуванні навчальної вибірки, узагальнюється досвід авторів у вирішенні завдань машинного навчання, пропонуються теоретичні моделі для опису явищ, пов'язаних з формуванням безлічі навчальних даних, наводяться методи поліпшення навчальної вибірки.

14) Чибирова М. Э. – Анализ данных и регрессионное моделирование с применением языков программирования Python и R [30].

В даній статті була розглянута методика регресійного аналізу даних та розробки математичних моделей, які в подальшому використовувались для прогнозування результатів. Навчання моделі відбувалося використовуючи метод найменших квадратів. Програмна реалізація даної процедури була розроблена двома мовами програмування – R та Python. В статті також відзначаються основні переваги та недоліки зазначеного методу прогнозування даних та висновки щодо отриманих результатів.

15) Краснянский М. Н., Обухов А. Д., Соломатина Е. М., Воякина А. А. – Сравнительный анализ методов машинного обучения для решения задачи классификации документов научно-образовательного учреждения [13].

Автори даної статті розглядають актуальність задачі класифікації документів з використанням методів машинного навчання. Було розроблено алгоритм, який враховуватиме специфіку документів та визначено підходи, які дозволять підвищити точніть та швидкість класифиікації документів при використанні методів машинного навчання.

16) Жуков Д. А., Клячкин В. Н. – Задачи обеспечения эффективности машинного обучения при диагностике технических объектов [6].

В статті визначаються ознаки, за якими слід оцінювати якість отриманих в ході прогнозування даних. Розглядаються підходи для підвищення точності результатів машинного навчання. В ході дослідження було встановлено, що для забезпечення ефективності машинного навчання необхідно розробити систему, яка буде аналізувати вплив різних факторів на якість класифікації при вихідних даних певного об’єкта та забезпечувати застосування оптимальних підходів для її реалізації.

17) Трифонов Т. В., Андреев И. Е., Глазкова А. В. – Сравнение эффективности методов машнного обучения для тоновой классификации текстов [26].

Автори статті проводять аналіз роботи методів машинного навчання для тонової класифікації текстів та порівнють між собою отримані результати. Таким чином визначається метод, який найкраще підходить для вирішення задач подібного типу.

18) Нигматуллин В. Р., Руднев Н. А. – Использование методов машинного обучения и искусственного интеллекта в химической технологии [18].

В роботі було розглянуто застосування методів машинного навчання і штучного інтелекту для різних завдань хімічної технології, таких як моделювання, автоматизація та оптимізація процесів, контроль якості і безпеки, пошук нових сполук і каталізаторів. Для цих цілей були використані штучні нейронні мережі, метод дерева прийняття рішень, бустінг, регресія, а також їх комбінації. Визначаються методи, які є найбільш ефективними для використання в даній галузі.

19) Горяинов А. Н. – Машинное обучение в логистических и транспортных системах [4].

В роботі була визначена практична цінність застосування машинного навчання в галузі логістики та транспорту, розглядаються можливості та перспективи розвитку. Наведено ряд задач, для вирішення яких буде доречним використання машиного навчання.

20) Мартынова Ю. А. – Выбор источника финансирования методами машинного обучения [14].

Автор розглядає застосування методів машинного навчання на прикладі вибору фінансової установи для обслуговування деякої організацій. Вирішення даної задачі відбувалося з використанням технології нейронних мереж та нечіткої логики. Було проведено оцінку ефективності роботи методів машинного навчання для вирішення задач подібного типу.

21) Федько В. В. – Аналіз даних в SQL Server засобами Python [27].

В статті було детально розглянуто спосіб реалізації методів машинного навчання в SQL Server засобами Machine Lerning. Було проведено порівняльний аналіз мов програмування, які найчастіше використовуються для вирішення подібних задач – R та Python. Розглянуто можливості інструментальних засобів аналізу даних. В результаті дослідження було виявлено, що даний спосіб аналізу даних є досить ефективним та значно полегшує обробку великих об’ємів даних.

22) Гусев А. В. – Перспективы нейронных сетей и глубокого машинного обучения в создании решений для здравоохранения [5].

В роботі автор розглядає перспективи застосування нейронних мереж і глибокого машинного навчання при створенні систем штучного інтелекту для охорони здоров'я. Наведено короткий опис технологій машинного навчання і нейронних мереж. Також було проведено огляд вже реалізованих проектів застосування штучного інтелекту, визначено їх переваги та недоліки, а також спрогнозовано найбільш перспективні напрямки розвитку застосування машинного навчання.

23) Боровский А. А. – Перспективы применения технологий машинного обучения к обработке больших массивов исторических данных [3].

Автор у даній статті дослідив аналітичні можливості сучасних методів машинного навчання і перспективи їх практичного використання для обробки і аналізу великих масивів даних. Було розглянуто різні стратегії застосування машинного навчання з урахуванням особливостей природи даних. Особливу увагу приділено проблемі інтерпретації різних типів результатів, які отримуються у процесі використання алгоритмів машинного навчання, а також можливості розпізнавання трендів та аномалій.

Зроблено висновок про здатність методів машинного навчання ефективно вирішувати великий клас завдань, пов'язаних з аналізом масивів даних, включаючи пошук прихованих залежностей і закономірностей.

Аналіз продуктивності роботи сервера

24) Peter Zaitsev – MySQL Query Performance Troubleshooting: Resource-Based Approach [45].

Автор відзначає, що основні ресурси, які, як правило, є вузьким місцем і обмежують продуктивність системи – це процесор, пам'ять, диск та мережа. Аналіз використання вищезазначених ресурів при роботі з SQL Server проводиться у програмі Percona Monitoring and Management. Даний продукт допомагає зменшити складність, оптимізувати продуктивність та покращити безпеку критично важливих для бізнесу середовищ баз даних незалежно від того, де вони розташовані або розгорнуті.

В статті досить детально описується процес роботи в програмі, відзначаються характеристики кожного з ресурсів, на які слід звернути увану під час проведення аналізу їх використання.

«Мета цих графіків – легко дозволити вам знайти проблемне місце, коли використання ресурсів було великим. Ви можете збільшити його, вибравши

заданий інтервал на графіку та переглянути активність запиту для конкретної діяльності» – зазначає автор.

На мою думку Percona Monitoring and Management – дуже потужний інструмент, який завдяки великій кількості графіків та таблиць дозволить швидко знайти проблеми, що можуть сповільнити обробку даних та вирішити їх.

25) Baron Schwartz – 10 essential performance tips for MySQL [35].

«Типові помилки лежать в основі більшості проблем із продуктивністю MySQL. Щоб ваш сервер працював на найвищій швидкості, забезпечуючи стабільну роботу, важливо усунути ці помилки, які часто непомітні відразу через якусь дрібницю у вашому навантаженні або конфігурації програмного забезпечення».

В статті наведено 10 порад, які допоможуть підвищити продуктивність роботи з сервером бази даних. Як і в попередній статті, тут визначені ті ж самі характеристики, які є ключовими при аналізі продуктивності системи, наведено приклад програм, за допомогою яких можливо проводити аналіз: MySQL Enterprise Monitor’s query analyzer або фреймворк компанії Percona – pt-query-digest.

26) Matt Watson – SQL Performance Tuning: 7 Practical Tips for Developers [43].

Автор розповідає про те, яким чином можна збільшити швидкість виконання запитів до бази даних, контролювати цей процес та відстежувати проблеми, які гальмують отримання результатів. Було проведено огляд сторонньої програми Retrace, яка надає можливість відстежувати продуктивність та статистику виконання SQL-запитів.

27) Шелухин О.И., Симонян А. Г., Ванюшина А. В. – Влияние структуры обучающей выборки на эффективность классификации приложений трафика методами машинного обучения [31].

В статті проводиться аналіз впливу типу вхідних даних на результат прогнозування. Дослідження відбувалося на основі інформації про споживання трафіку деякого оператору зв’язку. Вхідні дані подавалися у двох виглядах: байтовому та потоковому. Важливий вплив мав розмір вхідних даних: чим більша була вибірка, на основі якої проводилось навчання моделі, тим точніше були отримані результати. В ході дослідження було зроблено висновок, що ефективність класифікації за потоками була значно вищою, ніж за байтами.

28) Белов Ю. С., Козина А. В., Гришунов С. С. – Применение критерия «сигнал/шум» для определения эффективности методов машинного обучения [2].

В статті було розглянуті основні проблеми, які можуть виникати при використанні методів машинного навчання. Однією з найважливіших – є проблема оцінки ефективності роботи алгоритмів. Описано використання критерію «сигнал/шум», який дозволяє оцінити наскільки правильно було метод машинного навчання, спосіб його розрахунку. Чим вище значення даного критерію, тим точніше отримані дані.

29) Аксютина Е. М., Белов Ю. С. – Обзор архитектур и методов машинного обучения для анализа больших данных [1].

У статті були розглянуті основні архітектури, які використовуються для аналізу великих даних, їх особливості та обмеження, визначені вимоги до методів машинного навчання, виконання яких дозволить застосовувати їх для аналізу великих даних. У висновку наводиться порівняння архітектур і визначаються перспективи використання машинного навчання для аналізу великих даних.

30) Иванов О. Ю. – Использование методов машинного обучения для повышения эффективности систем управления базами данных [7].

У даній роботі розглядається пристрій оптимізатора запитів реляційних систем управління базами даних і пропонується метод усунення виявлених недоліків. Проводилось теоретичне та експеримтентальне дослідження даного методу, в ході якого було виявлено, що даний метод істотно збільшує точність передбачення кількості кортежів, що призводить до збільшення продуктивності систем управління базами даних.

Отже, аналізуючи літературні джерела, наведені вище, можна зробити висновок про широкий спектр галузей, в яких можуть застосовуватися методи машинного навчання для вирішення різних типів задач. Важливим підготовчим етапом – є вміння правильно сформувати вхідні дані, які будуть основою для навчання моделі, віднайти ознаки, які мають найбільший вплив на навчальну вибірку. Проте під час аналізу літератури не було знайдено матеріалів, які надають кількісну оцінку ефективності застосування SQL Server ML Services. Дослідженя цього питання розглядається в даній дипломній роботі.

Підсумовуючи представлену в розділі 1 інформацію можна сформувати цілі виконання даної роботи.

Мета дослідження: вивчення засобів вирішення задач в Machine Learning за допомогою SQL Server Machine Learning Services для визначення найбільш ефективного методу їх програмної реалізації.

Завдання дослідження:

1) Ознайомитися з існуючими програмними засобами, які використовуються для машинного навчання.
2) Підготувати комп’ютерну систему, на якій буде проводитись дослідження.
3) Підготувати дані, що будуть основою для навчання моделі, яка в подальшому буде використовуватися для отримання прогнозованих даних.
4) Провести обробку даних з використанням методів Machine Learning та визначити найбільш ефективний спосіб роботи з даними.
5) Проаналізувати отримані результати.

2. ТЕОРЕТИЧНЕ ТА МЕТОДИЧНЕ ДОСЛІДЖЕННЯ ВИРІШЕННЯ ЗАДАЧІ «ЕФЕКТИВНІСТЬ РОЗВ'ЯЗУВАННЯ ЗАДАЧ MACHINE LEARNING НА ОСНОВІ ВБУДОВАНИХ МОВНИХ ЗАСОБІВ SQL SERVER».

2.1. Опис концепції вирішення поставленої проблеми

Сфера застосувань машинного навчання постійно розширюється. Інформатизація призводить до накопичення величезних обсягів даних в науці, виробництві, бізнесі, транспорті, охороні здоров'я. Виникаючі при цьому завдання прогнозування, управління та прийняття рішень часто зводяться до навчання за прецедентами.

Машинне навчання передбачає автоматичне навчання системи без втручання в цей процес людини. Вхідні дані для проведення процесу навчання, зазвичай, подаються у вигляді матриці об'єкт-ознака, між якими існує певна залежність, але вона невідома. Суть машинного навчання полягає у пошуку цієї залежності, тобто необхідно побудувати алгоритм, який зможе на основі інформації про об'єкт досить точно класифікувати можливу ознаку.

Використання засобів машинного навчання для вирішення певних задач стрімко набуває популярності у різних сферах діяльності.

MIT Technology Review і Google Cloud провели спільне дослідження на тему «Машинне навчання: новий спосіб отримати конкурентну перевагу» [42]. Було опитано 375 кваліфікованих респондентів з різних країн світу, які працюють в дрібних і великих компаніях з різних галузей (промисловість, послуги, фінанси). В результаті дослідження з'ясувалося, що 60% компаній вже використовують машинне навчання (ML), а в третини з них ця технологія перейшла зі стадії інноваційної в стадію зрілості. Більш того, 26% компаній вже отримують за рахунок ML конкурентну перевагу. Чверть компаній інвестують в ML понад 15% від коштів, спрямованих на розвиток IT, і в значній мірі повертають зроблені інвестиції.

На рис. 2.1. представлено динаміку популярності пошукових запитів Google за останні 10 років на основі даних Google Trends [39]. Як бачимо, починаючи з 2016 року зацікавленість в машинному навчання стрімко почала зростати.

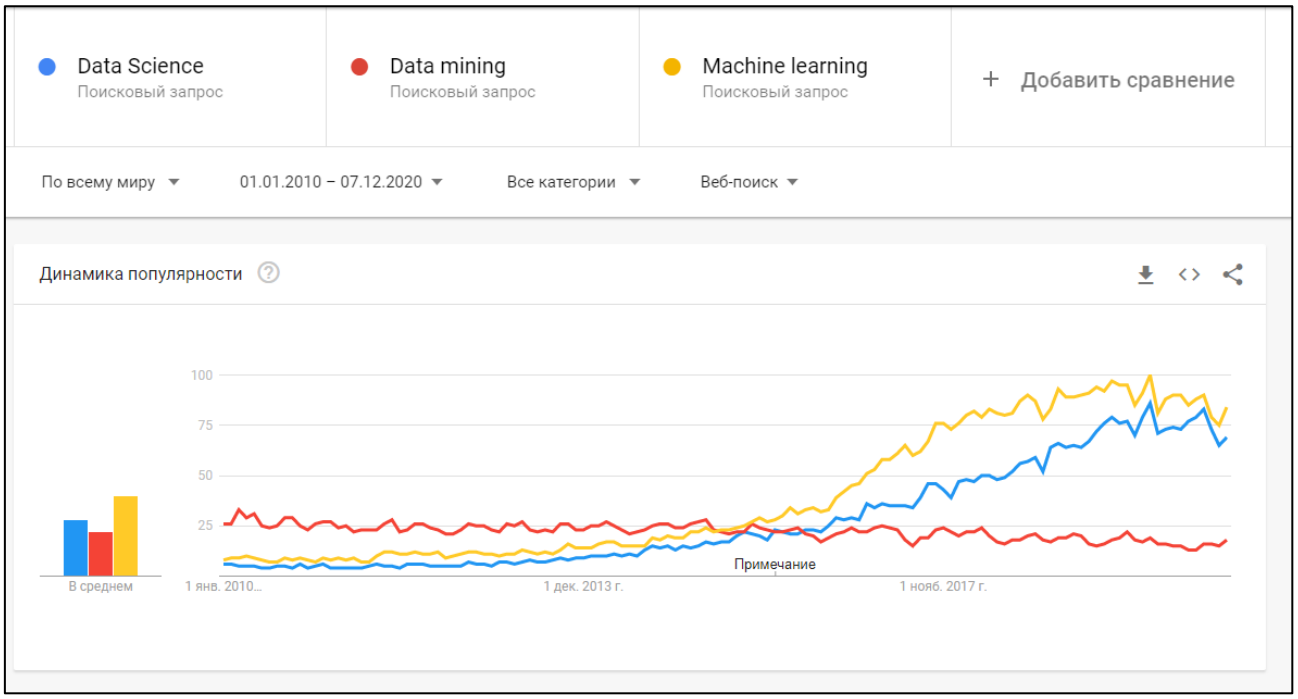

Рис. 2.1. Динаміка популярності пошукових запитів в Google, пов’язаних зі сферою машинного навчання за останні 10 років

Машинне навчання і, зокрема, нейронні мережі доцільно використовувати для вирішення бізнес-завдань у випадках, коли:

- накопичено велику кількість різних даних, але програми для їх обробки і систематизації відсутні;
- наявні дані спотворені, не повні або не систематизовані;
- дані настільки різні, що важко виявити зв'язки і закономірності, що існують між ними [37].

Бізнес-задачі, які можуть вирішуватися засобами машинного навчання і нейронних мереж:

- прогнозування: попиту, обсягу продажів, наповнення складу, завантаження устаткування і інших ресурсів, подальшого розвитку підприємства і т.п.;
- виявлення: тенденцій, прихованих взаємозв'язків, аномалій, повторюваних елементів і т.п.;
- розпізнавання: фото-, відео-, аудіоконтенту, спроб шахрайства, брехні, внутрішніх загроз, зовнішніх атак на систему безпеки і т.п.;
- автоматизація: роботи операторів в онлайн-чатах, телефонних операторів і т.п.;
- класифікація: аналіз складу покупців, клієнтів, замовників і сегментація їх за різними параметрами;

- кластеризація: класифікація за параметрами, які з самого початку не були відомі;
- розробка: чат-ботів [16].

Вибір методу машинного навчання залежить від багатьох чинників: об’єм вхідних даних, їх якість, час, за який буде відбуватися прогнозування даних. Модифікувати вхідні дані задля більш ефективного прогнозування не слід, оскільки при неправильному редагуванні даних можна отримати результат, який не буде відповідати дійсності. Щодо часу прогнозування – його можна корегувати в залежності від способу роботи з даними.

Можливі 2 способи: обробка даних на сервері, на якому зберігаються вхідні дані та класичний підхід – дані завантажуються з сервера на комп’ютер і потім обробляються за допомогою певного методу. Оскільки робота з даними відбувається по-різному, то й час, необхідний для отримання результатів, може відрізнятися.

Також слід відзначити, що програмна реалізація може бути виконана різними мовами програмування. В кожної з них є свій набір бібліотек, які будуть використовуватися при прогнозуванні результатів. Цей факт також може впливати на ефективність обробки даних.

В теорії виконання прогнозування на сервері має відбуватися швидше, оскільки не потрібно витрачати час на завантаження даних, проте це припущення може виявитися хибним.

Обробка даних на сервері буде виконуватися засобами Machine Learning Services, яка надає можливість запускати скрипти, написані мовами програмування R та Python.

Дослідження буде проводись для двох типів розташування сервера – локального (SQL Server встановлюється на комп’ютер користувача і там створюється база даних) та віддаленого (БД створюється в хмарній платформі). В якості хмарної платформи буде використовуватися Microsoft Azure.

В ході проведення дослідження буде виявлено який із способів обробки даних є найбільш ефективним та потребує меншого часу. Також з’ясуємо як розташування сервера впливає на час роботи з даними.

Постановка задачі

Перевірка ефективності обробки даних проводилася на основі вирішення задачі прогнозування методами ML. Мета – на основі інсуючої інформації про кількість прокатів лиж за попередні роки спрогнозувати кількість прокатів за деякий майбутній період [46]. Результати прогнозування будуть корисними для власників компаній з прокату лиж оскільки це дасть їм змогу оцінити обсяги

потенційних клієнтів та завчасно підготувати небхідне обладнання, кількість робочого персоналу та вдале розташування пунктів прокату.

Для проведення дослідження було взято базу даних, яка містить інформацію про кількість прокатів лиж за 2013 – 2015 роки. Таблиця містить 453 рядки. База даних була відтворена з резервної копії, яку можна завантажити за посиланням [21]. На рис. 2.2 представлена стисла інформація про дані таблиці.

	Year	Month	Day	RentalCount	WeekDay	Holiday	Snow
1	2014	1	20	445	2	1	0
2	2014	2	13	40	5	0	0
3	2013	3	10	456	1	0	0
4	2014	3	31	38	2	0	0
5	2014	4	24	23	5	0	0
6	2015	2	11	42	4	0	0
7	2013	4	28	310	1	0	0
8	2014	3	8	240	7	0	0
9	2013	4	5	22	6	0	0
10	2015	3	29	360	1	0	0
11	2015	4	22	20	4	0	0
12	2014	4	1	36	3	0	1
13	2015	3	6	42	6	0	0
14	2014	1	26	729	1	0	1
15	2013	1	30	55	4	0	1
16	2013	3	4	39	2	0	1
17	2015	2	28	405	7	0	1
18	2015	1	12	38	2	0	1

Рис. 2.2. Дані таблиці rental_data

Зміст даних таблиці наступний:

- Year – рік, Month – місяць, Day – день прокату;
- RentalCount – кількість прокатів;
- WeekDay – день тижня;
- Holiday – 1 – святковий день, 0 – звичайний;
- Show – 1 – йшов сніг в цей день, 0 – ні.

Оскільки маємо невелику кількість даних є потреба у програмній реалізації генерації даних, які в подальшому будуть використовуватися для аналізу ефективності використання різних способів їх обробки. Детальний опис процесу генерації даних представлено в розділі 3.

2.2. Сфери застосування Machine Learning

Одним з прикладів важливості ефективної обробки даних засобами машинного навчання є його застосування у виробництві. Завдяки оперативній роботі системи буде змога досить швидко виявити дефекти в роботі пристроїв та вчасно їх замінити.

Іншим прикладом, який найкращим чином відображає потужність можливостей застосування машинного навчання є продукт компанії IBM – Watson, який розуміє питання, сформульовані на звичайній мові, і знаходить на них відповіді за допомогою штучного інтелекту. Даний продукт вже набув широкого застосування в медицині. Провідні лікарі всіх країн світу звертаються за допомогою до Watson аби на ранніх стадіях діагностувати та вилікувати онкологічні захворювання. В якості вихідних даних в пам'ять суперкомп'ютера завантажено понад 600 тисяч медичних висновків і діагнозів, 2 мільйони сторінок текстів, взятих з 42 медичних журналів і результатів клінічних випробувань в області онкології. Завдяки високій потужності Watson може «проаналізувати» 1,5 мільйона записів з історій хвороби різних пацієнтів і, грунтуючись на даних з історій успішної боротьби з подібними захворюваннями, виявити найбільш відповідні методи лікування в кожному конкретному випадку [24, 51].

Звісно, звичайній людині виконати подібну роботу не під силу, для проведення подібного аналізу могли б знадобитися роки, що не допустимо в даному випадку.

Ще один яскравий приклад застосування ML – робота з транспортними системами. Всі показники транспортного засобу моніторяться системою, яка миттєво проводить їх аналіз. У разі виявлення занижених або навпаки, завищених, показників система одразу подає сигнал на транспортний засіб, що дозволить вчасно зреагувати та віднайти причину неполадки і уникнути подальших проблем у керуванні транспортним засобом.

В зарубіжних країнах великої популярності набуває використання машинного навчання для аналізу дорожного стану та виявлення ділянок дороги з високим виникненням ДТП. Дані автоматично збираються з камер відеонагляду, встановлених на перехрестях, небезпечних поворотах та світлофорах, а також зовнішніх факторів (опади, туман, освітлення і температура). Алгоритми машинного навчання (Machine Learning) аналізують погодні та дорожні умови (ширину і зміну пропускної здатності ділянки дороги, середній бал заторів в місті і швидкість потоку). Якщо, з урахуванням

поточної транспортної ситуації і погоди, певна ділянка відноситься до критичних (аварія в цьому місці призведе до сильних заторів), то система присвоює даній ситуації найвищий пріоритет у разі виникнення аварії та виклику служб швидкої допомоги та ДПС. Також інформація транслюється у всі навігаційні системи, що дозволить водіям завчасно оцінити маршрут і уникнути потрапляння в затори [8].

Як бачимо, машинне навчання має широкий спектр застосування і головним показником, який визначає ефективність вирішення задач засобами ML – є час. Він повинен бути мінімальним.

2.3. Розгляд та опис методів розв’язання завдань дослідження

В ході дослідження необхідно створити 2 програми. В результаті повинні бути отримані однакові результати прогнозування, але перша програма буде виконувати обробку даних безпосередньо на сервері (використовуються засоби SQL Server Machine Learning Services), а в іншій – дані будуть завантажуватися з сервера і потім застосовуватимуться методи їх обробки. Методи обробки даних повинні бути написані мовою програмування Python або R. В даному випадку буде використовуватися метод лінійної регресії.

Схема виконання дослідження представлена на рис. 2.3.

Дослідження виконуватиметься у декілька етапів, змінюючи об’єм вхідних даних та аналізуючи, як ці зміни впливають на час проведення прогнозування. На кожному етапі експеримент повторюватиметься декілька разів та буде розраховано середній час, необхідний для обробки даних, за методом визначення арифметичного середнього.

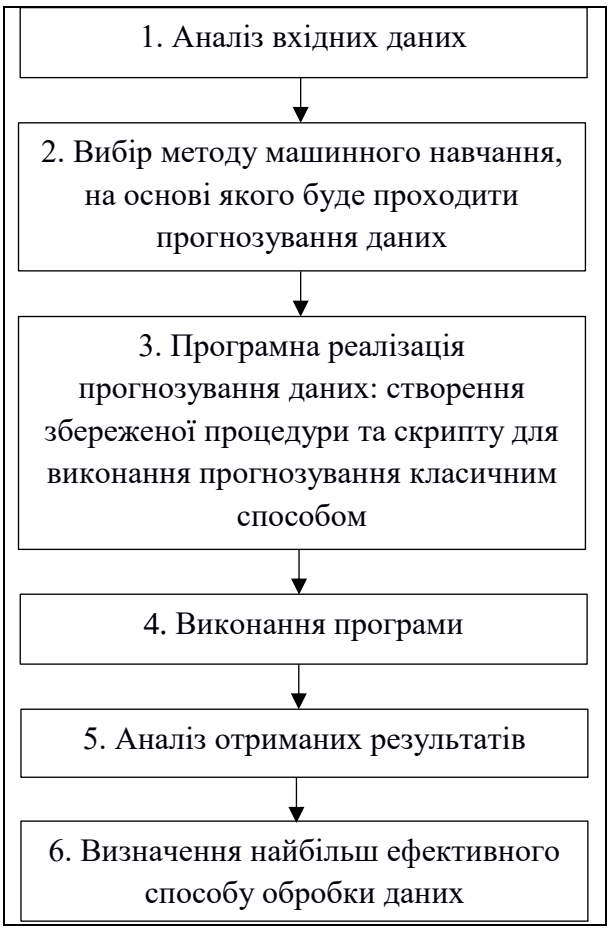

Рис. 2.3. Схема виконання дослідження

Зазвичай при вирішенні задач методами машинного навчання в якості вхідних даних використовуються бази даних з мільонами, мільярдами записів ключової інформації і правильна організація роботи з сервером є важливим етапом, оскільки від цього залежить наскільки швидко будуть отримані кінцеві результати, на основі яких в майбутньому будуть прийматися рішення щодо подальшої роботи.

Як вже зазначалось, основним ресурсом, який буде досліджуватися в ході проведення експериментів, є час. Для оцінки часу використовуватимуться функції timeit.default_timer() – для коду, написаного мовою Python, та Sys.time() – для R. Алгоритми оцінки часу для різних способів вирішення задач ML представлено на рис. 2.4 – 2.5.

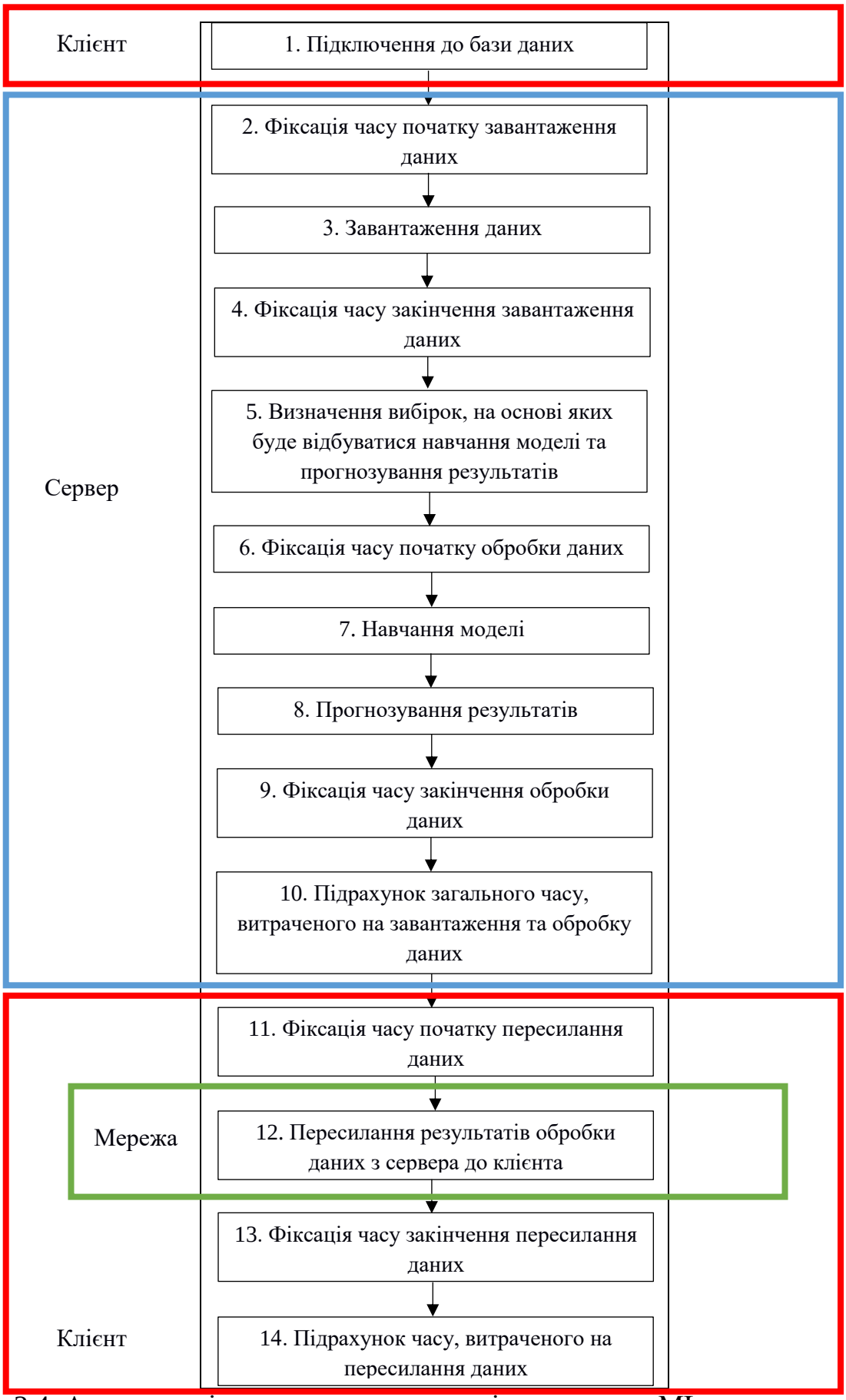

Рис. 2.4. Алгоритм підрахунку часу для вирішення задач ML з використанням вбудованих мовних засобів SQL Server

Оскільки для завантаження вхідних даних та їх обробки використовуються вбудовані мовні засоби SQL Server, то етапи 2-10 відбуваються на сервері бази даних. Етапи 11 – 14 відбуваються на комп'ютері клієнта.

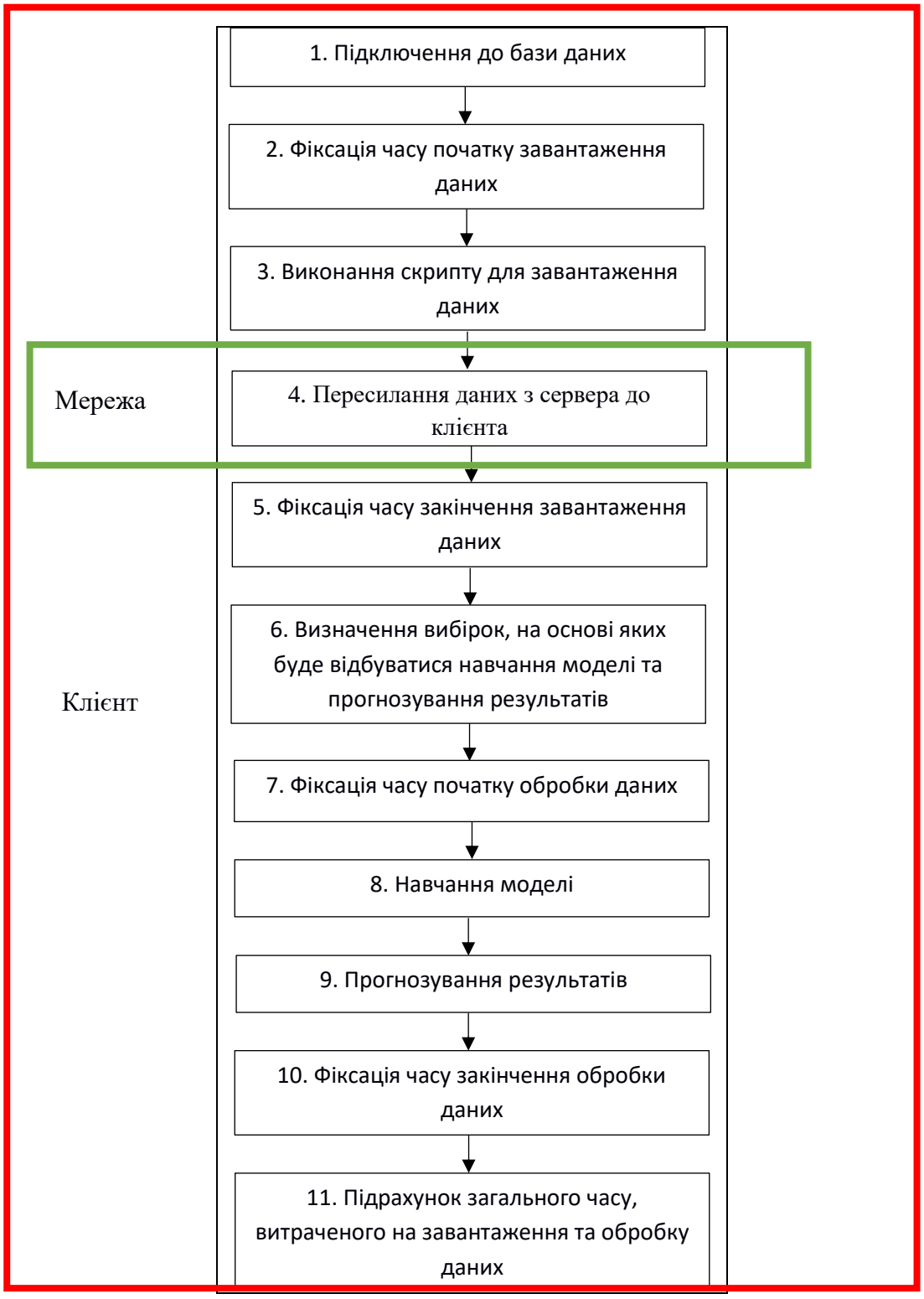

Рис. 2.5. Алгоритм підрахунку часу для вирішення задач ML з використанням класичного підходу

Оскільки використовується класичний спосіб роботи з даними, то всі етапи, окрім 4, відбуваються на комп’ютері клієнта. Етапи підрахунку часу відсутні з тої причини, що всі дані вже завантажені на комп’ютер ще на 3 етапі, тож час на пересилку результів в цьому випадку не витрачається.

2.4. Розгляд та опис моделей, що пропонуються для розв’язання завдань дослідження

Застосування методів математичної статистики (статистичних методів) для обробки результатів дослідження є обов'язковою вимогою, оскільки їх використання допоможе оцінити якість та достовірніть отриманих результатів.

Методами статистичної обробки результатів дослідження називаються математичні прийоми, формули, способи кількісних розрахунків, за допомогою яких показники, одержувані в ході дослідження, можна узагальнювати, приводити в систему, виявляючи приховані в них закономірності [17].

Усі методи статистичного аналізу можна умовно поділити на первинні і вторинні. Первинними називаються методи, за допомогою яких можна отримати показники, які безпосередньо відображають результати проведених в експерименті вимірювань. До первинних методів статистичної обробки відносять визначення:

- середнього арифметичного;
- дисперсії;
- середньоквадратичного відхилення;
- розмаху вибірки;
- моди;
- медіани.

Середнє арифметичне значення – це відношення суми всіх значень даних до числа доданків. Середнє значення як статистичний показник являє собою середню оцінку досліджуваного в експерименті критерію. Середнє арифметичне значення визначається за формулою 2.1.

$$x = \frac{1}{n}\sum_{k=1}^{n} x_k \qquad (2.1)$$

де x – вибіркова середня величина або середнє арифметичне значення по вибірці;

n – кількість показників вибірки, на основі яких обчислюється середня величина;

x_k – значення показників у окремих дослідженнях. Всього таких показників n, тому індекс k даної змінної приймає значення від 1 до n;

∑ – прийнятий в математиці знак підсумовування величин тих змінних, які знаходяться праворуч від цього знака.

Дисперсія – це міра розкиду даних щодо середнього значення. Чим більше дисперсія, тим більше відхилення або розкид даних. Її визначають для того, щоб можна було відрізняти величини одна від одної, які мають однакову середню, але різний розкид. Дисперсія визначається за формулою 2.2.

$$S^2 = \frac{1}{n}\sum_{k=1}^{n}(x_k - \bar{x})^2 \quad (2.2)$$

де S^2 – дисперсія;

n – кількість показників вибірки, на основі яких обчислюється дисперсія;

$\sum_{k=1}^{n}(x_k - \bar{x})^2$ – вираз, що означає, що для всіх k, від першого до останнього в даній вибірці необхідно обчислити різниці між і середніми значеннями, звести ці різниці в квадрат і підсумувати;

∑ – прийнятий в математиці знак підсумовування величин тих змінних, які знаходяться праворуч від цього знака.

Середньоквадратичне відхилення – в теорії ймовірностей і статистиці найбільш поширений показник розсіювання значень випадкової величини щодо її математичного очікування. Середньоквадратичне відхилення визначається за формулою 2.3.

$$s = \sqrt{S^2} = \sqrt{\frac{1}{n}\sum_{k=1}^{n}(x_k - \bar{x})^2} \quad (2.3)$$

де s – середньоквадратичне відхилення;

S^2 – дисперсія.

Вторинними називають методи статистичної обробки, за допомогою яких на базі первинних даних виявляють приховані в них статистичні закономірності. До вторинних методів статистичної обробки відносять:

- кореляційний аналіз;
- регресійний аналіз;
- факторний аналіз;
- методи порівняння даних двох або декількох вибірок [23].

Кореляційний аналіз.

Метод, за допомогою якого з'ясовується зв'язок або пряма залежність між двома рядами експериментальних даних. Він показує, яким чином одна зміна даних впливає на інші.

Є кілька різновидів даного методу: лінійний, ранговий, парний і множинний. Лінійний кореляційний аналіз дозволяє встановлювати прямі зв'язки між змінними величинами з їх абсолютними значеннями. Ці зв'язки графічно виражаються прямою лінією, звідси назва «лінійний».

Коефіцієнт лінійної кореляції визначається за формулою 2.4.

$$r_{xy} = \frac{\sum_{k=1}^{n}[(x_k - \bar{x})^2 (y_k - \bar{y})]}{n\sqrt{\bar{S}_x{}^2 \bar{S}_y{}^2}} \quad (2.4)$$

де r_{xy} – коефіцієнт лінійної кореляції;

$\bar{x}, \bar{y}$ – середні вибіркові значення порівнюваних величин;

x_k, y_k – приватні вибіркові значення порівнюваних величин;

n – загальне число величин в порівнюваних рядах показників;

$\bar{S}_x{}^2 \bar{S}_y{}^2$ – дисперсії, відхилення порівнюваних величин від середніх значень.

Коефіцієнт кореляції приймає значення від -1 до +1. Кореляція може бути позитивною та негативною (можлива також ситуація відсутності статистичного зв'язку – наприклад, для незалежних випадкових величин). Від'ємна кореляція – кореляція, при якій збільшення однієї змінної пов'язане зі зменшенням іншої, при цьому коефіцієнт кореляції від'ємний. Додатна кореляція – кореляція, при якій збільшення однієї змінної пов'язане зі збільшенням іншої, при цьому коефіцієнт кореляції додатний.

Зв'язки між ознаками можуть бути слабкими і сильними (тісними). Їх критерії оцінюються за шкалою Чеддока (табл. 2.1).

Таблиця 2.1

Шкала Чеддока

Кількісна міра кореляції	Якісна міра кореляції
0.1 – 0.3	Слабка
0.3 – 0.5	Помірна
0.5 – 0.7	Помітна
0.7 – 0.9	Висока
0.9 – 1	Дуже висока

Для перевірки значимості коефіцієнта кореляції зазвичай використовують критерій Стьюдента.

Критерій Стьюдента (t-критерій) – це статистичний метод, який дозволяє порівнювати середні значення двох вибірок і на основі результатів тесту робити висновок про те, чи розрізняються вони один від одного статистично чи ні. На основі коефіцієнта кореляції розраховується теоретичне значення t (формула 2.5) та порівнюється із табличним [25].

$$t_{\text{теор}} = r\frac{\sqrt{n-2}}{\sqrt{1-r^2}} \quad (2.5)$$

де $t_{\text{теор}}$ – теоретичне значення критерію Стьюдента;

r – коефіцієнт кореляції;

n – загальне число величин в порівнюваних рядах показників.

Регресійний аналіз.

У регресійному аналізі моделюється взаємозв'язок однієї випадкової змінної від однієї або декількох інших випадкових змінних. При цьому, перша змінна називається залежною, а решта – незалежними. Вибір або призначення залежних і незалежних змінних є довільним (умовним) і здійснюється дослідником в залежності від розв'язуваного їм завдання. Незалежні змінні називаються факторами, регресорами або предикторами, а залежна змінна – результативною ознакою, або відгуком [19].

Якщо число предикторів дорівнює 1, регресію називають простою, або однофакторною, якщо число предикторів більше 1 – множинною або багатофакторною. У загальному випадку регресійну модель можна записати в такий спосіб:

$$y = f(x_1, x_2, \dots, x_n) \quad (2.6)$$

де y – залежна змінна;

xi (i = 1, …, n) – предиктори (фактори);

n – число предикторів.

Методи порівняння даних.

Суть полягає у порівнянні результатів, які отримані в ході дослідження та їх оцінці, виявленні способу, за домогою якого були отримані кращі результати.

Для визначення часу, який потребується на обробку даних різними способами, буде використовуватися первинний метод статистичної обробки – визначення середнього арифметичного. Використання даного методу необхідне, оскільки для різного об’єму вхідних даних буде проводитися декілька експериментів і таким чином можна буде визначити середній час обробки даних.

Для пошуку найбільш ефективного способу буде використовуватися вторинний метод статистичної обробки – метод порівняння даних. На основі даних, отриманих в ході проведення експериментів, буде порівнюватися час, необхідний на завантаження, обробку та пересилку інформації. В результаті чого буде визначено найбільш ефективний спосіб роботи з даними.

2.5. Методичне забезпечення щодо організації проведення досліджень

Для вирішення задач машинного навчання існує багато систем, які можуть реалізувати дану функціональність: RStudio, Visual Studio, PyCharm, Spyder. В роботі проводився аналіз можливостей машинного навчання засобами SQL Server Management Studio та допоміжних програм для компіляції вихідного коду.

SQL Server Management Studio (SSMS) – це інтегроване середовище для управління будь-якою інфраструктурою SQL. SSMS використовується для доступу, налаштування та адміністрування всіх компонентів SQL Server, бази даних SQL Azure і сховища даних SQL, а також управління ними. Середовище SSMS надає єдину повнофункціональну службову програму, яка поєднує в собі велику групу графічних інструментів з рядом відмінних редакторів сценаріїв для доступу до служби SQL Server для розробників і адміністраторів баз даних всіх професійних рівнів [53].

Однією з особливостей SQL Server, яка з’явилася у 2016 році є підтримка служб машинного навчання (Machine Learning Services). Це функція дозволяє компілювати скрипти, написані мовою R або Python, в яких використовуються реляційні дані.

R – це мова та середовище для статистичних обчислень та графіки. Це проект GNU, подібний до мови та середовища S. R надає широкий спектр статистичних (лінійне та нелінійне моделювання, класичні статистичні тести, аналіз часових рядів, класифікація, кластеризація) та графічних прийомів, і є дуже розширюваним. Мова S часто є засобом вибору для дослідження

статистичної методології, а R забезпечує відкритий вихідний шлях до участі в цій діяльності. Однією з сильних сторін R є простота написання коду [54].

Python – високорівнева мова програмування загального призначення, орієнтована на підвищення продуктивності розробника і читання коду. Розробляється за ліцензією відкритого коду, затвердженою OSI, що робить її вільною для використання та розповсюдження, навіть для комерційного використання [33].

Служби машинного навчання розроблені на підставі платформ і пакетів, які використовують відкритий вихідний код, а також пакетів Microsoft Python і R для здійснення прогнозної аналітики і машинного навчання. Дані не переміщаються за межі SQL Server або по мережі, виконання скриптів відбувається безпосередньо в базі даних, що може істотно зменшити час виконання скрипта при роботі з великими даними.

Скрипти можуть бути використані, якщо необхідно підготувати або очистити дані, розробити функції для виконання будь-якої задачі, провести навчання, оцінку або розгортання моделі машинного навчання в базі даних [53].

Перед початком проведення дослідження необхідно налаштувати комп'ютерну систему, встановивши відповідні програмні засоби на комп'ютері клієнта:

1. SQL Server 2017.

Оскільки вхідна інформація для прогнозування даних зберігається на сервері, необхідно встановити SQL Server для можливості доступу до даних.

2. SQL Server Management Studio 2017 – утиліта з Microsoft SQL Server для конфігурації, управління і адміністрування всіх компонентів Microsoft SQL Server.

Щоб мати можливість працювати зі скриптами, необхідно виконати установку відповідних мовних пакетів (рис. 2.6).

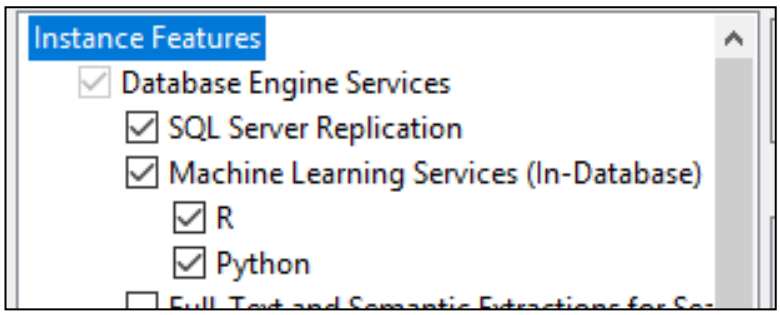

Рис. 2.6. Установка мовних пакетів для використання R, Phyton

Наступним кроком є налаштування SQL Server Management Studio (SSMS). Після запуску команд, наведених нижче, компіляція скриптів повинна відбуватися без помилок:

sp_configure 'external scripts enabled', 1;

RECONFIGURE WITH OVERRIDE.

Аби мати змогу проподити експерименти з даними, які зберігаються на віддаленому сервері (Microsoft Azure) необхідно налаштувати віртуальну машину з SQL Server в хмарній платформі. Порядок налаштування наступний:

1) На порталі Azure [44] в меню вибрати Azure SQL.

2) Натиснути + Додати, щоб відкрити сторінку Вибір варіанту розгортання SQL.

3) Ввести 2017 в поле пошуку образів SQL Server на панелі віртуальні машини SQL, а потім вибрати безкоштовна SQL Server Ліцензія: SQL Server 2017 Developer на Windows Server 2016 (Free SQL Server License: SQL Server 2017 Developer on Windows Server 2016) із списку (рис. 2.7) та натиснути кнопку Створити.

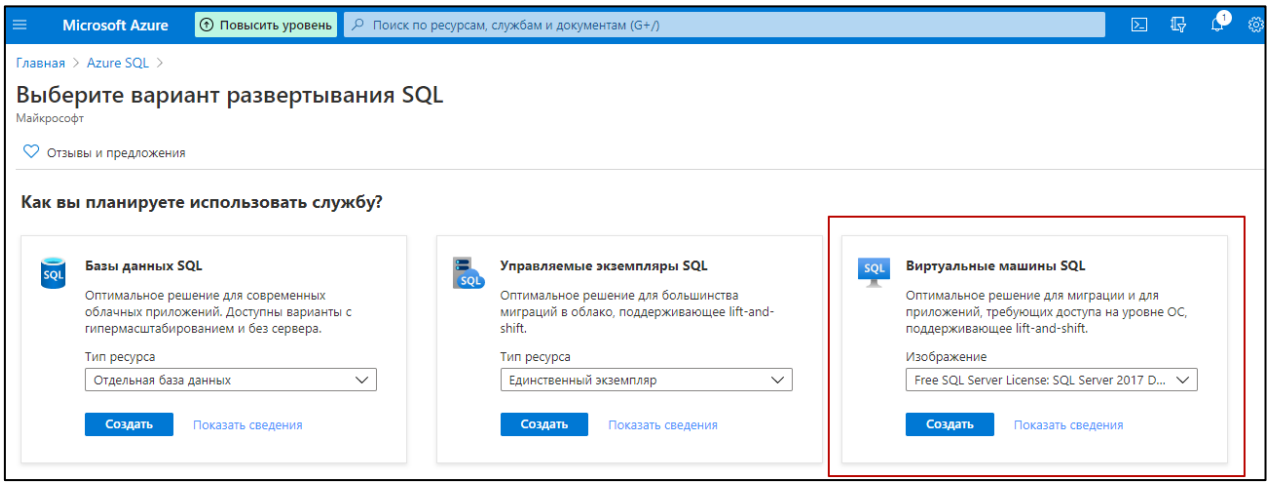

Рис. 2.7. Вибір версії SQL Server для віртуальної машини

4) Заповнити необхідну інформацію, представлену на рис. 2.8. – 2.10.

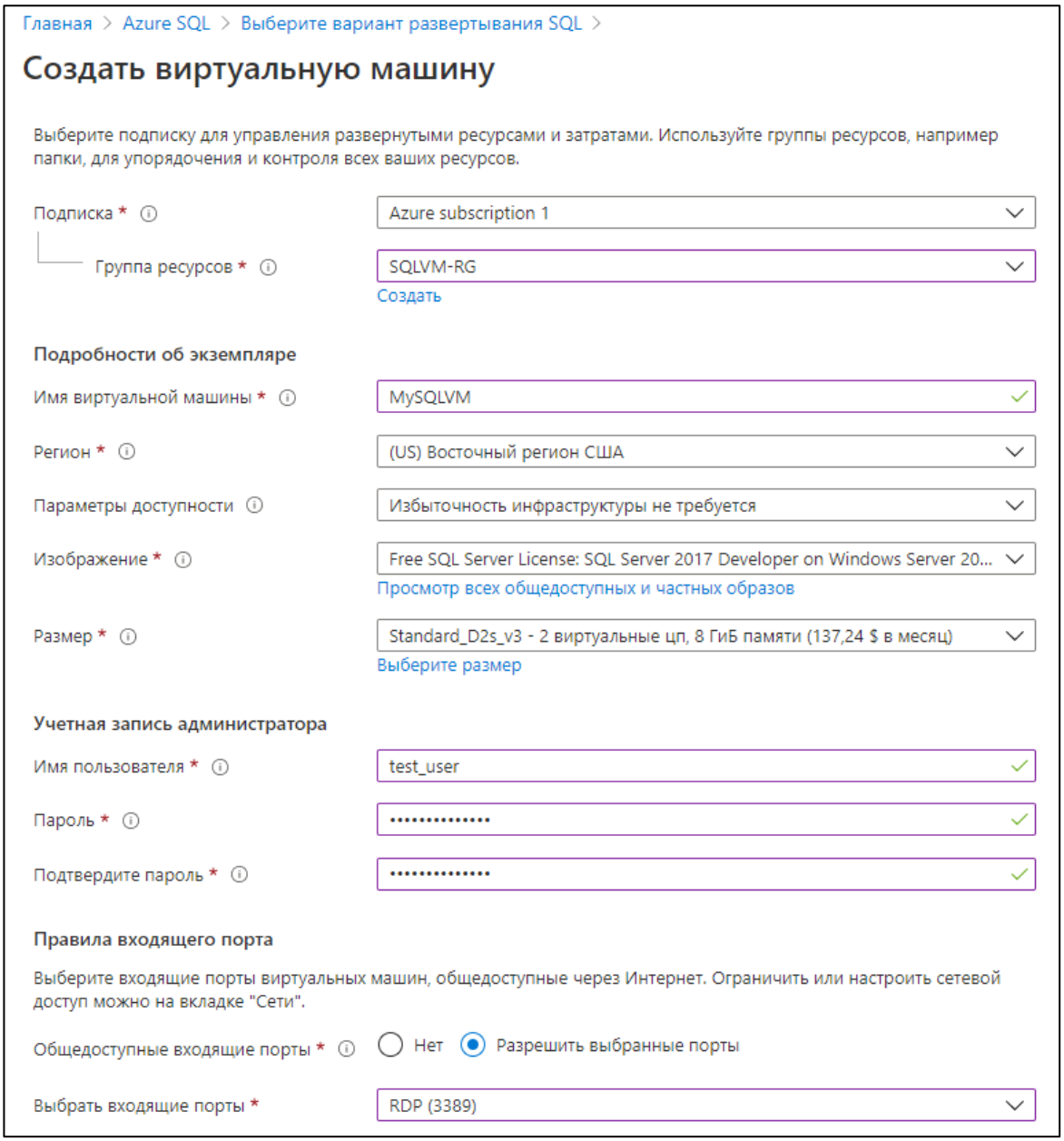

Рис. 2.8. Заповнення основної інформації

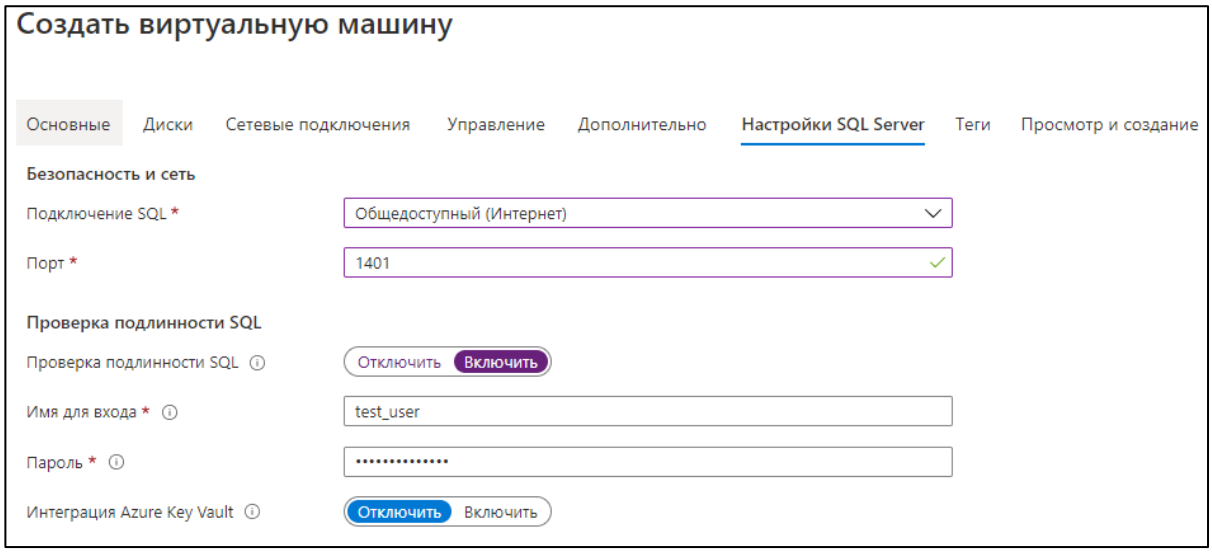

Рис. 2.9. Налаштування параметрів SQL Server

Для можливості роботи із зовнішніми скриптами (R), необхідно ввімкнути служби R для сервера. Це також можна зробити на вкладці Налаштування SQL Server (рис. 2.10).

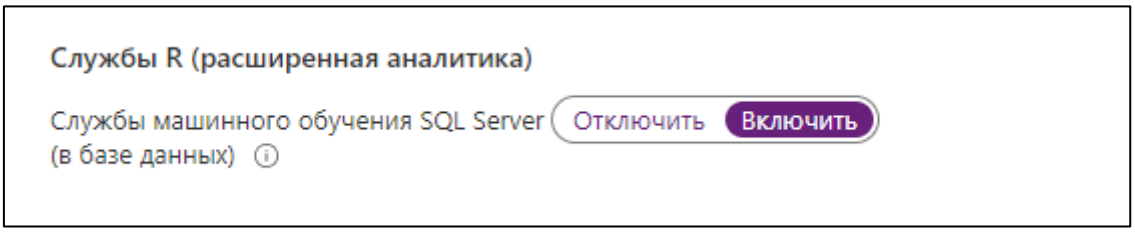

Рис. 2.10. Підключення служб R

Скрипти, написані мовою R або Phyton, виконуються в SQL за допомогою використання T-SQL і процедури sp_execute_external_script. Скрипт, який буде виконуватися, можна написати в тілі процедури, але також можна підключити вже існуючий файл. Синтаксис процедури sp_execute_external_script представлено на рис. 2.11.

```
sp_execute_external_script
    @language = N'language',
    @script = N'script'
    [ , @input_data_1 = N'input_data_1' ]
    [ , @input_data_1_name = N'input_data_1_name' ]
    [ , @input_data_1_order_by_columns = N'input_data_1_order_by_columns' ]
    [ , @input_data_1_partition_by_columns = N'input_data_1_partition_by_columns' ]
    [ , @output_data_1_name = N'output_data_1_name' ]
    [ , @parallel = 0 | 1 ]
    [ , @params = N'@parameter_name data_type [ OUT | OUTPUT ] [ ,...n ]' ]
    [ , @parameter1 = 'value1' [ OUT | OUTPUT ] [ ,...n ] ]
```

Рис. 2.11. Синтаксис процедури sp_execute_external_script

3. Пакети мови програмування Python.

Програмна реалізація вирішення задачі за допомогою методів машинного навчання буде виконана мовою програмування Python – для локального сервера та мовою R – для сервера Microsoft Azure (оскільки для нього не передбачено використання мови Python). Задля уникнення помилок під час компіляції скриптів необхідно встановити пакети, представлені на рис. 2.12 – 2.13.

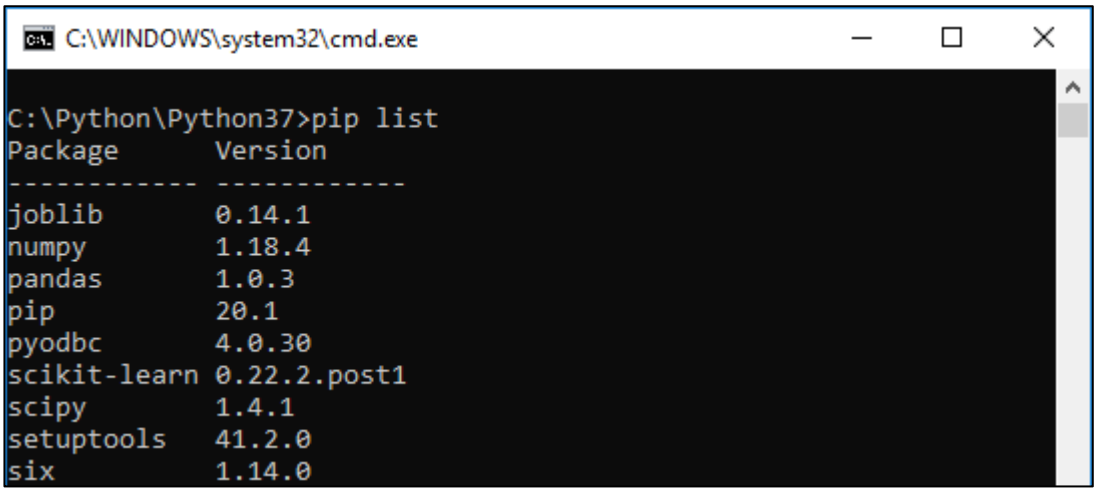

Рис. 2.12. Список встановлених програмних пакетів Python

Рис. 2.13. Список встановлених програмних пакетів R

4. SQL Server Machine Learning Service – використовуватиметься для можливості запуску скриптів, написаних мовою Python.

5. Spyder – программа для компіляції скриптів Python.

6. Excel 2016 – програма для роботи з електронними таблицями, створена корпорацією Microsoft. Необхідна для побудови графіків.

Дослідження можливостей інтеграції SQL Server та мов програмування виконуватиметься з використанням мов R та Python.

Для більш кращого розуміння основних переваг та недоліків цих мов програмування представлена табл. 2.2.

Таблиця 2.2

Порівняльна характеристика мов програмування R та Python

Критерій	R	Python
1	2	3
1. Мета	Аналіз даних та статистика	Розгортання та виробництво
2. Основні користувачі	Наукові співробітники (проведення науково-дослідних робіт)	Програмісти та розробники
3. Гнучкість	Проста у користуванні наявними бібліотеками	Легко конструювати нові моделі з нуля (обчислення матриць та оптимізація)
4. Крива навчання	Складна на початку	Лінійна та гладка
5. Популярність мови програмування (зміна відсотків)	4,23% у 2018 році	21,69% у 2018 році
6. Інтеграція	Запуск локально	Добре інтегрований із додатками, які забезпечують компіляцію вихідного коду
7. Завдання	Легко отримати первинні результати	Добре розгортати алгоритм
8. Розмір бази даних	Великий	Великий
9. IDE	Rstudio	Spyder, IPython Notebook
10. Важливі пакети та бібліотеки	tydiverse, ggplot2, caret, zoo	pandas, scipy, scikit-learn, TensorFlow, caret

Закінчення табл. 2.2

1	2	3
11. Недоліки	Повільна крива високого навчання, залежності між бібліотеками	Не так багато бібліотек, як у R
12. Переваги	• Графіки легкі у розумінні. R робить це красивим • Великий каталог для аналізу даних • Інтерфейс GitHub • RMarkdown	• Jupyter notebook: допомагають обмінюватися даними з колегами • Математичні обчислення • Розгортання • Читання коду • Швидкість • Функції в Python

Проаналізувавши інформацію з таблиці 2.2 можна зробити висновок, що обидві мови програмування є достойними конкурентами і мають потужний функціонал для вирішення задач машинного навчання. Вибір мови багато в чому залежить від особливостей завдань, які представлені для Machine Learning.

Python користується більшою популярністю, оскільки він простіший у розумінні та реалізації, є більш універсальним, завдяки чому може використовуватися у різних сферах діяльності.

R – є вузькоспрямованою мовою програмування і здебільшого використовується для статистичного аналізу.

Для початківців краще починати роботу з ML з використанням Python. Проте варто відзначити, що для досягнення найкращих результатів в області машинного навчання потрібно оволодіти обома мовами програмування. В цьому випадку буде можливість комбінувати код, написаний різними мовами. Наприклад, статистичний аналіз даних проводити засобами R, а обробку результатів та їх представлення – з використанням Python..

3. ЕКСПЕРИМЕНТАЛЬНЕ ДОСЛІДЖЕННЯ ТЕОРЕТИЧНИХ РЕЗУЛЬТАТІВ НА ОСНОВІ МЕТОДІВ СТАТИСТИЧНОГО, ІМІТАЦІЙНОГО МОДЕЛЮВАННЯ ЗА ДОПОМОГОЮ ПРОГРАМНИХ ПАКЕТІВ

3.1. Підготовка даних для проведення дослідження

Як вже було зазначено вище, основна задача, яка вирішувалася в даній роботі – це дослідження засобів вирішення задач в Machine Learning за допомогою SQL Server Machine Learning Services для визначення найбільш ефективного методу їх програмної реалізації.

Дослідження проводилось на комп’ютері, який має характеристики, представлені в табл. 3.1.

Таблиця 3.1

Характеристики робочої машини

Назва	Характеристика
1. Виробник	Lenovo
2. Операційна система	Windows 10 Pro
3. Процессор	Intel(R) Pentium(R) CPU 3550M @ 2.30GHz, 2295 Mhz, 2 Core(s), 2 Logical Processor(s)
4. Тип оперативної пам'яті	DDR3-1600 МГц (1 x 4 ГБ)
5. Встановлене ОЗУ	4.00 Гб
6. Загальний об’єм фізичної пам’яті	3.93 Гб
7. Доступний об’єм фізичної пам’яті	1.09 Гб
8. Загальний об’єм віртуальної пам’яті	9.68 Гб
9. Доступний об’єм віртуальної пам’яті	5.65 Гб
10. Тип системи	x64 розрядний процессор

Як вже зазначалося раніше, для проведення експериментів з ефективності обробки даних використовувалася база даних, яка містить інформацію про кількість прокатів лиж (див. рис. 2.2). Прокат лиж має сезонний характер, тож необхідно правильно згрупувати дані таблиці rental_data, щоб в подальшому мати змогу їх аналізувати. Сезон розпочинається з грудня місяця попереднього року та закінчується в квітні поточного. Таблиця містить інформацію з 2013 по 2015 роки. Алгоритм групування даних за сезонами представлено на рис. 3.1.

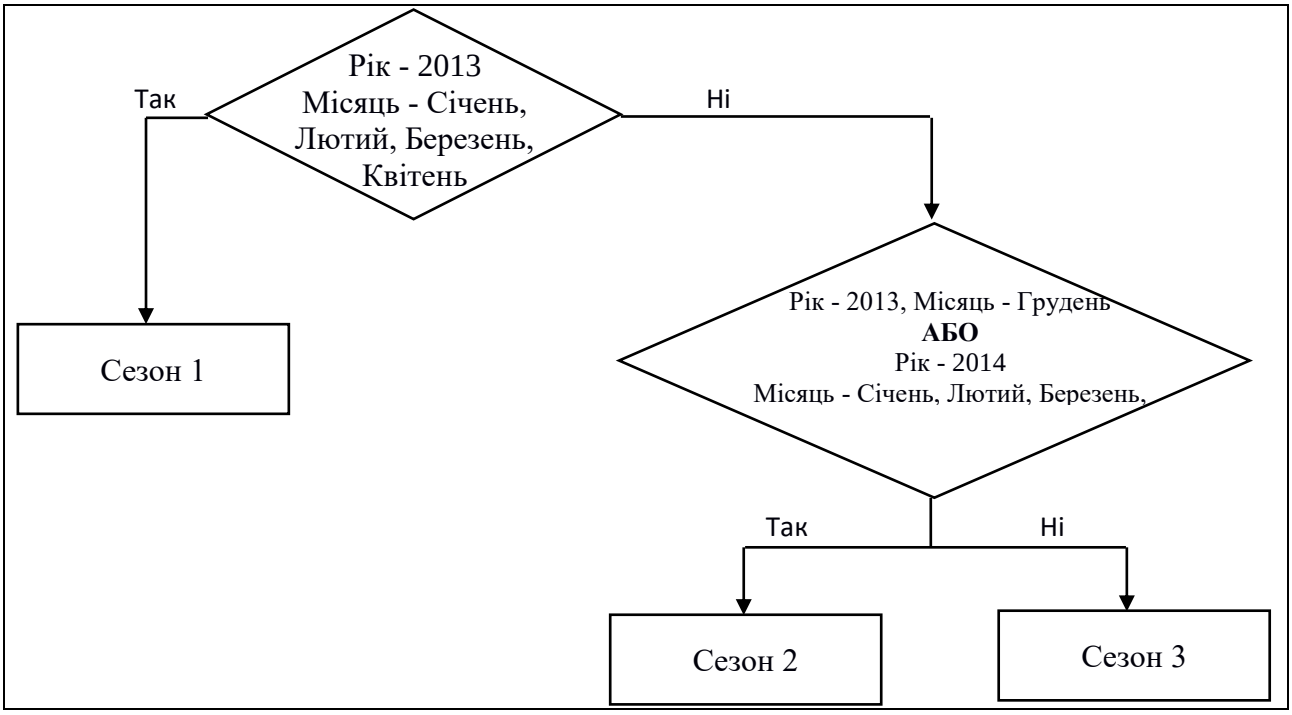

Рис. 3.1. Алгоритм групування даних за сезонами

Засобами Excel було створено нові колонки, в яких вказана інформація про сезон. Для заповнення колонок даними використовувалася наступна формула (3.1):

```
=IF(AND([@Year]=2013, [@Month]<>12), "Season 1",
IF(OR(AND([@Year]=2014, [@Month]<>12), AND([@Year]=2013,
[@Month]=12)), "Season 2", "Season 3"))
```
(3.1)

Також, для кращого відображення даних на графіку необхідно перерахувати порядок місяців, використовуючи формулу 3.2.

```
=IF([@Month]<>12,[@Month]+1,1)
```
(3.2)

В результаті було отримано таблицю з даними, представлену на рис. 3.2.

Year	Month	Day	Season	Season month	RentalCount	WeekDay	Holiday	Snow
2013	1	31	Season 1	2	49	5	0	0
2013	1	15	Season 1	2	62	3	0	1
2013	1	13	Season 1	2	507	1	0	1
2013	1	12	Season 1	2	507	7	0	1
2013	2	18	Season 1	3	690	2	1	1
2013	3	3	Season 1	4	240	1	0	0
2013	4	28	Season 1	5	310	1	0	0
2013	4	13	Season 1	5	310	7	0	0
2013	12	15	Season 2	1	273	1	0	1
2013	12	1	Season 2	1	273	1	0	1
2014	1	16	Season 2	2	49	5	0	0
2014	2	26	Season 2	3	62	4	0	1
2014	2	5	Season 2	3	62	4	0	1
2014	3	8	Season 2	4	240	7	0	0
2014	4	5	Season 2	5	481	7	0	1
2014	12	26	Season 3	1	180	6	0	0
2014	12	21	Season 3	1	312	1	0	1
2014	12	7	Season 3	1	312	1	0	1
2014	12	27	Season 3	1	507	7	0	1
2015	1	10	Season 3	2	507	7	0	1
2015	1	19	Season 3	2	824	2	1	1
2015	2	13	Season 3	3	49	6	0	0
2015	2	17	Season 3	3	49	3	0	0
2015	3	13	Season 3	4	49	6	0	1
2015	3	10	Season 3	4	49	3	0	1
2015	4	12	Season 3	5	312	1	0	1
2015	4	25	Season 3	5	312	7	0	1
2015	4	26	Season 3	5	507	1	0	1
2015	12	25	Season 3	1	220	6	1	0
2015	12	19	Season 3	1	220	7	0	0

Рис. 3.2. Дані з урахуванням сезонності

Для визначення того, чи існує будь-яка залежність в даних таблиці було побудовано зведену таблицу за сезонами та графіки в Excel (рис. 3.3 – 3.6). Зведена таблиця містить загальну інформацію про кількість прокатів лижного спорядження за кожний місяць сезону. Ключовим показником, який впливає на кількість прокатів є місяць сезону.

Sum of RentalCount	Month					
Row Labels	1	2	3	4	5	Grand Total
Season 1		6493	5852	4137	3365	19847
Season 2	4739	6973	5309	4510	3359	24890
Season 3	7731	6493	5513	3940	3358	27035
Grand Total	**12470**	**19959**	**16674**	**12587**	**10082**	**71772**

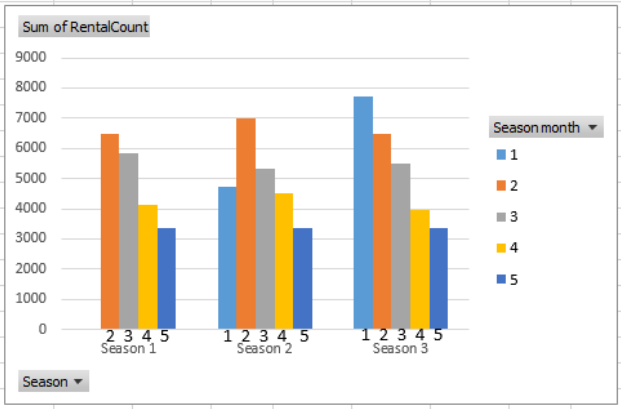

Рис. 3.3. Зведена таблиця rental_data за сезонами

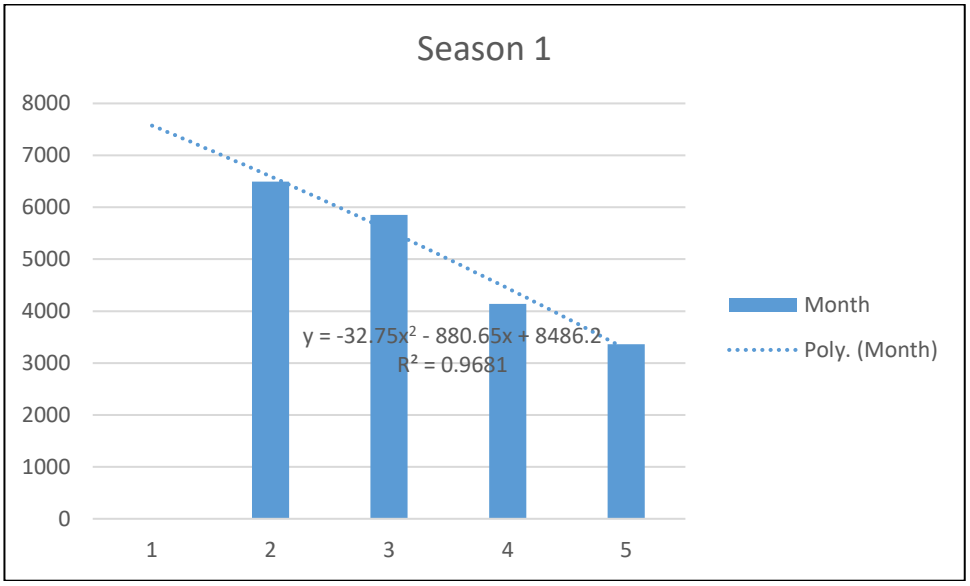

Рис. 3.4. Кількість прокатів лиж за 1 сезон

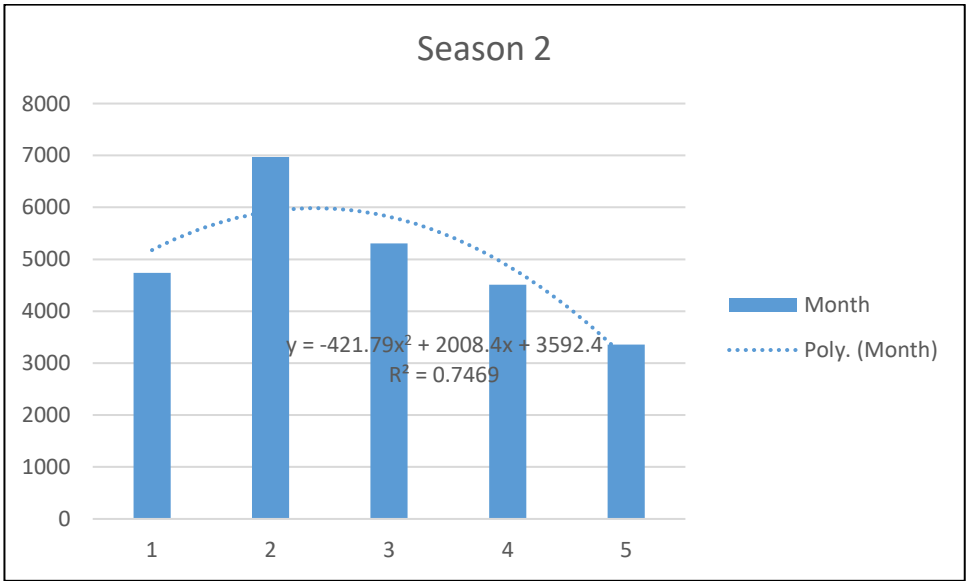

Рис. 3.5. Кількість прокатів лиж за 2 сезон

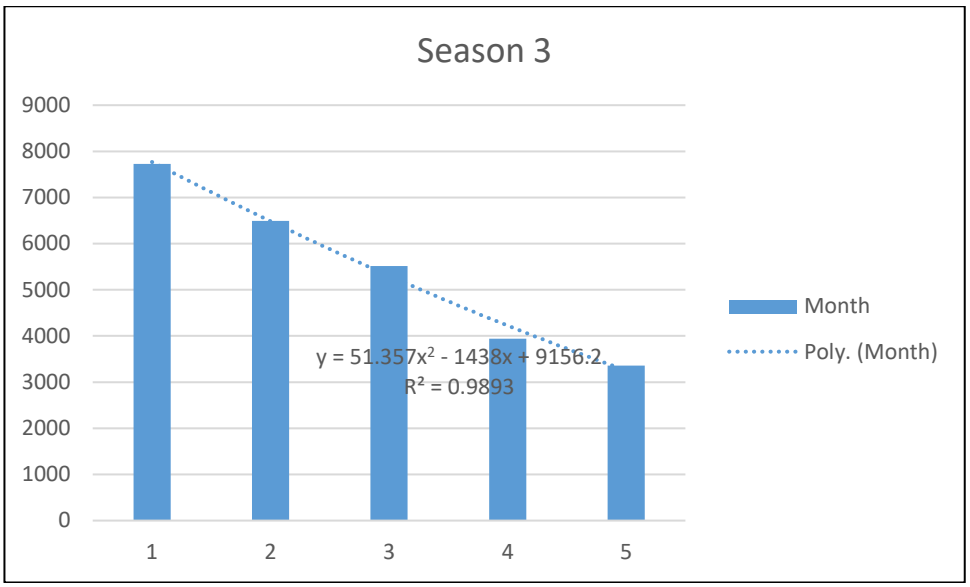

Рис. 3.6. Кількість прокатів лиж за 3 сезон

Аналізуючи графіки, можна зробити висновок, що між даними різних сезонів загальної залежності немає, проте спільним є те, що з другого місяця кожного сезону кількість прокатів лижного спорядження починає зменшуватися. Така ситуація є цілком реальною, оскільки з огляду на предметну область очевидно, що найбільший попит на аренду лижного спорядження буде саме з грудня по березень.

На графіках представлено лінії трендів з визначеними рівняннями регресії, які дозволяють оцінити динаміку зміни даних. Коефіцієнти рівняння регресії показують силу і характер впливу незалежних змінних на залежну і характеризують ступінь значущості окремих змінних для підвищення точності моделі. Параметр x відвовідає значенню місяця лижного сезону.

Рівняння ліній тренду у загальній формі має вигляд (формула 3.3):

$$y(x) = ax^2 + bx + c \tag{3.3}$$

Зв'язок між y і x визначає знак коефіцієнта регресії a (якщо a > 0 - прямий зв'язок, інакше - зворотній). Табличне значення для перевірки значущості коефіцієнта кореляції дорівнює $t_{крит}(120;0.05) = 1.9719$

За допомогою методів математичної статистики було визначено значущість отриманих коефіцієнтів. Результати представлено в табл.3.2.

Таблиця 3.2

Оцінка значущості коефіцієнтів рівнянь

Характеристика	Сезон 1	Сезон 2	Сезон 3
Дисперсія	43 728.99	40 808.48	31 732.45
Середньоквадратичне відхилення	209.11	202.011	178.14
Коефіцієнт кореляції, r	-0.196	-0.113	-0.586
$t_{теор}$	2.177	1.384	9.694
Коефіцієнт детермінації, R^2	0.968	0.746	0.9893
Результат	• Дисперсія – велика, що свідчить про великий розмах данних. • Коефіцієнт кореляції – низький, зв'язок між ознаками слабкий. • Кореляція від'мна. • Коефіцієнт кореляції статистично-значущий. • Коефіцієнт детермінації – високий, точність підбору рівняння тренда – висока.	• Дисперсія – велика, що свідчить про великий розмах данних. • Коефіцієнт кореляції – низький, зв'язок між ознаками слабкий. • Кореляція від'мна. • Коефіцієнт кореляції статистично-незначущий. • Коефіцієнт детермінації – високий, точність підбору рівняння тренда - висока	• Дисперсія – велика, що свідчить про великий розмах данних. • Коефіцієнт кореляції – низький, зв'язок між ознаками слабкий. • Кореляція від'мна. • Коефіцієнт кореляції статистично-значущий. • Коефіцієнт детермінації – високий, точність підбору рівняння тренда - висока

Математична постановка задачі генерації даних.

Як вже зазначалося вище, для проведення дослідження необхідно створити таблицю з більшим об'ємом даних, ніж ми маємо. Також було б гарно мати можливість регулювати кількість сгенерованих даних за деяким показником.

Формула для генерації кількості аренд лижного спорядження має наступний вигляд:

$$RentalCount = 1000 * MonthCoeficient * EXP(-0.07 * \text{Поточний день}) \quad (3.4)$$

1000 – базова кількість аренд, яку фірма може забезпечити. В формулі використовується функція EXP. Від'ємний показник степені (-0,07) характеризує спад функції.

Для забезпечення унікальності даних було розраховано додатковий коефіцієнт (MonthCoeficient) за наступною формулою:

- якщо поточний місяць - грудень

$$MonthCoeficient = 1 - \text{Номер\_місяця} \ / 100 \ * 0.5 \quad (3.5)$$

- якщо поточний місяць – січень, лютий, березень або квітень

$$MonthCoeficient = 1 - \text{Номер\_місяця} \ / 100 \ * \text{Номер\_місяця} \quad (3.6)$$

Така генерація коефіцієнтів регулює те, як номер місяця впливає на кількість аренд. Так, в січні коефіцієнт буде найбільшим, оскільки саме в цьому місяці спостерігається найбільший попит на лижне спорядження.

Окрім місяця на результат також можуть впливати додаткові фактори, які сприятимуть зростанню кількості аренд:

- чи йшов сніг в цей день (якщо так – кількість збільшується на 20);
- тип дня (святковий, вихідний, будній). Для святкового дня – 20 додаткових аренд, для вихідного – 30.

Дані про те, чи йшов сніг генерувалися випадковим чином за наступною умовою:

Якщо поточний місяць – січень, лютий або грудень – @Show = ROUND(RAND() * (1 - 0), 0).

В іншому випадку @Show = 0.

Для генерації великої кількості даних була використана збережена процедура з параметром, скрипт якої наведений на рис. 3.7. Параметр визначає кількість ітерацій, протягом яких буде відбуватися генерація вихідних даних. Скрипт включає всі умови, викладені в математичній постановці.

```
CREATE PROCEDURE generateData
@Number INT
AS
-- Clear data of rental_data_new_exp table
DELETE from rental_data_new_exp;
-- Declare variables
DECLARE @StartDate DATETIME
DECLARE @EndDate DATETIME
DECLARE @RentalCount INT
DECLARE @WeekDay INT
DECLARE @Holiday INT
DECLARE @Show INT
DECLARE @NumberOfIterations INT
DECLARE @MonthCoeficient DECIMAL(5,2)
DECLARE @i INT
SET @NumberOfIterations = @Number
SET @i = 0
-- Set date boundaries for data to be analyzed in the future
SET @StartDate = '01/01/2000'
SET @EndDate = '12/31/2015'
-- Set CurrentDate for using WHILE cycle
DECLARE @CurrentDate DATETIME
SET @CurrentDate = @StartDate
WHILE @CurrentDate <= @EndDate
BEGIN
       WHILE @i < @NumberOfIterations
       BEGIN
        -- Generate random data
              -- Generate @MonthCoeficient
              IF (Month(@CurrentDate) = 12) SET @MonthCoeficient = (1 -
Month(@CurrentDate) *1. /100 * 0.5)
              ELSE SET @MonthCoeficient = (1 - Month(@CurrentDate) * 1. / 100 *
Month(@CurrentDate));
              -- Generate @RentalCount
              SET @RentalCount = 1000 * @MonthCoeficient * EXP(-0.07 *
Day(@CurrentDate));
              -- If Winter month, set Show parameter
              IF(Month(@CurrentDate) in (1, 2, 12)) SET @Show = ROUND(RAND() * (1 -
0), 0);
              ELSE SET @Show = 0;
              SET @Holiday = ROUND(RAND() * (1 - 0), 0);
              SET @WeekDay = DATEPART(dw, @CurrentDate) + 1;
              -- If the day is specific - add extra rental count
              IF(@Show = 1) SET @RentalCount = @RentalCount + 20;
              IF(@Holiday = 1) SET @RentalCount = @RentalCount + 20;
              IF(@WeekDay in(6,7)) SET @RentalCount = @RentalCount + 30;
              -- Since we are generating RentalCount for the same date several times,
a Random value must be used
              SET @RentalCount = @RentalCount + RAND() * (100 - 0);
        -- Add new data to the rental_data_new_exp
              INSERT INTO rental_data_new_exp VALUES (
                 Year(@CurrentDate),
                 Month(@CurrentDate),
                 Day(@CurrentDate),
                 @RentalCount,
                 @WeekDay,
                 @Holiday,
                 @Show);
       SET @i = @i + 1
```

Рис. 3.7. Скрипт для генерації даних

```
        END;
        -- Increase day of the currentDate by 1 for the next step of cycle
              SET @CurrentDate = DateAdd(d, 1, @CurrentDate)
              -- Ski renting is actual only for a specific season (From December to
April)
              -- So skip other months
              IF (Month(@CurrentDate) = 5)
              SET @CurrentDate = DateAdd(m, 7, @CurrentDate)
        SET @i = 0
END;
```

Рис. 3.7. (Закінчення)

Діапазон вхідних даних – з 01/01/2000 по 12/31/2015. Наведений алгоритм є зручним, оскільки можна з легкістю корегувати кількість згенерованих даних шляхом редагування змінної @NumberOfIterations. Для кожної дати в циклі здійснюється підрахунок кількості аренд лижного спорядження.

Оскільки генерація результату могла проводитися декілька разів для тієї ж самої дати, то використовувалася функція RAND() для запобігання дублікатів.

В результаті було сформовано таблицю rental_data_new_exp, яка містить 484 000 рядків. Генерація даних проводилась досить швидко (до 10 хвилин для найбільшого об’єму даних).

На основі згенерованих даних також було побудовано зведену таблицю (рис. 3.8) та діаграму (рис. 3.9).

Sum of RentalCount	Month					
Season	1	2	3	4	5	Grand Total
Season 1		2899781	2748250	2649070	2437246	10734347
Season 10	2769916	2909094	2711718	2636413	2440811	13467952
Season 11	2777233	2903177	2708626	2638227	2450565	13477828
Season 12	2778694	2896874	2711285	2642756	2442793	13472402
Season 13	2778495	2894447	2747917	2650085	2434950	13505894
Season 14	2774158	2899520	2711993	2644536	2435879	13466086
Season 15	2774842	2906163	2707510	2636099	2435022	13459636
Season 16	2771583	2903880	2711650	2636253	2445763	13469129
Season 2	2777347	2893451	2713473	2648489	2438152	13470912
Season 3	2772799	2903478	2708584	2644500	2437124	13466485
Season 4	2776681	2910036	2707507	2639725	2431774	13465723
Season 5	2772172	2908688	2747290	2642346	2446466	13516962
Season 7	2783560	2896061	2711691	2651256	2436094	13478662
Season 8	2776175	2894430	2707724	2647932	2437329	13463590
Season 9	2774286	2900506	2755066	2637027	2436116	13503001
Season 6	2786921	2897158	2712688	2640716	2441910	13479393
Season 17	2777999					2777999
Grand Total	**44422861**	**46416744**	**43522972**	**42285430**	**39027994**	**215676001**

Рис. 3.8. Зведена таблиця згенерованих даних

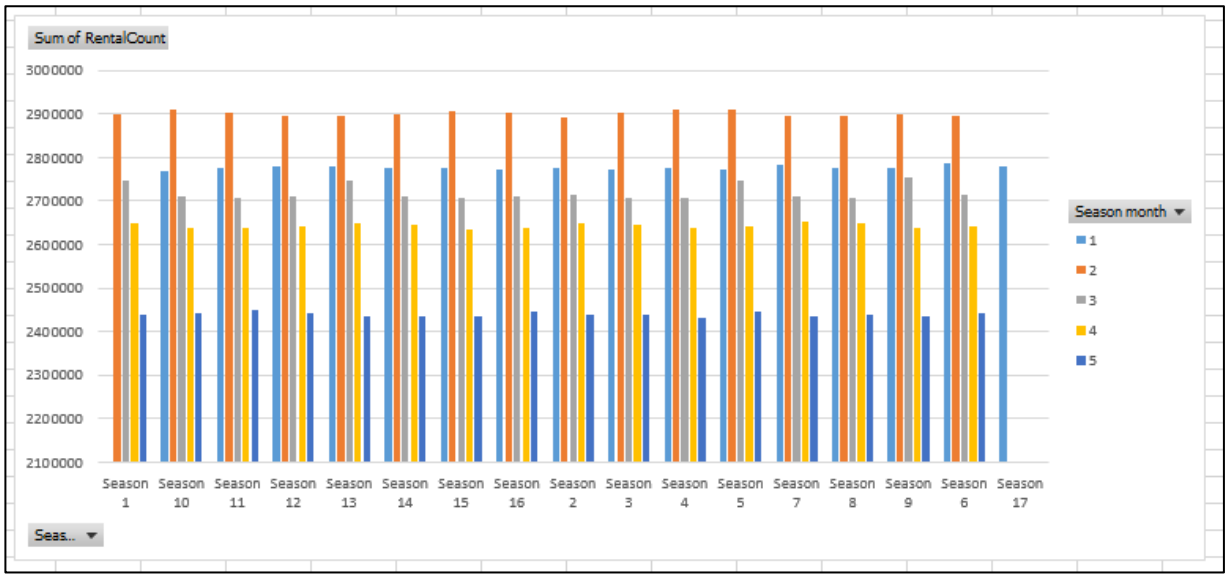

Рис. 3.9. Графіки, побудовані на основі зведеної таблиці згенерованих даних

Як бачимо, графіки мають вигляд, аналогічний графікам, побудованих на початкових даних (див. рис. 3.3). Отже можна зробити висновок, що дані було згенеровано правильно.

3.2. Опис результатів моделювання на основі контрольного прикладу

Найбільш популярним алгоритмом машинного навчання є лінійна регресія завдяки його простоті та швидкості здійснення пронозування. Тому обробка даних при проведенні досліджень будет відбуватися з використанням цього методу.

Проведемо дослідження аби визначити як кількість даних впливає на час їх обробки різними способами: з використанням вбудованих мовних засобів SQL Server та класичного підходу.

Для аналізу часи обробки даних на локальному сервері використаємо функцію бібліотеки Python timeit default_timer(), під час роботи з сервером Microsoft Azure будемо використутовувати функцію R Sys.time(). Основними процесами, які взаємодіють з даними є їх завантаження, обробка (навчання моделі та прогнозування результатів) та пересилка.

Для доступу до даних необхідно вказати рядок підключення, в якому зазначити ім’я сервера, базу даних та таблицю.

Оскільки експерименти будуть проводитися з використанням серверів, які мають різне розташування, то в табл. 3.3 представлено рядки підключення до них.

Таблиця 3.3

Підключення до сервера БД

Сервера	Рядок підключення
Локальний	DRIVER={ODBC Driver 13 for SQL Server}; SERVER=DESKTOP-A6FSIKB\SQLEXPRESS; DATABASE=TutorialDB; Trusted_Connection=yes;
В хмарній платформі Microsoft Azure	Driver={ODBC Driver 13 for SQL Server};Server=tcp:23.96.37.83,1401;Database=TutorialDB;Uid=anna_koskina;Pwd=******;Encrypt=yes;TrustServerCertificate=no;Connection Timeout=30;

Властивості локально розташованого сервера та сервера в Microsoft Azure представлені в табл. 3.4.

Таблиця 3.4

Властивості серверів

Властивість	Локальний сервер	Сервер віртуальної машини в Microsoft Azure
Версія	14.0.2027.2	14.0.3356.20
Кількість процесорів	2	2
Обсяг фізичної пам'яті	4020 MB	8192 MB
Обсяг віртуальної пам'яті	131 071 GB	128 000 GB

Проведення процесу машинного навчання потребує визначення вибірки, на основі якої буде проводитись навчання моделі та іншої – для проведення прогнозування результатів. Отже, оскільки в якості вхідних даних використовувалася інформація за 2000 – 2015 роки, то навчання моделі проводилося на основі даних за 2000 – 2014 роки, прогнозування – для 2015 року.

Програмна реалізація роботи з даними за допомогою класичного способу представлена на рис. 3.10.

```
import timeit
import pyodbc
import pandas
from sklearn.linear_model import LinearRegression
# Connection string to your SQL Server instance
conn_str = pyodbc.connect("Driver={ODBC Driver 13 for SQL
Server};Server=23.96.37.83,1401;Database=TutorialDB;Uid=anna_koskina;Pwd=********;
Encrypt=yes;TrustServerCertificate=no;Connection Timeout=30;")
query_str = "SELECT  Year, Month, Day, Rentalcount, Weekday, Holiday, Snow FROM
dbo.rental_data_new_exp"
```

Рис. 3.10. Програмна реалізація застосування методу машинного навчання для обробки даних

```
# Load data and calculate the time required for this action
# startDataLoading - capture the beginning of data loading
# endDataLoading - capture the end of data loading
startDataLoading = timeit.default_timer()
df = pandas.read_sql(sql=query_str, con=conn_str)
endDataLoading = timeit.default_timer()

train = df[df.Year < 2015]
test = df[df.Year == 2015]
# Get all the columns from the dataframe.
columns = df.columns.tolist()

# Store the variable we`ll be predicting on.
target = "Rentalcount"

#Process data using the LinearRegression method of ML and calculate the time requied
for this action
# startDataProcessing - capture the beginning of data processing
# endDataProcessing - capture the end of data processing
startDataProcessing = timeit.default_timer()

# Initialize the model class.
lin_model = LinearRegression()

# Fit the model to the training data.
lin_model.fit(train[columns], train[target])

# Generate our predictions for the test set.
lin_predictions = lin_model.predict(test[columns])
predictions_df = pandas.DataFrame(lin_predictions)
test = test.reset_index()
OutputDataSet = pandas.concat([predictions_df, test["Rentalcount"], test["Month"],
test["Day"], test["Weekday"], test["Snow"], test["Holiday"], test["Year"]], axis=1)
#print(OutputDataSet)

endDataProcessing = timeit.default_timer()

print('Time data loading: ', endDataLoading - startDataLoading)
print('Time data processing: ', endDataProcessing - startDataProcessing)
conn_str.close()
```

Рис. 3.10. (Закінчення)

Обробка даних на сервері відбувалася з використанням вбудованих мовних засобів SQL Server. Для цього виконувалася збережена процедура, текст якої представлено на рис. 3.11 – 3.12.

```
CREATE PROCEDURE [dbo].[LinearRegressionPrediction]
AS
BEGIN
    EXECUTE sp_execute_external_script
      @language = N'Python'
    , @script = N'
from sklearn.linear_model import LinearRegression
import timeit
```

Рис. 3.11. Текст збереженої процедури, яка виконує обробку даних на сервері використовуючи метод машинного навчання (мова програмування Python)

```
#Load data and calculate the time requied for this action
# startDataLoading - capture the beginning of data loading
# endDataLoading - capture the end of data loading

startDataLoading = timeit.default_timer()
df = rental_train_data
endDataLoading = timeit.default_timer()

train = df[df.Year < 2015]
test = df[df.Year == 2015]

# Get all the columns from the dataframe.
columns = df.columns.tolist()

# Store the variable will be predicting on.
target = "RentalCount"

#Process data using the LinearRegression method of ML and calculate the time requied
for this action
# startDataProcessing - capture the beginning of data processing
# endDataProcessing - capture the end of data processing
startDataProcessing = timeit.default_timer()

# Initialize the model class.
lin_model = LinearRegression()

# Fit the model to the training data.
lin_model.fit(train[columns], train[target])

# Generate our predictions for the test set.
lin_predictions = lin_model.predict(test[columns])

predictions_df = pandas.DataFrame(lin_predictions)
test = test.reset_index()

endDataProcessing = timeit.default_timer()

dataLoading = endDataLoading - startDataLoading
dataProcessing = endDataProcessing - startDataProcessing

result = pandas.DataFrame([dataLoading, dataProcessing])
OutputDataSet = pandas.concat([ predictions_df, test["RentalCount"], test["Month"],
test["Day"], test["WeekDay"], test["Snow"], test["Holiday"], test["Year"]], axis=1)
OutputDataSet = OutputDataSet.append(result)
'
, @input_data_1 = N'select "RentalCount", "Year", "Month", "Day", "WeekDay", "Snow",
"Holiday" from dbo.rental_data_new_exp'
, @input_data_1_name = N'rental_train_data';
END;
GO
```

Рис. 3.11. (Закінчення)

```
CREATE PROCEDURE [dbo].[LinearRegressionPredictionR]
AS
BEGIN
EXECUTE sp_execute_external_script
      @language = N'R'
    , @script = N'
```

Рис. 3.12. Текст збереженої процедури, яка виконує обробку даних на сервері використовуючи метод машинного навчання (мова програмування R)

```
#Load data and calculate the time requied for this action
# startDataLoading - capture the beginning of data loading
# endDataLoading - capture the end of data loading

startDataLoading  <- Sys.time()
df = rentaldata
endDataLoading <- Sys.time()
df$Holiday <- factor(rentaldata$Holiday);
df$Snow    <- factor(rentaldata$Snow);
df$WeekDay <- factor(rentaldata$WeekDay);

train_data = df[df$Year < 2015,];
test_data  = df[df$Year == 2015,];

#Use the RentalCount column to check the quality of the prediction against actual
values
actual_counts <- test_data$RentalCount;

#Process data using the LinearRegression method of ML and calculate the time requied
for this action
# startDataProcessing - capture the beginning of data processing
# endDataProcessing - capture the end of data processing

startDataProcessing <- Sys.time()

#Model 1: Use lm to create a linear regression model, trained with the training data
set
model_lm <- lm(RentalCount ~  Month + Day + WeekDay + Snow + Holiday, data =
train_data);

#Use model to make predictions using the test data set.
predict_lm <- predict(model_lm, test_data)
predict_lm <- data.frame(RentalCount_Pred = predict_lm, RentalCount =
test_data$RentalCount,
                         Year = test_data$Year, Month = test_data$Month,
                         Day = test_data$Day, Weekday = test_data$WeekDay,
                         Snow = test_data$Snow, Holiday = test_data$Holiday)
endDataProcessing <- Sys.time()

dataLoading <- endDataLoading - startDataLoading
dataProcessing <- endDataProcessing - startDataProcessing

result <- rbind(predict_lm, c(dataLoading))
result <- rbind(result, c(dataProcessing))
            '
      , @input_data_1 = N'select "RentalCount", "Year", "Month", "Day", "WeekDay",
"Snow", "Holiday" from dbo.rental_data_new_exp'
, @input_data_1_name = N'rentaldata'
, @output_data_1_name = N'result'
with result sets ((RentalCount_Pred float, "RentalCount" float, "Month" float,"Day"
float,"WeekDay" float,"Snow" float,"Holiday" float, "Year" float));
END;
```

Рис. 3.12. (Закінчення)

Скрипти для проведення дослідження ефективності роботи вбудованих мовних засобів SQL Server представлено на рис. 3.13 – 3.14.

```
import pyodbc
import timeit
# Connection string to your SQL Server instance
conn_str = pyodbc.connect("DRIVER={ODBC Driver 13 for SQL Server}; SERVER=DESKTOP-
A6FSIKB\SQLEXPRESS; DATABASE=TutorialDB; Trusted_Connection=yes")

cursor = conn_str.cursor()
cursor.execute("CHECKPOINT;")
cursor.execute("DBCC FREEPROCCACHE;")
cursor.execute("DBCC DROPCLEANBUFFERS;")
cursor.execute("EXEC LinearRegressionPrediction;")
# Calculate the time required for transfering data from SQL Server
# startDataTransfering - capture the beginning of data transfering
# endDataTransfering - capture the end of data transfering
startDataTransfering = timeit.default_timer()
rc = cursor.fetchall()
endDataTransfering = timeit.default_timer()
print(rc)

print('Time data transfering: ', endDataTransfering - startDataTransfering)
cursor.close()
conn_str.close()
```

Рис. 3.13. Дослідження ефективності роботи вбудованих мовних засобів SQL Server (локальний сервер)

```
library(odbc)
library(DBI)

cn <- dbConnect(odbc::odbc(), driver = "ODBC Driver 13 for SQL Server",
                server="tcp:23.96.37.83,1401",
                database = "TutorialDB",
                Uid="anna_koskina",
                Pwd="ItCraftDeveloper_2020")
query <- dbSendQuery(cn, "EXEC LinearRegressionPredictionR")

# Calculate the time required for transfering data from SQL Server
# startDataTransfering - capture the beginning of data transfering
# endDataTransfering - capture the end of data transfering

startDataTransfering  <- Sys.time()

res <- dbFetch(query)
tail(res, 2)

endDataTransfering <- Sys.time()

dataTransfering <- endDataTransfering - startDataTransfering

dataTransfering
```

Рис. 3.14. Дослідження ефективності роботи вбудованих мовних засобів SQL Server (сервер Microsoft Azure)

3.3. Оцінка адекватності експериментального дослідження методу

Статистична обробка результатів дослідження відбувалася за домогою використання методу порівняння даних. Експерименти проводилися з різним

об’ємом вхідних даних. Оскільки для генерації даних використовувалася збережена процедура, то в табл. 3.5 представлено приклад її запуску з урахуванням параметру та число створених записів.

Таблиця 3.5

Запуск збереженої процедури для генерації даних

SQL запит	Кількість сгенерованих даних
EXEC generateData 100	242 000
EXEC generateData 200	484 000
EXEC generateData 300	726 000
EXEC generateData 400	968 000
EXEC generateData 500	1 210 000

Деякі часові показники, які були отримані в результаті проведення експериментів, представлено на рис. 3.15 – 3.16.

```
Time data loading:  7.8474061499850905
Time data processing:  0.4942911737878717
```

Рис. 3.15. Часові показники, отримані в результаті класичного підходу обробки даних

```
[(4.46246088653033e-07, ), (0.8359443191980422, )]
```

Рис. 3.16. Часові показники, отримані в результаті обробки даних на сервері

Іноді при підрахунку часу завантаження даних з використанням SQL Server ML Services було отримано результат у вигляді 4.46246088653033e-07. Оскільки в кінці є частина e-07, то це свідчить про те, що дане число дуже близьке до 0. Отже можна вважати, що на завантаження даних час не витрачався.

Для кожного об’єму вхідних даних і розташування сервера проводилося по 10 експериметнів. Результати всіх проведених досліджень представлено в табл. 3.6 – 3.17.

Використання локального серверу

Таблиця 3.6

Результати дослідження для об’єму вхідних даних 242 000 записів

№ експерименту	Час отримання результатів, с					
	Використання вбудованих мовних засобів SQL Server			Використання класичного підходу		
	Завантаження даних	Обробка даних	Пересилка даних	Завантаження даних	Обробка даних	Пересилка даних
1	0	0.87	1.35	3.8	1.28	0
2	0	0.3	0.68	1.9	0.15	0
3	0	2.09	1.3	2.6	0.31	0
4	0	1.1	2.34	1.98	0.14	0
5	0	0.28	1.4	2.07	0.15	0
6	0	0.46	0.93	2.08	0.19	0
7	0	1.05	1.34	2.04	0.15	0
8	0	0.76	0.85	1.84	0.15	0
9	0	0.71	0.89	2.38	0.18	0
10	0	0.61	1.12	2.31	0.16	0
Середнє значення	0	0.823	1.22	2.3	0.286	0

Таблиця 3.7

Результати дослідження для об’єму вхідних даних 484 000 записів

№ експерименту	Час отримання результатів, с					
	Використання вбудованих мовних засобів SQL Server			Використання класичного підходу		
	Завантаження даних	Обробка даних	Пересилка даних	Завантаження даних	Обробка даних	Пересилка даних
1	2	3	4	5	6	7
1	0	2.7	1.21	5.97	1.44	0
2	0	0.52	1.43	3.77	0.25	0
3	0	0.46	1.65	3.77	0.26	0
4	0	0.89	1.65	3.69	0.25	0
5	0	1.12	1.95	3.81	0.25	0
6	0	0.48	1.59	3.85	0.26	0
7	0	0.53	1.52	4.22	0.25	0
8	0	0.35	1.81	3.64	0.25	0

Закінчення табл. 3.7

1	2	3	4	5	6	7
9	0	0.69	1.65	4.01	0.24	0
10	0	0.75	0.61	4.41	0.26	0
Середнє значення	0	0.849	1.507	4.114	0.371	0

Таблиця 3.8

Результати дослідження для об’єму вхідних даних 726 000 записів

№ експерименту	Час отримання результатів, с					
	Використання вбудованих мовних засобів SQL Server			Використання класичного підходу		
	Завантаження даних	Обробка даних	Пересилка даних	Завантаження даних	Обробка даних	Пересилка даних
1	0	2.87	0.77	8.94	1.67	0
2	0	0.62	3.65	6.44	0.5	0
3	0	0.86	0.9	5.73	0.39	0
4	0	0.44	1.02	5.63	0.4	0
5	0	0.64	0.69	6.9	0.45	0
6	0	1.03	0.67	6.66	0.39	0
7	0	0.43	0.65	6.18	0.4	0
8	0	0.58	0.94	6.43	0.41	0
9	0	0.49	0.67	5.93	0.39	0
10	0	0.91	0.77	6.24	0.39	0
Середнє значення	0	0.887	1.073	6.508	0.539	0

Таблиця 3.9

Результати дослідження для об’єму вхідних даних 968 000 записів

№ експерименту	Час отримання результатів, с					
	Використання вбудованих мовних засобів SQL Server			Використання класичного підходу		
	Завантаження даних	Обробка даних	Пересилка даних	Завантаження даних	Обробка даних	Пересилка даних
1	2	3	4	5	6	7
1	0	1.32	4.03	11.39	2.66	0
2	0	0.51	5.37	8.56	0.9	0

Закінчення табл. 3.9

1	2	3	4	5	6	7
3	0	1.13	0.74	8.62	0.48	0
4	0	1.1	0.83	7.85	0.5	0
5	0	0.84	0.91	9.52	0.49	0
6	0	0.77	0.74	7.94	0.51	0
7	0	0.56	0.81	10.63	0.75	0
8	0	0.61	0.74	7.55	0.5	0
9	0	1.03	0.78	8.42	0.7	0
10	0	2.2	0.85	8.09	0.72	0
Середнє значення	0	1.007	1.58	8.857	0.821	0

Таблиця 3.10

Результати дослідження для об’єму вхідних даних 1 210 000 записів

№ експерименту	Час отримання результатів, с					
	Використання вбудованих мовних засобів SQL Server			Використання класичного підходу		
	Завантаження даних	Обробка даних	Пересилка даних	Завантаження даних	Обробка даних	Пересилка даних
1	0	3.97	7.5	17.32	4.3	0
2	0	0.77	1.66	12.87	0.62	0
3	0	0.77	0.81	9.82	0.58	0
4	0	0.57	0.82	10	0.71	0
5	0	0.84	1.02	10.25	0.68	0
6	0	1.33	1.38	9.1	0.61	0
7	0	1.28	0.98	9.01	0.63	0
8	0	0.77	0.91	9.84	0.61	0
9	0	1.13	0.79	9.42	0.6	0
10	0	0.9	0.75	10.86	0.9	0
Середнє значення	0	1.233	1.662	10.849	1.024	0

Таблиця 3.11

Результати дослідження для об’єму вхідних даних 1 452 000 записів

№ експерименту	Час отримання результатів, с					
	Використання вбудованих мовних засобів SQL Server			Використання класичного підходу		
	Завантаження даних	Обробка даних	Пересилка даних	Завантаження даних	Обробка даних	Пересилка даних
1	0	6.33	1.55	32.67	15.28	0
2	0	0.73	6.65	64.58	0.97	0
3	0	6.46	3.01	12.64	0.78	0
4	0	1.35	1.01	12.5	0.78	0
5	0	1.09	3.31	12.1	0.77	0
6	0	0.88	2.32	12.6	1	0
7	0	0.8	1.22	12.61	0.79	0
8	0	1.08	1.34	12.2	0.96	0
9	0	0.94	0.91	12.1	0.78	0
10	0	1.25	0.8	12.54	1.22	0
Середнє значення	0	2.091	2.212	32.67	15.28	0

Сервер Microsoft Azure

Таблиця 3.12

Результати дослідження для об’єму вхідних даних 242 000 записів

№ експерименту	Час отримання результатів, с					
	Використання вбудованих мовних засобів SQL Server			Використання класичного підходу		
	Завантаження даних	Обробка даних	Пересилка даних	Завантаження даних	Обробка даних	Пересилка даних
1	2	3	4	5	6	7
1	0	0.38	1.02	7.34	1.79	0
2	0	0.33	1.04	12.07	0.1	0
3	0	0.39	1.04	8.28	0.1	0
4	0	0.38	0.96	7.91	0.11	0
5	0	0.38	0.9	5.51	0.27	0
6	0	0.39	1	8.26	0.12	0

Закінчення табл. 3.12

1	2	3	4	5	6	7
7	0	0.38	1.1	5.71	0.12	0
8	0	0.39	1	5.6	0.25	0
9	0	0.38	0.95	7.03	0.12	0
10	0	0.38	1.1	4.13	0.12	0
Середнє значення	0	0.378	1.011	7.184	0.31	0

Таблиця 3.13

Результати дослідження для об’єму вхідних даних 484 000 записів

№ експерименту	Час отримання результатів, с					
	Використання вбудованих мовних засобів SQL Server			Використання класичного підходу		
	Завантаження даних	Обробка даних	Пересилка даних	Завантаження даних	Обробка даних	Пересилка даних
1	0	0.59	1.18	11.69	1.8	0
2	0	0.73	1.22	15.03	0.21	0
3	0	0.56	1.42	11.74	0.22	0
4	0	0.59	1.52	10.95	0.21	0
5	0	0.52	1.23	11.28	0.22	0
6	0	0.52	3.36	13.99	0.22	0
7	0	0.56	3.81	12.1	0.21	0
8	0	0.56	1.57	15.72	0.21	0
9	0	0.5	3.8	11.31	0.22	0
10	0	0.58	1.23	8.51	0.22	0
Середнє значення	0	0.571	2.034	12.232	0.374	0

Таблиця 3.14

Результати дослідження для об’єму вхідних даних 726 000 записів

№ експерименту	Час отримання результатів, с					
	Використання вбудованих мовних засобів SQL Server			Використання класичного підходу		
	Завантаження даних	Обробка даних	Пересилка даних	Завантаження даних	Обробка даних	Пересилка даних
1	0	0.78	1.78	19.22	2.06	0
2	0	0.72	2.87	16.82	0.3	0
3	0	0.69	1.45	15.17	0.31	0
4	0	0.67	1.98	18.5	0.31	0
5	0	0.77	3.04	15.16	0.3	0
6	0	0.67	3.67	14.96	0.34	0
7	0	0.7	2.35	15.15	0.3	0
8	0	0.69	4.34	12.18	0.3	0
9	0	0.72	3.94	14.35	0.31	0
10	0	0.72	4.57	15.74	0.3	0
Середнє значення	0	0.713	2.999	15.725	0.483	0

Таблиця 3.15

Результати дослідження для об’єму вхідних даних 968 000 записів

№ експерименту	Час отримання результатів, с					
	Використання вбудованих мовних засобів SQL Server			Використання класичного підходу		
	Завантаження даних	Обробка даних	Пересилка даних	Завантаження даних	Обробка даних	Пересилка даних
1	2	3	4	5	6	7
1	0	0.92	2.83	26.91	1.81	0
2	0	0.86	3	23.21	0.39	0
3	0	0.86	4.71	26.65	0.4	0
4	0	0.86	1.71	23.69	0.43	0
5	0	0.86	2.5	22.14	0.4	0
6	0	0.84	2.44	24.76	0.4	0
7	0	0.86	5.25	29.35	0.4	0
8	0	0.91	3.1	22.81	0.42	0

Закінчення табл. 3.15

1	2	3	4	5	6	7
9	0	0.84	3.72	18.28	0.4	0
10	0	0.88	5.29	20	0.41	0
Середнє значення	0	0.869	3.455	23.78	0.546	0

Таблиця 3.16

Результати дослідження для об’єму вхідних даних 1 210 000 записів

№ експерименту	Час отримання результатів, с					
	Використання вбудованих мовних засобів SQL Server			Використання класичного підходу		
	Завантаження даних	Обробка даних	Пересилка даних	Завантаження даних	Обробка даних	Пересилка даних
1	0	1.23	5.34	31.48	10.72	0
2	0	1.16	6.4	23.58	0.52	0
3	0	1.03	6.38	24.29	0.49	0
4	0	1.11	2.06	23.76	0.58	0
5	0	1.09	2.51	30.61	0.48	0
6	0	1.09	5.25	27.06	0.49	0
7	0	1.06	5.86	25.49	0.79	0
8	0	1.08	4.15	21.12	0.64	0
9	0	1.09	6.32	33.57	0.53	0
10	0	1.05	5.19	28.6	0.56	0
Середнє значення	0	1.099	4.946	26.956	1.58	0

Таблиця 3.17

Результати дослідження для об’єму вхідних даних 1 452 000 записів

№ експерименту	Час отримання результатів, с					
	Використання вбудованих мовних засобів SQL Server			Використання класичного підходу		
	Завантаження даних	Обробка даних	Пересилка даних	Завантаження даних	Обробка даних	Пересилка даних
1	2	3	4	5	6	7
1	0	1.39	6.54	45.72	18.73	0
2	0	1.22	5.9	23.83	1.1	0

Закінчення табл. 3.17

1	2	3	4	5	6	7
3	0	1.25	10.48	24.69	0.64	0
4	0	1.48	3.98	27.35	0.59	0
5	0	1.31	3.07	30.96	0.62	0
6	0	1.28	6.3	29.02	0.65	0
7	0	1.31	2.38	30.9	0.58	0
8	0	1.5	5.27	37.73	0.6	0
9	0	1.28	4.75	32.44	0.59	0
10	0	1.2	4.68	23.65	0.57	0
Середнє значення	0	1.322	5.335	30.629	2.467	0

На основі табл. 3.6 – 3.17 було проведено статистичну обробку даних шляхом визначення середнього часу виконання їх обробки на різних етапах, який було розраховано за методом визначення середнього арифметичного та сумарного часу, розрахованого за допомогою додавання всіх часових характеристик.

В результаті було отримано інформацію, представлену в табл. 3.18 – 3.21.

Таблиця 3.18

Результати дослідження після статистичної обробки (класичний підхід, локальний сервер)

Об’єм вхідних даних	Час отримання результатів, с		
	Використання класичного підходу		
	Завантаження даних	Обробка даних	Сумарний час, с
	Середній час, с	Середній час, с	
242 000	2.3	0.286	2.586
484 000	4.114	0.371	4.485
726 000	6.508	0.539	7.047
968 000	8.857	0.821	9.678
1 210 000	10.849	1.024	11.873
1 452 000	19.654	2.333	21.987

Таблиця 3.19

Результати дослідження після статистичної обробки (використання вбудованих мовних засобів SQL Server, локальний сервер)

Об’єм вхідних даних	Час отримання результатів, с		
	Використання вбудованих мовних засобів SQL Server		
	Обробка даних	Пересилка даних	Сумарний час, с
	Середній час, с	Середній час, с	
242 000	0.823	1.22	2.043
484 000	0.849	1.507	2.356
726 000	0.887	1.073	1.96
968 000	1.007	1.58	2.587
1 210 000	1.233	1.662	2.895
1 452 000	2.091	2.212	4.303

Для більш кращого розуміння даних статистичної обробки було побудовано графіки (рис. 3.17 – 3.21).

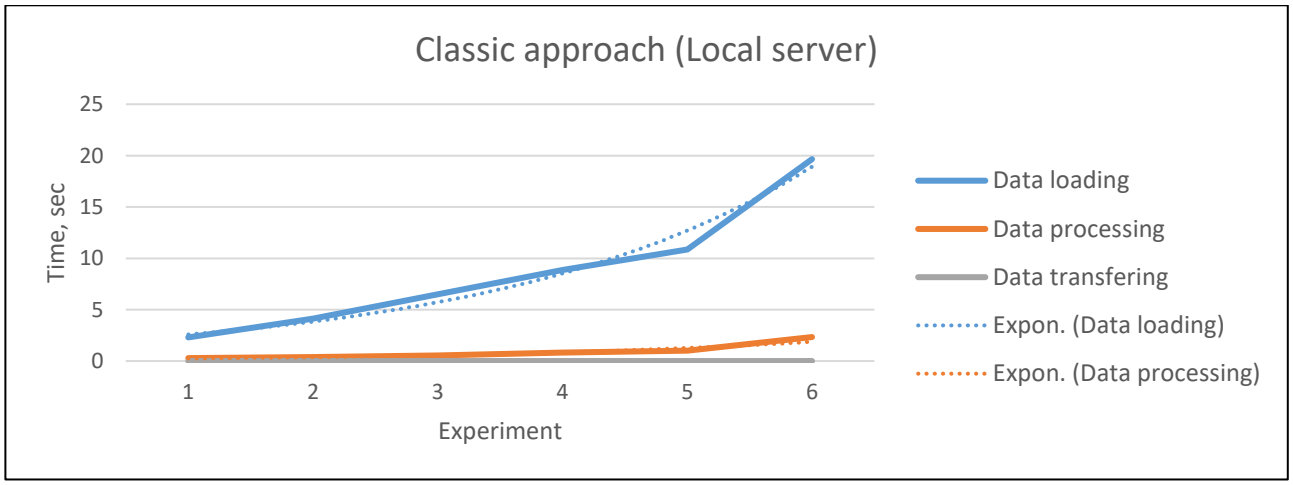

Рис. 3.17. Результати дослідження використання класичного способу роботи з даними (локальний сервер)

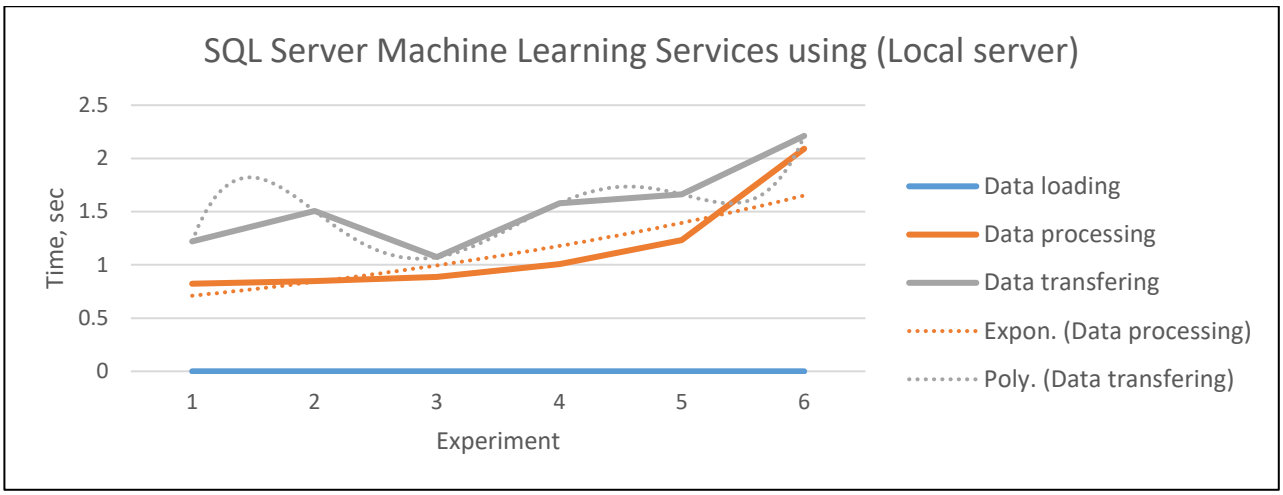

Рис. 3.18. Результати дослідження використання вбудованих мовних засобів SQL Server для роботи з даними(локальний сервер)

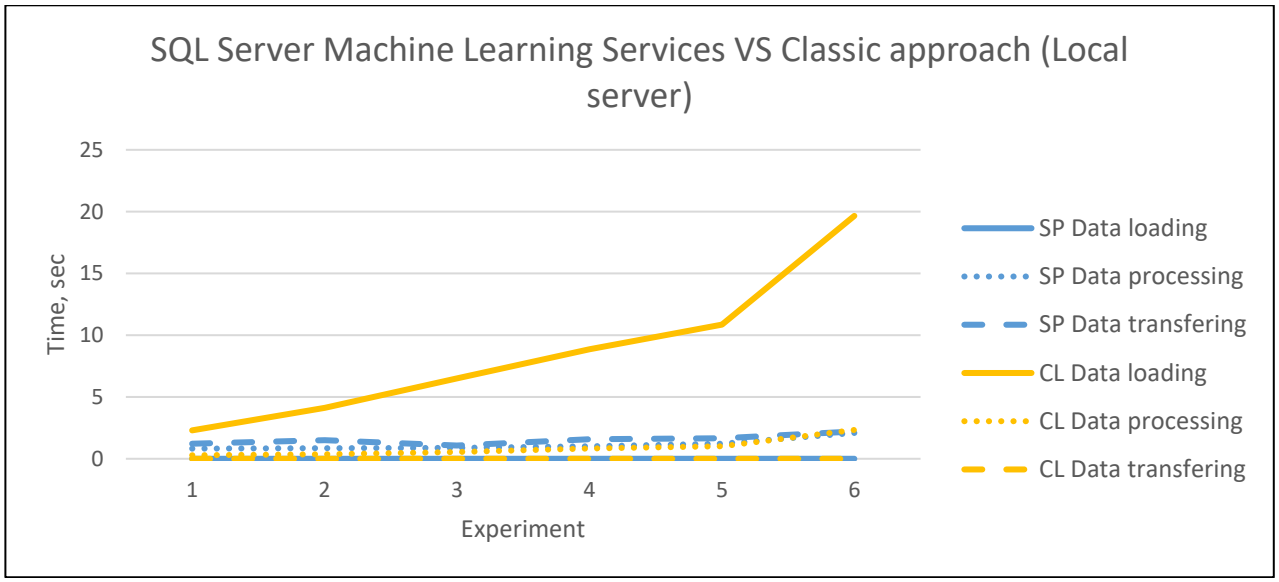

Рис. 3.19. Порівняння результатів, отриманих в ході проведення досліджень класичним способом та з використанням вбудованих мовних засобів SQL Server(локальний сервер)

Таблиця 3.20

Результати дослідження після статистичної обробки (класичний підхід, сервер Microsoft Azure)

Об'єм вхідних даних	Час отримання результатів, с		
	Використання класичного підходу		
	Завантаження даних	Обробка даних	Сумарний час, с
	Середній час, с	Середній час, с	
1	2	3	4
242 000	7.184	0.31	7.494
484 000	12.232	0.374	12.606

Закінчення табл. 3.20

1	2	3	4
726 000	15.725	0.483	16.208
968 000	23.78	0.546	24.326
1 210 000	26.956	1.58	28.536
1 452 000	30.629	2.467	33.096

Таблиця 3.21

Результати дослідження після статистичної обробки (використання вбудованих мовних засобів SQL Server, сервер Microsoft Azure)

Об'єм вхідних даних	Час отримання результатів, с		
	Використання вбудованих мовних засобів SQL Server		
	Обробка даних	Пересилка даних	Сумарний час, с
	Середній час, с	Середній час, с	
242 000	0.378	1.011	1.389
484 000	0.571	2.034	2.605
726 000	0.713	2.999	3.712
968 000	0.869	3.455	4.324
1 210 000	1.099	4.946	6.045
1 452 000	1.322	5.335	6.657

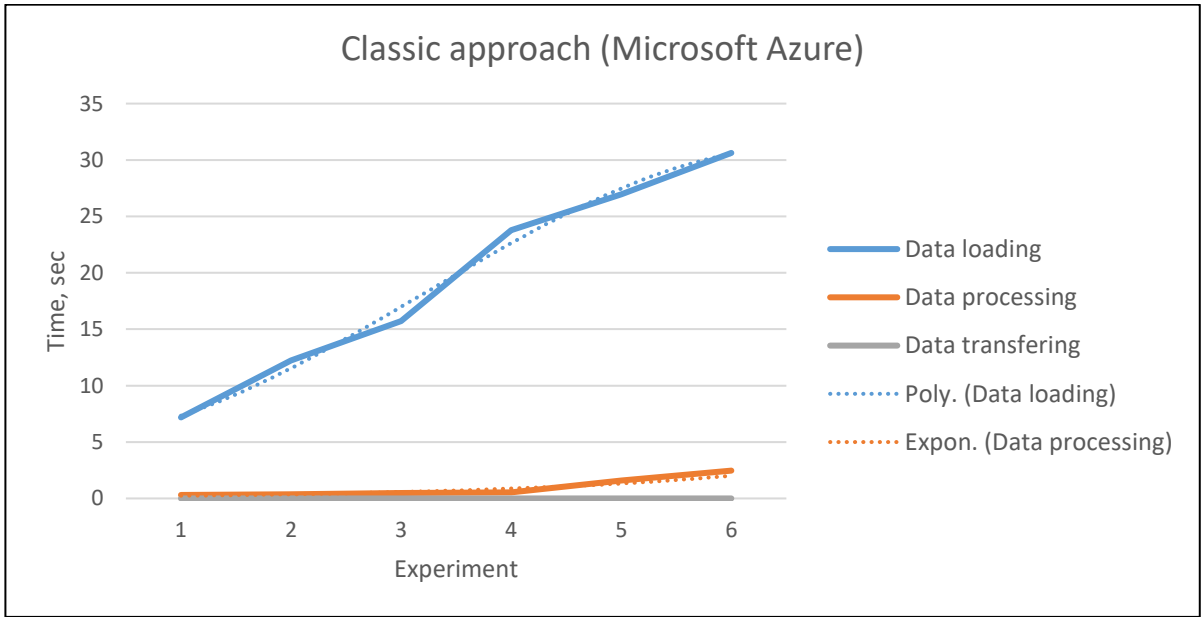

Рис. 3.20. Результати дослідження використання класичного способу роботи з даними(сервер Microsoft Azure)

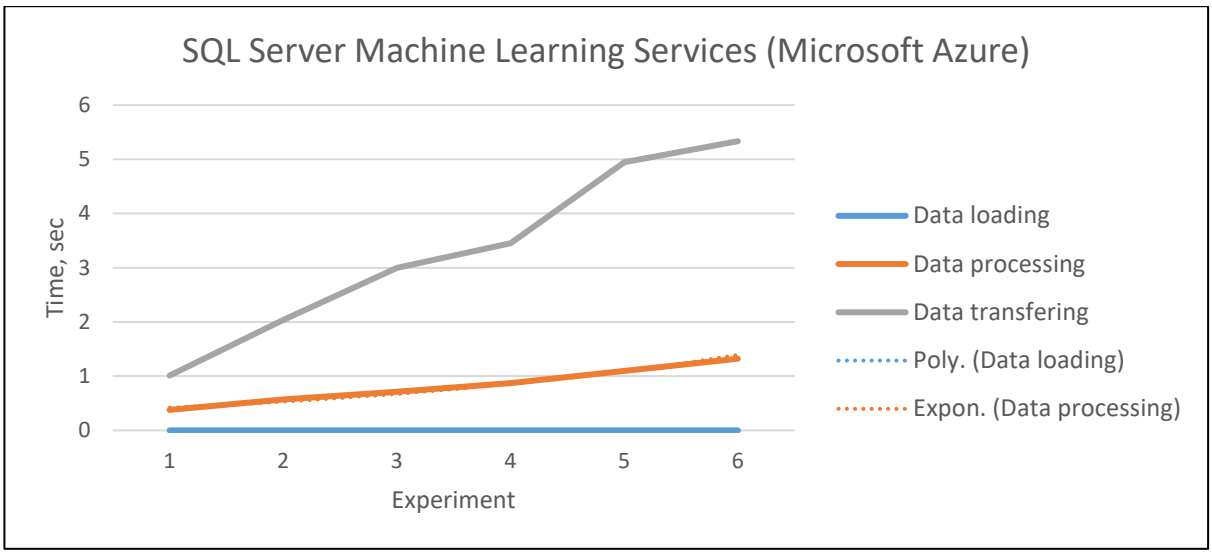

Рис. 3.21. Результати дослідження використання вбудованих мовних засобів SQL Server для роботи з даними(сервер Microsoft Azure)

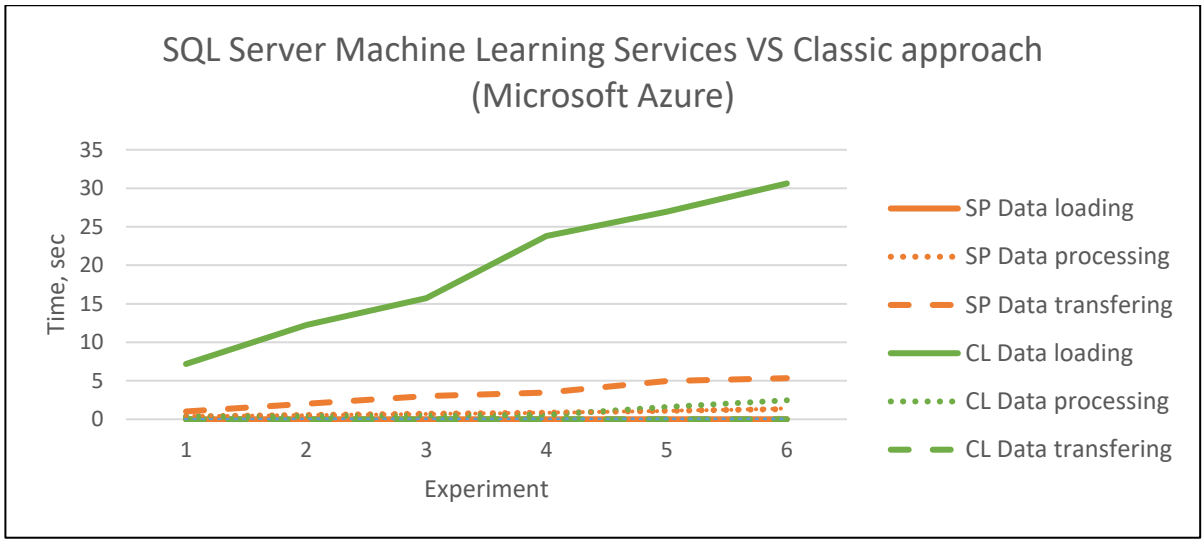

Рис. 3.22. Порівняння результатів, отриманих в ході проведення досліджень класичним способом та з використанням вбудованих мовних засобів SQL Server(сервер Microsoft Azure)

Через специфіку обробки даних різними способами маємо наступний результат: час завантаження даних буде завжди нульовим при використанні ML Services, на відміну від обробки даних класичним способом. В той же час при використанні класичного способу не потребується час на пересилку даних (дані були завантажені на початку роботи). Зважаючи на отриманий результат можемо порівняти час завантаження даних за класичного способу та час, необхідний на пересилку даних при використанні вбудованих мовних засобів SQL Server. Дане порівння дасть змогу оцінити ефективність зазначених способів обробки даних.

Аналізуючи табл. 3.18 – 3.21 та рис. 3.23 – 3.24 можна зробити висновок, що на пересилку даних витрачається менше часу, тож використання вбудованих мовних засобів SQL Server Machine Learning Services виявляється ефективнішим.

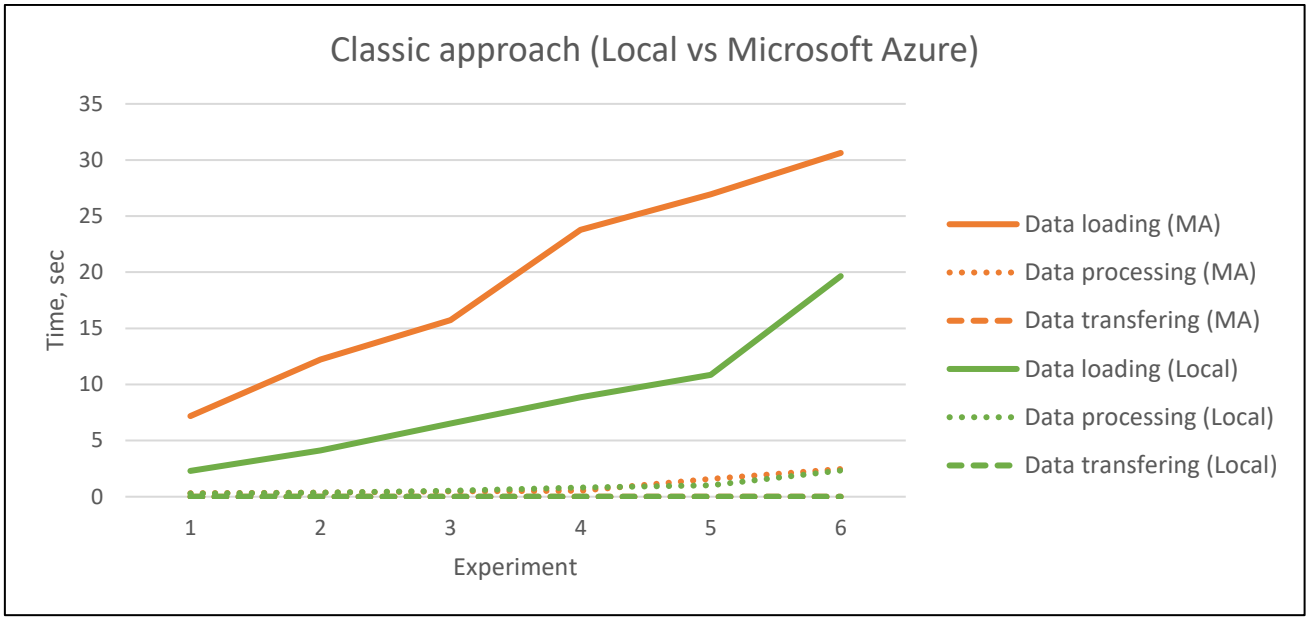

Рис. 3.23. Порівняння результатів, отриманих в ході проведення досліджень класичним способом на локальному сервері та в Microsoft Azure

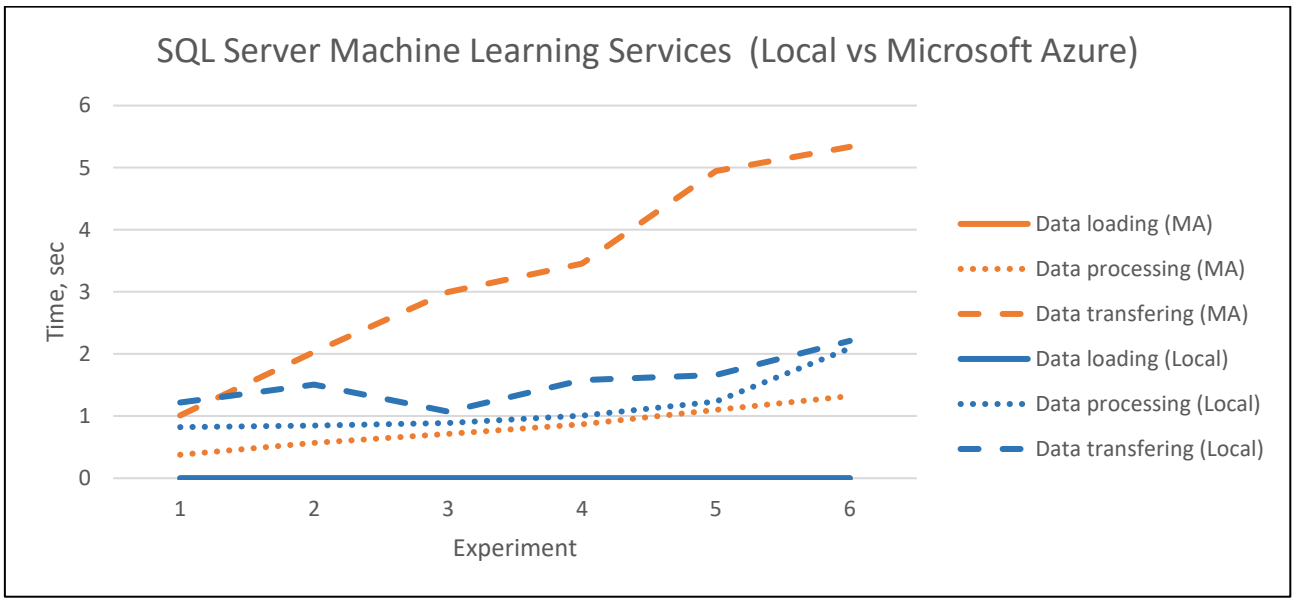

Рис. 3.24. Порівняння результатів, отриманих в ході проведення досліджень з використанням вбудованих мовних засобів SQL Server на локальному сервері та в Microsoft Azure

Якщо порівнювати час обробки даних за класичного способу та з використанням ML Servives, проте з використанням різних серверів (рис. 3.23 – 3.24), то можна побачити, що при роботі з локальним сервером отримані часові

показники менше, ніж про роботі с Microsoft Azure. Виходячи з цього можна зробити висновок, що краще використовувати локальний сервер або покращити характеристики сервера Microsoft Azure. Друге можна зробити, оформивши платну підписку для роботи з хмарною платформою.

Для кращого розуміння переваг використання SQL Server ML Services на основі даних табл. 3.18 – 3.21 було побудовано графіки, представлені на рис. 3.25 – 3.26.

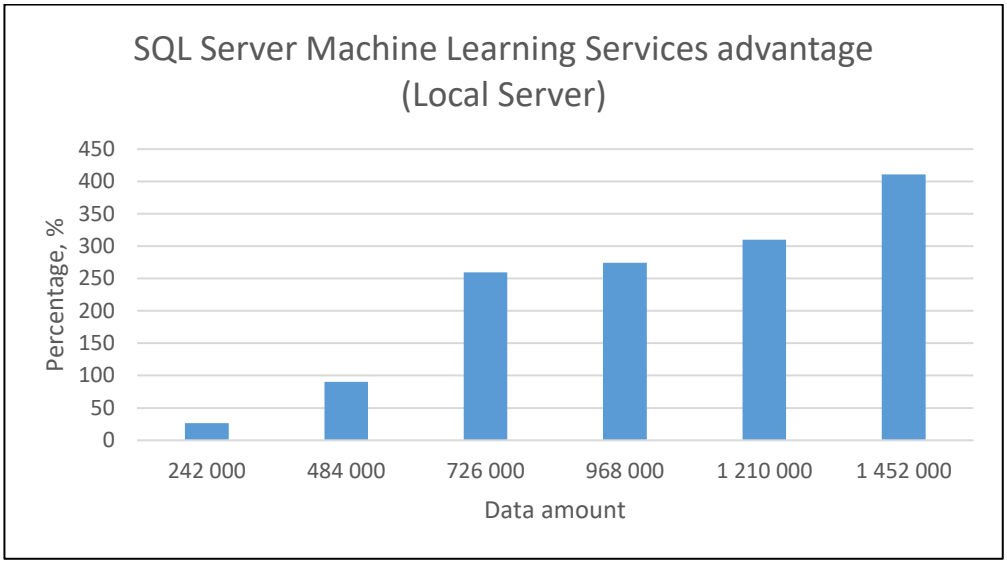

Рис. 3.25. Перевага використання вбудованих мовних засобів SQL Server Machine Learning Services у відсотковому відношенні часових показників (локальний сервер)

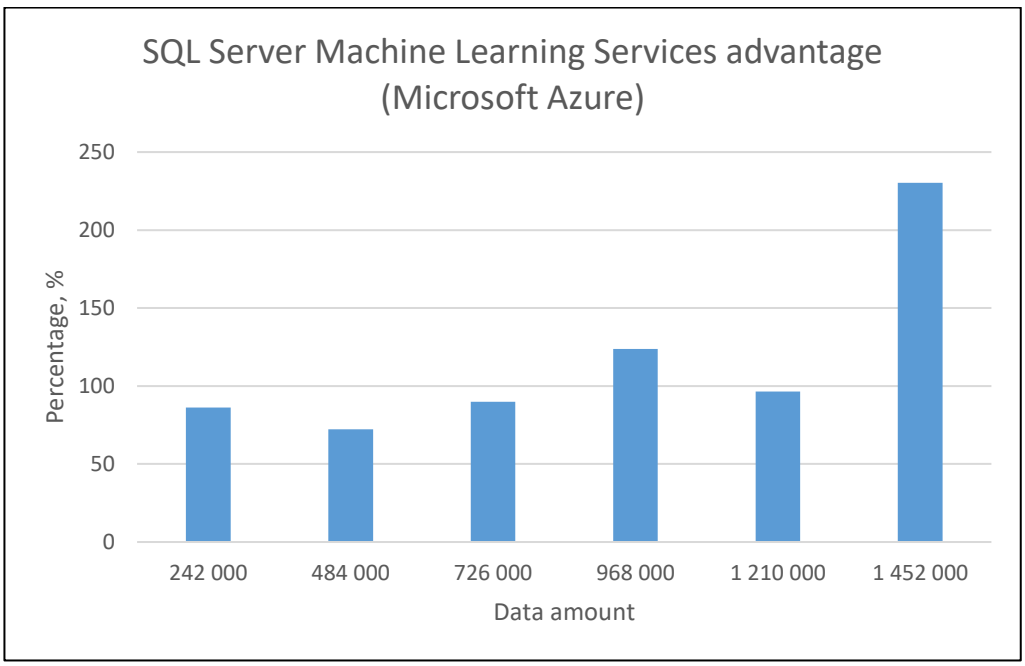

Рис. 3.26. Перевага використання вбудованих мовних засобів SQL Server Machine Learning Services у відсотковому відношенні часових показників (сервер Microsoft Azure)

Як бачимо, при збільшенні обсягу вхідних даних перевега використання SQL Server Machine Learning Services для обробки даних стає більш очевидною. В середньому застосування ML Services дає змогу зменшити час обробки даних в 2 – 4 рази для локально розташованого сервера, та в 2.5 – для сервера Microsoft Azure.

Також вважаю за потрібне відзначити, що при обробці даних (розташованих на локальному сервері) методом машинного навчання з використанням вбудованих мовних засобів SQL Server і об'ємом вхідних даних більше ніж 1 000 000 записів виникала помилка, представлена на рис. 3.27.

```
Msg 39004, Level 16, State 20, Line 1
A 'Python' script error occurred during execution of 'sp_execute_external_script' with HRESULT 0x80004004.
Msg 39019, Level 16, State 2, Line 1
An external script error occurred:

Error in execution.  Check the output for more information.
Traceback (most recent call last):
  File "<string>", line 5, in <module>
  File "C:\SQL-SQLEXPRESS-ExtensibilityData-PY\SQLEXPRESS01\C28BC659-61E2-4D15-B964-C7429CF65461\sqlindb.py", line 77, in transform
    lin_model.fit(train[columns], train[target])
  File "C:\Program Files\Microsoft SQL Server\MSSQL14.SQLEXPRESS\PYTHON_SERVICES\lib\site-packages\sklearn\linear_model\base.py", li
    copy=self.copy_X, sample_weight=sample_weight)
  File "C:\Program Files\Microsoft SQL Server\MSSQL14.SQLEXPRESS\PYTHON_SERVICES\lib\site-packages\sklearn\linear_model\base.py", li
    dtype=FLOAT_DTYPES)
  File "C:\Program Files\Microsoft SQL Server\MSSQL14.SQLEXPRESS\PYTHON_SERVICES\lib\site-packages\sklearn\utils\validation.py", lin
    array = np.array(array, dtype=dtype, order=order, copy=copy)
MemoryError
```

Рис. 3.27. Проблема обробки даних об'ємом більше 1 000 000

Провівши аналіз помилки було визначено, що причина полягає в тому, що за замовчуванням SQL Server обмежує виконання процесу Python до 20% пам'яті. Тож задля її виправлення було виконано команди, представлені на рис. 3.28, які дозволяють збільшити відсоток пам'яті, яка буде відводитися для виконання процесів Python до 50.

```
SELECT * FROM sys.resource_governor_external_resource_pools WHERE name = 'default'

ALTER EXTERNAL RESOURCE POOL "default"
WITH (MAX_MEMORY_PERCENT = 50);
GO
ALTER RESOURCE GOVERNOR RECONFICURE;
GO
```

Рис. 3.28. Команди для збільшення відсотку пам'яті для виконання процесів Python

За класичного способу обробки даних подібної помилки не виникало.

Варто зазначити, що процес компіляції коду з використанням збереженої процедури (див. рис. 3.11 – 3.12) відбувався дещо довше, ніж за класичного способу обробки даних, проте цей факт не впливав на швидкість завантаження

даних та час, необхідний на процес вирішення задачі за допомогою методу машинного навчання.

Одним із варіантів підвищення ефективності обробки даних може бути використання так званих баз даних і таблиць в пам’яті (in-memory databases and tables). In-Memory OLTP – це найкраща технологія, доступна в SQL Server та SQL Database для оптимізації продуктивності обробки транзакцій, передачі даних, завантаження даних та перехідних сценаріїв даних. Простими словами таблиця в пам’яті – це просто таблиця, яка має дві копії, одну в активній пам’яті і одну на диску. Оскільки пам'ять очищається під час перезапуску служб SQL, SQL Server зберігає фізичну копію таблиці, яку можна потім відновити.

Бази даних в пам'яті швидше, ніж бази даних, оптимізовані для використання дискових накопичувачів, оскільки доступ до диска повільніше, ніж доступ до пам'яті, а внутрішні алгоритми оптимізації простіше і виконують менше інструкцій ЦП. Доступ до даних з пам'яті виключає час на пошук при їх запиті, що забезпечує більш швидшу і передбачувану роботу, ніж при використанні дискового накопичувача.

Щоб мати можливість працювати з базою даних в пам’яті необхідно створити базу даних з оптимізованою для пам'яті файловою групою (рис. 3.29) та таблицю, яка буде зберігати дані в пам’яті (рис. 3.30). Схема створеної таблиці в пам’яті аналогічна тій таблиці, яку використовувалася в попердніх дослідженнях. Ключовим є використання параметру MEMORY_OPTIMIZED = ON.

```
CREATE DATABASE TUTORIALDB_InMemory
ALTER DATABASE TUTORIALDB_InMemory
ADD FILEGROUP TUTORIALDB_InMemory_Mem_optimized CONTAINS MEMORY_OPTIMIZED_DATA
ALTER DATABASE TUTORIALDB_InMemory
ADD FILE(name='TUTORIALDB_InMemory_Mem_optimized',
filename='c:\Temp\TUTORIALDB_InMemory_Mem_optimized')
TO FILEGROUP TUTORIALDB_InMemory_Mem_optimized
```

Рис. 3.29. Створення бази даних в пам’яті

```
CREATE TABLE [dbo].[rental_data_new_expInMemory](
       [Id] [int] INDEX [ixnc_Id] NONCLUSTERED  NOT NULL,
       [Year] [int] NULL,
       [Month] [int] NULL,
       [Day] [int] NULL,
       [RentalCount] [int] NULL,
       [WeekDay] [int] NULL,
       [Holiday] [int] NULL,
       [Snow] [int] NULL,
 CONSTRAINT [PK_rental_data_new_expInMemoryTable_Id_HashIDX]  PRIMARY KEY NONCLUSTERED
HASH ([Id]) WITH ( BUCKET_COUNT = 10000000)
) WITH ( MEMORY_OPTIMIZED = ON) GO
```

Рис. 3.30. Створення таблиці в пам’яті

Заповнення таблиці даними відбувалося аналогічно алгоритму, представленому раніше (див. рис. 3.7). Підрахунок часу заповнення таблиці пам’яті даними представлено на рис. 3.31. Він складає 87.52 секунди.

```
DECLARE @start DATETIME
DECLARE @end DATETIME
SET @start = GETDATE()
EXEC generateData
SET @end = GETDATE()

SELECT datediff(ms, @start, @end)
```

Рис. 3.31. Визначення часу заповнення таблиці в пам’яті даними

Для однакового набору даних, які зберігаються в різних за типом базах даних було виконано процедуру LinearRegressionPrediction(). Результати представлені на рис. 3.32 – 3.33.

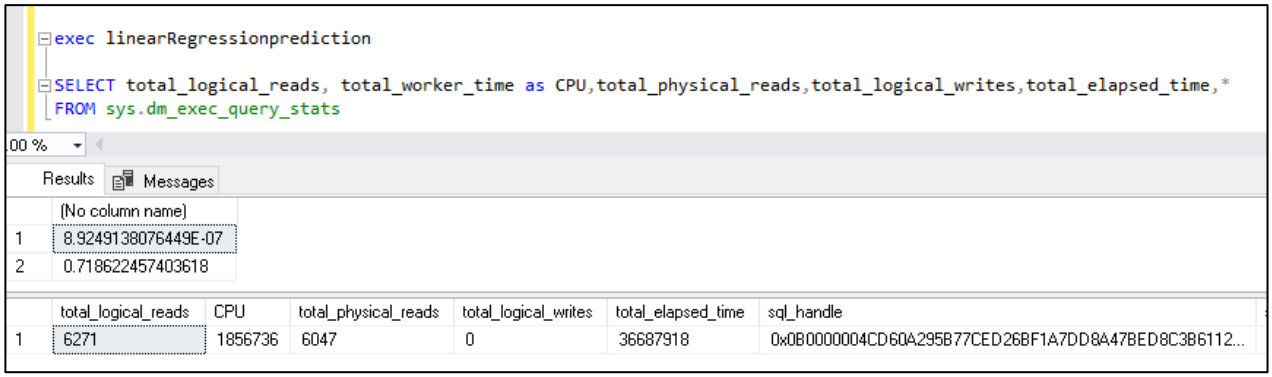

Рис. 3.32. Виконання збереженої процедури на основі даних таблиці на диску

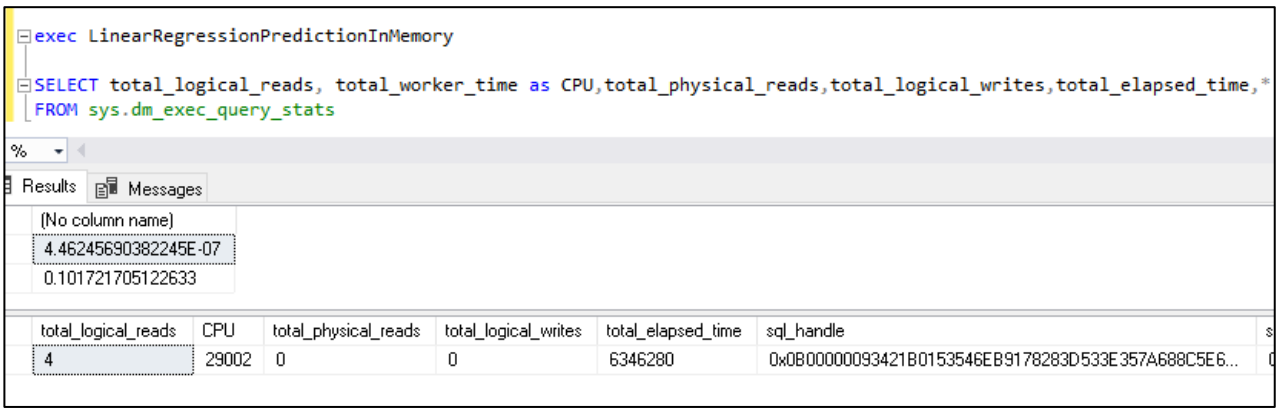

Рис. 3.33. Виконання збереженої процедури на основі даних таблиці в пам’яті

Основними показниками, які характеризують ефективність використання різного типу таблиць є:

- total_logical_reads – загальна кількість логічних читань, виконаних при виконанні запиту;

- total_worker_time – загальна кількість процесорного часу (в мікросекундах, але з точністю до мілісекунд), яка витрачалася на виконання запиту з моменту його компіляції;
- total_physical_reads – загальна кількість фізичних зчитувань, виконаних при виконанні цього запиту з моменту його компіляції;
- total_logical_writes – загальна кількість логічних записів, виконаних при виконанні цього запиту з моменту його компіляції;
- total_elapsed_time – загальний час, що пройшов у мікросекундах (але з точністю до мілісекунд) для завершених виконання цього запиту.

Як бачимо, для результатів дослідження, яке проводилося для таблиці в пам’яті всі показники значно нижче тих, які було отримані в результаті дослідження з вхідними даними, які зберігаються в таблиці на диску. Даний факт свідчить про кращу ефективність роботи з таблицями в пам’яті.

ВИСНОВКИ

В результаті виконання дипломної роботи було проведено змістовний опис і аналіз предметної області, ознайомлення з мовами програмування, які використовуються для здійснення машинного навчання, а саме R та Python, проведено огляд систем, що реалізують методи машинного навчання.

Аналізуючи всі отримані результати можна зробити висновок, що обробка даних на сервері засобами SQL Server Machine Learning Services виявилась значно ефективнішою, ніж використання класичного підходу роботи з даними. Такий результат обумовлюється наступними причинами:

- не витрачався додатково час на завантаження даних (оскільки робота з даними відбувалася на тому ж сервері, де вони зберігаються);
- процес обробки даних шляхом використання методу машинного навчання відбувався значно швидше у порівнянні з класичним способом;
- SQL Server надає можливість створення збережених процедур з вихідним кодом Python, що сприяє підвищенню продуктивності роботи системи, дозволяє багаторазове використання процедур різними користувачами, є більш безпечним;
- можливий запуск паралельних робіт на сервері для підвищення ефективності роботи з даними (параметр parallel = 1 команди sp_execute_external_script);
- можливість роботи з таблицями в пам’яті.

Також було виявлено, що ефективність застосування SQL Server Machine Learning Services ставала більш значимою при збільшенні даних, які використовувалися для дослідження, що є вагомим аргументом, оскільки зазвичай для аналізу використовуються мільйони записів, які потрібно швидко обробити та показати результат.

Пришвидшити процес обробки даних на сервері можна, відредагуваши певні характеристики сервера – збільшити кількість процесорів, обсяг фізичної та віртуальної пам’яті, використовувати останні версії SQL Server.

Машинне навчання є важливою технологією майбутнього, яка надасть змогу значно зменшити використання людських ресурсів та спростити вирішення задач шляхом їх автоматизації, прогнозування роботи систем та уникнення небажаних результатів. Проте необхідно ретельно підходити до вибору алгоритму машинного навчання та способу його реалізації, аби отримання прогнозованих даних відбувалося найбільш ефективно.

Результати проведених досліджень можуть бути впроваджені в в багатьох сферах: виробництво, транспорті системи, медицина, освіта і т.д., що дозволить оптимізувати процес роботи з даними та працювати в режимі реального часу. Окрім того, вони будть корисними для викладачів навчальних закладів, дисципліни яких пов’язані з процесами машинного навчання, що дозволить покращити навчальний процес за рахунок збільшення кількості вирішуваних задач.

Було визначено найбільш ефективний спосіб обробки даних, використання якого дозволяє зменшити час, необхідний на отримання кінцевого результату вирішення задачі засобами машинного навчання в 2 – 4 рази для локально розташованого сервера та в 2.5 – для сервера Microsoft Azure. Ефективність застосування ML Services на сервері Microsoft Azure простежувалися, починаючи з використання даних, об’ємом в 1.5 млн.

За темою дослідження були опубліковані тези [15].

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Аксютина Е. М., Белов Ю. С. Обзор архитектур и методов машинного обучения для анализа больших данных // Электронный журнал: наука, техника и образование. 2016. №1 (5). С. 132–139. [Электронный ресурс] – Режим доступа: http://nto-journal.ru/uploads/articles/0b9bd6d9833003ed0d6f9bb16fab81f1.pdf.

2. Белов Ю. С., Козина А. В., Гришунов С. С. – Применение критерия «сигнал/шум» для определения эффективности методов машинного обучения // Известия ТулГУ. Технические науки. 2018. Вып. 12 С. 292–295.

3. Боровский А. А. — Перспективы применения технологий машинного обучения к обработке больших массивов исторических данных // Кибернетика и программирование. – 2015. – № 1. – С. 77 – 114. DOI: 10.7256/2306-4196.2015.1.13730 [Электронный ресурс] – Режим доступа: https://nbpublish.com/library_read_article.php?id=13730.

4. Горяинов А. Н. Машинное обучение в логистических и транспортных системах [Электронный ресурс] // Україна – ЄС: проблеми наукової та галузевої інтеграції: Матер V Всеукр. заоч. наук.-пр. конф. «Україна – ЄС: проблеми наукової та галузевої інтеграції» (м. Харків, 31 січня – 01 лютого 2020 року) / Наукове партнерство «Центр наукових технологій». – Харків: НП «ЦНТ», 2020. – С. 34–42 (72 с.)

5. Гусев А. В. – Перспективы нейронных сетей и глубокого машинного обучения в создании решений для здравоохранения // Искусственный интеллект в здравоохранении. 2017, №3 С. 92–105.

6. Жуков Д. А., Клячкин В. Н. – Задачи обеспечения эффективности машинного обучения при диагностике технических объектов // Электронный журнал: Современные проблемы проектирования, производства и эксплуатации радиотехнических систем. 2016. № 10. С. 172–174.

7. Иванов О. Ю. – Использование методов машинного обучения для повышения эффективности систем управления базами данных [Электронный ресурс] – Режим доступа: http://www.machinelearning.ru/wiki/images/2/25/2016_417_IvanovOYu.pdf.

8. Как Big Data с Machine Learning борются с пробками и улучшают дороги [Электронный ресурс] – Режим доступа: https://www.bigdataschool.ru/blog/big-data-machine-learning-iot-transport-traffic.html.

9. Кафедра інформаційних систем [Електронний ресурс] – Режим доступу: http://www.is.hneu.edu.ua/.

10. Кафтанников И. Л., Парасич А. В. – Проблемы формирования обучающей выборки в задачах машинного обучения. // Вестник ЮУрГУ. Серия «Компьютерные технологии, управление, радиоэлектроника». 2016. Т. 16, № 3. С. 15–24.

11. Кондрашов Ю. Н. – Анализ данных и машинное обучение на платформе MS SQL Server [Электронный ресурс] – Режим доступа: https://aldebaran.ru/author/n_kondrashov_yu/kniga_analiz_dannyih_i_mashinnoe_obuchenie_na_/.

12. Коротеев М. В. – Обзор некоторых современных тенденций в технологии машинного обучения // Технологии искусственного интеллекта в менеджменте. – 2018. № 1, С. 26–35. DOI: 10.26425/2658-3445-2018-1-26-35.

13. Краснянский М. Н., Обухов А. Д., Соломатина Е. М., Воякина А. А., – Сравнительный анализ методов машинного обучения для решения задачи классификации документов научно-образовательного учреждения – 2018 // ВЕСТНИК ВГУ, СЕРИЯ: СИСТЕМНЫЙ АНАЛИЗ И ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ, 2018, № 3 С. 1038–1082.

14. Мартынова Ю. А. – Выбор источника финансирования методами машинного обучения // Вопросы инновационной экономики. – 2019. – Том 9. – № 3. – С.1037–1048. doi:10.18334/vinec.9.3.41177.

15. Матеріали Міжнародної науково-практичної конференції «Інформаційні технології та системи»: тези доповідей, 9-10 квітня 2020 р. – Х.: ХНЕУ імені Семена Кузнеця, 2020. – 60 с.

16. Машинное обучение: как оно применяется в жизни [Электронный ресурс]. – Режим доступа: https://rb.ru/opinion/mashinnoe-obuchenie/.

17. Методы статистической обработки данных [Электронный ресурс]. – Режим доступа: https://studfile.net/preview/3859470/page:12/.

18. Нигматуллин В. Р., Руднев Н. А. – Использование методов машинного обучения и искусственного интеллекта в химической технологии. // Сетевое издание «Нефтегазовое дело». 2019. №4 С. 243–268.

19. Обзор методов статистического анализа данных [Электронный ресурс] – Режим доступа: http://statlab.kubsu.ru/node/4.

20. Орельен Ж. – Прикладное машинное обучение с помощью Scikit-Learn и TensorFlow [Электронный ресурс] – Режим доступа: https://ru.pdfdrive.com/%D0%9F%D1%80%D0%B8%D0%BA%D0%BB%D0%B0%D0%B4%D0%BD%D0%BE%D0%B5-

%D0%BC%D0%B0%D1%88%D0%B8%D0%BD%D0%BD%D0%BE%D0%B5-%D0%BE%D0%B1%D1%83%D1%87%D0%B5%D0%BD%D0%B8%D0%B5-%D1%81-%D0%BF%D0%BE%D0%BC%D0%BE%D1%89%D1%8C%D1%8E-scikit-learn-%D0%B8-tensorflow-e188679436.html.

21. Резервна копія бази даних дослідження [Електронний ресурс] – Режим доступу: https://sqlchoice.blob.core.windows.net/sqlchoice/static/TutorialDB.bak.

22. Сайт персональних навчальних систем Харківського національного економічного університету імені Семена Кузнеця [Електронний ресурс] – Режим доступу: https://pns.hneu.edu.ua/.

23. Статистическая обработка результатов исследования [Электронный ресурс] – Режим доступа: http://cito-web.yspu.org/link1/metod/met125/node23.html.

24. Суперкомпьютер IBM Watson: революция в диагностике и терапии рака [Электронный ресурс] – Режим доступа: http://www.diakonlab.ru/vse_novosti/industry_news/superkompyuter_ibm_watson_revolyuciya_v_diagnostike_i_terapii_raka/.

25. Таблица значений критерия Стьюдента (t-критерия) [Электронный ресурс] – Режим доступа: https://www.matematicus.ru/teoriya-veroyatnosti/tablitsy/tablitsa-znachenij-kriteriya-styudenta-t-kriteriya.

26. Трифонов Т.В., Андреев И.Е., Глазкова А.В. – Сравнение эффективности методов машнного обучения для тоновой классификации текстов. // Конференция: МАТЕМАТИЧЕСКОЕ И ИНФОРМАЦИОННОЕ МОДЕЛИРОВАНИЕ 2019.

27. Федько В. В. – Аналіз даних в SQL Server засобами Python // Збірник наукових праць Харківського національного університету Повітряних Сил, 2018, №2 (5б) С. 99–104 DOI: 10.30748/zhups.2018.56.14.

28. Франсуа Шолле – Глубокое обучение на Python [Электронный ресурс] – Режим доступа: https://ru.pdfdrive.com/%D0%93%D0%BB%D1%83%D0%B1%D0%BE%D0%BA%D0%BE%D0%B5-%D0%BE%D0%B1%D1%83%D1%87%D0%B5%D0%BD%D0%B8%D0%B5-%D0%BD%D0%B0-python-e184715133.html.

29. Хомутов Н. Ю. – Методы повышения эффективности моделей машинного обучения, основанные на различных принципах снижения размерности [Электронный ресурс] – Режим доступа: http://www.machinelearning.ru/wiki/images/9/9d/2017_617_KhomutovNY.pdf.

30. Чибирова М. Э. – Анализ данных и регрессионное моделирование с применением языков программирования Python и R // Научные записки молодых исследователей № 2/2019 С. 37–45.

31. Шелухин О. И., Симонян А. Г., Ванюшина А. В. – Влияние структуры обучающей выборки на эффективность классификации приложений трафика методами машинного обучения // T-Comm: Телекоммуникации и транспорт. 2017. Том 11. №2. С. 25–31.

32. Шибайкин С. Д., Никулин В. В., Аббакумов А. А. – Анализ применения методов машинного обучения компьютерных систем для повышения защищенности от мошеннических текстов // Вестник Астраханского государственного технического университета. Серия: Управление, вычислительная техника и информатика. 2020. № 1. С. 29–40. DOI: 10.24143/2072-9502-2020-1-29-40.

33. About Python [Electronic resource] – Access mode: https://www.python.org/about/.

34. Alison DeNisco Rayome – R vs. Python: Which is a better programming language for data science? [Electronic resource]. – Access mode: https://www.techrepublic.com/article/r-vs-python-which-is-a-better-programming-language-for-data-science/.

35. Baron Schwartz – 10 essential performance tips for MySQL. [Electronic resource]. – Access mode: https://www.infoworld.com/article/3210905/10-essential-performance-tips-for-mysql.html.

36. Brien Posey – A closer look at Python-SQL Server 2017 integration [Electronic resource]. – Access mode: https://searchsqlserver.techtarget.com/tip/A-closer-look-at-Python-SQL-Server-2017-integration.

37. Common ML Problems [Electronic resource]. – Access mode: https://developers.google.com/machine-learning/problem-framing/cases.

38. Georg Thomas – AI and ML: Why have machine learning in SQL Server at all [Electronic resource]. – Access mode: https://blog.greglow.com/2018/05/25/ai-and-ml-why-have-machine-learning-in-sql-server-at-all/.

39. Google Trends – Comparison of search Data Science, Data Mining and Machine Learning [Electronic resource]. – Access mode: https://trends.google.com/trends/explore?date=2010-01-01%202020-12-07&q=Data%20Science,Data%20mining,Machine%20learning.

40. How to select algorithms for Azure Machine Learning [Electronic resource] – Access mode: https://docs.microsoft.com/ru-ru/azure/machine-learning/how-to-select-algorithms.

41. Machine Learning Systems [Electronic resource] – Access mode: https://www.manning.com/books/machine-learning-systems.

42. Machine Learning: The New Proving Ground for Competitive Advantage [Electronic resource] – Access mode: https://s3.amazonaws.com/files.technologyreview.com/whitepapers/MITTR_GoogleforWork_Survey.pdf.

43. Matt Watson – SQL Performance Tuning: 7 Practical Tips for Developers [Electronic resource]. – Access mode: https://stackify.com/performance-tuning-in-sql-server-find-slow-queries.

44. Microsoft Azure Portal [Electronic resource] – Access mode: https://portal.azure.com/.

45. Peter Zaitsev – MySQL Query Performance Troubleshooting: Resource-Based Approach [Electronic resource]. – Access mode: https://www.percona.com/blog/2020/07/15/mysql-query-performance-troubleshooting-resource-based-approach/.

46. Python tutorial: Predict ski rental with linear regression with SQL machine learning [Electronic resource]. – Access mode: https://docs.microsoft.com/en-us/sql/machine-learning/tutorials/python-ski-rental-linear-regression?view=sql-server-ver15.

47. Reena Shaw – The 10 Best Machine Learning Algorithms for Data Science Beginners [Electronic resource]. – Access mode: https://www.dataquest.io/blog/top-10-machine-learning-algorithms-for-beginners/.

48. The Limitations of Machine Learning [Electronic resource] – Access mode: https://towardsdatascience.com/the-limitations-of-machine-learning-a00e0c3040c6.

49. Top 10 real-life examples of Machine Learning [Electronic resource] – Access mode: https://bigdata-madesimple.com/top-10-real-life-examples-of-machine-learning/.

50. R vs. Python: Which is a better programming language for data science? [Electronic resource] – Access mode: https://www.techrepublic.com/article/r-vs-python-which-is-a-better-programming-language-for-data-science/.

51. Watson. IBM [Electronic resource] – Access mode: https://www.ibm.com/ru/watson/.

52. What is SQL Management Studio (SSMS) [Electronic resource] – Access mode: https://docs.microsoft.com/en-us/sql/ssms/sql-server-management-studio-ssms?view=sql-server-ver15.

53. What is SQL Server Machine Learning Services (Python and R)? [Electronic resource] – Access mode: https://docs.microsoft.com/en-us/sql/advanced-analytics/what-is-sql-server-machine-learning?view=sql-server-ver15.

54. What is R? [Electronic resource] – Access mode: https://www.r-project.org/about.html.